Learning Grasp Ability Enhancement through Deep Shape Generation

Junnan Jiang\textsuperscript{1}, Xiaohui Xiao\textsuperscript{2}, Fei Chen\textsuperscript{3}, and Miao Li\textsuperscript{1,2}

\textsuperscript{1} The Institute of Technological Sciences, Wuhan University, Wuhan, China. \texttt{miao.li@whu.edu.cn}

\textsuperscript{2} The School of Power and Mechanical Engineering, Wuhan University, Wuhan, China.

\textsuperscript{3} Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, China \texttt{feichen@cuhk.edu.hk}

Abstract. Data-driven especially deep learning-based approaches have become a dominant paradigm for robotic grasp planning during the past decade \cite{2}. However, the performance of these methods is greatly influenced by the quality of the training dataset available. In this paper, we propose a framework to generate object shapes to augment the grasping dataset and thus can improve the grasp ability of a pre-designed deep neural network. First, the object shapes are embedded into a low dimensional feature space using an encoder-decoder structure network. Then, the rarity and graspness scores are computed for each object shape using outlier detection and grasp quality criteria. Finally, new objects are generated in feature space leveraging the original high rarity and graspness score objects’ feature. Experimental results show that the grasp ability of a deep-learning-based grasp planning network can be effectively improved with the generated object shapes.

Keywords: Data Augmentation · Grasp Planning

1 Introduction

Despite significant progress, it is still a difficult task to plan a grasp for unknown object and robot \cite{215}. During the past decade, data-driven especially deep learning-based approaches have demonstrated significant superiority compared with traditional model-based approaches \cite{8}. However, the performance of the deep learning-based approaches is greatly limited by the quality of the available training dataset.

During the past few years, a large number of grasping datasets have been proposed \cite{5,7,9,12,17,21}, which can be generally categorized into two classes: CAD (Computer Aided Design) model based datasets \cite{5,17} and sensory model based datasets \cite{9,21}. These datasets play an important role in training and qualifying this work was supported by Suzhou Key Industry Technology Innovation Project under the grant agreement number SYG202121. (Corresponding authors: Fei Chen, Miao Li.) Also thanks Yasemin Bekiroglu for her valuable advice.
the performance of different grasp planning algorithms. However, given a new object, a trained grasp planning system can still lead to a bad grasp. Most grasp planning frameworks will normally add these “failed objects” into the training dataset and train the model again. However, this trial and error process can be tedious, non-systematic and expensive.

To address this issue, in this paper we propose a learning-based approach to encapsulate object shape information into a low dimensional feature space. In this feature space, the shapes in the original dataset can be better encoded, interpolated and generated. Moreover, in this low dimensional feature space, two grasp-related metrics are proposed to generate new shapes that can further enhance the grasp ability of the original dataset.

**Fig. 1.** Original and augmented data distribution comparison. We use t-SNE to project the shapes’ feature vectors to a 2D plane, the Euclidean distances between scattered points represent their feature similarity, and the color represents their graspness scores. We generate new data that leveraging the feature of original high rarity and graspness shapes so as to improve the dataset quality.

The main contributions of this paper are two folds:

- An encoder-decoder framework is proposed to map a voxelized shape into a low dimensional feature space.
- A data augmentation method using two new metrics is proposed to enhance the grasp ability

2 Related Work

2.1 Grasping Dataset

With the great success achieved in the data-driven grasp planning methods, many grasping datasets have been proposed (Table 1). Among these datasets,
YCB [5] is one of the first datasets that provided high-resolution 3D scans of everyday objects. Later, datasets containing more shapes, more annotated grasps and more sensory information have been proposed one after another. Due to subjective or random shape selection when making these datasets, although existing grasping algorithms can perform well on one dataset, it is still difficult to grasp a new unseen object. To solve this problem, EGAD dataset [18] hopes to scientifically include richer information on shape geometry and grasp complexity. However, since the shapes in EGAD are heuristically generated and do not utilize the existing real-world shapes features, they can only be used for evaluation but is difficult to be applied in the real application.

Table 1. Comparison of publicly available grasping datasets.

| Dataset       | Data Type | Objects | Annotate Method | Grasps |
|---------------|-----------|---------|-----------------|--------|
| Cornell [12]  | RGB       | 240     | hand            | 8k     |
| Google Brain [14] | RGB     | N/A*    | real-robot trials | 650k   |
| YCB [5]       | CAD       | 97      | N/A**           | 233k   |
| EGAD [18]     | CAD       | 2331    | analytical      | 6.7M   |
| DexNet [17]   | CAD       | 1500    | analytical      |        |
| 6DOF-GraspNet [20] | CAD     | 206     | simulator       | 7.07M  |
| GraspNet [7]  | CAD RGBD  | 88      | analytical      | 1.1B   |
| THU Dataset [11] | RGBD Tactile | 17   | real-robot trials | 1.7k |
| Object Folder [9] | CAD Tactile Audio | 1000 | N/A**           | N/A**  |

* around one hundred everyday objects, not provided.
** just single CAD files but no annotations or grasps provided.

