2021 IEEE International Conference on Robotics and Automation (ICRA 2021)
May 31 - June 4, 2021, Xi'an, China

Residual Squeeze-and-Excitation Network with Multi-scale Spatial Pyramid Module for Fast Robotic Grasping Detection

Hu Cao [1,3], Guang Chen [2,3*], Zhijun Li[4], Jianjie Lin [1], Alois Knoll [1]

2021 IEEE International Conference on Robotics and Automation (ICRA) | 978-1-7281-9077-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICRA48506.2021.9561836

Abstract— **This paper proposes an efficient, fully convolutional neural network to generate robotic grasps by using 300×300 depth images as input. Specifically, a residual squeeze-and-excitation network (RSEN) is introduced for deep feature extraction. Following the RSEN block, a multi-scale spatial pyramid module (MSSPM) is developed to obtain multi-scale contextual information. The outputs of each RSEN block and MSSPM are combined as inputs for hierarchical feature fusion. Then, the fused global features are upsampled to perform pixel-wise learning for grasping pose estimation. The experimental results on Cornell and Jacquard grasping datasets indicate that the proposed method has a fast inference speed of $5ms$ while achieving high grasp detection accuracy of 96.4% and 94.8% on Cornell and Jacquard, respectively, which strikes a balance between accuracy and running speed. Our method also gets a 90% physical grasp success rate with a UR5 robot arm.**

I. Introduction

The goal of grasping detection is to find the appropriate grasp pose for the robot through the grasping object's visual information to provide reliable perception information for subsequent planning and control process and achieve a successful grasp. Grasp is a widely studied topic in the field of robotics, and the approaches used can be summarized as analytic methods and empirical methods. The analytical methods use mathematical and physical models in geometry, motion, and dynamics to carry out the calculation for grasping [1]. Its theoretical foundation is solid, but the deficiency lies in that the model between the robot manipulator and the grasping object in the real 3-dimensional world is complex. It is difficult to realize the model with high precision. In contrast, empirical methods do not strictly rely on real-world modeling methods, and some works utilize data information from known objects to build models to predict the grasping pose of new objects [2], [3], [4]. A new grasp representation is proposed in [5], where a simplified five-dimensional oriented rectangle grasp representation is used to replace the seven-dimensional grasp pose consisting of 3D location, 3D orientation, and the opening and closing distance of the plate gripper. Based on the oriented rectangles grasp configuration, the deep learning approaches can be successfully applied to the grasping detection task, which mainly includes classification-based methods, regression-based methods, and detection-based methods [6]. The comparison of the two grasp representations is presented in Fig. 1. Someone or two-stage deep learning methods [7], [8], [9] that have achieved great success in object detection have been modified to perform grasping detection tasks. For example, [10] refers to some key ideas of Faster RCNN [9] in the field of object detection to carry out robotic grasping from the input RGB-D images. In addition, other works, such as [11], [12], implemented high-precision grasp detection on Cornell grasping dataset [5] based on the one stage object detection method [7], [8]. Although these object detection-based methods achieve better accuracy in robotic grasping detection, their design based on the horizontal rectangular box is not suitable for the angular grasp detection task. Most of them have complex network structures, so it is difficult to achieve a good balance in detection accuracy and speed. This paper develops a lightweight residual squeeze-and-excitation network with a multi-scale spatial pyramid module for fast robotic grasping detection. Squeeze-and-excitation network (RSEN) is constructed by combining residual learning and channel attention mechanisms. Multiple dilated convolution with different rate parameters forms a multi-scale spatial pyramid module (MSSPM). After the input depth data is down-sampled, the hierarchical features are generated from RSEN and MSSPM. All these diverse features will be adaptively fused to obtain meaningful features to improve the network model's accuracy. The experimental results on two public datasets, Cornell and Jacquard, show that the proposed method can achieve fast running speed and high detection accuracy. Using a UR5 robot arm, we also obtain a 90% grasp success rate under the real environment.

II. Related Work

For 2D planar robotic grasping where the grasp is constrained in one direction, the methods can be divided into oriented rectangle-based grasp representation methods and contact point-based grasp representation methods. We will review the relevant works below.

