Learning height for top-down grasps with the DIGIT sensor

Thais Bernardi, Yoann Fleytoux, Jean-Baptiste Mouret, Serena Ivaldi

▶ **To cite this version:**

Thais Bernardi, Yoann Fleytoux, Jean-Baptiste Mouret, Serena Ivaldi. Learning height for top-down grasps with the DIGIT sensor. IEEE RAS Int. Conf. Robotics and Automation (ICRA), 2023, London, United Kingdom. hal-03989704

HAL Id: hal-03989704
https://hal.science/hal-03989704v1

Submitted on 14 Feb 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Learning height for top-down grasps with the DIGIT sensor

Thais Bernardi[1,2,3,*], Yoann Fleytoux[1,*], Jean-Baptiste Mouret[1], Serena Ivaldi[1]

***Abstract*— We address the problem of grasping unknown objects identified from top-down images with a parallel gripper. When no object 3D model is available, the state-of-the-art grasp generators identify the best candidate locations for planar grasps using the RGBD image. However, while they generate the Cartesian location and orientation of the gripper, the height of the grasp center is often determined by heuristics based on the highest point in the depth map, which leads to unsuccessful grasps when the objects are not thick, or have transparencies or curved shapes. In this paper, we propose to learn a regressor that predicts the best grasp height based from the image. We train this regressor with a dataset that is automatically acquired thanks to the DIGIT optical tactile sensors, which can evaluate grasp success and stability. Using our predictor, the grasping success is improved by 6% for all objects, by 16% on average on difficult objects, and by 40% for objects that are notably very difficult to grasp (e.g., transparent, curved, thin).**

I. Introduction

We consider the case of a robotic manipulator equipped with a parallel gripper and a RGBD camera mounted at the end-effector, which needs to grasp unknown objects (i.e., the object 3D model is unknown) using a 4D grasp, also known as *top-grasp*. A 4D grasp is generally defined as the (x, y, z, θ), where (x, y, z) are the Cartesian coordinates of the grasp center and θ is the yaw orientation of the gripper on the planar surface. In the last years, several grasp generators based on images have been proposed, and they proved successful in finding 4D grasps. For example, Dex-Net [1] generates grasps candidates for an object and ranks them according to a grasp quality functions. GG-CNN [2] and GR-ConvNet [3] are other data-driven methods that generate pixel-wise grasp affordance maps.

However, while the x, y component of the grasp center are set, the grasp height z is often determined by heuristics based on the highest height determined by the point cloud. Widely used grasp representations such as oriented rectangles [4], [5] or pixel-level grasp maps [6], [7], [8], [9] take into account the center of the grasp, the distance between two jaws, the size of the gripper and its orientation, but do not encode the grasp height z. Heuristics for the height work most of the time, especially when objects are thick or full, but they are often inadequate when objects have thin parts, transparent parts, holes or curved shapes, because of the intrinsic error in the height estimation from the point cloud or simply the absence of knowledge on what is behind the object's surface, which requires the object 3D model. Figure 1 shows an example of grasp failures in this sense: two state-of-the art grasp generators, Dex-Net [1] and GPD [10], fail because of the grasp height (in one case too low, in the other too high, in both cases the gripper does not properly touch the object while closing).

Fig. 1. Typical problem of planar grasps: the grasp center height is often set with an heuristic based on the highest surface. If the object's shape is unknown, this can lead to unsuccessful grasps. On the left, a depth gray-scale image of an object generated from the point cloud captured by the RGBD camera mounted on the robot's end effector. On the center and on the right, two top-down grasps generated by Dex-Net [1] and GPD [10], two classical grasp generators. The Dex-Net grasp in the center is too low, the object is not grasped by the gripper jaws, while the GPD grasp on the right is too high to be successful. Without prior knowledge of the object and how thick it is, a wrong height prediction can lead to failure.

In this paper, we address this problem by proposing to predict the best grasp height from the RGBD image, using a regressor that is previously trained with a dataset of the best grasp height from several grasp candidates. We posit that the "best grasp height" can be automatically learned from a sensor-driven data collection, where the robot attempts different heights candidates $(z_0, z_1, \dots)$ on the same 3D candidate grasp (x, y, θ), using the contact information provided by a tactile sensor. In our work, we used DIGIT, an optical tactile sensor that provides a contact ellipsoid measure, which we found experimentally to be a good predictor of grasp stability and grasp success. Our experiments with several objects show that the predicted grasp height slightly improves the grasp success, by 6% overall on all the objects, but most importantly it enables to successfully grasp objects that were otherwise failing, with improvement up to 40%.

The main contributions in this paper are: i) the characterization of the DIGIT sensor, with several experiments to investigate whether its output can be used for stable grasp prediction (note: we found that its output does not relate to the contact force, which is a useful information); ii) the experimental validation of the optical contact ellipsoid as a

*These authors contributed equally to this work.

This work was partially funded by the European projects HEAP and EUROBIN, and the Creativ'Lab facility in Loria.

