2016 IEEE International Conference on Robotics and Automation (ICRA)
Stockholm, Sweden, May 16-21, 2016

Dex-Net 1.0: A Cloud-Based Network of 3D Objects for Robust Grasp Planning Using a Multi-Armed Bandit Model with Correlated Rewards

Jeffrey Mahler[1], Florian T. Pokorny[1], Brian Hou[1], Melrose Roderick[1], Michael Laskey[1], Mathieu Aubry[1], Kai Kohlhoff[2], Torsten Kröger[2], James Kuffner[2], Ken Goldberg[1]

***Abstract*— This paper presents the Dexterity Network (Dex-Net) 1.0, a dataset of 3D object models and a sampling-based planning algorithm to explore how Cloud Robotics can be used for robust grasp planning. The algorithm uses a Multi-Armed Bandit model with correlated rewards to leverage prior grasps and 3D object models in a growing dataset that currently includes over 10,000 unique 3D object models and 2.5 million parallel-jaw grasps. Each grasp includes an estimate of the probability of force closure under uncertainty in object and gripper pose and friction. Dex-Net 1.0 uses Multi-View Convolutional Neural Networks (MV-CNNs), a new deep learning method for 3D object classification, to provide a similarity metric between objects, and the Google Cloud Platform to simultaneously run up to 1,500 virtual cores, reducing experiment runtime by up to three orders of magnitude. Experiments suggest that correlated bandit techniques can use a cloud-based network of object models to significantly reduce the number of samples required for robust grasp planning. We report on system sensitivity to variations in similarity metrics and in uncertainty in pose and friction. Code and updated information is available at `http://berkeleyautomation.github.io/dex-net/`.**

I. Introduction

Cloud-based Robotics and Automation systems exchange data and perform computation via networks instead of operating in isolation with limited computation and memory. Potential advantages to using the Cloud include Big Data: access to updated libraries of images, maps, and object/product data; and Parallel Computation: access to grid computing for statistical analysis, machine learning, and planning [22]. Scaling effects have recently been demonstrated in computer vision and speech recognition, where learning from large datasets such as ImageNet has increased performance significantly [14], [24] over decades of previous research. Can analogous scaling effects emerge when datasets of 3D object models are applied to learning robust grasping and manipulation policies for robots? This question is being explored by others [12], [18], [27], [32], [33], and in this paper we present initial results using a new dataset of 3D models and grasp planning algorithm.

The primary contribution of this paper is the Dex-Net 1.0 algorithm for efficiently computing robust parallel-jaw grasps (pairs of contact points that define a grasp axis) with high probability of success based on a binary grasp quality metric with uncertainty due to imprecision in sensing and control. The algorithm speeds up robust grasp planning with Multi-Armed Bandits (MABs) [26] by learning from a large dataset of prior grasps and 3D object models using Continuous Correlated Beta Processes (CCBPs) [11], [31], an efficient model for estimating a belief distribution on predicted grasp quality based on prior data. This paper also presents Dex-Net 1.0, a growing dataset of 10,000 3D object models typically found in warehouses and homes such as containers, tools, tableware, and toys, and an implemented cloud-based algorithm for efficiently finding grasps with high probability of force closure (P_F) under perturbations in sensing and control.

Dex-Net 1.0 contains approximately 2.5 million parallel-

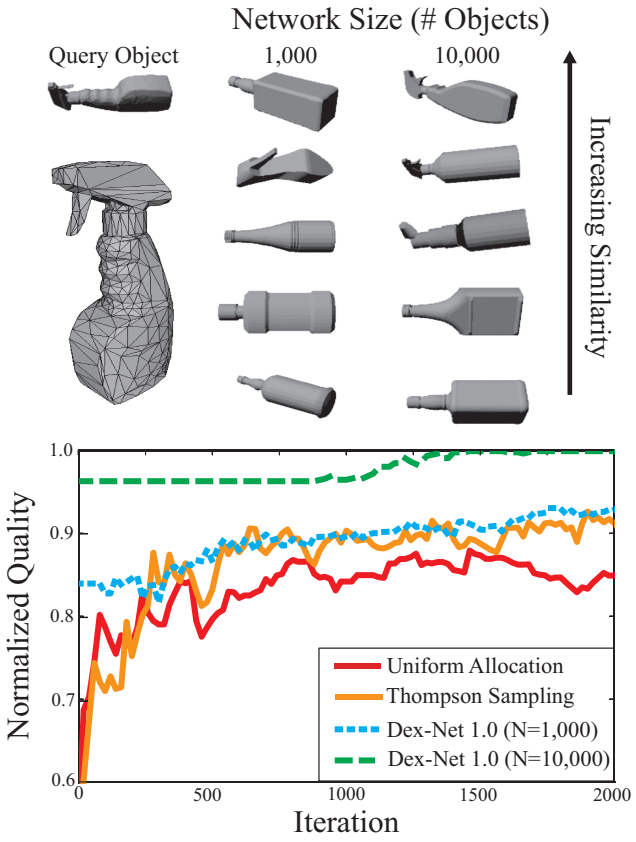

Fig. 1: Average normalized grasp quality versus iteration for 25 trials for the Dex-Net 1.0 Algorithm with 1,000 and 10,000 prior 3D objects from Dex-Net (bottom) and illustrations of five nearest neighbors in Dex-Net (top) for a spray bottle. We measure quality by the probability of force closure of the best grasp predicted by the algorithm on each iteration and compare with Thompson sampling without priors [26] and uniform allocation [20], [43]. (Top) The spray bottle has no similar neighbors with 1,000 objects, but two other spray bottles are found by the MV-CNN in the 10,000 object set. (Bottom) As a result, the Dex-Net 1.0 algorithm quickly converges to the optimal grasp with 10,000 prior objects.

[1] University of California, Berkeley, USA; {jmahler, ftpokorny, brian.hou, laskeymd, goldberg}@berkeley.edu, melrose_roderick@brown.edu, mathieu.aubry@enpc.fc
[2] Google Inc., Mountain View, USA; {kohlhoff, tkr, kuffner}@google.com

jaw grasps, as each object is labelled with up to 250 grasps and an estimate of P_F for each under uncertainty in object pose, gripper pose, and friction coefficient. To the best of our knowledge, this is the largest object dataset used for grasping research to-date. We incorporate Multi-View Convolutional Neural Networks (MV-CNNs) [42], a state-of-the-art method for 3D shape classification, to efficiently retrieve similar 3D objects for predicting robust grasp quality with CCBPs.

We implemented the Dex-Net 1.0 algorithm on Google Compute Engine and store Dex-Net 1.0 on Google Cloud Storage, with a system that can distribute grasp evaluations for 3D objects across up to 1,500 instances at once. Experiments suggest that using 10,000 prior object models from Dex-Net reduces the number of samples needed to plan parallel-jaw grasps by up to $2\times$ on average over 45 objects when using P_F as a success metric.