A. Methods of oriented rectangle-based grasp representation

Classification-based methods: A first deep learning-based robotic grasping detection method is presented in [13]. The authors achieve excellent results by using a two-step cascaded structure with two deep networks. In [14], grasping proposals is estimated by sampling grasping locations and adjacent image patches. The grasp orientation is predicted by dividing the angle into 18 discrete angles. Since the grasping dataset is scant, a large simulation database called Dex-Net 2.0 is built in [15]. Based on Dex-Net 2.0, a Grasp-Quality

[*] Guang Chen is the corresponding author of this work

Authors Affiliation: [1]Chair of Robotics, Artificial Intelligence and Real-time Systems, Technische Universität München, München , Germany, [2]Tongji University, Shanghai, China, [3]State Key Laboratory of Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, [4]University of Science and Technology of China, China,

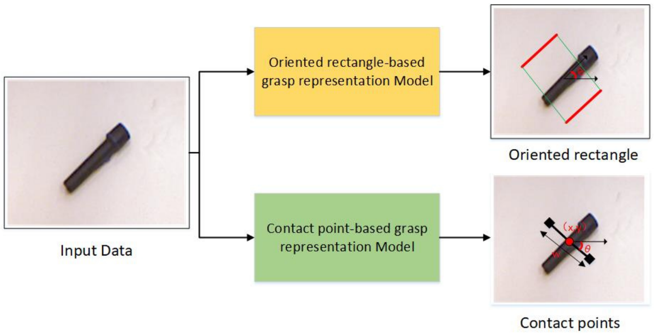
Oriented rectangle-based grasp representation Model
Contact point-based grasp representation Model
Input Data
Oriented rectangle
(x,y)
Contact points

Fig. 1: A comparison between the methods of oriented rectangle-based grasp representation and the methods of contact point-based grasp representation.

Convolutional Neural Network (GQ-CNN) is developed to classify the potential grasps. Although the network is trained on synthetic data, the proposed method still works well in the real world. Moreover, a classification-based robotic grasping detection method with spatial transformer network (STN) is proposed in [16]. The results of evaluating on Cornell grasping dataset indicate that their multi-stage STN algorithm performs well. The grasping detection method based on classification is more direct and reasonable.

Regression-based methods: Regression-based methods are to directly predict grasp parameters of location and orientation by training a model. A first regression-based single shot grasping detection approach is proposed in [17], in which the authors use AlexNet to extract features and achieve real-time performance by removing the process of searching potential grasps. Combing RGB and depth data, a multi-modal fusion method is introduced in [18]. With fusing RGB and depth features, the proposed method directly regresses the grasp parameters and improves the grasping detection accuracy on the Cornell grasping dataset. Similar to [18], the authors of [19] use ResNet as a backbone to integrate RGB and depth information and further improves the performance of grasping detection. Besides, a grasping detection method based on Region of Interest (ROI) is proposed in [4], which regressed grasp pose on ROI features and achieve better performance in object overlapping challenge scene. The regression-based method is effective, but its disadvantage is that it is more inclined to learn the ground truth grasps' mean value.

Detection-based methods: Many detection-based methods refer to some key ideas from object detection, such as anchor box. Based on the prior knowledge of these anchor boxes, the regression problem of grasping parameters is simplified. In [20], vision and tactile sensing are fused to build a hybrid architecture for robotic grasping. The authors use the anchor box to do axis-aligned, and grasp orientation is predicted by considering grasp angle estimation as a classification problem. The grasp angle estimation method used in [20] is extended by [10]. By transforming the angle estimation into a classification problem, the method of [10] achieves high grasping detection accuracy on the Cornell dataset based on FasterRCNN [9]. Different from the horizontal anchor box used in object detection, the authors of [21] specially design an oriented anchor box mechanism for grasping tasks and improve the performance of the model by combing end-to-end fully convolutional neural network. Moreover, [22] further extends the method of [21] and proposes a deep neural network architecture that performs better on the Jacquard dataset.

B. Methods of contact point-based grasp representation

The grasping representation based on the oriented rectangle is widely used in a robotic grasping detection task. Regarding the real plate grasping task, the gripper does not need so much information to perform the grasping action. A new simplified contact point-based grasping representation is introduced in [23], which consists of grasp quality, center point, oriented angle, and grasp width. Based on this grasping representation, GGCNN and GGCNN2 are developed to predict the grasping pose, and their methods achieve excellent performance in both detection accuracy and inference speed. Refer to [23], a fully convolutional neural network improves the grasping detection performance with pixel-wise way in [24]. Both [23] and [24] take depth data as input, and a generative residual convolutional neural network is proposed in [25] generate grasps, which take n-channel images as input. Recently, the authors of [26] take some ideas from image segmentation to perform three-finger robotic grasping detection. Similar to [26], an orientation attentive grasp synthesis (ORANGE) framework is developed in [27], which achieves better results on the Jacquard dataset based on the GGCNN and Unet model.

