Optimizing for what matters: The Top Grasp Hypothesis

Daniel Kappler[1], Stefan Schaal[1,2] and Jeannette Bohg[1]

Abstract— **In this paper, we consider the problem of robotic grasping of objects when only partial and noisy sensor data of the environment is available. We are specifically interested in the problem of reliably selecting the *best* hypothesis from a whole set. This is commonly the case when trying to grasp an object for which we can only observe a partial point cloud from one viewpoint through noisy sensors. There will be many possible ways to successfully grasp this object, and even more which will fail. We propose a supervised learning method that is trained with a *ranking loss*. This explicitly encourages that the top-ranked training grasp in a hypothesis set is also positively labeled. We show how we adapt the standard ranking loss to work with data that has binary labels and explain the benefits of this formulation. Additionally, we show how we can efficiently optimize this loss with stochastic gradient descent. In quantitative experiments, we show that we can outperform previous models by a large margin.**

I. Introduction

Grasping unknown objects from partial and noisy sensor data is still an open problem in the robotics community. For objects with a known polygonal mesh model, experience databases can be built offline and serve as grasp look-up table once this object has been detected in the scene. In [19, 13, 8] it has been shown that in this case robust grasping and manipulation can be achieved by applying force control and exploiting constraints in the environment. However, to transfer successful grasps between different objects of which only partial and noisy information is known, remains a challenge.

There are many supervised learning approaches towards grasping. The majority of those formulate grasping as a problem of classifying a grasp hypothesis as either stable or unstable. A grasp hypothesis in this context is usually a grasp preshape, 6D pose of the gripper and the gripper joint configuration. Examples of such supervised methods include [12, 18, 16, 17, 20], to name just a few. For a more comprehensive overview, we refer to Bohg et al. [1]. These approaches commonly use a learning method that returns some confidence value for each query grasp hypothesis. Given these scores, the grasp with the highest score is typically selected for grasp execution, if it is reachable.

However, even though these methods select the best hypothesis of all candidates at query time, the underlying classification models have not been directly trained for this objective. Instead they are optimized for accurately predicting the binary labels of the entire training dataset. While for some subsets of data points separating postives from negatives may be easy to achieve, it generally can be very hard to achieve this separation for all data points. This is particularly a problem when training on datasets with noisy labels or where the employed feature representation is not rich enough to carry all the necessary information for making a decision.

Here, we argue that for grasping, we should be training models on subsets of data, where one subset may for instance represent all possible grasp hypotheses obtained from one viewpoint of the object. Furthermore, we should optimize an objective that rewards when the highest scoring training data point of such a set is also positive. For example, when considering a partial, segmented point cloud of an object, there exists a large set of potential grasps, most of which are not stable. The best scoring hypothesis within this set should correspond to a stable grasp. Such an objective is called a *ranking loss*. Thus far, only few grasp learning models in the literature consider this kind of objective. At first glance, this problem seems to default to standard classification for binary labeled data.

In this paper, we introduce a ranking formulation for grasp stability prediction for binary labeled data. There are three main differences of our problem formulation to typical ranking problems. First, our hypothesis set consists only of binary data, hence there is no inherent ranking between different examples other than the distinction between positive and negative hypotheses. Second, we want to optimize solely for the top-1 ranked hypothesis in a set of hypotheses, and we are not interested in the remaining order of hypotheses. Third, our resulting ranking score can also be interpreted as a score for classification, deciding whether or not the top-1 ranked hypothesis is a positive or negative one.

We show that this formulation outperforms large-capacity models such as *Convolutional Neural Networks* (CNNs) and *Random Decision Forests* (RDFs) trained on the same grasping dataset but optimized with a binary classification objective.

In the remainder of this paper, we review related work on ranking in general and for grasping in particular. This is followed by a discussion of classification versus ranking objectives. In Section IV the proposed ranking loss is described in detail. Details on how the model is optimized using *Stochastic Gradient Descent* (SGD) are given in Section V. This is followed by experiments in Section VI.

II. Related Work

Many different problems such as information retrieval, optimizing the click through rates, even multi-class classifi-

[1] Autonomous Motion Department at the Max-Planck-Institute for Intelligent Systems, Tübingen, Germany Email: first.lastname@tue.mpg.de

[2] Computational Learning and Motor Control lab at the University of Southern California, Los Angeles, CA, USA

cation problems can be formulated as ranking problems. The most common approach to learning how to rank is based on pair-wise classification. For instance, in order to rank documents, Herbrich et al. [6] proposed to use a hinge loss SVM formulation to learn regression on ordinal data. Another pair-wise formulation is based on a probabilistic loss, which can be optimized using gradient descent, and has been applied to information retrieval by Burges et al. [3], in connection with a neural network based function approximator. A common issue for pair-wise approaches is the biased data distribution, often violating the i.i.d. assumption. Cao et al. [4] addressed this issue with a list based loss, to learn to rank.