II. Related Work

Grasp planning considers the problem of finding grasps for a given object that achieve force closure or optimize a related quality metric, such as the epsilon quality [35], correlation with human labels [2], [18], or success in physical trials. Often it is assumed that the object is known exactly and that contacts are placed exactly, and mechanical wrench space analysis is applied. As computing quality metrics can be time consuming, a common grasp planning method is to store a database of 3D objects labelled with grasps and their quality and to transfer the stored grasps to similar objects at runtime [5], [41]. To study grasp planning at scale, Goldfeder et al. [12], [13] developed the Columbia grasp database, a dataset of 1,814 distinct models and over 200,000 force closure grasps generated using the GraspIt! sampling-based grasp planner. Pokorny et al. [34] introduced Grasp Moduli Spaces, enabling joint grasp and shape interpolation, and analyzed a set of 100 million sampled grasp configurations.

Robust grasp planning considers optimizing a grasp quality metric in the presence of bounded perturbations in properties such as object shape, pose, or mechanical properties such as friction, which are inevitable due to imprecision in perception and control. One way to treat perturbations is statistical sampling. Since sampling in high dimensions can be computationally demanding, recent research has studied labelling grasps in a database with metrics that are robust to imprecision in perception and control using probability of force closure (P_F) [43] or expected Ferrari-Canny quality [23]. Experiments by Weisz et al. [43] and Kim et al. [23] suggest that the robust metrics are better correlated with success on a physical robot than deterministic wrench space metrics. Brook et al. [7] planned robust grasps for a database of 892 point clouds and developed a model to predict grasp success on a physical robot based on correlations with grasps in the database. Kehoe et al. [21] created a Cloud-based system to transfer grasps evaluated by P_F on 100 objects in a database to a physical robot by indexing the objects with the Google Goggles object recognition engine.

Another line of research has focused on synthesizing grasps using statistical models [5] learned from a database of images [27] or point clouds [9], [15], [46] of objects annotated with grasps from human demonstrators [15], [27] or physical execution [15]. Kappler et al. [18] created a database of over 700 object instances, each labelled with 500 Barrett hand grasps and their associated quality from human annotations and the results of simulations with the ODE physics engine. The authors trained a deep neural network to predict grasp quality from heightmaps of the local object surface. In comparison, we predict P_F given a known 3D object model from similar objects and grasps using Continuous Correlated Beta Processes (CCBPs) and Multi-Armed Bandits (MABs).

Our work is also closely related to research on actively sampling grasps to build a statistical model of grasp quality from fewer examples [10], [25], [36]. Montesano and Lopes [31] used Continuous Correlated Beta Processes [11] to actively acquire grasp executions on a physical robot using image filters to measure similarity. Pinto et al. [33] used importance sampling over grasp success probabilities predicted from images by a Convolutional Neural Network (CNN) to actively acquire over 700 hours of labels for successful and unsuccessful 3 DOF crane grasps. Oberlin and Tellex [32] developed a budgeted MAB algorithm for planning 3 DOF crane grasps using priors from the responses of hand-designed depth image filters, but did not study the effects of orders of magnitude of prior data on convergence. In this work we extend the MAB model of [26] from 2D to 3D and study the scaling effects of using prior data from Dex-Net on robust grasp planning.

To use the prior information contained in Dex-Net, we also draw on research on 3D model similarity. One line of research has focused on shape geometry, such as characteristics of harmonic functions on the shape [6], or CNNs trained on a voxel representation of shape [30], [45]. Another line of research relies on the description of rendered views of a 3D model [12]. One of the key difficulties of these methods is comparing views from different objects, which may be oriented inconsistently. Su et al. [42] address this issue by using CNNs trained for ImageNet classification as descriptors for the different views and aggregating them with a second CNN that learns the invariance to orientation. Using this method, the authors improve state-of-the-art classification accuracy on ModelNet40 by 10%. We use max-pooling to aggregate views, similar to the average pooling of [1].

III. Definitions and Problem Statement

One goal of Cloud Robotics is to pre-compute a large set of robust grasps for each object so that when the object is encountered, the set can be downloaded such that at least one grasp is achievable in the presence of clutter and occlusions. In this paper we consider the sub-problem of efficiently planning a parallel-jaw grasp that maximizes the expected value of a binary quality metric, such as the probability of force closure (P_F), for a given 3D object model under perturbations in object pose, gripper pose, and friction coefficient. The Dex-Net 1.0 algorithm can also produce a set of grasps in ranked order of robustness. We assume the exact object shape is given

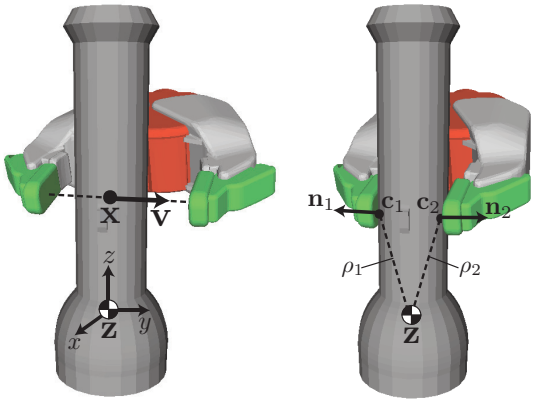

Fig. 2: Illustration of grasp parameterization and contact model. (Left) We parameterize parallel-jaw grasps by the centroid of the jaws $\mathbf{x} \in \mathbb{R}^3$ and approach direction, or direction along which the jaws close, $\mathbf{v} \in \mathbb{S}^2$. The parameters $\mathbf{x}$ and $\mathbf{v}$ are specified with respect to a coordinate frame at the object center of mass $\mathbf{z}$ and oriented along the principal directions of the object. (Right) The jaws are closed until contacting the object surface at locations $\mathbf{c}_1, \mathbf{c}_2 \in \mathbb{R}^3$, at which the surface has normals $\mathbf{n}_1, \mathbf{n}_2 \in \mathbb{S}^2$. The contacts are used to compute the moment arms $\rho_i = \mathbf{c}_i - \mathbf{z}$.

as a signed distance function (SDF) $f : \mathbb{R}^3 \to \mathbb{R}$ [29], which is zero on the object surface, positive outside the object, and negative within. We assume the object is specified in units of meters with given center of mass $\mathbf{z} \in \mathbb{R}^3$. Furthermore, we assume soft-finger point contacts with a Coulomb friction model [47]. We also assume that the gripper jaws are always opened to their maximal width $w \in \mathbb{R}$ before closing.

A. Grasp and Object Parameterization

The grasp parameters are illustrated in Fig. 2. Let $\mathbf{g} = (\mathbf{x}, \mathbf{v})$ be a parallel-jaw grasp parameterized by the centroid of the jaws in 3D space $\mathbf{x} \in \mathbb{R}^3$ and an approach axis $\mathbf{v} \in \mathbb{S}^2$. We denote by $\mathcal{O} = (f, \mathbf{z})$ an object with SDF f and center of mass $\mathbf{z}$, and denote by $\mathcal{S} = \{\mathbf{y} \in \mathbb{R}^3 \big| f(\mathbf{y}) = 0\}$ the surface of $\mathcal{O}$. We specify all points with respect to a reference frame centered at the object center of mass $\mathbf{z}$ and oriented along the principal axes of $\mathcal{S}$.