[1] University of Lorraine, CNRS, Inria, Loria, F-54000, France. name.surname@inria.fr

[2] Telecom Nancy - Université de Lorraine, France.

[3] Federal University of Technology – Paraná (UTFPR), Brazil

predictor of grasp stability and success; iii) the method to identify the best candidate grasp using DIGIT's measures and then train a grasp height regressor based on latent grasp patches. All our findings are the result of several real-world experiments with the Franka robot equipped with the standard gripper, mounting two Digit sensors (acquired from GelSight).

II. Related Work

Tactile sensors have become widely used in robotics to extract relevant information about the manipulated objects [11], such as contact wrench, contact area and location, texture [12]. Tactile sensors can have a different physical nature [11]: for example, they can be capacitive, piezoresistive, piezoelectric, inductive, opto-electric. A wide range of materials can be used in their fabrication [13], such as substrate materials, active materials, or flexible electrodes, which makes them useful in a range of applications, from robotics to healthcare and even surgery.

In this context, vision-based sensors such as the DIGIT [14] have the initial idea of correlating contact force with the changes in the surface of an elastometer. Deformable elastometers have been developed as a medium of contact for this use, notably GelSight [15] (used in the upper part of the DIGIT sensor), which measures high-resolution geometry that can be used to infer local normal and shear force. The DIGIT is designed to be used naturally as a "fingertip" mounted on existing grippers.

In [14], the authors describe the design and manufacturing process of DIGIT, the analysis of its properties, and use it in a task that involves learning to manipulate small objects from visual images. With the sensor, the PyTouch library was developed and released as open-source [16]: the library can operate on real-world data to provide touch detection, slip and object pose estimations. A simulator for vision-based tactile sensing, supporting DIGIT, is also available [17]. Its properties make it suitable for precise manipulation and grasping, even of soft objects [18].

An analogous sensor, GelSlim 3.0 [19], was used to reconstruct 3D geometry, estimate the spatial distribution of 3D contact forces, and detect slips.

While optical tactile sensors are appealing for data-driven learning methods (especially deep-learning methods that are efficient in processing images), the visual measure is unusual for traditional grasping methods to assess stable grasps, which usually require force information – see [20] for a review on grasping methods and performance. To predict grasp success, in [21] the authors use reasoning in the wrench space of the task; to evaluate the effectiveness of adding tactile feedback to the analytic grasp success prediction, the study relies on the tactile feedback to alleviate contact placement uncertainties. To the best of our knowledge, the grasping performance improvement brought by DIGIT/GelSight is only empirically proved by [22]. The authors use an end-to-end learning approach for predicting grasp outcome, training deep neural-networks for vision-based, touch-based and combined vision and touch-based grasp outcome prediction. The subsequent work, [23] uses the GelSight in a grasping robot, employing a deep, multimodal convolutional network that predicts the outcome of a candidate grasp adjustment, and using the raw visuo-tactile data, iteratively selects the most promising actions. The operating performance of the sensors, however, is not fully known. In particular, it is unknown whether the optical measure of the sensor is correlated with the contact wrench or contact normal force, an information that is extremely important to assess the grasp stability and success using traditional measures based on force. For this reason, in this paper, we conducted an experimental characterization of the behavior of DIGIT, before using it for assessing grasping performance.

III. MATERIALS AND METHOD

A. Robotics setup

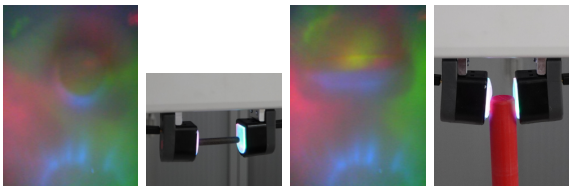
Our experimental setup consists of a Franka Emika Panda 7 DOF robot arm, equipped with an Intel RealSense D415 Depth Camera mounted on the standard gripper, hosting two DIGIT tactile sensor (Fig. 1).We use the Expanding Space Tree (ESTk) [24] motion planner from MoveIt to plan the trajectories for the robot, and libfranka 5.0 library to control the arm.

B. DIGIT sensor characterization

DIGIT [14] is a vision-based tactile sensor: it consists of an acrylic window that is, in the outermost side, covered by an elastometer, and in the internal side it is lighted by three colored LEDs (red, green and blue). A CMOS camera, inside, captures the acrylic window as an image, which is the sensor's output. When an object pushes on the surface of the sensor, the acrylic window deforms and the effect of the object is noticeable in the output image. All DIGITS used in this work used the reflective elastometer; they were set to the default configurations (image acquisition set at 60 *fps*, illumination intensity at maximum), except when stated otherwise. Some examples of sensor's outputs for different objects are shown in Fig. 2 and in the Video attachment[1]. Flat surface objects are the least perceived by DIGIT, and not obvious to recognize by the human eye in the output image (Fig. 2(d)), opposing to small objects or objects with edges (a and b). Curved objects, when grabbed anywhere other then the center, also are less perceptible (c).

