2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

GraspAda: Deep Grasp Adaptation through Domain Transfer

Yiting Chen[1], Junnan Jiang[1], Ruiqi Lei[2], Yasemin Bekiroglu[3,4], Fei Chen[5†], and Miao Li[1†]

***Abstract*— Learning-based methods for robotic grasping have been shown to yield high performance. However, they rely on expensive-to-acquire and well-labeled datasets. In addition, how to generalize the learned grasping ability across different scenarios is still unsolved. In this paper, we present a novel grasp adaptation strategy to transfer the learned grasping ability to new domains based on visual data using a new grasp feature representation. We present a conditional generative model for visual data transformation. By leveraging the deep feature representational capacity from the well-trained grasp synthesis model, our approach utilizes feature-level contrastive representation learning and adopts adversarial learning on output space. This way we bridge the domain gap between the new domain and the training domain while keeping consistency during the adaptation process. Based on transformed input grasp data via the generator, our trained model can generalize to new domains without any fine-tuning. The proposed method is evaluated on benchmark datasets and based on real robot experiments. The results show that our approach leads to high performance in new scenarios.**

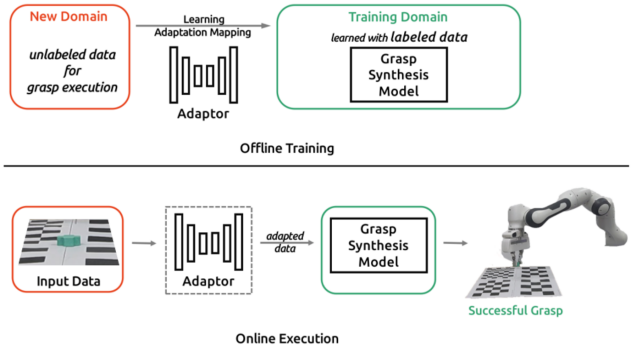

Fig. 1. Our proposed grasp feature-oriented domain adaptation method operates on input grasp data. We adopted a GAN-based generator as our adaptor to transfer the data from the new domain close to the training domain while keeping its grasp feature consistency, in order to encourage the pre-trained model to remain functional.

I. INTRODUCTION

Thanks to the advances in supervised learning approaches, vision-based end-to-end methods have become a dominant paradigm for direct robotic grasp synthesis during the past decade. How to generalize the learned grasping ability obtained from a specific domain to new domains remains a challenging problem.

Supervised learning-based method is a promising approach to learning the relationship between the input visual data and the output grasp predictions. With the help of well-labeled datasets [1]–[6], recent research has demonstrated considerable progress and reached relatively high precision on existing datasets [7]–[12]. Unfortunately, each dataset, regardless of being generated in simulation or using a real robot, has its intrinsic features including lighting, texture, camera conditions, etc. These intrinsic features can bring bias to trained neural networks which encumber their performance in other scenarios. Even though neural network models can be well designed, and trained, fully-supervised deep architectures are prone to overfit limited training data, and will still lose reliability when encountering outliers, i.e. inputs outside the training dataset domain.

One feasible way is following the idea of domain randomization to generate large-scale training data [2], [4]. By extensively randomizing important factors that affect perception and have a great impact on final grasp performance, such as lighting conditions, object shapes, textures, background, etc., these methods are trying to build a unified model to represent all the possible scenarios. However, this approach severely increases the data amounts which heavily burdens the training process of neural network models. Instead of domain-level randomization, other researchers [13]–[15] aim at enriching the grasp feature variation to further enhance the grasp networks' performance on less graspable objects or alleviate the data insufficiency problem. Though these approaches are capable of enhancing the model's capability to some extent, they didn't consider sensor conditions and other domain-variant ambient noises.

In this paper, we propose a novel grasp feature oriented input data adaptation strategy, GraspAda, to generalize the trained model from the training domain to the new domain. By involving adversarial learning on output space, we directly make the predicted grasp distributions close to each other across adapted and training data domains. The idea of contrastive representation learning is also adopted to enforce invariance of grasp relative features.

GraspAda is a general approach to grasp data adaptation that provides reliable generalization for trained grasp synthesis models from a specific domain (Fig. 1). The main contributions of the work are summarised as follows:

*This work was supported by Suzhou Key Industry Technology Innovation Project under the grant agreement number SYG202121.

