ARTIFICIAL INTELLIGENCE

Learning ambidextrous robot grasping policies

Jeffrey Mahler[1,2]*, Matthew Matl[1], Vishal Satish[1], Michael Danielczuk[1], Bill DeRose[2], Stephen McKinley[2], Ken Goldberg[1,2]

Universal picking (UP), or reliable robot grasping of a diverse range of novel objects from heaps, is a grand challenge for e-commerce order fulfillment, manufacturing, inspection, and home service robots. Optimizing the rate, reliability, and range of UP is difficult due to inherent uncertainty in sensing, control, and contact physics. This paper explores "ambidextrous" robot grasping, where two or more heterogeneous grippers are used. We present Dexterity Network (Dex-Net) 4.0, a substantial extension to previous versions of Dex-Net that learns policies for a given set of grippers by training on synthetic datasets using domain randomization with analytic models of physics and geometry. We train policies for a parallel-jaw and a vacuum-based suction cup gripper on 5 million synthetic depth images, grasps, and rewards generated from heaps of three-dimensional objects. On a physical robot with two grippers, the Dex-Net 4.0 policy consistently clears bins of up to 25 novel objects with reliability greater than 95% at a rate of more than 300 mean picks per hour.

INTRODUCTION

Universal picking (UP), or the ability of robots to rapidly and reliably grasp a wide range of novel objects, can benefit applications in warehousing, manufacturing, medicine, retail, and service robots. UP is highly challenging because of inherent limitations in robot perception and control. Sensor noise and occlusions obscure the exact geometry and position of objects in the environment. Parameters governing physics such as center of mass and friction cannot be observed directly. Imprecise actuation and calibration lead to inaccuracies in arm positioning. Thus, a policy for UP cannot assume precise knowledge of the state of the robot or objects in the environment.

One approach to UP is to create a database of grasps on three-dimensional (3D) object models using grasp performance metrics derived from geometry and physics (*1*, *2*) with stochastic sampling to model uncertainty (*3*, *4*). This analytic method requires a perception system to register sensor data to known objects and does not generalize well to a large variety of novel objects in practice (*5*, *6*). A second approach uses machine learning to train function approximators such as deep neural networks to predict the probability of success of candidate grasps from images using large training datasets of empirical successes and failures. Training datasets are collected from humans (*7*–*9*) or physical experiments (*10*–*12*). Collecting such data may be tedious and prone to inaccuracies due to changes in calibration or hardware (*13*).

To reduce the cost of data collection, we explored a hybrid approach that uses models from geometry and mechanics to generate synthetic training datasets. However, policies trained on synthetic data may have reduced performance on a physical robot due to inherent differences between models and real-world systems. This simulation-to-reality transfer problem is a long-standing challenge in robot learning (*14*–*17*). To bridge the gap, the hybrid method uses domain randomization (*17*–*22*) over objects, sensors, and physical parameters. This encourages policies to learn grasps that are robust to imprecision in sensing, control, and physics. Furthermore, the method plans grasps based on depth images, which can be simulated accurately using ray tracing (*18*, *19*, *23*) and are invariant to object color (*24*).

The hybrid approach has been used to learn reliable UP policies on a physical robot with a single gripper (*25*–*28*). However, different grasp modalities are needed to reliably handle a wide range of objects in practice. For example, vacuum-based suction-cup grippers can easily grasp objects with nonporous, planar surfaces such as boxes, but they may not be able to grasp small objects, such as paper clips, or porous objects, such as cloth.

In applications such as the Amazon Robotics Challenge, it is common to expand range by equipping robots with more than one end effector (e.g., both a parallel-jaw gripper and a suction cup). Domain experts typically hand-code a policy to decide which gripper to use at runtime (*29*–*32*). These hand-coded strategies are difficult to tune and may be difficult to extend to new cameras, grippers, and robots.

Here, we introduce "ambidextrous" robot policy learning using the hybrid approach to UP. We propose the Dexterity Network (Dex-Net) 4.0 dataset generation model, extending the gripper-specific models of Dex-Net 2.0 (*19*) and Dex-Net 3.0 (*19*). The framework evaluates all grasps with a common metric: expected wrench resistance, or the ability to resist task-specific forces and torques, such as gravity, under random perturbations.

We implement the model for a parallel-jaw gripper and a vacuum-based suction cup gripper and generate the Dex-Net 4.0 training dataset containing more than 5 million grasps associated with synthetic point clouds and grasp metrics computed from 1664 unique 3D objects in simulated heaps. We train separate Grasp Quality Convolutional Neural Networks (GQ-CNNs) for each gripper and combine them to plan grasps for objects in a given point cloud.

The contributions of this paper are as follows:

1) A partially observable Markov decision process (POMDP) framework for ambidextrous robot grasping based on robust wrench resistance as a common reward function.

2) An ambidextrous grasping policy trained on the Dex-Net 4.0 dataset that plans a grasp to maximize quality using a separate GQ-CNN for each gripper.

3) Experiments evaluating performance on bin picking with heaps of up to 50 diverse, novel objects and an ABB YuMi robot with a parallel-jaw and suction-cup gripper in comparison with hand-coded and learned baselines.

Experiments suggest that the Dex-Net 4.0 policy achieves 95% reliability on a physical robot with 300 mean picks per hour (MPPH) (successful grasps per hour).

[1]Department of Electrical Engineering and Computer Sciences, UC Berkeley, Berkeley, CA 94720, USA. [2]Department of Industrial Engineering and Operations Research, UC Berkeley, Berkeley, CA 94720, USA.
*Corresponding author. Email: jmahler@berkeley.edu

RESULTS

Ambidextrous robot grasping

We consider the problem of ambidextrous grasping of a wide range of novel objects from cluttered heaps using a robot with a depth camera and two or more available grippers, such as a vacuum-based suction-cup gripper and/or a parallel-jaw gripper. To provide context for the metrics and approaches considered in experiments, we formalize this problem as a POMDP (*33*) in which a robot plans grasps to maximize expected reward (probability of grasp success) given imperfect observations of the environment.

A robot with an overhead depth camera views a heap of novel objects in a bin. On grasp attempt t, a robot observes a point cloud $\mathbf{y}_t$ from the depth camera. The robot uses a policy $\mathbf{u}_t = \pi(\mathbf{y}_t)$ to plan a grasp action $\mathbf{u}_t$ for a gripper g consisting of a 3D rigid position and orientation of the gripper $\mathbf{T}_g = (\mathbf{R}_g, \mathbf{t}_g) \in \mathrm{SE}(3)$. Upon executing $\mathbf{u}_t$, the robot receives a reward $R_t = 1$ if it successfully lifts and transports exactly one object from the bin to a receptacle and $R_t = 0$ otherwise. The observations and rewards depend on a latent state $\mathbf{x}_t$ that is unknown to the robot and describes geometry, pose, center of mass, and material properties of each object. After either the bin is empty or T total grasp attempts, the process terminates.

These variables evolve according to an environment distribution that reflects sensor noise, control imprecision, and variation in the initial bin state:

1) Initial state distribution. Let $p(\mathbf{x}_0)$ be a distribution over possible states of the environment that the robot is expected to handle due to variation in objects and tolerances in camera positioning.

2) Observation distribution. Let $p(\mathbf{y}_t|\mathbf{x}_t)$ be a distribution over observations given a state due to sensor noise and tolerances in the camera optical parameters.

3) Transition distribution. Let $p(\mathbf{x}_{t+1}|\mathbf{x}_t,\mathbf{u}_t)$ be a distribution over next states given the current state and grasp action due to imprecision in control and physics.

The goal is to learn a policy π to maximize the rate of reward, or MPPH ρ, up to a maximum of T grasp attempts:

$$\rho(\pi) = \mathbb{E}\left[\left(\sum_{t=0}^{T-1} R_t\right) \Big/ \left(\sum_{t=0}^{T-1} \Delta_t\right)\right]$$

where T is the number of grasp attempts and Δ_t is the duration of executing grasp action $\mathbf{u}_t$ in hours. The expectation is taken with respect to the environment distribution:

$$p(\mathbf{x}_0, \mathbf{y}_0, \ldots, \mathbf{x}_T, \mathbf{y}_T|\pi) = p(\mathbf{x}_0) \prod_{t=0}^{T-1} p(\mathbf{y}_t|\mathbf{x}_t) p(\mathbf{x}_{t+1}|\mathbf{x}_t, \pi(\mathbf{y}_t))$$

It is common to measure performance in terms of the mean rate ν and reliability Φ (also known as the success rate) of a grasping policy for a given range of objects:

$$\nu = 1/\mathbb{E}[\Delta_t] \quad \Phi(\pi) = \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1} R_t\right]$$

If the time per grasp is constant, the MPPH is the product of rate and reliability: $\rho = \nu\Phi$.

Learning from synthetic data

We propose a hybrid approach to ambidextrous grasping that learns policies on synthetic training datasets generated using analytic models and domain randomization over a diverse range of objects, cameras, and parameters of physics for robust transfer from simulation to reality (*17*, *20*, *22*). The method optimizes for a policy to maximize MPPH under the assumption of a constant time per grasp: $\pi^* = \mathrm{argmax}_\pi \Phi(\pi)$.