III. Method

A. Grasp representation

The grasping detection system should learn how to obtain the optimal grasp configuration for subsequent tasks for given RGB images or depth information of different objects. Many works, such as [20], [10], [21], are based on five-dimensional grasping representation to generate grasp pose,

$$g = (x, y, \theta, w, h) \tag{1}$$

where, (x, y) are the coordinates of the center point, θ represents the grasping rectangle's orientation, and the weight and height of the grasping rectangle are denoted by (w, h). A rectangular box is frequently used in object detection, but it is not suitable for grasping detection tasks. As the gripper's size is usually a known variable, a simplified representation is introduced in [23] for high-precision, real-time robotic grasping. The new grasping representation for the 3-D pose is defined as:

$$g = (\mathbf{p}, \varphi, w, q) \tag{2}$$

where the center point location in Cartesian coordinates is $\mathbf{p} = (x, y, z)$, φ and w are the gripper's rotation angle around the z axis and the opening and closing distance of the gripper, respectively. Since the five-dimensional grasping

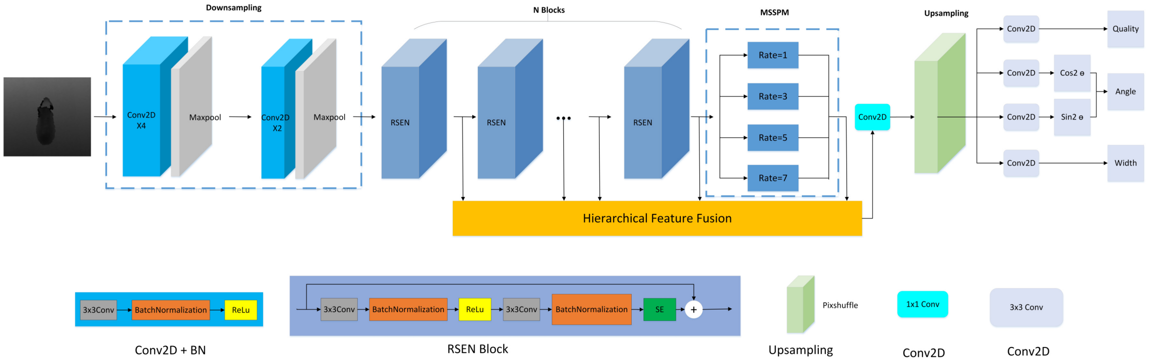

Fig. 2: The overall structure of the proposed lightweight fully convolutional neural network.

representation lacks the scale factor to evaluate the grasping quality, q is added to the new representation as a scale to measure the probability of grasp success. Besides, the definition of the new grasping representation in 2-D space can be described as:

$$\hat{g} = (\hat{p}, \hat{\varphi}, \hat{w}, \hat{q}) \tag{3}$$

where $\hat{p} = (u, v)$ represents the center point in the image coordinates, $\hat{\varphi}$ denotes the orientation in the camera frame, $\hat{w}$ and $\hat{q}$ still represent the opening and closing distance of the gripper and the grasp quality, respectively. When we know the calibration result of the grasping system, the grasp pose $\hat{g}$ can be converted to the world coordinates g by matrix operation,

$$\mathbf{g} = \mathbf{T}_{\text{RC}}(\mathbf{T}_{\text{CI}}(\hat{g})) \tag{4}$$

where $\mathbf{T}_{\text{RC}}$ and $\mathbf{T}_{\text{CI}}$ represent the transform matrices of the camera frame to the world frame and 2-D image space to the camera frame respectively. Moreover, the grasp map in the image space is denoted as:

$$\mathbf{G} = (\Phi, W, Q) \in \mathbb{R}^{3 \times W \times H} \tag{5}$$

where, each pixel in the grasp maps, Φ, W, Q, is filled with the corresponding $\hat{\varphi}, \hat{w}, \hat{q}$ values. In this way, it can be ensured that the center point coordinates in the subsequent inference process can be found by searching for the pixel value of the maximum grasp quality, $\hat{g^*} = \max_{\hat{Q}} \hat{G}$.

B. Network architecture

A lightweight fully convolutional neural network for robotic grasping detection is introduced, shown in Fig. 2. The input depth image passes through the downsampling block to extract features. A downsampling block is composed of 3x3 covolutional layer and 2x2 maxpooling layer with the formulation:

$$x_d = f_{\text{maxpool}}\left(f^k_{\text{conv}}\left(f^{k-1}_{\text{conv}}(\ldots f^0_{\text{conv}}(I)\ldots)\right)\right) \tag{6}$$

where I represent the input depth image, f_{conv} and f_{maxpool} denote the convolution filter and max-pooling filter, respectively. Two downsampling blocks are used to obtain the image features x_d. The first downsampling block consists of 4 convolutional layers (k=3) with the kernel size of 3x3 and 1 max-pooling layer with the kernel size of 2x2. The last downsampling block consists of 2 convolutional layers (k=1) with the kernel size of 3x3 and the same max-pooling layer.

