Robotica (2025), **43**, pp. 415–432
doi:10.1017/S0263574724001929

CAMBRIDGE UNIVERSITY PRESS

RESEARCH ARTICLE

A refined robotic grasp detection network based on coarse-to-fine feature and residual attention

Zhenwei Zhu, Saike Huang, Jialong Xie, Yue Meng, Chaoqun Wang and Fengyu Zhou

School of Control Science and Engineering, Shandong University, Jinan, Shandong, China
Corresponding author: Fengyu Zhou; Email: zhoufengyu@sdu.edu.cn

Received: 26 February 2024; **Revised:** 11 September 2024; **Accepted:** 9 October 2024;
First published online: 28 November 2024

Keywords: Robotic grasping; deep learning; grasping detection; multi-scale features; attention mechanism

Abstract

Precise and efficient grasping detection is vital for robotic arms to execute stable grasping tasks in industrial and household applications. However, existing methods fail to consider refining different scale features and detecting critical regions, resulting in coarse grasping rectangles. To address these issues, we propose a real-time coarse and fine granularity residual attention (CFRA) grasping detection network. First, to enable the network to detect different sizes of objects, we extract and fuse the coarse and fine granularity features. Then, we refine these fused features by introducing a feature refinement module, which enables the network to distinguish between object and background features effectively. Finally, we introduce a residual attention module that handles different shapes of objects adaptively, achieving refined grasping detection. We complete training and testing on both Cornell and Jacquard datasets, achieving detection accuracy of 98.7% and 94.2%, respectively. Moreover, the grasping success rate on the real-world UR3e robot achieves 98%. These results demonstrate the effectiveness and superiority of CFRA.

1. Introduction

Grasping is a fundamental skill for robots and has been applied in various domains such as manufacturing, health care, and domestic settings [1–3]. To achieve precise and efficient manipulation, grasping detection is a pivotal step. In the face of intricate scenarios involving objects of diverse shapes, sizes, and types, robots often exhibit a low success rate in grasping detection. As a result, achieving robust and reliable grasping remains a significant challenge [4].

To address these challenges, there have been extensive works on grasping detection [5, 6]. Grasping detection methods are typically categorized into analytical and data-driven approaches [7]. Analytical methods based on templates are effective but heavily rely on the intrinsic properties of the robot and the physical models of objects. Le *et al.* [8] employed learning-based template matching algorithm for grasping. Studies [9] directly detected the edge map of objects, but these methods are designed based on specific tasks and are susceptible to human subjective factors.

With the advancement of high-performance computers, data-driven methods have shown brightly and attracted the attention of many researchers [10, 11]. For example, Lenz *et al.* [12] employed a sliding window to extract the grasping information of objects and used a classifier to assess the grasping likelihood of each area. Meanwhile, studies [13–15] utilized anchor box-based approaches to predict numerous candidate grasping rectangles. Nevertheless, these methods depend on pre-setting the size of the anchor box, which is both time-consuming and limits the adaptability to objects of different sizes in real-world environments. To address the limitations of relying on existing rules or templates, Cheng

The first two authors contributed equally to this work.

© The Author(s), 2024. Published by Cambridge University Press.

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

et al. [16] utilized a U-shaped architecture comprising fully convolutional layers to predict grasping rectangles. This network excels in pixel-level prediction but overlooks the feature extraction about different object sizes, lacking finer features from images. Advancements in computer vision have spurred research efforts to explore the multi-scale image features and crucial grasping regions. For instance, Yu *et al.* [17] employed diverse convolutional kernels to extract multi-scale features, incorporating spatial attention mechanisms to learn weights for different spatial regions. Zhai *et al.* [18] introduced a local refinement module to predict grasping key points and integrated residual connections to optimize global features but overlooked the critical grasping regions and failed to consider object similarities among different training samples. Additionally, Kumra *et al.* [19] proposed a GRCNN network, which enhances accuracy by stacking residual modules after initial feature extraction. However, these methods still have several limitations: (1) Insufficient consideration for the fusion and refinement of multi-scale features in original images capturing objects of varying sizes. (2) Lack of emphasis on critical grasping regions within objects. The attention mechanism exhibits high computational complexity, and stacking residual modules leads to increased network parameters, resulting in slower inference speed. (3) Overlooking the correlation in feature extraction among different training samples, particularly among objects of different shapes.

To address the above challenges, this paper proposes a **c**oarse and **f**ine granularity **r**esidual **a**ttention grasping detection network (CFRA), comprising three core modules: coarse and fine granularity features fusion (CF), feature refinement (FR), and residual attention (RA). The network inputs RGB images and outputs grasping center position, angle, quality, and gripper opening width. CF extracts coarse and fine granularity features from the original image, achieving macro-level observation of the object's position and micro-level assessment for optimal grasping rectangles. FR repairs diverse granularity features to enhance object–background distinction. RA considers training sample correlations with an external attention mechanism. The main contributions of this paper are summarized as follows:

We propose a refined grasping detection network to fuse multi-scale visual features shown in Figure 1. Additionally, the feature refinement module enables the network to distinguish between objects and backgrounds more effectively. We introduce a novel residual attention module, which not only enables the network to focus on critical regions of the object but also extracts different features depending on different objects in a low-parameter manner. We validate CFRA on both Cornell and Jacquard datasets, as well as real-world applications. These results demonstrate that the CFRA network achieves refined detection with outstanding performance.

The rest of this article is organized as follows. Section 2 presents an overview of related work on grasping detection networks. Section 3 formulates the problem. Section 4 detailedly describes the CFRA network. In Section 5, we implement a series of experiments to demonstrate the superiority of the model CFRA. Finally, Section 6 concludes this work.

2. Related work

2.1. Grasping detection

In recent years, with the development of computer vision, various deep learning algorithms for grasping detection have emerged [12, 19, 20]. These methods typically input RGB images and employ convolution neural networks (CNN) to generate a series of grasping candidate rectangles. The optimal box is usually selected based on the quality score. For instance, Yu *et al.* [21] introduced the EfficientNet, strategically adjusting specific hyperparameters to enhance the performance of grasping detection. Kumra *et al.* [22] proposed a lightweight CNN to achieve grasping detection. However, these approaches still encounter challenges arising from the diverse shapes of objects. Regression-based grasping detection methods are effective solutions [23–25]. Cheng *et al.* [23] directly regressed and encode the grasping angles. Similarly, [24] treated grasping angle prediction as a regression classification task, combining center sampling and regression weights to improve grasping detection accuracy. Although these methods exhibit specific performance, they fail to consider different granularities of image features, potentially overlooking crucial information and leading to coarse grasping rectangles.

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

2.2. Coarse and fine features fusion

Preliminary feature extraction methods are crucial for grasping detection tasks. Earlier approaches primarily rely on CNN for feature extraction [12, 13], which always extracts coarse-grained features and neglects fined details in images. To improve the performance and efficiency of the model's feature extraction, some works employ pre-trained models such as ResNet [26], CLIP [27] as feature extractors to accelerate model convergence. For example, Wang *et al.* [28] used ResNet to extract the original image features and then located the key areas of the object based on the transformer framework, which increases network parameters and decreases inference speed. To address these limitations, researchers have shifted their focus toward different granularity features and critical grasping regions. Methods emphasizing the refinement of image features have emerged. Yu *et al.* [17] proposed an attention mechanism and selective kernel (SK) convolutional network to enable the network to not only fully focus on grasp regions but also flexibly adjust them based on the object's size. Zhai *et al.* [18] optimized and de-duplicate multi-scale feature maps, improving network detection accuracy and real-time performance. To reduce the limitation of the grasp detection rectangle measurement, Li *et al.* [29] introduced a Gaussian-guided training method and utilized a global–local feature fusion approach to direct the network's attention toward grasping regions. Cheng *et al.* [16] achieved pixel-level dense prediction of grasp poses, obtaining optimal grasping regions through a non-maximum suppression strategy to fully consider the features at each pixel level. However, the experimental results fail to satisfy the performance requirements in the real world.