B. Objective

The objective of the Dex-Net 1.0 algorithm is to find a grasp $\mathbf{g}^*$ that maximizes an expected binary grasp quality metric $S(\mathbf{g}) \in \{0, 1\}$ such as force closure [23], [26], [29], [43] subject to uncertainty in the state of the object, environment, or robot. We refer to the expected quality as the probability of success, $P_S(\mathbf{g}) = \mathbb{E}[S(\mathbf{g})]$. Since sampling in high-dimensional spaces can be computationally expensive, we attempt to solve for $\mathbf{g}^*$ in as few samples T as possible by maximizing over the sum of $P_S(\mathbf{g}_t)$ for grasps sampled at times $t = 1, ..., T$ [26], [40]:

$$\underset{\mathbf{g}_1,..,\mathbf{g}_T \in \mathcal{G}}{\text{maximize}} \sum_{t=1}^{T} P_S(\mathbf{g}_t). \tag{III.1}$$

Past work has solved this objective by evaluating and ranking a discrete set of K candidate grasps $\Gamma = \{\mathbf{g}_1, ..., \mathbf{g}_K\}$ using Monte-Carlo integration [20], [43] or Multi-Armed Bandits (MAB) [26]. In this work, we extend the 2D MAB model of [26] to leverage similarities between prior grasps and 3D objects in Dex-Net to reduce the number of samples [16].

C. Quality Metric

In this work we evaluate our algorithm using the probability of force closure (P_F), or the ability to resist external force and torques in arbitrary directions [29], as a quality metric. P_F allows us to study the effects of large amounts of data on approximate solutions of Equation III.1 because it is relatively inexpensive to evaluate, and P_F has also shown promise in physical experiments [23], [43].

Let $F \in \{0, 1\}$ denote the occurrence of force closure. For a grasp $\mathbf{g}$ on object $\mathcal{O}$ under uncertainty in object pose ξ, gripper pose ν, and friction coefficient γ the probability of force closure $P_F(\mathbf{g}, \mathcal{O}) = \mathbb{P}(F = 1 \mid \mathbf{g}, \mathcal{O}, \xi, \nu, \gamma)$. To compute force closure for a grasp $\mathbf{g} \in \mathcal{G}$ on object $\mathcal{O} \in \mathcal{H}$ given samples of object pose $\hat{\xi}$, gripper pose $\hat{\nu}$, and friction coefficient $\hat{\gamma}$, we first compute a set of possible contact wrenches $\mathcal{W}$ using a soft finger contact model [47]. Then $F = 1$ if $\mathbf{0}$ is in the convex hull of $\mathcal{W}$ [43].

D. Sources of Uncertainty

For P_F evaluation we assume Gaussian distributions on object pose, gripper pose, and friction coefficient to model errors in registration, robot calibration, or classification of material properties. Let $\upsilon \sim \mathcal{N}(\mathbf{0}, \Sigma_\upsilon)$ denote a zero-mean Gaussian on $\mathbb{R}^6$ and $\mu_\xi \in SE(3)$ be the mean object pose. We define the object pose random variable $\xi = \exp(\upsilon^\wedge)\, \mu_\xi$, where the $\wedge$ operator maps from $\mathbb{R}^6$ to the Lie algebra $\mathfrak{se}(3)$ [3]. Let $\nu \sim \mathcal{N}(\mathbf{0}, \Sigma_\nu)$ denote zero-mean Gaussian gripper pose uncertainty with mean $\mu_\nu \in \mathcal{G}$. Let $\gamma \sim \mathcal{N}(\mu_\gamma, \Sigma_\gamma)$ denote a Gaussian distribution on the friction coefficient with mean $\mu_\gamma \in \mathbb{R}$. We denote by $\hat{\xi}, \hat{\nu}$, and $\hat{\gamma}$ samples of the random variables.

E. Contact Model

Given a grasp $\mathbf{g}$ on object surface f and samples $\hat{\xi}, \hat{\nu}$, and $\hat{\gamma}$, let $\mathbf{c}_i \in \mathbb{R}^3$ denote the 3D contact location between the i-th jaw and surface as shown in Fig. 2. Let $\mathbf{n}_i = \nabla f(\mathbf{c}_i) / \|\nabla f(\mathbf{c}_i)\|_2$ denote the surface normal and let $\mathbf{t}_{i,1}, \mathbf{t}_{i,2} \in \mathbb{S}^2$ be its tangent vectors. To compute the forces that each soft contact can apply to the object for friction coefficient $\hat{\gamma}$, we discretize the friction cone at $\mathbf{c}_i$ [35] into a set of l facets with vertices $\mathcal{F}_i = \left\{\mathbf{n}_i + \hat{\gamma} \cos\left(\frac{2\pi j}{l}\right) \mathbf{t}_{i,1} + \hat{\gamma} \sin\left(\frac{2\pi j}{l}\right) \mathbf{t}_{i,2} \middle| j = 1, ..., l\right\}$. Thus the set of wrenches that $\mathbf{g}$ can apply to $\mathcal{O}$ is $\mathcal{W} = \{\mathbf{w}_{i,j} = (\mathbf{f}_{i,j}, \mathbf{f}_{i,j} \times \rho_i) \big| i = 1, 2 \text{ and } \mathbf{f}_{i,j} \in \mathcal{F}_i\}$ where $\rho_i = (\mathbf{c}_i - \mathbf{z})$ is the moment arm at $\mathbf{c}_i$.

IV. Dexterity Network

The Dexterity Network (Dex-Net) 1.0 dataset is a growing set that currently includes over 10,000 unique 3D object models annotated with 2.5 million parallel-jaw grasps.

A. Data

Dex-Net 1.0 contains 13,252 3D mesh models: 8,987 from the SHREC 2014 challenge dataset [28], 2,539 from ModelNet40 [45], 1,371 from 3DNet [44], 129 from the KIT object database* [19], 120 from BigBIRD* [38], 80 from the Yale-CMU-Berkeley dataset* [8], and 26 from the Amazon

Picking Challenge* scans (* indicates laser-scanner data). We preprocess each mesh by removing unreferenced vertices, computing a reference frame with Principal Component Analysis (PCA) on the mesh vertices, setting the mesh center of mass $\mathbf{z}$ to the center of the mesh bounding box, and rescaling the synthetic meshes to fit the smallest dimension of the bounding box within $w = 0.1m$. To resolve orientation ambiguity in the reference frame, we orient the positive z-axis toward the side of the xy plane with more vertices. We also convert each mesh to an SDF using SDFGen [4].