[1]Yiting Chen, Junnan Jiang and Miao Li are with the Institute of Technological Sciences, Wuhan University, Wuhan 430072, China. {chenyiting, jiangjunnan, miao.li}@whu.edu.cn

[2]Ruiqi Lei is with the Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen 518055, China. leirq22@mails.tsinghua.edu.cn

[3]Yasemin Bekiroglu is with the Department of Electrical Engineering, Chalmers University of Technology, Goteborg SE-41296, Sweden. [4]Yasemin Bekiroglu is also with the Department of Computer Science, University College London, UK. yaseminb@chalmers.se; y.bekiroglu@ucl.ac.uk

[5]Fei Chen is with the Department of Mechanical and Automation Engineering, T-Stone Robotics Institute, The Chinese University of Hong Kong, Hong Kong, China. f.chen@ieee.org

†Corresponding authors: Miao Li, Fei Chen

1) We propose a method for feature-level domain adaptation via adversarial learning on output space. A feature-level contrastive learning scheme is developed to enforce the grasp relative feature consistency during adaptation.
2) We demonstrate that a trained grasp synthesis model from a specific domain can be easily generalized to a new domain only through data adaptation without any model fine-tuning.
3) The proposed approach is tested using both benchmark datasets and a real robot. The results show that the trained grasp synthesis model's performance in new domains is significantly improved.

II. Related Work

Grasping Data Augmentation can compellingly improve the performance of learning-based grasp synthesis methods. Handcrafted methods such as shifting, scaling, and rotating contribute empirically to the trained model's accuracy on a dataset [7]. These traditional methods have strong limitations on cross-domain feature variation. Randomization of scene configurations is a common approach in data generation [2], [4], [6], [16]. Though the grasping models trained on datasets with extensive generalized visual representations have certain generalization performances in other scenarios, it becomes challenging to train with large-scale data. [13]–[15] generate new data at the grasp feature level to further improve the grasp model's robustness for new scenarios. Most grasp data augmentation approaches try to improve the scale and quality of the training dataset, to obtain a better grasping model in a trial-and-error manner. However, the way of randomized data augmentation is heuristic, which is less efficient and a large amount of data burdens the model training process.

Transfer Learning methods for robotic grasping focus on how to transfer a learned neural network model from the source domain to the target domain, which enables robots to efficiently gain knowledge from simulation or limited data. Domain randomization and domain adaptation are both promising ways to bridge the gap between different domains. Domain randomization involves a tremendous variation in non-essential aspects of training data distribution, where data modality plays an important role because different modality of data carries different physical information. The depth image is relatively easy to simulate while also containing highly abstract appearance properties of real-world objects. With appropriate simulated depth images [17]–[19], the trained model can be directly transferred into the real world and achieve satisfactory results. Randomization on other data modalities such as RGB [20]–[22] requires more tuning simulation parameters, scene configurations, and data amounts, which can also perform cross-domain object grasping. Domain adaptation approaches focus on how to learn a neural network model to map from the source domain to the target domain. Approaches based on generative adversarial networks [23] are commonly preferred in domain adaptation. Due to the high demand for data consistency during the adaptation in robotic grasping, the idea of cycle consistency introduced in CycleGAN [24] is adopted to enforce the data feature consistency [25]–[27]. However, the underlying bijective assumption behind cycle consistency is sometimes too hard to meet. Recent research in computer vision has demonstrated that contrastive representation learning [28]–[30] is also able to achieve one-direction image translation while reflecting the input image's content. Differently than these approaches, we achieve input grasp data adaptation via a GAN-based learning framework, in which the conditional generator transfers data from the new domain into the adapted domain. In detail, an adversarial learning approach is deployed on the grasp prediction space to bridge the gap between the training domain and the adapted domain. Moreover, contrastive representation learning is adopted to keep the grasp feature consistent during the adaptation process. By adapting grasp data into the adapted domain, which shares enough similarities with the training domain, the trained grasp model retains its function in the new domain.

III. Grasp Feature Oriented Data Adaptation

Even if a good mapping between visual data and grasp label is learned from an existing dataset, the trained grasp synthesis model is generally biased due to intrinsic bias in the training data, which hinders the model's performance in new domains. Our strategy aims at solving the following challenges:

- Given a trained grasp synthesis model G, how to generalize it to new domains through input data adaptation using a GAN-based generator Gen.
- How to maintain the deep grasp feature consistency during the Gen transfer process, while bridging the gap of domain-specific bias between the new domain and the training data domain.

The architecture of our strategy is shown in Fig. 2. Our proposed framework does not rely on a specific grasp synthesis method or even the input data modality. Here we choose a state-of-the-art method [8] with RGB-D input data modality as our grasp synthesis network.

First, a grasp synthesis network model is trained in a supervised fashion on an available dataset, and the available dataset belongs to *training domain*. Once trained on a given dataset, our goal is to leverage the effective feature representation ability of our pre-trained model and learn a conditional generator to transfer data from *new domain* into an *adapted domain*. The *adapted domain* should be close enough to the *training domain* in order to encourage the trained neural network model to remain functional.