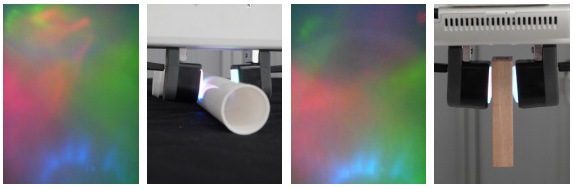
To quantify the DIGIT response of any contact with the sensor, we consider two metrics. The first is δP, i.e., the multi-channel (since the sensor has RGB channels) sum of the pixel by pixel difference between the current image output P_{image} and a baseline image $P_{no-contact}$, obtained when there is no contact: $\delta P = \frac{1}{n*m}\sum_{i=0}^{n}\sum_{j=0}^{m}(p_{i,j_{image}} - p_{i,j_{no-contact}})$, where p is the value of each pixel in the $[0, 255]$ range and images are of size $n \times m$. The second is the area of the contact surface of the object with the sensor. The contact area, often difficult to identify with the naked eye, is computed by the *ContactArea* function from the *PyTouch* library [16], which finds the ellipse enclosing the contact

[1]Video also available at `https://youtu.be/aZ1Hjaziv6Y`



(a) Small object completely inside the sensor

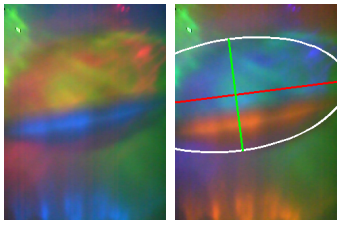
(b) Big object with edges and not completely inside the sensor



(c) Curved object without rough edges

(d) Flat surface object in touch with the entire surface of the sensor

Fig. 2. DIGIT's output to different types of contact and objects.



(a) (b)

Fig. 3. (a) DIGIT's output in presence of contact with an object; (b) the ellipsoid ϵ enclosing the contact area, computed by *PyTouch*.

surface; as shown in Fig. 3, ϵ, the contact area approximated by the ellipsoid can be calculated in pixels.

To characterize the behavior of the sensor, we conducted the following tests:

Test 1: drift. The goal is to determine if there is a drift on the baseline measure, i.e., the one when the sensor is not in contact with an object, to plan suitable re-calibration or reset procedures if necessary. The test consists of collecting data at 1Hz rate in two separate sessions of 30 and 90 minutes, without any contact.

Test 2: sensitivity to environment lightning. To verify if the environment lighting changes the sensor's output, we collect data at 1Hz rate in a 2-minute session, with three different levels of environmental lighting that are artificially generated by a lamp and randomly set.

Test 3: relation between the sensor's output and the contact force. The goal is to determine whether the output of DIGIT relates to the contact force. For simplicity, we only conduct the test for the normal force. To carry out the test, we use the Franka gripper, mounting DIGITs on both "fingers", and an Optoforce sensor (model OMD-20-FG-100N) to measure the contact force. The first part of the test consists in closing the gripper with different force and velocity settings, using the libfranka API. Then, the second consists in closing the gripper to grasp a pin (the one shown in Fig. 2 (a)) of 0.005m of diameter and 0.015m of height, vertically positioned, its edge close to the middle of the sensor. This kind of object presents well defined edges that are easily detected by DIGIT. The open-close gripper commands are sent across several minutes, in loop, with different set forces. Images from DIGIT are acquired at 60Hz.

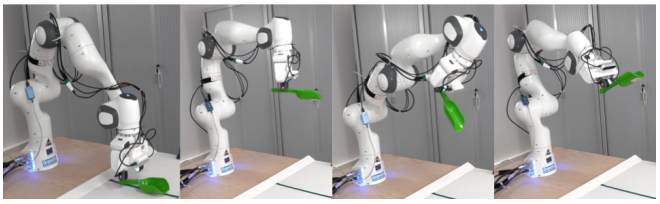


Fig. 4. Some frames of the GRASPA stability procedure. Once the object is grasped, the robot lifts it and executes an excitation trajectory.

Test 4: repeatability of contact measures. The goal is to verify if the images have consistent measures in presence of the same contact, same force, to detect possible hysteresis or drift. The test is executed with the gripper equipped with two DIGITs as "fingers". It consists in closing and opening the gripper 180 times, applying the same force, with an object placed between the 2 DIGITs, collecting images at 60Hz.