2.2 Data Augmentation

Data augmentation [6,22] is a very popular approach in the learning-based methods because it effectively improves dataset quality and reduces network overfitting problems. Handcrafted methods have been widely used in computer vision, such as shifting, scaling and rotating. Similarly, grasping data augmentation by randomly rotating and cropping images [19], or randomly combining different shapes [23] do show some effect. However, except for requiring expert knowledge, these methods can only generate data in instance space, which means that just simply sense the original data multiple times with the original sensor (e.g., camera), but is difficult to combine the feature of grasping into generated data. With the development of generation methods such as AutoEncoder [3] and GAN [10,26], shape features can be better encoded and new shapes can be generated more flexibly. Wang [24] generates adversarial grasp objects which leverage the feature of the difficult-to-grasp object to generate new data, and generated data are difficult to grasp by the grasp planning algorithm. This gives a great motivation for generating new data to expand dataset with original objects’ shape and grasp feature.
3 Object Shape Encoding

3.1 Network Architecture

In order to generate new shapes leveraging the feature of the original shapes, we propose an AutoEncoder-Critic network shown in Fig. 2. We use voxel to represent a shape, which maps a shape to a $64 \times 64 \times 64$ binary matrix. The AutoEncoder of AE-Critic contains an Encoder $E$ and a Decoder $D$. The Encoder maps a voxel $x$ to a 128-dimension feature vector $z$, containing 5 3D convolution layers and 2 fully connected layers. The convolution layers use $4 \times 4 \times 4$ kernel size and 2 strides, with batch normalization and ReLU layers added in between, mapping the voxel to a $512 \times 4 \times 4 \times 4$ size feature map. The fully connected layers have 32768 and 128 neurons separately, map the feature map to a 128-dim feature vector. The Decoder mirrors the Encoder, maps a 128-dim feature vector to a $64 \times 64 \times 64$ reconstructed voxel $\hat{x}$, and contains 2 fully connected layers and 5 3D transposed convolution layers, the layers configurations are the same as Encoder. Using AutoEncoder, we can generate new shapes leveraging original shapes’ feature by changing their feature vectors like interpolation, by changing the interpolation pairs and weights, we can theoretically generate infinite shapes.

In addition, to make the objects generated by interpolation more realistic, we add a Critic $C$, which aims to estimate the interpolated weight $\alpha$ corresponding to an interpolated shape, to regularize the training process of AutoEncoder [1]. The AE is trained to generate shapes to fool the Critic output lower interpolate weights, which means that the Critic is more willing to think the interpolated input is non-interpolated original shape. And thus, more realistic shapes similar to the original shapes can be generated. The structure of Critic is similar to Encoder, only with one more fully connected layer to map the 128-dim feature vector to a 1-dim interpolated weight.

3.2 Network Training

The Critic is trained to minimize the loss function (1), its first term trains the Critic to recover interpolated weight $\alpha$ from reconstructed voxel $\hat{x}_\alpha$, and its second term ensures the Critic’s training process to be more stable.

$$L_C = ||C(\hat{x}_\alpha) - \alpha||^2 + ||C(\gamma x + (1 - \gamma)D(E(x)))||^2$$ (1)

The AutoEncoder is trained to minimize the loss function (2), its first term trains the AutoEncoder to generate voxel $\hat{x}$ similar to original voxel $x$ and its second term is to encourage the generated voxel by AutoEncoder to fool the Critic so as to generate more realistic voxel. Here $L_B$ represents binary cross-entropy loss.

$$L_{(E,D)} = L_B(x , D(E(x))) + \lambda ||C(\hat{x}_\alpha)||^2$$ (2)

Finally, we can map a voxel to a 128-dim feature vector. To better visualize the distribution of feature vectors, we use t-SNE [10] to project feature vectors to a 2D plane. The distribution of shapes can be seen in Fig. 4 which shows that similar objects are closer together, indicating that the feature vectors extracted by AE-Critic can be used for clustering or similarity measurement.
Fig. 2. AE-Critic network architecture. The Critic tries to estimate the interpolated weight $\alpha$ corresponding to an interpolated shape and thus can regularize the AutoEncoder (Encoder+Decoder) to generate more realistic interpolated shapes by fooling the Critic output lower interpolated weight.