B. Grasp Sampling

Each 3D object $\mathcal{O}_i$ in Dex-Net is labelled with up to 250 parallel-jaw grasps and their P_F. We generate K grasps for each object using a modification of the 2D algorithm presented in Smith et al. [39] to concentrate samples on grasps that are antipodal [29]. To sample a single grasp, we generate a contact point $\mathbf{c}_1$ by sampling uniformly from the object surface $\mathcal{S}$, sampling a direction $\mathbf{v} \in \mathbb{S}^2$ uniformly at random from the friction cone, and finding an antipodal contact $\mathbf{c}_2$ on the line $\mathbf{c}_1 + t\mathbf{v}$ where $t \geqslant 0$. We add the grasp $\mathbf{g}_{i,k} = (0.5(\mathbf{c}_1 + \mathbf{c}_2), \mathbf{v})$ to the candidate set if the contacts are antipodal [29]. We evaluated $P_F(\mathbf{g}_{i,k})$ using Monte-Carlo integration [20], [43] by sampling the object pose, gripper pose, and friction random variables $N = 500$ times and recording $Z_{i,k}$, the number of samples for which $\mathbf{g}_{i,k}$ achieved force closure ($F = 1$).

C. Depthmap Gradient Features

To measure grasp similarity in the Dex-Net 1.0 algorithm, we embed each grasp $\mathbf{g} = (\mathbf{x}, \mathbf{v})$ of object $\mathcal{O}$ in Dex-Net in a feature space based on a 2D map of the local surface orientation at the contacts, inspired by grasp heightmaps [15], [18]. We generate a depthmap $\mathbf{d}_i$ for contact $\mathbf{c}_i$ by orthogonally projecting the local object surface onto an $m \times m$ grid centered at $\mathbf{c}_i$ and oriented along the line to the object center of mass, $\mathbf{a}_i = \mathbf{z} - \mathbf{c}_i$. Since F only depends on $\mathbf{c}_i$ and its surface normal, rotations of $\mathbf{d}_i$ about $\mathbf{a}_i$ correspond to grasps of the equivalent quality. We therefore make each $\mathbf{d}_i$ rotation-invariant by orienting its axes along the eigenvectors of a weighted covariance matrix of the 3D surface points that generate $\mathbf{d}_i$ as described in [37]. Fig. 3 illustrates local surface patches extracted by this procedure. We finally take the x- and y-image gradients of $\mathbf{d}_i$ to form depthmap gradients $\nabla\mathbf{d}_i = (\nabla_x\mathbf{d}_i, \nabla_y\mathbf{d}_i)$, motivated by the dependence of F on surface normals [35], and we store each in Dex-Net 1.0.

V. Deep Learning for Object Similarity

We use Multi-View Convolutional Neural Networks (MV-CNNs) [42] to efficiently index prior 3D object and grasp data from Dex-Net by embedding each object in a vector space where distance represents object similarity, as shown in Fig. 4. We first render every object on a white background in a total of $C = 50$ virtual camera views oriented toward the object center and spaced on a grid of angle increments $\delta_\theta = \frac{2\pi}{5}$ and $\delta_\varphi = \frac{2\pi}{5}$ on a viewing sphere with radii $r = R, 2R$, where R is the maximum dimension of the object bounding box.

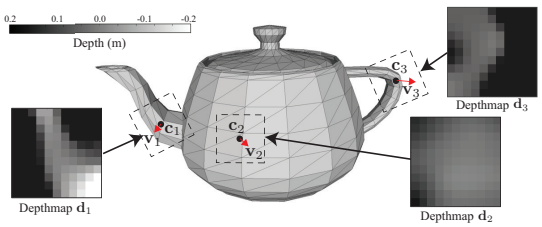

Fig. 3: Illustration of three local surface depthmaps extracted on a teapot. Each depthmap is "rendered" along the grasp axis $\mathbf{v}_i$ at contact $\mathbf{c}_i$ and oriented by the directions of maximum variation in the depthmap. We use gradients of the depthmaps for similarity between grasps in Dex-Net.

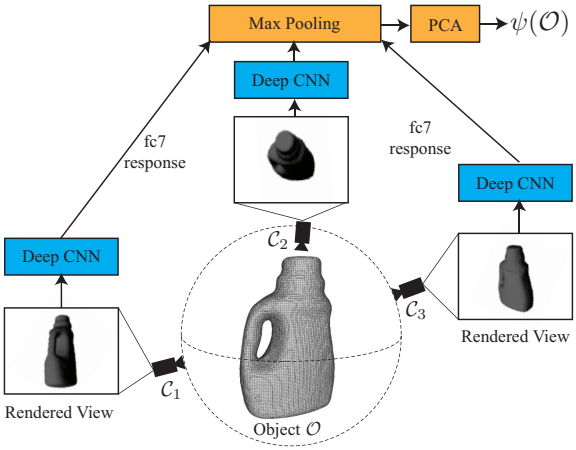

Fig. 4: Illustration of our Multi-View Convolutional Neural Network (MV-CNN) deep learning method for embedding 3D object models in a Euclidean vector space to compute global shape similarity. We pass a set of 50 virtually rendered camera viewpoints discretized around a sphere through a deep Convolutional Neural Network (CNN) with the AlexNet [24] architecture. Finally, we take the maximum fc7 response across each of the 50 views for each dimension and run PCA to reduce dimensionality.

Then we train a CNN with the architecture of AlexNet [24] to predict the 3D object class label for the rendered images on a training set of models. We initialize the weights of the network with the weights learned on ImageNet by Krizhevsky et al. [24] and optimize using Stochastic Gradient Descent (SGD). Next, we pass each of the C views of each object through the optimized CNN and max-pool the output of the fc7 layer, the highest layer of the network before the class label prediction. Finally, we use Principal Component Analysis (PCA) to reduce the max-pooled output from 4,096 dimensions to a 100 dimensional feature vector $\psi(\mathcal{O})$.

Given the MV-CNN object representation, we measure the dissimilarity between two objects $\mathcal{O}_i$ and $\mathcal{O}_j$ by the Euclidean distance $\|\psi(\mathcal{O}_i) - \psi(\mathcal{O}_j)\|_2$. For efficient lookups of similar objects, Dex-Net contains a KD-Tree nearest neighbor query structure with the feature vectors of all prior objects. In our implementation, we trained the MV-CNN using the Caffe library [17] on rendered images from a training set of approximately 6,000 3D models sampled from the SHREC 2014 [28] portion of Dex-Net, which has 171 unique categories, for 500,000 iterations of SGD. To validate the implementation, we tested on the SHREC 2014 challenge dataset and achieved a 1-NN accuracy of 86.7%, compared to 86.8% achieved by the winner of SHREC 2014 [28].

VI. Correlated Multi-Armed Bandit Algorithm

The Dex-Net 1.0 algorithm (see pseudocode below) optimizes the probability of success P_S (Equation III.1) for a binary quality metric such as force closure over a discrete set of candidate grasps Γ on an object $\mathcal{O}$ using a Bayesian Multi-Armed Bandit (MAB) model [26], [40] with correlated rewards [16] and priors computed from Dex-Net 1.0. We first generate the set of K candidate grasps using the antipodal grasp sampling described in Section IV-B and treat the grasps as "arms" in the MAB model. Next, we predict P_S for each grasp using the M most similar objects from the Dex-Net 1.0 dataset and estimate a Bayesian posterior distribution on our prediction. Then, for iterations $t = 1, ..., T$ we use Thompson sampling [26], [32] to select a grasp $\mathbf{g}_{t,k} \in \Gamma$ to evaluate, sample the quality $S(\mathbf{g}_{t,k})$, and update a posterior belief distribution on P_S for each grasp. Finally, we rank Γ by the q-lower confidence bound on P_S for each grasp and store the ranking in the database.

