EGAD! an Evolved Grasping Analysis Dataset
for diversity and reproducibility in robotic manipulation

Douglas Morrison[1], Peter Corke[1] and Jürgen Leitner[1,2]

Abstract— We present the Evolved Grasping Analysis Dataset (EGAD), comprising over 2000 generated objects aimed at training and evaluating robotic visual grasp detection algorithms. The objects in EGAD are geometrically diverse, filling a space ranging from simple to complex shapes and from easy to difficult to grasp, compared to other datasets for robotic grasping, which may be limited in size or contain only a small number of object classes. Additionally, we specify a set of 49 diverse 3D-printable evaluation objects to encourage reproducible testing of robotic grasping systems across a range of complexity and difficulty. The dataset, code and videos can be found at `https://dougsm.github.io/egad/`

I. INTRODUCTION

The ability to grasp previously unseen objects is a fundamental trait for robots that need to interact with their environments, and underpins many higher-level manipulation capabilities. The last few years have seen a large amount of work focused on visual grasp detection, greatly driven by advanced deep learning techniques. As such, the need for diverse object dataset specific to robotic grasping is crucial for both training and evaluating these systems.

The need for large and diverse datasets for training robust deep learning algorithms that generalise well to unknown conditions is widely recognised. However, many current visual grasp detection algorithms are trained on either very small, manually collected datasets, or datasets of objects adapted from other domains with a small number of semantic classes, which may not be representative of the type of challenges faced in robotic grasping.

Furthermore, there currently exists very little standardisation between the physical objects used for evaluating robotic grasping algorithms. While some physical object datasets do exist, they have not been widely adopted by the robotic grasping community. Instead, researchers tend to test their algorithms using sets of random "household" objects, relying largely on the author's intuition as to the diversity and complexity of the test set, making effective comparison difficult. For evaluation, results are typically reported as the average grasp success rate over an object set, however this does not allow for easy identification of system limitations or performance of a function of object difficulty or complexity.

To address these issues, we use evolutionary algorithms to generate a dataset of objects that is diverse in the space of
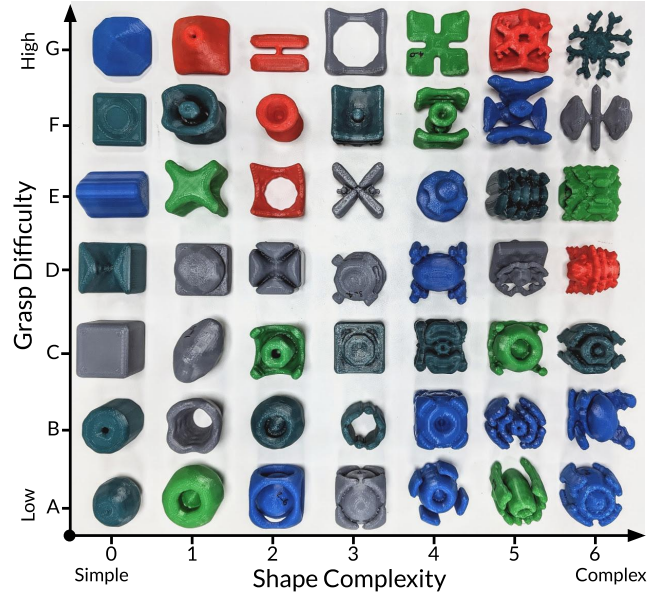
Fig. 1. 49 3D-printed evaluation objects chosen from over 2000 diverse objects in EGAD. The objects provide a range of objects from simple to complex geometry (left to right), and easy to difficult graspability (bottom to top), allowing for improved and reproducible evaluation of robotic grasping algorithms.

shape complexity, grasp difficulty and geometric similarity, aimed specifically at training and evaluating visual grasp detection algorithms. The Evolved Grasping Analysis Dataset (EGAD) contains over 2000 generated objects, including a specified 3D-printable subset of 49 evaluation objects (Fig. 1). As such, EGAD can be used for training and evaluating robotic grasping algorithms in both simulation and the real world. To summarise our contributions, in this paper we:

- Use evolutionary algorithms to create EGAD, a large dataset of over 2000 diverse objects, which fill a space of both shape complexity and grasp difficulty;
- Release EGAD as both 3D mesh files and in the commonly used Dex-Net [1] database format, with over 1 million precomputed grasp poses and the ability to easily create vision-based datasets for training grasp prediction networks;
- Specify a set of 49 diverse objects from the dataset which can be used as a reproducible, real-world testing suite, along with guidelines for reproducing objects and reporting results; and
- Perform robotic experiments using a state-of-the-art grasp detection algorithm as a template for using the evaluation set, and use the results to gain insights for future improvement of the algorithm.

II. RELATED WORK

A. Visual Grasp Detection

Within the larger scope of robotic manipulation, visual grasp detection algorithms are used to predict the poses of stable robotic grasps on previously unseen objects using visual information. Many different approaches to visual grasp detection have been proposed in recent years, with the majority using deep learning techniques [1–22].

Such approaches provide two key challenges. The first is the availability of diverse and high quality data, which is widely recognised as an important prerequisite for training robust and generalisable models [23–25]. The second is providing standardised, reproducible and comparable evaluation methods and metrics.

In other domains, large, curated datasets exist to accelerate research and standardise evaluation, e.g. ImageNet [26] or COCO [27] for computer vision. However, the physical nature of robotics makes benchmarking and comparing work in robotic manipulation particularly difficult, with results easily influenced by robotic hardware and choice of evaluation objects. Mahler *et al.* [28] provide a set of best practices for robotic grasping to ensure fair evaluation and comparison across all system aspects. In this work we focus specifically on the aspect of object sets.

