Instance-wise Grasp Synthesis for Robotic Grasping

Yucheng Xu[1], Mohammadreza Kasaei[1], Hamidreza Kasaei[2], and Zhibin Li[3]

Abstract—Generating high-quality instance-wise grasp configurations provides critical information of how to grasp specific objects in a multi-object environment and is of high importance for robot manipulation tasks. This work proposed a novel Single-Stage Grasp (SSG) synthesis network, which performs high-quality instance-wise grasp synthesis in a single stage: instance mask and grasp configurations are generated for each object simultaneously. Our method outperforms state-of-the-art on robotic grasp prediction based on the OCID-Grasp dataset, and performs competitively on the JACQUARD dataset. The benchmarking results showed significant improvements compared to the baseline on the accuracy of generated grasp configurations. The performance of the proposed method has been validated through both extensive simulations and real robot experiments for three tasks including single object pick-and-place, grasp synthesis in cluttered environments and table cleaning task.

I. INTRODUCTION

In human-centered environments, robots are becoming increasingly useful in a variety of applications related to manipulating specific objects, thus a robust and efficient instance-wise grasp synthesis approach is of great importance, as it provides vital information (e.g., location and grasp configurations) for manipulating target objects. Image-based instance-wise grasp synthesis in cluttered environments is yet a very challenging task. It aims at generating high-quality grasp configurations for specific objects in the multi-object scenario using a single image as the input. In this paper, we seek to leverage the success of prior research on semantic instance segmentation as well as generative grasp synthesis to design a novel model, which solves instance-wise grasp synthesis tasks in a single-stage manner for robotic manipulations.

Designing an image-based instance-wise grasp synthesis model is difficult for two key reasons: (i) current 2D grasp synthesis approaches either employ a region proposal network to find graspable regions [1], [2], [3], [4], or adopt generative model to predict pixel-wise grasp configurations [5], [6], [7], [8]. Both of these approaches are limited to scene-level grasp synthesis; in other words, they can only determine which parts of the scene are graspable, but not which objects. (ii) Since the grasp configurations are mostly generated from regional or global features, the relationship between objects and grasps is not clear. Thus, it is difficult to determine the grasp affiliations.

Recent research tackle instance-wise grasp synthesis tasks in a two-stage way [9], [10], [11], [12]: (i) in the first stage, grasp configurations will be generated for all graspable regions of the global input; (ii) then, the generated grasp configurations will be assigned to specific objects with the help of additional information, which is often derived from object detection or semantic segmentation. Two-stage methods inherently lack the relationship between predicted grasp configurations and detected objects, since the object detection task and object detection/segmentation task are completed separately. These methods mostly suffer from inaccurate grasp assignment, lack of class-specific information, and inefficiency stemmed from its cascade structure [9], [12] (Details in Fig. 3, Section IV-A).

To address these limitations and solve instance grasp synthesis tasks in a more efficient and accurate way, we proposed the **S**ingle-**S**tage **G**rasp (SSG) synthesis model. The term "single-stage" stands for the way of generating instance-wise grasp configurations. The grasp configurations are generated for each object instance directly without additional refinement or assignment modules which are commonly used in previous methods [9], [10], [11], [12].

Our proposed SSG formulates the instance-wise grasp synthesis as two parallel tasks. The first task focuses on generating a set of prototype masks for the input RGB-D image, which can be regarded as vocabulary or global descriptors. The second task is to detect objects in the image and predicts extra sets of coefficients for each detected object. Finally, for each object that survives Non-Maximum Suppression (NMS), those sets of coefficients are used to linearly assemble prototype masks to generate both instance segmentation and grasp masks. Here, grasp masks refer to pixel-wise grasp configurations proposed by [5], [6]. In our proposed method, SSG, bounding box, class label, instance mask, and grasp masks are generated in parallel for each detected object which strongly keep the relationship between objects and grasps. The overall architecture (Fig. 1) clearly delineates the unique process of the proposed method.