2.3. Residual attention mechanisms

In the evolving field of computer vision, Bahdanau *et al.* [30] first introduced attention mechanisms to mimic human visual and cognitive systems. This innovation enables neural networks to autonomously learn and selectively focus on crucial parts of input data, suppressing irrelevant features and enhancing the model's robustness and generalization. Li *et al.* [31] integrated attention mechanisms into grasp detection, enabling each neuron to dynamically adjust its receptive field size according to multi-scale input information. This adaptation captures crucial features for accurate detection. However, this work is inspired by self-attention, resulting in high computational complexity. To address the $O(n^2)$ computational complexity of the self-attention mechanism and the neglect of sample correlations, Guo *et al.* [32] proposed an external attention method, which utilized only two concatenated MLP structures and memory units, reducing the computational complexity to $O(n)$ while implicitly considering correlations between different samples. In grasping detection, it is necessary to select different features based on the shape of objects. Liu *et al.* [33] improved the SOLOv2 instance segmentation model by incorporating the Channel Attention Module and Spatial Attention Module. Despite this, work enhances grasping detection accuracy but lowers the inference speed. Thus, balancing between minimizing network parameters and ensuring attention to crucial grasp regions remains a challenge.

3. Problem formulation

Grasping methods based on grasping rectangles have been extensively studied [13, 21, 34]. In this paper, we use it to describe the detection pose of the target object. The network inputs the RGB image $I \in \mathbb{R}^{c \times h \times w}$ and outputs the grasping detection results, which can be defined as follows:

$$G = \{Q, \cos(2\Phi), \sin(2\Phi), W\} \tag{1}$$

where Q denotes grasping quality, Φ denotes the gripper rotation angle, and W denotes gripper opening width. The size of Q, Φ, and W is the same as the input image. The optimal grasping pose $g = (x, y, \theta, w, q)$ can be formulated as follows:

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

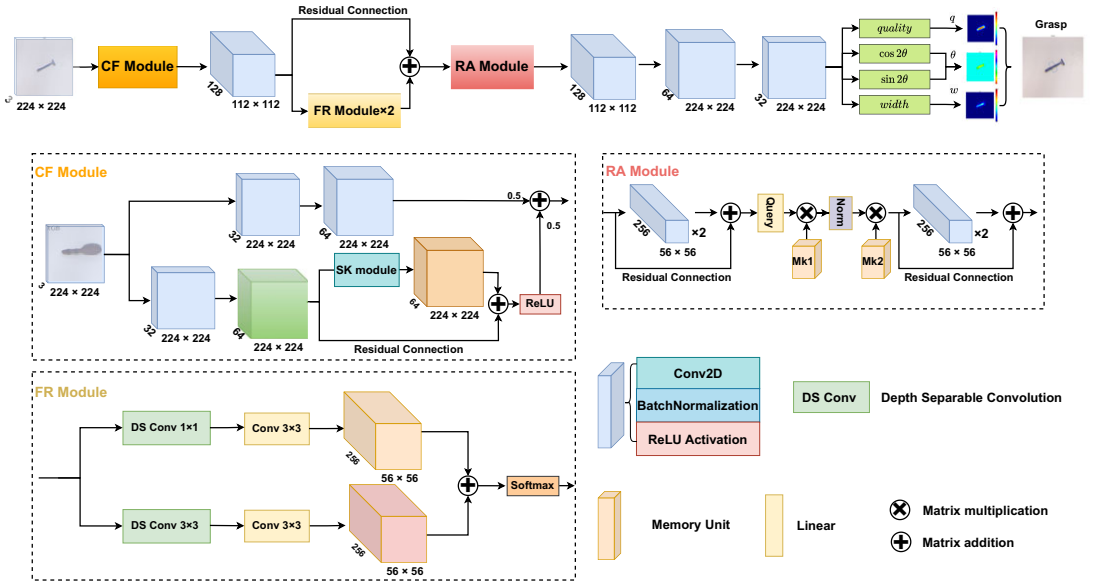


Figure 1. *The overview of CFRA network. Above: the network implementation process. Below: the network details of each module.*

$$
\begin{aligned}
(x, y) &= \arg\max(Q) \\
\theta &= \frac{1}{2}\arctan\frac{\sin(2\Phi)}{\cos(2\Phi)}\Big|_{(x,y)} \\
w &= W\,|_{(x,y)} \\
q &= Q\,|_{(x,y)}
\end{aligned}
\tag{2}
$$

where (x, y) represents the coordinates of the center of the grasping rectangles, w denotes the gripper opening width at this coordinate, θ stands for the gripper's rotation around the z axis at this coordinate, and q denotes the grasp quality. The height of the grasping box corresponds to the size of the grippers, which is typically known and does not require prediction [35].

After hand-eye calibration, the grasping pose identified in the image coordinate system can be converted to the robot's base coordinate system. The target pose is then sent to the robot controller, which performs path planning and grasps the object.

4. Approach

This paper proposes a grasping detection network named **c**oarse and **f**ine granularity **r**esidual **a**ttention (CFRA), which can be used for grasping detection in household scenes. The network architecture is shown in Figure 1, comprising three primary modules: coarse and fine granularity features fusion (CF), feature refinement module (FR), and residual attention module (RA). Within the CF module, we combine various convolutional kernels and SK module to extract multi-scale features, which ensures refined image feature extraction. Subsequently, in FR, we utilize a deep separable (DS) convolution network to refine the fused features, enabling the network to discern between background and object features. Lastly, we leverage the RA module, and the network selectively focuses on critical grasping regions, employing residual connections to reuse prior features, thereby enabling precise and efficient grasping detection.

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

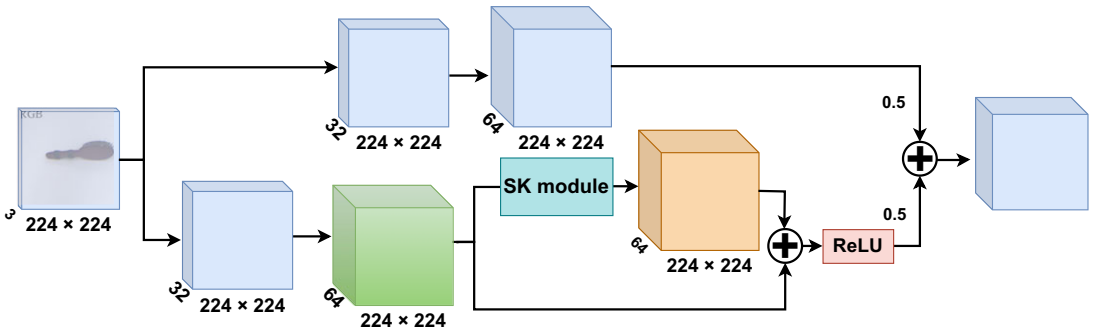


Figure 2. *The architecture of coarse and fine granularity features fusion module.*

4.1. Coarse and fine features fusion module

In practical grasping detection, smaller objects often demand smaller convolutional kernels to extract features, while larger objects benefit from slightly larger kernels to capture crucial edge information. Therefore, we use two branches to extract features of the input image, as illustrated in Figure 2. The upper-level branch of the network extracts coarse-grained features, employing a tandem of large-sized 9×9 and 7×7 convolutional kernels to extract coarse-grained feature $O = T(I)$ from the input image $I \in \mathbb{R}^{c \times h \times w}$ $(3 \times 224 \times 224)$. Simultaneously, the lower-level branch focuses on fine-grained feature extraction, utilizing two layers of smaller 5×5 and 7×7 convolutional kernels to capture finer features $\tilde{O} = T'(I)$. Here, T and T' denote feature transformations after two convolutional layers. To accurately predict grasping rectangles, we introduce the SK module [17] to extract features further, benefiting from its adaptability in adjusting receptive field sizes to handle multi-scale features. Besides, we introduce residual connections to reuse finer features $\tilde{O}$, which can also mitigate potential issues such as gradient vanishing and dimensionality errors associated with increased network depth. Finally, fuse the feature, as shown in the following:

$$E = \tilde{O} \oplus Conv(sk(\tilde{O})), \tag{3}$$

where *sk* represents the result of the feature map obtained from the *SK* module. The initial fused features are represented by E, which is processed through the ReLU function and is finally fused with coarse-grained feature O, and U represents the final fused feature as follows:

$$U = 0.5 \times O + 0.5 \times \mathrm{ReLU}(E). \tag{4}$$

Thus, realizing different granularity feature extraction from the input image facilitates future grasping detection work.