To learn a policy, the method uses a training dataset generation distribution based on models from physics and geometry, μ, to computationally synthesize a massive training dataset of point clouds, grasps, and reward labels for heterogeneous grippers. The distribution μ consists of two stochastic components: (i) a synthetic training environment $\xi(\mathbf{y}_0, R_0, \ldots, \mathbf{y}_T, R_T|\pi)$ that can sample paired observations and rewards given a policy and (ii) a data collection policy $\tau(\mathbf{u}_t|\mathbf{x}_t, \mathbf{y}_t)$ that can sample a diverse set of grasps using full-state knowledge. The synthetic training environment simulates grasp outcomes by evaluating rewards according to the ability of a grasp to resist forces and torques due to gravity and random peturbations. The environment also stochastically samples heaps of 3D objects in a bin and renders depth images of the scene using domain randomization over the camera position, focal length, and optical center pixel. The dataset collection policy evaluates actions in the synthetic training environment using algorithmic supervision to guide learning toward successful grasps.

We explore large-scale supervised learning on samples from μ to train the ambidextrous policy π_θ across a set of two or more available grippers $\mathcal{G}$, as illustrated in Fig. 1. First, we sample a massive training dataset $\mathcal{D} = \{(R_i \mathbf{y}_i \mathbf{u}_i)\}_{i=1}^N$ from a software implementation of μ. Then, we learn a GC-CNN $Q_{\theta,g}(\mathbf{y}, \mathbf{u}) \in [0, 1]$ to estimate the probability of success for a grasp with gripper g given a depth image. Specifically, we optimize the weights θ_g to minimize the cross entropy loss $\mathcal{L}$ between the GQ-CNN prediction and the true reward over the dataset $\mathcal{D}$:

$$\theta_g^* = \underset{\theta_g \in \Theta}{\mathrm{argmin}} \sum_{(R_i, \mathbf{u}_i, \mathbf{y}_i) \in \mathcal{D}_g} \mathcal{L}(R_i, Q_\theta(\mathbf{y}_i, \mathbf{u}_i))$$

where $\mathcal{D}_g$ denotes the subset of the training dataset $\mathcal{D}$ containing only grasps for gripper g. We construct a robot policy π_θ from the GQ-CNNs by planning the grasp that maximizes quality across all available grippers:

$$\pi_\theta(\mathbf{y}_t) = \underset{g \in \mathcal{G}}{\mathrm{argmax}} \left\{ \max_{\mathbf{u}_g \in \mathcal{U}_g} Q_{\theta,g}(\mathbf{y}_t, \mathbf{u}_g) \right\}$$

where $\mathcal{U}_g$ is a set of candidate grasps for gripper g sampled from the depth image.

To evaluate the method, we learn the Dex-Net 4.0 ambidextrous policy on the Dex-Net 4.0 training dataset, which contains 5 million synthetic point clouds, grasps, and reward labels. Dex-Net 4.0 is generated from 5000 unique 3D object heaps with about 2.5 million data points each for a vacuum-based suction-cup gripper and a parallel-jaw gripper. Figure S1 analyzes the features learned by the Dex-Net 4.0 GQ-CNNs.

Physical experiments

We executed more than 2500 grasp attempts on a physical robot system with a parallel-jaw and suction-cup gripper to characterize the reliability of the Dex-Net 4.0 policy on a bin-picking task with 50 novel test objects. The experiments aimed to evaluate (i) the reliability and range of the Dex-Net 4.0 policy compared with a set of baselines; (ii) the effect of training dataset diversity, neural network architecture, and learning from real data; and (iii) the failure modes of the Dex-Net 4.0 policy.

To analyze performance, we selected a dataset of 50 objects with diverse shapes, sizes, and material properties. The dataset, described in the

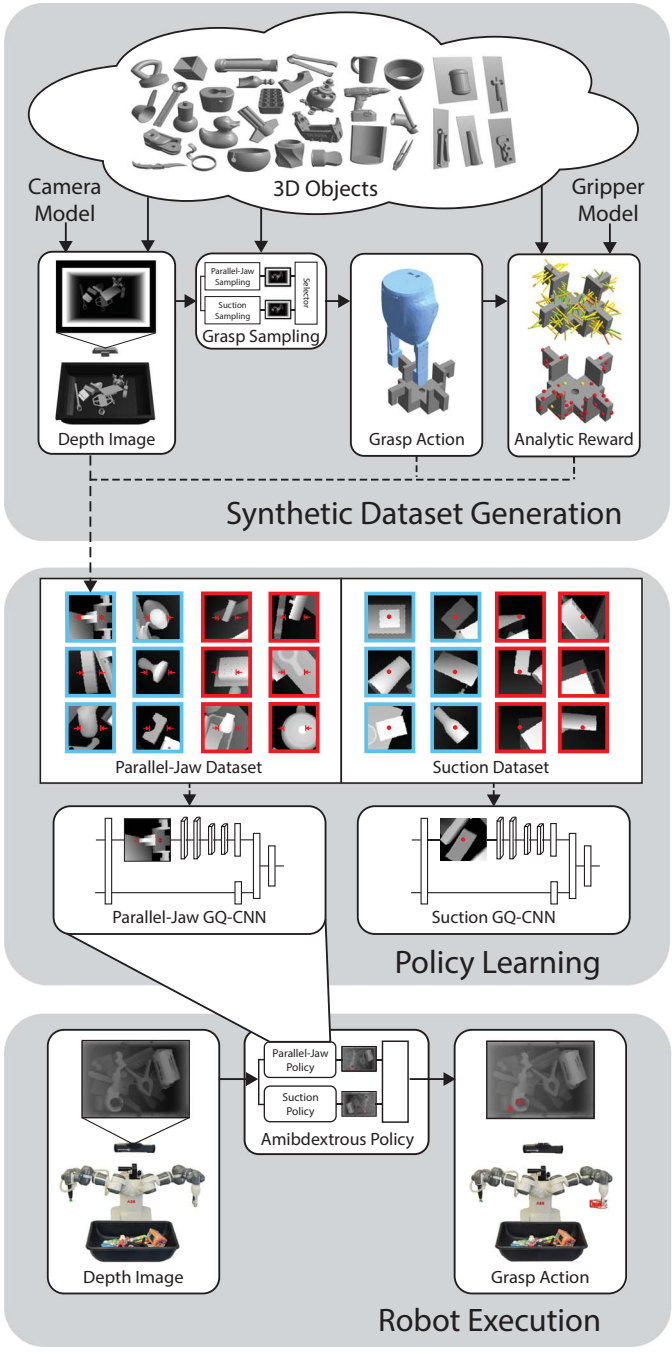

Fig. 1. Learning ambidextrous grasping policies for UP. (**Top**) Synthetic training datasets of depth images, grasps, and rewards are generated from a set of 3D computer-aided design (CAD) models using analytic models based on physics and domain randomization. Specifically, a data collection policy proposes actions given a simulated heap of objects, and the synthetic training environment evaluates rewards. Reward is computed consistently across grippers by considering the ability of a grasp to resist a given wrench (force and torque) based on the grasp wrench space, or the set of wrenches that the grasp can resist through contact. (**Middle**) For each gripper, a policy is trained by optimizing a deep GQ-CNN to predict the probability of grasp success given a point cloud over a large training dataset containing millions of synthetic examples from the training environment. Data points are labeled as successes (blue) or failures (red) according to the analytic reward metric. (**Bottom**) The ambidextrous policy is deployed on the real robot to select a gripper by maximizing grasp quality using a separate GQ-CNN for each gripper.

Supplementary Materials, includes retail products, groceries, tools, office supplies, toys, and 3D-printed industrial parts. We separated objects into two difficulty levels with 25 objects each, illustrated in Fig. 2:

1) Level 1. Prismatic and circular solids (e.g., rectangular prisms, spheres, and cylinders).

2) Level 2. Common household objects including examples with flat cardboard backing and clear plastic covers ("blisterpack"), varied geometry, and masses up to 500 g (the payload of the ABB YuMi).

For each trial, we placed all objects in the bin and allowed the robot to iteratively attempt grasps until either no objects remained or a maximum number of attempts were reached. Each grasp was planned on the basis of a depth image from an overhead 3D camera. For details on the experimental setup and procedure, see Materials and Methods. A video of each grasp attempt is available in the Supplementary Materials.