The contributions are summarized as follows: **(1)** A novel **S**ingle-**S**tage **G**rasp (SSG) synthesis model, which solves the challenging instance-wise grasp synthesis tasks in a single-stage manner; **(2)** The proposed SSG outperforms state-of-the-art on OCID-Grasp dataset, and performs competitively on JACQUARD dataset, through the evaluation on extensive tests and validations in both simulations and real robot experiments.

The proposed SSG succeeded in synthesizing instance-wise grasp configurations in highly cluttered scenarios, where objects had 10% to 25% of overlapped areas, while other two-stage methods failed due to the mismatch between grasps and objects, and segmentation errors. Further, we demonstrate the scalability of the proposed method by extending it to affordance detection tasks (See Section IV-D in details), and show the proposed method can be used as a general pipeline for multiple robot manipulation tasks.
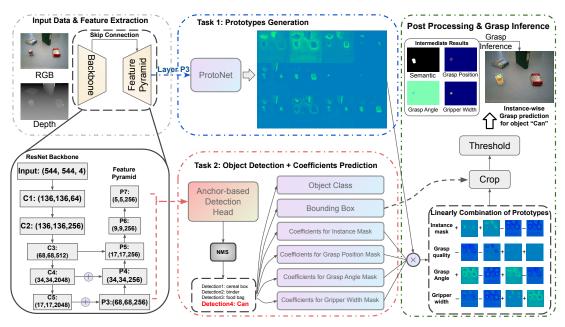
Fig. 1: System structure of the proposed model. The grasp configurations is generated as follows: (1) Feature extraction; (2) Generating of prototype masks; (3) Object detection, prediction of coefficients; (4) Linearly combination of prototypes with different predicted coefficients to generate instance mask and grasp masks; (5) Post processing to infer grasp configurations from generated grasp masks.

II. RELATED WORKS

Learning-based 2D robotic grasp synthesis has been increasingly attracting attention in past years [13]. Modern learning-based 2D grasp synthesis approaches can be roughly categorized into detection-based and generative approaches. Detection-based approaches adopt object detection pipelines and treat grasp synthesis task as a detection task, since grasp configurations can be represented as rotated rectangles in image plane [1], [2], [3], [14], [15]. The work of [1] performed transfer learning between object detection and grasp detection. A Rotated Region Proposal Network (RRPN), which is pretrained on object detection dataset, is adopted to generate graspable region proposals. A single-stage grasp detection network purely based on Region Proposal Network (RPN) was proposed in [2]. The grasp rectangles are directly regressed and classified from oriented anchors which are generated from RPN. ROI-GD [3] is a two-stage approach that detects grasp synthesis for specific regions by leveraging features from the object region rather than global input.

On the other hand, generative approaches produce pixel-wise grasp configurations for an input image [5], [6], [7], [8]. In this category, GG-CNN [5] approach aims to predict pixel-wise grasp configurations from depth images using generative neural network, where grasp configurations are embedded into three target masks representing grasp quality, grasp angle, and width of gripper respectively. Based on such representation of grasp configurations, the work in [6] introduced residual structure into generative neural network. Also, Guassian kernel are introduced in [7], [8], [16] to better represent grasp configurations. In comparison with detection-based grasp synthesis methods, generative grasp synthesis methods avoid the gener-

ation of redundant region proposals and discrete sampling.

Despite improvements in learning based grasp synthesis [3], [6], instance-wise grasp synthesis is still challenging. Most approaches solve instance-wise grasp prediction problems indirectly, by defining a set of surrogate detection and assignment tasks [9], [10], [11], [12]. In such pipelines, additional semantic segmentation or object detection branches are commonly adopted to assign grasp candidates to a specific object.

Representative multi-task frameworks were proposed in [9], [12] which include two networks for object detection and grasp detection respectively. The results of object detection were adopted to assign grasp candidates to specific objects. TOG-CRFs proposed in [10] adds a Conditional Random Field (CRF) to the grasp detection network, which models semantic contents of object regions to enable task-oriented grasp synthesis. Another work in [11] adds an semantic segmentation branch alongside the grasp detection branch to refine grasp candidates and assign them to target objects.