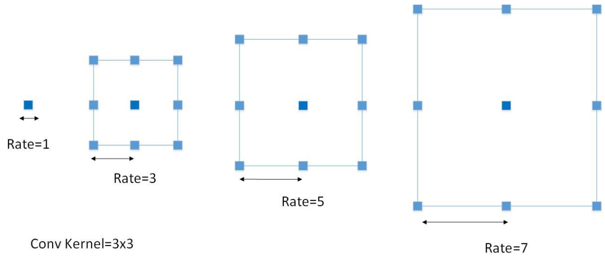

Fig. 3: Multi-scale spatial pyramid module: dilated convolution with kernel size of 3x3 and different rates.

To get more meaningful semantic features, we introduce a residual squeeze-and-excitation network (RSEN) to produce hierarchical features based on local skip connection, which include 2 convolution layers followed by a squeeze-and-excitation block [28] for enhancing hierarchical features at a channel-wise level. The computation of RSEN, f_{RSEN}, can be represented as follows:

$$\begin{aligned} M_0 &= f^0_{\text{RSEN}}(x_d) \\ &\vdots \\ M_n &= f^n_{\text{RSEN}}(M_{n-1}) \end{aligned} \tag{7}$$

where M represents the feature map and n denotes the number of extracted feature maps. 7 RSEN blocks (n=6) are applied to acquire enough relevant hierarchical feature maps in this work. The output M_6 of the RSEN block is fed into a multi-scale spatial pyramid module (MSSPM) for dilated convolution filter to extract multi-scale context information. These feature maps will be sent to a hierarchical feature fusion block to reduce the loss of information as the network deepens. The details will be described in subsection III-C and III-D. Furthermore, the pixelshuffle [29], $f_{\text{pixelshuffle}}$, is adopted to perform upsampling for the fused features. The output can be expressed as:

$$x_u = f_{\text{pixelshuffle}}(x_{\text{fuse}}) \tag{8}$$

where, x_u and x_{fuse} denote the upsampled features and fused features, respectively. Final network layer is composed of 4 task-specific convolutional filters ($f^0_{\text{conv}}, f^1_{\text{conv}}, f^2_{\text{conv}}, f^3_{\text{conv}}$) with kernel size of 3x3. The output is given as Eq. 9,

$$\begin{aligned} g_q &= \max_q f^0_{\text{conv}}(x_u), \\ g_{\cos(2\theta)} &= \max_q f^1_{\text{conv}}(x_u), \\ g_{\sin(2\theta)} &= \max_q f^2_{\text{conv}}(x_u), \\ g_w &= \max_q f^3_{\text{conv}}(x_u), \end{aligned} \tag{9}$$

where the position of the center point is the pixel coordinates of the largest grasp quality g_q, the opening and closing distance of the gripper is g_w, and the grasp angle can be computed by $g_{\text{angle}} = \arctan(\frac{g_{\sin(2\theta)}}{g_{\cos(2\theta)}})/2$.

C. Multi-scale spatial pyramid module

Due to max-pooling and the deepening of the network, the spatial information of the input data is lost gradually. To overcome this problem, a dilated convolution is introduced in [30] to enhance the receptive field of convolution operation without adding any training parameters. Considering one-dimensional data as input, dilated convolution can be formulated as:

$$y[i] = \sum_{j=1}^{k} w[j]x[i + r \cdot j] \tag{10}$$

where x is input signal, $w[j]$ is a kernel filter with size of j, and r represents the rate parameter. In Fig. 3, the concept of dilated convolution in two dimensional is presented. We use a combination of 4 different dilated convolutions with rate parameter from [1,3,5,7] to obtain more discriminative features. The multi-scale spatial pyramid module can effectively control the receptive field of view by applying this mechanism.

D. Hierarchical feature fusion

The deepening of the network decrease the ability of spatial expression to extract the features, which can be remedied by a semantic information [31]. Therefore, we proposed a simple hierarchical feature fusion architecture to fully utilization of the feature produced from RSEN and MSSPM with:

$$x_{\text{fuse}} = w \times [M_0, M_1, ..., M_n, x_{\text{msspm}}] + b \tag{11}$$

where w and b are the weight parameters of the convolution filter with kernel size of 1x1. x_{msspm} denotes the output of multi-scale spatial pyramid module (MSSPM), and $[M_0, M_1, ..., M_n, x_{\text{msspm}}]$ represents the concatenation operation. Using this fusion method, we can extract the useful information by adaptive learning and suppress the redundant information. Therefore the number of feature channels can be reduced effectively.