Ranking data is a fundamental problem and has been applied to various sub-problems in different domains. Lehmann et al. [15] proposed to speed up object search by reducing the number of expensive classifier evaluations by learning how to rank sets of hypotheses. In robotics, ranking has been used to learn to select footholds based on terrain templates by Kalakrishnan et al. [11], enabling robust walking over rough terrain. Data-driven approaches for grasp stability prediction are commonly formulated as binary classification problems, often due to the nature of the provided data labels. There are however a few examples that employ ranking. For example [7] iteratively improves a matching cost that is computed based on a library of labeled local shape templates. While the matching function does not change, the library is continuously extended and thereby the ranking of different grasp hypotheses changes over time. Finding the best fingertip placement on objects for object grasping has been identified as a ranking problem by Le et al. [14]. The authors manually label the training data with three different classes (Bad, Good and Very Good) instead of two. As a learning method, they employ a ranking SVM that optimizes a measure that prefers better scores for the top grasp candidates. Jiang et al. [9] presents an extension to this work with a different representation of the grasp but otherwise the same SVM-based ranking method. Our proposed approach differs from this line of work by being able to exploit binary labeled training data points to optimize a ranking loss. We re-formulate the loss function such that the best ranked grasp hypothesis is also positively labeled and train a CNN with this loss.

III. Grasp Stability Prediction: Classification versus Ranking

To the best of our knowledge data-driven learning methods for grasp planning are almost exclusively formulated as classification problems of the form:

$$\min_w \mathcal{R}(w) + \sum_{(x,y)\in\mathcal{D}} L(F(x;w), y) \tag{1}$$

Here, the target of the function approximator $F(x;w)$ is to predict the binary grasp stability $y \in \{1, -1\}$ for a feature representation x (e.g. 2D templates shown in Fig. 4), associated with a grasp preshape (6D pose and gripper joint configuration). The target objective is to optimize the parameters w, regularized by $\mathcal{R}$ (e.g. $||w||_1$ to have a sparse set of parameters) to achieve the minimal loss L (e.g. $\max(0, 1 - yF(x;w))$ hinge loss) on the training dataset $\mathcal{D}$, assuming that the test data distribution is similar to the training data distribution.

We argue that for grasp planning, classification is a sub-optimal objective. The canonical problem for grasp planning is to predict a successful grasp for a target object given an entire set of hypotheses. Such a hypothesis set can for example contain all possible grasp rectangles for a view of an object as provided by the Cornell dataset [16] or all possible templates extracted from a 3D point-cloud as in [12]. Obtaining all possible successful grasps is not necessary, since the goal is that the robot succeeds in grasping the object with the first grasp attempt. Therefore, the grasp stability prediction should be reformulated as a ranking problem, trying to robustly identify **one** successful grasp within each hypothesis set. Additionally it should provide a calibrated score to decide whether the best hypothesis is stable or not.

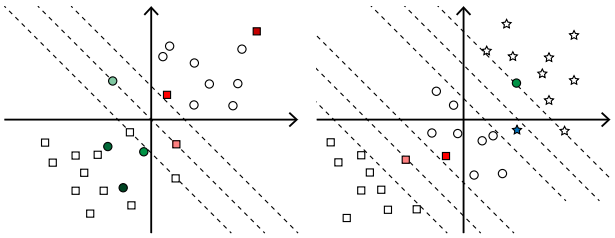
The canonical pair-wise ranking problem for different hypothesis sets (x, y) and $(x, y)'$, formulated as a standard classification problem, is illustrated in Fig. 1b and defined as

$$\min_w \quad \mathcal{R}(w) + \sum_{(x,y)} \sum_{(x,y)'} L(F(x;w) - F(x';w), \Delta(y, y')). \tag{2}$$

The main difference to the standard classification problem (Eq. 1) is the pair-wise classification and the pair-wise loss Δ. Notice this optimization problem can be specialized to the ranking SVM formulation proposed in [10] by applying the appropriate max-margin loss L. As shown in Fig. 1b, the ranking problem as described in Eq. 2 is concerned with ordering examples from different hypothesis sets according to the Δ loss. For typical grasp datasets, consisting of binary labeled grasp hypotheses, this ranking formulation would result in a solution similar to the binary classification problem (Eq. 1) up to a hypothesis set dependent scaling and offset. The scaling and offset are necessary since the ranking formulation is a relative problem. The remainder of this paper is concerned with a ranking formulation for binary hypothesis sets that allows top-1 prediction within the given hypthesis set as well as classification of that top-1 choice. We further propose a method to optimize such a problem formulation within the standard stochastic gradient descent optimization framework.

