

# Grasp Detection with Hierarchical Multi-Scale Feature Fusion and Inverted Shuffle Residual

Wenjie Geng<sup>†</sup>, Zhiqiang Cao<sup>\*</sup>, Peiyu Guan<sup>†</sup>, Fengshui Jing, Min Tan, and Junzhi Yu

**Abstract:** Grasp detection plays a critical role for robot manipulation. Mainstream pixel-wise grasp detection networks with encoder-decoder structure receive much attention due to good accuracy and efficiency. However, they usually transmit the high-level feature in the encoder to the decoder, and low-level features are neglected. It is noted that low-level features contain abundant detail information, and how to fully exploit low-level features remains unsolved. Meanwhile, the channel information in high-level feature is also not well mined. Inevitably, the performance of grasp detection is degraded. To solve these problems, we propose a grasp detection network with hierarchical multi-scale feature fusion and inverted shuffle residual. Both low-level and high-level features in the encoder are firstly fused by the designed skip connections with attention module, and the fused information is then propagated to corresponding layers of the decoder for in-depth feature fusion. Such a hierarchical fusion guarantees the quality of grasp prediction. Furthermore, an inverted shuffle residual module is created, where the high-level feature from encoder is split in channel and the resultant split features are processed in their respective branches. By such differentiation processing, more high-dimensional channel information is kept, which enhances the representation ability of the network. Besides, an information enhancement module is added before the encoder to reinforce input information. The proposed method attains 98.9% and 97.8% in image-wise and object-wise accuracy on the Cornell grasping dataset, respectively, and the experimental results verify the effectiveness of the method.

**Key words:** grasp detection; hierarchical multi-scale feature fusion; skip connections with attention; inverted shuffle residual

## 1 Introduction

In recent years, tremendous progresses have been made towards high-quality image understanding, including object detection<sup>[1]</sup>, segmentation<sup>[2]</sup>, inpainting<sup>[3, 4]</sup>, grasp detection, etc. As a crucial component in robotic manipulation, grasp detection can provide the suitable

grasping position of the target object in the image, which has received much attention.

Earlier studies usually infer grasp detection results in a geometrical way based on point cloud of the target object, which can be categorized into principal axis based, algebraic expression, and template matching methods. The first type follows the procedure of

• Wenjie Geng, Zhiqiang Cao, Peiyu Guan, Fengshui Jing, and Min Tan are with the State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China. E-mail: {gengwenjie2017, zhiqiang.cao, guanpeiyu2017, fengshui.jing, min.tan}@ia.ac.cn.

• Junzhi Yu is with the Department of Advanced Manufacturing and Robotics, College of Engineering, Peking University, Beijing 100871, China. E-mail: yujunzhi@pku.edu.cn.

<sup>†</sup> Wenjie Geng and Peiyu Guan contribute equally to this work.

\* To whom correspondence should be addressed.

Manuscript received: 2022-09-02; revised: 2023-01-07; accepted: 2023-01-15

clustering-then-detecting, which first clusters the point cloud of each object to calculate its center and principal axis based on Principal Component Analysis (PCA), and then gets the proper grasp position and direction<sup>[5, 6]</sup>. The second solution achieves grasp detection with the surface properties of object point cloud such as surface equation<sup>[7]</sup> and local curvature<sup>[8]</sup>. The above two solutions are susceptible to noise, and some researchers attempt to analyze grasp detection by template matching on the basis of a pre-built grasp template library<sup>[9, 10]</sup>. Nevertheless, the template library might lead to poor generalization.

With the development of deep learning<sup>[11–13]</sup>, more researches focus on grasp detection based on Convolution Neural Network (CNN). A representative solution resorts to convolution layers followed by fully connected layers to predict the grasp quality<sup>[14, 15]</sup> or regress the grasp results<sup>[16, 17]</sup>. However, since the feature map needs to be flattened to a vector with fixed length before fully connected operation, the input image is preferred to a fixed size. Such a case prevents the flexibility of the network with higher computational cost. In order to adapt the network to different input sizes, some works adopt Region Of Interest (ROI) pooling layer to pool features of different sizes into the same size<sup>[18, 19]</sup>. Due to the fact that the proposal network needs to be constructed before ROI pooling layer, the network complexity is increased. Still, the problem of computation burden is unsolved. To solve the drawback from fully connected layers, the solution of Fully Convolutional Network (FCN) is concerned. It simplifies the network structure with the adaptability to arbitrary image sizes. Compared with the FCN-based grasp detection that generates feature maps in a decreasing way in size<sup>[20–22]</sup>, the grasp detection network based on encoder-decoder structure is more attractive and valuable as fine-grained features which are obtained by the upsampling deconvolution operation with good accuracy and efficiency<sup>[23–25]</sup>. In this way, a pixel-wise grasp detection is achieved.

It is noted that the existing encoder-decoder-based methods mainly concern the high-level feature in the encoder and low-level features are ignored. As a matter of fact, low-level features contain abundant local detailed information. The fusion of low-level and high-level features shall no doubt benefit the grasp detection. Skip connection in the field of object detection and segmentation provides a preferable way to combine multi-level features. Sistu et al.<sup>[26]</sup> implemented skip

connections by connecting the corresponding feature layers with the same spatial scales from encoder and decoder. A problem of this connection relationship is that the information interaction among features of different scales is not considered, which affects the system performance. How to design the skip connections with multi-scale feature interaction deserves further investigation. Moreover, as the object detection methods SSD<sup>[27]</sup> and FPN<sup>[28]</sup> point out, high-level feature possesses a large receptive field with rich semantic information. However, if the high-level feature of the encoder is directly transmitted to the decoder for grasp prediction, it may be lack of the information transmission among different channels in high-level feature. Effective mining of channel information of high-level feature is beneficial to improve the quality of grasp detection. The aforementioned analyses motivate us to build advanced encoder-decoder network. For this paper, the main contributions are as follows:

- (1) A novel grasp detection network with hierarchical multi-scale feature fusion and Inverted Shuffle Residual (ISR) is proposed, and it attains good accuracy and efficiency.

- (2) Multi-scale features from different convolution layers in the encoder are fused by our Skip Connections with Attention (SCA) module. In SCA, detailed information from low-level features of the encoder is effectively exploited. The resultant features are further processed with the secondary feature fusion in the decoder for better grasp prediction.

- (3) The ISR module is designed, where the high-level feature from the encoder is split into two features in channel, and one of them is projected to a high-dimensional space to acquire more discriminative information. By processing these two split features in a differentiated way, more high-dimensional channel information is mined and retained.

- (4) The experiments on the Cornell grasping dataset, the robustness verification under interferences, and grasp detection in an actual scene prove the effectiveness of the proposed method.

This paper is organized as follows. Section 2 gives the related work. Section 3 presents the proposed grasp detection method in detail. Experiments are presented in Section 4, and Section 5 concludes this paper.

## 2 Related Work

This section discusses the grasp detection methods from three aspects: traditional geometry, grasp detection with

fully connected layer, and grasp detection with fully convolutional network.