E. Loss function

For a dataset including grasping objects $O = \{O_1...O_n\}$, input images $I = \{I_1...I_n\}$, and corresponding grasp labels $L = \{L_1...L_n\}$, We propose a lightweight fully convolutional neural network to approximate the complex function $F : I \longmapsto \hat{G}$, where F represents a neural network model with weighted parameters, I is input image data, and $\hat{G}$ denotes grasp prediction. We train our model to learn the mapping function F by optimizing the minimum error between grasp prediction $\hat{G}$ and the corresponding label L. In this work, we consider the grasp pose estimation as a regression problem. Therefore the Smooth L1 loss is used as our regression loss function. The loss function $Loss$ is defined as :

$$Loss(\hat{G}, L) = \sum_i^N \sum_{m \in \{q,\theta,w\}} \text{Smooth}_{L1}(\hat{G}^m_i - L^m_i) \tag{12}$$

with Smooth_{L1}:

$$\text{Smooth}_{L1}(x) = \begin{cases} (\sigma x)^2/2, & \text{if } |x|<1; \\ |x| - 0.5/\sigma^2, & \text{otherwise.} \end{cases}$$

where N is the number of grasp candidates, q, w represents the grasp quality and the opening and closing distance of the gripper, respectively, and $(\cos(2\theta), \sin(2\theta))$ is the form of orientation angle θ. In the Smooth_{L1} function, σ is the hyperparameter that controls the smooth area, and it is set to 1 in this work.

IV. Experiments

Following the previous works [4], [10], [25], [32], we use the rectangle metric as an accuracy metric to evaluate our robotic grasping detection method. Specifically, the proposed model is validated on two public grasping datasets, Cornell [5] and Jacquard [33]. The experimental results show that the proposed algorithm can achieve high prediction accuracy and fast inference speed with only the depth data as input.

A. Dataset

In Tab. I, it presents a summary of these grasping datasets. We choose two of them as the benchmark to evaluate our model. The details are as follows:

Cornell grasping dataset: The Cornell dataset was collected in the real world with the RGB-D camera. The dataset is composed of 885 images with a resolution of 640×480 pixels of 240 different objects with positive grasps (5110) and negative grasps (2909). RGB images and corresponding point cloud data of each object with various poses are provided. However, the Cornell dataset scale is small for training our convolutional neural network model. We use online data augment methods in this work, including random cropping, zooms, and rotation, to extend the dataset to avoid overfitting during training.

Jacquard grasping dataset: Jacquard is a large grasping dataset created through simulation based on CAD models.

TABLE I: Description of the public Grasping Datasets

Dataset	Modality	Objects	Images	Grasps
Dexnet	Depth	1500	6.7M	6.7M
Cornell	RGB-D	240	885	8019
Jacquard	RGB-D	11K	54K	1.1M

TABLE II: Detection accuracy (%) on Cornell dataset

Author	Input Modality	Accuracy(%)	Time (ms)
Jiang [5]	RGB-D	60.5	5000
Lenz [13]	RGB-D	73.9	1350
Karaoguz [34]	RGB	88.7	200
Chu [10]	RGD	96.0	120
Zhang [18]	RGB-D	88.9	117
Wang [35]	RGB-D	85.3	140
Redmon [17]	RGB-D	88.0	76
Kumra [25]	D	95.4	-
Asif [36]	RGB-D	90.6	24
Morrison [23]	D	73.0	-
Zhang [4]	RGB	93.6	40
Song [37]	RGB	96.2	-
Wang [38]	D	94.4	8
Ours	D	**96.4**	5

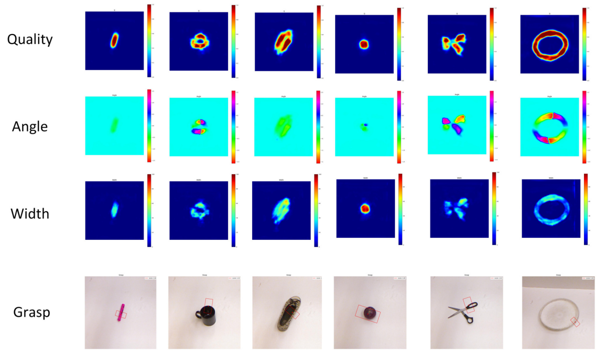

Fig. 4: The selected detection outputs of the propsed model on Cornell dataset (best visualization in color).

