Predicting Grasp Success in the Real World - A Study of Quality Metrics and Human Assessment

Carlos Rubert[a,*], Daniel Kappler[b], Jeannette Bohg[c], Antonio Morales[a]

[a] *Robotic Intelligence Laboratory at the Department of Computer Science and Engineering, Universitat Jaume I of Castellon, Spain*
[b] *Google X Robotics, Mountain View, CA, USA. ***
[c] *Department of Computer Science, Stanford University, USA*

Abstract

Grasp quality metrics aim at quantifying different aspects of a grasp configuration between a specific robot hand and object. They produce a numerical value that allows to rank grasp configurations and optimize based on them. Grasp quality metrics are a key part of most analytical grasp-planning approaches. Additionally, they are often used to generate ground-truth labels for synthetically generated grasp exemplars required for learning-based approaches. Recent studies have highlighted the limitations of grasp quality metrics when used to predict the outcome of a grasp execution on a real robot. In this paper, we systematically study how well seven commonly-used grasp quality metrics perform in the real world. To this end, we generated two datasets of grasp candidates in simulation, each one for a different robotic system. The quality of these synthetic grasp candidates is quantified by the aforementioned metrics. For validation, we developed an experimental procedure to accurately replicate grasp candidates on two real robotic systems and to evaluate the performance of each grasp. Given the resulting datasets, we trained different classifiers to predict grasp success using only grasp quality metrics as input. Our results show that combinations of quality metrics can achieve up to a 85% classification accuracy for real grasps.

Keywords: Grasping, Grasp simulation, Machine learning, Prediction model, Real grasp execution.

1. Introduction

Grasp quality metrics are computational tools that evaluate grasp configurations consisting of contact points between the robot end-effector and the object surface. These metrics quantify grasp quality based on the measured forces and torques exerted at the contact points. Desirable properties of a grasp that are evaluated by existing quality metrics are force-closure, equilibrium, dexterity, stability and others [43, 37].

Grasp quality metrics aspire to predict the outcome of a grasp on a specific object when executed with a real robotic system. They are central to analytic approaches in grasp planning, which are formulated as an optimization problem over grasp configurations given the object and robot hand dynamic models [5]. Another common use of grasp quality metrics is to generate the ground- truth labels for synthetically generated grasp configurations. The resulting data sets are then used for learning-based approaches in grasp planning e.g. [18, 27, 28]. Unlike analytic approaches, learned grasp planning models often take partial sensory data as input instead of full 3D object models. They are also able to generalize over objects that the trained model has not yet seen.

Multiple studies have emphasized the limitations of classic grasp quality metrics when predicting grasp success in the real world [2, 44, 9, 18]. First, quality metrics rely on precise models of robot hands and objects. These are not always available, in particular for the wide variety of objects that exist in the real world. Second, the desired contacts have to be precisely achieved for the grasp quality metric to be valid. This is challenging due to the inherent inaccuracy of robot control, noise in sensor measurements and other sources of uncertainty. And third, individual quality metrics typically focus only on specific aspects of the physical interaction. Real executions are however affected by a variety of aspects that may not be taken into account by a particular metric. Thus, grasp configurations may fail in the real world despite a high-quality value. For example, most commonly used quality metrics consider only the moment *after* which contact is established between robot hand and object. However, when establishing the grasp and lifting the object other aspects will have a large influence on the success of the grasp.

The main contribution of this work is to evaluate to what extent grasp metrics obtained in simulation transfer

*Corresponding Author
**The majority of this work has been conducted while the author was with the Autonomous Motion Department at the MPI for Intelligent Systems, Tübingen, Germany.
Email addresses: carlos.rubert@uji.es (Carlos Rubert), daniel.kappler@gmail.com (Daniel Kappler), bohg@cs.stanford.edu (Jeannette Bohg), morales@uji.es (Antonio Morales)

to the real world. Prior work [2, 44, 18] has typically only analyzed the ϵ-metric by [13]. In this paper, we analyze seven quality metrics as proposed in [40] and validate them against grasp executions on two different real robotic platforms. Our aim is to find one metric or a combination of metrics that can most accurately predict grasp success in the real world. This has implications for both, analytical grasp planning methods as well as learning-based approaches that are trained on synthetically generated grasp exemplars.

For this purpose, we execute and monitor a set of grasps on two real robotic setups. These grasps were generated and evaluated in simulation. In this way, we obtained two large grasp databases, one for each robotic platform. Each grasp is labelled with a success label from the real execution as well as with the values from the seven grasp quality metrics. These databases contain more than 1700 real grasps. We designed an experimental protocol to precisely replicate simulated grasps in the real world. We 3D printed highly accurate replicas of those virtual objects. For executing the grasp, we use the real-time visual tracking and motion generation system proposed in [19] to achieve the desired, synthetic grasp configuration as precisely as possible. We further re-evaluate the recorded grasp configuration in simulation to eliminate remaining inconsistencies.

Given these datasets, we numerically analyse the predictive power of individual metrics. We also use them as input to linear and non-linear classifiers to understand to what extend combinations of grasp quality metrics improve grasp success prediction. The best binary classification model achieves a 80% accuracy when predicting for similar grasps and 70% when predicting for completely novel grasps. These scores increase to 85% and 80% when using a cascaded classifier.

In our previous work [38], we studied the same quality metrics using ground-truth labels provided by physics simulation. This metric was proposed in [18] and was found to be highly correlated with human labels as opposed to the aforementioned, classic ϵ-metric of grasp quality. In this work, we collected grasp success labels in the real world for the same grasp configurations. This allowed us to validate the labels from physics simulation and human subjects collected in [18]. We found that both are good at rejecting bad grasps, but have a low success rate when predicting good ones.

This paper is organized as follows. Section 2, presents related work that develops similar or alternative approaches to the problem of grasp synthesis. Section 3 describes the foundations of this work. The detailed methodology for data acquisition is explained in Section 4. Specifically, it describes the robotic platforms, grippers, objects and experimental protocol followed to execute and record experimental grasps. In Section 5, we describe the properties of the experimental grasp databases. 1 Sections 6, and 7 present the analyses and results to answer the core questions of this paper. Each section includes a detailed discussion of the results. Finally, Section 8 provides the conclusions and future lines of this work.

2. Related work

Several works have evaluated the predictions of popular quality metrics and have compared them with the outcomes from experiments with real robots. Some results showed that highly-ranked grasp using the ϵ-metric perform poorly when executed in real experiments [2, 10], and they are very sensitive to position inaccuracy [44]. Incidentally, the ϵ-metric is not considered in this paper because in a previous work [40] was found to be highly correlated with other metrics, in especial Q_{C2} in table 1.

Some approaches have been proposed to address these limitations. From an analytical side, some authors have tried to create global metrics that combine existing quality metrics. An analytical correlation study showed the existence of at least seven independent metrics [40] but no combination rule was proposed. A common approach has been the parallel combination of metrics, that is, every grasp is evaluated by a set of metrics, and the values obtained are combined, usually by addition, to produce a unique evaluation index [6, 1]. Different normalization procedures can be applied to every metric for their combination, as well as different weighting coefficients [8]. The limitation of parallel combinations is that there are no clear rules to choose and weigh the importance of every metric. Hester et al. [16] propose a serial approach where one metric is used to generate and select a subset of grasp configurations, and a second metric is used to rank them.

There have also been some attempts to build more realistic and robust metrics. Kappler et al. [18] propose a *physics metric* that requires a complete dynamic simulation of the execution of a grass configuration. Others have proposed metrics like the *probability of force closure* [20], or a *pose error robust metric* [44] which apply probabilistic principles to address several sources of uncertainty in the executions of grasps and incorporate thme in the metric.

The difficulty of finding reliable grasp success metrics and their assumptions of full object knownledge has boosted the development of *data-driven* approaches for grasp planning [5]. These solutions require the existence of datasets of labelled grasps. Some works execute and automatically label grasp candidates in simulation [18, 36]. This allows to efficiently generate large amounts of grasp data points to train prediction models. However, there is no clear validation of the resulting ground-truth labels against real-world grasp success.

Following these trend of creating synthetic datasets, Mahler et al. [27] built *Dex-Net*, a database with 2.5M simulated grasps that are evaluated with the *probability of force closure* metric. This database was used for robust grasp planning. In follow-up work [28], the authors extended this database to 6.7M grasp samples and evaluated the trained models on a real robot. More than 10.000

object models were used to create the database and experiments resulted in a 99% classification precision. Compared to the work presented in this paper, Mahler et al. [28] present results on 2-finger grasps and only with a single robot gripper. We evaluate grasp metrics on two different, three-fingered robotic hands. Also, we directly validate the metrics against real grasps to understand their prediction accuracy.

Recently, Bousmails et al. [7] created synthetic images of virtual scenes with random object shapes and combined them with real-world images in training datasets. The paper studies the option for randomization to transfer models trained on synthetic images to real world cases. Their results indicate that combining synthetic and real data can reduce drastically the number of required real experiments, while maintaining similar success ratios.

These approaches use simulation to bootstrap the construction of training datasets for learning approaches. Most of these cases replicate on simulation situations of the real world with either synthetic images or arrangements of object shapes, and record this raw data without any intermediate representation. The consequence is that huge number of simulations are necessary, in the order of hundreds of thousands or millions. Realistic grasp quality metrics can be an optimal intermediate representation of the mechanical and kinematic properties of a grasp configuration, almost independent of the shape of the object and the design of the hand, and thus reduce the number of simulations required.

Another trend to annotate synthetic data is to ask humans for the evaluation of grasps before executing them. This is based on the assumption that humans intuitively know how a successful grasp looks like [2]. These methods differ in which data is presented to humans, either RGB-D images with areas indicating the existence of potential grasps regions [23], or visualizations of robot hand configurations over target objects [18]. An open question about these approaches is whether human predictions can be fully trusted.

Finally, an important trend is to collect large grasp datasets on real robots where the grasp success is automatically labelled. The goal is to train models with these datasets to establish a relationship between raw sensory input (e.g. an RGB image) and a successful grasp. Morales et al. [32] proposed a first version of this idea in form of an active learning approach to choose the next grasp, given the accumulated experience from previous trials.