### 2.1 Traditional geometry methods

The principal axis based solution is popular in traditional grasp detection. Suzuki and Oka<sup>[5]</sup> extracted the planar surface and the object by randomly sample the consensus, and then the center point and the principal axis of object are computed via PCA. Further, the grasp direction is calculated by cross product between the direction of the principal axis of object and the normal of the plane. Zapata-Impata et al.<sup>[6]</sup> computed a cutting plane whose normal vector is parallel to the principal axis of object. On this basis, sub-cloud of object is generated within a certain distance to the cutting plane, and the best grasping points are chosen according to point curvature in sub-cloud, antipodal configuration, and perpendicular grasp constraint. Another scheme is to model object using the algebraic expression. Vezzani et al.<sup>[7]</sup> modeled the object and the graspable volume of the hand with superquadric functions for grasp detection. Gori et al.<sup>[8]</sup> matched local curvature of object to the surface of the robot's palm, and a score function is designed to measure the quality of graspable points on the object surface. Besides, Li and Pollard<sup>[9]</sup> treated grasping as a shape matching problem, where a grasp database with object models and corresponding hand poses is pre-built, and then the features of a tested object are compared with those of models in the database to identify candidate grasps. Herzog et al.<sup>[10]</sup> constructed a grasp template library composed of local shape descriptors of objects and corresponding grasp configurations through kinesthetic teaching, and a good grasp configuration is acquired by matching.

### 2.2 Grasp detection with fully connected layer

With the fully connected layers, Mahler et al.<sup>[14]</sup> proposed a grasp quality CNN to predict the probability of successful grasps from depth image. A two-step cascaded grasp detection method is presented in Ref. [15], which first produces candidate rectangles from an RGB-D image using a small deep network, and then the corresponding raw features, including color, depth images, and surface normals of rectangles, are inputted to a relatively large deep network to score each rectangle. Finally, the top-ranked one is chosen. Different from the scoring evaluation on candidate grasps<sup>[14, 15]</sup>, some researches directly regressed the grasp. Redmon and Angelova<sup>[16]</sup> presented a regression grasp model to predict both the category of the object and

the corresponding grasp result. Meanwhile, considering the drawback of average effect, an improved MultiGrasp network is designed to predict a reasonable grasp by dividing the image into grids. In Ref. [17], two ResNet-50 modules are utilized in parallel to extract features from the inputs of RGB and depth data. After the results are merged, they are processed by two fully connected layers for grasp configuration prediction. Constrained by the fixed size of input image, the flexibility of the above-mentioned methods is weak. To solve this problem, ROI pooling layer is embedded<sup>[18, 19]</sup>. Karaoguz and Jensfelt<sup>[18]</sup> proposed a Grasping Rectangle Proposal Network (GRPN) with the rotated region proposals to detect the grasp rectangles and the corresponding probabilities, and the predicted grasping result with the highest probability is used for robot manipulation. An ROI of interest-based Grasping Detection method (ROI-GD) is proposed in Ref. [19]. It first provides object bounding box proposals by an ROI generator from the input RGB image, and then uses features from ROIs to generate grasp candidates and confidence scores. After non-maximum suppression, the best grasp candidate is determined.

### 2.3 Grasp detection with fully convolutional network

One implementation based on FCN is to capture feature in deep layer and generate grasp results, where the sizes of feature maps decrease gradually. Satish et al.<sup>[20]</sup> proposed a fully convolutional grasp quality CNN. Zhou et al.<sup>[21]</sup> proposed an end-to-end fully convolutional network with the oriented anchor box mechanism to predict an accurate grasp for a parallel-plate robotic gripper. A convolutional neural network combined with regression and classification was presented in Ref. [22], which employs atrous convolution to improve local expression ability of features. In these methods, coarse-grained feature is mainly used and fine-grained feature is deficient. To tackle this issue, grasp detection networks based on the encoder-decoder structure are proposed. A pioneering work is the Generative Grasping Convolutional Neural Network (GG-CNN) proposed by Morrison et al.<sup>[23]</sup>, which adopts cascaded convolutional layers and transposed convolutional layers in encoder and decoder, respectively. This network directly generates a grasp pose and grasp quality for every pixel with a small number of parameters. Afterwards, a series of improvements emerge. Kumra et al.<sup>[24]</sup> designed a Generative Residual Convolutional

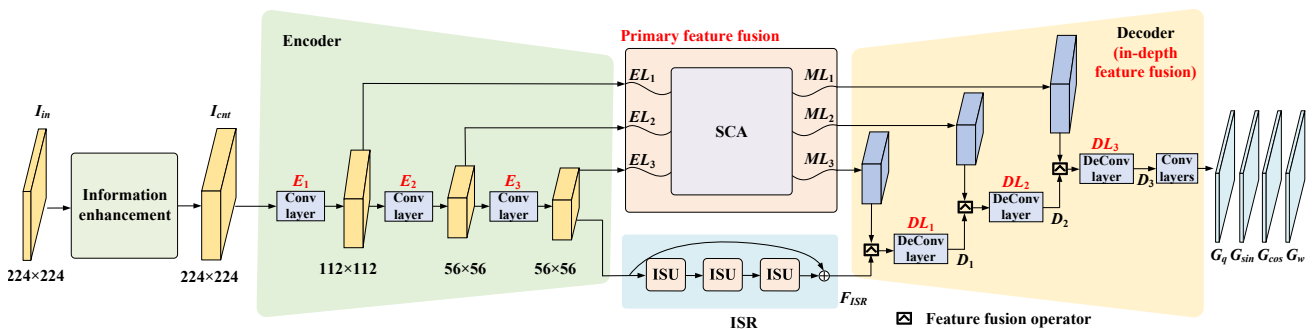
neural Network (GR-ConvNet) to output the grasp result of each pixel, where residual layers are inserted between encoder and decoder to avoid the problem of vanishing gradients. Yu et al.<sup>[25]</sup> employed the improved BlitzNet to simultaneously detect and segment the target object, and then the depth and grayscale maps of target object are inputted to the Two-stream Grasping convolutional neural Network (TsGNet), where TsGNet adopts depthwise separable convolution in the encoder and global deconvolution network in the decoder for better feature expression. In these encoders, high-level feature is chosen for subsequent deconvolution and low-level features are neglected. In this paper, low-level and high-level features are fused by skip connections with attention. Besides, the channel information in high-level feature is also enriched by the inverted shuffle residual.

### 3 Method

Figure 1 presents the structure of the grasping detection network with hierarchical multi-scale feature fusion and ISR, which is termed as HMISR. The proposed network originates from the encoder-decoder structure with three new modules: information enhancement, SCA, and ISR. The information enhancement module is added before the encoder to capture the context information with different receptive fields for the enhanced feature representation. SCA and ISR modules are inserted between encoder and decoder to refine features in both spatial and channel dimensions. In view of the flexibility in handling information of attention mechanism<sup>[29, 30]</sup>, SCA module adopts channel attention to fuse the feature maps from different layers in the encoder. In this way, low-level features are effectively explored. ISR mainly

focuses on mining the channel information of the output of the last layer from the encoder, which enhances the information flow among different channels in high-level feature with good semantic representation ability. On the basis of the results of SCA and ISR, a customized decoder is designed to further process these multi-scale feature maps for grasp prediction. It is worth mentioning that multi-scale features are fused hierarchically in SCA and decoder. In addition, the input of the network can be one of the following three forms: RGB, depth, and RGB-D images.