A. Problem Formulation

In this paper, the grasp synthesis model is expressed as a mapping between visual data X and grasp pose Y. $X = (C, D) \in \mathbb{R}^{4\times H\times W}$ is an RGB-D image that consists of an RGB image $C \in \mathbb{R}^{3\times H\times W}$ and a depth image $D \in \mathbb{R}^{H\times W}$. $Y = (\Theta, W, Q) \in \mathbb{R}^{3\times H\times W}$ represents the planar antipodal grasp pose of the end-effector, (Θ, W, Q) represents three images in the form of grasp angle, Θ, grasp width, W, and grasp quality score, Q, respectively calculated at every pixel

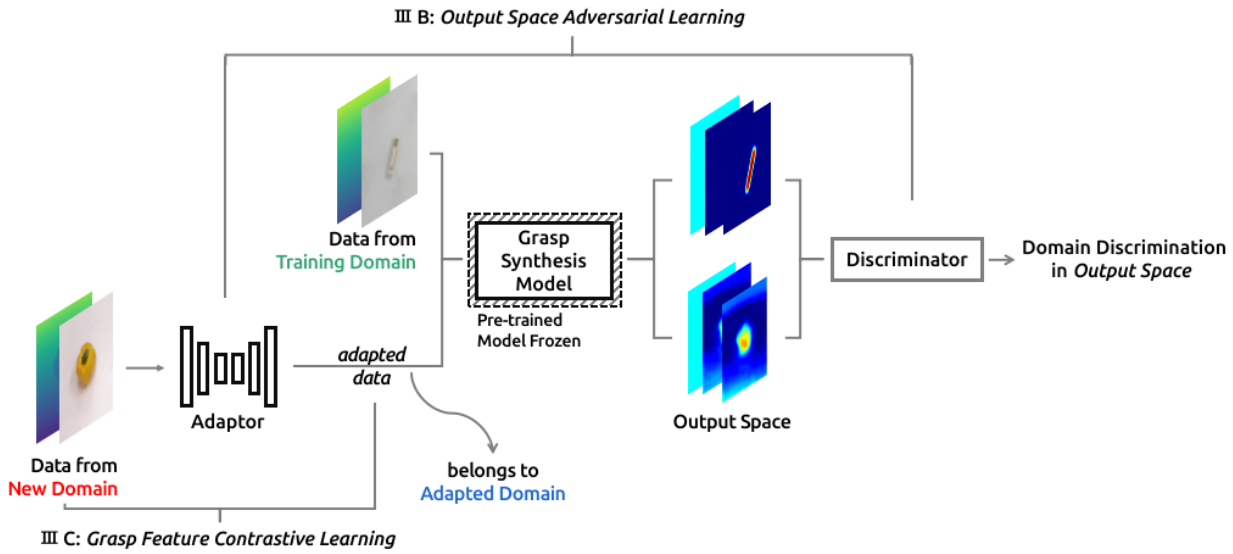

Fig. 2. The overall structure of our data adaptation strategy. III B: By learning a discriminator capable of distinguishing the output predictions, our adaptor can transfer data into the adapted domain similar to the training domain. III C: A contrastive learning scheme is adopted to preserve the grasp feature consistency during the adaptation process.

of an RGB-D image. We make two assumptions to abstract this process better:

1. There exists an ideal mapping between domain-invariant grasp features and their corresponding applicable grasp poses. However, for models trained on datasets from a specific domain, they are actually learning the mapping between domain-specific grasp features and grasp labels. Domain-specific grasp feature consists of domain-invariant grasp feature and the intrinsic bias of each data domain.
2. The well-trained model on grasp dataset from a specific domain carries the ability to extract domain-specific grasp feature from X and map it to Y.

The feature extraction backbone of the pre-trained grasp synthesis model is responsible for encoding low-level grasp features and the heads of the model are responsible for mapping the predicted grasp pose as follows:

$$F' = (F, N) = G_{backbone}(X) \tag{1}$$

$$Y = G_{heads}(F') \tag{2}$$

in which F' denotes the domain-specific grasp feature that consists of domain-invariant grasp feature F and domain-specific bias N. We use grasp data that was collected from two different domains: $X_t = \{x_t \in \mathcal{T}\}$ denotes well-labeled training dataset from training domain $\mathcal{T}$ (e.g., synthetic data from the robotic simulator or limited manually labeled data from the real world), and $X_s = \{x_s \in \mathcal{S}\}$ denotes data without labels from new domain $\mathcal{S}$ (e.g., sensor data from new scenarios in the real world).

A grasp synthesis model trained with data from $\mathcal{T}$, is capable of extracting domain-specific grasp feature in $\mathcal{T}$ and mapping it to grasp label but might fail in other domains. Our approach tackles this problem by only adapting data using a GAN-based generator Gen by transferring data as follows:

$$Gen(X_s \in \mathcal{S}) \rightarrow X_{t'} \in \mathcal{T}' \tag{3}$$

in which $\mathcal{T}'$ denotes adapted domain. By replacing the original bias carried by X_s with adapted bias which shares enough similarity to $\mathcal{T}$, while keeping the low-level grasp features F consistent, our goal is to adapt $F_s = (F, N_s)$ into $F_{t'} = (F, N_{t'})$. After the adaptation process, our biased model is actually facing the adapted data $X_{t'}$ with familiar bias $N_{t'}$.