4.2. Feature refinement module

The coarse and fine granularity features extracted may contain redundant information, particularly in scenarios where the object shares similar colors with the background or possesses a regular shape. In such cases, the output features of the CF might be difficult to distinguish them. To address this challenge, we utilize FR to suppress the weight of background features, enabling the network to discern between the background and the object features. This module takes features previously output features U by the CF as input, as depicted in Figure 3.

We also use a dual-branch network structure to process feature U. First, We use two depth-wise separable convolutions in the two branches to adjust the weights of the output features U, with convolution kernels of 1×1 and 3×3, respectively. Subsequently, we utilize two 3×3 and 1×1 kernel sizes convolution in two branches to refine features further. For DS convolutions with a 1×1 kernel size, we utilize a 3×3 convolutional kernel; for DS convolutions with a 3×3 kernel size, we apply a 1×1 convolutional kernel. This facilitates a reduction in the network's parameter, consequently improving

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

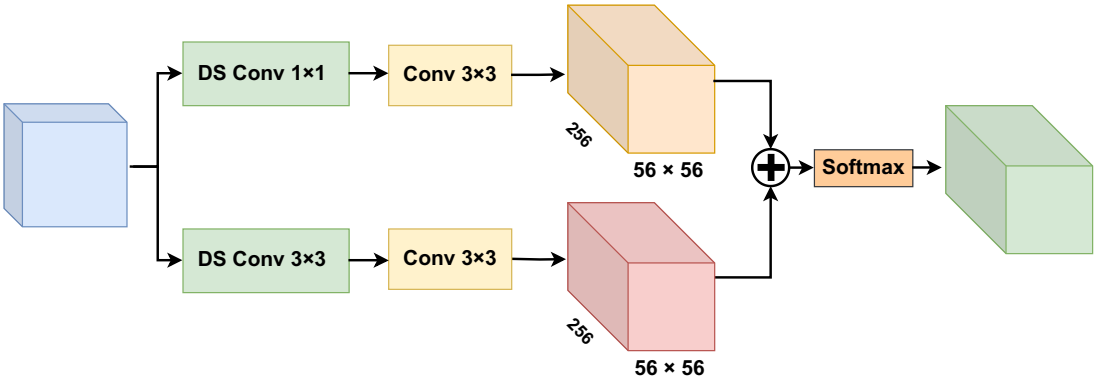


Figure 3. *The structure of feature refinement module.*

inference speed [18]. Additionally, leveraging alternative convolutional kernels enables the network to capture features across different scales adaptively. The formula is as follows:

$$\begin{aligned} F_1 &= C_3(DS_1(F_{local})), \\ F_2 &= C_1(DS_3(F_{local})), \end{aligned} \tag{5}$$

where C_i denotes a convolution operation, i represents the kernel size of the convolutional layer, and DS_j denotes depth separable convolution, with j indicating the kernel size of the DS convolutional layer and representing the output feature map. The extracted features undergo summative fusion and use the softmax function to be normalized, as follows:

$$F = SoftMax(F_1 + F_2), \tag{6}$$

where F represents the features obtained by FR. The augmentation of refinement modules enhances the network's capacity to perceive the object, allowing for the introduction of additional contextual information. However, this also results in increased parameters. Conversely, fewer refinement modules minimize the impact on the fused features. Consequently, we chose two FR layers.

4.3. Residual attention module

In grasping detection, deeper networks yield richer features, but training becomes more challenging due to potential issues like gradient vanishing and explosion. Residual connections are often employed to address these challenges. Previous studies, such as Kumra *et al.* [19], utilized five residual connections to identify grasping detection rectangles. However, each residual connection layer encompasses 590,592 substantial parameters, significantly limiting the network's prediction speed. Furthermore, attention mechanisms effectively guide the network's focus on critical grasping regions. Nevertheless, self-attention mechanisms entail a computational complexity of $O(n^2)$ and overlook training sample relationships. Hence, inspired by the external attention, we propose an RA module, as depicted in Figure 4. This module comprises solely two residual connection layers and one external attention layer. The first residual connection layer takes in the output of FR module to further extract object features. The external attention layer uses the attention mechanism to focus the network on the key areas of the object, and the last residual connection layer ensures that the image features are not lost. Thus, it not only ensures network training well but also aids in directing the model's attention toward crucial regions. Additionally, it considers feature extraction disparities for objects of various sizes and shapes during the detecting process [32]. The following is a detailed introduction to the external attention module.

External attention consists of two additional, small, learnable, and shared memory units. It is realized through two cascaded linear layers and two normalization layers. Its computational complexity is roughly equivalent to a 1×1 convolution, significantly lower than a single residual module.

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

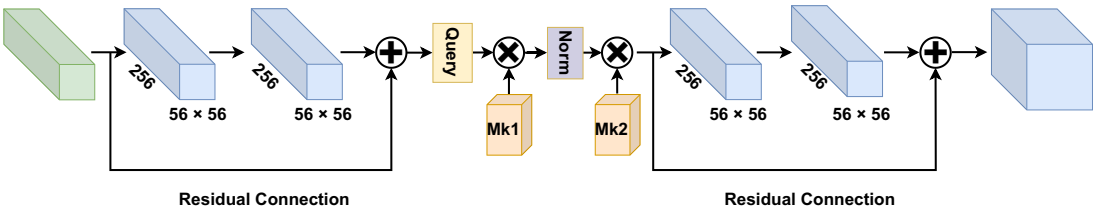


Figure 4. *The structure of residual attention module.*

External attention initially computes the affinity between the query and an externally learnable key memory to generate an attention map. Subsequently, this attention map refines the feature map by element-wise multiplication with another externally learnable value memory. Notably, the memory units contain fewer elements than the input features, resulting in computational complexity linearly related to the input's element count. Both memory units undergo computation through linear layers and are optimized via backpropagation. They operate independently for individual samples while being shared across the entire dataset, serving as a robust regularization mechanism that enhances the attention mechanism's generalization capabilities. The primary goal is to learn discriminative features prevalent across the dataset, capturing rich information while filtering out irrelevant details in other samples. For instance, when detecting elongated objects, where optimal grasp rectangles tend to align parallel to the object's edges, multiplying the memory unit with the original features accentuates edge features, facilitating the grasping rectangles parallel to the elongated object, thereby imposing precise constraints.

External attention calculates the attention between input pixels and external storage units as follows:

$$\begin{aligned} A &= (\alpha)_{i,j} = \mathrm{Norm}(FM^T), \\ F_{out} &= AM, \end{aligned} \tag{7}$$

where i,j is the similarity between the i-th feature and the j-th row of M, respectively, with M being an independently learnable parameter matrix acting as a memory for the training dataset. A denotes the attention map inferred from learned dataset-level prior knowledge. After normalization, A is utilized to update the input features within M based on their similarity scores according to A.

In practice, we employ two distinct memory units, M_k and M_v, serving as keys and values to enhance the network's capacity. The computational formula is as follows:

$$\begin{aligned} A &= \mathrm{Norm}(FM_k^T), \\ F_{out} &= AM_v. \end{aligned} \tag{8}$$

In self-attention, softmax is applied to normalize the attention map. As the attention map is sensitive to the scale of the input features, we adapt double normalization, independently normalizing rows and columns as follows:

$$\begin{aligned} (\tilde{\alpha})_{i,j} &= FM_k^T, \\ (\hat{\alpha})_{i,j} &= \exp(\tilde{\alpha}_{i,j}) / \sum_k \exp(\tilde{\alpha}_{k,j}), \\ \alpha_{i,j} &= \hat{\alpha}_{i,j} / \sum_k \hat{\alpha}_{i,k}. \end{aligned} \tag{9}$$

After the residual attention module, the image size is reduced to 56×56. To more effectively extract the coordinates of the grasp rectangles center, the gripper opening width w, the angle of rotation θ around the z axis of the gripper, and the grasp quality q, we employ transposed convolution operations to upsample the image.