To illustrate convergence of the algorithm, we use force closure [47] as our binary quality metric. We plan to study other quality metrics such as success on physical trials [25], [31] and alternate MAB methods based on upper confidence bounds [25], [32] or Gittins indices [26] in future work.

A. Model of Correlated Rewards

Let $S_j = S(\mathbf{g}_j) \in \{0, 1\}$ be a random binary quality metric evaluated on grasp $\mathbf{g}_j \in \Gamma$. For example, S_j might model force closure under uncertainty in object pose, gripper pose, or friction. Each S_j is a Bernoulli random variable with probability of success $\theta_j = P_S(\mathbf{g}_j)$.

We use Continuous Correlated Beta Processes (CCBPs) [11], [31] to model a joint posterior belief distribution over the θ_j for all grasps in Dex-Net, which enables us to predict θ_j from prior grasp and object data in Dex-Net 1.0 using a closed-form posterior update. The joint distribution models pairwise correlations of θ between grasp-object pairs $\mathcal{P} = (\mathbf{g}, \mathcal{O})$ (points in a Grasp Moduli Space [34]) measured using a normalized kernel function $k(\mathcal{P}_i, \mathcal{P}_j)$ that approaches 1 as the arguments become increasingly similar and approaches 0 as the arguments become dissimilar.

Dex-Net 1.0 measures similarity using a set of feature maps $\varphi_m \in \mathbb{R}^{d_m}$ for $m = 1, ..., 3$, where d_m is the dimension of the feature space for each. The first feature map $\varphi_1(\mathcal{P}) = (\mathbf{x}, \mathbf{v}, \|\rho_1\|_2, \|\rho_2\|_2)$ captures similarity in the grasp parameters, where $\mathbf{x} \in \mathbb{R}^3$ is the grasp center, $\mathbf{v} \in \mathbb{S}^2$ is the approach axis, and $\rho_i \in \mathbb{R}^3$ is the i-th moment arm. The second feature map $\varphi_2(\mathcal{P}) = (\nabla \mathbf{d}_1, \nabla \mathbf{d}_2)$ uses the depthmap gradients described in Section IV-C. Our third feature map $\varphi_3(\mathcal{P}) = \psi(\mathcal{O})$ is the object similarity map described in Section V to capture global shape similarity.

Given the feature maps, we use the squared exponential kernel

$$k(\mathcal{Y}_p, \mathcal{Y}_q) = \exp\left(-\frac{1}{2}\sum_{m=1}^{3} \|\varphi_m(\mathcal{P}_p) - \varphi_m(\mathcal{P}_q)\|^2_{C_m}\right).$$

where $C_m \in \mathbb{R}^{d_m \times d_m}$ is the bandwidth for φ_m and $\|\mathbf{y}\|_{C_m} = \mathbf{y}^T C_m^{-1} \mathbf{y}$. The bandwidths are set by maximizing the log-likelihood [11] of the true θ on a set of training data.

B. Predicting Grasp Quality Using Prior Data

Before evaluating any grasps in Γ, the Dex-Net 1.0 algorithm predicts θ_j for each candidate grasp $\mathbf{g}_j$ based on its kernel similarity to all grasps and objects from the Dex-Net 1.0 dataset $\mathcal{D}$. In particular, we estimate a Bayesian posterior density $p(\theta_j)$ by treating $\mathcal{D}$ as prior observations and using the closed form posterior update for CCBPs [11]:

$$p(\theta_j \mid \alpha_{j,0}, \beta_{j,0}) \propto \theta_j^{\alpha_{j,0}-1}(1-\theta_j)^{\beta_{j,0}-1} \quad \text{(VI.1)}$$

$$\alpha_{j,0} = \alpha_0 + \sum_{i=1}^{|\mathcal{D}|}\sum_{k=1}^{K} k(\mathcal{P}_j, \mathcal{P}_{i,k}) Z_{i,k} \quad \text{(VI.2)}$$

$$\beta_{j,0} = \beta_0 + \sum_{i=1}^{|\mathcal{D}|}\sum_{k=1}^{K} k(\mathcal{P}_j, \mathcal{P}_{i,k})(N - Z_{i,k}) \quad \text{(VI.3)}$$

where α_0 and β_0 are prior parameters for the Beta distribution [26], N is the number of times each grasp $\mathbf{g}_{i,k} \in \mathcal{D}$ was sampled to estimate θ_i, and $Z_{i,k}$ is the number of observed successes for $\mathbf{g}_{i,k}$. Intuitively, the prior dataset contributes fractional observations of successes and failures for the grasp candidates Γ proportional to the kernel similarity. We estimate the above sums using the M nearest neighbors to $\mathcal{O}$ in the object similarity KD-Tree described in Section V.

C. Grasp Selection Policy

On iteration t we select the next grasp to sample $\mathbf{g}_j \in \Gamma$ using Thompson Sampling. In Thompson Sampling we draw a sample $\hat{\theta}_\ell \sim p(\theta_\ell \mid \alpha_{\ell,t}, \beta_{\ell,t})$ for each grasp $\mathbf{g}_\ell \in \Gamma$, then choose the grasp $\mathbf{g}_j$ with the highest $\hat{\theta}_j$ [26]. After observing S_j, we update the belief for all grasps $\mathbf{g}_\ell \in \Gamma$ by updating a running count of the fractional successes and failures [11]:

$$\alpha_{\ell,t} = \alpha_{\ell,t-1} + k(\mathcal{P}_\ell, \mathcal{P}_j) S_j \quad \text{(VI.4)}$$

$$\beta_{\ell,t} = \beta_{\ell,t-1} + k(\mathcal{P}_\ell, \mathcal{P}_j)(1 - S_j). \quad \text{(VI.5)}$$

VII. Experiments

We evaluate the performance of the Dex-Net 1.0 algorithm on robust grasp planning for varying sizes of prior data used from Dex-Net using force closure as our binary quality metric, and we explore the sensitivity of the convergence rate to object shape, the similarity kernel bandwidths, and uncertainty. We created two training sets of 1,000, and 10,000 objects by uniformly sampling objects from Dex-Net. We uniformly sampled a set of 300 validation objects for selecting algorithm hyperparameters and selected a set of 45 test objects from the remaining objects. We ran the algorithm for $T = 2,000$ iterations with $M = 10$ nearest neighbors, $\alpha_0 = \beta_0 = 1$ [26], and a lower confidence bound containing $q = 75\%$ of the belief distribution. We used isotropic Gaussian uncertainty with object and gripper translation variance $\sigma_t = 0.005$, object and gripper rotation variance $\sigma_r = 0.1$, and friction variance $\sigma_\gamma = 0.1$. For each experiment we compare the Dex-Net algorithm to Thompson sampling without priors