IV. Top-1 Ranking

This paper addresses the problem of optimizing a function that predicts **one** possible successful grasp within any given hypothesis set, if the hypothesis set contains **at least one** *stable* grasp. In addition to that, the resulting score has to be discriminative to classify whether or not the best predicted hypothesis is *positive* ($y = 1$) or *negative* ($y = -1$). In our work, hypothesis sets only contain binary labeled grasps, meaning a grasp is either considered *positive* (stable) or



(a) classification (b) ranking

Fig. 1: We illustrate the difference between the standard (a) max-margin classification problem and (b) pair-wise max-margin ranking problem. All symbols of the same shape are within the same hypothesis set. (a) Binary classification aims at separating these two sets. The magnitude of the error is indicated by the color saturation of the data samples where white means no error. Each set has its own color. The (b) ranking problem attempts to not only separate the 3 sets, but also maintains an order such that stars are always further to the top right than circles, and circles are further top right than squares. The resulting pair-wise classification problems illustrate the similarity of the ranking problem to the standard classification problem in (a).

negative (unstable). We assume no additional label information for the data which would allow to further discriminate between different examples in a set, e.g. if one *positive* grasp is better than another *positive* grasp. A concrete example for hypothesis sets is the grasp database introduced in [12]. In this particular example, every partially observed object is associated with a point cloud and several labeled grasp templates, where the grasp template takes the role of the feature representation x and the binary labels y indicate a stable or unstable grasp. Thus in this setting, a hypothesis set contains all pairs (x, y) available for a particular object view.

Every hypothesis set can either contain only positive examples, or only negative examples or both. To simplify notation we introduce three different index sets:

- $\mathcal{I}^+$ refers to all sets with **only** *positive* examples,
- $\mathcal{I}^-$ refers to all sets with **at least one** *negative* example,
- $\mathcal{I}^{+-}$ refers to all sets with **at least one** *positive* **and** *negative* example.

Every hypothesis set is assigned to at least one index set. Hypothesis sets with positive and negative examples are assigned to both $\mathcal{I}^{+-}$ and $\mathcal{I}^-$.

In the following we re-formulate and adapt the general ranking problem (Eq. 2) to the top-1 grasp prediction problem. We use a max-margin formulation with a margin ($t = 1$)

$$l(t - yk), \tag{3}$$

both for classification ($k = F(x; w)$) and ranking ($k = F(x; w) - F(x'; w)$). Here, we use the squared hinge loss $l(v) = \frac{1}{2}\max(0, v)^2$, since it is differentiable everywhere, a property that has been proven useful for stochastic gradient descent based neural network optimization [21].

Our proposed loss fuction is comprised of three parts – $L^{+-}(w)$, $L^{+}(w)$ and $L^{-}(w)$ operating on the previously introduced index sets $\mathcal{I}^{+-}$, $\mathcal{I}^{+}$ and $\mathcal{I}^{-}$, respectively. The goal of the first part of our loss, $L^{+-}(w)$, is to rank positive and negative hypotheses using a max-margin formulation, and is given as:

$$L^{+-}(w) = \sum_{i \in \mathcal{I}^{+-}} \sum_{x_i^+} \Big[l(1 - F(x_i^+; w)) + \sum_{x_i^-} l(1 - (F(x_i^+; w) - F(x_i^-; w)) \Big]$$

where x_i^- represents all negative and x_i^+ all positive hypotheses in the corresponding hypothesis sets in $\mathcal{I}^{+-}$. Notice, we obtain $l(1 - (F(x_i^+; w) - F(x_i^-; w))$ from Eq. 2, using Eq. 3 with $t = \Delta(y, y') = 1$ if y is a positive and y' a negative hypothesis. Furthermore, we ensure that positive examples get a calibrated score by adding the max-margin formulation $l(1 - F(x_i^+; w))$ for positive examples.

In the case of separable data we can rewrite L^{+-} to

$$L^{+-} = \sum_{i \in \mathcal{I}^{+-}} \sum_{x_i^+} \Big[l(1 - (F(x_i^+; w) - \max_{x_i^-} F(x_i^-; w)) + l(1 - F(x_i^+; w)) \Big] \tag{4}$$

If the data is separable, summing over all negative examples (as done in initial $L^{+-}(w)$) will result in the same loss value, as this max formulation. The second part of our loss, $L^{+}(w)$, operating on index set $\mathcal{I}^{+}(w)$, ensures that the prediction scores are calibrated in the same manner as positive examples in $\mathcal{I}^{+-}$, again by using the max-margin formulation:

$$L^{+}(w) = \sum_{i \in \mathcal{I}^{+}} \sum_{x_i^+} l(1 - F(x_i^+; w))$$

The third component $L^{-}(w)$, establishes that negative examples in the index set $\mathcal{I}^{-}$ are separated from positive ones to ensure the overall calibration of the score such that the final ranking score for a hypothesis set can be used for classification:

$$L^{-}(w) = \sum_{i \in \mathcal{I}^{-}} \sum_{x_i^-} l(1 + F(x_i^-; w))$$

Finally, we obtain the joint ranking and classification loss formulation

$$\min_w \left[L^{+-}(w) + L^{+}(w) + L^{-}(w) \right] \tag{5}$$

If our binary labeled training data, organized in hypothesis sets, is perfectly separable, this formulation will result in the same solution as the standard max-margin classification problem (Eq. 1). The pair-wise terms in Eq. 4 will vanish as soon as the two classes are perfectly separated. If the dataset is not separable, the pair-wise term will function as an additional loss on all positive examples within hypothesis sets for which the ranking loss cannot be fulfilled. This can be interpreted as a difference in importance of positive and negative misclassifications. However, this does not resolve the issue that the top-1 prediction might be a negative example, an illustration of that case is shown in Fig. 1a. The reason for misclassification might be the similarity to a positive example within a different hypothesis set. Hence, the perfect order/separation is still not achievable.