Comparison with baseline policies

We evaluated the Dex-Net 4.0 ambidextrous policy against three baselines in five independent trials. To compare with hand-coded methods used in practice, we implemented a best-effort suction-only policy and an ambidextrous policy based on geometric heuristics similar to those used in the Amazon Robotics Challenge (*29*, *30*, *32*). To study the importance of the consistent reward model, we also evaluated a policy that

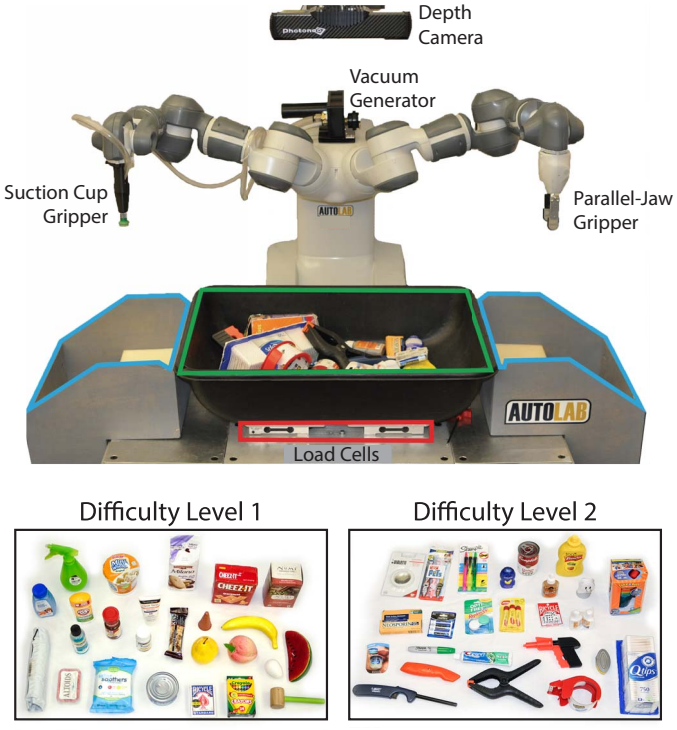

Fig. 2. Physical benchmark for evaluating UP policies. (**Top**) The robot plans a grasp to iteratively transport each object from the picking bin (green) to a receptacle (blue) using either a suction-cup or a parallel-jaw gripper. Grasp planning is based on 3D point clouds from an overhead Photoneo PhoXi S industrial depth camera. (**Bottom**) Performance is evaluated on two datasets of novel test objects not used in training. (**Left-Bottom**) Level 1 objects consist of prismatic and circular solids (e.g., boxes and cylinders) spanning groceries, toys, and medicine. (**Right-Bottom**) Level 2 objects are more challenging, including common objects with clear plastic and varied geometry, such as products with cardboard blisterpack packaging.

ranks grasps using Dex-Net 2.0 and 3.0 fine-tuned on simulated heaps with separate reward metrics for each gripper (see Materials and Methods for details).

Figure 3 shows the performance on the two object datasets. Dex-Net 4.0 achieves the highest success rate across all object datasets with a reliability of 97 and 95% on the level 1 and level 2 objects, respectively. The policy uses the suction cup gripper on 82% of grasps. The best baseline method has a reliability of 93 and 80%, respectively. Analysis of the number of objects picked versus the number of attempts suggests that the baseline methods take longer to clear the last few objects from the bin, sometimes failing to clear several of the objects.

We detail additional metrics for each policy in Table 1, including the reliability and MPPH of the learned quality functions. The Dex-Net 4.0 policy has the highest reliability on both level 1 and level 2 objects. The policy has a slightly lower MPPH than the suction heuristic on the level 1 objects because the heuristic can be evaluated more rapidly than the GQ-CNN.

We analyze the per-object reliability of each policy in fig. S2. The results suggest that differences in reliability between the policies on the level 1 objects may be due to specific configurations of objects (e.g., a thin object leaning against a wall of the bin) rather than the objects themselves. Figure S3 details the difficulty of each object according to the reliability across all policies. The most difficult objects were a box of Q-tips, a bottle of mustard, and the "bialetti," an espresso filter in a thin blisterpack package.

To further quantify the range of the Dex-Net 4.0 ambidextrous policy, we measured the performance of grasping each of the 50 objects from level 1 and level 2 in isolation in the bin for five attempts each. Dex-Net 4.0 achieved 98% reliability versus 52 and 94% reliability for the Dex-Net 2.0 and 3.0 policies, respectively.

Performance with large heaps

To investigate whether heap size affects performance, we benchmarked the policy on a dataset of 50 test objects combining all objects from the level 1 and level 2 datasets. Figure 3 displays the results for five independent trials with each policy. Dex-Net 4.0 has the highest reliability at 90%. In comparison, the performance of the heuristics is relatively unchanged, with success rates near 80%. Some failures of Dex-Net 4.0 are due to attempts to lift objects from underneath others.

Effects of training dataset diversity

We quantified the importance of dataset diversity by training the GQ-CNNs on three alternative synthetic training datasets:

1) Fewer unique objects: 100 unique 3D objects in 2500 unique heaps.

2) Very few unique heaps: 1664 unique 3D objects in 100 unique heaps.

3) Fewer unique heaps: 1664 unique 3D objects in 500 unique heaps.

Figure 3 displays the performance on level 1 and level 2 objects. The policies have reduced reliability and appeared to be particularly sensitive to the number of unique heaps used in training.

Varying the neural network architecture

We studied whether changes to the neural network architecture affect the performance of the resulting policy by training on the Dex-Net 4.0 dataset using the "Improved GQ-CNN" architecture (*34*). As seen in Fig. 3, the architecture has comparable performance with the standard GQ-CNN architecture.

Training on physical grasp outcomes

We also explored whether performance can be improved by training on labeled grasp attempts on a physical system. We used a dataset of more than 13,000 labeled grasp attempts collected over 6 months of experiments and demonstrations of the system. About 5000 data points were labeled by human operators, and the remaining 8000 were labeled automatically using weight differences measured by load cells.

We trained 10 variants of the Dex-Net 4.0 policy on varying ratios of synthetic and real data using fine-tuning on the fully connected (FC) layers, including a model trained on only empirical data (see Materials

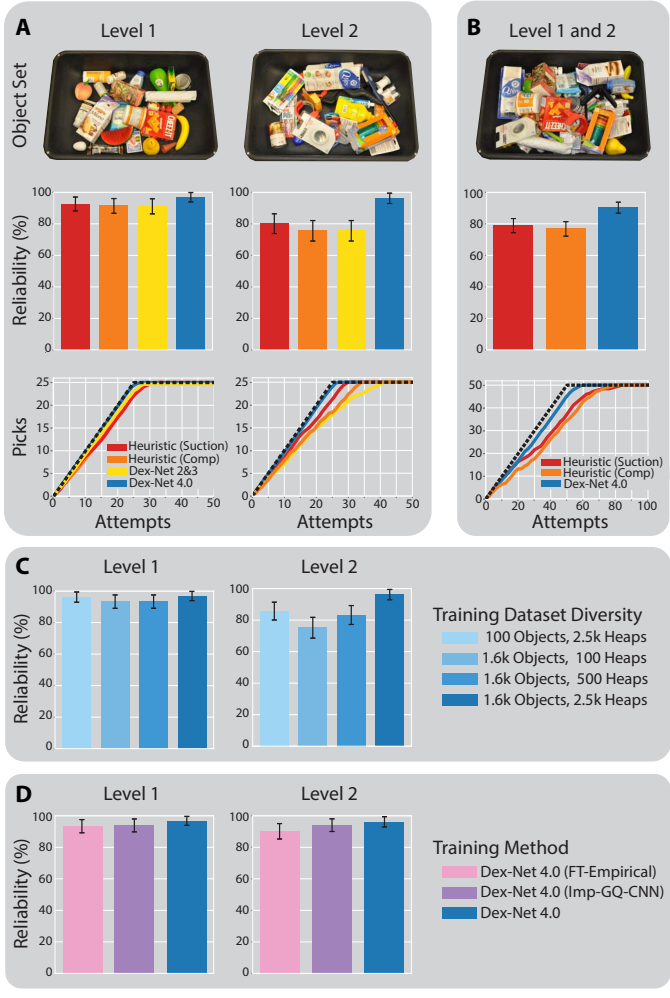

Fig. 3. Performance of the Dex-Net 4.0 ambidextrous policy on the bin picking benchmark. Error bars show the 95% confidence interval on reliability using the standard error of the mean (SEM). (**A**) Comparison with three baseline methods on level 1 and level 2 objects on heaps of 25 novel objects: (i) a hand-coded heuristic for the suction cup [Heuristic (suction)], (ii) a hand-coded heuristic for selecting between suction-cup and parallel-jaw grippers [Heuristic (comp)], and (iii) an ambidextrous policy fine-tuned on simulated heaps from the Dex-Net 2.0 and 3.0 base GQ-CNNs and reward metrics. For reference, the best possible performance (succeeding on every pick until the bin is cleared) is illustrated with a dashed-dotted black line. (**B**) Performance with large heaps of 50 novel objects. (**C**) Ablation study measuring the effect of training on less diverse datasets with either fewer unique heaps or fewer unique 3D object models. (**D**) Performance of two training alternatives: the improved GQ-CNN (Imp-GQ-CNN) architecture (*34*) and fine-tuning (FT) on 13,000 real data points.

Table 1. Detailed performance analysis of the Dex-Net 4.0 and baseline policies on the bin-picking benchmark for five trials on level 1 and level 2 datasets of 25 novel objects each. We report the reliability, MPPH, average precision (AP), total number of grasps attempts (minimum of 125), and total number of failures.