Mask-Grasp RCNN proposed in [17] is based on Mask-RCNN [18]: a instance segmentation network. The method in [17] adds additional regression heads to the Mask-RCNN [18] to detect and regress grasps from aligned Regions-of-Interest (RoIs) directly for each detected object instance. Mask-Grasp RCNN [17] is the first single-stage instance-wise grasp synthesis method, which is used in this work as a baseline of a single-stage method for the comparison study.

III. PROPOSED METHOD

A. Problem Formulation

This work aims to synthesize grasp configurations for each object from an RGB-D image in a single-stage way. The task is defined as: to predict grasp configurations for each
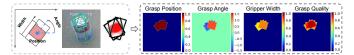
Fig. 2: 2D grasp rectangles are embedded into four different masks to represent grasp quality, grasp position, grasp angle and gripper width.

detected instance in an image plane. Grasp configurations in an image plane are commonly represented as rotated rectangle: $\text{Grasp}_{rect} = (x, y, \theta, w, q)$, such as the formulation in [1], [2], [3], [14], [15], where (x, y) corresponds to the center of grasp rectangle in the image coordinates, θ is the rotation in camera's frame of reference, w is the required width of gripper, q is the quality of grasping. In our method, we formulated an additional label (cls) for each grasp configurations that indicate which object it belongs to.

To enable instance-wise grasp prediction, we adopted an approach similar to [5], [6], and further developed a grasp representation that can be integrated with the existing instance segmentation framework. For each object instance, we embed its ground truth grasp configurations into multiple masks indicating grasp position, grasp quality, grasp angle and gripper width (see Fig. 2). For a better representation of grasp quality, for each pixel, we calculate the number of overlapped grasp rectangles which include the pixel itself, and use $Sigmoid$ function to generate the grasp quality mask.

B. Architecture

We developed the SSG, a single-stage grasp synthesis model, with insights from YOLACT [19]. Fig. 1 details the sub-modules and the workflow of our proposed method. First, a feature extraction module, consisting of ResNet-101 [20] and Feature Pyramid Network (FPN) [21], is adopted to extract shared multi-scale features from input RGB-D image. Deep layers (C3, C4, C5) from ResNet-101 module are linked to FPN. Then, the ProtoNet branch, which is a fully convolutional network [22] with k-channel output, is used to generate a set of 32 prototype masks ($k = 32$) for the entire input RGB-D image. P3 layer of FPN is used as the input of ProtoNet branch, as the largest and deepest feature layer of FPN, to produce more robust and fine-grained prototype masks. The concept of prototype masks is similar to those representation learning concepts for object detection from [23], [24], [25].

We note an important observation here: the learned prototype masks (feature embeddings) are generalized to different domains. We found that by using different coefficients to linearly assemble the same set of prototype masks, we can generate instance masks and grasp masks.

For the object detection branch, a typical anchor-based object detection branch is extended by adding N extra heads predicting $N \times k$ coefficients for each detected objects. For each object that survives NMS, we predict its class, bounding box, k coefficients for assembling its instance mask, k coefficients for assembling its grasp position mask, $2 \times k$ coefficients for assembling its grasp angle masks (represented in $sin(2\theta)$ and $cos(2\theta)$, θ is the valid grasp angle), and k

coefficients for assembling its width mask. These predicted sets of coefficients will be used to linearly assemble prototype masks generated from ProtoNet and form target output masks: semantic instance mask, grasp quality mask, grasp angle masks and gripper width mask.

C. Post Processing

Target Masks Generation. As shown in Fig. 1, the ProtoNet branch will generate $h \times w \times k$ prototype masks $\mathbf{P}$ for the input RGB-D image, where h, w denote the size of the prototype mask. $N \times k$ coefficients $\mathbf{C}$ are predicted for every detected object ($N = 5$). Then prototype masks $\mathbf{P}$ are linearly assembled with coefficients $\mathbf{C}$ to generate target masks, $\mathbf{M} = \text{Activation}(\mathbf{P}\mathbf{C}^\top)$. In this study, $\mathbf{M}$ is composed of five masks corresponding to instance mask, grasp quality mask, grasp angle masks and gripper width mask. For instance mask, grasp quality mask and gripper width mask, $Sigmoid$ activation function is used to limit the output range from 0 to 1. For grasp angle masks, $tanh$ activation function is used to limit the output range from -1 to 1.