Because no manual collection and annotation are required, the Jacquard dataset is larger than the Cornell dataset, containing 50k images of 11k objects and over 1 million grasp labels. Since the Jacquard dataset is large enough, we do not use any data augmentation methods to it.

B. Implementation Details

Each input image is scaled to the size of 300×300 before being fed into the network. Meanwhile, the corresponding grasp labels are encoded for training and learning. Specifically, we use the form of $\sin(2\theta)$ and $\cos(2\theta)$ to represent the gripper's angle and width to represent the opening and closing distance of the gripper. The center coordinate of the grasp box is obtained by searching for the position of the maximal grasp quality, where the pixel value of the corresponding area is set to 1, and the other pixels are set to 0. The proposed model is trained on an Nvidia RTX2080Ti GPU with an Adam optimizer where the batch size is set to 8, and the initial learning rate is set to 0.001. Moreover, our grasping algorithm is implemented by Pytorch 1.2.0.

TABLE III: Detection accuracy (%) on Jacquard dataset

Author	Input Modality	Accuracy(%)
Depierre [33]	RGB-D	74.2
Zhou [21]	RGD	92.8
Zhang [4]	RGD	93.6
Morrison [23]	D	84
Song [37]	RGD	93.2
Kumra [25]	D	93.7
Ours	D	**94.8**

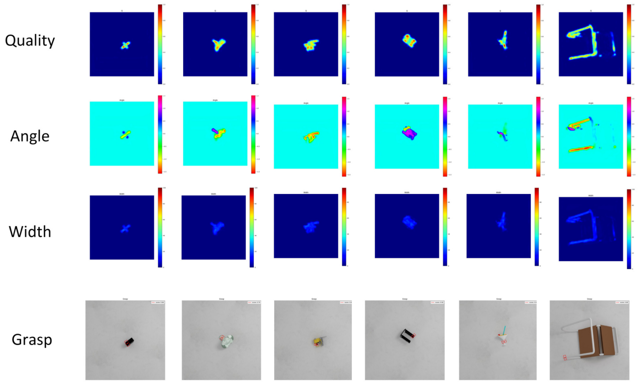

Fig. 5: The selected detection outputs of the proposed model on Jacquard dataset (best visualization in color).

C. Experimental results on Cornell dataset

We train our network on the Cornell grasping dataset, and the results are presented in Tab. II. Using only the depth data as input, our method achieves an accuracy of 96.4% and a fast inference speed of 5ms. Tab. II demonstrates that the proposed grasping detection algorithm strikes a better balance between accuracy and speed without using color information, which makes it more suitable for real-time applications, compared with other competitive methods. Fig. 4 illustrates some selected grasping objects. The map of grasp quality, angle, and width is displayed in the first three rows. The grasp candidate with the best grasp quality is chosen, which is visualized in the last row. The results demonstrate that our method can effectively predict different types of objects.

D. Experimental results on Jacquard dataset

For the Jacquard dataset, we adopt a different training strategy. Since the Jacquard dataset's grasp labels are very dense, and many of them overlap each other, we maximize the grasp quality of the center point coordinate and gradually reduce the value farther away from the center point to improve the robustness of network learning. In Tab. III, the experimental results indicate that the proposed model achieves better detection accuracy than existing algorithms on the Jacquard dataset. The selected detection results are visualized in Fig. 5.

TABLE IV: Comparison of network size and execution time for different methods

Author	Parameters (Approx.)	Time
Lenz [13]	-	13.5s
Pinto and Gupta [14]	60 M	-
Levine [39]	1 M	0.2-0.5s
Johns [40]	60 M	-
Morrison [23]	66 k	-
Ours	2.94 M	5ms

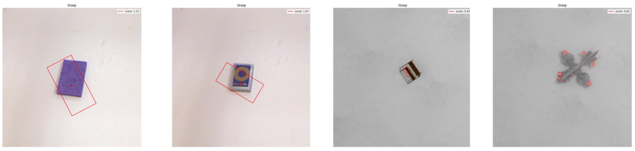

Fig. 6: Failed detection cases: the first two cases from Cornell dataset and the last two cases from Jacquard dataset (best visualization in color).

E. Comparison of network parameter sizes

To make the designed network more suitable for real-time application, we develop a fully convolutional neural network with a small parameter size. The comparison of parameter size and execution time for different methods is summarized in Table. IV. It is difficult for the existing deep learning methods to meet the requirements of speed and accuracy simultaneously. [23] has fewer parameters but less accuracy, other works, such as [13], [10], [14], improve accuracy while the running speed decrease and the number of parameters increase. Our model achieves high detection accuracy and fast running speed with a parameter size of 2.94M.