B. Object Datasets for Robotic Manipulation

The YCB object set [29] comprises a set of common household objects along with high-resolution 3D scans, aimed at evaluating robotic grasping and high-level robotic manipulation tasks. However, the physical nature of this dataset means it is limited in size and object diversity from a training point of view.

On the other hand, virtual datasets for robotic grasping have been created by repurposing existing databases of 3D object meshes. Goldfeder *et al.* [30] created the Columbia Grasping Dataset using 1800 3D meshes from the Princeton Shape Benchmark (PSB) [31]. Similarly, Mahler *et al.* [1] compiled the Dex-Net dataset from 1500 3D meshes from 3DNet [32] and the KIT object database [33]. Such datasets have proven vital for training machine-learning based grasp detection algorithms. However, the underlying object sets are derived from 3D object recognition tasks which contain only small number of semantic classes (10 classes for 3DNet), resulting in low geometric diversity within the data sets.

Rather than rely on realistic object models, Tobin *et al.* [11] show that simulated objects, generated by randomly combining convex shape primitives, can be successfully used to train a grasping algorithm that generalises to real-world objects. We build on this promising result by using creating a dataset of objects that are geometrically diverse and provide a gradient of grasping difficultly and shape complexity.

The Cornell Grasping Dataset [34] provides 885 top-down RGB-D images of single objects placed on a table, hand-labelled with positive and negative grasp examples represented by a rectangle. Due to the manual collection process, the dataset is limited in size, containing only approximately

TABLE I

SURVEY OF USE OF EVALUATION DATASETS IN VISUAL GRASP DETECTION LITERATURE. THE MAJORITY OF WORK USES UNREPRODUCIBLE "HOUSEHOLD" OBJECTS FOR EVALUATION.

Reference	Evaluation Objects						
Lenz *et al.* [2], 2015					C		R
Redmon *et al.* [3], 2015					C		
Pinto *et al.* [4], 2016							R
Johns *et al.* [5], 2016						S	R
Wang *et al.* [6], 2016					C		R
Mahler *et al.* [1], 2017	D				C		R
Pas *et al.* [7], 2017							R
Mahler *et al.* [8], 2017							R
Kumra *et al.* [9], 2017					C		
Viereck *et al.* [10], 2017						S	R
Tobin *et al.* [11], 2017			Y				
Zhou *et al.* [12], 2018					C		
Depierre *et al.* [13], 2018				J	C		R
Chu *et al.* [14], 2018					C		R
Morrison *et al.* [15], 2018	D	A	Y				
Morrison *et al.* [17], 2019	D	A	Y	J	C		R
Morrison *et al.* [16], 2019	D						R
Mahler *et al.* [18], 2019							R
Satish *et al.* [19], 2019	D					S	R
Asif *et al.* [20], 2019					C		R
Yu *et al.* [21], 2019							R
Liang *et al.* [22], 2019			Y				R

Legend (See text for full details)	
Random ("Household") Objects	R
Objects in Simulation	S
Cornell Grasping Dataset (IoU Metric) [34]	C
Jacquard SGTs [13]	J
YCB Objects (or subset) [29]	Y
APB Objects (or subset) [35]	A
Dex-Net Adversarial Objects [1]	D

8000 labelled grasps. The Jacquard dataset [13] overcomes this limitation by using a simulator to generate 54k images of 11k objects, labelled with over 1 million grasps using the rectangle representation. However, as these datasets are image-based and don't provide 3D models of objects, they are limited to training for top-down, tabletop grasping.

C. Evaluation for Robotic Grasping

Previous work has specified datasets of physical objects and protocols for manipulation, with notable examples being the YCB dataset [29] and ACRV picking benchmark [35]. However, despite the prevalence of these datasets, Table I clearly shows that neither has become commonplace for evaluating visual grasp detection systems. Instead, authors often opt to test on sets of random "household" objects, making comparing results between different algorithms very difficult, as the decision of which items are included is ultimately left to the intuition of the researchers.

Mahler *et al.* [1] proposed a set of eight adversarial objects with complex geometry which allow testing of algorithms under difficult conditions. The objects are easily reproducible with 3D-printing, removing the need to purchase matching
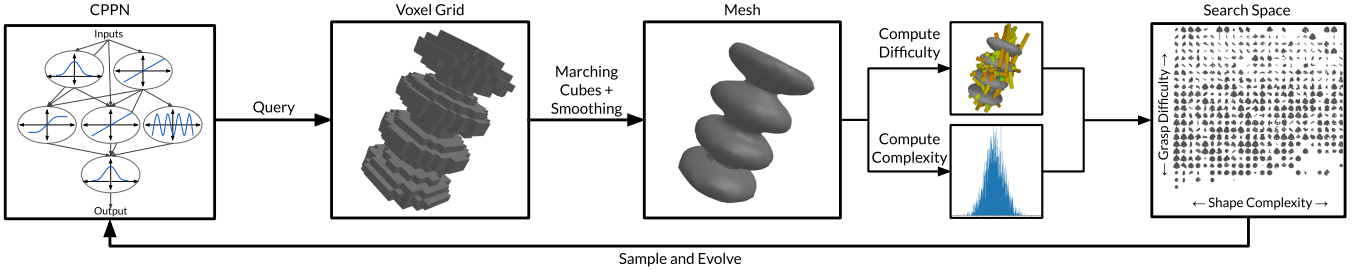
Fig. 2. Overview of our method for creating EGAD. We define a discretised search space for objects in terms of shape complexity and grasp difficulty. Shapes are encoded using 3D CPPNs, which are queried to generate a voxel grid and processed into a 3D mesh. We compute the shape complexity and grasp difficulty metrics to place the object in a cell of the search space. Only the most geometrically diverse objects are kept at each cell. At each iteration, objects are are sampled from the search space and evolved to create new objects, until the search space is full of diverse objects.

objects. As such, the objects are also scalable relative to gripper size, allowing for fairer comparisons between robotic systems. We expand on this idea by specifying a larger and more diverse set of reproducible objects of varying complexity and difficulty that can be used to comprehensively evaluate a visual grasp detection algorithm.