Mask Crop. Generated target masks for each object are cropped using its bounding box. The ground truth bounding boxes are used in training, while the predicted ones are used during evaluation.

Grasp Configuration Inference. The grasp configurations are inferred based on the grasp masks obtained by linearly assembling prototype masks and cropping with bounding boxes. For each object instance, firstly a local maximum point is searched in its grasp quality map to find the point with the highest grasp quality and its pixel coordinates, then the grasp angles and gripper width are obtained from corresponding masks with pixel coordinates of the best grasp point.

D. Loss Function

Our loss function is composed of five different losses as: object classification (ℓ_{cls}), bounding box regression (ℓ_{box}), global semantic segmentation (ℓ_{smask}), instance segmentation (ℓ_{imask}), and grasp synthesis (ℓ_{gr}). ℓ_{cls}, ℓ_{box} and ℓ_{imask} are defined the same as in [19]. ℓ_{smask} is used to accelerate the convergence of our model. ℓ_g consists of five losses including grasp quality loss (ℓ_{gr-q}), grasp position loss (ℓ_{gr-p}), grasp angle loss (in sin and cos, $\ell_{gr-sin}, \ell_{gr-cos}$) and gripper width loss (ℓ_{gr-w}). $\ell_{gr-q}, \ell_{gr-sin}, \ell_{gr-cos}$ and ℓ_{gr-w} are calculated using smooth-L1 loss, ℓ_{gr-p} is calculated using binary cross entropy loss. The total loss $\mathcal{L}$ is summed as:

$$\mathcal{L} = \alpha_{cls}\ell_{cls} + \alpha_{box}\ell_{box} + \alpha_{imask}\ell_{imask}, \\ + \alpha_{gr}\ell_{gr} + \alpha_{smask}\ell_{smask} \tag{1}$$

where $\ell_{gr} = \alpha_p\ell_{gr-p} + \alpha_q\ell_{gr-q} + \alpha_{sin}\ell_{gr-sin} + \alpha_{cos}\ell_{gr-cos} + \alpha_w\ell_{gr-w}$.

IV. EVALUATION

We evaluate and benchmark the performance of the proposed method on instance-wise robotic grasp detection dataset OCID-Grasp [11], and class-agnostic robotic grasp detection dataset JACQUARD [26]. Moreover, a set of simulations and

TABLE I: Simulations results: 20 simulated tests were conducted for each object.

Objects	Apple	Banana	Lemon	Mug	Bowl	Bottle	Marker	Cereal Box
Success Rate [%]	90	85	85	80	75	75	90	90
Objects	Sponge	Soda Can	Juice Box	Cup	Spatula	Knife	Soap	Power Driller
Success Rate [%]	85	90	80	80	80	75	80	85

Fig. 3: Failure cases of two-stage method, compared to the correct results from the proposed single-stage method: (left) Failures of two-stage method caused by inaccurate grasp assignment; (right) Failures of two-stage method caused by segmentation errors.

TABLE II: Results of grasp accuracy on OCID-Grasp Dataset [11].

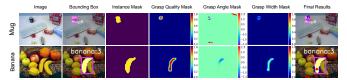| Method	Grasp Accuracy	Speed (FPS)
Deg_Seg_RGB[11]	89.02 %	31
Deg_Seg_RGBD[11]	89.84 %	31
SSG_RGB (ours)	**91.97** %	**39**
SSG_RGBD (ours)	**92.93** %	**39**

Fig. 4: Test results on OCID-Grasp dataset where grasp configurations were generated for each target object.

real robot experiments have been conducted to validate the performance of the proposed method for real-world robotics applications.