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

4.4. Loss function

We chose the smooth L1 loss as a loss function, defined as:

$$L(G_i, \hat{G}_i) = \frac{1}{n} * \sum^{k} z_k, \tag{10}$$

where z_i is computed by the following formula:

$$z_k = \begin{cases} 0.5(G_{i_k} - \hat{G}_{i_k})^2, & if \left| G_{i_k} - \hat{G}_{i_k} \right| < 1 \\ \left| G_{i_k} - \hat{G}_{i_k} \right| - 0.5 & \text{otherwise} \end{cases}, \tag{11}$$

where $G_i \in \{q, \theta, w\}$ represents the predicted values generated by the network and $\hat{G}_i$ is the ground truth. The total loss of the CFRA network is:

$$L = \lambda_1 z_q + \lambda_2 z_\theta + \lambda_3 z_w, \tag{12}$$

where z_q, z_θ, and z_w correspond to the loss functions for quality, angle, and width, respectively. All of these are smooth L1 losses. This paper selects λ_1 as 1.2, and λ_2 and λ_3 are both set to 1.

5. Experiments and results

5.1. Dataset

a) Cornell

The Cornell dataset comprises 1035 RGB-D images and 240 objects. Each image corresponds to a single object, with 5110 positive grasps and 2909 negative grasps annotated. It is not enough to train the network using only these images, as deep learning networks demand extensive data for robust performance, particularly to mitigate overfitting risks. Thus, we apply preprocessing during the training. Initially, images are cropped to a size of $3 \times 224 \times 224$ pixels. Subsequently, the dataset is augmented by incorporating random rotations and random zoom. For images subjected to rotation and width adjustments, the ground truth for the grasp region is set to the rotation angle to ensure accurate prediction during training.

b) Jacquard

The Jacquard dataset comprises 54,000 RGB-D images and 11,000 objects, with each image annotated with a series of positive grasp rectangles. In total, there are approximately 1.1 million grasp rectangles. In contrast to the Cornell dataset, the Jacquard dataset is characterized by a sufficiently large number of images that do not require data augmentation.

5.2. Implementation details

The CFRA network is trained on a single NVIDIA 3090 24 GB GPU for both the Cornell and Jacquard datasets. The experiments are conducted on Ubuntu 18.04 and PyTorch 1.13 with CUDA version 11.2. The optimizer is Adam, with a learning rate of 1e-4, and the number of epochs is set to 50. The batch size for the Cornell dataset is configured as 8, whereas, owing to the larger scale of the Jacquard dataset, a batch size of 16 is utilized during training. During the training phase, 90% of the dataset is allocated for training, while the remaining 10% is set aside for testing to assess the network's performance.

5.3. Evaluation metric

Building on prior research [36], we utilize the commonly adopted Jacquard index and Angle threshold as evaluation metrics to ensure fair comparisons. A predicted grasp candidate is considered correct under the following conditions:

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

Table I. *Comparison results on the Cornell dataset.*

Algorithm	IW (%)	OW (%)	Speed (ms)
SAE, struct. reg [12]	73.9	75.6	1350
AlexNet, MultiGrasp [15]	88.0	87.1	76
GRPN [37]	88.7	-	200
GG-CNN [20]	73.0	69.0	19
ResNet-50 [22]	89.2	88.9	103
ZF-net [38]	93.2	89.1	-
GraspNet [39]	90.2	90.6	24
ROI-GD [34]	93.6	93.5	40
MultiGrasp, ResNet-50 [14]	96.0	96.1	120
FCGN, ResNet-101 [13]	97.7	96.6	117
FCNN [40]	96.6	95.4	20
GR-ConvNet [19]	91.5	95.5	20
SKGNet [17]	93.2	97.7	35
Ours	**94.4**	**98.9**	**26**

1) The difference between the predicted grasp angle and the ground truth grasp angle is within 30°.
2) The Intersection over Union (IoU) score between the predicted grasp and the ground truth grasp is greater than 0.25, as defined by the following formula:

$$J(g_p, g_t) = \frac{|g_p \cap g_t|}{g_p \cup g_t}. \tag{13}$$

5.4. Results and analysis

For comprehensive performance comparisons with related works, both image-wise (IW) and object-wise (OW) evaluations are conducted on the Cornell and Jacquard datasets.

1) Image-wise evaluation (IW): All images in the dataset are randomly split into training and validation sets at 9:1. While objects in the validation set may have been seen during training, their positions and orientations are randomized. The IW evaluation aims to assess the network's ability to predict objects in different poses.
2) Object-wise evaluation (OW): All images are split based on object instances, with objects used in testing not present during the training process. Object-wise split aims to evaluate the network's generalization ability when faced with new objects.

a) Results on Cornell dataset

The CFRA network is trained and tested using both IW and OW. The evaluation indicators are as described in Section 5.3. A comparative analysis with previous grasp detection networks under the same experimental conditions is presented in Table I. The input to the CFRA network is RGB images, and it achieves 94.4% and 98.9% detection accuracy in IW and OW, respectively. The processing speed reaches 26 ms. CFRA represents a 3% improvement compared to GR-ConvNet. Compared to the state-of-the-art SKGNet, the CFRA network demonstrates a 1% increase in detection accuracy and faster processing speeds, highlighting superior performance. Additionally, compared to FCNN, the CFRA network exhibits a 2% improvement in OW evaluation, making it more suitable for diverse object grasping detection. The experimental results underscore the effectiveness of the CFRA network in both accuracy and efficiency aspects.

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

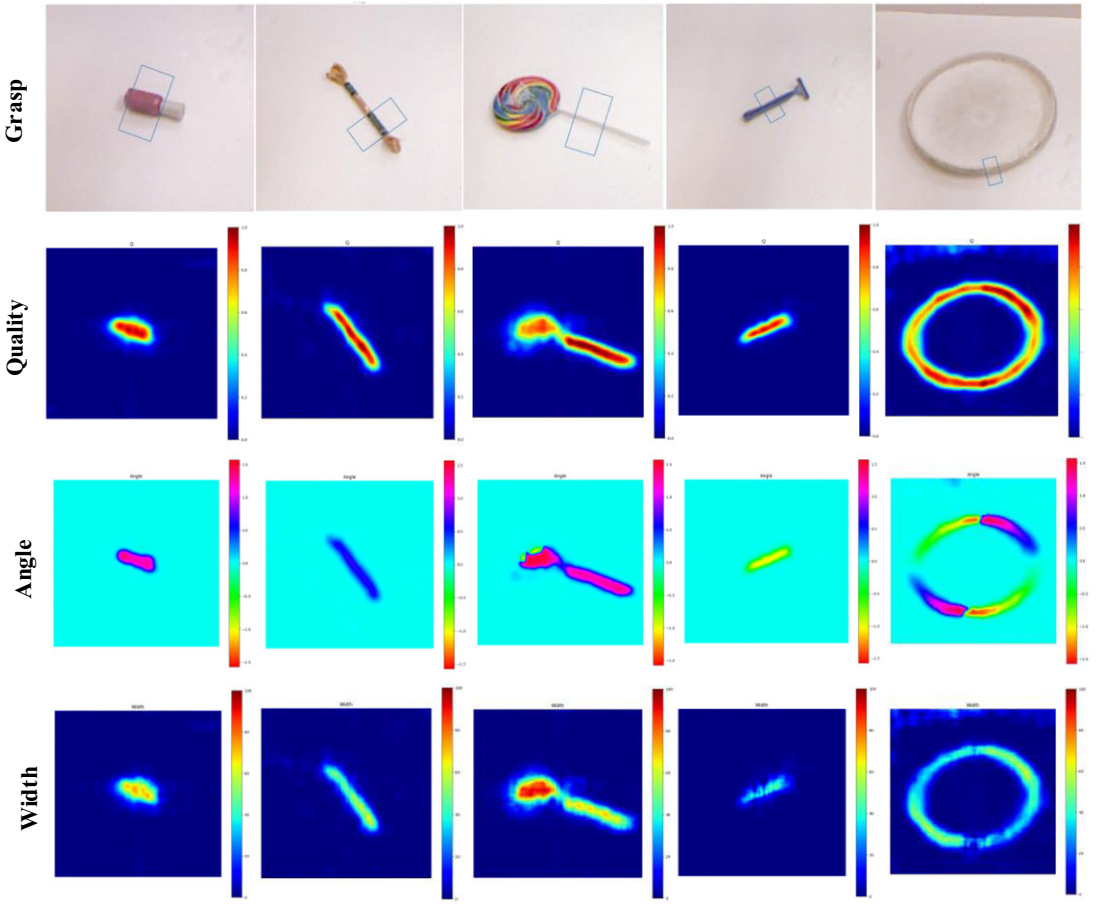


Figure 5. *Grasp detection results of CFRA on the Cornell dataset. The first row is the best grasp rectangle predicted by CFRA. The second to the last rows are the images of grasp quality, rotation angle, and opening width, respectively.*

The visual results of the CFRA network's detection on the Cornell dataset are illustrated in Figure 5. The first row presents the grasp rectangles predicted by the CFRA network, demonstrating its ability to accurately predict optimal grasp positions for regular objects and irregular shapes, like lollipops and circular objects. The second row displays visualizations of grasp quality, showcasing the CFRA network's proficiency in extracting image features and distinguishing between objects and backgrounds. The third and fourth rows depict predicted grasp angles and widths, demonstrating the CFRA network's accurate predictions of rectangles of various sizes and shapes on the Cornell dataset. The visualizations affirm the network's capability to handle diverse grasp scenarios and highlight the effectiveness of the CFRA model.