The pipeline of the proposed method is as follows. The input image  $I_{in}$  is firstly preprocessed by the information enhancement module, and the feature map  $I_{cnt}$  after enhancement is sent to the encoder for multi-scale convolutional results  $EL_1$ ,  $EL_2$ , and  $EL_3$ . It is pointed out that the channel number of  $I_{cnt}$  is varied according to the input forms of  $I_{in}$  (RGB, depth, or RGB-D). Thus, the channel number of  $EL_1$  is fixed to adapt to different input forms. These feature maps are then combined by SCA to achieve the first phase of fusion, where the spatial information of these features is merged by size adjustment and concatenation, and the merged features are reorganized by adaptively learning the importance of each channel. Also, the high-level feature map  $EL_3$  from the last layer of encoder is processed by ISR, which maintains the high-dimensional information by cascading multiple Inverted Shuffle Units (ISU) with residual connection. The multi-scale feature maps  $ML_1$ ,  $ML_2$ , and  $EL_3$  of SCA and the feature  $F_{ISR}$  generated from ISR are gradually integrated in the decoder. The fused feature map  $D_3$  holds the same size as the original input image, which is followed by 4



**Fig. 1** Structure of the proposed grasp detection network. The input image  $I_{in}$  is firstly pre-processed by the information enhancement module to output the enhanced result  $I_{cnt}$ , which is fed into the encoder module, and feature maps  $EL_1$ ,  $EL_2$ , and  $EL_3$  with different sizes are obtained. These three feature maps are fused through the SCA module in both spatial and channel dimensions to generate new feature maps  $ML_1$ ,  $ML_2$ , and  $ML_3$ . Besides,  $EL_3$  is also inputted to the ISR module to obtain high-dimensional feature maps  $F_{ISR}$  in channel. Taking  $ML_1$ ,  $ML_2$ ,  $ML_3$ , and  $F_{ISR}$  as inputs, the decoder module further executes multi-scale feature fusion for grasp prediction ( $G_q$ ,  $G_{sin}$ ,  $G_{cos}$ ,  $G_w$ ). The feature fusion operator is implemented by an elementwise addition followed by a  $1 \times 1$  convolution layer; the unit of the images in the structure is pixel.

parallel convolutional layers to predict grasp ( $G_q, G_{sin}, G_{cos}, G_w$ ).  $G_q, G_{sin}, G_{cos}$ , and  $G_w$  refer to the feature maps of grasp quality, sine and cosine of grasp angle, and the width of grasp rectangle, respectively.

### 3.1 Information enhancement module

This module is used to strengthen the diversity of input information with pyramid scene parsing<sup>[31]</sup>. Similar to Ref. [32] that efficiently captures information about different regions using different kernels for adaptability improvement, we leverage four adaptive average pooling operations with different kernels in parallel to capture contextual information from the input image  $I_{in}$ , as illustrated in Fig. 2. Four pointwise convolution operations are separately executed on the pooled feature maps to change the number of channels. The generated four feature maps are respectively up-sampled to the same size for further concatenation with  $I_{in}$ , then we have feature map  $I_{cnt}$ . For the input of RGB, depth, or RGB-D, the channel numbers of each  $1 \times 1$  convolution layer are set to 3, 5, and 4, respectively. Correspondingly, the channel numbers of  $I_{cnt}$  become 15, 21, and 20, respectively.

### 3.2 SCA

SCA aims to facilitate the information interaction of multi-scale features among different layers of the encoder. In particular, the detail information in low-level features is mined for better feature refinement. The detailed structure is shown in Fig. 3a, which takes feature maps  $EL_t (t = 1, 2, 3)$  as inputs. In order to achieve the fusion of input feature maps, three groups of parallel

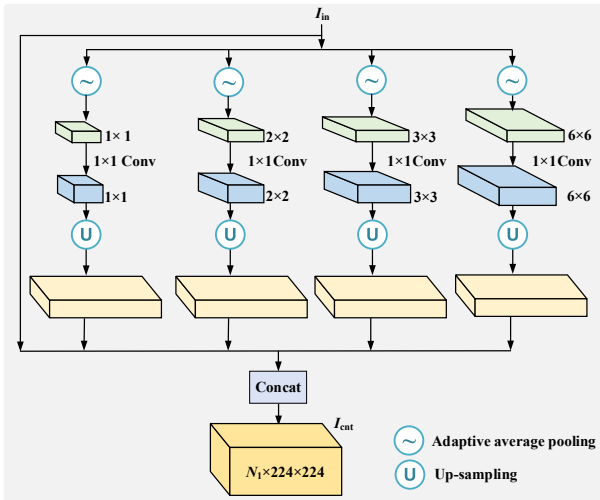


Fig. 2 Structure of information enhancement module, where  $N_1$  denotes the channel number of feature maps (the unit of images in the structure is pixel).

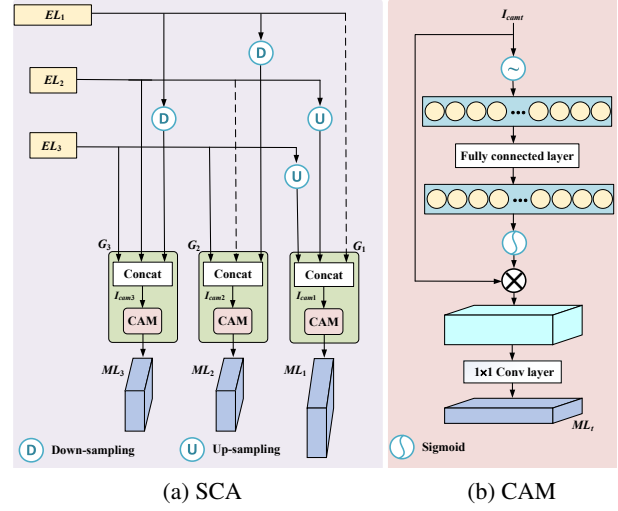


Fig. 3 Structure of skip connections (dashed lines) with attention module.  $I_{cant}$  and  $ML_t$  refer to the input and output of the  $t$ -th CAM<sup>[33]</sup>, where  $t = 1, 2, 3$ .

operations with channel attention are conducted to yield three feature maps  $ML_t$ . For a group, the input  $EL_t$  is resized to the required size of output feature map by down-sampling or up-sampling operation. Take the second group  $G_2$  as an example and the required size of output feature map is the same as that of  $EL_2$ . Thus,  $EL_1$  and  $EL_3$  should be aligned to  $EL_2$ . Concretely,  $EL_3$  is kept fixed and  $EL_1$  is down-sampled according to their size relationship. Further, the aligned feature maps are concatenated in channel to generate  $I_{cam2}$ , which is followed by a Channel Attention Module (CAM)<sup>[33]</sup> to adaptively recalibrate channel-wise feature responses (see Fig. 3b). Since the channel dimension of the concatenated feature map is increased, a pointwise convolution is employed at the end of CAM to reduce the channel number to that of  $EL_2$ . Eventually, the output  $ML_2$  of  $G_2$  is generated. Similarly, the outputs  $ML_1$  and  $ML_3$  are obtained.