A common method of evaluation is an offline metric using the Cornell Grasping Dataset or Jacquard dataset. A predicted grasp is successful if it has an intersection-over-union (IoU) of greater than 25% with and is within $30°$ of a positive labelled grasp when using the grasping rectangle representation [34]. While reproducible, this metric is susceptible to a large number of false-positive and false-negative detections due to the sparse labelling of the dataset, and low requirements for considering a match. The Jacquard dataset [13] provides a cloud-based physics simulator where results are evaluated using Simulated Grasping Trials (SGTs), however this relies on a closed-source evaluation server. Recent work has also shown that offline performance on either dataset may not be representative of real-world performance due to the domain shift from the dataset to reality [17].

III. Dataset Generation

In order to facilitate both learning and evaluation for robotic grasping, we aim to provide a large set of objects that is diverse in terms of geometry, shape complexity and grasp difficulty. For this we use evolutionary algorithms. Compared to gradient based methods, the benefit of evolutionary algorithms in this case is the ability to handle complex, non-linear objectives in a high-dimensional design space. Evolutionary algorithms have been widely used to generate complex 3D designs, with applications ranging from art [36] to engineering [37] and robotics [38, 39].

To summarise our approach, we first define a discretised two-dimensional search space for objects in terms of shape complexity and grasp difficulty. Objects are encoded using Compositional Pattern Producing Networks (CPPNs) [40, 41], from which 3D meshes are obtained. We compute the grasp difficulty and shape complexity of each mesh to assign it to a cell in the search space. New objects are compared to all other objects in the search space, and only the most geometrically diverse objects in each cell are kept. The dataset is evolved using the MAP-Elites algorithm [39], where at each iteration, objects are sampled from the search

space and evolved into new, different shapes. This process is repeated until the search space is full of diverse objects. An overview of our approach is shown in Fig. 2, and the following sections describe each component in more detail.

A. Search Space

Our search space is defined by two features, shape complexity and grasp difficulty, and is discretised uniformly into cells. Given an object represented by a 3D triangular mesh, it is assigned to a cell in the search space using the shape complexity and grasp difficulty metrics defined below. A maximum number of objects is allowed at each cell. If the number of objects in a cell exceeds the maximum, the objects are compared to all other objects in the search space, and the least geometrically diverse object (as per the metric below) is removed. This ensures that the search space is filled uniformly and with geometrically diverse objects.

1) Shape Complexity: To compute a measure of shape complexity, we use the measure of morphological complexity from [42, 43]. The measure is based in information theory and has been shown to also correlate well with humans' intuition about shape complexity [44]. To compute the complexity metric for a given mesh, we first compute the angular deficit Φ_j for each vertex j:

$$\Phi_j = 2\pi - \sum_i \phi_i \quad (1)$$

where ϕ_i is the internal angle of each triangle i where it meets vertex j. The deficit values are placed in a histogram over the range $[-2\pi, 2\pi)$ with bin width Δ, which is normalised as a probability density function (PDF) such that each bin b contains a probability $p(\Phi_b)$. The shape complexity is then equivalent to the entropy of the PDF:

$$H = -\sum_b p(\Phi_b) \log p(\Phi_b) \quad (2)$$

2) Grasp Difficulty: To estimate a single scalar feature representing grasping difficulty per object, we use the 75th percentile method described by Wang *et al.* [45]. Using the Dex-Net analytical grasp planner [1], we sample a number of antipodal grasps on each object and compute the robust Ferrari-Canny quality metric for each. The grasp difficulty feature is then obtained by taking the 75th percentile grasp quality of all sampled grasps.

TABLE II
PARAMETERS USED FOR DATASET GENERATION

Parameter	Value
Population Size	100
Evolution Steps	200,000 (2000 steps $\times$ population 100)
Search Space Size	25$\times$25
Max Objects per Cell	4
Probability of Crossover	0.5
Difficulty Feature Range	[0.0005, 0.004]
Complexity Feature Range	[1, 5]
Histogram Bin Width (Δ)	$4\pi/512$
Sampled Grasps per Object	100
Diversity neighbours (k)	10
CPPN Resolution	$25 \times 25 \times 25$
CPPN Inputs	$x, y, z, \sqrt{x^2 + y^2}, \sqrt{x^2 + z^2},$ $\sqrt{y^2 + z^2}, \sqrt{x^2 + y^2 + z^2}$ $x, y, z \in [-1, 1]$
CPPN Activation Functions	sin, sigmoid, gaussian, idenity