More concretely, let us assume that there exists a negative grasp which has an indistinguishable feature representation from several positive grasps in multiple sets. In this case multiple failure cases can occur. If this particular (negative) hypothesis is in the same hypothesis set as the indistinguishable positive hypotheses, the negative hypothesis can be picked at random. The reason for this is that the negative hypothesis achieves exactly the same score as the positive hypotheses and one hypothesis has to be selected based on this score. Another possibility is that this negative hypothesis is in a different hypothesis set than the indistinguishable positive hypotheses and no *easy* positive example exists for the function approximator $F(x; w)$ in the hypothesis set containing the negative hypothesis. Thus, this negative hypothesis will achieve the highest score.

In the following we present our approach to obtain a top-1 ranking problem despite the binary nature of the hypothesis sets. Since there are no label differences within the set of positive or negative hypotheses, we propose to use the induced difference by the function approximator itself. Thus, while optimizing the function approximator, the currently best positive and negative example, given the current function approximator prediction, is used for the pair-wise loss, resulting in:

$$\begin{aligned}\min_{w} \sum_{i \in \mathcal{I}^{+-}} \big[& l(1 - (\max_{x_i^+} F(x_i^+; w) - \max_{x_i^-} F(x_i^-; w)) + \\ & l(1 - \max_{x_i^+} F(x_i^+; w)) \big] + \\ & \sum_{i \in \mathcal{I}^{+}} l(1 - \max_{x_i^+} F(x_i^+; w)) + \\ & \sum_{i \in \mathcal{I}^{-}} \sum_{x_i^-} l(1 + (F(x_i^-; w)) \end{aligned} \tag{6}$$

Fig. 2 shows an example why this simple change to the optimization objective does achieve the top-1 ranking property for binary datasets. Intuitively, our formulation does not penalize any prediction for positive examples except for the current best positive and negative one in each hypothesis set. The best examples are determined by the current ranking of the latest function approximator parameterization. This ranking is not optimized by an explicit supervised quantity but it rather reflects the difficulty for the function approximator to distinguish positive from negative hypotheses. Hence, the function approximator has the ability to select one positive example in each hypothesis set, which contains at least one positive example, which is easy to separate from all negative examples. This change enables our formulation to ignore negative examples which are indistinguishable from positive hypotheses, as long as there exists at least one other positive hypothesis which is distinguishable. Notice, that we do not select these positive examples, but the optimization itself will determine these examples. Different learning methods for $F(x; w)$ therefore might result in different top-1 candidates.

This problem formulation enables automatic selection of positive top-1 examples which are easy to separate from negative examples. Indistinguishable examples under the implicit function approximator similarity measure, existing e.g. in different hypothesis sets, are not enforced to obtain a positive score any more.

To be more concrete, this behavior is useful for positive hypotheses for which important information, e.g. the surface points of an object, is not available, due to e.g. partial occlusion. In this scenario, the feature representation for the hypothesis might not contain enough information to distinguish this example from other negative examples. Using our ranking formulation (Eq. 6), the function approximator is not penalized if it assigns a low score to such examples, as long as there is another positive hypothesis in the set, for which the feature representation contains enough information to separate this example from all negative ones.

The pair-wise loss, solely applied to the two currently maximum examples of different class can be interpreted as a virtual target for the positive example. Alternatively, the pair-wise loss can be seen as a ranking problem on exactly two hypotheses (highest scoring positive and negative), selected by the score of the function approximator. The optimization tries to increase the score of that particular positive example to outperform the best negative one by a fixed margin (Eq. 3 and 6).

For each hypothesis set we have to solve at most two different problems. For hypothesis sets in $\mathcal{I}^{+-}$ the pair-wise loss and negative calibration are optimized. For hypothesis sets in $\mathcal{I}^{+}$, the best positive example is calibrated and for hypothesis sets in $\mathcal{I}^{-}$ all negative examples are calibrated. Despite the simple nature of these problems, obtaining an efficient optimization of Eq. 6 is not straight forward as discussed in the following section.

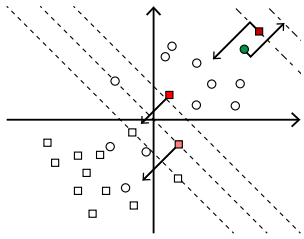


Fig. 2: This figure illustrates the proposed ranking objective applied to a single binary set of hypotheses. Squares represent negative examples and circles positive ones. The saturation of the color filling the shapes represents the error magnitude for each sample. The three dashed lines through zero represent the standard hinge loss. Notice that positive examples (circles) are not enforced to be separated but negative (squares) are. Since the current best hypothesis is a negative example, an additional classification problem for the best positive hypothesis is created, creating a *virtual* target higher than the current best negative example plus a margin. Arrows indicate the direction in which the optimization objective attempts to change the prediction scores.