C. Validation procedure of the contact ellipsoid as grasp success predictor

Following the observations of [22], we hypothesize that the contact ellipsoid is a good predictor for grasp success and stability. To test this hypothesis, we design the following procedure. We select N objects, placed one by one on the robot's workspace in the same location, and perform several grasps (e.g., 7-10) with different grasp heights, applying the same force. For each grasp, we evaluate two metrics: the grasp success and grasp stability. The grasp is successful if the object is lifted from the table and brought to a fix location (0.55cm above the table). The grasping is stable if the object does not slip during or after the GRASPA stability procedure, proposed by [25]: the object is held by the gripper (without further squeezing) while the end-effector moves for 40 seconds along an excitation trajectory consisting of fast roto-translations, in particular rotations around the end-effector axis. Fig. 4 shows some postures during this procedure; an example can be viewed in the Video attachment. For each grasp, we log the DIGIT output, the minimal contact area ϵ measured by PyTouch *ContactArea*, and the binary information of success or fail for both grasp success and stability. Note that the ellipse computation of PyTouch sometimes fails and return nothing, e.g., there is a contact with an object but it is not detected. Then we consider the following cases: True positive: the algorithm finds an ellipse and the grasp is successful; True negative: the algorithm does not find an ellipse and the grasp was unsuccessful; False positive: the algorithm finds an ellipse and and the grasp was unsuccessful; False negative: the algorithm does not find an ellipse and the grasp was successful.

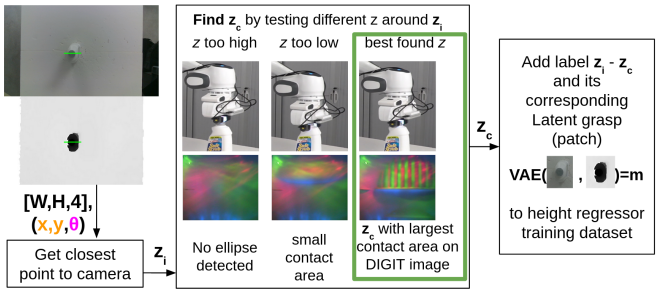


Fig. 5. Finding the best grasp height: 5 heights are tried for each 3D grasp, the height associated with the highest ϵ for the two DIGITs is retained. The best height and its grasp, encoded by the VAE of [26], is saved into a dataset used to train the height regressor.

D. Best grasp height collection procedure

The experiments in the Section IV will show that grasp contact area ϵ is a good predictor of grasp success, which means we can use it to automatize the finding of a good grasping height. To collect a dataset of the best grasp height for given 3D grasp candidates, we design the following procedure. We collect a series of top down grasps, where the gripper's center position (x, y) and orientation θ are manually set, and the initial depth of the grasp (the z-position of the gripper) z_i is computed by finding in the height of the point cloud the closest distance to the camera near the grasp position, as done in [26]. The object is then grasped sequentially at 5 different heights: 2cm above z_i, 1cm above z_i, at z_i, 1cm below, 2cm below. Light and unstable objects are held in place by the experimenter, heights below the workspace plan are ignored. For each grasp, we compute the contact area ϵ of each DIGIT. The height of the grasp with the highest combined contact areas, z_c, is considered the best. Fig. 5 illustrates the procedure. The best height associated to each 3D grasp (x, y, θ), encoded by the latent representation (VAE of grasps represented by image patches) of [26], is added to the dataset of 4D grasp demonstrations, used to train the height regressor described in the next step. We collected 54 demonstrations from 10 objects from the YCB dataset (see Table. III objects 1-10) using this sensor driven collection.

E. Training the grasp height prediction

For a given RGB-D image $(W, H, 4)$, the dataset of the previous section III-D contains grasp demonstrations, represented in the image coordinates by the gripper's center position (x, y), rotated according to the orientation θ and encoded using the same grasp encoding from [26]. Each grasp is represented as a rotated patch $(w, h, 7)$ centered on the middle of the grasp, the patch is fed to a Variational Auto-Econder (VAE) to get a latent representation m. The patch representation is a practical representation inherited from [27], and the latent encoding proved to be data efficient in [26]. This representation is then used to train a height regressor that learns to predict the correction c in meters between the initial height z_i estimation and the height z_c found using the two DIGITs (section III-D). In principle, there is not a preferable method for the design of the regressor, so we compared different methods: SVR, Random Forest, AdaBoost, Gaussian Process, Linear Regression and Neural Network. Training can be done offline. Online, the initial height z_i is computed using the depth data from the RGB-D camera: we extract an oriented cropped patch of the depth point cloud with the width of the selected grasp and a fixed height (5 pixels), and we use the closest point to the gripper (that is, the highest point of the object). z_c is found by adding the output c of the regressor to z_i. Fig. 6 illustrates how the height prediction module is incorporated in the grasping pipeline.

IV. EXPERIMENTS & RESULTS

A. Sensor characterization

We report here the results of the 4 tests to characterize the behavior of DIGIT. Fig. 7 shows the results of **Test 1 & 2**. Each color channel has values in the range [0,255]. In the first 2 plots (short and long session) the values for each channel are almost constant and close to 0, which is consistent with absence of contact (although there are some visible image stream errors causing spikes[2]). The third plot shows that the lightning does not influence the baseline measure.

Fig. 8 shows the results of **Test 3**, where 3 set of forces were applied to the two DIGITs. We use the metric δP (section III-B), i.e., the pixel-by-pixel difference averaged on the 3 channels. The pin object was selected such that the contact area ϵ would not change during the experiment. The results for different repetitions is the same: there is no relation between the DIGIT's output in terms of pixel values and the contact force. This means that **pixel-by-pixel measures with DIGIT cannot be used to distinguish contact forces, but only contact areas**.