To facilitate a comprehensive demonstration of network detection performance, we compare the detection results with GR-ConvNet and SKGNet, as shown in Figure 6. Notably, the objects being detected are unseen during training. From the results, it is evident that when predicting the grasp for a toothbrush, both GR-ConvNet and SKGNet exhibit misalignment in predicted grasp rectangles. Moreover, in terms of grasp quality, only GR-ConvNet and the CFRA network demonstrate the ability to effectively differentiate between objects and the background. When predicting the grasp for a chalk, SKGNet fails to accurately predict the grasp rectangle, and GR-ConvNet predicts rectangles that are noticeably larger. Consequently, the CFRA network not only effectively distinguishes between

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

Table II. *Comparison results on Jacquard dataset.*

Algorithm	Accuracy (%)	Speed (ms)
Jacquard [41]	74.2	-
GG-CNN [20]	84.0	19
FCGN, ResNet-101 [13]	91.8	117
GR-ConvNet-RGB [19]	91.8	20
Efficient Grasping-RGB [35]	91.6	16
SKGNet [17]	93.2	35
Ours	**94.2**	**26**

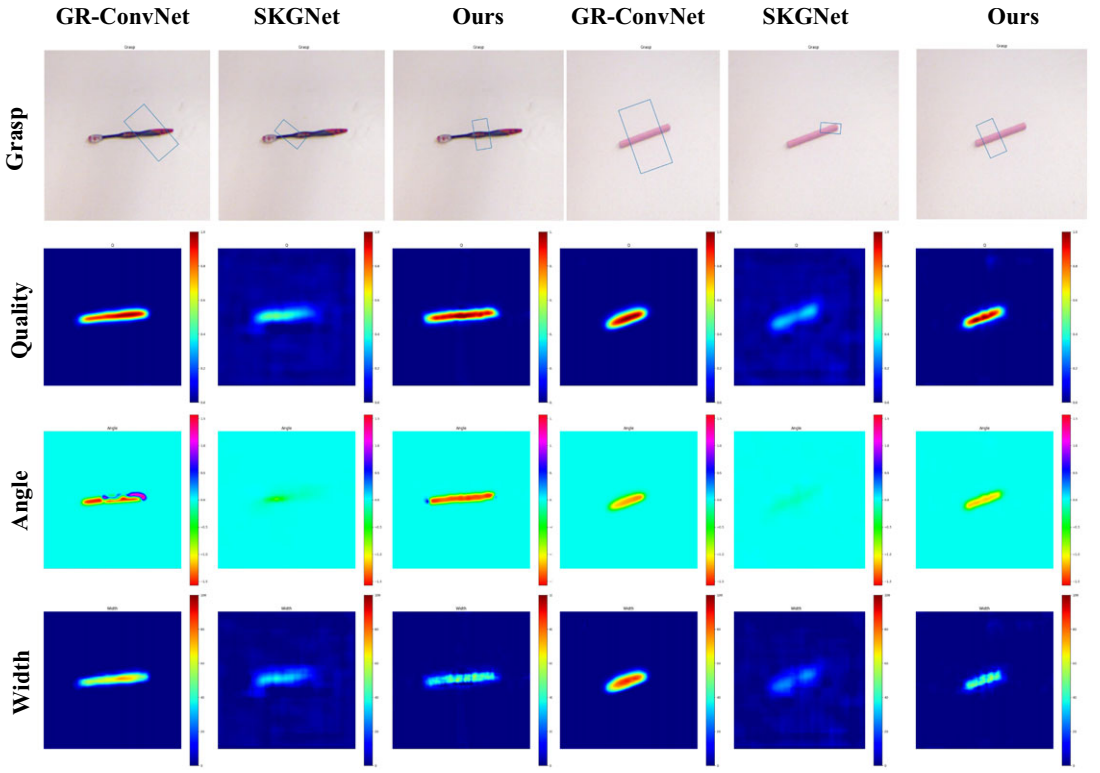


Figure 6. *Comparison studies on the Cornell dataset. The first row is the best grasp rectangles predicted by the networks. The second to the last rows are the images of grasp quality, rotation angle, and opening width.*

background and objects but also predicts reasonably sized grasp rectangles, showcasing its superior performance in both object recognition and grasp prediction.

b) Results on Jacquard dataset

We also validate CFRA on the Jacquard dataset. The input images are resized to 300 × 300, resulting in a detection accuracy of 94.2%. Upon reproducing FCGN, GR-ConvNet-RGB, Efficient Grasping-RGB, and SKGNet, the CFRA network achieves an accuracy improvement of over two percentage points compared to the current state of the art, as illustrated in Table II.

Visualization of partial detection results on the Jacquard dataset is presented in Figure 7. Considering the varying sizes and shapes of all objects, the CFRA network demonstrates excellent performance in

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

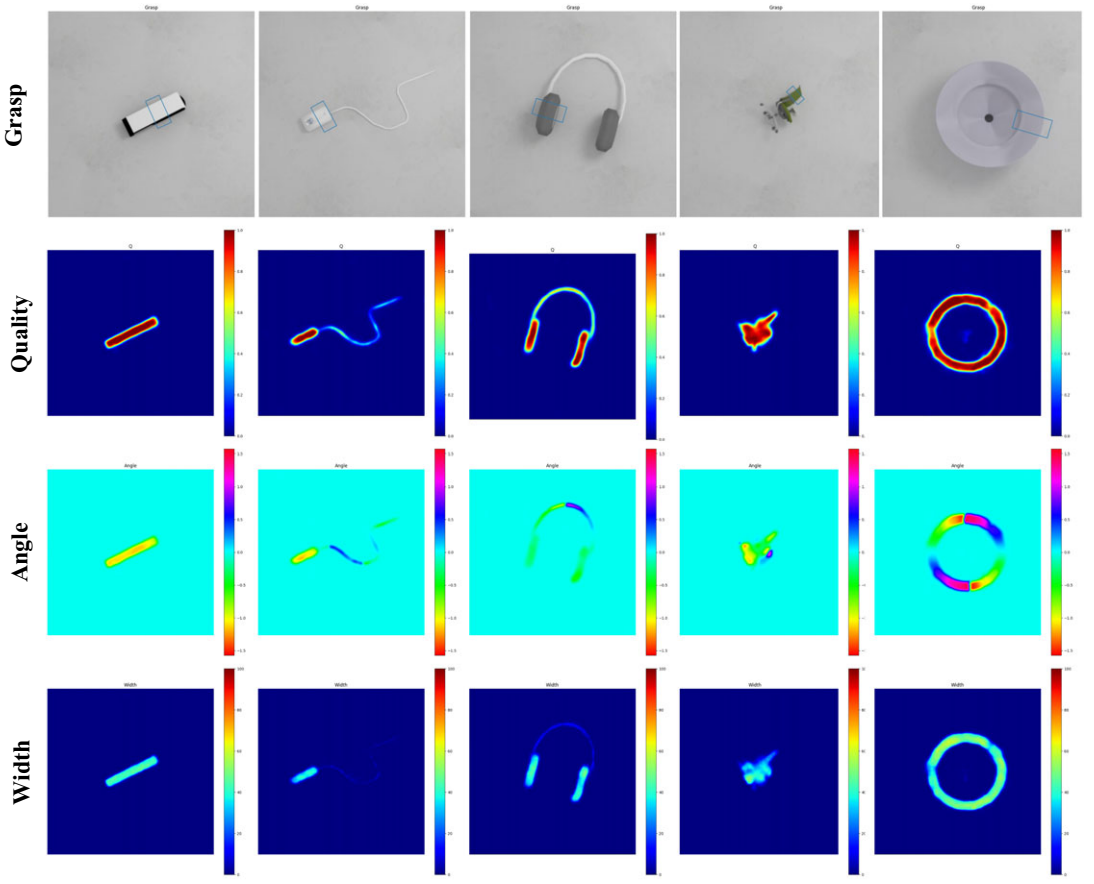


Figure 7. *Grasp detection results of CFRA on the Jacquard dataset. The first row is the best grasp rectangle predicted by CFRA. The second to the last rows are the images of grasp quality, rotation angle, and opening width, respectively.*