Input: Object $\mathcal{O}$, Number of Candidate Grasps K, Number of Nearest Neighbors M, Dex-Net 1.0 Dataset $\mathcal{D}$, Feature maps φ, Maximum Iterations T, Prior beta shape α_0, β_0, Lower Bound Confidence q, Quality Metric S
Result: Estimate of the grasp with highest P_F, $\hat{\mathbf{g}}^*$

```
// Generate candidate grasps and priors
```
Γ = AntipodalGraspSample($\mathcal{O}, K$) ;
$\mathcal{A}_0 = \varnothing, \mathcal{B}_0 = \varnothing$;
for $\mathbf{g}_k \in \Gamma$ **do**
```
    // Equations VI.2 and  VI.3
```
    $\alpha_{k,0}, \beta_{k,0}$ = ComputePriors($\mathcal{O}, \mathbf{g}_k, \mathcal{D}, M, \varphi, \alpha_0, \beta_0$);
    $\mathcal{A}_0 = \mathcal{A}_0 \cup \{\alpha_{k,0}\}, \mathcal{B}_0 = \mathcal{B}_0 \cup \{\beta_{k,0}\}$;
end
```
// Run MAB to Evaluate Grasps
```
for $t = 1, .., T$ **do**
    j = ThompsonSample($\mathcal{A}_{t-1}, \mathcal{B}_{t-1}$);
    S_j = SampleQuality($\mathbf{g}_j, \mathcal{O}, S$);
```
    // Equations VI.4 and  VI.5
```
    $\mathcal{A}_t, \mathcal{B}_t$ = UpdateBeta(j, S_j, Γ);
    $\mathbf{g}_t^*$ =MaxLowerConfidence($q, \mathcal{A}_t, \mathcal{B}_t$);
end
return $\mathbf{g}_T^*$;

Dex-Net 1.0 Algorithm: Robust Grasp Planning Using Multi-Armed Bandits with Correlated Rewards

(TS) [26], a state-of-the-art method for robust grasp planning, and uniform allocation (UA), a widely-used method for robust grasp planning that selects the next grasp to evaluate uniformly at random [20], [23], [43].

The inverse kernel bandwidths were selected by maximizing the log-likelihood of the true P_F under the CCBP model [11] on the validation set using a grid search over hyperparameters. The inverse bandwidths of the similarity kernel were $C_g = diag(0, 0, 3\times10^{-5}, 3\times10^{-5})$ for the grasp parameter features, an isotropic Gaussian mask C_d with mean $\mu_d = 500.0$ and $\sigma_d = 0.33$ for the differential depthmaps, and $C_s = 10^6 * I$ for the shape similarity features.

To scale experiments, we developed a Cloud-based system on top of Google Cloud Platform. We used Google Compute Engine (GCE) to construct the Dex-Net 1.0 dataset and to distribute subsets of objects to virtual machines for MAB experiments, and we used Google Cloud Storage to store Dex-Net. The system launched up to 1,500 GCE virtual instances at once for experiments, reducing the runtime by an estimated three orders of magnitude to approximately 315 seconds per object for both loading the dataset and running the Dex-Net 1.0 algorithm. Each virtual instance ran Ubuntu 12.04 on a single core with 3.75 GB of RAM.

A. Scaling of Average Convergence Rate

To examine the effects of orders of magnitude of prior data on convergence to a grasp with high P_F, we ran the Dex-Net 1.0 algorithm on the test objects with priors computed from 1,000 and 10,000 objects from Dex-Net. Fig. 5 shows the normalized P_F (the ratio of the P_F for the best grasp predicted by the algorithm to the highest P_F of the candidate grasps) versus iteration averaged over 25 trials for each of the 45 test objects to facilitate comparison across objects. The average runtime per iteration was 16 ms for UA, 17 ms for TS, and 22 ms for Dex-Net 1.0. The algorithm with 10,000

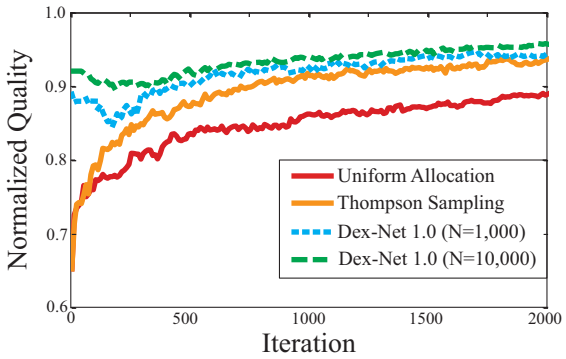

Fig. 5: Average normalized grasp quality versus iteration over 45 test objects and 25 trials per object for the Dex-Net 1.0 algorithm with 1,000 and 10,000 prior 3D objects from Dex-Net. We measure quality by the P_F for the best grasp predicted by the algorithm on each iteration and compare with Thompson sampling without priors and uniform allocation. The algorithm converges faster with 10,000 models, never dropping below approximately 90% of the grasp with highest P_F from a set of 250 candidate grasps.

objects takes approximately $2\times$ fewer iterations to reach the maximum normalized P_F value reached by TS, which is particularly promising for binary success metrics that are expensive to evaluate such as detailed physics simulations, human labels, or physical grasping trials. Furthermore, the 10,000 object curve does not fall below approximately 90% of the best grasp in the set across all iterations, suggesting that a grasp with high P_F is found using prior data alone. The maximum standard error of the mean over all iterations was 2×10^{-3} for UA, 2×10^{-3} for TS, and 1×10^{-3} for Dex-Net 1.0 with 1,000 and 10,000 objects.

B. Sensitivity to Object Shape

To understand the behavior of the Dex-Net 1.0 algorithm on individual 3D objects, we examined performance on a drill and spray bottle from the test set, both uncommon object categories in Dex-Net 1.0. Fig. 1 and Fig. 6 show the normalized P_F versus iteration averaged over 25 trials for 2,000 iterations on the spray bottle and drill, respectively. We see that the spray bottle converges very quickly when using a prior dataset of 10,000 objects, finding the optimal grasp in the set in about 1,500 iterations. This convergence may be explained by the two similar spray bottles retrieved by the MV-CNN from the 10,000 object dataset. Fig. 7 illustrates the grasps predicted to have the highest P_F on the spray bottle by the different algorithms after 100 iterations. On the other hand, performance on the drill does not improve using either 1,000 or 10,000 objects, perhaps because the closest model in Dex-Net according to the similarity metric is a phone.