3) Geometric Diversity: In order to compute the geometric diversity of an object, we first define a metric of geometric similarity between any two objects. We use the Topology Matching metric based on Multiresolutional Reeb Graphs (MRGs) proposed by Hilaga *et al.* [46], which provides a shape similarity score *sim* $\in [0, 1]$ between two object meshes that is robust to translation, rotation, scale and changes in mesh connectivity (e.g. through mesh resampling or decimation). This method has been shown to work effectively with arbitrary meshes and CAD models [46, 47]. We define the distance *dist* between two object meshes m_1 and m_2 as the inverse of similarity:

$$dist(m_1, m_2) = 1 - sim(m_1, m_2) \quad (3)$$

Similar to [48], we then define the diversity of a mesh $\rho(m)$ as the mean distance to the k most similar meshes to m in the whole search space:

$$\rho(x) = \frac{1}{k} \sum_{i=1}^{k} \text{dist}(m, m_i) \quad (4)$$

B. Evolutionary Algorithm

1) Shape Encoding: We use Compositional Pattern Producing Networks (CPPNs) [40, 41] to encode and generate 3D shapes. Each CPPN is an arbitrary neural network, which allows for a compact, functional representation of a 3D volume obtained by querying the network at discrete spatial coordinates (e.g. x, y and z), and thresholding the scalar output to create a voxel grid. The voxel grid is converted to a triangle mesh representation using marching cubes [49] and smoothing is applied. In the case that multiple disconnected meshes are generated, we keep only the largest volume mesh. Small features that would prevent 3D-printing are removed by performing a morphological opening.

CPPNs are evolved by the principles of NeuroEvolution of Augmenting Topologies (NEAT) [50]. At each evolution, the CPPN architecture is mutated by randomly adding and removing network nodes and connections, and changing weights, biases and activation functions. CPPNs also undergo crossover, where components of two CPPN architectures are combined into a new CPPN. As such, the CPPNs and their resulting shapes build up complexity and diversity over time.

2) Search Algorithm: Many traditional evolutionary algorithms are very effective optimisation algorithms, but are susceptible finding local minima and not exploring the search space. To overcome this, we use the Multi-dimensional Archive of Phenotypic Elites (MAP-Elites) algorithm [39].

Our implementation of MAP-Elites begins with a population of randomly initialised CPPNs, which are queried and placed into their respective cells of the search space. At each subsequent iteration, a population is randomly sampled from the search space to undergo evolution to produce a new population of objects, which are subsequently evaluated and assigned to cells in the search space. If any cell contains more than the maximum number of individuals, the lowest performing (least geometrically diverse) occupants are iteratively removed until all cells contain at most the maximum number of occupants, with the diversity of all occupants being recalculated after each removal.

IV. EVOLVED GRASPING ANALYSIS DATASET

Using the method described in the previous section and the parameters defined in Table II, we generate a set of objects which we call the Evolved Grasping Analysis Dataset (EGAD). In this section we present the dataset with some analysis of the objects and provide a brief comparison to other existing grasping datasets.

A. Overview

In total EGAD contains 2331 objects which uniformly fill the search space. While the total size of the search space is 2500 (25×25 cells $\times$ 4 objects per cell), some cells at the extremes were unable to be filled, e.g. objects that are in the simplest geometric column but difficult to grasp, or geometrically complex and easy to grasp. The coverage of EGAD is thus 93% of the search space. The distribution of objects and their diversity is shown in Fig. 3.

The diversity of the objects ranges from 0.07 to 0.27, with a mean of 0.19. Unsurprisingly, the lowest diversity objects largely found in the area of simple shape complexity, and the object diversity increases with higher shape complexity.

Many of the objects exhibit symmetry about multiples axes, which is a result of the CPPN inputs which provide distances to the x, y and z planes and origin in addition to the x, y and z positions. This is advantageous, as it makes the grasping difficulty of the objects less sensitive to changes in orientation.

B. Dataset Comparison

For comparison, we also embed the YCB and Dex-Net 2.0 object datasets into the same search space (Fig. 3), with a quantitative comparison also given in Table III. Due to the small number of objects, the YCB has only very sparse coverage of the object search space, filling 3% of all cells.
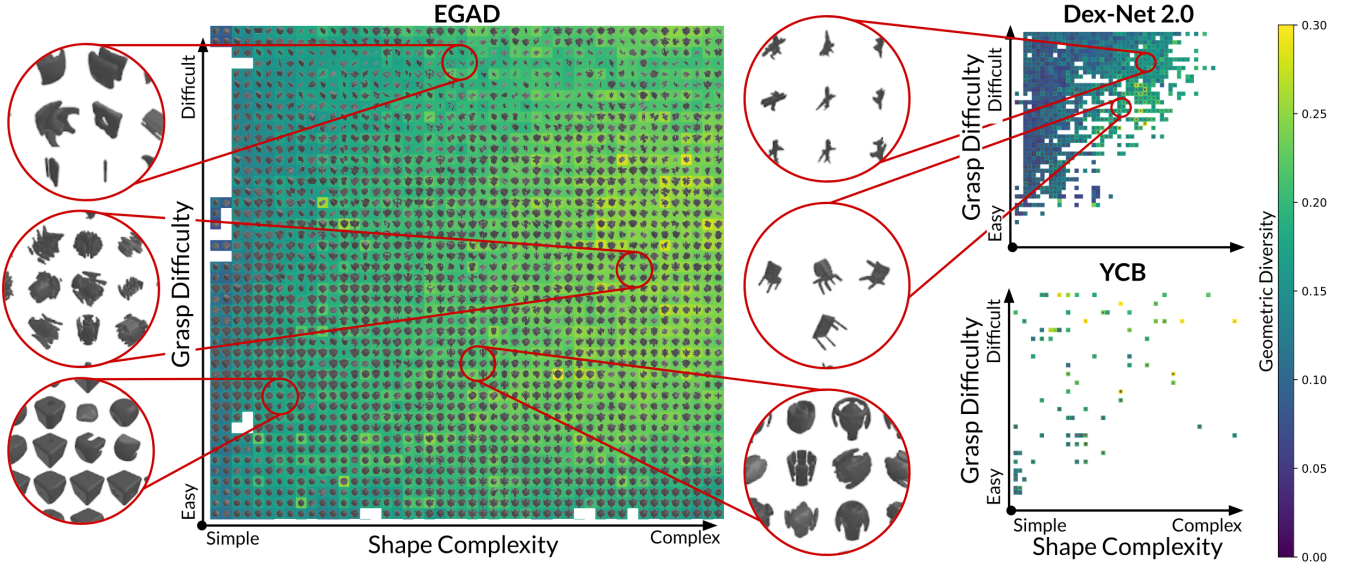
Fig. 3. The distribution and diversity of EGAD in our object search space, compared to the object models found in Dex-Net 2.0 and YCB object sets. NB: For YCB objects, only models with an associated laser scan were used. High resolution versions are available on the project website.