V. Efficient First Order Optimization

The naive problem formulation as proposed in Eq. 6 could be optimized with first order batch gradient descent. However, this would not allow us to use large-scale databases such as [12]. The standard approach to optimizing a loss of the type (Eq. 1 and Eq. 6) for large datasets is to use mini-batch stochastic gradient descent. This makes each optimization step independent of the total number of available datapoints. Current state-of-the-art approaches such as

CNNs, which can exploit large datasets due to the large number of open parameters, also follow this optimization scheme. Usually n datapoints (x, y) are sampled uniform at random from the training dataset, constructing one mini-batch.

For our proposed loss, every mini-batch has to contain all positive examples of a hypothesis set due to the max operation. Notice this is only restricted to the positive examples. Using any subset of the negative examples which is already fulfilled would simply result in zero loss for the pair-wise terms. Thus the naïve approach for our loss would be to sample a hypothesis set uniform at random. All positive hypotheses of this set have to be in the mini-batch together with any subset of negative hypothesis. This process is continued until the mini-batch is filled with samples.

This naïve approach to construct the mini-batches for stochastic gradient descent has two main drawbacks. First, the number of positive examples would put a lower bound on the mini-batch size. Second, the majority of the computation would result in no improvement, since only the largest positive and negative example will be affected.

In the following we present our approach to overcome the limitations of the naïve approach.

A. Pair-Wise Loss Relaxation

As pointed out before, the max operation in the pair-wise term of our ranking loss Eq. 6, is the limiting factor to draw individual samples from each hypothesis set. Thus, next we show how to address this issue such that we can use stochastic gradient descent effectively.

Typical state-of-the-art methods for classification and regression such as *(Convolutional) Neural Networks* are global function approximators. Hence, every update of $F(x; w)$ can affect the prediction of any other data sample. We assume that $F(x; w)$ changes slowly for not affected values and more so for values for which gradients are applied. This is not a very restrictive assumption since we use stochastic gradient descent which requires to take small steps to converge. Using this assumption we can exploit that the $\max_{x_i} F(x_i; w)$ within a hypothesis set is unlikely to change very frequently. Thus, we propose rewrite the pair-wise term as two max-margin classification problems with a hypothesis set dependent margin t_i:

$$\begin{aligned}\min_w \sum_{i\in\mathcal{I}^{+-}} \big[& l(t_i^+ - \max_{x_i^+} F(x_i^+; w)) + \\ & l(t_i^- + \max_{x_i^-} F(x_i^-; w)) + \\ & l(1 - \max_{x_i^+} F(x_i^+; w))\big] + \\ \sum_{i\in\mathcal{I}^+} & l(1 - \max_{x_i^+} F(x_i^+; w)) + \\ \sum_{i\in\mathcal{I}^-} & \sum_{x_i^-} l(1 + F(x_i^-; w)) \end{aligned} \tag{7}$$

where $t_i^+ = 1 + \max_{x_i^-} F(x_i^-; w)$ is computed for each hypothesis set, as well as $t_i^- = 1 - \max_{x_i^+} F(x_i^+; w)$. The basic idea is to fix the maximum positive hypothesis for one hypothesis set to compute the corresponding margin for the negative hypothesis and vice versa. Instead of always evaluating the function approximator to obtain the true t_i, the last known prediction for every sample is used to update the estimates. This optimization problem will result in the same minimum as Eq. 6, if our assumption, that the maximum hypothesis for a particular hypothesis set does not change frequently, holds. Now, it is possible to draw individual samples from each hypothesis set.

Note however, the most informative examples are the best positive and negative examples. Other positive examples of a hypothesis set in $\mathcal{I}^{+-}$ do not contribute to the loss Eq. 7. Thus, to improve the loss the sample distribution over the hypothesis and hypothesis sets is not uniform but dependent on the loss and an additional term described in the following section.

B. Loss Optimization using Sampling

Random data sample selection is crucial for stochastic gradient descent based optimization. Yet, selecting data which most likely results in zero loss, thus zero gradients, simply slows down the optimization convergence. Using the previously introduced ranking loss Eq. 7, the problem with drawing sample hypotheses is to trade of the impact on the loss and the accuracy of the t_i estimation. The latter will ensure that the actual maximum of each hypothesis set is used to compute the loss and not an out of date estimate. Thus, we propose a heuristic to update the distribution for hypothesis sampling, which trades of the following two quantities (i) the error given the current loss (Eq. 7) and (ii) the iterations since the last update of the function evaluation of each data sample.