Finally, Fig. 9 shows the result of **Test 4**. We executed periodic grasps with the gripper, applying the same force on the pin, 180 times. Two things should be noted. First, the value of δP for the two digits is different: this could be attributed to an asymmetrical distribution of the contact force, even if we used a symmetrical object to avoid this issue. Second, as time progresses we observe strange consecutive "leaps" in the pixel-based measure, appearing at different times for both sensors. We carried out the same experiment many times, and always observed a similar behavior, though the "leaps" seems to happen randomly. This experiment suggests that the pixel response of DIGIT to same contact forces is not consistent over time, but it is subject to additive constant noise. This test further confirms that DIGIT cannot be used in relation to forces across real-world experiments without further investigation.

In addition, we report other issues that we observed during the experiments with DIGIT, that strongly limits its use for repeated grasping in real-world experiments: the elastomer layer is very fragile and easily wear and breaks, which

[2]During our extensive tests with the DIGIT, we observed frequent image stream errors: the output image is transmitted, but it is shifted and so unusable.

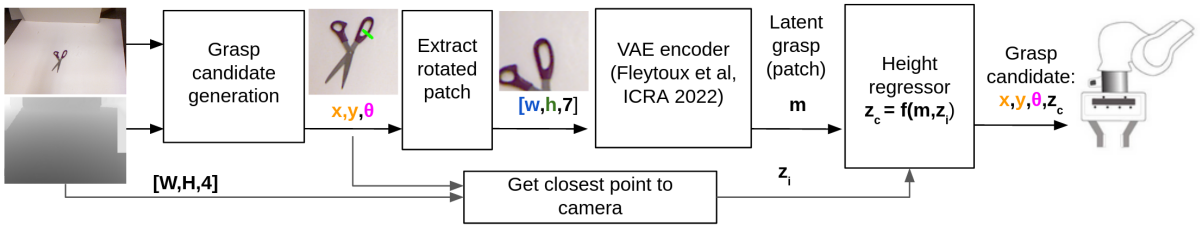


Fig. 6. The grasping pipeline, assuming that the VAE and depth regressor have been trained before and that a suitable top-down grasp candidate generation algorithm is provided (e.g., GR-ConvNet [3], Dex-Net [1], ...). From a RGB-D image, a grasp generator outputs a grasp candidate. The initial grasp height z_i is computed using the depth image. The grasp candidate is represented as rotated patches centered on the the gripper's center position (x, y). It is fed to a VAE to get its latent representation m, which is, in turn, the input of the height regressor (Sec. III-E) trained on the dataset collected with the help of DIGIT (Sec. III-D) to obtain the corrected height z_c.

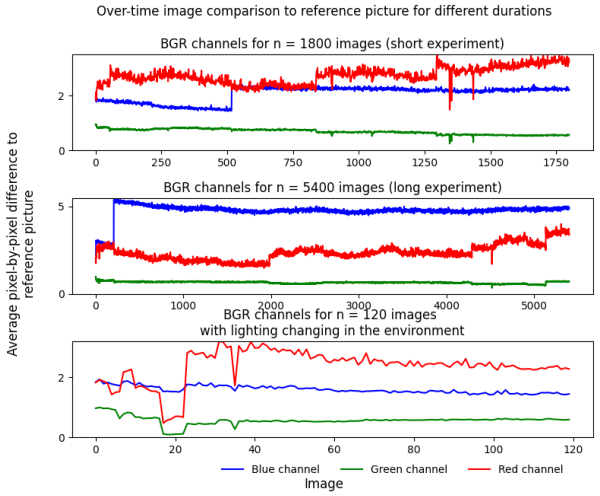


Fig. 7. Test 1 & 2: DIGIT's output as δP to repeatability test over time for different sessions, and for changing environment lighting conditions.

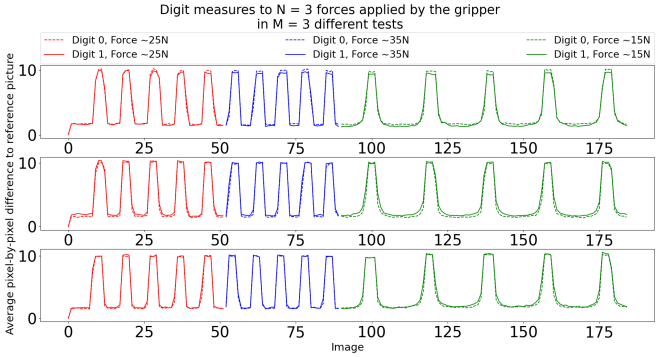


Fig. 8. Test 3: DIGIT's output as δP in relation to different contact forces.

changes the output images; contacts with the edge of the sensor are not well detected; finally, textured objects are often not perceived by PyTouch, which limits the slip detection performance.

For all these considerations, DIGIT does not seem to be suitable for repetitive, long-lasting real-world grasping experiments. It seems to be indicated for limited use for contact detection and contact area estimation.