In a more recent work, Levine et al. [24] executed, recorded and automatically labelled more than 1.7 million grasp attempts in two experiments using an automatic robot farm composed of similar manipulator robot cells. Deep learning techniques were used to establish a relation between RGB images from scenes and successful sequences of gripper actions. Similar to [34], this work was an ambitious attempt to use raw input data to learn a grasp execution procedure, avoiding any type of labeling bias. The work was constrained to two-finger grippers and no previous models of the objects were required. The results demonstrated that effective control sequences can be learned, novel objects can be grasped successfully, and knowledge can be transferred between different robot setups. However, it also was a very expensive grasp labeling procedure. Simulation in the spirit of [28, 18, 7] would have been an option to have a cheaper and faster labeling procedure.

We use experimental data from a real robot to analyze how well individual or combinations of grasp quality metrics can predict grasp success. To this end, we build a large database of grasps that are evaluated on two robotic platforms as well as in simulation. Since our approach is based on quality metrics and contact points, results are not restricted to two-finger grippers. We question the reliability of simplified physics engines and human judgment based on the analysis of a previous database.

3. Foundations

The work presented in this paper is based on the methods and results described in our prior work. A short summary of them is presented here for the reader convenience.

3.1. A physics metric

[18] proposes a physics metric that consists of a full dynamic simulation of a grasp when closing the fingers on the target object. The main hypothesis of this work is that physical forward simulation of grasps is more suitable for automatic labeling of data than the classical and most common Ferrari & Canny [13] metric. The final label of each grasp was computed by averaging over the physically-simulated outcomes of 30 grasps on a slightly perturbed object pose around the original, reference pose. This mimics noise in perception and actuation of a robot. For the computation of the physics metric gravity force was not considered and no support plane was present in the simulated scene.

For reference, this physics metric will be reported in the results section. It is a good indication of the prediction performance of grasp stability that can be achieved when investing in a computationally expensive simulation.

3.2. Human oracle

To validate the hypothesis explained in the previous subsection, grasps labelled automatically in simulation with the physics metric are compared against the answers of human subjects on a subset of the generated grasps. The authors assume that human subjects are perfect oracles and therefore use them to validate the physics metrics.

Using the Amazon Mechanical Turk (AMT), human subjects were presented a subset of grasps and were asked to use their judgment to estimate whether a grasp would succeed or not. Several images of a robot gripper configured in a grasping pose on a target object were shown to human subjects who evaluated them as *Stable*, *Unstable*,

or *Unknown* in the case that the human was unsure of the outcome of the grasp.

Each grasp was evaluated by at least 5 humans and the average of *valid* evaluations were used to set a grasp as successful or not. It is important to note that the subjects from AMT are likely lacking any kind of experience in robotic grasping. Thus, their assessment on grasping is based on their own experience in human manipulation and not on their knowledge of kinematics of typical robotic hands.

The comparison results show that the labels based on the physics simulation are closer to human judgment and are therefore considered more realistic. Kappler et al. [18] also showed that physics-based labels were easier to learn from. However, physics simulation is also orders of magnitude more expensive than computing the classic metrics.

3.3. A selection of quality metrics

In previous work [40], we performed a statistical analysis of ten quality metrics. We recorded the values of these metrics on simulated grasps computed on more than a hundred objects with seven hand models. The analysis consisted of (i) establishing an upper and lower bound to normalize the metrics, (ii) measuring the stability of the metrics in the presence of small disturbances, and most importantly (iii) visualizing the correlation between different metrics. This last analysis allowed to discard three metrics, reducing the initial set to seven. These seven independent metrics will be used in this article (see Tables 1 and 2).

Table 1: Quality Metrics

Name	Description	Formula
Q_{A1}	Smallest singular value of G [25]	$\sigma_{min}(G)$
Q_{B1}	Distance between the centroid of the contact polygon and the center of mass of the object [11, 35]	$1 - \frac{distance(p, p_c)}{distance_{max}}$
Q_{B2}	Area of the grasp polygon [31]	$\frac{Area(Polygon(p_1, ...p_n))}{Area_{max}}$
Q_{B3}	Shape of the grasp polygon [21]	$1 - \frac{1}{\theta_{max}} \sum_{i=1}^{n_f} \lvert\theta_i - \bar{\theta}\rvert$
Q_{C2}	Volume of the convex hull [30]	$\frac{Volume(CW)}{Volume_{max}}$
Q_{D1}	Posture of manipulator joints [26]	$1 - \frac{1}{n_q} \sum_{i=1}^{n_q} (\frac{y_i - a_i}{a_i - y_{iM}})^2$
Q_{D2}	Inverse of the condition number of $\mathbf{G_J}$ [41, 22]	$\frac{\sigma_{min}(G_J)}{\sigma_{max}(G_J)}$

See table 2 for the definition of the terms in the formulas.

3.4. Prior databases

In this work, we use the following database containing thousands of grasps:

Physics metric DB : [18] generated a large-scale grasp database on a wide variety of objects in simulation using OpenRave [10]. The grasps were generated by sampling grasp candidates uniformly around the object surface. Altogether, it contains approximately half a million grasps generated on more than 600 different object models with the BarretHand. [3]. Each of them is evaluated with the *Physics Metric* and with the ϵ-metric averaged over 30 grasp trials on a slightly perturbed object pose.

Human labelled DB Some of the grasps on the *Physics Metric DB* were labelled by humans. The resulting database contains 4752 grasps applied to a wide variety of different objects (616) with a known 3D shape model.

Quality Metrics DB Rubert el al. [40] simulated over 900.000 grasps configurations on a set of 126 object models using 7 robot hands. For each grasp, the seven quality metrics were computed. This database was later extended [39] to evaluate the quality metrics of the grasps included in the two previous databases. This ensures that all the grasp candidates included in the present paper have their quality metrics calculated.

4. Robot setup and protocol

In order to have data from real robot experiments, we define a protocol to execute and record the results of exhaustive experiments. This section describes the two robot setups on which we carry out the experiments and the protocol applied to acquire the experimental data.

4.1. Robotic platforms

4.1.1. Apollo

Apollo[1] (see Fig. 3) is a dual-arm manipulation platform used to study active perception, grasping, and manipulation. It has two arms, hands with tactile sensors, and an active vision head. The robot has two KUKA lightweight robot arms (7 DOFs), two Barrett hands, and a Sarcos head featuring different vision sensors, including an *Asus Xtion PRO*.

[1]Apollo Robot: `https://am.is.tuebingen.mpg.de/pages/robots`

4.1.2. Tombatossals

Tombatossals (see Fig. 4) is a multipurpose humanoid torso aimed at performing research on autonomous grasping and manipulation tasks in unstructured household scenarios. The humanoid torso is composed of two arms, two hands and a head for a total of 29 DOF. Both arms are *Mitsubishi PA10-7C*, 7 DOFs industrial manipulators with position repeatability of 0.1mm. Each arm weights 40-kg and has a 10-kg payload.

Tombatossals has a *Barrett Hand* on its right arm and a *Schunk SDH Hand* on the left arm. Both hands are equipped with tactile sensors from *Weiss Robotics*.[2] The head is composed of a *TO40* pan-tilt-vergence system and a *Kinect*. More details on Tombatossals are described in [12].

4.2. Robotic Grippers

The *Barrett* and *Schunk SDH* grippers (see Fig. 5) are used to perform the real experiments. The *Barrett* hand has a weight of 980 g. Its payload is 6 kg. It has three fingers with two joints each. Two of them have an extra degree of freedom, with 180 degrees of lateral mobility supporting a large variety of grasp types. All joints have high-precision position encoders. It has 3 fingertip torque sensors, one per finger.

The Schunk *SDH* hand weights 1.95 Kg. It has 3 fingers with two links each. Two fingers can rotate up to 90°in reverse. It features high flexibility in terms of shape, size and position of the objects to be gripped. It has 6 tactile sensors for pressure and surface recognition.

When performing a grasp, all the fingers of the robot hands are closed simultaneously until contact is detected with any of the links. The corresponding joint is then blocked. If no contact was yet detected with a distal joint, it continues to close until contact is sensed. For the Barrett hand, a strain gauge in the finger knuckle is used to detect collisions. In the case of the Schunk SDH, there are tactile sensors for detecting the contact on each link of the hand.

[2]Weiss Robotics sensors: `http://www.weiss-robotics.de/`

Table 2: Notation

G	Grasp matrix
σ_{min}	Minimum singular value
σ_{max}	Maximum singular value
p	Centroid of contact polygon
p_c	Object centre of mass
pi	Vertex of the grasp polygon
pi_p	Projected vertex of the grasp polygon on a plane
θ_{max}	Sum of differences between the internal angles when the polygon has the most ill-conditioned shape and those of a regular polygon
θ_i	Inner angle at the vertex$_i$ of the grasp polygon
$\bar{\theta}$	Average angle of all inner angles of the grasp polygon
CW	Convex hull of the primitive wrenches
n_q	Number of joints of the hand
a_i	Middle range position of a joint
y_i	Angle of joint i
y_{iM}	Maximum angle limits of joint i
G_J	Grasp Jacobian matrix
$distance_{max}$	Maximum distance from the object's centre of mass to any point in the object's contour
$Area_{max}$	Maximum possible area of the hand, calculated as the area of the polygon when the hand is fully opened
$Volume_{max}$	Maximum volume of the convex hull of the primitive wrenches

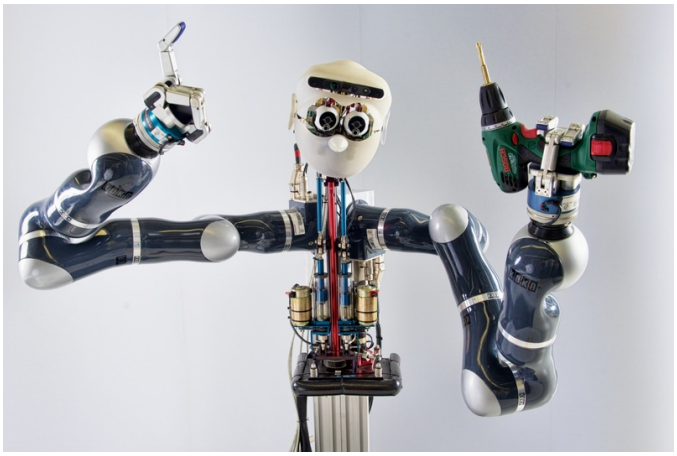


Figure 3: The Apollo robot system.