### 3.3 ISR

This module is designed to exploit channel information of high-level feature in the encoder, and thus achieving high-dimensional feature enhancement. As is shown in Fig. 1, it consists of three cascaded ISUs with a residual connection<sup>[34, 35]</sup>, where the structure of ISU is illustrated in Fig. 4. For input feature map  $U_{in}$  with  $2N_a$  channels, ISU first executes channel split, and two split features with  $N_a$  channels are sent to their respective branches. Both branches utilize the depthwise convolution to capture the neighboring information at each spatial location of the feature map with a

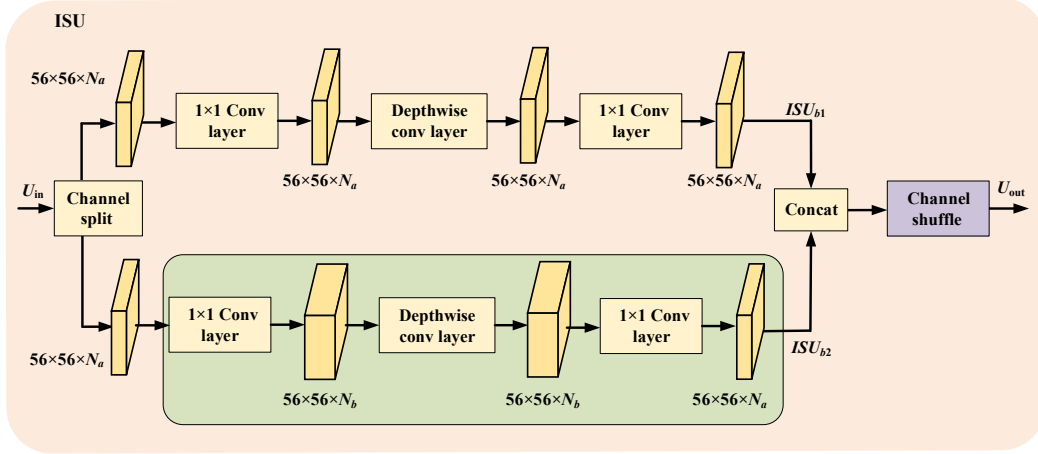


Fig. 4 Structure of an ISU.

small number of parameters. Meanwhile, in order to compensate for the deficiency of depthwise convolution without channel interaction, two pointwise convolutions are employed before and after depthwise convolution to enhance the association among different channels. Different from the upper branch with unchanged channel number  $N_a$ , the down branch expands the channel number of input feature map from  $N_a$  to  $N_b$  ( $N_b \geq N_a$ ) to obtain rich feature representation. After spatial interaction of features by the depthwise convolution, the high-dimensional feature is re-mapped to lower-dimensional space by the second pointwise convolution with batch normalization and ReLU to restore the original input channel, which ensures the efficiency of network. Formally,

$$ISU_{b1} = Conv_s(DC(Conv_s(f_{split}(U_{in}))))),$$

$$ISU_{b2} = Conv_r(DC(Conv_i(f_{split}(U_{in})))) \quad (1)$$

where  $ISU_{b1}$  and  $ISU_{b2}$  are the output feature maps for the two branches of ISU, respectively.  $f_{split}(\cdot)$  and  $DC(\cdot)$  refer to channel split operation and depthwise convolution, respectively.  $Conv_s(\cdot)$ ,  $Conv_i(\cdot)$ , and  $Conv_r(\cdot)$  denote convolutions with the channel number unchanged, increased, and reduced, respectively.

The results  $ISU_{b1}$  and  $ISU_{b2}$  from two branches are fused through concatenation in channel. Furthermore, channel shuffle from ShuffleNet<sup>[36, 37]</sup> is taken to promote the information interaction in channel between these two branches. The output of ISU can be described as follows:

$$U_{out} = f_{cs}(Cat[ISU_{b1}, ISU_{b2}]) \quad (2)$$

where  $Cat[\cdot]$  denotes concatenation,  $f_{cs}(\cdot)$  represents the operation of channel shuffle.

ISU serves as a basic building block to construct ISR. In the ISR module, ISU is recursively used 3 times with

a residual structure connecting the input of the first ISU and the output of the last one for fast convergence, and the feature  $F_{ISR}$  is obtained. In the first ISU, the channel number  $N_b$  is the same as  $N_a$ , while  $N_b$  is twice as much as  $N_a$  in the latter two ISUs, where  $N_a = 128$ .

### 3.4 Feature fusion in the decoder

Given the multi-scale features  $ML_t$  ( $t = 1, 2, 3$ ) from SCA and the feature  $F_{ISR}$  from ISR module, the decoder fulfills the secondary fusion of features. As is illustrated in Fig. 1, the feature map  $ML_3$  with the smallest spatial size is combined with  $F_{ISR}$  by a feature fusion operator  $f_{fusion}(\cdot)$ , which consists of an elementwise addition operation and a pointwise convolution with batch normalization. The former merges two input feature maps at each spatial position and the latter promotes the information interaction along channel. The fused result is further processed by a deconvolution layer, which outputs the feature map  $D_1$ , whose spatial size is the same as that of  $ML_2$  for the next fusion,

$$D_1 = DeConv(f_{fusion}(ML_3, F_{ISR})) = DeConv(Conv(ML_3 + F_{ISR})) \quad (3)$$

where  $Conv(\cdot)$  and  $DeConv(\cdot)$  stand for a convolutional layer and a deconvolution layer, respectively.

Similar fusion processes are conducted three times to obtain the feature map  $D_3$ , which is then converted to the pixel-wise grasp prediction  $G = (G_q, G_{sin}, G_{cos}, G_w)$  by four parallel convolutional layers, where  $G_q$  represents the grasp quality feature map.  $G_{sin}$  and  $G_{cos}$  record sine and cosine related to predicted grasp angle  $\theta$  at every pixel. The whole grasp detection process is shown in Algorithm 1,  $f_{EnInf}(\cdot)$  and  $f_{Encoder}(\cdot)$  denote the information enhancement and encoding,

**Algorithm 1** Grasp detection process**Input:** Input image  $I_{in}$ **Output:** Results of grasp detection  $G_q, G_{sin}, G_{cos}$ , and  $G_w$ 


---

```

1:  $I_{cnt} = f_{EnInf}(I_{in});$ 
2:  $EL_1, EL_2, EL_3 \leftarrow f_{Encoder}(I_{cnt});$ 
3:  $ML_1, ML_2, ML_3 \leftarrow f_{SCA}(EL_1, EL_2, EL_3);$ 
4:  $ISR_{mid} = EL_3;$ 
5: for  $r \in [1, N_{ISU}]$  do
6:    $F_{ISR} \leftarrow f_{ISU}^r(ISR_{mid});$ 
7:    $ISR_{mid} = F_{ISR};$ 
8: end for
9:  $F_{ISR} = F_{ISR} + EL_3;$ 
10:  $F_{Demid} = \mathbf{0}, F_{in} = F_{ISR};$ 
11: for  $p \in [1, 3]$  do
12:    $F_{Demid} = f_{fusion}(F_{in}, ML_{4-p});$ 
13:    $D_p = DeConv(F_{Demid});$ 
14:    $F_{in} = D_p;$ 
15: end for
16:  $G_q, G_{sin}, G_{cos}, G_w \leftarrow {}^4Conv(D_3);$ 
17: return  $G_q, G_{sin}, G_{cos}, G_w$ 

```

---

respectively.  $f_{SCA}(\cdot)$  describes the SCA processing and  $f_{ISU}^r(\cdot)$  is the  $r$ -th ISU function.  $N_{ISU}$  is the cascaded number of ISU.  ${}^4Conv$  refers to convolution operation in four parallel branches.

The grasp angle feature map  $G_\theta$  can be derived by  $G_\theta = \frac{1}{2}arctan(G_{sin}/G_{cos})$ .  $G_w$  contains the width of grasp rectangle at each pixel position. The best grasp result is then determined according to the grasp quality of each pixel. Concretely, the pixel coordinate  $p^* = argmax_{(p)}(G_q)$  corresponding to the maximal grasp quality is first selected, where  $p$  denotes the pixel coordinate. The best grasp rectangle can be expressed as  $G^* = (p^*, G_\theta|_{p^*}, G_w|_{p^*})$ .