	Level 1					Level 2				
Policy	**Reliability (%)**	**MPPH**	**AP (%)**	**No. of attempts**	**No. of failures**	**Reliability (%)**	**MPPH**	**AP (%)**	**No. of attempts**	**No. of failures**
Heuristic (suction)	93	**331**	95	135	10	80	304	87	156	31
Heuristic (composite)	91	281	93	139	14	76	238	83	168	43
Dex-Net 2 and 3 composite	91	306	93	135	10	76	255	64	168	43
Dex-Net 4.0	**97**	309	**100**	**129**	**4**	**95**	**312**	**99**	**131**	**6**

and Methods for details). The best-performing empirically trained policy had comparable reliability with the original Dex-Net 4.0 policy on the physical benchmark, as shown in Fig. 3, and did not lead to substantial performance increases.

Adversarial objects

To probe the boundaries of the range of the Dex-Net 4.0 policy, we evaluated its performance on a third object dataset that contained 25 novel objects with few accessible and perceptible grasps due to adversarial geometry, transparency, specularity, and deformability. The results are illustrated in Fig. 4. Dex-Net 4.0 was still the highest performing policy, but the reliability was reduced to 63%.

Failures of the Dex-Net 4.0 policy often occur several times in sequence. To characterize these sequential failures, we explored a first-order memory-based policy to encourage exploration upon repeated failures, a technique that has been used to improve performance in the Amazon Robotics Challenge (*32*). The policy uses an instance recognition system to match object segments to previous failures in a database (*12*) and pushes objects to create accessible grasps when none are available (see Materials and Methods for details on the memory-based policy). The addition of memory increased the reliability to 80% at 242 MPPH.

DISCUSSION

Experiments suggest that ambidextrous policies trained on Dex-Net 4.0 achieve high reliability on novel objects on a physical robot, with more than 95% reliability on heaps of 25 novel objects at more than 300 MPPH. Dex-Net 4.0 outperforms hand-coded baselines similar to those used in applications such as the Amazon Robotics Challenge and also outperforms an ambidextrous policy based on previous versions of Dex-Net that use separate reward functions for each gripper. This suggests that learning with consistent reward functions across grippers can lead to increased reliability on a physical robot.

Experiments also suggest that performance is sensitive to several factors. Heaps containing more objects lead to decreased reliability because the policy attempts to lift objects that are occluded by others in the heap. Performance also depends on the diversity of the training dataset, with more diverse datasets leading to higher performance on a physical robot. Last, performance varies based on the test objects, with more complex geometries and material properties leading to reduced reliability. Use of a memory system can help compensate for repeated failures, increasing reliability on adversarial objects from 63 to 80%.

Benefits of ambidextrous grasping

The experimental results highlight the advantage of using a set of two or more heterogeneous grippers. Although a policy with only a single suction cup can achieve high reliability on the level 1 prismatic and circular objects, performance drops to 80% on the level 2 objects with more complex geometry. In comparison, the ambidextrous grasping policy uses the parallel jaws on 20% of grasp attempts to achieve 95% reliability on the level 2 objects. Furthermore, a consistent reward appears to be important for learning an ambidextrous policy to reliably decide between grippers. However, this study only considers a single combination of grippers. Future research could study applications to new grippers, such as two-finger underactuated hands or multi–suction-cup arrays. Future work could also consider extensions of ambidextrous grasping, such as simultaneous grasping with multiple arms or planning grasps for three or more grippers.

Physics-based reward design

The results of this paper also suggest that analytic quasi-static grasp quality metrics (*35*, *36*) with domain randomization can be used as a computationally efficient reward function for learning ambidextrous grasping policies that are robust to sensor noise and imprecision. This stands in contrast to past research (*5*, *6*) that has criticized quasi-static metrics for making strong assumptions and considering only a necessary, not sufficient, condition for dynamic grasp stability. Experiments suggest that the Dex-Net 4.0 policy generalizes to objects with deformable surfaces, moving parts, and transparencies that do not satisfy the assumptions of the analytic metrics. This may be because grasps with high analytic quality over a diverse range of 3D objects tend to correlate with grasp affordances: geometric features of objects that facilitate grasping, such as handles or flat suctionable surfaces. Further studies may be necessary to understand why grasps are often dynamically stable in practice. One hypothesis is that material compliance in the fingertips acts as a passive stability controller. Future research could investigate whether this result generalizes to additional grippers such as multifingered (*20*) or soft hands (*37*).

Bias-variance tradeoff in dataset collection

Experiments suggest that a policy fine-tuned on 13,000 examples collected from physical experiments does not substantially improve the Dex-Net 4.0 ambidextrous grasping policy trained on only synthetic data. This may appear counterintuitive, because the model used to generate synthetic training data cannot possibly model the exact behavior of the real-world system and therefore may induce bias (*38*). This may

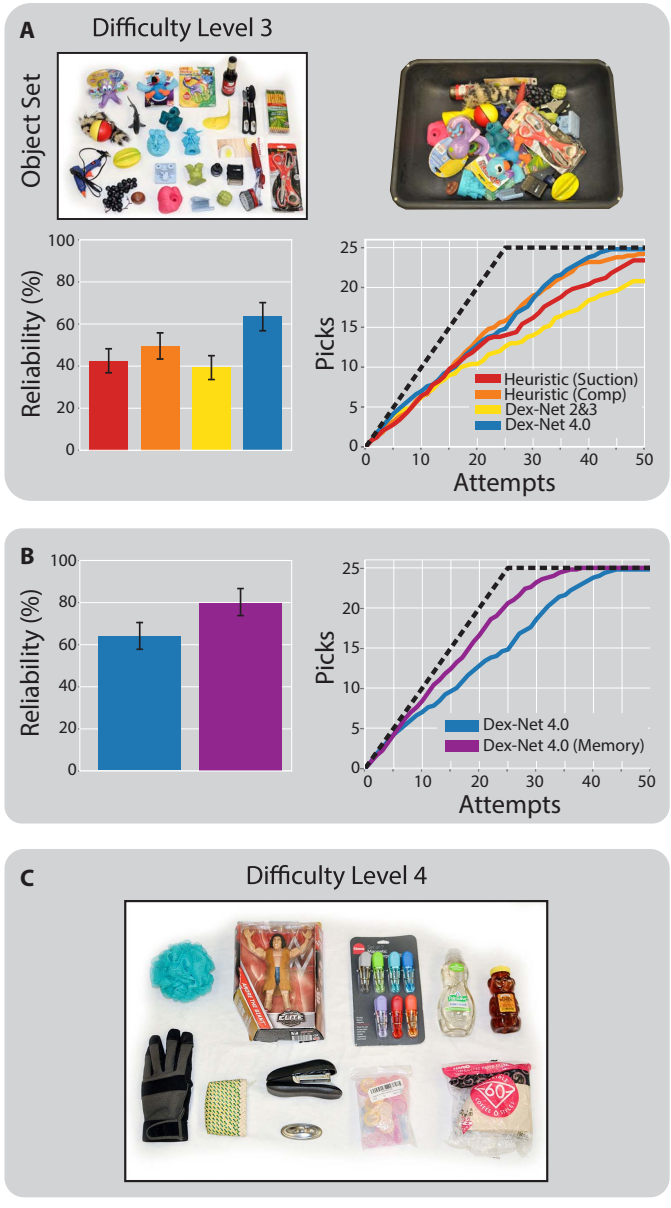

Fig. 4. Failure modes of the Dex-Net 4.0 policy. Error bars show the 95% confidence interval on reliability using the SEM. (**A**) Performance on level 3: a dataset of 25 novel objects with adversarial geometry and material properties. (**B**) Evaluation of a first-order memory-based policy for UP that masks regions of the point cloud with an instance recognition system to avoid repeated failures. (**C**) Pathological objects that cannot be grasped with Dex-Net 4.0 due to reflectance properties such as transparency, which affect depth sensing, and material properties such as porosity and deformability (e.g., loose packaging), which affect the ability to form a vacuum seal on the object surface.

relate to the bias-variance tradeoff in machine learning (*39*). Although the tradeoff is typically analyzed in terms of the function class, the results of this paper suggest that the training data distribution is also relevant. Using a biased model for rapid data collection may improve the scale and consistency of training datasets, leading to better performance on a physical system in comparison with methods based on smaller training datasets with high rates of mislabeled examples. Future research could consider novel methods for learning with a combination of synthetic and real data, such as using analytic models to guide empirical data collection.

Sequential learning for UP

Finding a policy to maximize MPPH is inherently a sequential learning problem, in which grasp actions affect future states of the heap. Theory on imitation learning (*40*) and reinforcement learning (*41*) suggests that policies should take actions that lead to states with high expected future reward to guarantee high reliability. However, experiments in this paper suggest that the Dex-Net 4.0 policy performs well on the sequential task of bin picking, although it was trained with supervised learning to greedily maximize the probability of success for a single timestep. This suggests that performance is not particularly sensitive to the sequence of states of the object heap. This may be due to the random configuration of objects, which often have one or more exposed graspable surfaces in every state of the heap. Furthermore, performance may be increased on difficult objects by augmenting the policy with a memory system that avoids repeated mistakes.