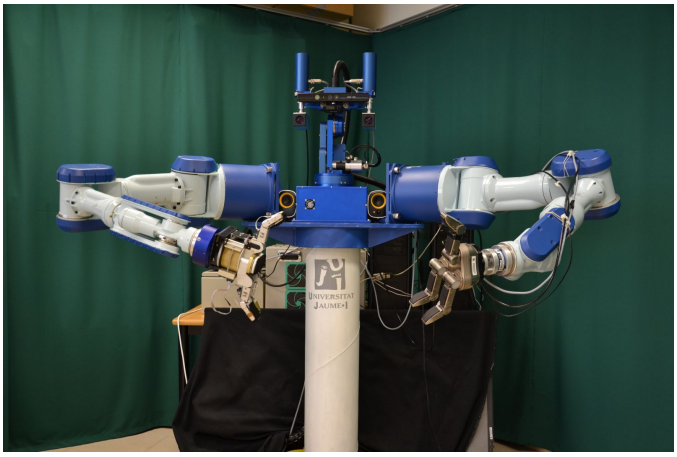


Figure 4: The Tombatossals robot system.

4.3. Objects

Our experiments with the Apollo platform consider up to 9 different physical objects. These objects are created

Table 3: Glossary

For a better understanding of the experiments and results, the main definitions used through the paper are detailed here:

Pose: a 7-dimensional vector containing a 3D vector $(x, y, z)^T$ representing location and a quaternion $(w, ax, ay, az)^T$ representing orientation.

Grasp: a pose, a robot gripper configuration and an object id. The pose indicates the position of the gripper relative to the object before closing its fingers. The gripper configuration determines the type of gripper and joint angles before starting the grasps.

Quality metrics (QM): a set of 7 independent metrics used to evaluate the quality of a grasp. They are computed using the simulator *Openrave* [40]. See Subsection 3.3 and Table 1 for a short description of them.

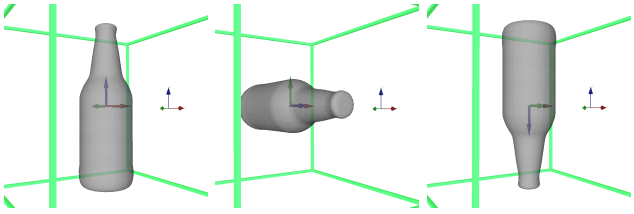
Physics metric score: obtained by evaluating a grasp using a dynamic simulation [18]. Its outcome is binary: *Stable*, *Unstable*. See Subsection 3.1.

Human-labelled grasp: grasp evaluated by human subjects on *Amazon Mechanical Turk*. The grasp is labelled with a binary value, *Stable* or *Unstable*. See subsection 3.2.

Candidate grasp: a grasp generated in simulation for which quality metrics are computed. In some cases, the physics metric and/or a human label has also been obtained and are part of the candidate grasp data.

Experimental grasp: a candidate grasp executed on a real robot following the protocol explained in Subsection 4.4. Each experimental grasp has an associated candidate grasp and a gravity vector. The unavoidable uncertainty when executing a grasp with a real robot causes the candidate and experimental grasps to be slightly different. Therefore, after each grasp execution the gripper configuration and object poses are recorded and then re-evaluated with OpenHand[40] to obtain their quality metrics (QM). See Subsection 4.4 for the full experimental protocol.

An experimental grasp consists of a candidate grasp, a gravity vector, a final gripper configuration and object pose, a set of QM values, and an experimental score.



(a) Gravity 0,0,-1 (b) Gravity 0,-1,0 (c) Gravity 0,0,1

Figure 1: Example of an object with three different gravity vectors. World frame is showed as a reference for the normal gravity vector (0,0,-1).

Experimental score: experimental grasps are assigned a score which consists of a binary value, *Stable* or *Unstable*. See Subsection 4.4.

Grasp cluster: a set of experimental grasps which have the same candidate grasp and the same gravity vector. A cluster typically contains from 3 to 5 experimental grasps.

three-tier experimental score: experimental grasps in the same cluster might have different experimental scores. If all the grasp in a cluster are scored as *Stable*, they are considered *Robust* experimental grasps; if all are scored *Unstable*, they are considered *Futile*; and if there is a mix of *Stable* and *Unstable* scores in a cluster, they are considered *Fragile*.

Gravity Vector: represents the direction of gravity relative to the object frame. For example, a bottle in its natural pose will have a gravity vector [0,0,-1]. If it is upside down it will be [0,0,1]. Each experimental grasp has an associated gravity vector.

Average Grasp: consists of the common candidate grasp, the gravity vector, and the average of the QM values of the experimental grasps contained in the cluster. It is also assigned an experimental score, which is the most frequent experimental score among the experimental grasps of the cluster. The candidate grasp may also be labelled with the physics metrics score and/or a human label.

The diagram in Fig. 2 summarizes the relationship between candidate grasps, experimental grasps, grasp clusters and average grasps.

KNN, CT, GP and NN: Abbreviations of the four classification approaches used in this paper: *KNN* stands for *K-Nearest Neighbors*, *CT* for *Classification Trees*, *GP* for *Gaussian Process* and *NN* for *Neural Networks*. Full description of them is given in the Appendix A.

Input Signal: a set or subset of quality metrics, object properties and gravity vector related to an experimental grasp. This input signal is used by the classification methods to train and test different prediction models.

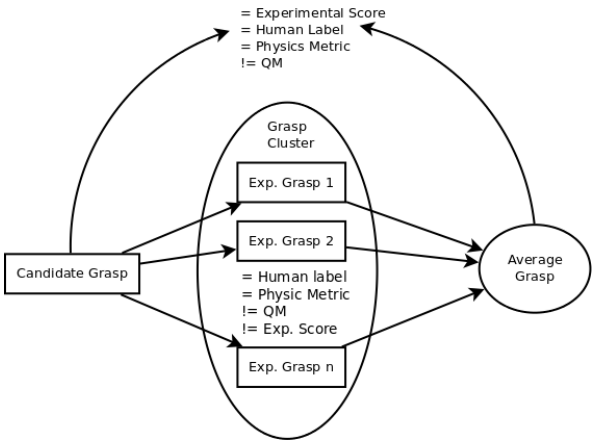
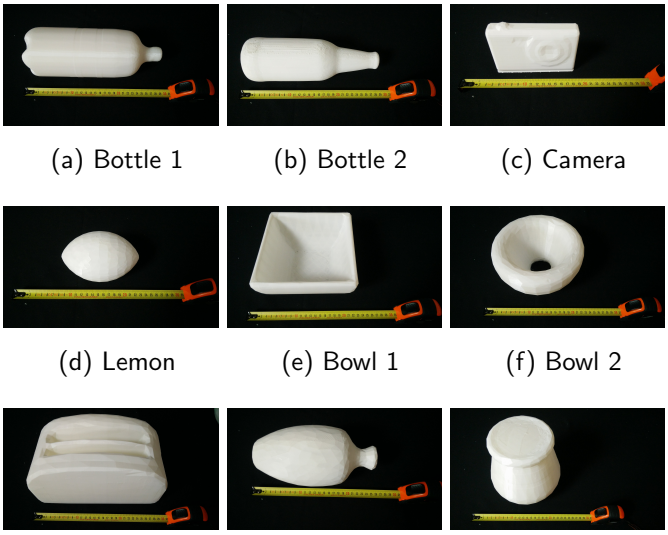


Figure 2: Relationships between candidate grasps, experimental grasps, grasp clusters and average grasps.

(a) Barrett hand (b) Schunk SDH hand

Figure 5: The two robotic grippers used to collect the experimental data.

using 3D printing technology and cover different weights, dimensions and shapes. Figure 6 shows the objects with their names.



(a) Bottle 1 (b) Bottle 2 (c) Camera

(d) Lemon (e) Bowl 1 (f) Bowl 2

(g) Toaster (h) Jar 1 (i) Jar 2

Figure 6: Object used for the experiments with the Apollo robot system. Subcaptions show the names of the objects.

In the experiments with Tombatossals we use only two different objects. Both objects were printed twice, using different percentages for the *infill*. This provided identical object shapes but with different weights (light and heavy). Objects printed with different weights are *bottle 1* and *toaster*.

4.4. Experimental protocol

To obtain the *experimental score* of a candidate grasp, it is necessary to evaluate it on a real robotic platform. For the purpose of the experiments, we use only one arm and one hand on each robotic platform. Prior to any experiment, a set of grasp candidates with a hand on a specific object has been simulated. The purpose of the experiment is to execute the candidate grasp on the real object with a real robot hand as precisely as possible.

To perform each experiment, we apply the following experimental protocol:

Step 1: place the object The target object is manually placed by a human operator on a table in front of the robot inside its reachable workspace.

Step 2: move the arm/gripper to an initial pose. The gripper is placed initially in a top-left position from the robot's point of view such that it does not occlude the objects.

Step 3: detect and track the object pose. The object is visually recognized and tracked during grasping approach using the Bayesian filtering methods implemented in [17]. In our experiment we employ the Particle Filter as proposed by [45].

Step 4: move the arm/gripper to the grasp target pose next to the object. During this step, there are two cases to consider: in the first case, the robot may correctly bring its hand to the grasp pose without disturbing the pose of the object. In that case, the experiment continues with the next step. In the second case, while aligning the robot hand with the desired grasp pose, the robot may hit the object during the movement, the object pose may be unstable prior to grasping, the motion planner may not have converged and the robot moves the hand to a wrong pose or the object tracking fails and loses the object. In those cases the execution is considered invalid and the procedure is aborted. The object is repositioned by the human operator on the table and a new execution is attempted. Figure 7 illustrates an example of an *Invalid* attempt.