To evaluate and quantify the accuracy of predicted grasp configurations of each object on datasets, we applied an extended version of metrics by adding a new condition, based on the Jacquard Index proposed in [26]. That is, a grasp candidate is valid if the following conditions are satisfied: (i) The predicted class label of the grasp candidate is correct; (ii) The angle difference between the predicted grasp candidate and ground truth grasp is within $30°$; (iii) The Intersection over Union (IoU) of the predicted grasp rectangle and the ground truth is greater than 0.25.

A. Evaluation on OCID-Grasp Dataset

The **OCID-Grasp dataset** is an extension of Object Clutter Indoor Dataset (OCID) [27] annotated by [11], which consists of 1763 selected RGB-D images with over 11.4K segmented instance masks and 75K hand-annotated grasp rectangles with corresponding object class information. Objects in OCID-Grasp dataset are classified into 31 different categories. For each scenario, RGB image, depth image, semantic segmentation mask, and grasp annotation with instance labels are provided.

On OCID-Grasp dataset, our model is trained on the official train set and validated on the test set. To augment the size of datasets for training our network, based on OCID-Grasp, we applied random data augmentation including random photometric distortion, random clip and multi-scale resize. We outperform state-of-the-art on OCID-Grasp dataset with an overall grasp accuracy of 92.9%. The results are summarised in TABLE II.

In comparison with the two-stage baseline method as in [11], our method perform instance segmentation and instance-wise grasp synthesis *simultaneously* to synthesize grasp configuration in a single stage, with the accuracy of 92.9% and the inference speed of 39 frames per second – which has outperformed the baseline with a significant margin by 3.91% in accuracy and 25% in inference speed. Fig. 4 shows the results of two representative test samples from the OCID-Grasp dataset. To better support the advance of our proposed method, representative cases are show in Fig. 3 in which two-stage method [11]) failed while our proposed method succeeded.

TABLE III: Comparison of grasp accuracy ([%]) on JACQUARD Dataset [26], with different IoU thresholds and angle threshold of $30°$. Results for [11], [28], [29] are taken from [11]. Results of [6] are reproduced.

Method	IoU 25%	IoU 30%	IoU 35%
Method of [28]	81.95	78.26	74.33
Method of [29]	85.74	82.58	78.71
Mask-Grasp RCNN [17]	89.80	-	-
Method of [2]	91.5	89.7	87.3
Gr-ConvNet [6]	91.83	89.55	85.99
Det [11]	92.69	91.29	88.99
Det-Seg-Refine [11]	92.95	91.33	88.96
SSG(Ours)	**91.8**	**89.95**	**88.49**

B. Evaluation on JACQUARD Dataset

The **JACQUARD Dataset** is built on a subset of ShapeNet [30] which is a large CAD model dataset. It consists of 54485 different scenes from 11619 distinct objects. In total, it has over 4.9M grasp annotations (from over 1.1M

TABLE IV: Comparison of grasp accuracy ([%]) for JACQUARD Dataset [26], with different angle thresholds and IoU threshold of 25%. Results of [11], [28], [29] are referenced from [11], and results of [6] are reproduced.

Method	30°	25°	20°	15°	10°	5°
Method of [28]	81.95	81.76	81.27	80.23	77.79	-
Method of [29]	85.74	85.55	85.01	83.65	80.82	-
Mask-Grasp RCNN [17]	89.80	-	-	-	-	-
Gr-ConvNet [6]	91.83	90.00	87.34	83.45	77.94	63.67
Det [11]	92.68	92.34	92.08	91.40	88.12	56.23
Det-Seg-Refine [11]	92.95	92.88	92.42	91.52	88.12	72.79
SSG(Ours)	**91.8**	**91.11**	**90.05**	**87.97**	**81.68**	**60.87**

Fig. 5: Simulation and experiment setups.

unique locations). For each scenario, a render RGB image, a segmentation mask, two depth images and grasp annotations are provided.