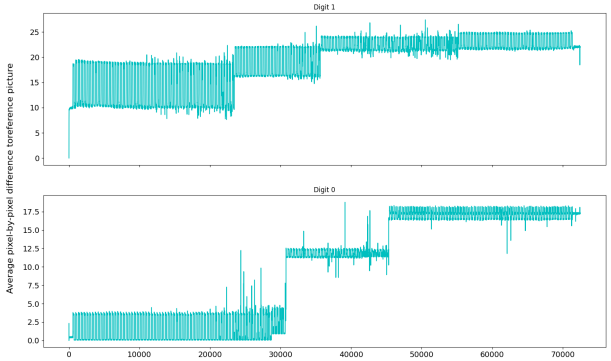


Fig. 9. Test 4: DIGIT's output as δP in relation to repeated contacts of the same force, generated by the gripper.

TABLE I

CONFUSION MATRIX FOR *Contact surface* METHOD, FILTERING THE ELLIPSES FOUND FOR THE CONTACT SURFACE AREA BASED IN THE GRASP INFORMATION LOGGED FOR EACH OBJECT AND EXPERIENCE.

		Prediction label (ellipse presence)	
		Positive	Negative
True label (object grasped)	Positive	114	60
	Negative	7	15

B. *Grasp stability and contact ellipsoid*

In this experiment we want to evaluate if the contact area ϵ can be used to predict the grasp success and stability. We expect that higher contact areas ϵ lead to more stable grasps. We selected 10 objects with different levels of grasping difficulty and evaluated several grasps (up to 11, for each object) with different height, following the procedure described in section III-C, for a total of 192 images.

Table I shows a bigger value for the false negatives, meaning that there were more experiments were there was an object touching the DIGIT sensor, but it was not perceived by the processing of the sensor image. In particular, as shown in Table II, the plastic bolt, which has a particular shape and ridged texture, was "perceived" by the sensor (visible change in the visual output, at least to the human eye) but PyTouch never detected any ellipse. The accuracy and precision of the perception of an object were 65.81% and 94.21%, respectively. Out of the 114 true positive images, 14 had ellipses that did not correspond to the contour of the

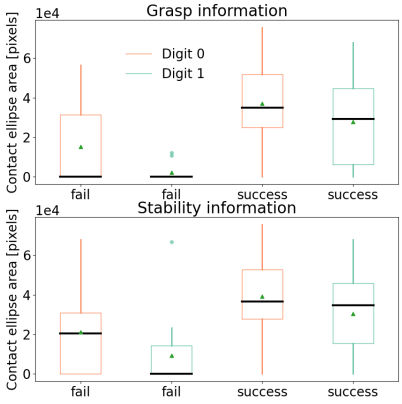


Fig. 10. Boxplot of the contact ellipse area of the two DIGITs in the fail/success cases, for grasp and stability tests.

TABLE II

RELATION BETWEEN THE SUCCESS RATIO OF GRASP AND STABILITY IN THE PHYSICAL TESTS, AND THE SUCCESS RATIO FOR THE ELLIPSE CALCULATION BY THE *ContactArea* METHOD, CONSIDERING THE UNFILTERED DATA.

object	Grasp success	Stability success	Ellipse calculated	
			Digit 0	Digit 1
bottle	9/11	5/11	9/11	9/11
dust shovel	7/9	6/9	7/9	7/9
screwdriver	7/8	7/8	7/8	6/8
white tube	6/8	5/8	7/8	7/8
green shovel (handle)	10/10	7/10	9/10	4/10
strawberry	7/7	7/7	3/7	3/7
green shovel (inside)	10/10	9/10	7/10	9/10
ear protector	6/9	6/9	5/9	1/9
golf ball	9/9	7/9	7/9	7/9
plastic bolt	8/8	8/8	0/8	0/8

object. Fig. 10 shows the relation between the contact area ϵ and grasp success and stability. The results confirm that the higher contact area is associated to higher probability of grasp stability and success.

C. Training the stable grasp height predictor

Using the procedure described in Section III-E, stable grasps are used to train a grasp height predictor. We conducted a 4 fold cross-validation to compare several regressor models (from Scikit-learn) with different parameters found from a grid-search. the same grasp encoding from [26] with a VAE trained on a dataset of 2349 RGB-D scenes (339 objects different from the one used in the experiment), with grasps generated using the GR-ConvNet [3], Dex-Net [1] and GPD [10] algorithms.The models were trained using a Nvidia GTX 1080, and a Intel(R) Xeon(R) Gold 5118 CPU at 2.30GHz. Fig. 11 compares their performance: the best results were obtained by a simple linear regression algorithm.