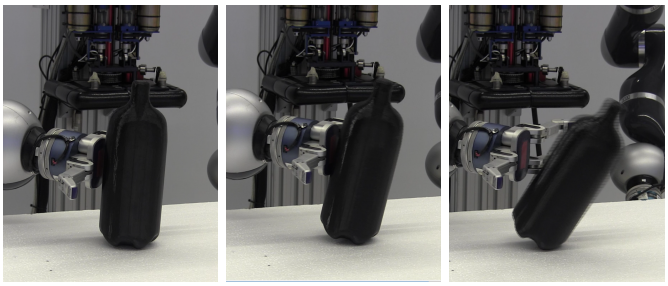


Figure 7: Example of a failed execution. The gripper pushes the object when approaching to the grasp pose.

Step 5: close the hand. The robot hand starts closing its fingers until a minimum strain or contact is detected. We determined strain and force thresholds on each joint to ensure the gripper applies enough pressure on the object and not just touches it. Tactile sensors are used in the *Tombatossals* platform and a strain gauge in the *Apollo* platform to detect such contacts.

While the fingers are closing, the object may also topple over, leading to a failed grasp. This execution is also considered invalid and the procedure is aborted. In some cases, the object moves when the fingers close but a grasp is still achieved. However, the resulting configuration of the gripper relative to the object may differ significantly from the attempted candidate grasp. In such cases, the human operator also aborts and restarts the procedure.

Step 6: move the gripper up. The joints of the fingers are locked and the arm moves up the gripper 15cm for small/medium objects. If the object falls or slips during this movement, the grasp is labelled as *Unstable*.

Step 7: hold the object for three seconds. The arm keeps the pose for 3 seconds. If the object remains in the gripper motionless for this time, the grasp is labelled *Stable*. If this time has expired and the object is not in the gripper, the grasp is considered *Unstable*. Figures 8 and 9 illustrates two examples of successful and unsuccessful experimental grasps. Both experimental grasps correspond to the same candidate grasp.

In the last two steps, the grasp execution is labelled by a human operator who continuously monitors the execution.

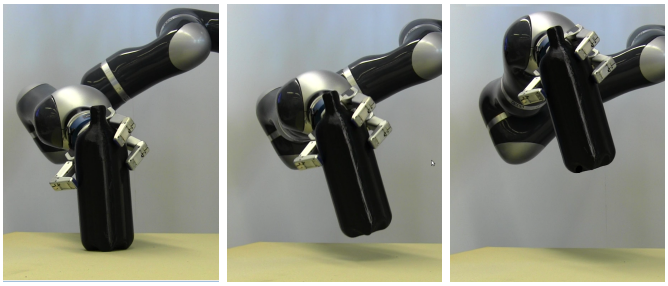


Figure 8: Example of a *stable* grasp execution.

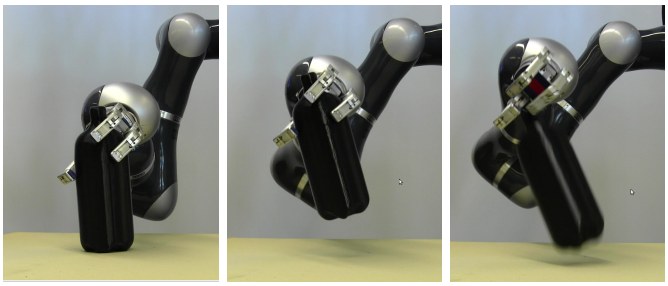


Figure 9: Example of an *Unstable* grasp execution.

Step 8: place down the object on the table. The arm moves down to the initial grasping pose. A margin of +2cm is applied in the *Z axis* to avoid the object hitting the table surface.

Step 9: release the object.

Step 10: move the arm/gripper back to the initial pose.

At the end of the protocol an experimental grasp is obtained and labelled as *Stable*/*Unstable*. Gripper configuration and object poses before and after closing the fingers are recorded as part of the experimental grasp data.

For every object, several candidate grasps are selected and tested using this protocol according to certain criteria. First of all, a candidate grasp must be feasible, that is, the object must be reachable by the robot arm/gripper without colliding with the table. In case that a candidate grasp is not reachable, it can be attempted on a different pose of the object. In any case, if possible, each selected candidate grasp will be tested on different poses of the object.

5. Experimental databases

Several experimental grasp databases have been collected using the experimental protocol explained in the previous section. To collect these databases, exhaustive tests on the available object models are carried out. For each object, several grasp candidates are selected and executed up to five times; if the execution results consistently in a *Stable* or *Unstable* result, the candidate grasp is attempted three times; if the results vary, the candidate grasp is executed five times. Each candidate grasp results in up to 5 experimental grasps, which conform a *grasp cluster*. In some cases, candidate grasps are used several times on the same object but having different gravity vectors (object poses), consequently forming different grasp clusters. Using this procedure two databases have been built:

Apollo experimental database. This database contains 1349 experimental grasps. These grasps were performed with the *Apollo* robotic platform and the *Barrett hand* on 9 different object models. On average, 50 candidate grasps are tested and 150 different experimental grasps are attempted with each object. Due to the exceptional cases in the experimental protocol, out of the 1349 experimental grasps attempted, 243 (18%) were not valid and, thus, discarded. Our final database contains 1106 *Stable/Unstable* experimental grasps: 830 of them had been previously evaluated with the physics metric, 600 were human labelled and 324 have both.

In addition to this, for 183 grasps the object's tracking procedure failed and it was not possible to record the final object and gripper configurations. For these grasps, the evaluation with *QM* using the final contact points was not possible. In total, 923 grasps were evaluated with quality metrics after their execution.

Tombatossals experimental database. This database contains experimental grasps executed with the *Tombatossals* robotic platform using the *Schunk SDH* gripper. Experimental grasps on this database are obtained from two different objects, each one printed twice, resulting in a light and heavy version. The purpose of this database

is to analyze the influence of the object's weight in the grasp success and whether the prediction, combined with *quality metrics*, can benefit from it. Up to 100 candidate grasps are evaluated using the above protocol.

This database contains more than 600 experimental grasps, distributed as shown in Table 4.

Table 4: Tombatossals experimental database

	Binary score		Three-tier score		
	Stable	Unstable	Robust	Futile	Fragile
Toaster	127	204	48	132	151
Bottle 1	156	129	78	39	168
Total	283	333	126	171	319

Dataset distribution of grasps and scores among different objects. The columns regarding the three-tier score will be described in subsection 7.3.

6. Reliability of physics and human labels

In previous works [18, 40], we employed a simplified physics simulation to annotate simulated grasps with success labels. We used humans as oracles of grasp success and analyzed the correlation between the labels generated by humans, physics, and a classic metric. We found that humans correlate much more strongly with the physics metric than with the classical metric. Assuming that humans are perfect oracles, we concluded that the physics metrics generates more realistic labels. However, we had not yet validated the metrics or humans labels against the real grasp outcomes. Humans evaluate grasps from simulated images and they do not receive any hint about object surfaces, object orientation or gravity. This section analyses how reliable the physics metric and the human labels from our previous work [18] are.

In this analysis, we use the *Apollo experimental database*, which contains grasps that are labelled with real grasp outcomes, human labels and the *physics metric*. The Apollo database contains many experimental grasps that are the result of executions of the same candidate grasp, we call them *grasp clusters*. In order to have a single experimental score for each candidate grasp, all the experimental grasps in a grasp cluster are summarized in a single *average grasp*. For the present analysis, an average grasp inherits the human label and physics metric score from the original grasp candidate. It is assigned the most common experimental score from all the experimental grasps in the same cluster.

Taking the above into account, the Apollo database consists of 343 average grasps, from which 105 have been labelled only by humans, 176 are scored only with the *physics metric*, and 62 are scored with both approaches. A total of 167 grasps has been scored by humans. This number is reduced to 89 because grasps that were scored by humans as *Unknown* (see section 3.2) cannot be matched as *Stable* or *Unstable* and are then discarded.

6.1. Results

Table 5 shows the results of the prediction accuracy of the three scores: experimental score, human labeling and physics metric labeling. To calculate each cell, the grasps that are labelled using both approaches are considered. The number represents the percentage of those grasps that has the same label, either *Stable* or *Unstable*. This number is the same as the accuracy.

A more detailed analysis of the predictive capability of Human Oracle and Physics metric is shown in Table 6.

Table 5: Predictive accuracy of Human oracle and the Physics Metric over experimental scores.

	Humans	Physics	Experimental
Humans	1.00		
Physics	**0.85**	1.00	
Experimental	0.61	0.64	1.00

For each cell, the number indicates the percentage of grasps that have the same label, either *Stable* or *Unstable*. For example, in the case of the *Human vs. Experimental* pairing, 61% of all the experimental grasps that have also been labelled by Humans have the same score. This would be equivalent to the accuracy of the Human labelling predicting experimental results.

Table 6: Score report on the accuracy of human labelling and *Physics metric.* to predict the grasp success on real experiments.

Experimental	Human Labelling			
Score	Precision	Recall	f1-Score	Support
Unstable	**1.00**	0.03	0.05	36
Stable	0.60	**1.00**	**0.75**	53
avg/total	**0.76**	0.61	0.47	89

Experimental	Physics Metric Scoring			
Score	Precision	Recall	f1-Score	Support
Unstable	**0.78**	0.53	0.63	129
Stable	0.55	**0.80**	0.65	94
avg/total	0.68	0.64	0.64	223

The upper table shows results for human labels. Lower table shows results for the physics metric. For the meaning of *Precision*, *Recall*, *f1-score* and *Support*, see Appendix A.5.

Finally, figure 10 illustrates an example of a grasp positively assessed by both, physics and human, evaluation, but unsuccessful in real world experiments.

