Transferability of learning-based grasping algorithms: a case study

Brice Denoun[1,2], Beatriz Leon[2], Miles Hansard[1] and Lorenzo Jamone[1].

Introduction

The tasks of robotic grasping and manipulation have attracted substantial interest, in both research and industry, over the past few years. This has resulted in impressive capabilities for robots, which are able to tackle increasingly complex tasks [1]. One specific focus of recent research is the ability to automatically generate grasps for unknown objects from RGB-D input data [2]. However, there have been few comprehensive evaluations of these algorithms, and so their advantages and limitations are not fully understood. We have recently addressed this problem[3], by performing an exhaustive comparison between four state-of-the-art grasping algorithms. Each method was tested on 1500 grasps: 20 different objects, 5 different poses, and 15 repetitions. Two of the four evaluated methods are based on Deep Learning [3], [4], while the other two use traditional geometric computations [5], [6]. Beyond the results of our previous study, we observed that the two Deep Learning based methods we benchmarked were more sensitive to changes in the experimental conditions than the more traditional approaches. In this work, we quantify this phenomenon and show that only the two Deep Learning based methods have significantly different performance when run on modified experimental setups. This leads us to consider a new metric evaluating the transferability of Deep Learning based methods for grasping and manipulation.

Previous work

This section summarises our previous work, and provides a context for the new results. We carried out a benchmark of methods that generate grasps based on depth data, in the form of a depth map or point cloud. The selected algorithms generate a grasp configuration as a set of either four or six parameters. A 4D grasp configuration G is generally defined as (x, y, z, θ) whereas a 6D grasp is represented as $(x, y, z, \phi, \psi, \theta)$. Here (x, y, z) represents the position of the gripper in 3D space, and (ϕ, ψ, θ) represents the roll, pitch and yaw angles. In particular, methods generating 4D grasps have a smaller search space, usually corresponding to *top grasps* [5], [4]. On the other hand, methods generating 6D grasps cover the entire configuration space, and thus are theoretically able to provide a wider range of grasps [3], [6].

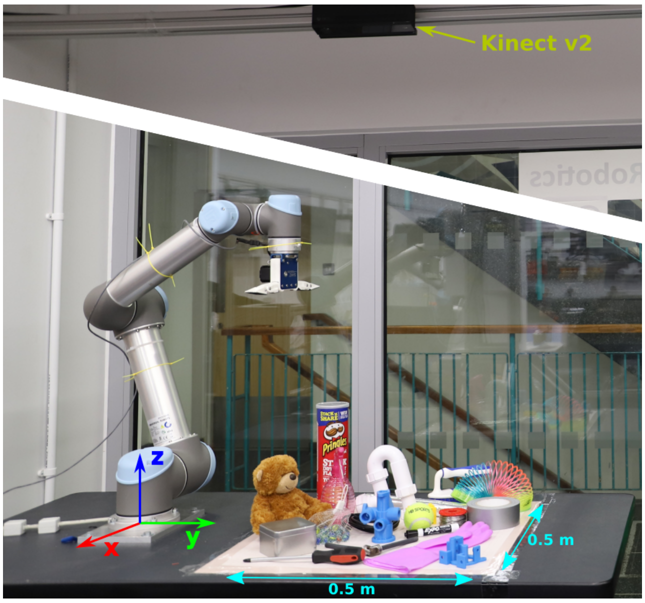

Fig. 1. The robotic setup used to perform the benchmark. Note that the image is discontinuous: the Kinect is farther from the table than it appears. All methods were executed in the same conditions: each of the 20 objects was grasped in isolation, in 5 different poses within a 0.5×0.5m workspace.

All the methods were evaluated on the same robot platform and following the same protocol.