D. Testing the stable grasp height predictor

We evaluate the performance of the learned stable grasp height predictor by grasping the 10 YCB objects (objects

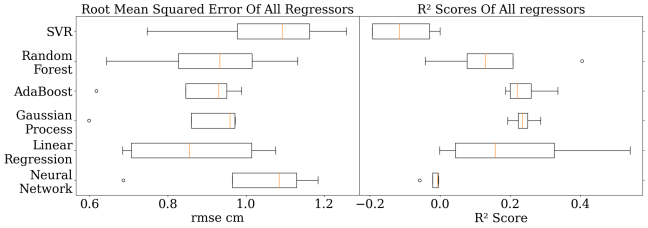


Fig. 11. Performances of different regressor model using 4 fold cross validation. For each fold the models was trained with the same train and validation data (using 75% of the dataset), a grid-search was used to find suitable parameters, the results above relates the performances of each splits on their remaining test grasp demonstrations.

TABLE III

RESULTS ROBOT GRASPING EXPERIMENT WITH AND WITHOUT USING THE DEPTH REGRESSOR.

	with regressor		without	
object	Grasp success	Stability success	Grasp success	Stability success
1 YCB_screwdriver	10/10	10/10	10/10	10/10
2 YCB_power_drill	7/10	3/10	6/10	2/10
3 YCB_scissors	8/10	7/10	9/10	8/10
4 YCB_orange_plastic_bolt	10/10	10/10	10/10	10/10
5 YCB_adjustable_wrench	6/10	6/10	6/10	5/10
6 YCB_hammer	9/10	3/10	8/10	2/10
7 YCB_glass_cleaner	10/10	10/10	10/10	10/10
8 YCB_big_spring_clamps	10/10	10/10	10/10	10/10
9 YCB_bleach_cleanser	10/10	7/10	10/10	3/10
10 YCB_rope	10/10	10/10	10/10	10/10
11 ear protector	8/10	7/10	8/10	8/10
12 bottle	10/10	10/10	10/10	10/10
13 white tube	10/10	10/10	10/10	9/10
14 green_shovel (inside)	4/10	4/10	0/10	0/10
15 dust shovel	7/10	4/10	5/10	5/10

1-10 in Table III) seen in training, but presented in new positions, and 5 new and difficult to grasp objects that are not part of the training dataset.

To evaluate the impact of the height regressor model, we compare the grasp and stability success with and without the height correction (i.e., with and without the regressor). All the objects were grasped at a similar (x, y) location in the workspace. Using the regressor led to an improvement of 5-6% over 300 grasps (20 by objects) for both seen and unseen objects, as reported in Table III, showing that the regressor was able to generalise to new scenes of previously seen objects and also novel objects (objects 11-15) . While the improvement is overall very modest, it must be noted that the height correction is crucial to enable grasping of challenging objects (objects 2-6-9-14-15), that otherwise have little to zero chance of being successfully grasped, with an overall improvement of 16% for grasping and 18% in stability.

V. CONCLUSIONS

Despite its limits, DIGIT can be used for automated data collection of stable grasps, as we experimentally found that the higher contact ellipsoid is associated to higher grasp success and stability. We used DIGIT to automatically determine the best grasp height z for top-down grasp candidates (x, y, θ) which otherwise would use heuristics.

The overall improvement in terms of grasp success and stability is relatively modest across all the objects, meaning that the heuristic is often enough for most of everyday objects. However, our height prediction becomes significant for challenging objects (with transparencies, curved shapes, etc.) that would be otherwise very difficult or impossible to grasp with the simple heuristics.

REFERENCES

[1] J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. J. Kuffner, and K. Goldberg, "Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards," in *IEEE International Conference on Robotics and Automation, (ICRA)*, 2016.

[2] D. Morrison, J. Leitner, and P. Corke, "Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach," in *Robotics: Science and Systems XIV*, 2018.

[3] S. Kumra, S. Joshi, and F. Sahin, "Antipodal robotic grasping using generative residual convolutional neural network," *CoRR*, vol. abs/1909.04810, 2019. [Online]. Available: http://arxiv.org/abs/1909.04810

[4] Y. Jiang, S. Moseson, and A. Saxena, "Efficient grasping from RGBD images: Learning using a new rectangle representation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[5] A. Depierre, E. Dellandréa, and L. Chen, "Jacquard: A large scale dataset for robotic grasp detection," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[6] U. Asif, J. Tang, and S. Harrer, "Graspnet: An efficient convolutional neural network for real-time grasp detection for low-powered devices," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, J. Lang, Ed. ijcai.org, 2018, pp. 4875–4882. [Online]. Available: https://doi.org/10.24963/ijcai.2018/677

[7] N. Gkanatsios, G. Chalvatzaki, P. Maragos, and J. Peters, "Orientation attentive robot grasp synthesis," *CoRR*, vol. abs/2006.05123, 2020. [Online]. Available: https://arxiv.org/abs/2006.05123

[8] D. Wang, C. Liu, F. Chang, N. Li, and G. Li, "High-performance pixel-level grasp detection based on adaptive grasping and grasp-aware network," *IEEE Trans. Ind. Electron.*, vol. 69, no. 11, pp. 11 611–11 621, 2022. [Online]. Available: https://doi.org/10.1109/TIE.2021.3120474