Therefore, an output space adaptation is implemented using adversarial learning, where we directly make the predicted label distributions by the pre-trained model G close to each other across adapted and training domains. Meanwhile, the grasping feature is maintained using contrastive representation learning between the new data and the adapted data. With the proposed network, we formulate the adaptation task containing two loss functions from both modules:

$$\begin{aligned}\mathcal{L}(X_s, X_t) =& \lambda_1 \mathcal{L}_{adv}(X_s \rightarrow X_{t'}, X_t) \\ &+ \lambda_2 \mathcal{L}_{con}(X_s, X_s \rightarrow X_{t'})\end{aligned} \tag{4}$$

where $\mathcal{L}_{adv}$ is the MSE loss to evaluate the domain discrimination accuracy of our discriminator. The output predictions with domain label (0 denotes T and 1 denotes $\mathcal{T}'$) are adopted as training data to train the discriminator. $\mathcal{L}_{con}$ is the PatchNCE loss that keeps the grasp feature consistent, by deploying a classification calculation between sampled patches. In (4), λ_1, λ_2 are the weights used to balance the two losses.

B. Output Space Adaptation

We consider grasp pose representations $Y = (\Theta, W, Q)$ in similar scenarios as structured outputs that contain global similarities between the new domain $\mathcal{S}$ and the training domain $\mathcal{T}$ [31]. Our pre-trained model is capable of predicting

high quality Y_t in domain $\mathcal{T}$. Our pre-trained model should predict similar structured output between adapted domain $\mathcal{T}'$ and training domain $\mathcal{T}$ as shown in Fig. 3. Thus, we utilize this property to adapt low-dimensional grasp representation of model predictions via an adversarial learning scheme as shown in Fig. 2. A Resnet-based conditional generator Gen with 9 residual blocks is utilized for data adaptation, and a convolutional discriminator Dis with similar architecture as [32] is used for domain discrimination. Parameters of the pre-trained grasp synthesis model remain frozen during the output space adversarial learning. First, the parameters in Gen are frozen, parameters will only be updated in Dis. Given the grasp data x from domain $\mathcal{T}'$ or $\mathcal{T}$, we feed x to the pre-trained grasp synthesis model to obtain the grasp prediction representation y. Based on y and its domain label, a discriminator that is capable of distinguishing the difference between the output predictions from $\mathcal{T}'$ and $\mathcal{T}$ is learned. Then we freeze Dis and only update the Gen to adapt the output predictions of the data from $\mathcal{T}'$ similar to $\mathcal{T}$, under the evaluation criterion provided by Dis. The adversarial loss can be written as:

$$\begin{aligned}\mathcal{L}_{adv}(X_s, X_t) =& \mathbb{E}_{x_t \sim X_t} \log Dis(G(x_t)) + \\ & \mathbb{E}_{x_s \sim X_s} \log Dis(1 - G(Gen(x_s)))\end{aligned} \tag{5}$$

By alternately learning Dis and Gen, the ultimate goal is to bridge the gap between the output spaces of the adapted domain and the training domain, through the data adaptation process.

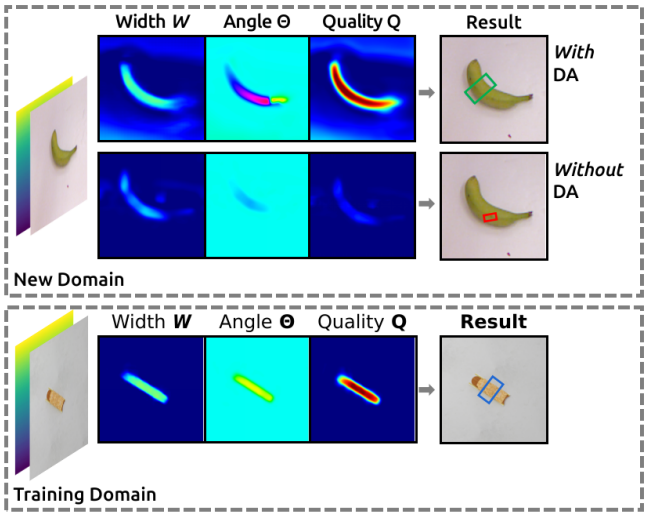

Fig. 3. Example grasp prediction results using the trained model. Similar scenarios from different domains shares similar output space. The data adaptation (DA) process bridges the gap between the output space of the adapted domain and the training domain.