4 Data Augmentation

With the AE-Critic network, shapes can be easily generated to expand the dataset, but what kinds of shapes are not sufficient and need to be generated? To let the network learn more diverse data, unseen or rare shapes are definitely what we want. And based on the property of grasping, when random sample grasps pose on an object [17], successful grasps are far less than failed grasps. This lead to the network can not learn many successful grasps, so shapes containing more successful grasps are also what we want to generate. Thus, we define shape rarity and graspness metrics, and new shapes leveraging the feature of original high rarity and graspness shapes are generated to augment the dataset.

4.1 Shape Rarity and Graspness Metrics

**Shape Rarity Metric:** We assume that rare shapes are those whose features are distinct from others. Thus, we use outlier detection [4] to evaluate each feature vector, for one shape which is rarer, the score of outlier detection is higher. In detail, we use L2 distance to measure two feature vectors’ similarity. For one shape $O$, we select k-nearest shapes $\hat{N}_k(O)$, and the local reachability density(lrd) can be defined by equation (3), which is the reciprocal of the mean of distances between feature vectors from the shape $O$ to its $k$ neighbors.

$$lrd(O) = \frac{1}{k} \sum_{P \in N_k(O)} \text{dist}(O, P)$$  \hspace{1cm} (3)
Then the score of outlier detection defined as local outlier factor (LOF) can be calculated by equation (4):

\[
\text{LOF}(O) = \frac{1}{k} \sum_{P \in N_k(O)} \frac{\text{Ird}(P)}{\text{Ird}(O)}
\]  

(4)

**Shape Graspness Metric:** We define a shape’s graspness score which represents the level of difficulty to find a stable grasp for an object. Firstly, using the Dex-Net analytical grasp planner [17], a number of antipodal grasps on a shape’s surface can be sampled. Then, we use robust Ferrari-Canny [8] to compute each grasp quality \( Q \):

\[
Q = \min_w LQ(w)
\]  

(5)

where \( LQ \) is local quality metric that measures how efficiently a given wrench \( w \) can resist disturbances given applied forces \( f \) and the approximated friction cone \( FC \):

\[
LQ(w) = \max_f \frac{||w||}{||f||} \text{ s.t. } f \in FC
\]  

(6)

Finally, we use a threshold of 0.002 [13] to distinguish successful and failed grasps, and define the proportion of successful grasps of an object as its graspness score. The lower the graspness score, the harder it is for the object to grasp. The Fig. 3 is the histogram of graspness scores for all objects in the 3dnet dataset [25], the lack of high graspness score objects verifies that they are insufficient and needed to be generated.

![Histogram of Graspness Score](image)

**Fig. 3.** The graspness score histogram of all objects in the 3dnet. Three objects with different graspness scores are displayed above, each cylinder represents an antipodal grasp, and the color of the cylinder indicates the grasp quality.
4.2 Shape Generation

Based on AE-Critic network and two defined metrics, we can finally generate shapes leveraging the feature of insufficient data so as to augment the dataset. The overall pipeline is shown in Fig. 4.

After calculating the rarity and graspness scores of all objects, we only select each two objects whose scores are higher than $t\%$ of all scores in each metric as generation pair. Then, to avoid the feature vectors of shapes in generation pairs being too close cause shapes duplicated or too far cause the intermediate properties disappear, we group the nearest $N$-th to $(N + K)$-th neighbors into generation pairs. With these generation pairs, we finally linear interpolate with interpolated weight $\alpha$ in each generation pair, and decode the mix feature vector to a new generated shape.

Up to now, we leverage the feature of high rarity and graspness score shapes to generate new shapes and get a higher quality grasping dataset. The generated number of augmented shapes depends on the parameters $t$, $N$, $K$ and $\alpha$. 

![Fig. 4. The whole pipeline of shape generation for grasping dataset augmentation. Original shapes’ rarity and graspness scores are computed through outlier detection and grasp quality criteria. Then every two high-score nearby objects are grouped as a generation pair. Finally, the AE-Critic network is used for shape generation through interpolation between two shapes’ feature vectors.](image)
5 Experiments

The goal of our experiment is to answer the following questions:

- Can Critic regularization help the AutoEncoder generate more realistic shapes?
- What detailed improvement can generated data bring to the network?
- What is the optimal ratio of generated and original data, and what are the augment method limitations?

5.1 Experiment Setup

All our experiments are based on the 3dnet dataset [25] and GQ-CNN [17] grasp planning algorithm. We first randomly select 1000 shapes as the training dataset and 363 shapes as the test dataset. Considering the network is needed to classify the uneven distribution of successful and failed grasps, we use the AP (Average Precision) score on the test dataset to measure the performance of a network.