More concretely, after every function approximator evaluation we will update the prediction for the corresponding hypothesis and the iteration when the prediction was performed. For all hypothesis sets for which a hypothesis prediction was updated the estimates for the corresponding t_i^+ and t_i^- are updated and the loss based error for the hypothesis is updated. Notice, almost all hypothesis in a set have zero loss, since only negative and the maximum positive hypothesis are strictly enforced. If we normalize the error per hypothesis with the total error for all hypothesis, we obtain a distribution. Sampling hypotheses from this distribution will solely focus on improving the loss under the current t_i^+ and t_i^- estimates. Yet, due to the assumed global nature of the function approximator, we have to ensure that these estimates are still true. Therefore, we augment this error with an *artificial* error term that captures the number of iterations since the last update of a data point. It is of the following form:

$$e(c, u; o, b) = \exp(-t + (c - u)/b) \tag{8}$$

where c is the current iteration, u is the last update iteration of the example, o a trade-off parameter to determine the base influence of not evaluating, and b determines how fast the influence grows.

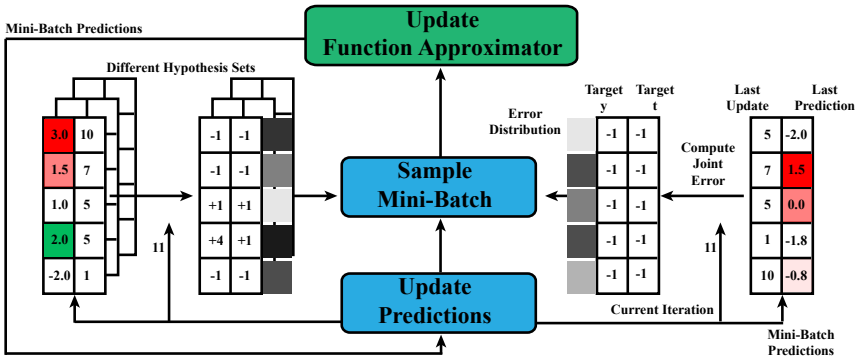


Fig. 3: This figure illustrates the general optimization loop, sampling a mini-batch, performing one function approximator update step, feeding back the latest prediction values and updating the error distribution. We show two exemplary sets of hypotheses, the one on the left contains positive and negative examples and the one on the right only negative ones. The gray value of the computed error distribution signals the importance for of this sample for the mini-batch sampling. Notice how the error due to the loss, indicated in red and green and the time since the last update affects the error distribution. The error distribution is normalized across all hypothesis sets and samples are drawn without replacement from the joint distribution.

Finally, after each optimization iteration the hypothesis predictions and loss errors are updated as previously described. In addition to that, we add the artificial iteration dependent error term Eq. 8 to the hypothesis error. The overall error for all hypothesis is normalized to get the discrete distribution from which we draw n samples (without replacement) to fill the new mini-batch. This means, hypotheses which have low influence on the loss are sampled very infrequently, basically not until Eq. 8 increases to a similar error magnitude as the maximum loss violating hypotheses. The maximum positive and negative hypothesis per hypothesis set are sampled more frequently if they do not fulfill the ranking loss. Fig. 3 illustrates the optimization loop for our proposed loss and mini-batch sampling.

VI. Experiments

A. Dataset

For evaluation we use a large scale dataset [12] which has been generated in OpenRave [5] by simulating numerous grasps on each of more than 700 distinct object mesh models. This dataset is split into 4 different subsets: a toy dataset containing only bottles, and three diverse sets of small, medium, and large objects. For our experiments we use the physics-metric proposed in [12] to automatically evaluate and label all the grasps. We binarize the dataset based on this metric $((y = 1 : p > 0.9), (y = -1 : p <= 0.9))$ with the same threshold as used for the evaluation within [12].

In addition to the grasps, the dataset also contains simulated point clouds that are reconstructed from multiple viewing angles distributed on a sphere around the object centroid. From each point cloud, a set of local shape templates is extracted that essentially encode object shape as seen from the hand (Fig. 4). Apart from object surface information, it also contains information about free and occluded space. Thus a template can be interpreted as an image with 3 color channels. The first channel represents the surface points of the object projected onto the plane spanned by the surface normal. The second channel represents the occluded space which is computed based on the viewpoint and the surface points. Points are again projected onto the same surface plane. Cells in the grid on the surface plane which are neither filled by surface points nor by occlusion points are marked as free space. Each template is linked to exactly two grasp poses that only differ in the initial distance between the palm of the hand and the object surface (the stand-off). The surface normal of a template is equal to the approach vector of the hand. One grasp can however be linked to multiple templates as its associated object surface normal may be visible from multiple viewpoints. An example template representation is shown in Fig. 4. This figure also visualizes different 3D versions of grasp templates for one grasp.

When the angle between the viewpoint and the surface normal is too big, the majority of the local shape information cannot be captured by the template representation, thus it is difficult for a learning method to discriminate these examples. The feature representation simply does not contain enough information to separate positive from negative examples under such conditions.