C. Grasp Feature Contrastive Learning

Although performing adversarial learning on the output space directly adapts grasp prediction distribution, low-level grasp features are not guaranteed to be consistent through such a process. Even slight changes in grasp features may cause failed grasp prediction on grasp data. We adopted the idea of patchwise contrastive learning [29] to keep the grasp features consistent during the adaptation process. The scheme of our grasp feature-level contrastive learning is shown in Fig. 4. Given a specific patch from adapted data $x_{t'}$, its grasp feature vector $v \in \mathbb{R}^K$ should associate with its corresponding location patch's feature $v^+ \in \mathbb{R}^K$ from new data x_s and disassociate with the other N random sampled patches' grasp features $v_n^- \in \mathbb{R}^K$. We adopted the $G_{backbone}$ from the pre-trained grasp synthesis model to extract low-level grasp feature from both new data x_s and adapted data $x_{t'}$. The features are passed through a two-layer MLP network to map them to the K-dimensional vector space as used in SimCLR [28]. We aim to match corresponding input-output patches at a specific location and leverage the other patches within the input as negatives. By comparing with corresponding positive and other negative feature vectors, we enforce each patch to keep its own grasp feature consistent during adaptation. The comparison process is calculated using PatchNCE loss introduced by [29].

$$\mathcal{L}(v, v^+, v_n^-) = -\log \left[\frac{e^{(v \cdot v^+ / \tau)}}{e^{(v \cdot v^+ / \tau)} + \sum_{n=1}^{N} e^{(v \cdot v_n^- / \tau)}} \right] \tag{6}$$

$$\mathcal{L}_{con}(X_t \rightarrow X_{t'}, X_t) = \mathbb{E}_{x_{t'} \sim X_{t'}} \sum_{m=1}^{M} \mathcal{L}(v, v^+, v_n^-) \tag{7}$$

In (7), M denotes the number of comparisons and $\tau = 0.2$ is the temperature to scale the distance between the compared samples. The parameters of our grasp feature extraction model $G_{backbone}$ also remain frozen during the training process, to maintain its feature extraction function as unchanged. To do so, the Gen learns to pay attention to the domain-invariant grasp feature between the two domains, such as object geometry, while being able to adapt the domain-specific feature, such as the textures or lighting condition.

IV. Experiments

We evaluate the domain adaptation ability of our proposed strategy by comparing the performance of a grasp synthesis model (trained on the Jacquard Grasp Dataset [2]) on the Cornell Grasp Dataset [1] with and without the adaptation process. In addition, we also perform extensive physical evaluations on a Franka Emika Panda with a 2-finger gripper. As introduced above, we choose GR-ConvNet [8] as our grasp synthesis network. All experiments ran on a desktop running Ubuntu 18.04 with a 2.7 GHz Intel Core i5-6400 Quad-Core CPU and an NVIDIA GeForce 1060, and we used an NVIDIA GeForce GTX 3070 for training adaptation models.

A. Evaluation on Grasp Datasets

The rectangle metric [33] is used to evaluate whether the predicted grasp pose is a successful one. The evaluation standard is formalized by the intersection over union (IOU) and the offset of grasp orientation between the predicted grasp rectangle and labeled rectangle adopted by [8].

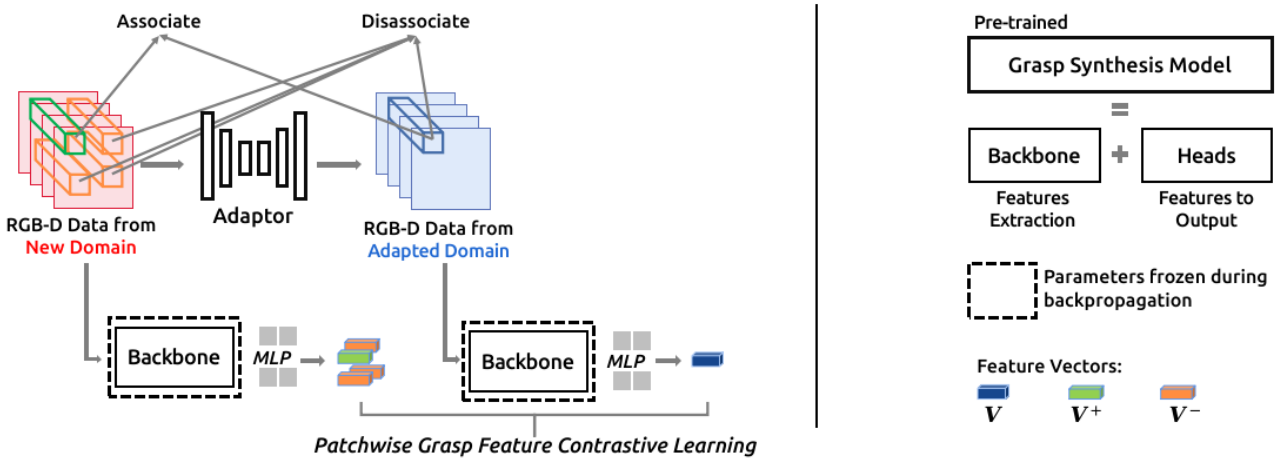

Fig. 4. Given a sampled patch from the adapted data, we compare it with the input patch at the same location. An (N+1)-way classification problem is set up as [29], where N negative patches are sampled from the same input image at different locations. The feature extraction backbone of the pre-trained grasp synthesis model is reused and a two-layer MLP network is added for feature vector space projection.