5.2 Different Network Comparison

We train AE and AE-Critic networks on 3dnet using the Adam optimizer. After training AE with a learning rate of 0.001, we use the parameters of AE as the pre-trained network for AE-Critic, and train AutoEncoder and Critic of AE-Critic with learning rates of 0.0001 and 0.001 respectively.

![Fig. 5. Example of interpolation data from 3dnet [25], produced by AE and AE-Critic network. AE-Critic generates significantly fewer scattered points and the generated shape is more complete. While the shapes generated by AE are just like a simple addition of two objects.](image)
Fig. 5 is the shapes we generated using two networks with 0.1 interval interpolation. The result shows that the shape generated by AE will contain more scattered points, while the shapes generated by AE-Critic are more complete. In our understanding, AE tends to simply remember the voxel distribution corresponding to feature vectors, while AE-Critic tends to learn higher-level semantic information and decode a feature vector into a single object shape.

5.3 Improvement from Generated Data

To see the detailed improvement from generated data, we first randomly select 200 shapes from the training dataset and augment it with 190 and 219 new generated shapes leveraging the feature of the original high rarity or graspness score shapes separately. Then we train GQ-CNN [17] on the original and augmented dataset, and compute their AP score on the test dataset. Finally, we compare the number distribution of selected high-scoring data and generated data on rarity or graspness scores, and the AP score improvement of each object with different scores on the test dataset. The result is shown in Fig. 6, all three kinds of data are normalized for better visualization. It tells that more selected data will result in more generated data in the same score, which means that the generated data have the same property as the original data to a certain extent, and the generated data at the same time will lead to a greater AP score improvement.

![Fig. 6. The distribution between selected high-scoring data, generated data and AP improvement on test dataset with different rarity or graspness scores.](image)

5.4 Augmentation Ratio and limitation

To find the optimal augmentation ratio and augmentation limitation, we randomly select 50, 100, and 200 shapes from 1000 training datasets, and generate
new shapes with augmentation ratios 1:0, 1:0.5, 1:1, 1:1.5, 1:2. This means that for 50 original shapes, 0, 25, 50, 75, and 100 generated shapes similar to both high rarity and graspness scores shapes are used for augmentation, the same as 100 and 200 shapes. Then the whole 15 augmented datasets are used for GQ-CNN training, all the 15 AP scores on the test dataset are shown in Fig. 7.

![Average Precision on Different Dataset](image)

**Fig. 7.** GQCNN is trained on 50, 100, 200 3dnet data with augmentation ratios of 1:0, 1:0.5, 1:1, 1:1.5, 1:2, and tested on the same test dataset. The results show that the larger dataset, the higher the ratio of augmented data is allowed.

Although the results show that our augmentation methods can improve the accuracy of the network on the test dataset, the augmentation ratios allowed by different amounts of data are different. The 50, 100, and 200 data achieve the highest AP values at 1:0.5, 1:1.5, and 1:2 respectively. This means that blindly increasing generated data will lead the network overfitting to the generated data and cause a bad performance on the original dataset.

### 6 Discussion and Conclusion

#### 6.1 Discussion

Although the data we generate can play the role of the original object to a certain extent and help the network achieve higher accuracy on the unseen data, there are still some limitations in this paper. First, shape rarity and graspness metrics
are not unique. Not considered properties, such as object size or multi-fingers grasp quality evaluation, will also affect the metric score to a certain extent. In addition, in shape generation, the shape representation of voxel, generation pairs selection method and interpolated weight may not optimal. Perhaps new generation methods can enable generated objects to better utilize and extend the characteristic of original data.

6.2 Conclusion

In this paper, we present a general pipeline for grasping dataset augmentation. Objects are encoded into feature vectors using the AutoEncoder-Critic network, and generated objects, leveraging the feature of original high rarity and graspness score objects, are used to augment the original grasping dataset. Experiments show our generated data can play the role of original data to a certain extent, improving the network performance on the test dataset. In the future work, we hope to investigate in more detail about what kind of data can best enhance the robustness of grasp planning algorithms and try new generation methods to reduce the gap between original and generated data.