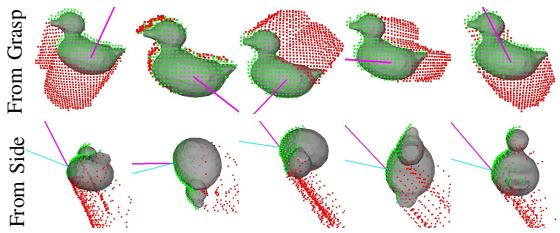


Fig. 4: Variation of the local shape representation given different viewpoints. The grasp for each of these templates is the same, i.e. approach direction along the cyan line and fixed wrist roll. The viewpoint is indicated by the pink line. Each column shows the same template from two different directions. (Top) Template viewed from the approach direction. (Bottom) Template viewed from the side. The occlusion area is the most affected by the varying viewpoint. Figure adopted from our previous work [12].

All templates extracted from one point cloud that are within a maximum angle between the surface normal of the object and the viewing point of the sensor frame, are grouped into one hypothesis set. Similar to [12], we reject templates with less than 30 surface points in the template.

B. Baselines

We compare the proposed method to two baseline models that are optimized for classification accuracy. The first one was already proposed in [12]. It is a simple CNN that consists of one convolution layer, a subsequent pooling layer and 3 fully connected ones, using a rectified linear unit as nonlinearity. The last nonlinearity is a sigmoid function to map to the binary grasp label. As input, it uses the same local shape template representation as described above.

As a second baseline, we use a Random Decision Forest that is trained to perform binary classification on this dataset [2]. As input to the model, it uses a set of randomly sampled probes for each information channel of the shape template and stacks it together into one feature vector. Both baseline models are very similar in classification performance.

C. Evaluation

A common use case in robotics is to select the best grasp for a given point cloud. Due to the nature of the dataset, the point cloud is already segmented to contain only points from the target object. In future work, we want to analyze how precise the target object point cloud segmentation has to be.

In Table 5 we evaluate the accuracy of the top-1 predictions. In this case, a true positive is a prediction for a hypothesis set from an object point cloud for which the highest scored hypothesis is classified positive and the ground truth label is positive. A true negative in this experiment is a prediction for a hypothesis set for which the highest ranked hypothesis is classified negative and there is no positive labeled hypothesis in this set. The scalar threshold for the classification prediction, based on the ranking score, is obtained by cross validation. We compare the performance of the proposed method with the two classification baselines. The results show that the proposed model trained on a ranking objective outperforms the two baselines by a large margin. For the dataset containing large objects, the performance is more than doubled. For the toy dataset of bottles the improvement is moderate. This is probably due to the simplicity of this subset of data where positive samples can be easily separated from negative ones. Notice that the datasets are highly unbalanced, meaning that the majority of the grasp hypotheses across all hypothesis sets are negative.

The results on the other datasets suggest that it is much harder to perfectly separate positive from negative data while it is easy to ensure that the top-ranking one refers to a stable grasp. This can be due to remaining label noise in the dataset where similarly looking templates can be either positive or negative.

In Fig. 6 we illustrate how our proposed sampling procedure (Section V-B) affects the sample usage for optimization, focusing on the difficult examples the most. This supports our hypothesis that during the course of the optimization of our proposed loss, the majority of the hypotheses are easy to address, resulting in low errors. However, every example is revisited due to the suggested heuristic to ensure that, despite changes to the parameters of the learning method, the error on these examples is still low.

	Bottles	**Small**	**Medium**	**Large**
data ratio	0.13	0.15	0.08	0.03
Forest	0.83	0.43	0.45	0.31
CNN	0.83	0.39	0.51	0.41
OURS	**0.85**	**0.59**	**0.70**	**0.84**

Fig. 5: We report the data ratio (all positive grasps divided by all grasps) for each test dataset and the top-1 score on the test dataset obtained by three different methods. The top-1 accuracy indicates the ratio of point clouds in the test data set for which the best scoring template was classified positive and also had a positive ground truth label or the best scoring template was classified negative and there was no positive ground truth example in the set. Results are reported per object group (bottles, small, medium, and large) and for gripper stand-off 0 from the object surface before closing the fingers. The proposed model that is trained on a ranking objective outperforms the baselines by a large margin. For large objects, the performance has more than doubled.

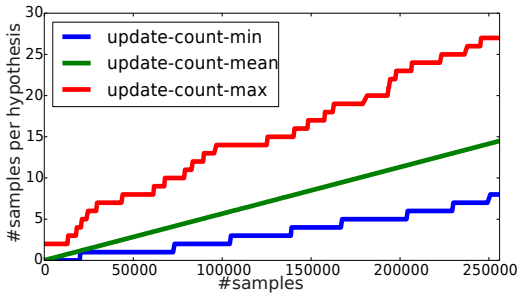


Fig. 6: This figure shows the influence of the error distribution based sampling for the optimization. The minimal update count (blue) illustrates that due to the error component based on the iterations, all data samples are revisited over time. However, the maximum update count (red) shows that the optimization is mostly focusing on the difficult hypotheses.