6.2. Detailed discussion

Results show that neither the human labels nor the physics metric from [18] are good choices for accurately

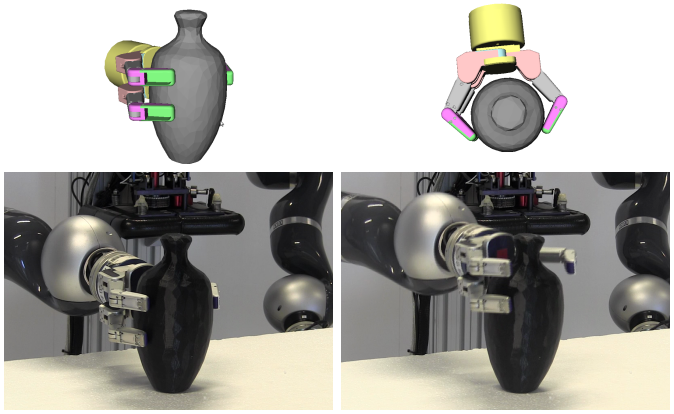


Figure 10: Comparison between Physics/human prediction and real grasp execution. The human oracle and physics metric classify this grasp as successful, but it does not work in real world. Above, images from simulation, below images recorded from real experiments.

predicting the outcome of real grasp executions. The human labels for the grasp candidates have 61% accuracy while the physics metric shows similar accuracy, 64%.

In the case of human labeling, the most relevant number is the low recall of the *Unstable* grasps. This indicates that human labelled as *Stable* most of the actually *Unstable*. In the few cases that a grasp was human labelled as *Unstable*, they were correct. These numbers indicate that human labeling is too "optimistic".

The presented results show a first evidence that providing grasp success labels is difficult even for human annotators. The reasons for this may be manifold. The simple physics environment used in [18] may not align well enough with the reality to accurately predict grasp success in the real world. Furthermore, the resulting grasps were presented to human annotators without physical reference, such as, for example, a supporting plane or a gravity direction. Another explanation might be related to the insufficient ability of the actual gripper to adapt to loose conditions. All this may explain the reduced correlation between real grasp outcomes and these two types of labels. This suggests that more careful analysis is required of what is required for humans or physics to produce more accurate predictions of grasp outcomes.

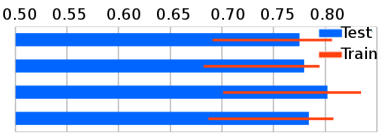
7. Quality metrics as stability predictors

The main question we address in this work is whether a single quality metric or a combination of them can be a good predictor of the execution of a grasp on a real robot. To answer, we make use of the large databases of experimental grasps that have been gathered on the Apollo and Tombatossals platforms.

In order to build prediction combinations we employ four classification and learning techniques (described in Appendix A) and consider several types of input vectors as classification vectors. The purpose is to explore which

Table 7: Classification results with random sampling of *Train* and *Test* sets

Clas.	CrossVal	Test
CT	0.75 ± 0.06	0.77
GP	0.74 ± 0.06	0.78
Knn	0.77 ± 0.07	0.80
NN	0.75 ± 0.06	0.78



The *Train* and *Test* set are sampled randomly. Table in the left shows the accuracy after training the classification methods and applying a cross-validation measurement. Graph in the right shows a blue column with the accuracy on the *Test* set and red line *Train* ± std. deviation of the accuracy.

combination of dataset, input vector and classification method is able to produce the best results.

For each analysis, the samples in each dataset are split between the *Train set* (around 80% of the samples), which is used to train the classification methods, and the *Test set* (around 20% of the samples), which is used to measure the classifier. In addition, the *Train* set is also split into sub-folders to perform a *cross-validation*.

7.1. Analyses and results

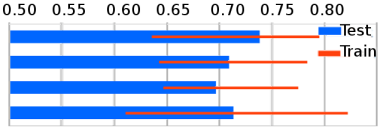
The first analysis consists in training the four classification methods with the full dataset of experimental grasps. The grasps contained in the *Test* and *Train* sets are selected randomly. The input vector for each experimental grasp is composed of the values of all the quality metrics, and the experimental score, which is the value that has to be predicted. Results are shown in table 7.

The data samples contained in the experimental dataset can be naturally grouped in clusters composed of experimental grasps derived from the same candidates grasps. Intuitively, it can be argued that these experimental grasps might be very similar, and thus the *Train* and *Test* are not fully independent and the trained classification methods can overfit the data. To analyze this effect we carry out an alternative approach for the training. In this case, the experimental grasps in the data set are grouped in grasp clusters. The *Train*, *Test* and folders for the cross-validation are composed of whole grasp clusters, which are selected randomly. Results are shown in table 8.

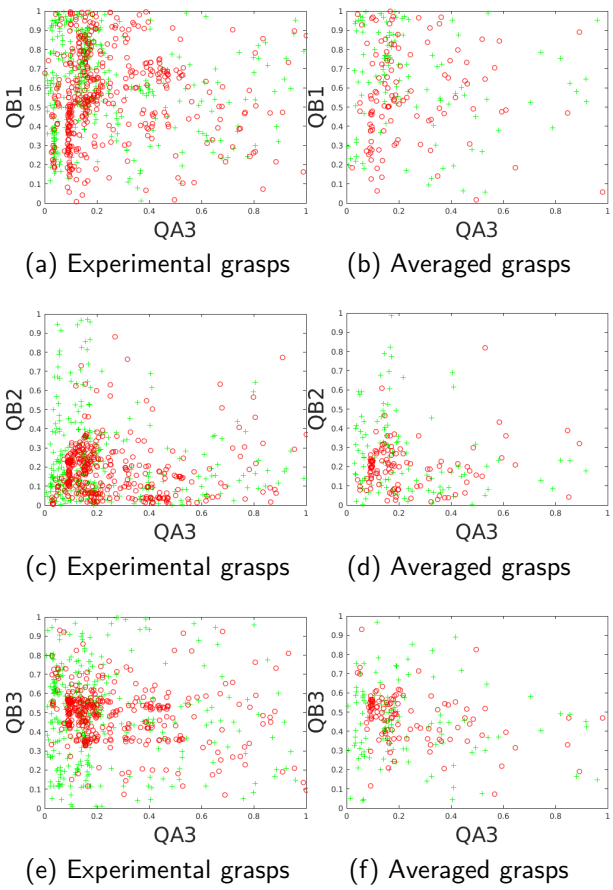
A slightly similar approach to the same problem mentioned above is to create a synthetic dataset composed of average grasps. An *average grasp* is a summary of the

Table 8: Classification results with cluster-based sampling of *Train* and *Test* sets

Clas.	CrossVal	Test
CT	0.71 ± 0.08	0.74
GP	0.71 ± 0.07	0.71
Knn	0.71 ± 0.06	0.70
NN	0.72 ± 0.11	0.71



The *Train* and *Test* set are sampled using clusters. Table in the left shows the accuracy after training the classification methods and applying a cross-validation measurement. The graph in the right shows a blue column with the accuracy on the *Test* set and a red line with the *Train* ± std. deviation of the accuracy.



(a) Experimental grasps (b) Averaged grasps

(c) Experimental grasps (d) Averaged grasps

(e) Experimental grasps (f) Averaged grasps

Figure 11: Maps of values for metrics Q_{A3} vs Q_{B1}, Q_{B2} and Q_{B3}. Plots on left columns show the data from experimental grasps dataset (923 grasps); plots on the right column show data from average grasps dataset (343 grasps).

experimental grasps in a cluster. It is composed of the averages of the values of the quality metrics of the experimental grasp in the cluster and the most frequent experimental score. This approach also addresses the possible problem mentioned above, but in a different way, since the averaging of the experimental scores filters out some detailed data. Many clusters are composed of both *Stable* and *Unstable* scores. When averaging, only one score is chosen as the representative score of a cluster. This produces a lot of information, but also reduces the amount of data necessary for training the classification method and helps to improve the categories in the data sets. The latter can be observed in the plots of figure 11, which show the scatter distribution of the experimental scores of the experimental and average grasps for several quality metrics.

Table 9 shows the results of training the classification methods using averaged grasps.

The previous analyses use the values of the seven selected quality metrics for each experimental grasp. The next analysis considers whether fewer quality metrics are able to provide similar performance to the use of all of them. The first analysis in this line is shown in table 10. It shows the performance of the CT classification method using a single quality metric. The *Test* and *Train* datasets are composed of experimental grasp selected at random, and the input vectors are composed of the value of the quality metric and the experimental score.

A further analysis seeks for the combination of quality metrics that provides the best performance results. Similarly to the previous experiment, the full dataset of experimental results are used, and *Train* and *Test* are selected randomly. For each combination of metrics the input vector consists of the values of these quality metrics and the experimental score. Table 11 show results of the combinations with the highest performance. On this occasion the *KNN* classification method is used.

7.2. Detailed discussion

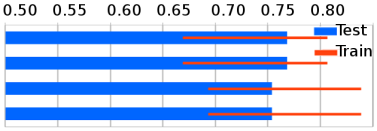
A first technical conclusion that can be drawn from the results is related to the performance of the classification methods. Tables 7, 8, and 9 do not provide any strong evidence of the superiority of any method. *KNN* and *CT* seem to provide slightly better results but the differences are not significant. To avoid unnecessary calculations, these former methods are used in the next analyses. The predictive accuracy varies between 70% and 80% depending on the classification metric and the *Train* or *Test* datasets.

Regarding the predictive capability of individual quality metrics, none of them individually reaches the performance of using all of them together (see table 10). Only the metric Q_{D1} is clearly superior to the others. However, when metrics are combined, the performance quickly improves. As table 11 shows combinations of 3 or 4 metrics that are able to obtain similar results to those using all the metrics (last line of the table). The accuracy of the predictions reaches a maximum around 80%.

There are several additional questions that need to be addressed to better limit the generality of our results. In the first place, there might be a risk of *over-fitting* in the classification methods. The results in table 8 are obtained removing whole grasp clusters from the *Train* test. This ensures that the methods are tested against experimental grasps corresponding to grasp candidates not considered in the *Train* test. The comparison with the results on table 7 shows that the results are slightly worse but still in the range error, so no *over-fitting* seems to be happening. These results also show that the trained classification

Table 9: Classification results with random sampling of *Train* and *Test* sets from averaged grasp dataset

Clas.	CrossVal	Test
CT	0.73 ± 0.08	0.83
GP	0.72 ± 0.08	0.74
Knn	0.74 ± 0.07	0.77
NN	0.77 ± 0.08	0.72



The *Train* and *Test* set are sampled randomly from the averaged grasp dataset. Table in the left shows the accuracy after training the classification methods and applying a cross-validation measurement. The graph in the right shows a blue column with the accuracy on the *Test* set and red line with the *Train* ± std. deviation of the accuracy.