Experimental setup

We used a robotic setup that is representative of typical industrial applications, while also being easy to replicate in a lab. As shown in Figure 1, we use a Universal Robot 5 arm, mounted on a workbench. The manipulator used is the EZGripper, an under-actuated two fingered gripper. It is actuated by a position controller after which a medium torque (EZGripper setting 200) is applied to hold the object. The reference frame is located at the base of the robot. The vision device used to capture data is a Kinect v2, located at $x = -0.08$, $y = 0.430$, $z = 1.86$) meters from the origin frame, and pointing perpendicular to the workbench. The robot's workspace is defined as a 0.5×0.5m area centred at ($x = 0.1$, $y = 0.595$, $z = 0$) meters from the base of the robot. For reproducibility, we used the open source and publicly available Modular Benchmarking Framework[4]. This software makes it easy to integrate and run different grasping methods, while keeping all other parameters (e.g. robot speed, motion planner, controller) constant.

[1] B. Denoun, L. Jamone and M. Hansard are with ARQ (Advanced Robotics at Queen Mary), School of Electronic Engineering and Computer Science, Queen Mary University of London, UK {b.d.denoun, l.jamone, m.hansard}@qmul.ac.uk.

[2] The Shadow Robot Company, London, United Kingdom

[3] Submitted to IROS 2020

[4] `https://modular-benchmarking-documentation.readthedocs.io/en/latest/`

Set of objects

The YCB dataset [7] is a well established resource for robotic grasping research. It comprises a set of physical objects, as well as their corresponding point clouds. The ubiquity of this dataset is somewhat problematic, in the present context, because it is commonly used to train learning based models [3]. In order to avoid re-using the YCB dataset for testing (which would unfairly benefit certain methods), we have gathered a new set of 20 objects, with visual and structural properties that make them challenging to grasp. This set comprises the following items: a net of marbles, a metallic box, a cardboard tube, a screwdriver, a roll of duct tape, an HDMI cable, a thick plastic glove, an empty spray bottle, a marker, a socket universal joint (with bars), a tennis ball, a Duplo block, a soft teddy bear, a roll of kitchen foil, a sink pipe, a brush, a spring toy, a spool of solder, and two 3D printed adversarial objects [8]. Exact copies of these objects can be purchased online at a reasonable cost[5]. We argue that this set gathers objects with a wide range of interesting properties, including softness, articulation, slipperiness, asymmetry and so on.

Protocol

We used a procedure similar that of GRASPA [9], which was designed to evaluate the success rate and stability of grasps. For each repetition, the system starts from a pre-recorded state, from which the robot moves into the defined workspace. The following steps are then performed, on each trial:

- Pre-grasp and grasp pose are generated by the algorithm
- Robot arm moves to the generated pre-grasp pose
- Gripper opens to a predefined and constant posture
- Robot arm moves to the generated grasp pose
- Gripper closes completely
- Robot moves to a predefined stable position and waits for 2 seconds
- Robot executes a predefined and constant trajectory shaking the object (stability test)
- Robot moves back to the previously generated pose
- Gripper opens to release the object
- Robot moves back to the starting pose

The shaking motion is designed to test the *stability* of the grasp, bearing in mind that the gripper is not squeezing with the highest torque possible. In addition, we parameterised the trajectory to reach successive way-points within 0.3 seconds, pausing at each one for 0.2 seconds. This leads to an *energetic* shake, unlike the one performed in [9].

Performance evaluation

The definition of a *good* grasp remains an open question in the robotics community. Some works define a *good* grasp as a configuration that successfully picks and lifts the object. Others are slightly stricter and wait for five seconds before taking a decision. We argue that the definition of a *good* grasp is closely related to the application. Unlike our previous work[5], we consider here that a grasp can have only three mutually exclusive outcomes: Failure (the object is not grasped before the shaking trajectory is executed), Unstable (the object falls during the shaking motion) and Success (the object is deposited on the table after the shaking motion). These metrics define the ability of a grasp to keep the object even after a set of quick movements. Although the object might not be deposited at the same location, such grasp configurations are still valid for a wide range of applications. We can for instance think of the classical problem of automatically clearing a table, for which once grasped, the object will be dropped inside a crate, regardless of its pose.