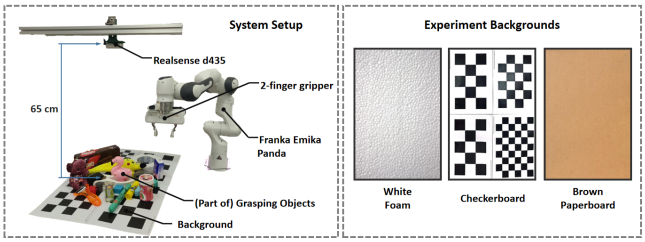

Fig. 5. Left part demonstrates the real-world robotic system we used to run grasping tests. We employ a Franka Emika Panda robot and an Intel Realsense D435 depth camera, to test the grasp success rate on different objects with different backgrounds. The right part shows the three different backgrounds we used for real-world grasping experiments.

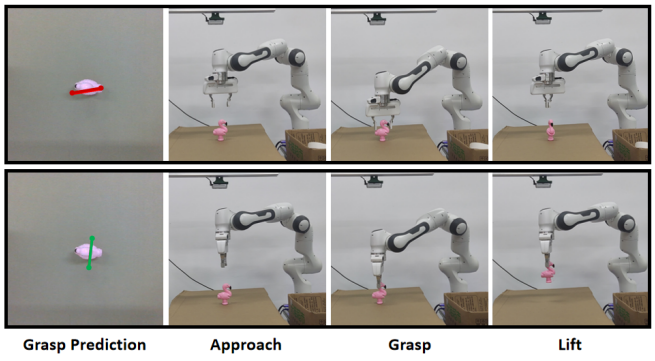

Fig. 6. We respectively run 10 grasp attempts on each object with the same pose with and without the adaptation process. The upper line figures show the grasping process without adaptation and the bottom line figures show the grasping process with adaptation.

The grasp synthesis model is completely trained on Jacquard Grasp Dataset and achieves a relatively high accuracy of 91.3%. First, we directly feed the grasp data from Cornell Grasp Dataset into our trained model without any adaptation process, following the evaluation method introduced before, our trained model only achieves a success rate of 23.4%. The prediction results and training details

TABLE I
PERFORMANCE OF PRE-TRAINED MODEL ON CORNELL DATASET AND REAL-WORLD SCENARIOS

Dataset/Background	direct	with DA	Improvement
Cornell	23.4%	74.6%	**218.8%**
White Foam	76.7%	86.7%	11.3%
Checkerboard	40%	80%	**100%**

are shown in Fig. 7, which indicates that our trained model totally loses its ability to grasp width detection. The dataset for training the adaptor consists of 150 unlabeled RGB-D images from Cornell Grasp Dataset and 150 unlabeled RGB-D images from Jacquard Grasp Dataset. We set $\lambda_1 = 1$ and $\lambda_2 = 0.6$ to train the adaptor, with a learning rate of 0.002 and batch size of 1. The pre-trained model achieves an accuracy of 74.6% on the Cornell Grasp Dataset after data adaptation, which is 218.8% higher than before (Table I). By comparing the grasp performance on Cornell Dataset with and without the data adaptation process, the experiment result shows that our data adaptation method greatly improves the grasp success rate, making the trained model perform more robust and precise on data from the new domain.

B. Real World Experiments

To evaluate the adaptation ability of our method in real-world applications, we also conduct extensive real-robot experiments. As shown in the left side of Fig. 5, the experiment is carried out on a Franka Emika Panda robot with a two-finger gripper. A fixed RealSense camera is mounted 65 cm in height right above our target objects. We set up two kinds of real-world grasping experiments with three different backgrounds as shown on the right side of Fig. 5, to comprehensively demonstrate the adaptation ability of our proposed method:

1) Experiment A: We select 30 novel objects with two different backgrounds, one is white foam board background

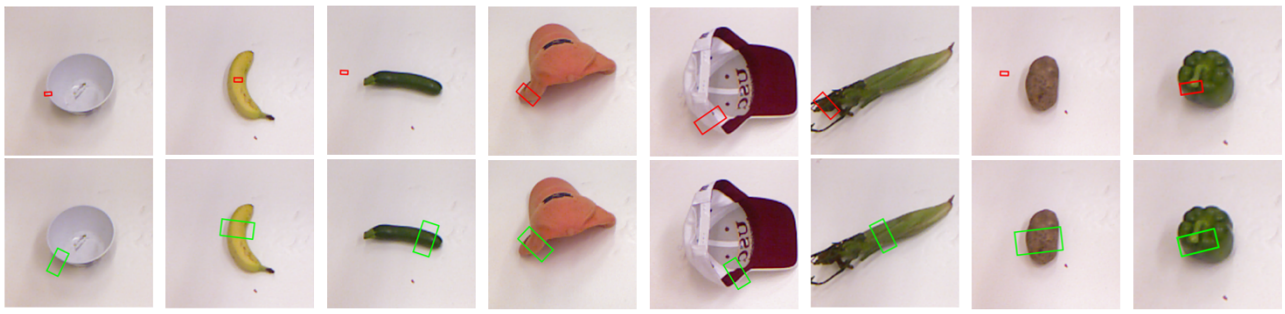

Fig. 7. We test our pre-trained model with and without the proposed data adaptation process. We set $\lambda_1 = 1$ and $\lambda_2 = 0.6$ to train the adaptor. The figures in the upper row show the generation result without our adaptation method, and the figures in the bottom row show the generation result after data adaptation. The comparison result on each object clearly shows that our data adaptation method makes the trained model more robust and more precise.