VII. Discussion and Conclusion

In this paper we have proposed to treat grasp prediction on sets of hypotheses as a ranking problem. An important distinction to other ranking approaches is that our method works for binary classification datasets, as long as the dataset is organized in sets of hypotheses, which is the typical case for grasp prediction. The experimental results support our hypothesis that the proposed ranking problem formulation significantly improves top-1 grasp stability prediction since difficult and ambiguous examples can simply be ignored by the function approximator. Another advantage of this formulation is that ambiguous and difficult examples are determined automatically by the optimization process. This is achieved by using the ranking of the function approximator at the particular moment of optimization. We believe that top-1 prediction is a better objective for grasp prediction, since perfect classification of all possible grasp hypotheses for a particular scene is unrealistic due to uncertainty in sensing and partial information in general. Even if the grasp predictor is trained with an optimization objective, one stable grasp has to be selected. In this case, most often the distance to the decision border of the classifier is used as a proxy to achieve a ranking within the positive predicted grasps. In this work we have shown that this proxy results in worse performance

compared to a grasp predictor which was optimized for ranking.

Conceptually the biggest drawback of the proposed approach is that we are solely optimizing for the top-1 grasp hypothesis. In the case that this hypothesis is not feasible due to e.g. kinematic or environmental constraints, the robot has to alter its position to either get a different view or make this grasp reachable, since no alternative prediction has a meaning for this set. Therefore, we believe an interesting extension of this approach is to optimize for top-n ranking as long as no other top-1 hypothesis performance is affected.

Another interesting extension to this work is to replace the heuristic for the hypotheses sampling, for mini-batch construction, by a stochastic non-stationary multi-armed bandit formulation. Such a formulation could further improve the optimization convergence.

REFERENCES

[1] J. Bohg, A. Morales, T. Asfour, and D. Kragic. Data-driven grasp synthesis: A survey. *IEEE Transactions on Robotics*, 2014.

[2] J. Bohg, D. Kappler, and S. Schaal. Exemplar-based prediction of global object shape from local shape similarity. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2016.

[3] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *In Proc. of Int. Conf. on Machine Learning (ICML)*. ACM, 2005.

[4] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li. Learning to rank: from pairwise approach to listwise approach. In *In Proc. of Int. Conf. on Machine Learning (ICML)*. ACM, 2007.

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

[6] R. Herbrich, T. Graepel, and K. Obermayer. Support vector learning for ordinal regression. 1999.

[7] A. Herzog, P. Pastor, M. Kalakrishnan, L. Righetti, J. Bohg, T. Asfour, and S. Schaal. Learning of grasp selection based on shape-templates. *Autonomous Robots*, 2014.

[8] N. Hudson, T. Howard, J. Ma, A. Jain, M. Bajracharya, S. Myint, C. Kuo, L. Matthies, P. Backes, P. Hebert, T. J. Fuchs, and J. W. Burdick. End-to-end dexterous manipulation with deliberate interactive estimation. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2012.

[9] Y. Jiang, S. Moseson, and A. Saxena. Efficient grasping from rgbd images: Learning using a new rectangle representation. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2011.

[10] T. Joachims. Optimizing search engines using clickthrough data. In *Int. Conf. on Knowledge Discovery and Data Mining (ACM)*, 2002.

[11] M. Kalakrishnan, J. Buchli, P. Pastor, and S. Schaal. Learning locomotion over rough terrain using terrain templates. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2009.

[12] D. Kappler, J. Bohg, and S. Schaal. Leveraging big data for grasp planning. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2015.

[13] M. Kazemi, J. Valois, J. A. Bagnell, and N. S. Pollard. Robust object grasping using force compliant motion primitives. In *Robotics: Science and Systems VIII*.

[14] Q. V. Le, D. Kamm, A. F. Kara, and A. Y. Ng. Learning to grasp objects with multiple contact points. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2010.

[15] A. D. Lehmann, P. V. Gehler, and L. J. Van Gool. Branch&rank: Non-linear object detection. In *BMVC*, 2011.

[16] I. Lenz, H. Lee, and A. Saxena. Deep learning for detecting robotic grasps. *IJRR*, 2015.

[17] R. Pelossof, A. Miller, P. Allen, and T. Jebera. An SVM learning approach to robotic grasping. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2004.

[18] J. Redmon and A. Angelova. Real-time grasp detection using convolutional neural networks. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2015.

[19] L. Righetti, M. Kalakrishnan, P. Pastor, J. Binney, J. Kelly, R. Voorhies, G. S. Sukhatme, and S. Schaal. An autonomous manipulation system based on force control and optimization. *Autonomous Robots*, 2014.

[20] A. Saxena, J. Driemeyer, and A. Y. Ng. Robotic grasping of novel objects using vision. *The Int. Jour. of Robotics Research (IJRR)*.

[21] Y. Tang. Deep learning using linear support vector machines. *arXiv preprint arXiv:1306.0239*, 2013.