F. Failure cases analysis

During the experiment, it is found that although the proposed algorithm achieves high detection accuracy, it is still not effective in some cases. In Fig. 6, some failure examples are presented. The proposed model does not predict the orientation of larger rectangular objects in the Cornell dataset very well and does not have good generalization ability for some objects with more complex shapes in the Jacquard dataset. However, these deficiencies can be effectively alleviated by increasing the amount and diversity of training data.

G. Evaluation on real robot

TABLE V: The results of physical grasping experiment.

Objects	Detected	Physical grasp
Welding gun	9/10	9/10
Mouse	8/10	6/10
Pencil	10/10	10/10
Brush	10/10	8/10
Griper finger	10/10	10/10
Staples box	10/10	8/10
Remote control	10/10	9/10
Knife	10/10	10/10
Eraser	10/10	10/10
Hammer	10/10	10/10
Average	**97%**	**90%**

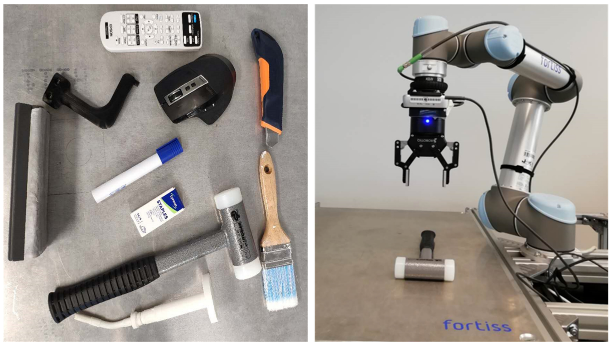

Fig. 7: Robot experiment setup with an UR5 robot arm

We evaluate our proposed approach on a Universal Robot (UR5) attached with a Robotic gripper2F-85 and an Intel Realsense camera. We mounted the camera orthogonal to the workstation to satisfy the constraints of a 2D plane. Therefore we have a constant value in the z-direction. Furthermore, we employ the Real-Time Data Exchange (RTDE) interface from UR5 with an update rate of 8ms to enable real-time grasping. The robotic gripper and UR5 are communicated to each other via the OPC-UA platform. The whole setup is illustrated in Fig. 7. The experiment pipeline is described as the proposed approach predicts the grasp configuration in the camera image coordinate. Then it will be transferred to the robot world coordinate. We apply the inverse kinematics to obtain the corresponding joint position. Sequentially, a point-to-point motion is executed to approach the desired position. In the end, the robotic gripper with the predicted configuration grasps the objects. We evaluate our approach with 10 novel objects, and each object is randomly placed on the table with 10 random positions. The result in Tab. V demonstrates that the proposed algorithm can achieve a 90% grasp success rate.

V. Conclusion

We propose a lightweight residual squeeze-and-excitation network with a multi-scale spatial pyramid module for fast robotic grasping detection in this work. Through effective network design, our model achieves a fast inference speed of 5ms with a parameter number of 2.94M. Furthermore, the proposed grasping algorithm achieves detection accuracy of 96.4% and 94.8% respectively on Cornell and Jacquard grasping datasets. Our method also achieves a success rate of 90% by using the UR5 robot arm in the physical grasping experiment.

VI. Acknowledgement

This research was funded by the open project of State Key Laboratory of Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences under Grant No.20190109.

References

[1] A. Bicchi and V. Kumar, “Robotic grasping and contact: a review,” in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, vol. 1, 2000, pp. 348–353 vol.1.

[2] Y. Inagaki, R. Araki, T. Yamashita, and H. Fujiyoshi, “Detecting layered structures of partially occluded objects for bin picking,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 5786–5791.

[3] A. Gariépy, J.-C. Ruel, B. Chaib-draa, and P. Giguère, “Gq-stn: Optimizing one-shot grasp detection based on robustness classifier,” *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3996–4003, 2019.

[4] 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)*, 2019, pp. 4768–4775.

[5] Yun 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*, 2011, pp. 3304–3311.

[6] G. Du, K. Wang, and S. Lian, “Vision-based robotic grasping from object localization, pose estimation, grasp detection to motion planning: A review,” *CoRR*, vol. abs/1905.06658, 2019.

[7] J. Redmon and A. Farhadi, “YOLO9000: better, faster, stronger,” *CoRR*, vol. abs/1612.08242, 2016.

[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single shot multibox detector,” 2015, cite arxiv:1512.02325Comment: ECCV 2016.

[9] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 91–99.