Application to different sensors and grippers

The Dex-Net 4.0 method for training UP policies could be applied to other objects, cameras, and grippers by implementing a new dataset generation distribution and training a new GQ-CNN on samples from this distribution. For example, objects could be placed in structured configurations, such as packed in boxes or placed on shelves, and camera intrinsic parameters could be set to model a different sensor. However, the experiments in this paper are limited in scope. This study only evaluates performance on heaps of 50 unique, randomly arranged objects, which do not represent all possible object geometries. Furthermore, the hardware benchmark uses only one industrial high-resolution depth camera positioned directly overhead. The experiments only test a single parallel-jaw and vacuum-based suction cup gripper. Last, the constant time assumption that relates MPPH maximization to supervised learning may not be applicable to all robot picking systems. For example, there may be a time cost for switching grippers due to time spent physically mounting each tool. Future studies could evaluate performance in new applications with variations in objects, cameras, grippers, and robots.

Opportunities for future research

The most common failure modes of the policy are (i) attempting to grasp objects that are occluded due to overlap in the heap and (ii) repeated failures on objects with adversarial geometry and material properties. A subset of objects that cannot yet be reliably grasped with Dex-Net 4.0 is pictured in Fig. 4. One category includes objects imperceptible to a depth camera, such as those with transparent or specular surfaces. Another category is characterized by structured surface variations, such as parallel lines or buttons on a remote, which can trigger false positives in the suction network. Other classes include porous objects and objects with loose packaging.

Some failure modes could be addressed by increasing the diversity of objects in the training dataset or improving the dataset generation model. Rendering synthetic color images using domain randomization (*17*) could enable the system to grasp transparent, specular, or highly textured objects. Models of deformation and porosity could be used to reduce suction failures due to incorrect assumptions of the Dex-Net 4.0 model. The reward model could also be extended to compute the wrench set from all contacts between objects instead of only

considering the grippers and gravity, which could reduce failures due to object overlap.

Other extensions could substantially increase the reliability and range. The observed performance increase on the level 3 objects using a first-order memory system suggests that reinforcement learning could be used to reduce repeated failures. Furthermore, training on larger datasets of empirically collected data could reduce the simulation-to-reality gap. Another way to increase rate is to use feedback policies that actively regrasp dropped objects based on visual servoing (*10*, *28*, *42*), force sensing (*43*–*45*), or tactile sensing (*46*–*49*).

MATERIALS AND METHODS

Synthetic training environment

The Dex-Net 4.0 synthetic training environment is based on the following assumptions: (i) quasi-static physics (e.g., inertial terms are negligible) with Coulomb friction, (ii) objects are rigid and made of nonporous material, (iii) the robot has a single overhead depth sensor with known intrinsics, and (iv) the robot has two end effectors with known geometry—a vacuum-based gripper consisting of a single disc-shaped linear-elastic suction-cup and a parallel-jaw gripper (see the Supplementary Materials for detailed values of parameters). Dex-Net 4.0 uses the POMDP definition described in the following sections.

States

Let $\mathbf{x} = (\mathcal{O}_1, \ldots \mathcal{O}_m, \mathcal{C}, \mathbf{w}_1, \ldots \mathbf{w}_m)$ denote the state of the environment at time t, consisting of a single overhead depth camera, a set of objects, and a perturbation wrench on each object (e.g., gravity and disturbances). Each object state $\mathcal{O}_i$ specifies the geometry $\mathcal{M}_i$, pose $\mathbf{T}_{o,i}$, friction coefficient γ_i, and center of mass $\mathbf{z}_i$. The camera state $\mathcal{C}$ specifies the intrinsic parameters $\mathcal{I}$ and pose $\mathbf{T}_c$. Each wrench $\mathbf{w}_i$ is specified as a vector $\mathbf{w}_i \in \mathbb{R}^6$.

Grasp actions

Let $\mathbf{u}_s \in \mathcal{U}_s$ denote a suction grasp in 3D space defined by a suction gripper $\mathcal{G}_s$ and a rigid pose of the gripper $\mathbf{T}_s = (\mathbf{R}_s, \mathbf{t}_s) \in \mathrm{SE}(3)$, where the rotation $\mathbf{R}_s \in \mathrm{SO}(3)$ defines the orientation of the suction tip and the translation $\mathbf{t}_s \in \mathbb{R}^3$ specifies the target location for the center of the suction disc. Let $\mathbf{u}_p \in \mathcal{U}_p$ denote a parallel-jaw grasp in 3D space defined by a parallel-jaw gripper $\mathcal{G}_p$ and a rigid pose of the gripper $\mathbf{T}_p = (\mathbf{R}_p, \mathbf{t}_p) \in \mathrm{SE}(3)$, where the rotation $\mathbf{R}_p \in \mathrm{SO}(3)$ defines the grasp axis and approach direction and the translation $\mathbf{t}_p \in \mathbb{R}^3$ specifies the target center point of the jaws. The set of all possible grasps is $\mathcal{U} = \mathcal{U}_s \cup \mathcal{U}_p$.

Point clouds

Let $\mathbf{y} = \mathbb{R}_+^{H \times W}$ be a 2.5D point cloud represented as a depth image with height H and width W taken by a camera with known intrinsics (*50*).

State distribution

The initial state distribution $\xi(\mathbf{x}_0)$ is the product of distributions on (*26*):

1) Object count (m): Poisson distribution with mean λ truncated to [1, 10].

2) Object heap ($\mathcal{O}$): Uniform distribution over 3D object models and the pose from which each model is dropped into the heap. Objects are sampled without replacement.

3) Depth camera ($\mathcal{C}$): Uniform distribution over the camera pose and intrinsic parameters.

4) Coulomb friction (γ): Truncated Gaussian constrained to [0, 1].

The initial state is sampled by drawing an object count m, drawing a subset of m objects, dropping the objects one by one from a fixed height above the bin, and running dynamic simulation with pybullet (*51*) until all objects have about zero velocity. The 3D object models are sampled from a dataset of 1664 3D objects models selected to reflect a broad range of products that are commonly encountered in applications such as warehousing, manufacturing, or home decluttering. The dataset was augmented with synthetic blisterpack meshes to reflect cardboard-backed products encountered in retail applications. Augmentation was performed by placing each source mesh in a quasi-static stable resting pose (*52*) on an infinite planar work surface and attaching a thin, flat segment to the mesh at the triangle(s) touching the work surface.

Observation distribution

Depth image observations are rendered using the open source Python library meshrender using randomization in the camera focal length and optical center pixel. No noise was added to the rendered images, because experiments used a high-resolution Photoneo PhoXi S industrial depth camera.

Reward distribution

Binary rewards occur when a quasi-static equilibrium is feasible between the grasp and an external wrench perturbation (e.g., due to gravity or inertia). Let $\mathcal{O}_i \in \mathbf{x}_t$ be an object contacted by the gripper when executing action $\mathbf{u}_t$. Then, we measure grasp success with a binary-valued metric $S(\mathbf{x}_t, \mathbf{u}_t) \in \{0, 1\}$ that evaluates the following conditions.

1) The gripper geometry in the pose specified by $\mathbf{u}_t$ is collision free.

2) The gripper contacts exactly one object $\mathcal{O}_i$ when executing the grasp parameterized by $\mathbf{u}_t$.

3) The grasp can resist a random disturbing force and torque (wrench) $\mathbf{w}_t = \mathbf{w}_g + \varepsilon_w$ on the grasped object with more than 50% probability, where $\mathbf{w}_g$ is the fixed wrench due to gravity and ε_w is a random wrench sampled from a zero-mean Gaussian $\mathcal{N}(0, \sigma_w^2 \mathbf{I})$.

Given an object consisting of a geometry $\mathcal{M}$ in pose $\mathbf{T}_o$, the gripper g (geometry and physical parameters such as friction) and grasp pose $\mathbf{T}_g$ are used to determine the contacts $\mathbf{c}$, or set of points and normals between the fingers and object. This set of contacts is used to compute the set of wrenches Λ that the grasp can apply to the object under quasi-static physics and a point contact model. Specifically, the wrench space for grasp $\mathbf{u}$ using a contact model with m basis wrenches is $\Lambda(\mathbf{u}) = \{\mathbf{w} \in \mathbb{R}^6 | \mathbf{w} = G(\mathbf{u})\alpha \text{ for some } \alpha \in \mathcal{F}(\mathbf{u})\}$, as defined in (*27*). The grasp matrix $G(\mathbf{u}) \in \mathbb{R}^{6 \times m}$ is a set of basis wrenches in the object coordinate frame specifying the set of wrenches that the grasp can apply through contact via active (e.g., joint torques) and passive (e.g., inertia) means. The wrench constraint set $\mathcal{F}(\mathbf{u}) \subseteq \mathbb{R}^m$ limits contact wrench magnitudes based on the capabilities of the gripper (*1*). Last, the grasp wrench space is used to measure grasp reward based on wrench resistance, or the ability of the grasp to resist a perturbation wrench $\mathbf{w}$ (e.g., due to gravity) as defined in (*27*). The grasp reward R is 1 if the probability of wrench resistance is greater than a threshold over M samples from the stochastic model.