TABLE III
COMPARISON OF EGAD WITH OTHER GRASPING DATASETS

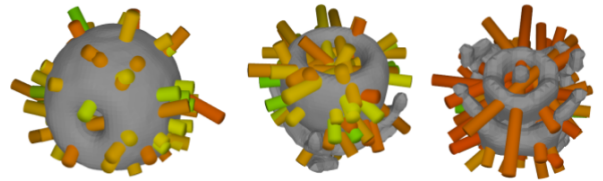| Dataset	Size	Coverage	Diversity		
			Min	Max	Mean
YCB [29]	78	0.03	0.11	0.65	0.19
Dex-Net 2.0 [1]	1497	0.37	0.03	0.26	0.11
EGAD	2331	0.93	0.07	0.27	0.19

Fig. 4. Examples from the Dex-Net compatible database of EGAD objects with pre-computed grasps. Colours indicate the robust grasp quality metric, ranging from red to green.

The small number of objects leads to an overall high object diversity mean of 0.19. However, this is still equivalent to EGAD which contains orders of magnitude more objects.

The Dex-Net 2.0 object dataset is much larger than YCB, but still only covers 37% of the space, and also exhibits much lower geometric diversity than EGAD on average. This is due to the small number of semantic classes of the object models, many of which are geometrically simple shapes (e.g. the container, fruits and foods categories from 3DNet), and similarly, the more complex parts of the space are largely filled by very similar instances of aircraft and chairs.

C. Mesh Dataset

We release EGAD in two formats. The first format is a set of 3D triangular meshes, which can be easily adapted to many robotics tasks such as training grasping algorithms or simulation. Compared to other available mesh datasets (e.g. [31, 32]), EGAD provides a number of benefits, in particular: (1) the dataset has been created specifically with robotic manipulation in mind, so it provides a diverse set of objects which are applicable to the task, rather than meshes representing only a small number of semantic classes; (2) objects are labelled according to their complexity and difficulty, and provide a range in both of these dimensions; (3) the generated meshes are watertight, cohesive (i.e. no missing

faces, inverted normals, etc.) and don't contain unnecessary internal detail, making direct usage in simulation, rendering/visualisation and reality easier (not the case for other datasets [1, 32]); and (4) meshes have been post-processed to allow 3D-printing, making sim-to-real comparisons possible.

To aid adoption of EGAD, the second format leverages the widely used Dex-Net project [1] to create a Dex-Net compatible database of EGAD objects. Each object is pre-labelled with up to 100 antipodal grasps labelled with a robust grasp-quality metric (Fig. 4). The Dex-Net project provides extra functionality, including the ability to label objects with other grasp sampling strategies or quality metrics, add custom grippers, and generate large image datasets for training visual grasp detection algorithms.

In addition, we provide code to enable the creation of other similar, custom datasets of objects that may better fit to specific applications or other gripper types.

D. Evaluation Set

As robotic grasping is an inherently physical problem, we believe that testing on a physical robotic system is the most important step in evaluating such systems. As such, we designate a set of 49 objects from EGAD as an evaluation set, which can be 3D-printed and used to test real-world robotic systems (Fig. 1). 49 objects serves as a practical number for manual robotic testing while also providing a
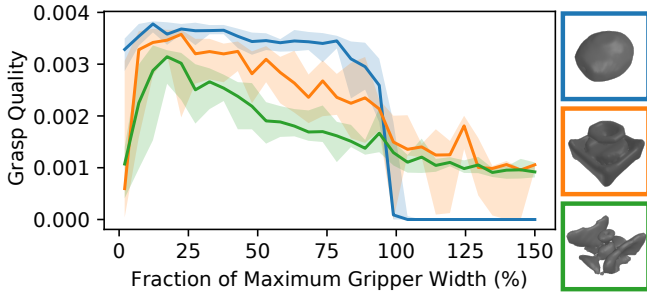
Fig. 5. 75th percentile grasp quality (Section III-A.2) as a function of object size as a fraction of maximum gripper width for three objects, highlighting the importance of object size when reproducing grasping experiments. Shaded area shows 65-85th percentile.

Fig. 6. Average grasp success rate for each object in the evaluation set. Labels correspond to the evaluation objects as shown in Fig. 1. Outer cells show the mean over the respective axes.

good spread of difficulty and complexity. The objects were chosen in an automated manner from the full dataset such that they are uniformly representative of the object space in a 7×7 grid while minimising geometric similarity between objects within the evaluation set. For evaluation, these objects provide a gradient of complexity and difficulty, which can identify the strengths and limitations of visual grasp detection algorithms better than collections of commonly used items.

Objects are labelled alphanumerically according to their position in the 7×7 grid, as per Fig. 1, with A0 being the simplest and easiest object, and G6 being the most complex and difficult.