However, the JACQUARD Dataset only contains single-object scenes without class labels for grasp annotations. Thus, we applied minimal adaptation of our method and make it a class-agnostic one, we labeled all objects as "object". Our model was trained on JACQUARD dataset in a class-agnostic way and evaluated using several metrics with different thresholds. Detailed results are summarised in TABLE IV and TABLE III (Unavailable results were denoted as "-").

The evaluation on the JACQUARD Dataset show that our method is generalized and can predict both high-quality instance masks and grasp masks for general objects without class-specific information. Despite the lack of class-specific information, our approach was very competitive among learning-based 2D grasp synthesis approaches. Further, we have conducted ablation study to support the importance of class-specific information (Details in Section V). Replacing the detection and segmentation heads with a class-agnostic one, such like [31], could be a potential way to boost the performance of our method on the JACQUARD Dataset.

We note that our proposed method, the SSG, significantly surpasses the Mask-Grasp RCNN [17] which is another single-stage instance-wise grasp synthesis method based on Mask-RCNN [18]. Our method has reached 91.8% grasp accuracy on the Jacquard dataset [26] which outperforms the Mask-Grasp RCNN [17] by 2%. Moreover, our method can run inference at 39FPS rate, which is almost three times faster than the Mask-Grasp RCNN [17] (14FPS).

C. Simulation and Real Robot Experiments

A set of simulations and real robot experiments have been conducted to validate that our model can be used to generate high-quality grasp candidates for robotic manipulators: (i) single object pick-and-place task in simulation; (ii) table cleaning task using a real robot.

Fig. 6: Real robot performing the table cleaning task in three different levels of difficulties: highly cluttered, cluttered and isolated real-world scenarios.

Our simulations and experiments focused on table top domains, where objects are in arbitrary spatial arrangements on the table. The simulation setup used a synthetic dataset from [32] which contains 90 simulated house-hold objects, imported from different resources, e.g., YCB dataset [33], Gazebo repository. The whole setup is composed of a dual arm robot with two UR5 manipulators and a Kinect sensor to acquire RGB-D images. For real robot experiments, we used exactly the same setup (see Fig. 5). We trained a model using OCID-Grasp dataset [11] which is used for both simulations and experiments.

In the first task, 16 objects, including 6 unseen objects (Juice Box, Cup, Spatula, Knife, Soap and Power Driller) have been selected. In each trial, one of them was randomly put on a table for 20 rounds of pick-and-place. A grasp configuration is considered successful if the object can be grasped, lifted up and placed at the designed place. The success rate for each object has been summarized in TABLE I.

The second task is focused on validating the proposed method on a real robot for table cleaning. In this task, an operator randomly places a set of unseen objects on the table and the robot should remove and place them into the predefined targets one by one. This task has been repeated in 3 different levels of difficulties: isolated (less than 3 objects), cluttered (less than 10 objects) and highly cluttered (more than 15 objects). A set of snapshots is shown in Fig. 6. We performed ten rounds of experiments per level, and assessed the performance by the success rate, where the attempt is considered as a success if the target object can be grasped and moved to the target successfully. The results showed that the robot is able to accomplish the task with success rates of 84.0%, 79.4% and 71.3%, respectively. It should be noted that in some failure cases in (highly) cluttered environments, although the grasp predictions were correct, execution of grasps were not feasible due to either the limitation of the motion planning or prevention of the grasp action in presence of the surrounding objects, rather than due to the grasp predictions themselves. The video of our experiments is available at https://youtu.be/riBXMgrupUw.

D. Scalability

The success of our proposed SSG shows the potential and scalability of feature assembling using linear coefficients. To

TABLE V: Ablation study on OCID-Grasp Dataset [11].

Model	SSG	SSG without instace segmentation	SSG without class prediction	SSG	SSG without instance segmentation	SSG without class prediction
Input Modalities	RGB	RGB	RGB	RGB+Depth	RGB+Depth	RGB+Depth
Grasp Accuracy	91.97%	90.92%	90.31%	92.93%	92.09%	90.81%

Fig. 7: Results of a set of tests in clutter environments on simulated and real objects.

further prove the scalability of our method, we re-train our model on Object Stacking Grasping Dataset (OSGD) [9], which includes additional affordance annotations from total 11 different types of grasping actions including cut, write, hammer, fork, shovel, wrench, pinch, screw, ladle, brush and hand-over. Here, the affordance annotation refers to the correct grasping action (e.g., knife – cut, screwdriver – screw, etc.).