C. Sensitivity to Similarity and Uncertainty

We also studied the sensitivity of the Dex-Net algorithm to the similarity kernel bandwidths described in Section VI-B and the levels of pose and friction uncertainty for the test object. We varied the inverse bandwidths of the kernel for the grasp parameters and depthmap gradients to the lower values $C_g = diag(0, 0, 15, 15)$, $\mu_d = 350.0$, and $\sigma_d = 3.0$ as well as the higher values $C_g = diag(0, 0, 300, 300)$, $\mu_d = 750.0$, and $\sigma_d = 1.75$. We also tested low uncertainty with variances

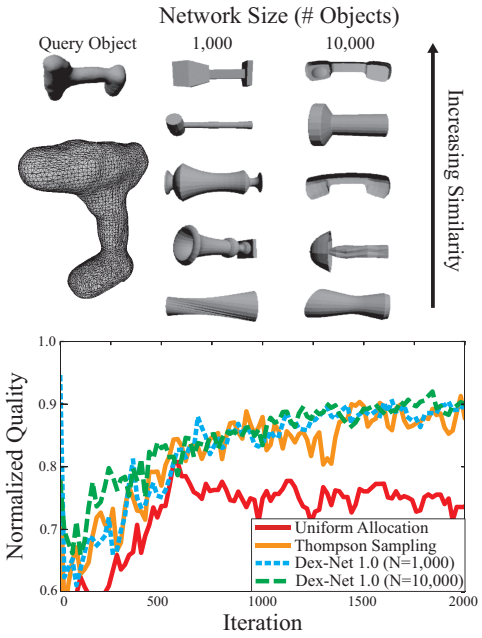

Fig. 6: Failure object for the Dex-Net 1.0 algorithm. (Top) The drill, which is relatively rare in the dataset, has no geometrically similar neighbors even with 10,000 objects. (Bottom) Plotted is the average normalized grasp quality versus iteration over 25 trials for the Dex-Net 1.0 algorithm with 1,000 and 10,000 prior 3D objects. The lack of similar objects leads to no significant performance increase over Thompson sampling without priors.

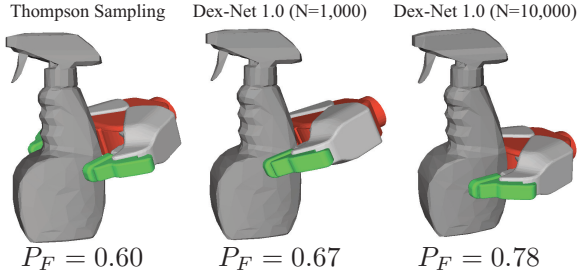

Fig. 7: Illustration of the grasps predicted to have the highest P_F after only 100 iterations by Thompson sampling without priors and the Dex-Net 1.0 algorithm with 1,000 and 10,000 prior objects on the spray bottle. Thompson sampling without priors chooses a grasp near the edge of the object, while the Dex-Net algorithm selects grasps closer to the object center-of-mass. For reference, the highest quality grasp for the spray bottle was $P_F = 0.81$.

$(\sigma_t, \sigma_r, \sigma_\gamma) = (0.0025, 0.05, 0.05)$ and high uncertainty with variances $(\sigma_t, \sigma_r, \sigma_\gamma) = (0.01, 0.2, 0.2)$. Fig. 8 shows the normalized P_F versus iteration averaged over 25 trials for 2,000 iterations on the 45 test objects. The results suggest that a conservative setting of similarity kernel bandwidth is important for convergence and that the algorithm is not sensitive to uncertainty levels.

VIII. Discussion and Future Work

We presented Dexterity Network (Dex-Net) 1.0, a new dataset and associated algorithm to study the scaling effects of Big Data and Cloud Computation on robust grasp planning with binary grasp quality metrics. The algorithm uses a Multi-Armed Bandit model with correlated rewards to leverage prior grasps and 3D object models. Experiments using the Google Cloud Platform suggest that prior data can speed robust grasp planning by a factor of 2 and that average grasp quality increases with the number of similar objects in the dataset.

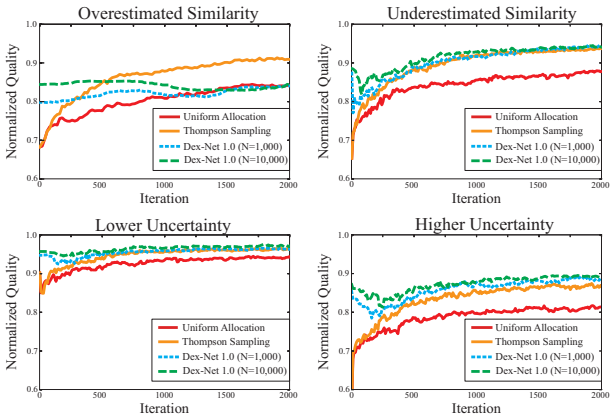

Fig. 8: Sensitivity to similiarity kernel (top) and pose and friction uncertainty (bottom) for the normalized grasp quality versus iteration averaged over 25 trials per object for the Dex-Net algorithm with 1,000 and 10,000 prior 3D objects. (Top-left) Using a higher inverse bandwidth causes the algorithm to measure false similarities between grasps, leading to performance on par with uniform allocation. (Top-right) A lower inverse bandwith decreases the convergence rate, but on average the Dex-Net algorithm still selects a grasp within approximately 85% of the grasp with highest P_F for all iterations. (Bottom-left) Lower uncertainty increases the quality for all methods, (bottom-right) higher uncertainty decreases the quality for all methods, and the Dex-Net algorithm with 10,000 prior objects still converges approximately $2\times$ faster than Thompson sampling without priors.

In future work, we will extend the Dex-Net 1.0 algorithm to computing a set of grasps that adequately "cover" each object from a variety of accessibility conditions depending on pose and clutter. We also plan to study Deep Learning [24] to better estimate grasp quality from prior object and grasp data. One shortcoming of the current work is our evaluation with a single quality metric. Thus, the next step is to perform physical experiments to evaluate the grasps found by Dex-Net 1.0, and to refine grasps using the Dex-Net 1.0 algorithm with physical success as a quality metric. We are experimenting with several robots and plan to share a subset of 3D objects and algorithms to facilitate comparison with related methods and experimentation with different robots and objects.

IX. Acknowledgments

This research was performed in UC Berkeley's Automation Sciences Lab under the UC Berkeley Center for Information Technology in the Interest of Society (CITRIS) "People and Robots" Initiative: http://robotics.citris-uc.org. The authors were supported in part by the U.S. National Science Foundation under NRI Award IIS-1227536, by grants from Google, UC Berkeley's Algorithms, Machines, and People Lab; the Knut and Alice Wallenberg Foundation; the NSF-Graduate Research Fellowship; and the Department of Defense (DoD) through the National Defense Science & Engineering Graduate Fellowship (NDSEG) Program. We thank our colleagues who wrote code for the project, in particular Raul Puri, Sahaana Suri, Nikhil Sharma, and Josh Price, as well as our colleagues who gave feedback and suggestions, in particular Pieter Abbeel, Alyosha Efros, Animesh Garg, Kevin Jamieson, Sanjay Krishnan, Sergey Levine, Zoe McCarthy, Stephen McKinley, Sachin Patil, Nan Tian, and Jur van den Berg.

References

[1] M. Aubry and B. Russell, "Understanding deep features with computer-generated imagery," *arXiv preprint arXiv:1506.01151*, 2015.

[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," *Robotics, IEEE Transactions on*, vol. 28, no. 4, pp. 899–910, 2012.

[3] T. D. Barfoot and P. T. Furgale, "Associating uncertainty with three-dimensional poses for use in estimation problems," *Robotics, IEEE Transactions on*, vol. 30, no. 3, pp. 679–693, 2014.