Table 10: Classification performance of individual metrics

Metric	CrossVal	Test
Q_{A1}	0.62 ± 0.05	0.64
Q_{B1}	0.58 ± 0.03	0.60
Q_{B2}	0.60 ± 0.03	0.64
Q_{B3}	0.64 ± 0.04	0.66
Q_{C2}	0.58 ± 0.03	0.64
$\mathbf{Q_{D1}}$	**0.69 ± 0.07**	**0.72**
Q_{D2}	0.55 ± 0.06	0.59
QM	0.77 ± 0.07	0.80

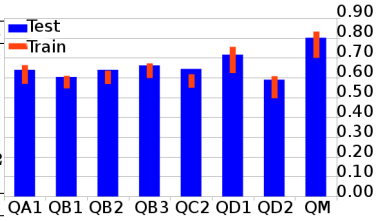


Table on the left shows the classification accuracy of each of the metrics using the *CT* classification method. It has been trained using the full experimental grasp dataset. The graph on the right shows the cross-validation average training accuracy ± std (red line) and the accuracy with the Test set (blue bar).

Table 11: Classification performance of combinations of quality metrics

Metrics Used	CrossVal	Test
(Q_{A1}, Q_{D1})	0.73 ± 0.07	0.78
(Q_{A1}, Q_{C2}, Q_{D1})	0.74 ± 0.07	0.81
(Q_{B1}, Q_{B3}, Q_{D1})	0.75 ± 0.03	0.80
(Q_{A1}, Q_{C2}, Q_{D1}, Q_{D2})	0.75 ± 0.05	0.81
(Q_{A1}, Q_{B1}, Q_{B3}, Q_{D1}, Q_{D2})	0.76 ± 0.06	0.81
(Q_{B1}, Q_{B2}, Q_{B3}, Q_{C2}, Q_{D1})	0.77 ± 0.04	0.81
(Q_{A1}, Q_{B1}, Q_{B2}, Q_{C2}, Q_{D1}, Q_{D2})	0.78 ± 0.04	0.81
(Q_{A1}, Q_{B1}, Q_{B2}, Q_{B3}, Q_{C2}, Q_{D1}, Q_{D2})	0.77 ± 0.07	0.80

Summary of best training results obtained using different combinations of metrics as input feature vector. In each line the table shows the best combination with two metrics, three metrics and so on. The classification method used is *KNN*.

methods are able to generalize to new grasp candidates.

When removing whole grasp clusters from the training set, an accuracy of around 70% is obtained. This would be the case when new grasp candidates, never seen during classification, are attempted. An 80% would be achieved when grasps similar to those used during the training are attempted.

A few other works study the combination of metrics and provide performance results of grasp executions. None of them, though, relate systematically and precisely the prediction made on virtual models with the results of the executed grasps on a real robot. In the most related one, [15], human operators guide the robot hand to stable grasp configurations from the operators point of view, which then executed on the real robot. These grasps are, then, recorded and quality metrics are computed on a simulation with the virtual model of the objects. For their analyses, quality metrics are combined using a Principal Components Analysis and a *Gaussian Process* is used for training a classification model.

Their results showed that combining metrics improve performance with respect a single metric up to a 66%. And the trained model showed a classification performance of 56% of True Positve Rate with a 15% of False Positive Rate. Other works reported a 76% [32] and 81% [42] success rate, but using a lot less ofexperimental cases and only using visual data from the target objects for grasp planning. It is unclear how these magnitudes could be compared to our results.

Finally, the question of whether the experimental database used was large enough is addressed in the results of table 9. These are obtained by training the classification with a dataset of averaged grasps which about a third of the size of the whole database. The results are similar to those in table 7. Hence, our conclusion is that the original database contained enough samples.

7.3. A three-tier experimental score

An important observation during the realization of the experiments is that there are candidate grasps that after being executed several times have been always *Stable*, and others *Unstable* regardless of the pose of the target object. And there exist a third type grasp clusters with a mix of *Stable* and *Unstable* experimental grasps. In order to handle this notion we propose a three-tier scoring: *Robust*, *Fragile* and *Futile* grasps.

A *Robust* experimental grasp is a grasp that belongs to a cluster where all of them have been scored as *Stable* grasps; a *Futile* experimental grasp belongs to a cluster where all of them have been scored as *Unstable*; and a *Fragile* grasp belongs to grasp cluster with *Stable* and *Unstable* grasp.

We carry out an analysis to find out whether the classification methods described in the previous sections are able to discriminate grasps based on this three-tier scoring. For this analysis, we chose the *Tombatossals* experimental database. In this database, all experimental grasps hs been labelled with the binary experimental score. In order to carry out the analysis, experimental grasps scores are post-processed and all of them are assigned a three-tier score depending on the distribution of *Unstable* and *Stable* in their grasp clusters. The distribution of experimental grasp in these thew grades is described in table 4.

Table 12 reports the accuracy results of the classification methods after training. Only two methods are reported, *KNN* and *CT*, since they consistently offered the best results in our previous analysis. For this case, the *Train* and *Test* were built randomly selecting experimental grasps from the database. Table 13 reports a similar analysis but in this case whole-grasp clusters have been selected to be part of the *Train* and *Test* set. This allows testing the classification methods with grasp candidates that have not been included in the *Train* set.

Table 12: Classification results using binary score and three-tier scorem for random *Train* and *Test* datasets

	Binary Score		three-tier score	
Classifier	Train ± Std	Test	Train ± Std	Test
K-Nearest Neighbours	0.70 ± 0.08	0.72	0.83 ± 0.04	0.83
Classification Trees	0.68 ± 0.05	0.65	0.86 ± 0.04	0.86

The *Train* and *Test* datasets are created by randomly selecting experimental grasps from the *Tombatossals* database. Mean and standard deviation accuracy are computed applying a 10 fold-cross validation on the *Train* set. In the third column the accuracy of the trained classificator on the *Test* set is reported.

Table 13: Classification results using binary score and three-tier score for clustered *Train* and *Test* datasets.

	Binary Score		three-tier score	
Classifier	Train ± Std	Test	Train ± Std	Test
K-Nearest Neighbors	0.63 ± 0.06	0.59	0.75 ± 0.15	0.75
Classification Trees	0.70 ± 0.10	0.81	0.78 ± 0.11	0.81

The *Train* and *Test* datasets are created by randomly selecting whole grasps clusters from the *Tombatossals* database. Mean and standard deviation accuracy are computed applying a 10 fold-cross validation on the *Train* set. In the third column the accuracy of the trained classificator on the *Test* set is reported.

Finally, an analysis of the accuracy of classification methods trained with samples using a single quality metric is presented in table 14. In this case the data for the *Train* and *Test* datasets are selected randomly from the *Tombatossals* experimental database.

Table 14: Classification results of individual metrics using binary score and three-tiers score

	Binary Score		Three-tier score	
Metric	Train ± Std	Test	Train ± Std	Test
Q_{A1}	0.71 ± 0.05	0.69	0.67 ± 0.09	0.68
Q_{B1}	0.60 ± 0.07	0.48	0.55 ± 0.04	0.56
Q_{B2}	0.59 ± 0.06	0.52	0.54 ± 0.04	0.52
Q_{B3}	0.54 ± 0.06	0.45	0.53 ± 0.01	0.48
Q_{C2}	0.60 ± 0.04	0.56	0.55 ± 0.03	0.52
Q_{D1}	0.73 ± 0.06	0.66	0.72 ± 0.04	0.73
Q_{D2}	0.55 ± 0.07	0.52	0.51 ± 0.02	0.50

Train and *Test* datasets are created by randomly selecting experimental grasps from the *Tombatossals* database. *CT* has been used as classification method. Mean and standard deviation accuracy are computed applying a 10 fold-cross validation on the *Train* set. In the third column the accuracy of the trained classificator on the *Test* dataset is reported.

7.4. Discussion on the three-tiers score

Results on tables 12 and 13 show that classification methods trained with a three-tier experimental score show better accuracy than those trained with the binary score. It shows up to a 10% accuracy increase, 75% - 80% accuracy in the case of clustered datasets and up to 86% in the case of random datasets. This supports the claim that a three-tier scoring approach could be useful to identify those candidate grasps which are definitely good or bad.

In the case of using single metrics for the training, the results (table 14) do not show any significant improvement from a binary to a three-tier scoring. The conclusion is that single metrics are limited for distinguishing between good and bad grasps.

8. Conclusions

This paper has presented a series of analyses of the predictive capability of quality metrics. To achieve this, we have created a protocol to execute and score candidate grasps computed on simulation on a real robot platform that has been implemented on two robot setups. Following this protocol, two datasets of experimental grasps executed on the real robots, evaluated on simulation and labelled by human subjects, have been created.

These datasets have allowed us to address and carry out several analyses comparing their simulated and real executions. The results of these analyses are described in sections 6 and7. Each section includes a detailed discussion of the results. The most relevant conclusions are:

- A combination of quality metrics can predict with an accuracy up to 80% when presenting candidate grasps already experienced, or up to 70% if new grasps are presented.
- No individual metric can deliver such performance.
- A combination of 3 quality metrics is enough to achieve an 80% accuracy.
- A three-tier scoring strategy allows achieving an 85% accuracy.
- Grasps in our human-labelled database are not reliable to predict the success of a grasp. Future works should be careful when collecting human labeling for the purpose of prediction.

9. Limitations and future work

The presented work has some limitations. First, only two manipulators were used to perform the experiments and evaluate grasps. Generalizing the results on prediction models to other grippers should be done carefully, since results may vary. It is open to discussion whether the results using each or both grippers to obtain experimental grasps could lead to different prediction models. Most of the quality metrics, except Q_{D1} and Q_{D2}, are independent from gripper kinematics and depend on the contact points only. However the different kinematics of a gripper produces a different set of simulated candidate grasps. This has been be mitigated by the exhaustive generation of candidate grasps which covers the whole space of possibilities. In addition, when the grasps are executed, different experimental success rates could be obtained for different hands. The question of how the gripper designs produce different outputs and which parts of the design of a gripper could be modified to improve the results of experiments deserves definitively a deeper research.