E. Effect of Object Size

One factor that effects the grasping difficulty of an object is its size relative to the gripper. Fig. 5 illustrates this on three different objects from EGAD, highlighting that the size effects the overall graspability in a nonlinear way depending on the object's geometry. Compared to 3D printing, a downside to datasets comprising commercial products is that grasp performance may be largely effected by the choice of robotic gripper, making isolation and direct comparison of the visual grasp prediction aspect more difficult.

As such, to ensure uniformity in our results and to aid in reproducibility, we use a constant scaling factor throughout all of our experiments, for both dataset generation and in producing the physical evaluation set. Each object is scaled such that its minimum bounding box dimension is no more than 80% of the gripper width. Along with our dataset, we also release a Python script for re-scaling the provided meshes by this metric for a given gripper width to allow fair results comparisons.

V. ROBOTIC EXPERIMENTS

In the following sections we use the EGAD evaluation set to evaluate a state-of-the-art visual grasp detection network. In doing so we provide a template for presenting results. The diversity of the evaluation set allows us to gain insights into the strengths and limitations of the grasp detection approach, providing future research directions to improve upon these baseline results.

A. Visual Grasp Detection

We use the Generative Grasping Convolutional Neural Network (GG-CNN) from [15, 17] as a visual grasp detection algorithm. GG-CNN is a fully convolutional network that provides a one-to-one mapping from an input depth image to a prediction of grasp quality and pose at every pixel in realtime. Each pixel in the output defines a top-down grasp pose, defined by $\mathbf{g} = (\mathbf{c}, \phi, w, q)$, where $\mathbf{c} = (x, y, z)$ is the position of the grasp's centre, ϕ is a rotation around the vertical axis, w is the desired width of the gripper and q represents a grasp quality. GG-CNN was trained on the Cornell Grasping Dataset [34].

B. Equipment

For robotic grasping experiments we use a Franka Emika Panda robot, fitted with 3D-printed fingers with silicone tips based on [51]. The maximum opening of the fingers is 75mm, and the evaluation objects were scaled according to this. An Intel Realsense D435 depth camera is attached to the end effector of the robot to provide visual input. Videos of the experiments are available on the project webpage.

C. Procedure

We perform 20 grasp attempts on each of the 49 3D-printed evaluation objects in isolation, for a total of 980 grasp attempts. Objects were placed one at a time into the workspace of the robot. For each grasp attempt, a depth image was captured from a fixed viewpoint and the best predicted grasp from GG-CNN was executed by the robot. The grasp success rate is the fraction of grasps after which the robot successfully lifted and held the object 40cm above the table. If the grasp was successful, the object was dropped by the robot into the workspace in order to randomise the pose and position for the next grasp attempt. Objects that fell outside of the workspace were replaced.

D. Results and Discussion

Fig. 6 shows the grasp success rate for each of the 49 3D-printed evaluation objects individually. The right-hand
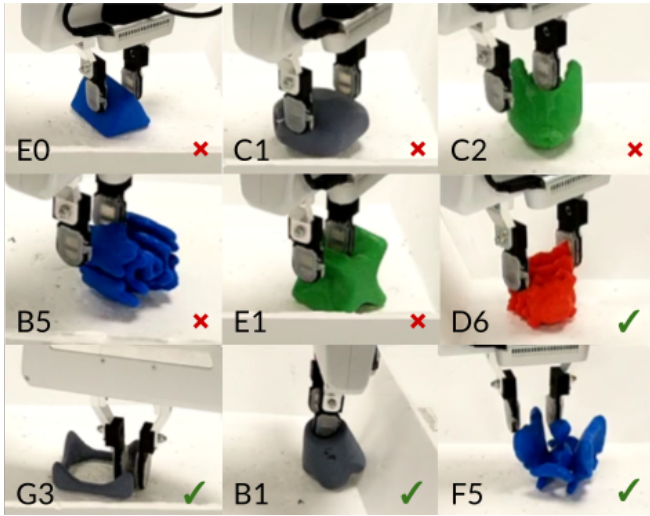
Fig. 7. Examples chosen from experimental grasp attempts. × indicates grasp failure and ✓ indicates success. Refer to Section V-D for details.

column shows that the grasp success rate decreased from 69% for the easiest 7 objects (A row), to 40% for the 7 most difficult (G row). The overall success rate across all objects was 58%. While this is lower than previously reported results for GG-CNN [15–17], this is expected since the EGAD evaluation objects are more complex and difficult than those previously used.

Despite the trend in difficulty, there are some clear outliers. For example, objects C1 and D1 performed much worse than their neighbours, and objects E5, F4, F5 and G3 that performed much better despite being difficult objects. Using the EGAD evaluation objects in this way allows us to analyse these results in a more introspective way than reporting a single success rate on a non-diverse object set. In the rest of this section we discuss some identified strengths and limitations of GG-CNN with regards to these results, and propose future improvements.

Grasp Depth A number of simpler and easier objects, in particular E0, C1 (pictured in Fig. 7) and D1, have very low grasp success rates. This highlights a major failure case for GG-CNN, where the depth of the grasp is computed relative to the grasp's centre, resulting in grasps that are too shallow resulting in failure. Round objects in the range A0 to B2 were also effected in the same way. A major improvement to GG-CNN may be to also encode the required depth of the grasp in the input or prediction.

Orientation Bias The grasps generated by GG-CNN are often oriented perpendicular to an object's major axis. For example, on object E0 this results in a high number of failures due to the sloping sides (Fig. 7). Meanwhile, grasping the object lengthways would provide a stable grasp on the object's flat ends. This bias is likely caused by the lack of similar examples in the training data, in which grasps on long objects are heavily biased towards perpendicular grasps, and could be corrected with improved training data.