[4] C. Batty, "Sdfgen," https://github.com/christopherbatty/SDFGen.

[5] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesisa survey," *Robotics, IEEE Transactions on*, vol. 30, no. 2, pp. 289–309, 2014.

[6] A. M. Bronstein, M. M. Bronstein, L. J. Guibas, and M. Ovsjanikov, "Shape google: Geometric words and expressions for invariant shape retrieval," *ACM Transactions on Graphics (TOG)*, vol. 30, 2011.

[7] P. Brook, M. Ciocarlie, and K. Hsiao, "Collaborative grasp planning with multiple object representations," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2011, pp. 2851–2858.

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

[9] R. Detry, C. H. Ek, M. Madry, and D. Kragic, "Learning a dictionary of prototypical grasp-predicting parts from grasping experience," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 601–608.

[10] R. Detry, D. Kraft, O. Kroemer, L. Bodenhagen, J. Peters, N. Krüger, and J. Piater, "Learning grasp affordance densities," *Paladyn, Journal of Behavioral Robotics*, vol. 2, no. 1, pp. 1–17, 2011.

[11] R. Goetschalckx, P. Poupart, and J. Hoey, "Continuous correlated beta processes," in *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, vol. 22, no. 1. Citeseer, 2011, p. 1269.

[12] C. Goldfeder and P. K. Allen, "Data-driven grasping," *Autonomous Robots*, vol. 31, no. 1, pp. 1–20, 2011.

[13] C. Goldfeder, M. Ciocarlie, H. Dang, and P. K. Allen, "The columbia grasp database," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 1710–1716.

[14] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, *et al.*, "Deep-speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[15] 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*, vol. 36, no. 1-2, pp. 51–65, 2014.

[16] M. W. Hoffman, B. Shahriari, and N. de Freitas, "Exploiting correlation and budget constraints in bayesian multi-armed bandit optimization," *arXiv preprint arXiv:1303.6746*, 2013.

[17] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[18] D. Kappler, J. Bohg, and S. Schaal, "Leveraging big data for grasp planning," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2015.

[19] A. Kasper, Z. Xue, and R. 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*, vol. 31, no. 8, pp. 927–934, 2012.

[20] B. Kehoe, D. Berenson, and K. Goldberg, "Toward cloud-based grasping with uncertainty in shape: Estimating lower bounds on achieving force closure with zero-slip push grasps," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2012, pp. 576–583.

[21] B. Kehoe, A. Matsukawa, S. Candido, J. Kuffner, and K. Goldberg, "Cloud-based robot grasping with the google object recognition engine," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 4263–4270.

[22] B. Kehoe, S. Patil, P. Abbeel, and K. Goldberg, "A survey of research on cloud robotics and automation," *Automation Science and Engineering, IEEE Transactions on*, vol. 12, no. 2, pp. 398–409, 2015.

[23] J. Kim, K. Iwamoto, J. J. Kuffner, Y. Ota, and N. S. Pollard, "Physically based grasp quality evaluation under pose uncertainty," *IEEE Trans. Robotics*, vol. 29, no. 6, pp. 1424–1439, 2013.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[25] O. Kroemer, R. Detry, J. Piater, and J. Peters, "Combining active learning and reactive control for robot grasping," *Robotics and Autonomous Systems*, vol. 58, no. 9, pp. 1105–1116, 2010.

[26] M. Laskey, J. Mahler, Z. McCarthy, F. Pokorny, S. Patil, J. van den Berg, D. Kragic, P. Abbeel, and K. Goldberg, "Multi-arm bandit models for 2d sample based grasp planning with uncertainty." in *Proc. IEEE Conf. on Automation Science and Engineering (CASE)*. IEEE, 2015.

[27] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[28] B. Li, Y. Lu, C. Li, A. Godil, T. Schreck, M. Aono, M. Burtscher, Q. Chen, N. K. Chowdhury, B. Fang, *et al.*, "A comparison of 3d shape retrieval methods based on a large-scale benchmark supporting multimodal queries," *Computer Vision and Image Understanding*, vol. 131, pp. 1–27, 2015.

[29] J. Mahler, S. Patil, B. Kehoe, J. van den Berg, M. Ciocarlie, P. Abbeel, and K. Goldberg, "Gp-gpis-opt: Grasp planning under shape uncertainty using gaussian process implicit surfaces and sequential convex programming," 2015.

[30] D. Maturana and S. Scherer, "Voxnet: A 3d convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2015.

[31] L. Montesano and M. Lopes, "Active learning of visual descriptors for grasping using non-parametric smoothed beta distributions," *Robotics and Autonomous Systems*, vol. 60, no. 3, pp. 452–462, 2012.

[32] J. Oberlin and S. Tellex, "Autonomously acquiring instance-based object models from experience," in *Int. S. Robotics Research (ISRR)*, 2015.

[33] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2016.

[34] F. T. Pokorny, K. Hang, and D. Kragic, "Grasp moduli spaces." in *Robotics: Science and Systems*, 2013.

[35] F. T. Pokorny and D. Kragic, "Classical grasp quality evaluation: New theory and algorithms," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[36] M. Salganicoff, L. H. Ungar, and R. Bajcsy, "Active learning for vision-based robot grasping," *Machine Learning*, vol. 23, no. 2-3, pp. 251–278, 1996.

[37] S. Salti, F. Tombari, and L. Di Stefano, "Shot: Unique signatures of histograms for surface and texture description," *Computer Vision and Image Understanding*, vol. 125, pp. 251–264, 2014.

[38] A. Singh, J. Sha, K. S. Narayan, T. Achim, and P. Abbeel, "Bigbird: A large-scale 3d database of object instances," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2014.

[39] G. Smith, E. Lee, K. Goldberg, K. Bohringer, and J. Craig, "Computing parallel-jaw grips," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 1999.

[40] N. Srinivas, A. Krause, S. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," in *Proc. International Conference on Machine Learning (ICML)*, 2010.

[41] T. Stouraitis, U. Hillenbrand, and M. A. Roa, "Functional power grasps transferred through warping and replanning," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4933–4940.

[42] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3d shape recognition," *arXiv preprint arXiv:1505.00880*, 2015.

[43] J. Weisz and P. K. Allen, "Pose error robust grasping from contact wrench space metrics," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 557–562.

[44] W. Wohlkinger, A. Aldoma, R. B. Rusu, and M. Vincze, "3dnet: Large-scale object class recognition from cad models," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2012.

[45] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: A deep representation for volumetric shape modeling," in *CVPR*, vol. 1, no. 2, 2015, p. 3.

[46] L. E. Zhang, M. Ciocarlie, and K. Hsiao, "Grasp evaluation with graspable feature matching," in *RSS Workshop on Mobile Manipulation: Learning to Manipulate*, 2011.

[47] Y. Zheng and W.-H. Qian, "Coping with the grasping uncertainties in force-closure analysis," *Int. J. Robotics Research (IJRR)*, vol. 24, no. 4, pp. 311–327, 2005.