terms of grasp quality, grasp angle, grasp width, and position. The network accurately locates the target objects and precisely predicts optimal grasp positions, showcasing its effectiveness in handling diverse objects and grasp scenarios.

To illustrate the predictive capabilities of the CFRA network on the Jacquard dataset further, we conduct a comparative analysis with GR-ConvNet and SKGNet. Several objects from the Jacquard dataset are selected for visualization, as depicted in Figure 8. Notably, when predicting the grasp for scissors, SKGNet's predicted grasp rectangle fails to facilitate a successful grasp, while GR-ConvNet's predicted grasp rectangle is noticeably larger. In contrast, the CFRA network demonstrates a more refined grasp rectangle prediction. For the prediction of small building blocks, only the CFRA network accurately predicts the optimal grasp position, while the other two networks fall short. These visualized results demonstrate the superiority of the CFRA network in accurately predicting optimal grasp positions, especially for challenging objects such as scissors and small building blocks.

5.5. Experiments in clutter scenes

Real-world scenarios are typically complex and seldom involve only a single object [42]. To validate the object grasping detection capabilities of the CFRA network in complex scenes, we test the performance of CFRA, GR-ConvNet, and SKGNet on a multi-object dataset. We utilize the dataset proposed in [14], which is characterized by cluttered scenes with multiple objects in each image. The detection results

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

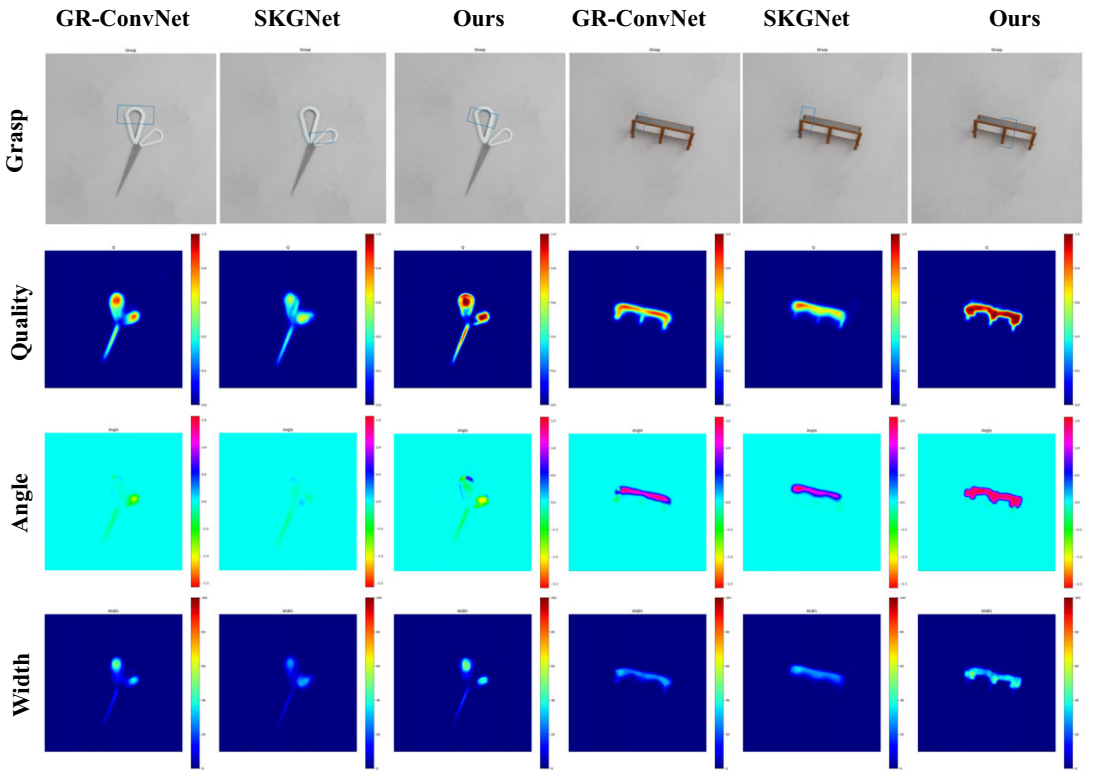


Figure 8. *Comparison studies on the Jacquard dataset. The first row is the best grasp rectangle predicted by CFRA. The second to the last rows are the images of grasp quality, rotation angle, and opening width, respectively.*

of the CFRA, GR-ConvNet, and SKGNet networks are presented in Figure 9. In multi-object grasp detection, the CFRA network accurately predicts grasp rectangles for each object, enabling the robotic arm to perform accurate grasping. In contrast, GR-ConvNet accurately predicts grasp rectangles for some regularly shaped objects but struggles with circular objects like tape and cups, producing tilted grasp rectangles that hinder successful grasping. For SKGNet, the predicted grasp rectangles are slightly larger, as observed in the case of elongated yellow blocks or a mouse. For circular objects like tape and cups, the predicted grasp rectangles have inaccurate center positions, preventing effective grasping. These results highlight the superior performance of the CFRA network in multi-object grasp detection, especially in scenarios involving diverse shapes and objects.

5.6. Ablation experiments

In this section, we validate the effectiveness of the three core modules: coarse and fine features fusion (CF), feature refinement (FR), and residual attention (RA). We conduct experiments by removing each of these modules individually and evaluating the network performance on the Cornell and Jacquard datasets. The experimental results are presented in Table III. The results indicate that the removal of the CF module alone leads to a 6.8% accuracy decrease on the Cornell dataset and a 1% decrease on the Jacquard dataset. When the LR module is removed individually, the accuracy drops by 7.9% on the Cornell dataset and 3.3% on the Jacquard dataset. Notably, removing the RA module alone results in a substantial accuracy decrease of 27% on the Cornell dataset and 8% on the Jacquard dataset. These findings demonstrate the significant contributions of each module throughout the entire

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

Table III. *Ablation experiments results.*

CF	LR	RA	Cornell (%)	Jacquard (%)	Speed (ms)
	√	√	92.1%	93.2%	22
√		√	91.0%	91.5%	25
√	√		71.9%	86.2%	23
√	√	√	**98.9%**	**94.2%**	**26**