Second, we used a reduced number of objects to perform the real experiments; although different shapes were tested, an extended study with more objects should be done. Third, the human-labelled datasets used were constrained by the goals and conditions in which they were obtained. In order to draw general conclusions about the predictive ability of human subjects, a more appropriate dataset should be collected following a careful protocol.

Fourth, an hypothesis that we considered at the beginning of the experiments is the influence of the physical

context, such as gravity and supporting surfaces. Some information regarding this contextual information was registered during the execution of the experimental grasps, i.e. the gravity vector. However, no conclusion could be drawn regarding the impact of contextual information because there was not enough data. An *ad hoc* experiment on the impact of context information should be done. The data used for this purpose in this work proved to be insufficient.

Fifth, the real grasps executions were restricted to an environment with a table holding the object prior to the grasp. Repeating these experiments in other environments with different restrictions or without restrictions is advisable. Finally, the prediction models were generated using a few different types of classification methods. A wider study with these data can be done using other algorithms or methods, as it could provide better results.

As future work, we suggest extending this study using other grippers and object models. It would provide a wider view of the predictive capability of combined or independent quality metrics. The generalization capability of the classification methods for different robot hands is still to be measured.

Also, it would be desirable to come up with new grasp metrics, that are able to achieve much higher success rates. A promising approach to obtain better metrics could be derived from real experimental data and learning approaches, in way similar to [28], bootstrapped by training on virtual environments.

Finally, a more ambitious experiment to answer the question of whether humans are good grasp stability oracles stills remains to be done.

Appendix A. Classification Methods

This appendix provides a brief description of the different methods used for classification, and in particular, the hyper-parameters required by each of them.

We aim to find a model $y = f(\mathbf{x}; \mathbf{w})$ that can predict binary grasp success y given an input feature vector $\mathbf{x}$ consisting of several grasp features, typically the values of the quality metrics calculated for that grasp configuration. Our approach is to learn a binary classifier from experimental data which minimizes the following equation:

$$\min_{\mathbf{w}} \sum_{(\mathbf{x},y)\in\mathcal{D}} 1 - l(f(\mathbf{x};\mathbf{w}), y) \tag{A.1}$$

where $\mathbf{w}$ denotes the parameter vector of the classifier and

$$l(f(\mathbf{x};\mathbf{w}), y) = \begin{cases} 1, & \text{if } f(\mathbf{x};\mathbf{w}) = y \\ 0, & \text{otherwise} \end{cases} \tag{A.2}$$

Every classifier considered in this chapter minimizes a loss in this setting. For various reasons, the exact loss formulations may vary per method, e.g. through different regularizers or by dropping the indicator function in order to get gradients.

Given an experimental data set $\mathcal{D}$, we train four different classifiers using SciKit-Learn [33]. The hyper-parameter search for each method is done using a grid search and cross-validation. Details are presented in each subsection. A more in-depth description of the classification methods used in this chapter can be found in [4].

Appendix A.1. k-Nearest neighbors

K-Nearest Neighbors (KNN) is a non-parametric approach. It makes no assumptions about the distribution that generates the data. Instead, it classifies a test data point based on the class membership of its K nearest neighbors in the feature space. To define distances, we assume that this space is Euclidean.

Formally, KNN uses Bayes theorem to model the posterior probability of a grasp being successful or not

$$P(y = c|\mathbf{x}) = K_c/K \tag{A.3}$$

where K_c is the number of data points belonging to class c among the K nearest neighbors of $\mathbf{x}$ and c corresponds to the grasps success category, i.e: *Stable* or *Unstable*. To classify a data point, we maximize this posterior distribution over the binary class labels.

K is the hyperparameter in this classification method. In general, a larger K suppresses the effects of noise but makes the classification boundaries less distinct. We used a validation set to find the optimal K for our data set. KNN is well suited for our relatively low-dimensional feature space. However, in its most basic form, the inference of this method does not scale well with the number of data points and the dimension of the feature space.

Two different weighting functions were considered during the training: *uniform* and *distance*. The former assigns uniform weights to each neighbor. The latter assigns weights proportional to the inverse of the distance from the query point. We tested a K ranging from 1 to 20. The best performance corresponded to a *distance* weighting with a value of 5 for K.

The *distance* weighting function is:

$$f(\mathbf{x};\mathbf{w}) = \sqrt{\sum_{1}^{k}(x_i - y_i)^2} \tag{A.4}$$

Appendix A.2. Classification trees

Classification Trees (CTs) in a non-parametric approach. A CT is a binary tree that divides the input feature space at each node j into two regions according to whether $x_i \leq \theta_j$ or $x_i > \theta_j$. θ_j is a parameter of the model. These subregions are independently subdivided further by moving down in the tree until a leaf node is reached.

Each leaf node of the tree encodes a region $\mathcal{R}_\tau$ in the feature space. This region contains N_τ training data points

associated with class labels y_n. Let us assume that a test data point with features $\mathbf{x}$ ended up in the leaf node corresponding to $\mathcal{R}_\tau$. The probability $P(y=c|\mathbf{x})$ is then the fraction of data points labelled with c in that region.

$$P(y=c|\mathbf{x}) = m_{c,\tau} \text{ with } m_{c,\tau} = \frac{1}{N_\tau} \sum_{(\mathbf{x}_n, y_n)\in\mathcal{R}_\tau} 1(y_n, c) \tag{A.5}$$

For the final classification of a test data point, we again maximize $P(y=c|\mathbf{x})$ over the class labels c.

To learn a CT, we need to find the optimal split parameters θ_j at each node of the tree. This is done by a standard greedy strategy where at each new node, we optimize a certain criteria over a set $\mathcal{S}$ of candidate pairs of input features x_i and thresholds θ_j. The criteria has to capture the purity of class labels within the sub-region $\mathcal{R}_\tau$ resulting from a candidate split of the input region. In this work, we chose the *Gini impurity* $g(\mathcal{R}_\tau)$:

$$\min_{s_\tau\in\mathcal{S}} g(\mathcal{R}_\tau) \text{ with } g(\mathcal{R}_\tau) = 1 - \sum_{c\in\{0,1\}} m_{c,\tau}^2 \tag{A.6}$$

The more nodes are added, the deeper the tree and therefore also the more complex the decision rule. To avoid overfitting, we have found a maximum tree *depth* $= 5$ using the validation set. We have further investigated using entropy as an impurity measure with no significant difference in the results.

Appendix A.3. Gaussian processes

A *Gaussian Process* (GP) is a probabilistic, non-parametric method defined as a collection of a finite number of random variables with a joint Gaussian distribution. In our case, this collection consists of the grasps in the training data $\mathcal{D} = \{\mathbf{x}, y\}_{\|\mathcal{D}\|}$. In the following, we summarize the data in the matrix $\mathbf{X}$ and the labels in the vector $\mathbf{y}$. A GP can be seen as a distribution over functions with a mean μ and covariance Σ. If we want to query the label y_0 of a test data point with feature vector $\mathbf{x}_0$, we have $f(\mathbf{x}_0) \sim N(\mu, \Sigma)$ with

$$\mu = k(\mathbf{x}_0, \mathbf{X})^T [K(\mathbf{X}, \mathbf{X}) + \sigma_M^2 I]^{-1} \mathbf{y} \tag{A.7}$$

$$\Sigma = k(\mathbf{x}_0, \mathbf{x}_0) - k(\mathbf{x}_0, \mathbf{X})^T [K(\mathbf{X}, \mathbf{X}) + \sigma_M^2 I]^{-1} k(\mathbf{x}_0, \mathbf{X}). \tag{A.8}$$

σ_M^2 is the variance of the noise on the target values.

The entries of the covariance matrix $K(\mathbf{X}, \mathbf{X})_{p,q}$ at row p and column q are defined based on a covariance function $k(\mathbf{x}_p, \mathbf{x}_q)$ with some hyperparameters θ. We use the squared exponential covariance function

$$k(\mathbf{x}_p, \mathbf{x}_q) = \sigma_l^2 exp(\frac{-(\mathbf{x}_p - \mathbf{x}_q)^T L^{-1} (\mathbf{x}_p - \mathbf{x}_q)}{2}) \tag{A.9}$$

where the hyperparameters are σ_l, the signal variance, and L, the identity matrix multiplied with the length scale l. We optimize these hyperparameters in the standard way by maximizing the marginal likelihood. To compute $P(y_0 = c|\mathbf{x}_0)$, we *squash* $f(\mathbf{x}_0)$ through the logistic function (A.10).

$$\sigma(a) = \frac{1}{1+e^{-a}} \tag{A.10}$$

Appendix A.4. Artificial neural networks

Artificial neural networks (ANN) are nonlinear functions that map from the input feature vector $\mathbf{x}$ to the binary labels y. They are organized in multiple layers. Except from the input layer, each layer takes the output of the previous layer as input, potentially transformed with some non-linear function. For a binary classification, the output of the last layer is transformed using the logistic function (A.10). To classify a test data point with, for instance, a two-layer network, we need to compute

$$P(y=c|\mathbf{x}) = \sigma\left(\sum j = 0^M w_{cj}^{(2)} h\left(\sum i = 1^7 w_{ji}^{(1)} x_i\right)\right) \tag{A.11}$$

where $h(\cdot)$ refers to the logsig transfer function of the output from the first layer. The superscript (1) and (2) indicates the different weights of each layer.

Up to 12 different algorithms and 100 hidden layers for training the *Neural Network* were tested using the Matlab Neural Network Toolbox[29].

The best one was Bayesian Regularization [14], with layer size 17. It relies on the Levenberg-Marquardt algorithm to learn the optimal weights $\mathbf{w}^{(l)}$ for each layer. The validation set is used for early stopping and manual network structure optimization.

Appendix A.5. Evaluation metrics for classifiers

To compare the different classification methods, the reporting uses the following measures:

Accuracy: Percentage of count of correct predictions against the number of total samples, i.e. the count of *true positives* (tp) and *true negatives* (tp).

Precision: Ratio $tp/(tp + fp)$ where fp is the number of false positives. Intuitively, it measures the capability of the classifier to avoid labeling a negative as a positive sample.