Data collection policy

The dataset collection policy $\tau(\mathbf{u}_t | \mathbf{x}_t, \mathbf{y}_t)$ samples a mixture of actions from the point cloud and from an algorithmic supervisor $\Omega(\mathbf{x})$ that guides data toward successful grasps. Grasp actions are sampled from the point cloud using the sampling techniques of (*19*) and (*27*) to model the set of actions that the policy will evaluate when tested on real point clouds. Because this distribution may contain a very small percentage of successful actions, we sample actions with high expected reward from an algorithmic supervisor that evaluates grasps using full-state

information (*26*). The supervisor precomputes grasps on a set of known 3D objects in a database [such as in Dex-Net 1.0 (*53*)] that are robust to different possible orientations of each object. Because the state of each object in the heap is not known ahead of time, the supervisor estimates the probability of success, or quality, for each grasp over a large range of possible object orientations using the Monte Carlo grasp computation methods of Dex-Net 2.0 (*19*) and Dex-Net 3.0 (*27*). Given a full state of the heap, the supervisor computes the set of collision-free grasps with quality above a threshold for each object and then samples a grasp uniformly at random from the candidate set. Formally, the supervisor-guided data collection policy is

$$\tau(\mathbf{u}_t|\mathbf{x}_t, \mathbf{y}_t) = \begin{cases} \Omega(\mathbf{x}_t) & \text{with prob. } \varepsilon \\ \text{Unif}(\mathcal{U}_g(\mathbf{y}_t)) & \text{otherwise} \end{cases}$$

where $\mathcal{U}_g(\mathbf{y})$ is the set of candidate actions sampled from the point cloud with equal numbers of suction and parallel-jaw grasps. We use $\varepsilon = 1\%$ to favor actions sampled from the policy's own action space.

Training details

The Dex-Net 4.0 training dataset contains a large set of labeled actions for each point cloud to improve the computational efficiency of generating a single data point. Specifically, data points were generated using a one–time step Monte Carlo evaluation of reward for a large set of grasp actions on each unique object state. This leads to faster dataset collection and can eliminate the need for fine-tuning, which is prone to a phenomenon known as "catastrophic forgetting" that can lead to unpredictable failures of the grasping policy (*54*). Every sampled state from $\xi(\mathbf{x})$ has five associated depth images in Dex-Net 4.0, representing 3D point clouds captured from randomized camera poses and intrinsic optical parameters. Each image sampled from $\xi(\mathbf{y}|\mathbf{x})$ has a set of labeled actions for each gripper with associated quality metrics. The intrinsic parameters for the virtual cameras were sampled around the nominal values of a Photoneo PhoXi S industrial depth sensor. Images were converted to 96 pixel–by–96 pixel training thumbnails translated to move the grasp center to the thumbnail center pixel and rotated to align the grasp approach direction or axis with the middle row of pixels for the suction and parallel-jaw grippers, respectively.

The GQ-CNN architectures are similar to those used in Dex-Net 2.0 (*19*) and Dex-Net 3.0 (*27*) with two primary changes. First, we removed local response normalization because experiments suggest that it does not affect training performance. Second, we modified the sizes and pooling of the following layers : conv1_1 (16 9 by 9 filters, 1× pooling), conv1_2 (16 5 by 5 filters, 2× pooling), conv2_1 (16 5 by 5 filters, 1× pooling), conv2_2 (16 5 by 5 filters, 2× pooling), fc3 (128 output neurons), pc1 (16 output neurons), and fc4 (128 output neurons).

We trained each GQ-CNN using stochastic gradient descent with momentum for 50 epochs using an 80-20 training-to-validation image-wise split of the Dex-Net 4.0 dataset. We used a learning rate of 0.01 with an exponential decay of 0.95 every 0.5 epochs, a momentum term of 0.9, and an $\ell 2$ weight regularization of 0.0005. We initialized the weights of the model by sampling from a zero-mean Gaussian with variance $\frac{2}{n_i}$, where n_i is the number of inputs to the ith network layer (*55*). To augment the dataset during training, we reflected each image about its vertical and horizontal axes and rotated each image by 180° because these lead to equivalent grasps. Training took about 24 hours on a single NVIDIA TITAN Xp graphics processing unit (GPU). The learned GQ-CNNs achieved 96 and 98% classification accuracy for the suction cup and parallel-jaw grippers, respectively, on the held-out validation set.

Implementation of policies

We used the trained GQ-CNNs to plan grasps from point clouds on a physical robot with derivative-free optimization to search for the highest-quality grasp across both grippers. The policy optimizes for the highest-quality grasp for each gripper separately, using the cross-entropy method (CEM) (*10*, *19*, *27*, *56*), and then selects the grasps with the highest estimated quality across the grippers. To avoid grasping the bin, we constrained grasps to the foreground by subtracting out the background pixels of the bin using a reference depth image of an empty bin. Grasps were also constrained to be collision free with the bin to avoid damage to the robot. Given the constraints, CEM sampled a set of initial candidate grasps uniformly at random from a point cloud and then iteratively resampled grasps from a Gaussian mixture model fit to the grasps with the highest estimated quality. For the suction-cup gripper, initial candidate grasps were sampled by selecting a 3D point and choosing an approach direction along the inward-facing surface normal. For the parallel-jaw gripper, initial candidate grasps were sampled by finding antipodal point pairs using friction cone analysis.

Study design

The objective of the UP benchmark is to measure the rate, reliability, and range of the Dex-Net 4.0 policy in reference to baseline methods. The number of trials and objects used in the benchmark was chosen to maximize the number of unique grasp attempts and baseline policies that could be evaluated in a fixed time budget. The objects were divided into categories of 25 objects each based on difficulty to quantify the range of each policy. Rather than experiment on hundreds of unique objects, we used a reduced subset of 75 to evaluate a larger number of grasps for each object given a 3-week time budget for experiments. A grasp was considered successful if it lifted and transported exactly one object from the bin to the receptacle. Successes and failures were labeled by a human operator to avoid labeling errors due to hardware failures or sensor calibration.

Experimental hardware setup

The experimental workspace is illustrated in Fig. 2. The benchmark hardware system consists of an ABB YuMi bimanual industrial collaborative robot with an overhead Photoneo PhoXi S industrial 3D scanner, a custom suction gripper, and custom 3D-printed parallel-jaw fingers with silicone fingertips (*57*). The suction gripper consists of a 20-mm-diameter silicone single-bellow suction cup seated in a 3D-printed housing mounted to the end of the right arm. The vacuum was created by supplying compressed air from a JUN-AIR 18-40 quiet air compressor to a VacMotion MSV-35 vacuum generator. The payload of the suction system was about 0.9 kg with a vacuum flow of about 8 standard cubic feet/min. Objects were grasped from a bin mounted on top of a set of Loadstar load cells that measured the weight with a resolution of about 5 g. Each gripper has a separate receptacle to drop the objects into on the side of the bin.

Experimental procedure

Each experiment consisted of five independent trials in which the bin was filled with a random configuration of one or more objects and the robot attempted to pick each object from the bin and transport it to a receptacle. Before each experiment, the camera position and orientation relative to the robot were measured using a chessboard. In each trial, the operator set a full dataset of objects in the bin by shaking the objects in a box, placing the box upside down in the bin, and mixing the bin manually to ensure that objects rested below the rim of the bin. Then, the

robot iteratively attempted to pick objects from the bin. On each attempt, the grasping policy received as input a point cloud of the objects in the bin and returned a grasp action for exactly one of the grippers, consisting of a pose for the gripper relative to the base of the robot. Then, the ABB RAPID linear motion planner and controller were used to move to the target pose, establish contact with the object, and drop the object in the receptacle. The operator labeled the grasp as successful if the robot lifted and transported the object to the receptacle on the side of the workspace. The operator also labeled the identity of each grasped object. A trial was considered complete after all objects were removed, 75 total attempts, or 10 consecutive failures. All experiments ran on a desktop running Ubuntu 16.04 with a 3.4-GHz Intel Core i7-6700 quad-core central processing unit and an NVIDIA TITAN Xp GPU [see the Supplementary Materials for a characterization of variables in the benchmark (*58*)].

Description of baselines

We compared performance with three baselines:

1) Heuristic (suction). Ranked planar grasps based on the inverse distance to the centroid of an object (*30*), where the object centroid was estimated as the mean pixel of an object instance segmask from a Euclidean clustering segmentation algorithm from the Point Cloud Library (PCL) (*59*). Planarity was determined by evaluating the mean squared error (MSE) of all 3D points within a sphere with a radius of 10 mm (based on the suction cup size) to the best-fit plane for the points. Grasps were considered planar if either (i) the MSE was less than an absolute threshold or (ii) the MSE was within the top 5% of all candidate grasps. The hyperparameters were hand-coded to optimize performance on the physical robot.