**3.5 Loss function**

The proposed network is trained by a smooth  $L_1$  loss under the supervision of ground truth, which is described as follows:

$$L_g = \frac{1}{N} \sum_{i=1}^n Smooth_{L_1}(G_i - \bar{G}_i) \quad (4)$$

$$Smooth_{L_1}(x) = \begin{cases} (\alpha x)^2/2, & \text{if } |x| < 1; \\ |x| - 1/2\alpha^2, & \text{others} \end{cases} \quad (5)$$

where  $N$  denotes the number of training samples.  $x$  represents the difference between  $G_i$  and  $\bar{G}_i$ , and  $G_i$  and  $\bar{G}_i$  refer to the prediction result of HMISR for the  $i$ -th sample and the corresponding ground truth, respectively.  $\alpha$  is a hyper-parameter ranging from 0 to 1 to control the smoothness. When the difference  $x$  is too large

or too small, Smooth  $L_1$  can provide an appropriate gradient value to avoid gradient explosion and gradient disappearance.

**4 Experiment****4.1 Experimental setup**

In this part, extensive experiments are conducted to verify the effectiveness of the proposed method. Following the existing grasp detection methods<sup>[22, 24, 25]</sup>, the proposed HMISR network is trained and tested on the public Cornell grasping dataset<sup>[15]</sup>, which includes RGB-D images of objects and is widely used as an evaluation platform for grasp detection. In detail, the dataset contains 885 images of 240 different objects with a resolution of 640 pixel $\times$ 480 pixel, where the number of positive grasps is 5110 and that of negative grasps is 2909. The ground truth including feature maps of grasp quality, angle, and width is obtained according to Ref. [23]. During the training process, random cropping, zooming, and rotation are applied for data augmentation to improve the robustness of network. The network is trained with Adam optimizer for 50 epochs, where the batch size is set as 8 and the probability of dropout is 0.1 for avoiding overfitting. At the phase of test, the predicted grasp quality map is filtered by applying a Gaussian kernel. Our method runs on a platform with NVIDIA GTX1080 GPU with 8 GB memory and Intel Core i7-7770HQ CPU.

For a grasp prediction result, it is regarded as valid if the following conditions are satisfied<sup>[23–25, 38]</sup>: (1) the Intersection over Union (IoU) between the predicted grasp rectangle and the corresponding ground truth is over 25%; (2) the difference between the predicted grasp angle and its corresponding ground truth is less than 30. Following Ref. [23], the dataset is split into training and test sets in two ways: Image-Wise (IW) and Object-Wise (OW). The former tests the generalization of the network for new grasp pose, while the latter aims to verify the generalization ability for new objects. Correspondingly, two evaluation metrics are adopted: image-wise accuracy (namely IW acc.) and object-wise accuracy (namely OW acc.).

**4.2 Ablation studies**

To verify the effectiveness of our HMISR network, its five variants are constructed according to whether information enhancement, SCA, ISR, basic decoder, and the proposed feature fusion decoder are involved. The basic decoder is composed of three cascaded

deconvolutional layers. All variants adopt the same encoder architecture as that of HMISR. Table 1 presents the comparison results of different variants on the Cornell grasping dataset in terms of image-wise and object-wise accuracies. The results of HMISR-I with a basic encoder-decoder architecture are the lowest. The addition of information enhancement module is helpful to improve the accuracy, which can be seen from the results of HMISR-I and HMISR-II. Comparing HMISR-II with HMISR-III, one can see that our decoder with multi-feature fusion performs better than the basic decoder. On this basis, HMISR-IV and HMISR-V adds the SCA and ISR modules, respectively, and both of them implement improvements in terms of IW acc. and OW acc. With the combination of SCA, ISR, and feature fusion decoder, the proposed HMISR attains the best accuracy. Actually, HMISR mainly enhances the OW acc. with the same IW acc. as that of HMISR-IV and HMISR-V. The reason is from the split way of dataset. Under the image-wise split, the training and test datasets contain objects with the same classes but different poses, which is less challenging. It is enough to rely on SCA or ISR on the basis of the information enhancement module and the proposed decoder. For the object-wise split, the training and test datasets are related to objects with different classes, which increases difficulty due to unseen object classes at test phase. The OW acc. of HMISR indicates that the combination of SCA and ISR promotes the generalization to new objects with a higher

accuracy.

Table 2 presents the ablation of the proposed ISR module, where different numbers of ISUs and connection ways are considered. 6 variants of ISR are constructed and their structures are illustrated in Table 2. As we can see, ISR with three ISUs obtains better results than ISR-I and ISR-II. Also, although ISR-III and ISR-V achieve the same accuracy as that of ISR, the complex architectures of these two variants increase memory usage with more network parameters. Thus, the proposed ISR structure is considered as the best.


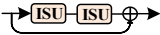
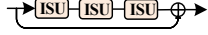
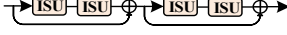


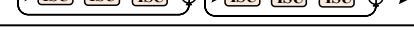
### 4.3 Comparisons with the existing methods

In this section, the proposed HMISR network is compared with the existing methods including SAE<sup>[15]</sup>, regression grasp<sup>[16]</sup>, DCNN<sup>[17]</sup>, GRPN<sup>[18]</sup>, ROI-GD<sup>[19]</sup>, closed-loop grasp<sup>[39]</sup>, GN<sup>[40]</sup>, GPN-GD<sup>[41]</sup>, FCGN<sup>[21]</sup>, multimodal fusion<sup>[22]</sup>, GG-CNN<sup>[23]</sup>, GR-ConvNet<sup>[24]</sup>, and TsGNet<sup>[25]</sup>, where the first 8 methods belong to the category based on fully connected layers, and the last 5 methods are fully convolution based methods. Combining these two types together, DSGD<sup>[42]</sup> constructs global, region, and pixel level networks, and each network is evaluated by the corresponding confidence score. The grasp detection result from the network with the highest score is outputted. Besides, TF-Grasp<sup>[43]</sup> is the first to achieve the grasp detection using transformer with the fusion of local and global features. Table 3 illustrates the results of different methods on the

**Table 1** Grasp detection accuracy of different variants of HMISR on the Cornell grasping dataset.

Method	Information enhancement	SCA	ISR	Basic decoder	Feature fusion decoder	IW acc. (%)	OW acc. (%)
HMISR-I	–	–	–	✓	–	91.0	89.9
HMISR-II	✓	–	–	✓	–	93.3	91.0
HMISR-III	✓	–	–	–	✓	94.4	91.0
HMISR-IV	✓	✓	–	–	✓	98.9	94.4
HMISR-V	✓	–	✓	–	✓	98.9	93.3
HMISR	✓	✓	✓	–	✓	98.9	97.8

**Table 2** Ablation of ISR module in terms of IW acc. and OW acc.

Method	Structure	IW acc. (%)	OW acc. (%)
ISR-I		97.8	97.8
ISR-II		97.8	94.4
ISR		98.9	97.8
ISR-III		98.9	97.8
ISR-IV		98.9	96.6
ISR-V		98.9	97.8
ISR-VI		98.9	96.6