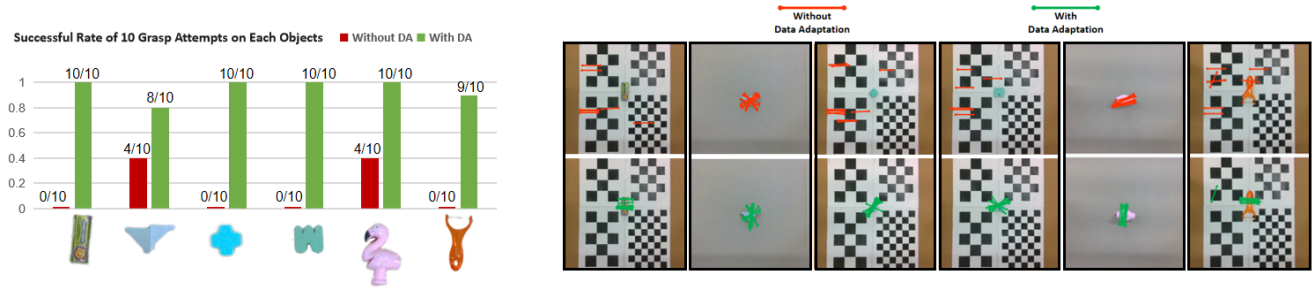

Fig. 8. Left table shows the evaluation result of Experiment B. Right figures visualize predicted grasp poses with a grasp quality score higher than 0.3 among our ten grasp attempts. The top line figures denote grasp attempts without data adaptation and the bottom line figures denote grasp attempts with data adaptation. We set $\lambda_1 = 1, \lambda_2 = 0.3$ and $\lambda_1 = 1, \lambda_2 = 0.5$ to train two adaptors respectively facing the adversarial and brown background.

which is similar to Jacquard Grasp Dataset, and the other one is an adversarial background with an asymmetry checkerboard pattern for grasping evaluation. By feeding the RGB-D data with and without data adaptation, we compare the 30 novel objects' overall success grasping rate. We separately collect a small quantity (less than 100 RGB-D images) of data from two different domains and train an adaptor for each domain. The experiment result is listed in Table I. The adversarial checkerboard pattern can be fatal for the RGB-D-based grasp synthesis model on small objects such as a wooden cube. A noisy new domain can cause damage to the trained model, leading to a success rate of only 12/30 (40%) on 30 novel objects. By leveraging the grasp feature-oriented data adaptation, the same trained model without any fine-tuning achieves a success rate of 24/30 (80%) on novel objects with an adversarial background. Even for a new domain with a similar background as training data, our data adaptation method is still able to improve the model's performance with an increased success rate of 11.3%. The experiment results show that our data adaptation strategy improves the grasping performance in both new domains. The greater the difference between the different domains, the worse the performance of the pre-trained model, and the higher the improvement of our method.

2) Experiment B: We select 6 different objects with characteristics such as plastic, metal, wood, irregular shape, semi-transparent material, etc., and respectively run 10 grasp attempts on each object with the same pose under adversarial checkerboard background or brown paperboard background with and without data adaptation. Then we compare the grasp success rate between them. The process of the grasping experiment is shown in Fig. 6. As Experiment A introduced above, we first collect a handful of data from both domains and respectively train a data adaptor. The grasp success rates and experiment results are listed in Fig. 8. For those scenarios our trained model still functions but may lose its stability, our adaptation method increases the grasp success rate greatly. For those small targets with shallow depth images and similar shapes with background noise such as a checker, our adaptation can save an invalid grasp synthesis model and help it back to function.

V. Conclusion and Discussion

In this paper, we present a domain adaptation strategy for grasp data transfer. We leverage the power of adversarial learning on output space to bridge the domain gap and contrastive learning to maintain feature consistency during the adaptation process. Experimental results show that our adapted data greatly improves the pre-trained model's performance on novel objects from a new domain.

In the future, we will test our proposed method in a higher-degree world such as 6D grasping. In addition, we will explore the effect of different generator and discriminator network structures, such as U-Net, on the performance of the proposed strategy and demonstrate more comparisons with other domain adaptation methods. We will also investigate the influence of different combinations of weights in loss functions.

REFERENCES

[1] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks," in *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2015, pp. 1316–1322.