2) Heuristic (composite). Ranked grasps planned with the suction heuristic above and a parallel-jaw heuristic based on antipodality. The parallel-jaw heuristic ranked antipodal grasps based on the inverse distance to the estimated centroid of an object, determining antipodality based on estimated point cloud surface normals. The height of the gripper above the bin surface was a constant offset from the highest point within the region of the grasp. The grasp closest to the estimated object centroid across both grippers was selected for execution.

3) Dex-Net 2.0 and 3.0 composite. Ranked grasps based on the estimated quality from separate GQ-CNNs trained to estimate the quality of parallel-jaw and suction-cup grasps in clutter. The GQ-CNNs were trained by fine-tuning the Dex-Net 2.0 and 3.0 base networks on simulated heaps with imitation learning from an algorithmic supervisor (*26*).

Details of empirical training

We train 10 variants of the Dex-Net 4.0 ambidextrous grasping policy on a dataset of 13,000 real grasp attempts: training from scratch and fine-tuning either all FC layers or only the last FC layer (fc5) on varying ratios of real to simulated data: 1:0, 1:1, 1:10, and 1:100. Each variant was evaluated on the adversarial level 3 objects on the physical robot, and the highest performing policy was the Dex-Net 4.0 policy with the last FC layer fine-tuned on the 1:10 combined real and synthetic dataset.

Details of memory system

To avoid repeated grasp failures, we implemented a first-order memory system that associates regions of the point cloud with past failures. A grasp was considered a failure if the weight reading from the load cells changed less than 5 g after a grasp attempt. When a failure occurred, the point cloud was segmented using PCL (*59*), and the segment corresponding to the grasped object was associated with a region in a grayscale image. The segmented image patch was featurized using the VGG-16 network and stored in a failure database corresponding to the gripper. On the next grasp attempt, the point cloud segments were matched to the failure database using the VGG-16 featurization. If a match was found, the region in the current image was marked as forbidden to the grasp sampler for the Dex-Net 4.0 policy. Furthermore, if more than three consecutive failures occurred, then the memory system rejected the planned grasp and used a pushing policy (*60*) to perturb the objects in the bin.

SUPPLEMENTARY MATERIALS

robotics.sciencemag.org/cgi/content/full/4/26/eaau4984/DC1

Text

Fig. S1. Analysis of features learned by the GQ-CNNs from the ambidextrous grasping policy.

Fig. S2. Per-object reliability of each policy on each test object.

Fig. S3. Difficulty of each object from the test object datasets characterized by the overall reliability averaged across methods.

Movie S1. Summary.

Raw data, code for data analysis, videos, and listing of objects used in experiments (.zip file)

REFERENCES AND NOTES

1. R. M. Murray, Z. Li, S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation* (CRC Press, 1994).
2. D. Prattichizzo, J. C. Trinkle, *Springer Handbook of Robotics* (Springer, 2008), pp. 671–700.
3. B. Kehoe, A. Matsukawa, S. Candido, J. Kuffner, K. Goldberg, Cloud-based robot grasping with the google object recognition engine, in *2013 IEEE International Conference on Robotics and Automation* (IEEE, 2013), pp. 4263–4270.
4. J. Weisz, P. K. Allen, Pose error robust grasping from contact wrench space metrics, in *2012 IEEE International Conference on Robotics and Automation* (IEEE, 2012), pp. 557–562.
5. R. Balasubramanian, L. Xu, P. D. Brook, J. R. Smith, Y. Matsuoka, Physical human interactive guidance: Identifying grasping principles from human-planned grasps. *IEEE Trans. Robot.* **28**, 899–910 (2012).
6. J. Bohg, A. Morales, T. Asfour, D. Kragic, Data-driven grasp synthesis—A survey. *IEEE Trans. Robot.* **30**, 289–309 (2014).
7. D. Kappler, J. Bohg, S. Schaal, Leveraging big data for grasp planning, in *2015 IEEE International Conference on Robotics and Automation* (ICRA, 2015), pp. 4304–4311.
8. I. Lenz, H. Lee, A. Saxena, Deep learning for detecting robotic grasps. *Int. J. Robot. Res.* **34**, 705–724 (2015).
9. A. Saxena, J. Driemeyer, A. Y. Ng, Robotic grasping of novel objects using vision. *Int. J. Rob. Res.* **27**, 157–173 (2008).
10. S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, D. Quillen, Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *Int. J. Rob. Res.* **37**, 421–436 (2018).
11. L. Pinto, A. Gupta, Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours, in *IEEE International Conference on Robotics and Automation* (ICRA, 2016), pp. 3406–3413.
12. A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, T. Funkhouser, Learning synergies between pushing and grasping with self-supervised deep reinforcement learning, in *IEEE/RSJ International Conference on Intelligent Robots and Systems* (IROS, 2018).
13. D. Ma, A. Rodriguez, Friction variability in planar pushing data: Anisotropic friction and data-collection bias. *IEEE Robot. Autom. Lett.* **3**, 3232–3239 (2018).
14. S. James, A. J. Davison, E. Johns, Transferring end-to-end visuomotor control from simulation to real world for a multi-stage task, in *1st Conference on Robot Learning* (CoRL, 2017), pp. 334–343.
15. X. B. Peng, M. Andrychowicz, W. Zaremba, P. Abbeel, Sim-to-real transfer of robotic control with dynamics randomization, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2018), pp. 3803–3810.
16. A. A. Rusu, M. Večerík, T. Rothörl, N. Heess, R. Pascanu, R. Hadsell, Sim-to-real robot learning from pixels with progressive nets, in *1st Conference on Robot Learning* (CoRL, 2017), pp. 262–270.
17. J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, P. Abbeel, Domain randomization for transferring deep neural networks from simulation to the real world, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2017), pp. 23–30.