**Table 3 Comparison results of different methods on the Cornell grasping dataset.**

Method	Input size (pixel×pixel)	Number of parameters	Input mode		Accuracy (%)		Time (ms)
			RGB	Depth	Image-wise	Object-wise	
SAE <sup>[15]</sup>	–	> 1 050 500	✓	✓	73.9	75.6	13 500 (without device information)
Regression grasp <sup>[16]</sup>	224×224	> 7 300 000	✓	✓	88.0	87.1	76 (NVIDIA Tesla K20 GPU)
DCNN <sup>[17]</sup>	224×224	> 20 000 000	✓	✓	89.2	89.0	103 (NVIDIA GeForce GTX 645 GPU)
GRPN <sup>[18]</sup>	–	–	✓	–	88.7	–	500 (NVIDIA GTX 1070 GPU)
ROI-GD <sup>[19]</sup>	–	> 30 000 000	✓	–	93.6	93.5	40 (GTX1080Ti GPU)
Closed-loop grasp <sup>[39]</sup>	–	–	✓	✓	81.8	–	141 (GeForce GTX 980 GPU)
FCGN <sup>[21]</sup>	320×320	> 27 000 000	✓	–	97.7	96.6	118 (NVIDIA TITAN-X)
Multimodal Fusion <sup>[22]</sup>	224×224	> 4 400 000	✓	✓	88.9	88.2	117 (NVIDIA Tesla K80 GPUs)
GG-CNN <sup>[23]</sup>	300×300	62 420	✓	✓	73.0	69.0	19 (NVIDIA GeForce GTX 1070)
GR-ConvNet <sup>[24]</sup>	224×224	1 900 900	✓	✓	97.7	96.6	20 (NVIDIA GeForce GTX 1080 Ti)
TsGNet <sup>[25]</sup>	300×300	66 754	✓	✓	93.1	93.0	–
GN <sup>[40]</sup>	227×227	> 11 000 000	✓	✓	96.0	96.1	120 (NVIDIA Titan-X)
GPN-GD <sup>[41]</sup>	227×227	> 31 000 000	✓	✓	97.2	97.1	81 (NVIDIA GeForce RTX 2080 Ti GPU)
DSGD <sup>[42]</sup>	–	> 13 000 000	✓	✓	97.5	–	111 (NVIDIA Tesla K80 GPU)
TF-Grasp <sup>[43]</sup>	224×224	–	✓	✓	98.0	96.7	41.6 (NVIDIA3090 GPU)
HMISR (depth)	224×224	1 089 856	–	✓	92.1	89.8	14 (NVIDIA GTX1080 GPU)
HMISR (RGB)	224×224	1 088 128	✓	–	95.5	95.5	15 (NVIDIA GTX1080 GPU)
HMISR (RGB-D)	224×224	1 089 604	✓	✓	98.9	97.8	15 (NVIDIA GTX1080 GPU)

Cornell grasping dataset. Besides the image-wise and object-wise accuracies, the running time and the number of parameters are also provided. The detailed calculation of parameters for the proposed HMISR method with RGB-D image is shown in Table 4, where there are three sequential processing blocks in each of encoder, ISR, and decoder. The output head refers to convolution operation in four branches on the decoder output. Overall, HMISR (RGB-D) performs well in both image-wise and object-wise accuracies. For the solution with only RGB input, FCGN attains higher accuracy than our HMISR with RGB input. In addition, GG-CNN is also run on our GPU platform and its running time reaches 13 ms. By

**Table 4 Parameter calculation of the proposed HMISR method.**

Item	Number of parameters
Information enhancement	96
Encoder	$E_1$ : 5856
	$E_2$ : 18 624
	$E_3$ : 74 112
SCA	483 392
ISR	ISU <sub>1</sub> : 22 976
	ISU <sub>2</sub> : 65 856
	ISU <sub>3</sub> : 65 856
Decoder	$DL_1$ : 221 760
	$DL_2$ : 78 112
	$DL_3$ : 52 448
Output head	516
<b>Total</b>	<b>1 089 604</b>

contrast, the running time (15 ms) including computation of the best grasp of our HMISR is slightly slower but with higher accuracy. Considering the accuracy and efficiency, the proposed HMISR is considered as effective.

Figure 5 visualizes the predicted grasp detection results of the proposed method on the Cornell dataset. The first two columns describe the input RGB and depth images of three objects, and their corresponding grasp rectangles are shown on RGB images. Also, the predicted feature maps of grasp quality, angle, and width are exhibited in columns 3, 4, and 5, respectively. It is seen that our method achieves the correct grasps. Figure 6 demonstrates the predicted grasp rectangles of our HMISR for objects with different shapes and poses on the Cornell grasping dataset, which shows the good adaptability of our method.

#### 4.4 Robustness verification

Different interferences are applied to the original RGB and depth images, and then the noisy images are fed to HMISR to verify the robustness. The grasp detection result of original image is shown in the first column of the Fig. 7. The remaining parts present the grasp detection results on different interference images. The second column describes the result of the noisy image after Gaussian blur with kernel size 10×10 exerted. The third to the sixth columns correspond to the interferences from Gaussian noise with standard

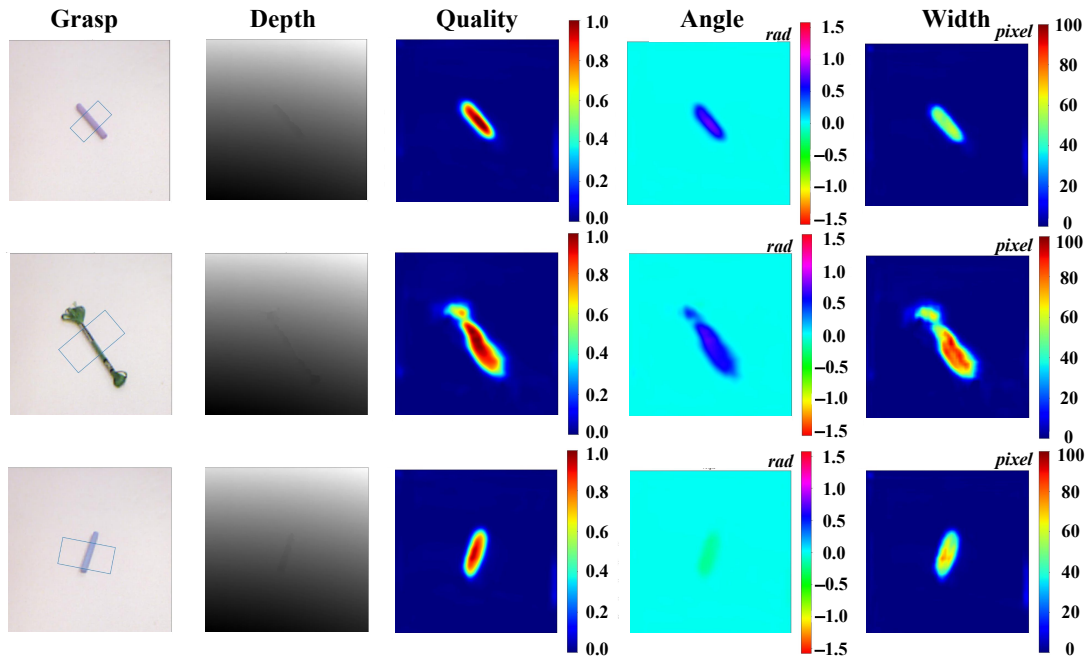


Fig. 5 Detection results of HMISR (RGB-D) on the Cornell grasping dataset, where the first column presents the RGB images with the predicted grasp rectangles, the second column describes the images of depth-type, and the subsequent columns show the feature maps of grasp quality, angle, and width, respectively.