[2] A. Depierre, E. Dellandréa, and L. Chen, "Jacquard: A large scale dataset for robotic grasp detection," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 3511–3516.

[3] F.-J. Chu, R. Xu, and P. A. Vela, "Real-world multiobject, multigrasp detection," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3355–3362, 2018.

[4] H.-S. Fang, C. Wang, M. Gou, and C. Lu, "Graspnet-1billion: A large-scale benchmark for general object grasping," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11 441–11 450.

[5] C. Eppner, A. Mousavian, and D. Fox, "Acronym: A large-scale grasp dataset based on simulation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6222–6227.

[6] H. Zhang, D. Yang, H. Wang, B. Zhao, X. Lan, J. Ding, and N. Zheng, "Regrad: A large-scale relational grasp dataset for safe and object-specific robotic grasping in clutter," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2929–2936, 2022.

[7] D. Morrison, P. Corke, and J. Leitner, "Learning robust, real-time, reactive robotic grasping," *The International journal of robotics research*, vol. 39, no. 2-3, pp. 183–201, 2020.

[8] S. Kumra, S. Joshi, and F. Sahin, "Antipodal robotic grasping using generative residual convolutional neural network," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 9626–9633.

[9] S. Ainetter and F. Fraundorfer, "End-to-end trainable deep neural network for robotic grasp detection and semantic segmentation from rgb," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 13 452–13 458.

[10] M. Gou, H.-S. Fang, Z. Zhu, S. Xu, C. Wang, and C. Lu, "Rgb matters: Learning 7-dof grasp poses on monocular rgbd images," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 13 459–13 466.

[11] X. Zhu, L. Sun, Y. Fan, and M. Tomizuka, "6-dof contrastive grasp proposal network," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6371–6377.

[12] X. Zhu, Y. Zhou, Y. Fan, L. Sun, J. Chen, and M. Tomizuka, "Learn to grasp with less supervision: A data-efficient maximum likelihood grasp sampling loss," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 721–727.

[13] D. Wang, D. Tseng, P. Li, Y. Jiang, M. Guo, M. Danielczuk, J. Mahler, J. Ichnowski, and K. Goldberg, "Adversarial grasp objects," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2019, pp. 241–248.

[14] J. Jiang, X. Xiao, F. Chen, and M. Li, "Learning grasp ability enhancement through deep shape generation," in *International Conference on Intelligent Robotics and Applications*. Springer, 2022, pp. 735–746.

[15] Y. Lin, C. Tang, F.-J. Chu, and P. A. Vela, "Using synthetic data and deep networks to recognize primitive shapes for object grasping," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 10 494–10 501.

[16] A. Mousavian, C. Eppner, and D. Fox, "6-dof graspnet: Variational grasp generation for object manipulation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2901–2910.

[17] M. Danielczuk, M. Matl, S. Gupta, A. Li, A. Lee, J. Mahler, and K. Goldberg, "Segmenting unknown 3d objects from real depth images using mask r-cnn trained on synthetic data," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7283–7290.

[18] 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*. PMLR, 2017, pp. 291–300.

[19] 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 *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 3482–3489.

[20] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "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)*. IEEE, 2017, pp. 23–30.

[21] S. James, A. J. Davison, and E. Johns, "Transferring end-to-end visuomotor control from simulation to real world for a multi-stage task," in *Conference on Robot Learning*. PMLR, 2017, pp. 334–343.

[22] F. Sadeghi, A. Toshev, E. Jang, and S. Levine, "Sim2real viewpoint invariant visual servoing by recurrent control," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4691–4699.

[23] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[24] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[25] O.-M. Pedersen, E. Misimi, and F. Chaumette, "Grasping unknown objects by coupling deep reinforcement learning, generative adversarial networks, and visual servoing," in *2020 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2020, pp. 5655–5662.

[26] D. Ho, K. Rao, Z. Xu, E. Jang, M. Khansari, and Y. Bai, "Retinagan: An object-aware approach to sim-to-real transfer," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 10 920–10 926.

[27] K. Rao, C. Harris, A. Irpan, S. Levine, J. Ibarz, and M. Khansari, "Rl-cyclegan: Reinforcement learning aware simulation-to-real," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[28] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[29] T. Park, A. A. Efros, R. Zhang, and J.-Y. Zhu, "Contrastive learning for conditional image synthesis," in *ECCV*, 2020.

[30] Z. Wu, Z. Zhu, J. Du, and X. Bai, "Ccpl: Contrastive coherence preserving loss for versatile style transfer," *arXiv preprint arXiv:2207.04808*, 2022.

[31] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. Chandraker, "Learning to adapt structured output space for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7472–7481.

[32] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[33] Y. Jiang, S. Moseson, and A. Saxena, "Efficient grasping from rgbd images: Learning using a new rectangle representation," in *2011 IEEE International conference on robotics and automation*. IEEE, 2011, pp. 3304–3311.