18. E. Johns, S. Leutenegger, A. J. Davison, Deep learning a grasp function for grasping under gripper pose uncertainty, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2016), pp. 4461–4468.
19. J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, K. Goldberg, Dex-Net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics, in *Proceedings of Robotics: Science and Systems* 10.15607/RSS.2017.XIII.058 (2017).
20. OpenAI, Learning dextrous in-hand manipulation. arXiv:1808.00177 [cs.LG] (1 August 2018).
21. A. ten Pas, M. Gualtieri, K. Saenko, R. Platt, Grasp pose detection in point clouds. *Int. J. Robot. Res.* **36**, 1455–1473 (2017).
22. F. Sadeghi, S. Levine, CAD2RL: Real single-image flight without a single real image, in *Proceedings of Robotics: Science and Systems* 10.15607/RSS.2017.XIII.034 (2017).
23. M. Danielczuk, M. Matl, S. Gupta, A. Li, A. Lee, J. Mahler, K. Goldberg, Segmenting unknown 3D objects from real depth images using mask R-CNN trained on synthetic point clouds. arXiv:1809.05825 [cs.CV] (16 September 2018).
24. D. Seita, N. Jamali, M. Laskey, R. Berenstein, A. K. Tanwani, P. Baskaran, S. Iba, J. Canny, K. Goldberg, Robot bed-making: Deep transfer learning using depth sensing of deformable fabric. arXiv:1809.09810 [cs.RO] (26 September 2018).
25. K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige, S. Levine, V. Vanhoucke, Using simulation and domain adaptation to improve efficiency of deep robotic grasping, in *IEEE International Conference on Robotics and Automation* (ICRA, 2018), pp. 4243–4250.
26. J. Mahler, K. Goldberg, Learning deep policies for robot bin picking by simulating robust grasping sequences, in *1st Conference on Robot Learning* (CoRL, 2017), pp. 515–524.
27. J. Mahler, M. Matl, X. Liu, A. Li, D. Gealy, K. Goldberg, Dex-Net 3.0: Computing robust vacuum suction grasp targets in point clouds using a new analytic model and deep learning, in *IEEE International Conference on Robotics and Automation* (ICRA, 2018), pp. 5620–5627.
28. U. Viereck, A. ten Pas, K. Saenko, R. Platt, Learning a visuomotor controller for real world robotic grasping using easily simulated depth images, in *1st Conference on Robot Learning* (CoRL, 2017), pp. 291–300.
29. C. Hernandez, M. Bharatheesha, W. Ko, H. Gaiser, J. Tan, K. van Deurzen, M. de Vries, B. Van Mil, J. van Egmond, R. Burger, M. Morariu, J. Ju, X. Gerrmann, R. Ensing, J. Van Frankenhuyzen, M. Wisse, Team Delft's robot winner of the amazon picking challenge 2016. arXiv:1610.05514 [cs.RO] (18 October 2016).
30. D. Morrison, A. W. Tow, M. McTaggart, R. Smith, N. Kelly-Boxall, S. Wade-McCue, J. Erskine, R. Grinover, A. Gurman, T. Hunn, D. Lee, A. Milan, T. Pham, G. Rallos, A. Razjigaev, T. Rowntree, K. Vijay, Z. Zhuang, C. Lehnert, I. Reid, P. Corke, J. Leitner, Cartman: The low-cost cartesian manipulator that won the amazon robotics challenge, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2018), pp. 7757–7764.
31. K.-T. Yu, N. Fazeli, N. Chavan-Dafle, O. Taylor, E. Donlon, G. D. Lankenau, A. Rodriguez, A summary of team MIT's approach to the amazon picking challenge 2015. arXiv:1604.03639 [cs.RO] (13 April 2016).
32. A. Zeng, S. Song, K.-T. Yu, E. Donlon, F. R. Hogan, M. Bauzá, D. Ma, O. Taylor, M. Liu, E. Romo, N. Fazeli, F. Alet, N. Chavan-Dafle, R. Holladay, I. Morona, P. Q. Nair, D. Green, I. Taylor, W. Liu, T. A. Funkhouser, A. Rodriguez, Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2018), pp. 3750–3757.
33. L. P. Kaelbling, M. L. Littman, A. R. Cassandra, Planning and acting in partially observable stochastic domains. *Artif. Intell.* **101**, 99–134 (1998).
34. M. Jaśkowski, J. Świątkowski, M. Zając, M. Klimek, J. Potiuk, P. Rybicki, P. Polatowski, P. Walczyk, K. Nowicki, M. Cygan, Improved GQ-CNN: Deep learning model for planning robust grasps. arXiv:1802.05992 [cs.LG] (16 February 2018).
35. R. Krug, Y. Bekiroglu, M. A. Roa, Grasp quality evaluation done right: How assumed contact force bounds affect wrench-based quality metrics, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2017), pp. 1595–1600.
36. F. T. Pokorny, D. Kragic, Classical grasp quality evaluation: New algorithms and theory, in *IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2013), pp. 3493–3500.
37. R. Deimel, O. Brock, A novel type of compliant and underactuated robotic hand for dexterous grasping. *Int. J. Robot. Res.* **35**, 161–185 (2016).
38. A. Gupta, A. Murali, D. Gandhi, L. Pinto, Robot learning in homes: Improving generalization and reducing dataset bias. arXiv:1807.07049 [cs.RO] (18 July 2018).
39. J. Friedman, T. Hastie, R. Tibshirani, *The Elements of Statistical Learning*, vol. 1 (Springer Series in Statistics, Springer, 2001).
40. S. Ross, G. Gordon, D. Bagnell, A reduction of imitation learning and structured prediction to no-regret online learning, in *Fourteenth International Conference on Artificial Intelligence and Statistics* (AISTATS, 2011), pp. 627–635.
41. A. G. Barto, *Reinforcement Learning: An Introduction* (MIT Press, 1998).
42. D. Morrison, J. Leitner, P. Corke, Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach, in *Proceedings of Robotics: Science and Systems* 10.15607/RSS.2018.XIV.021 (2018).
43. F. R. Hogan, E. R. Grau, A. Rodriguez, Reactive planar manipulation with convex hybrid MPC, in *IEEE International Conference on Robotics and Automation* (ICRA, 2018), pp. 247–253.
44. L. Righetti, M. Kalakrishnan, P. Pastor, J. Binney, J. Kelly, R. C. Voorhies, G. S. Sukhatme, S. Schaal, An autonomous manipulation system based on force control and optimization. *Auton. Robots* **36**, 11–30 (2014).
45. M. R. Tremblay, M. R. Cutkosky, Estimating friction using incipient slip sensing during a manipulation task, in *Proceedings IEEE International Conference on Robotics and Automation* (IEEE, 1993), pp. 429–434.
46. R. Calandra, A. Owens, D. Jayaraman, J. Lin, W. Yuan, J. Malik, E. H. Adelson, S. Levine, More than a feeling: Learning to grasp and regrasp using vision and touch, in *IEEE Robotics and Automation Letters* (IEEE, 2018), pp. 3300–3307.
47. F. R. Hogan, M. Bauza, O. Canal, E. Donlon, A. Rodriguez, Tactile regrasp: Grasp adjustments via simulated tactile transformations. arXiv:1803.01940 [cs.RO] (5 March 2018).
48. R. D. Howe, Tactile sensing and control of robotic manipulation. *Adv. Robot.* **8**, 245–261 (1993).
49. A. Molchanov, O. Kroemer, Z. Su, G. S. Sukhatme, Contact localization on grasped objects using tactile sensing, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2016), pp. 216–222.
50. R. Hartley, A. Zisserman, *Multiple View Geometry in Computer Vision* (Cambridge Univ. Press, 2003).
51. E. Coumans, Bullet physics library (2013); bulletphysics.org.
52. K. Goldberg, B. V. Mirtich, Y. Zhuang, J. Craig, B. R. Carlisle, J. Canny, Part pose statistics: Estimators and experiments. *IEEE Trans. Rob. Autom.* **15**, 849–857 (1999).
53. J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. Kuffner, K. Goldberg, Dex-Net 1.0: A cloud-based network of 3D objects for robust grasp planning using a multi-armed bandit model with correlated rewards, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2016), pp. 1957–1964.
54. J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, R. Hadsell, Overcoming catastrophic forgetting in neural networks. *Proc. Natl. Acad. Sci. U.S.A.* **13**, 3521–3526 (2017).
55. K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification, in *IEEE International Conference on Computer Vision (ICCV)* (IEEE, 2015), pp. 1026–1034.
56. R. Y. Rubinstein, A. Ridder, R. Vaisman, *Fast Sequential Monte Carlo Methods for Counting and Optimization* (John Wiley & Sons, 2013).
57. M. Guo, D. V. Gealy, J. Liang, J. Mahler, A. Goncalves, S. McKinley, J. A. Ojea, K. Goldberg, Design of parallel-jaw gripper tip surfaces for robust grasping, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2017), pp. 2831–2838.
58. J. Mahler, R. Platt, A. Rodriguez, M. Ciocarlie, A. Dollar, R. Detry, M. A. Roa, H. Yanco, A. Norton, J. Falco, K. van Wyk, E. Messina, J. J. Leitner, D. Morrison, M. Mason, O. Brock, L. Odhner, A. Kurenkov, M. Matl, K. Goldberg, Guest editorial open discussion of robot grasping benchmarks, protocols, and metrics. *IEEE Trans. Autom. Sci. Eng.* **15**, 1440–1442 (2018).
59. R. B. Rusu, S. Cousins, 3D is here: Point Cloud Library (PCL), in *IEEE International Conference on Robotics and Automation* (IEEE, 2011), pp. 1–4.
60. M. Danielczuk, J. Mahler, C. Correa, K. Goldberg, Linear push policies to increase grasp access for robot bin picking, in *IEEE 14th International Conference on Automation Science and Engineering (CASE)* (IEEE, 2018), pp. 1249–1256.

Acknowledgments: This research was performed at the AUTOLAB at UC Berkeley in affiliation with the Berkeley AI Research (BAIR) Lab, the Real-Time Intelligent Secure Execution (RISE) Lab, and the CITRIS "People and Robots" (CPAR) Initiative. We thank those who aided in experiments: L. Amladi, C. Correa, S. Dolasia, D. Gealy, and M. Guo. We also thank our colleagues who provided helpful feedback, code, and suggestions, particularly R. Bajcsy, O. Brock, M. Laskey, S. Krishnan, L. Manuelli, J. A. Ojea, P. Puchwein, A. Rodriguez, and D. Seita. **Funding:** We were supported, in part, by donations from Siemens, Google, Amazon Robotics, Toyota Research Institute, Autodesk, ABB, Samsung, Knapp, Loccioni, Honda, Intel, Comcast, Cisco, and Hewlett-Packard; by equipment grants from Photoneo and NVIDIA, by the U.S. Department of Defense (DOD) through the National Defense Science and Engineering Graduate Fellowship (NDSEG) Program; and by the Scalable Collaborative Human-Robot Learning (SCHooL) Project, NSF National Robotics Initiative Award 1734633. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the sponsors. **Author contributions:** J.M. devised the training method, designed the experiments, and wrote the manuscript. M.M. helped implement physics simulations, implemented the memory system, conducted

experiments, and edited the manuscript. V.S. implemented the neural network training, conducted experiments, and edited the manuscript. M.D. helped implement physics simulations, implemented the pushing policy used in the memory system, conducted experiments, and edited the manuscript. B.D. designed the experimental object datasets, conducted experiments, and edited the manuscript. S.M. designed and constructed the robotic picking cell used in experiments, conducted experiments, and edited the manuscript. K.G. supervised the project, advised the design and experiments, and edited the manuscript. **Competing interests:** J.M., M.M., S.M., and K.G. have equity in Ambidextrous Robotics Inc. J.M., M.M., and K.G. are co-inventors on a patent application (no. PCT/US18/26122) related to this work. **Data and materials availability:** All data needed to evaluate the conclusions in the paper are present in the paper or the Supplementary Materials.

Submitted 4 September 2018
Accepted 12 December 2018
Published 16 January 2019
10.1126/scirobotics.aau4984

Citation: J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, K. Goldberg, Learning ambidextrous robot grasping policies. *Sci. Robot.* **4**, eaau4984 (2019).

Science Robotics

Learning ambidextrous robot grasping policies

Jeffrey Mahler, Matthew Matl, Vishal Satish, Michael Danielczuk, Bill DeRose, Stephen McKinley and Ken Goldberg

Sci. Robotics **4**, eaau4984.
DOI: 10.1126/scirobotics.aau4984