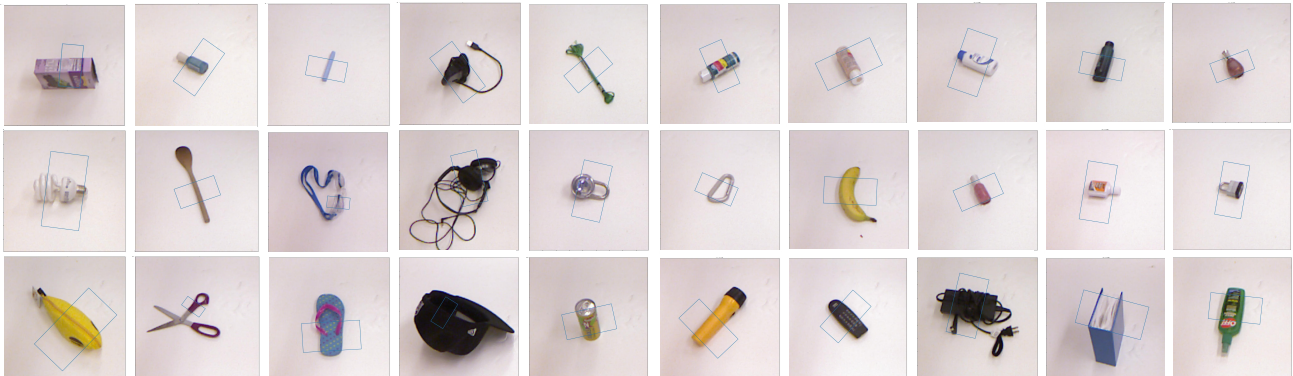


Fig. 6 Predicted grasp rectangles of HMISR for objects on the Cornell grasping dataset.

deviation 0.02, brightness enhancement (20%), salt-pepper noise with the ratio value 0.04, and GridMask with size 3 pixel $\times$ 3 pixel, respectively. As shown in the experimental results, the proposed method can still obtain correct results.

#### 4.5 Grasp detection in actual scene

The proposed HMISR is also applied to an actual scene to further testify its effectiveness. Herein we concern the fruits, which are detected by YOLACT<sup>[44]</sup>. The regions corresponding to the detected box of each target object are extracted in the original RGB and depth images, and border padding and resizing operations are imposed to adjust the sizes of regions to that of training image. The

resultant RGB and depth images are fed into HMISR network for grasp prediction. The experimental results are illustrated in Fig. 8, where the apple, orange, and banana are the concerned objects. The detection results and corresponding grasp detection results are presented in Figs. 8b and 8c, respectively. The results manifest the effectiveness of the proposed method.

## 5 Conclusion

In this paper, a grasp detection network with hierarchical multi-scale feature fusion and ISR is proposed. With the framework of encoder-decoder structure, three modules are added: information enhancement, SCA, and ISR. Firstly, in the information enhancement module, the



Fig. 7 Grasp detection results of HMISR under different interferences.

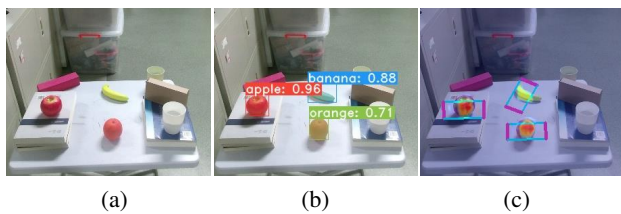


Fig. 8 Experiment in an actual scene. (a) RGB image, (b) detection results of concerned objects, and (c) grasp detection results.

contextual information of the input image is strengthened by adaptive average pooling operations in four parallel branches. Then, multi-scale features from the encoder are fused by SCA, which fully utilizes the detail information from low-level features for better feature refinement. Meanwhile, the high-level feature from the encoder is also enhanced at channel level based on ISR module. This enriches the channel information in high-level feature of the encoder. The resultant features from ISR and SCA modules are further aggregated and fused in the designed decoder. Through the hierarchical feature fusion, the quality of grasp prediction is improved. The proposed method is testified on the Cornell grasping dataset and an actual scene, and the results indicate that the proposed method achieves good accuracy with the robustness to disturbances.

### Acknowledgment

This work was supported by the National Natural Science Foundation of China (Nos. 62073322 and 61633020), the CIE-Tencent Robotics X Rhino-Bird Focused Research Program (No. 2022-07), and the Beijing Natural Science

Foundation (No. 2022MQ05).

### References

- [1] A. Bar, X. Wang, V. Kantorov, C. J. Reed, R. Herzig, G. Chechik, A. Rohrbach, T. Darrell, and A. Globerson, DETReg: Unsupervised pretraining with region priors for object detection, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, New Orleans, LA, USA, 2022, pp. 14585–14595.
- [2] W. Van Gansbeke, S. Vandenhende, S. Georgoulis, and L. Van Gool, Unsupervised semantic segmentation by contrasting object mask proposals, in *Proc. IEEE/CVF Int. Conf. Computer Vision*, Montreal, Canada, 2021, pp. 10032–10042.
- [3] Y. T. Chen, H. P. Zhang, L. W. Liu, J. J. Tao, Q. Zhang, K. Yang, R. L. Xia, and J. B. Xie, Research on image inpainting algorithm of improved total variation minimization method, *J. Ambient Intell. Human. Comput.*, vol. 14, no. 5, pp. 5555–5564, 2023.
- [4] Y. Y. Yu, Z. Q. Cao, S. Liang, W. J. Geng, and J. Z. Yu, A novel vision-based grasping method under occlusion for manipulating robotic system, *IEEE Sensors J.*, vol. 20, no. 18, pp. 10996–11006, 2020.
- [5] T. Suzuki and T. Oka, Grasping of unknown objects on a planar surface using a single depth image, in *Proc. IEEE Int. Conf. Advanced Intelligent Mechatronics*, Banff, Canada, 2016, pp. 572–577.
- [6] B. S. Zapata-Impata, P. Gil, J. Pomares, and F. Torres, Fast geometry-based computation of grasping points on three-dimensional point clouds, *Int. J. Adv. Rob. Syst.*, vol. 16, no. 1, pp. 1–18, 2019.
- [7] G. Vezzani, U. Pattacini, and L. Natale, A grasping approach based on superquadric models, in *Proc. IEEE Int. Conf. Robotics and Automation*, Singapore, 2017, pp. 1579–1586.
- [8] I. Gori, U. Pattacini, V. Tikhonoff, and G. Metta, Ranking the good points: A comprehensive method for humanoid