For each object sample from OSGD dataset, its class label, bounding box, grasp annotations (in rectangles) and affordance annotations are provided. To generate additional affordance masks for each detected object, we add 11 extra heads in the object detection branch to predict 11 sets of coefficients to linearly assemble the shared prototype masks, and generate 11 target affordance masks. Since this dataset does not provide the instance semantic masks, our architecture is adopted accordingly: the global semantic segmentation head and the instance mask head are removed. Moreover, the OSGD dataset only provides depth image as input, thus the input channel of the feature extraction module is changed and no pre-trained model is used to initialize the feature extraction module.

As is shown in Fig. 8, correct affordance masks as well as grasp configurations are generated for different target objects which prove the scalability and extendability of the our proposed model. It can be extended to predict extra target masks by simply adding more coefficients predicting heads without changing the overall complexity. This feature of our methodology, in our opinion, will have an great impact on the field of robotic grasping synthesis research.

E. Ablation Study

A set of ablation study was conducted to support the current design. The detection head of our model is composed of three modules: object detection, instance segmentation, and generation of grasp maps. To validate the proposed network design, we have retrained and tested two additional models on the OCID-Grasp dataset [11]: (1) a model without predicting object class label; (2) a model without generating instance mask. The detailed results are shown in the TABLE V.

It can be seen from TABLE V that the depth channel brings useful information and increases the performance. The
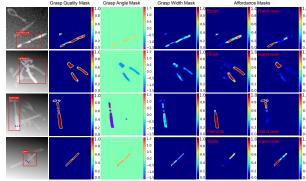

Fig. 8: Validation results on the OSGD dataset [9], further showing our method can generate additional *affordance masks* by predicting more sets of coefficients and assembling prototype masks with these coefficients.

instance segmentation module and the class prediction module also play an important role for grasp synthesis. Without instance segmentation head, the overall grasp accuracy of our model on OCID-Grasp dataset [11] decreases from 91.97% to 90.92% (with RGB input), and from 92.93% to 92.09% (with RGB-D input). Without class prediction head, the overall grasp accuracy of our model on OCID-Grasp dataset [11] decreases from 91.97% to 90.31% (with RGB input), and from 92.93% to 90.81% (with RGB-D input). The results of the ablation study has shown the benefits of our proposed network design: Generating different target masks by linearly assembling the same set of learned feature maps with different coefficients, which is able to exchange features across different domains, and also to make the learned feature maps more general and robust.

V. CONCLUSION

This work developed a novel single-stage grasp synthesis model – SSG – for tackling instance-wise grasp synthesis task in a single-stage manner. Our method formulated the instance-wise grasp synthesis as two sub-tasks: first, a set of learned feature embeddings is generated, which captures general features of the input RGB-D image; second, anchor-based object detection is conducted. For each detection, five sets of coefficients are predicted that will be used to linearly assemble generated feature embeddings to form a semantic instance mask and four grasp masks, simultaneously. We evaluated our method on the well-known JACQUARD dataset and a more challenging OCID-Grasp dataset. The results showed that our method outperforms the state-of-the-art on OCID-Grasp dataset and performs competitively on JACQUARD dataset. Moreover, the proposed method has been extensively tested both in simulation and on the real robot, using isolated, cluttered and highly cluttered scenarios. All these extensive results validated that our method can generate valid grasp configurations for target objects in multi-object scenarios.

References

[1] H. Karaoguz and P. Jensfelt, "Object detection approach for robot grasp detection," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 4953–4959.