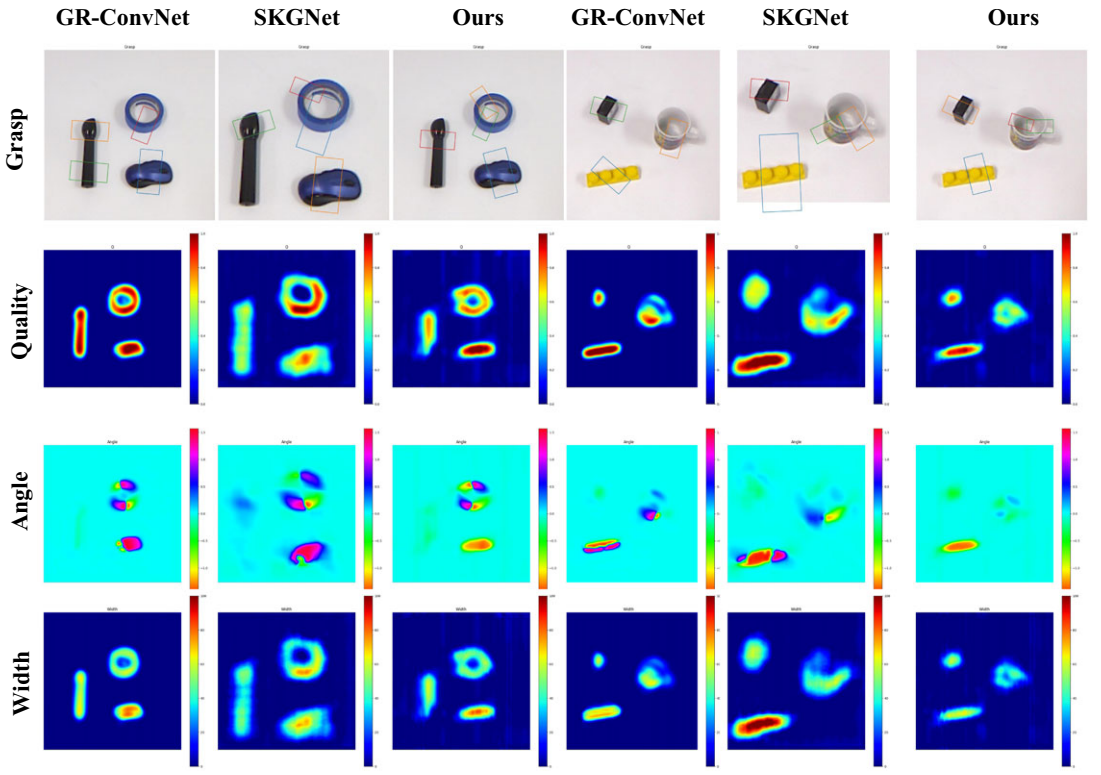


Figure 9. *Comparison studies on the Clutter dataset. The first row is the best grasp rectangles predicted by the networks. The second to the last rows are the images of grasp quality, rotation angle, and opening width.*

network training process, with the RA module having the most substantial impact on the network's performance.

5.7. Experiments in real world

In this experiment, we establish a robot grasping experimental platform to validate the performance of the grasp detection network CFRA in the real world, as illustrated in Figure 10. The platform comprises a UR3e robotic arm, a Robotiq140 gripper, and a Realsense D435i camera. On the right side, it displays the objects used during real-world experiments.

In real-world experiments, all objects are randomly placed, and the robot attempts to grasp the objects based on the predicted grasp rectangle with the highest quality score. The camera underwent precise hand-eye calibration. The robot makes 100 attempts at grasping to evaluate real-world grasping performance. The quantitative results are presented in Table IV. The "speed" column represents the time taken

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

Table IV. *Comparison results on real world.*

Algorithm	Success rate (%)	Speed (ms)
MultiGrasp, ResNet-50 [14]	89 (89/100)	120
SAE, struct. reg [12]	89 (89/100)	1350
GG-CNN [20]	92 (110/120)	19
GR-ConvNet [19]	95.4 (334/350)	20
SKGNet [17]	96 (96/100)	35
Ours	**98 (98/100)**	**26**

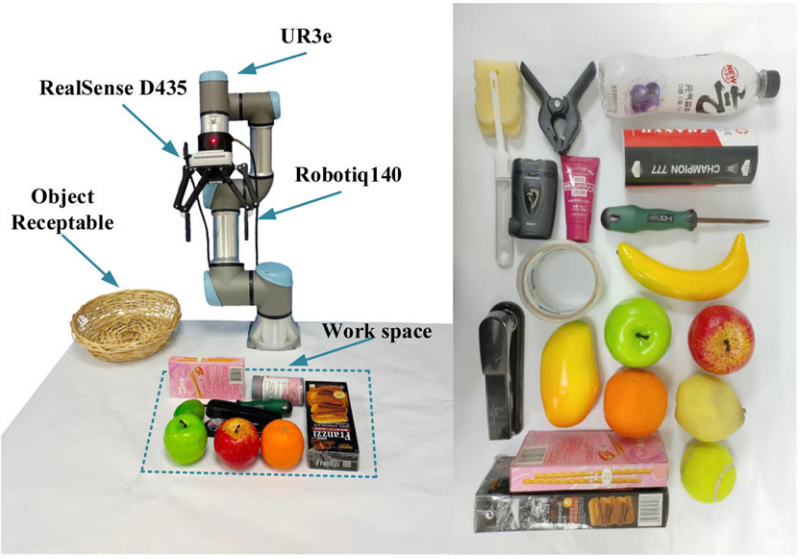


Figure 10. *Overview of the robotic grasping platform and objects from real world.*

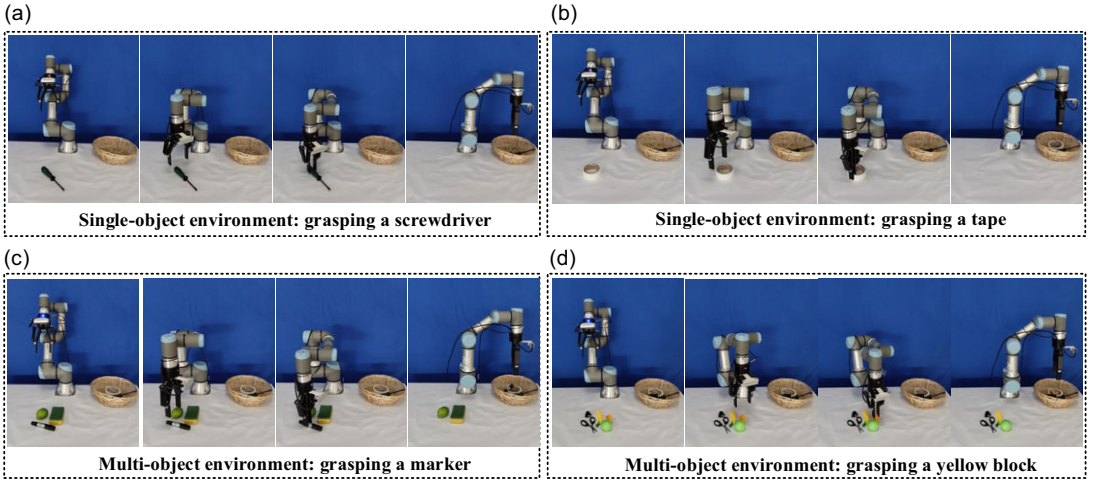


Figure 11. *Screenshots of robotic grasping experiments. (a) and (b) illustrate the grasp processes for a single-object scenario involving a screwdriver and tape. (c) and (d) illustrate the grasp processes for a multi-object scene, involving a marker and yellow block.*

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

for the neural network to predict the grasp rectangles. According to the table, the CFRA model achieves a grasp success rate of 98%, outperforming existing models. The results underscore the effectiveness of the CFRA model in real-world grasping scenarios.

The qualitative results are illustrated in Figure 11, where (a) and (b) illustrate the process of grasping and placing a screwdriver and tape in a single-object environment. Figures (c) and (d) depict the process of grasping and placing a marker and building blocks in a multi-object environment. It is evident that the CFRA network successfully achieves both object detection and grasping in real-world scenarios.

6. Conclusion

In this paper, we introduced a novel grasp detection neural network, CFRA, which inputs RGB images in robotic work scenarios and outputs the grasping poses. The network mainly consists of three modules: coarse and fine granularity fusion module extracts refined features to enable network to predict objects of different sizes, feature refinement module helps the network differentiate between background and object features, and residual attention effectively reduces parameters and enhances detection speed. To demonstrate the generalization and robustness of the CFRA network, extensive experiments were conducted on Cornell and Jacquard dataset, achieving detection accuracy of 98.7% and 94.2%, respectively. Furthermore, the network is deployed on a real-world UR3e robot to validate its practical applicability, which achieves a grasping success rate of 98%. The comprehensive experimental results highlight the superior adaptability of the CFRA network.

Author contributions. Zhenwei Zhu and Saike Huang contributed equally to this work. Zhenwei Zhu and Saike Huang wrote the original draft and analyzed the experiments. Jialong Xie and Yue Meng organized the data curation. Chaoqun Wang and Fengyu Zhou reviewed and edited the manuscript.