References

1. Berthelot, D., Raffel, C., Roy, A., Goodfellow, I.: Understanding and improving interpolation in autoencoders via an adversarial regularizer. arXiv preprint arXiv:1807.07543 (2018)
2. Bohg, J., Morales, A., Asfour, T., Kragic, D.: Data-driven grasp synthesis—a survey. IEEE Transactions on Robotics 30(2), 289–309 (2013)
3. Bourlard, H., Kamp, Y.: Auto-association by multilayer perceptrons and singular value decomposition. Biological cybernetics 59(4), 291–294 (1988)
4. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: Lof: identifying density-based local outliers. In: Proceedings of the 2000 ACM SIGMOD international conference on Management of data. pp. 93–104 (2000)
5. Calli, B., Singh, A., Walsman, A., Srinivasa, S., Abbeel, P., Dollar, A.M.: The ycb object and model set: Towards common benchmarks for manipulation research. In: 2015 international conference on advanced robotics (ICAR). pp. 510–517. IEEE (2015)
6. Cubuk, E.D., Zoph, B., Mane, D., Vasudevan, V., Le, Q.V.: Autoaugment: Learning augmentation strategies from data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 113–123 (2019)
7. Fang, H.S., Wang, C., Gou, M., Lu, C.: Graspnet-1billion: A large-scale benchmark for general object grasping. In: Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. pp. 11444–11453 (2020)
8. Ferrari, C., Canny, J.F.: Planning optimal grasps. In: ICRA. vol. 3, p. 6 (1992)
9. Gao, R., Si, Z., Chang, Y.Y., Clarke, S., Bohg, J., Fei-Fei, L., Yuan, W., Wu, J.: Objectfolder 2.0: A multisensory object dataset for sim2real transfer. arXiv preprint arXiv:2204.02389 (2022)
10. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. Advances in neural information processing systems 27 (2014)
11. Guo, D., Sun, F., Fang, B., Yang, C., Xi, N.: Robotic grasping using visual and tactile sensing. Information Sciences 417, 274–286 (2017)
12. Jiang, Y., Moseson, S., Saxena, A.: Efficient grasping from rgbd images: Learning using a new rectangle representation. In: 2011 IEEE International conference on robotics and automation. pp. 3304–3311. IEEE (2011)
13. Kappler, D., Bohg, J., Schaal, S.: Leveraging big data for grasp planning. In: 2015 IEEE International Conference on Robotics and Automation (ICRA). pp. 4304–4311. IEEE (2015)
14. Levine, S., Pastor, P., Krizhevsky, A., Ibarz, J., Quillen, D.: Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. The International journal of robotics research 37(4-5), 421–436 (2018)
15. Li, M., Hang, K., Kragic, D., Billard, A.: Dexterous grasping under shape uncertainty. Robotics and Autonomous Systems 75, 352–364 (2016)
16. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research 9(11) (2008)
17. Mahler, J., Liang, J., Niyaz, S., Laskey, M., Doan, R., Liu, X., Ojea, J.A., Goldberg, K.: Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. arXiv preprint arXiv:1703.09312 (2017)
18. Morrison, D., Corke, P., Leitner, J.: Egad! an evolved grasping analysis dataset for diversity and reproducibility in robotic manipulation. IEEE Robotics and Automation Letters 5(3), 4368–4375 (2020)
19. Morrison, D., Corke, P., Leitner, J.: Learning robust, real-time, reactive robotic grasping. The International journal of robotics research 39(2-3), 183–201 (2020)
20. Mousavian, A., Eppner, C., Fox, D.: 6-dof graspnet: Variational grasp generation for object manipulation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2901–2910 (2019)
21. Narang, Y.S., Sundaralingam, B., Van Wyk, K., Mousavian, A., Fox, D.: Interpreting and predicting tactile signals for the syntouch biotac. The International Journal of Robotics Research 40(12-14), 1467–1487 (2021)
22. Shorten, C., Khoshgoftaar, T.M.: A survey on image data augmentation for deep learning. Journal of big data 6(1), 1–48 (2019)
23. Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., Abbeel, P.: Domain randomization for transferring deep neural networks from simulation to the real world. In: 2017 IEEE/RSJ international conference on intelligent robots and systems (IROS). pp. 23–30. IEEE (2017)
24. Wang, D., Tseng, D., Li, P., Jiang, Y., Guo, M., Danielczuk, M., Mahler, J., Ichnowski, J., Goldberg, K.: Adversarial grasp objects. In: 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE). pp. 241–248. IEEE (2019)
25. Wohlkinger, W., Aldoma, A., Rusu, R.B., Vincze, M.: 3d-net: Large-scale object class recognition from cad models. In: 2012 IEEE international conference on robotics and automation. pp. 5384–5391. IEEE (2012)
26. Wu, J., Zhang, C., Xue, T., Freeman, W.T., Tenenbaum, J.B.: Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In: Proceedings of the 30th International Conference on Neural Information Processing Systems. pp. 82–90 (2016)