Top-Down Grasping Like many other visual grasp detection algorithms, GG-CNN is limited to producing top-down

(4-DoF) grasps. However, for objects such as C2, there are limited ways to grasp the object from a top-down orientation (Fig. 7). This provides a strong motivation for using 6-DoF grasps that align to the graspable parts of objects.

Grasping Object Parts Unlike many other visual grasp detection algorithms, GG-CNN predicts the gripper width for each grasp. This is advantageous for objects such as B1 (in certain orientations), G3 and F5 (Fig. 7), where precise, narrow grasps are required to avoid collision with object parts, resulting in a higher than average success rate despite these objects' complexity and difficulty. On the other hand, a number of failures were noted for objects such as B5 and E1, where depressions in the objects cause grasps that result in collisions and grasp failures.

Finger Material In addition to visual aspects, physical properties of the gripper also effect the results. Objects such as E5 and D6 (Fig. 7) are largely difficult due to their uneven surface with many acute protrusions, making grasps performed with a rigid gripper surface unstable. However, the compliant nature of the silicone fingertips used, as explored by [51] largely accounts for this by moulding to the surface resulting in stable grasps and a high success rate for these objects.

VI. CONCLUSION

We presented EGAD, a dataset of over 2000 evolved 3D objects for training and evaluating robotic grasping and manipulation. The objects uniformly fill a space of shape complexity and grasp difficulty, compared to other similar datasets which are limited in both size and diversity. This provides the necessary diversity for training robust visual grasp detection algorithms. Additionally, we specify a diverse evaluation set of 49 objects which are 3D-printable to allow for reproducible testing of grasping algorithms over a wide range of complexity and difficulty.

Using the EGAD evaluation set, we were able to identify a number of limitations of a state-of-the-art grasping algorithm GG-CNN, which has previously not been possible on simpler sets of "household" objects. In future work we propose to use these insights to improve on the baseline results, and to investigate the effect of diverse training data on the robustness of visual grasp detection algorithms.

REFERENCES

[1] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics," in *Robotics: Science and Systems (RSS)*, 2017.

[2] I. Lenz, H. Lee, and A. Saxena, "Deep Learning for Detecting Robotic Grasps," *The International Journal of Robotics Research (IJRR)*, 2015.

[3] J. Redmon and A. Angelova, "Real-Time Grasp Detection Using Convolutional Neural Networks," in *International Conference on Robotics and Automation (ICRA)*, 2015.

[4] L. Pinto and A. Gupta, "Supersizing Self-supervision: Learning to Grasp from 50K Tries and 700 Robot Hours," in *International Conference on Robotics and Automation (ICRA)*, 2016.

[5] E. Johns, S. Leutenegger, and A. J. Davison, "Deep Learning a Grasp Function for Grasping under Gripper Pose Uncertainty," in *International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[6] Z. Wang, Z. Li, B. Wang, and H. Liu, "Robot grasp detection using multimodal deep convolutional neural networks," *Advances in Mechanical Engineering*, 2016.

[7] A. ten Pas, M. Gualtieri, K. Saenko, and R. Platt, "Grasp Pose Detection in Point Clouds," *The International Journal of Robotics Research (IJRR)*, 2017.

[8] J. Mahler and K. Goldberg, "Learning Deep Policies for Robot Bin Picking by Simulating Robust Grasping Sequences," in *Conference on Robot Learning (CoRL)*, 2017.

[9] S. Kumra and C. Kanan, "Robotic Grasp Detection using Deep Convolutional Neural Networks," in *International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[10] U. Viereck, A. Pas, K. Saenko, and R. Platt, "Learning a visuomotor controller for real world robotic grasping using simulated depth images," in *Conference on Robot Learning (CoRL)*, 2017.

[11] J. Tobin, L. Biewald, R. Duan, M. Andrychowicz, A. Handa, V. Kumar, B. McGrew, A. Ray, J. Schneider, P. Welinder, *et al.*, "Domain randomization and generative models for robotic grasping," in *RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[12] X. Zhou, X. Lan, H. Zhang, Z. Tian, Y. Zhang, and N. Zheng, "Fully convolutional grasp detection network with oriented anchor box," in *International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[13] A. Depierre, E. Dellandréa, and L. Chen, "Jacquard: A large scale dataset for robotic grasp detection," in *International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[14] F.-J. Chu, R. Xu, and P. A. Vela, "Real-world multiobject, multigrasp detection," *IEEE Robotics and Automation Letters*, 2018.

[15] D. Morrison, P. Corke, and J. Leitner, "Closing the Loop for Robotic Grasping: A Real-time, Generative Grasp Synthesis Approach," in *Robotics: Science and Systems (RSS)*, 2018.

[16] D. Morrison, P. Corke, and J. Leitner, "Multi-view picking: Next-best-view reaching for improved grasping in clutter," in *International Conference on Robotics and Automation (ICRA)*, 2019.

[17] D. Morrison, P. Corke, and J. Leitner, "Learning robust, real-time, reactive robotic grasping," *The International Journal of Robotics Research*, 2019.

[18] J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, and K. Goldberg, "Learning ambidextrous robot grasping policies," *Science Robotics*, 2019.

[19] V. Satish, J. Mahler, and K. Goldberg, "On-policy dataset synthesis for learning robot grasping policies using fully convolutional deep networks," *IEEE Robotics and Automation Letters*, 2019.

[20] U. Asif, J. Tang, and S. Harrer, "Densely Supervised Grasp Detector (DSGD)," in *AAAI Conference on Artificial Intelligence*, 2019.