[9] S. Wang, X. Jiang, J. Zhao, X. Wang, W. Zhou, and Y. Liu, "Efficient fully convolution neural network for generating pixel wise robotic grasps with high resolution images," in *2019 IEEE International Conference on Robotics and Biomimetics, ROBIO 2019, Dali, China, December 6-8, 2019*. IEEE, 2019, pp. 474–480. [Online]. Available: https://doi.org/10.1109/ROBIO49542.2019.8961711

[10] A. ten Pas, M. Gualtieri, K. Saenko, and R. P. Jr., "Grasp pose detection in point clouds," *CoRR*, vol. abs/1706.09911, 2017. [Online]. Available: http://arxiv.org/abs/1706.09911

[11] R. Dsouza, "The art of tactile sensing: A state of art survey," *International Journal of Sciences: Basic and Applied Research (IJSBAR)*, vol. 26, pp. 252–266, 05 2016.

[12] A. Begalinova, "Approaches for intelligent robot grasping and manipulation via human demonstration," 2020.

[13] Y. Wan, Y. Wang, and C. F. Guo, "Recent progresses on flexible tactile sensors," *Materials Today Physics*, vol. 1, pp. 61–73, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2542529317301001

[14] M. Lambeta, P. Chou, S. Tian, B. H. Yang, B. Maloon, V. R. Most, D. Stroud, R. Santos, A. Byagowi, G. Kammerer, D. Jayaraman, and R. Calandra, "DIGIT: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation," *IEEE Robotics Autom. Lett.*, vol. 5, no. 3, pp. 3838–3845, 2020. [Online]. Available: https://doi.org/10.1109/LRA.2020.2977257

[15] W. Yuan, S. Dong, and E. H. Adelson, "Gelsight: High-resolution robot tactile sensors for estimating geometry and force," *Sensors*, vol. 17, no. 12, p. 2762, 2017.

[16] M. Lambeta, H. Xu, J. Xu, P.-W. Chou, S. Wang, T. Darrell, and R. Calandra, "PyTouch: A machine learning library for touch processing," *IEEE International Conference on Robotics and Automation (ICRA)*, 2021. [Online]. Available: https://arxiv.org/abs/2105.12791

[17] S. Wang, M. Lambeta, P.-W. Chou, and R. Calandra, "Tacto: A fast, flexible, and open-source simulator for high-resolution vision-based tactile sensors," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3930–3937, 2022.

[18] Y. She, S. Wang, S. Dong, N. Sunil, A. Rodriguez, and E. Adelson, "Cable manipulation with a tactile-reactive gripper," *The International Journal of Robotics Research*, vol. 40, no. 12-14, pp. 1385–1401, 2021. [Online]. Available: https://doi.org/10.1177/02783649211027233

[19] I. H. Taylor, S. Dong, and A. Rodriguez, "Gelslim 3.0: High-resolution measurement of shape, force and slip in a compact tactile-sensing finger," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 10 781–10 787.

[20] R. Platt, "Grasp learning: Models, methods, and performance," 2022. [Online]. Available: https://arxiv.org/abs/2211.04895

[21] R. Krug, A. J. Lilienthal, D. Kragic, and Y. Bekiroglu, "Analytic grasp success prediction with tactile feedback," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 165–171.

[22] R. Calandra, A. Owens, M. Upadhyaya, W. Yuan, J. Lin, E. H. Adelson, and S. Levine, "The feeling of success: Does touch sensing help predict grasp outcomes?" *arXiv preprint arXiv:1710.05512*, 2017.

[23] R. Calandra, A. Owens, D. Jayaraman, J. Lin, W. Yuan, J. Malik, E. H. Adelson, and S. Levine, "More than a feeling: Learning to grasp and regrasp using vision and touch," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3300–3307, 2018.

[24] D. Hsu, J. Latombe, and R. Motwani, "Path planning in expansive configuration spaces," in *Proceedings of the 1997 IEEE International Conference on Robotics and Automation, Albuquerque, New Mexico, USA, April 20-25, 1997*. IEEE, 1997, pp. 2719–2726. [Online]. Available: https://doi.org/10.1109/ROBOT.1997.619371

[25] F. Bottarel, G. Vezzani, U. Pattacini, and L. Natale, "GRASPA 1.0: GRASPA is a robot arm grasping performance benchmark," *IEEE Robotics Autom. Lett.*, vol. 5, no. 2, pp. 836–843, 2020. [Online]. Available: https://doi.org/10.1109/LRA.2020.2965865

[26] Y. Fleytoux, A. Ma, S. Ivaldi, and J. Mouret, "Data-efficient learning of object-centric grasp preferences," in *2022 International Conference on Robotics and Automation, ICRA 2022, Philadelphia, PA, USA, May 23-27, 2022*. IEEE, 2022, pp. 6337–6343. [Online]. Available: https://doi.org/10.1109/ICRA46639.2022.9811760

[27] 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," in *Robotics: Science and Systems XIII*, 2017.