- robots to grasp unknown objects, in *Proc. 16<sup>th</sup> Int. Conf. Advanced Robotics*, Montevideo, Uruguay, 2013, pp. 1–7.
- [9] Y. Li and N. S. Pollard, A shape matching algorithm for synthesizing humanlike enveloping grasps, in *Proc. 5<sup>th</sup> IEEE-RAS Int. Conf. Humanoid Robots*, Tsukuba, Japan, 2005, pp. 442–449.
- [10] A. Herzog, P. Pastor, M. Kalakrishnan, L. Righetti, T. Asfour, and S. Schaal, Template-based learning of grasp selection, in *Proc. IEEE Int. Conf. Robotics and Automation*, Saint Paul, MN, USA, 2012, pp. 2379–2384.
- [11] Z. Y. Wang, Z. D. Deng, and S. Y. Wang, CasNet: A cascade coarse-to-fine network for semantic segmentation, *Tsinghua Science and Technology*, vol. 24, no. 2, pp. 207–215, 2019.
- [12] R. Y. Xin, J. Zhang, and Y. T. Shao, Complex network classification with convolutional neural network, *Tsinghua Science and Technology*, vol. 25, no. 4, pp. 447–457, 2020.
- [13] Q. Hua, L. Chen, P. Li, S. Zhao, and Y. Li, A pixel-channel hybrid attention model for image processing, *Tsinghua Science and Technology*, vol. 27, no. 5, pp. 804–816, 2022.
- [14] 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, arXiv preprint arXiv: 1703.09312v3, 2017.
- [15] I. Lenz, H. Lee, and A. Saxena, Deep learning for detecting robotic grasps, arXiv preprint arXiv: 1301.3592v4, 2014.
- [16] J. Redmon and A. Angelova, Real-time grasp detection using convolutional neural networks, in *Proc. IEEE Int. Conf. Robotics and Automation*, Seattle, WA, USA, 2015, pp. 1316–1322.
- [17] S. Kumra and C. Kanan, Robotic grasp detection using deep convolutional neural networks, in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Vancouver, Canada, 2017, pp. 769–776.
- [18] H. Karaoguz and P. Jensfelt, Object detection approach for robot grasp detection, in *Proc. Int. Conf. Robotics and Automation*, Montreal, Canada, 2019, pp. 4953–4959.
- [19] H. Zhang, X. Lan, S. Bai, X. Zhou, Z. Tian, and N. Zheng, ROI-based robotic grasp detection for object overlapping scenes, in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Macau, China, 2019, pp. 4768–4775.
- [20] V. Satish, J. Mahler, and K. Goldberg, On-policy dataset synthesis for learning robot grasping policies using fully convolutional deep networks, *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1357–1364, 2019.
- [21] X. Zhou, X. Lan, H. Zhang, Z. Tian, Y. Zhang, and N. Zheng, Fully convolutional grasp detection network with oriented anchor box, in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Madrid, Spain, 2018, pp. 7223–7230.
- [22] Q. Zhang, D. Qu, F. Xu, and F. Zou, Robust robot grasp detection in multimodal fusion, *MATEC Web Conf.*, vol. 139, p. 00060, 2017.
- [23] 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.
- [24] S. Kumra, S. Joshi, and F. Sahin, Antipodal robotic grasping using generative residual convolutional neural network, in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Las Vegas, NV, USA, 2020, pp. 9626–9633.
- [25] Y. Y. Yu, Z. Q. Cao, Z. C. Liu, W. J. Geng, J. Z. Yu, and W. M. Zhang, A two-stream CNN with simultaneous detection and segmentation for robotic grasping, *IEEE Trans. Syst. Man Cybern.: Syst.*, vol. 52, no. 2, pp. 1167–1181, 2022.
- [26] G. Sistu, I. Leang, and S. Yogamani, Real-time joint object detection and semantic segmentation network for automated driving, arXiv preprint arXiv: 1901.03912v1, 2019.
- [27] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, SSD: Single shot MultiBox detector, arXiv preprint arXiv: 1512.02325v5, 2016.
- [28] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, Feature pyramid networks for object detection, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017, pp. 936–944.
- [29] Y. T. Chen, V. Phonevilay, J. J. Tao, X. Chen, R. L. Xia, Q. Zhang, K. Yang, J. Xiong, and J. B. Xie, The face image super-resolution algorithm based on combined representation learning, *Multimed. Tools Appl.*, vol. 80, no. 20, pp. 30839–30861, 2021.
- [30] Y. T. Chen, L. W. Liu, V. Phonevilay, K. Gu, R. L. Xia, J. B. Xie, Q. Zhang, and K. Yang, Image super-resolution reconstruction based on feature map attention mechanism, *Appl. Intell.*, vol. 51, no. 7, pp. 4367–4380, 2021.
- [31] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, Pyramid scene parsing network, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017, pp. 6230–6239.
- [32] R. L. Xia, Y. T. Chen, and B. B. Ren, Improved anti-occlusion object tracking algorithm using Unscented Rauch-Tung-Striebel smoother and kernel correlation filter, *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 8, pp. 6008–6018, 2022.
- [33] J. Hu, L. Shen, and G. Sun, Squeeze-and-excitation networks, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 7132–7141.
- [34] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016, pp. 770–778.
- [35] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, MobileNetV2: Inverted residuals and linear bottlenecks, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 4510–4520.
- [36] X. Zhang, X. Zhou, M. Lin, and J. Sun, ShuffleNet: An extremely efficient convolutional neural network for mobile devices, arXiv preprint arXiv: 1707.01083v2, 2017.
- [37] N. Ma, X. Zhang, H. T. Zheng, and J. Sun, ShuffleNet V2: Practical guidelines for efficient CNN architecture design, arXiv preprint arXiv: 1807.11164, 2018.
- [38] Y. Jiang, S. Moseson, and A. Saxena, Efficient grasping from RGBD images: Learning using a new rectangle representation, in *Proc. IEEE Int. Conf. Robotics and Automation*, Shanghai, China, 2011, pp. 3304–3311.
- [39] Z. Wang, Z. Li, B. Wang, and H. Liu, Robot grasp detection using multimodal deep convolutional neural networks, *Adv. Mech. Eng.*, vol. 8, no. 9, pp. 1–12, 2016.

- [40] F. J. Chu, R. Xu, and P. A. Vela, Real-world multiobject, multigrasp detection, *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3355–3362, 2018.
- [41] W. Ouyang, W. Huang, and H. Min, Robot grasp with multi-object detection based on RGB-D image, in *Proc. China Automation Congress*, Beijing, China, 2021, pp. 6543–6548.
- [42] U. Asif, J. Tang, and S. Harrer, Densely supervised grasp detector (DSGD), arXiv preprint arXiv: 1810.03962v2, 2019.
- [43] S. Wang, Z. Zhou, and Z. Kan, When transformer meets robotic grasping: Exploits context for efficient grasp detection, *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 8170–8177, 2022.
- [44] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, YOLACT: Real-time instance segmentation, in *Proc. IEEE/CVF Int. Conf. Computer Vision*, Seoul, Republic of Korea, 2019, pp. 9156–9165.



**Wengjie Geng** received the BEng and MEng degrees from Harbin Engineering University, China in 2015 and 2017, respectively. He is currently a PhD candidate in control theory and control engineering at the Institute of Automation, Chinese Academy of Sciences, Beijing, China. His current research interests include

visual perception and robotic grasping.



**Zhiqiang Cao** received the BEng degree in industrial automation and MEng degree in control theory and control engineering both from Shandong University of Technology, China in 1996 and 1999, respectively, and the PhD degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences,

China in 2002. He is currently a professor at the Institute of Automation, Chinese Academy of Sciences. His research interests include service robots and intelligent robot.



**Junzhi Yu** received the BEng degree in safety engineering and the MEng degree in precision instruments and mechatronics both from the North University of China, China in 1998 and 2001, respectively, and the PhD degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences,

China in 2003. From 2004 to 2006, he was a postdoctoral researcher at the Center for Systems and Control, Peking University, China. He was an associate professor at the Institute of Automation, Chinese Academy of Sciences in 2006, where he became a full professor in 2012. In 2018, he joined the College of Engineering, Peking University, as a tenured full professor. His current research interests include intelligent robots, motion control, and intelligent mechatronic systems.



**Peiyu Guan** received the BEng degree in electronic information science and technology from Jilin University, China in 2017, and the PhD degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, China in 2022. She is currently an assistant professor at the Institute of Automation,

Chinese Academy of Sciences. Her research interests include service robot and image processing.



**Fengshui Jing** received the BEng degree in mining engineering from Huainan Mining Institute, China in 1991, the MEng degree in safety technology and engineering from Shandong Mining Institute, China in 1994, and the PhD degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences,

China in 2002. He is currently a professor at the Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include robotics, computer vision, and manufacturing systems.



**Min Tan** received the BEng degree from Tsinghua University, China in 1986, and the PhD degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China in 1990. He is currently a professor at the Institute of Automation, Chinese Academy of Sciences. His research

interests include advanced robot control and biomimetic robot.