[2] Y. Song, L. Gao, X. Li, and W. Shen, "A novel robotic grasp detection method based on region proposal networks," *Robotics and Computer-Integrated Manufacturing*, vol. 65, p. 101963, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0736584519308105

[3] H. Zhang, X. Lan, S. Bai, X. Zhou, Z. Tian, and N. Zheng, "Roi-based robotic grasp detection for object overlapping scenes," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 4768–4775.

[4] Z. Luo, B. Tang, S. Jiang, M. Pang, and K. Xiang, "Grasp detection based on faster region cnn," in *2020 5th International Conference on Advanced Robotics and Mechatronics (ICARM)*. IEEE, 2020, pp. 323–328.

[5] D. Morrison, P. Corke, and J. Leitner, "Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach," *arXiv preprint arXiv:1804.05172*, 2018.

[6] 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.

[7] H. Cao, G. Chen, Z. Li, J. Lin, and A. Knoll, "Lightweight convolutional neural network with gaussian-based grasping representation for robotic grasping detection," *arXiv preprint arXiv:2101.10226*, 2021.

[8] Y. Li, Y. Liu, Z. Ma, and P. Huang, "A novel generative convolutional neural network for robot grasp detection on gaussian guidance," *arXiv preprint arXiv:2205.04003*, 2022.

[9] H. Zhang, X. Lan, S. Bai, L. Wan, C. Yang, and N. Zheng, "A multi-task convolutional neural network for autonomous robotic grasping in object stacking scenes," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 6435–6442.

[10] C. Yang, X. Lan, H. Zhang, and N. Zheng, "Task-oriented grasping in object stacking scenes with crf-based semantic model," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 6427–6434.

[11] 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.

[12] T. Li, F. Wang, C. Ru, Y. Jiang, and J. Li, "Keypoint-based robotic grasp detection scheme in multi-object scenes," *Sensors*, vol. 21, no. 6, p. 2132, 2021.

[13] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesis—a survey," *IEEE Transactions on robotics*, vol. 30, no. 2, pp. 289–309, 2013.

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

[15] 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.

[16] H. Cheng, Y. Wang, and M. Q.-H. Meng, "A robot grasping system with single-stage anchor-free deep grasp detector," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–12, 2022.

[17] M. S. Kamel and M. D. Naish, "Mask-grasp r-cnn: Simultaneous instance segmentation and robotic grasp detection," in *2021 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI)*. IEEE, 2021, pp. 1–6.

[18] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[19] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "Yolact: Real-time instance segmentation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9157–9166.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[21] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[22] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[23] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Computer Vision, IEEE International Conference on*, vol. 3. IEEE Computer Society, 2003, pp. 1470–1470.

[24] X. Ren and D. Ramanan, "Histograms of sparse codes for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 3246–3253.

[25] S. Agarwal and D. Roth, "Learning a sparse representation for object detection," in *European conference on computer vision*. Springer, 2002, pp. 113–127.

[26] 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.

[27] M. Suchi, T. Patten, D. Fischinger, and M. Vincze, "Easylabel: A semi-automatic pixel-wise object annotation tool for creating robotic rgb-d datasets," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 6678–6684.

[28] X. Zhou, X. Lan, H. Zhang, Z. Tian, Y. Zhang, and N. Zheng, "Fully convolutional grasp detection network with oriented anchor box," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 7223–7230.

[29] A. Depierre, E. Dellandréa, and L. Chen, "Scoring graspability based on grasp regression for better grasp prediction," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 4370–4376.

[30] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An Information-Rich 3D Model Repository," Stanford University — Princeton University — Toyota Technological Institute at Chicago, Tech. Rep. arXiv:1512.03012 [cs.GR], 2015.

[31] C. Xie, Y. Xiang, A. Mousavian, and D. Fox, "Unseen object instance segmentation for robotic environments," *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1343–1359, 2021.

[32] H. Kasaei and S. Xiong, "Lifelong ensemble learning based on multiple representations for few-shot object recognition," *arXiv preprint arXiv:2205.01982*, 2022.

[33] O. Sener and S. Savarese, "Active learning for convolutional neural networks: A core-set approach," *arXiv preprint arXiv:1708.00489*, 2017.