Financial support. This work was supported in part by the Jinan City and University Cooperation Development Strategy Project under Grant JNSX2023012, the Jinan "20 New Colleges and Universities" Funded Scientific Research Leader Studio under Grant 2021GXRC079.

Competing interests. The authors declare that they have no competing interests.

Ethical approval. None.

References

[1] M. Dong and J. Zhang, "A review of robotic grasp detection technology," *Robotica* **41**(12), 3846–3885 (2023).
[2] R. Upadhyay, A. Asi, P. Nayak, N. Prasad, D. Mishra and S. K. Pal, "Real-time deep learning based image processing for pose estimation and object localization in autonomous robot applications," *Int. J. Adv. Manuf. Technol.* **127**(3), 1905–1919 (2023).
[3] Z. Zhou and S. Li, "Self-sustained and coordinated rhythmic deformations with SMA for controller-free locomotion," *Adv. Intell. Syst.* **6**(5), 2300667 (2024).
[4] F. Khadivar and A. Billard, "Adaptive fingers coordination for robust grasp and in-hand manipulation under disturbances and unknown dynamics," *IEEE Trans. Robot.* **39**(5), 3350–3367 (2023).
[5] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei and P. Abbeel. Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding. **In:** *2010 IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, AK, USA (2010) pp. 2308–2315
[6] C. He, L. Meng, Z. Sun, J. Wang and M. Q.-H. Meng, "Fabricfolding: Learning efficient fabric folding without expert demonstrations," *Robotica* **42**(4), 1281–1296 (2024).
[7] G. Du, K. Wang, S. Lian and K. Zhao, "Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: A review," *Artif. Intell. Rev.* **54**(3), 1677–1734 (2021).
[8] M.-T. Le and J.-J. J. Lien, "Robot arm grasping using learning-based template matching and self-rotation learning network," *Int. J. Adv. Manuf. Technol.* **121**(3), 1915–1926 (2022).
[9] A. Ramisa, G. Alenya, F. Moreno-Noguer and C. Torras. Using depth and appearance features for informed robot grasping of highly wrinkled clothes. **In:** *2012 IEEE International Conference on Robotics and Automation (ICRA)*, Saint Paul, MN, USA (2012) pp. 1703–1708

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

[10] L. Zhang and D. Wu, "A single target grasp detection network based on convolutional neural network," *Comput. Intel. Neurosc.* **2021**(1), 5512728 (2021).

[11] C. Wang, X. Chen, C. Li, R. Song, Y. Li and M. Q.-H. Meng, "Chase and track: Toward safe and smooth trajectory planning for robotic navigation in dynamic environments," *IEEE Trans. Ind. Electron.* **70**(1), 604–613 (2022).

[12] I. Lenz, H. Lee and A. Saxena, "Deep learning for detecting robotic grasps," *Int. J. Robot. Res.* **34**(4-5), 705–724 (2015).

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

[14] F.-J. Chu, R. Xu and P. A. Vela, "Real-world multiobject, multigrasp detection," *IEEE Robot. Autom. Lett.* **3**(4), 3355–3362 (2018).

[15] J. Redmon and A. Angelova. Real-time grasp detection using convolutional neural networks. **In:** *2015 IEEE international conference on robotics and automation (ICRA)*, Seattle, WA, USA (2015) pp. 1316–1322

[16] H. Cheng, D. Ho and M. Q.-H. Meng. High accuracy and efficiency grasp pose detection scheme with dense predictions. **In:** *2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France (2020) pp. 3604–3610

[17] S. Yu, D.-H. Zhai and Y. Xia, "SKGNet: Robotic grasp detection with selective kernel convolution," *IEEE T. Autom. Sci. Eng.* **20**(4), 2241–2252 (2022).

[18] D.-H. Zhai, S. Yu and Y. Xia, "FANet: Fast and accurate robotic grasp detection based on keypoints," *IEEE T. Autom. Sci. Eng.* **21**(3), 2974–2986 (2023)

[19] S. Kumra, S. Joshi and F. Sahin. Antipodal robotic grasping using generative residual convolutional neural network. **In:** *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA (2020) pp. 9626–9633

[20] 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)

[21] S. Yu, D.-H. Zhai and Y. Xia, "Egnet: Efficient robotic grasp detection network," *IEEE T. Ind. Electron.* **70**(4), 4058–4067 (2022).

[22] 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

[23] H. Cheng, Y. Wang and M. Q.-H. Meng, "Anchor-based multi-scale deep grasp pose detector with encoded angle regression," *IEEE T. Autom. Sci. Eng.* **21**(3), 3130–3142 (2023)

[24] Y. Wu, F. Zhang and Y. Fu, "Real-time robotic multigrasp detection using anchor-free fully convolutional grasp detector," *IEEE T. Ind. Electron.* **69**(12), 13171–13181 (2021).

[25] G. Ren, W. Geng, P. Guan, Z. Cao and J. Yu, "Pixel-wise grasp detection via twin deconvolution and multi-dimensional attention," *IEEE T. Circ. Syst. Vid. Technol.* **33**(8), 4002–4010 (2023).

[26] K. He, X. Zhang, S. Ren and J. Sun. Deep residual learning for image recognition. **In:** *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2016) pp. 770–778.

[27] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark. Learning transferable visual models from natural language supervision. **In:** Proceedings of the ACM Conference on *International Conference on Machine Learning (ICML)*, Vienna, Austria (2021) pp. 8748–8763

[28] S. Wang, Z. Zhou, B. Li, Z. Li and Z. Kan, "Multi-modal interaction with transformers: Bridging robots and human with natural language," *Robotica* **42**(2), 415–434 (2024).

[29] Y. Li, Y. Liu, Z. Ma and P. Huang, "A novel generative convolutional neural network for robot grasp detection on Gaussian guidance," *IEEE Trans. Instrum. Meas.* **71**, 1–10 (2022).

[30] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, (2014).

[31] X. Li, W. Wang, X. Hu and J. Yang. Selective kernel networks. **In:** *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA (2019) pp. 510–519

[32] M.-H. Guo, Z.-N. Liu, T.-J. Mu and S.-M. Hu, "Beyond self-attention: External attention using two linear layers for visual tasks," *IEEE T. Pattern. Anal.* **45**(5), 5436–5447 (2022).

[33] Z. Liu, J. Wang, J. Li, Z. Li, K. Ren, and P. Shi, "A novel integrated method of detection-grasping for specific object based on the box coordinate matching," *arXiv preprint arXiv:2307.11783*, (2023).

[34] 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)*, Macau, China (2019) pp. 4768–4775

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

[36] S. Wang, Z. Zhou and Z. Kan, "When transformer meets robotic grasping: Exploits context for efficient grasp detection," *IEEE Robot. Autom. Lett.* **7**(3), 8170–8177 (2022).

[37] H. Karaoguz and P. Jensfelt. Object detection approach for robot grasp detection. **In:** *2019 IEEE International Conference on Robotics and Automation (ICRA)*, Montreal, QC, Canada (2019) pp. 4953–4959.

[38] 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)*, Singapore (2017) pp. 1609–1614

[39] U. Asif, J. Tang and S. Harrer, "Graspnet: An efficient convolutional neural network for real-time grasp detection for low-powered devices," *IJCAI* **7**, 4875–4882 (2018).

[40] D. Park, Y. Seo, and S. Y. Chun, "Real-time, highly accurate robotic grasp detection using fully convolutional neural networks with high-resolution images," *arXiv preprint arXiv:1809.05828*, (2018).

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press

[41] A. Depierre, E. Dellandréa and L. Chen. Jacquard: A large scale dataset for robotic grasp detection. **In:** *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain (2018) pp. 3511–3516.
[42] N. Lu, Y. Cai, T. Lu, X. Cao, W. Guo and S. Wang, "Picking out the impurities: Attention-based push-grasping in dense clutter," *Robotica* **41**(2), 470–485 (2023).

Cite this article: Z. Zhu, S. Huang, J. Xie, Y. Meng, C. Wang and F. Zhou (2025). "A refined robotic grasp detection network based on coarse-to-fine feature and residual attention", Robotica **43**, 415–432. https://doi.org/10.1017/S0263574724001929

https://doi.org/10.1017/S0263574724001929 Published online by Cambridge University Press