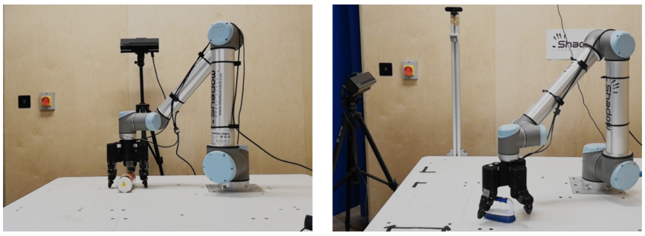

Fig. 2. Setups used for the second set of experiments. The cropping areas were the same for both camera pose. Pose 1 is illustrated on the left and Pose 2 on the right. We evaluated each method based on 30 grasps, 10 for each object (brush, cardboard tube and tennis ball).

Results

In order to perform our benchmark, each method had to be integrated into the same experimental setup. Before following the above protocol, we evaluated the different methods on a subset of objects (screwdriver, tennis ball, and marker) in order to make sure that all methods worked properly on our setup. The performance of the geometry based methods, in this preliminary test, was broadly as expected. However, the performance of the Deep Learning based methods was somewhat lower than reported. We attribute this to the larger size of our workspace (0.7×0.7m), with respect to the final setup (0.4×0.4m). These results are summarized in Table I.

TABLE I

Distribution of grasp outcomes for two different cropping areas, over 30 grasps.

	0.7×0.7m			0.5×0.5m		
	Succ	Unst	Fail	Succ	Unst	Fail
PointNetGPD* [3]	**0.4**	0.466	0.133	**0.7**	0.266	0.033
GGCNN2* [4]	**0.4**	0.167	0.433	**1.0**	0	0
Suzuki [5]	**0.8**	0.133	0.067	**0.9**	0.067	0.033
Makhal [6]	**0.533**	0.333	0.134	**0.666**	0.266	0.0666

* Deep Learning based method

If we consider the success rate as a performance metric, then the Chi-Square tests show a significant difference ($\alpha = 0.05$) between the two conditions only for the Deep Learning based methods ($\chi^2 = 22.9$, p-value=1.67×10^{-6} for [4] and $\chi^2 = 4.31$, p-value=0.0379 for [3]), whereas we cannot conclude to any difference for the geometry based

[5] www.eecs.qmul.ac.uk/~bdd30/benchmark_presentation

methods ($\chi^2 = 0.523$, p-value=0.470 for [5] and $\chi^2 = 0.625$, p-value=0.429 for [6]). This first experiment seems to show that the two selected Deep Learning based methods tend to be less robust to input data changes. We wanted to confirm this assumption and carried out similar experiments in a different setup. In this configuration, the cropping area (0.5×0.5m) remains the same but the camera pose changes between Pose 1 (Figure 2 left hand side) and Pose 2 (Figure 2 right hand side). The results are shown in Table II. The results of Chi-Square tests ($\chi^2 = 4.39$, p-value=0.0362 for [4] and $\chi^2 = 3.89$, p-value=0.0486 for [3] and $\chi^2 = 4.04 \times 10^{-33}$, p-value=1 for both [5] and [6]) indicate that the camera pose has a significant effect on the performance of the Deep Learning based methods.

TABLE II

DISTRIBUTION OF GRASP OUTCOMES FOR TWO DIFFERENT CAMERA POSES OVER 30 GRASPS.