[21] Y. Yu, Z. Cao, S. Liang, Z. Liu, J. Yu, and X. Chen, "A Grasping CNN with Image Segmentation for Mobile Manipulating Robot," in *International Conference on Robotics and Biomimetics (ROBIO)*, 2019.

[22] H. Liang, X. Ma, S. Li, M. Görner, S. Tang, B. Fang, F. Sun, and J. Zhang, "PointnetGPD: Detecting grasp configurations from point sets," in *International Conference on Robotics and Automation (ICRA)*, 2019.

[23] B. Eysenbach, A. Gupta, J. Ibarz, and S. Levine, "Diversity is all you need: Learning skills without a reward function," *ArXiv preprint arXiv:1802.06070*, 2018.

[24] A. Habib, C. Karmakar, and J. Yearwood, "Impact of ECG dataset diversity on generalization of CNN model for detecting QRS complex," *IEEE Access*, 2019.

[25] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *International Conference on Computer Vision (ICCV)*, 2017.

[26] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition (CVPR)*, 2009.

[27] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *European conference on computer vision*, 2014.

[28] J. Mahler, R. Platt, A. Rodriguez, M. Ciocarlie, A. Dollar, R. Detry, M. A. Roa, H. Yanco, A. Norton, J. Falco, *et al.*, "Guest editorial open discussion of robot grasping benchmarks, protocols, and metrics," *Transactions on Automation Science and Engineering*, 2018.

[29] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The YCB object and model set: Towards common benchmarks for manipulation research," in *International Conference on Advanced Robotics (ICAR)*, 2015.

[30] C. Goldfeder, M. Ciocarlie, H. Dang, and P. K. Allen, "The columbia grasp database," in *International Conference on Robotics and Automation (ICRA)*, 2009.

[31] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The princeton shape benchmark," in *Shape Modeling Applications*, 2004.

[32] W. Wohlkinger, A. Aldoma, R. B. Rusu, and M. Vincze, "3DNet: Large-scale object class recognition from cad models," in *International Conference on Robotics and Automation (ICRA)*, 2012.

[33] 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*, 2012.

[34] Y. Jiang, S. Moseson, and A. Saxena, "Efficient grasping from rgbd images: Learning using a new rectangle representation," in *International conference on robotics and automation*, 2011.

[35] J. Leitner, A. W. Tow, N. Sünderhauf, J. E. Dean, J. W. Durham, M. Cooper, M. Eich, C. Lehnert, R. Mangels, C. McCool, *et al.*, "The ACRV picking benchmark: A robotic shelf picking benchmark to foster reproducible research," in *International Conference on Robotics and Automation (ICRA)*, 2017.

[36] J. Lehman, S. Risi, and J. Clune, "Creative generation of 3D objects with deep learning and innovation engines," in *International Conference on Computational Creativity*, 2016.

[37] R. J. Preen and L. Bull, "Towards the evolution of vertical-axis wind turbines using supershapes," *Evolutionary Intelligence*, 2014.

[38] J. Collins, W. Geles, D. Howard, and F. Maire, "Towards the targeted environment-specific evolution of robot components," in *Genetic and Evolutionary Computation Conference*, 2018.

[39] J.-B. Mouret and J. Clune, "Illuminating search spaces by mapping elites," *ArXiv preprint arXiv:1504.04909*, 2015.

[40] K. O. Stanley, "Compositional pattern producing networks: A novel abstraction of development," *Genetic programming and evolvable machines*, 2007.

[41] J. Clune and H. Lipson, "Evolving 3D objects with a generative encoding inspired by developmental biology," *ACM SIGEVOlution*, 2011.

[42] J. E. Auerbach and J. C. Bongard, "Environmental influence on the evolution of morphological complexity in machines," *PLoS computational biology*, 2014.

[43] D. L. Page, A. F. Koschan, S. R. Sukumar, B. Roui-Abidi, and M. A. Abidi, "Shape analysis algorithm based on information theory," in *International Conference on Image Processing*, 2003.

[44] S. R. Sukumar, D. L. Page, A. F. Koschan, and M. A. Abidi, "Towards understanding what makes 3d objects appear simple or complex," in *Conference on Computer Vision and Pattern Recognition Workshops*, 2008.

[45] D. Wang, D. Tseng, P. Li, Y. Jiang, M. Guo, M. Danielczuk, J. Mahler, J. Ichnowski, and K. Goldberg, "Adversarial grasp objects," in *International Conference on Automation Science and Engineering (CASE)*, 2019.

[46] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii, "Topology matching for fully automatic similarity estimation of 3D shapes," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001.

[47] D. Bespalov, W. C. Regli, and A. Shokoufandeh, "Reeb graph based shape retrieval for CAD," in *International Design Engineering Technical Conferences and Computers and Information in Engineering (IDEC-CIE)*, 2003.

[48] J. Gomes, P. Mariano, and A. L. Christensen, "Devising effective novelty search algorithms: A comprehensive empirical study," in *Annual Conference on Genetic and Evolutionary Computation*, 2015.

[49] T. Lewiner, H. Lopes, A. W. Vieira, and G. Tavares, "Efficient implementation of marching cubes' cases with topological guarantees," *Journal of graphics tools*, 2003.

[50] K. O. Stanley and R. Miikkulainen, "Evolving neural networks through augmenting topologies," *Evolutionary computation*, 2002.

[51] M. Guo, D. V. Gealy, J. Liang, J. Mahler, A. Goncalves, S. McKinley, J. A. Ojea, and K. Goldberg, "Design of Parallel-Jaw Gripper Tip Surfaces for Robust Grasping," in *International Conference on Robotics and Automation (ICRA)*, 2017.