[10] F. Chu, R. Xu, and P. A. Vela, “Deep grasp: Detection and localization of grasps with deep neural networks,” *CoRR*, vol. abs/1802.00520, 2018.

[11] G. Wu, W. Chen, H. Cheng, W. Zuo, D. Zhang, and J. You, “Multi-object grasping detection with hierarchical feature fusion,” *IEEE Access*, vol. 7, pp. 43 884–43 894, 2019.

[12] Y. S. Dongwon Park, Y. S. Se Young Chun Dongwon Park, and S. Y. Chun, “Real-time, highly accurate robotic grasp detection using fully convolutional neural networks with high-resolution images,” *CoRR*, vol. abs/1809.05828, 2018, withdrawn.

[13] 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.

[14] L. Pinto and A. Gupta, “Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 3406–3413.

[15] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, “Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics,” *CoRR*, vol. abs/1703.09312, 2017.

[16] D. Park and S. Y. Chun, “Classification based grasp detection using spatial transformer network,” *CoRR*, vol. abs/1803.01356, 2018.

[17] J. Redmon and A. Angelova, “Real-time grasp detection using convolutional neural networks,” in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. Seattle: IEEE, July 2015.

[18] Zhang, Qiang, Qu, Daokui, Xu, Fang, and Zou, Fengshan, “Robust robot grasp detection in multimodal fusion,” *MATEC Web Conf.*, vol. 139, p. 00060, 2017.

[19] S. Kumra and C. Kanan, “Robotic grasp detection using deep convolutional neural networks,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 769–776.

[20] D. Guo, F. Sun, H. Liu, T. Kong, B. Fang, and N. Xi, “A hybrid deep architecture for robotic grasp detection,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 1609–1614.

[21] X. Zhou, X. Lan, H. Zhang, Z. Tian, Y. Zhang, and N. Zheng, “Fully convolutional grasp detection network with oriented anchor box,” *CoRR*, vol. abs/1803.02209, 2018.

[22] A. Depierre, E. Dellandréa, and L. Chen, “Optimizing correlated graspability score and grasp regression for better grasp prediction,” 2020.

[23] 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.

[24] S. Wang, X. Jiang, J. Zhao, X. Wang, W. Zhou, and Y. Liu, “Efficient fully convolution neural network for generating pixel wise robotic grasps with high resolution images,” *CoRR*, vol. abs/1902.08950, 2019.

[25] S. Kumra, S. Joshi, and F. Sahin, “Antipodal robotic grasping using generative residual convolutional neural network,” 2019.

[26] D. Wang, “Sgdn: Segmentation-based grasp detection network for unsymmetrical three-finger gripper,” 2020.

[27] N. Gkanatsios, G. Chalvatzaki, P. Maragos, and J. Peters, “Orientation attentive robot grasp synthesis,” 2020.

[28] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7132–7141.

[29] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, “Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network,” *CoRR*, vol. abs/1609.05158, 2016.

[30] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, “Encoder-decoder with atrous separable convolution for semantic image segmentation,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[31] J. Li, F. Fang, K. Mei, and G. Zhang, “Multi-scale residual network for image super-resolution,” in *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham: Springer International Publishing, 2018, pp. 527–542.

[32] B. Li, H. Cao, Z. Qu, Y. Hu, Z. Wang, and Z. Liang, “Event-based robotic grasping detection with neuromorphic vision sensor and event-grasping dataset,” *Frontiers in Neurorobotics*, vol. 14, p. 51, 2020. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnbot.2020.00051

[33] A. Depierre, E. Dellandréa, and L. Chen, “Jacquard: A large scale dataset for robotic grasp detection,” *CoRR*, vol. abs/1803.11469, 2018.

[34] 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.

[35] Z. Wang, Z. Li, B. Wang, and H. Liu, “Robot grasp detection using multimodal deep convolutional neural networks,” *Advances in Mechanical Engineering*, vol. 8, no. 9, p. 1687814016668077, 2016.

[36] U. Asif, J. Tang, and S. Harrer, “Graspnet: An efficient convolutional neural network for real-time grasp detection for low-powered devices,” in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*. International Joint Conferences on Artificial Intelligence Organization, 7 2018, pp. 4875–4882.

[37] 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.

[38] S. Wang, X. Jiang, J. Zhao, X. Wang, W. Zhou, and Y. Liu, “Efficient fully convolution neural network for generating pixel wise robotic grasps with high resolution images,” *CoRR*, vol. abs/1902.08950, 2019.

[39] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, “Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection,” *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 421–436, 2018.

[40] E. Johns, S. Leutenegger, and A. J. Davison, “Deep learning a grasp function for grasping under gripper pose uncertainty,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 4461–4468.