	Pose 1			Pose 2		
	Succ	Unst	Fail	Succ	Unst	Fail
PointNetGPD* [3]	**0.566**	0.333	0.1	**0.833**	0.1	0.066
GGCNN2* [4]	**0.733**	0.266	0	**0.433**	0.433	0.133
Suzuki [5]	**0.7**	0.3	0	**0.733**	0.266	0
Makhal [6]	**0.766**	0.133	0.1	**0.8**	0.166	0.033

* Deep Learning based method

DISCUSSION

It is important to note that the results shown in the previous section are not meant to be used to compare the success rates of the four methods, but rather to evaluate the impact of the *vision* based parameters. In particular, we can see that the performance variability, observed in this work, is wider for Deep Learning based methods than for more traditional methods. We can infer that the learning based methods have encoded certain properties of the training scene or environment, such as the size or shape of the workspace, which may not generalise to other setups. This is problematic, because it is very likely that such properties will differ between labs or industrial setups. Although re-training the model for each setup would be ideal, it would require detailed knowledge of the data collection process, and a substantial amount of time.

For this reason, we argue that we need a metric to assess how transferable a learning-based grasping algorithm is, without re-training. One approach would be to create a YCB-like protocol to evaluate the deployment of grasping and manipulation methods. Instead of evaluating a newly developed method on a single configuration, this protocol could contain several prototypical viewpoints that correspond to real-world scenarios (e.g. wrist-mounted camera, perpendicular top view, tilted side view). The details of the performance evaluation for each configuration would allow us to identify the shortcomings of a given method, which could then be addressed in future work. For example, the training dataset may need to be re-balanced, or the regularization parameters may need to be adjusted. To go further, a global metric could be computed from these tests, in order to estimate the *likelihood* that a given method would maintain a certain level of performance, when integrated with a new setup. A simple metric would be the variance of the recorded performance across the different possible camera poses. In this case, the smaller the variance is, the more transferable the model is without re-training. Although such a protocol and metric could be extended to geometry-based methods, our results indicate that it would be particularly valuable for Deep Learning based methods.

CONCLUSION

This work is an exploration of the results from a recent benchmark of vision-based grasping algorithms. We performed two sets of experiments, both of which indicate that the two Deep Learning based methods show significant performance differences, when parameters affecting the input data are modified. Although our results are based on a small number of algorithms (two geometry based and two Deep Learning based), they highlight the need for a metric that can quantify the generality of pre-trained grasping methods. We believe that such a transferability metric, of the kind envisaged here, would help to advance the use of robotic grasping in real-world scenarios.

ACKNOWLEDGEMENTS

This work was supported by the EPSRC UK (NCNR, EP/R02572X/1 and MAN^3, EP/S00453X/1) and by The Shadow Robot Company.

REFERENCES

[1] M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, *et al.*, “Learning dexterous in-hand manipulation,” *arXiv preprint arXiv:1808.00177*, 2018.

[2] J. Bohg, A. Morales, T. Asfour, and D. Kragic, “Data-driven grasp synthesis-a survey,” *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 289–309, 2013.

[3] H. Liang, X. Ma, S. Li, M. Görner, S. Tang, B. Fang, F. Sun, and J. Zhang, “PointNetGPD: Detecting grasp configurations from point sets,” in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 3629–3635, IEEE, 2019.

[4] D. Morrison, P. Corke, and J. Leitner, “Learning robust, real-time, reactive robotic grasping,” *The International Journal of Robotics Research*, p. 0278364919859066, 2019.

[5] T. Suzuki and T. Oka, “Grasping of unknown objects on a planar surface using a single depth image,” in *Advanced Intelligent Mechatronics (AIM), 2016 IEEE International Conference on*, pp. 572–577, IEEE, 2016.

[6] A. Makhal, F. Thomas, and A. P. Gracia, “Grasping unknown objects in clutter by superquadric representation,” in *2018 Second IEEE International Conference on Robotic Computing (IRC)*, pp. 292–299, IEEE, 2018.

[7] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, “Benchmarking in manipulation research: The ycb object and model set and benchmarking protocols,” *arXiv preprint arXiv:1502.03143*, 2015.

[8] 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,” *arXiv preprint arXiv:1703.09312*, 2017.

[9] F. Bottarel, G. Vezzani, U. Pattacini, and L. Natale, “GRASPA 1.0: GRASPA is a robot arm grasping performance benchmark,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 836–843, 2020.