Recall: Ratio $tp/(tp+fn)$ where fn the number of false negatives. Intuitively, *recall* measures the capability of a classifier to find all the positive samples.

f1-score: Harmonic mean of precision and recall: $F1 = 2 * (precision * recall)/(precision + recall)$. It reaches its maximum at 1 and minimum at 0.

Support: The number of occurrences of each label.

Acknowledgments

This research was partly supported by Ministerio de Educación, Ciencia y Tecnología (Grant No. R31-2008-000-10062-0), by Ministerio de Ciencia e Innovación (DPI 2011-27846), by Ministerio de Economía y Competitividad (DPI 2014-60635-R and DPI 2017-89910-R), by Generalitat Valenciana (PROMETEO 2009-052, PROMETEOII 2014-028), and by Fundació Caixa Castelló-Bancaixa (P1-1B2011-54 and PI-1B2011-25).

References

[1] A. Aleotti and S. Caselli. Grasp recognition in virtual reality for robot pregrasp planning by demonstration. *Proceedings - IEEE International Conference on Robotics and Automation*, 2006:2801, 2006.

[2] R. Balasubramanian, L. Xu, P. D. Brook, J. R. Smith, and Y. Matsuoka. Physical human interactive guidance: Identifying grasping principles from human-planned grasps. *IEEE Trans. on Robotics*, 28(4):899–910, Aug 2012.

[3] Barrett Technology Inc. BarrettHand. `http://www.barrett.com/robot/products-hand.htm`.

[4] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387310738.

[5] Jeannette Bohg, Antoinio Morales, Tamim Asfour, and Danica Kragic. Data-driven grasps synthesis - a survey. *IEEE Transactions on Robotics*, 30(2):289 – 309, April 2014.

[6] E. Boivin, I. Sharf, and M. Doyon. Optimum grasp of planar and revolute objects with gripper geometry constraints. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 1, pages 326 – 332, april-may 2004.

[7] K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige, S. Levine, and V. Vanhoucke. Using simulation and domain adaptation to improve efficiency of deep robotic grasping. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 4243–4250, May 2018. doi: 10.1109/ICRA.2018.8460875.

[8] E. Chinellato, A. Morales, R.B. Fisher, and A.P. del Pobil. Visual quality measures for characterizing planar robot grasps. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 35(1):30 –41, feb. 2005.

[9] Rosen Diankov. *Automated Construction of Robotic Manipulation Programs*. PhD thesis, Carnegie Mellon University, Robotics Institute, August 2010.

[10] Rosen Diankov and James Kuffner. Openrave: A planning architecture for autonomous robotics. Technical Report CMU-RI-TR-08-34, Robotics Institute, Pittsburgh, PA, July 2008.

[11] Dan Ding, Yun-Hui Lee, and Shuguo Wang. Computation of 3-d form-closure grasps. *IEEE Transactions on Robotics and Automation*, 17(4):515 –522, August 2001.

[12] Javier Felip, Angel J Durán, Marco Antonelli, Antonio Morales, and Angel P Del Pobil. Tombatossals: A humanoid torso for autonomous sensor-based tasks. In *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on*, pages 475–481. IEEE, 2015.

[13] C. Ferrari and J. Canny. Planning optimal grasps. *Proceedings 1992 IEEE International Conference on Robotics and Automation*, pages 2290–2295, 1992.

[14] F. Dan Foresee and M. T. Hagan. Gauss-newton approximation to bayesian learning. In *Neural Networks,1997., International Conference on*, volume 3, pages 1930–1935 vol.3, Jun 1997.

[15] Alex K. Goins, Ryan Carpenter, Weng-Keen Wong, and Ravi Balasubramanian. Evaluating the efficacy of grasp metrics for utilization in a gaussian process-based grasp predictor. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014) September 14-18, 2014, Chicago, IL, USA*. IEEE, 2014.

[16] R.D. Hester, M. Cetin, C. Kapoor, and D. Tesar. A criteria-based approach to grasp synthesis. In *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, volume 2, pages 1255–1260 vol.2, 1999.

[17] Jan Issac, Manuel Wüthrich, Cristina Garcia Cifuentes, Jeannette Bohg, Sebastian Trimpe, and Stefan Schaal. Depth-based object tracking using a robust gaussian filter. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) 2016*. IEEE, May 2016.

[18] Daniel Kappler, Jeannette Bohg, and Stefan Schaal. Leveraging big data for grasp planning. In *2015 IEEE International Conference on Robotics and Automation (ICRA) Washington State Convention Center Seattle, Washington, May 26-30, 2015*. IEEE, 2015.

[19] Daniel Kappler, Franzika Meier, Jan Issac, Jim Mainprice, Cristina Garcia Cifuentes, Manuel Wthrich, Vincent Berenz, Stefan Schaal, Nathan Ratliff, and Jeannette Bohg. Real-time perception meets reactive motion generation. *IEEE Robotics and Automation Letters*, 3(3):1864–1871, July 2018.

[20] Alexander Kasper, Zhixing Xue, and Rdiger Dillmann. The kit object models database: An object model database for object recognition, localization and manipulation in service robotics. *The International Journal of Robotics Research*, 31(8):927 – 934, July 2012.

[21] Byoung-Ho Kim, Sang-Rok Oh, Byung-Ju Yi, and Il Hong Suh. Optimal grasping based on non-dimensionalized performance indices. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, volume 2, pages 949 –956, 2001.

[22] Jin-Oh Kim and Pradeep Khosla. Dexterity measures for design and control of manipulators. *Proceedings IROS Workshop on Intelligent Robots and Systems*, pages 758–763, 1991.

[23] Ian Lenz, Honglak Lee, and Ashutosh Saxena. Deep learning for detecting robotic grasps. *IJRR*, 2015.

[24] Sergey Levine, Peter Pastor, Alex Krizhevsky, Julian Ibarz, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *I. J. Robotics Res.*, 37(4-5):421–436, 2018. doi: 10.1177/0278364917710318.

[25] Z. Li and S.S. Sastry. Task-oriented optimal grasping by multifingered robot hands. *IEEE Journal of Robotics and Automation,*, 4(1):32 –44, February 1987.

[26] A. Liegeois. Automatic supervisory control of the configuration and behavior of multibody mechanisms. *IEEE Trans. Systems, Man, and Cybernetics*, 7(12):842–868, 1977.

[27] Jeffrey Mahler, Florian T. Pokorny, Brian Hou, Melrose Roderick, Michael Laskey, Mathieu Aubry, Kai Kohlhoff, Torsten Kroger, James Kuffner, and Ken 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 *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016.

[28] Jeffrey Mahler, Jacky Liang, Sherdil Niyaz, Michael Laskey, Richard Doan, Xinyu Liu, Juan Aparicio Ojea, and Ken Goldberg. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. In *Robotics: Science and Systems (RSS)*, 2017.

[29] MATLAB. *version 7.10.0 (R2010a)*. The MathWorks Inc., Natick, Massachusetts, 2010.

[30] Andrew T Miller and Peter K Allen. Examples of 3d grasp quality computations. In *Proceedingsof the IEEE International Conference on Robotics and Automation*, volume 2, pages 1240–1246. IEEE, 1999. ISBN 0780351800.

[31] B. Mirtich and J. Canny. Easily computable optimum grasps in 2-d and 3-d. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 739–747, May 1994.

[32] Antonio Morales, Eris Chinellato, Andrew H. Fagg, and Angel P. Pobil. Using experienve for assessing grasp reliability. *International Journal of Humanoid Robotics*, 01(04):671–691, 2004. doi: 10.1142/s0219843604000290.

[33] 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*, 12:2825–2830, 2011.

[34] Lerrel Pinto, Abhinav Gupta, The Robotics Institute, and Carnegie Mellon University. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *2016 IEEE International Conference on Robotics and Automation (ICRA) Stockholm, Sweden, May 16-21, 2016*. IEEE, 2016.

[35] Jean Ponce, Steve Sullivan, Attawith Sudsang, Jean-Daniel Boissonnat, and Jean-Pierre Merlet. On computing four-finger equilibrium and force-closure grasps of polyhedral objects. *The International Journal of Robotics Research*, 16(1):11–35, 1997.

[36] M. Popovic, G. Kootstra, J. A. Jorgensen, D. Kragic, and N. Krüger. Grasping unknown objects using an early cognitive vision system for general scene understanding. pages 987–994, Sept 2011. doi: 10.1109/IROS.2011.6094932.

[37] Mximo A. Roa and Ral Suárez. Grasp quality measures: review and performance. *Autonomous Robots*, pages 1–24, 2014. ISSN 0929-5593.

[38] C. Rubert, D. Kappler, A. Morales, S. Schaal, and J. Bohg. On the relevance of grasp metrics for predicting grasp success. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, pages 265–272, September 2017. doi: 10.1109/IROS.2017.8202167.

[39] Carlos Rubert, Daniel Kappler, Antonio Morales, Stefan Schaal, and Jeannette Bohg. On the relevance of grasp metrics for predicting grasp success. *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[40] Carlos Rubert, Beatriz León, Antonio Morales, and Joaquín Sancho-Bru. Characterisation of grasp quality metrics. *Journal of Intelligent & Robotic Systems*, 89(3):319–342, Mar 2018.

[41] J K Salisbury and J J Craig. Articulated hands: Force control and kinematic issues. *The International Journal of Robotics Research*, 1(1):4–17, 1982.

[42] A. Saxena, L. Wong, and A. Y. Ng. Learning grasp strategies with partial shape information. In *AAAI Conf. on Artificial Intelligence*, pages 1491–1494, 2008.

[43] K.B. Shimoga. Robot Grasp Synthesis Algorithms: A Survey. *Int. Jour. of Robotic Research*, 15(3):230–266, 1996.

[44] Jonathan Weisz and Peter K Allen. Pose error robust grasping from contact wrench space metrics. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 557–562, 2012.

[45] M. Wüthrich, P. Pastor, M. Kalakrishnan, J. Bohg, and S. Schaal. Probabilistic object tracking using a range camera. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3195–3202. IEEE, November 2013.