Grasp Detection Based on Light-Weight Hierarchical Fusion Convolutional Neural Network

Ziang Xu
School of Artificial Intelligence and Automation, Beijing University Of Technology, Beijing, China.
xuziang@emails bjut.edu.cn

Abstract. This paper presents a light-weight Hierarchical Fusion Convolutional Neural Network (HF-CNN) which can be used for grasping detection. The network mainly employs residual structures, atrous spatial pyramid pooling (ASPP) and coding-decoding based feature fusion. Compared with the usual grasping detection, the network in this paper greatly improves the robustness and generalizability on detecting tasks by extensively extracting feature information of the images. In our test with the Cornell University dataset, we achieve 85% accuracy when detecting the unknown objects.

1. Introduction
With the development of robotics in recent years, there has been an increasing focus on the ability of robots to perform tasks and interact with the environment. Among these capabilities, robot grasping, especially for unknown objects, plays a critical role as the basic function of robots and is getting more and more attention. However, there are many problems in it, such as how to grasp dynamic objects with a high accuracy, how to detect the target objects among many objects, and how to improve the robustness and generalizability of the g function. Therefore, more and more researchers have devoted themselves to the study of related algorithms, and they have proposed approaches to try to solve the existing problems.

[1, 2] access the optimal gripping posture by calculating from the geometric relationship of objects. Although this method is able to perform a variety of grasping tasks, it encounters difficulties when facing objects with complex shapes. [3] proposed a way to capture objects by repeatedly extracting visual information and constantly reconstructing the point cloud. This method solves the problem of grasping for unknown objects to a certain extent, but it cannot adapt well to the grasping task in different scenes and has low generalizability. And it lacks the ability to select the grasping among multiple objects. [4] solved these problems by simplifying the gripper into a C-shape cylinder, which makes the efficiency of grasping improved. But this method still cannot improve the accuracy of object grasping very well.

All of the above solutions use mathematical theory base on geometric information of the object. Therefore, the accuracy is largely affected by the occlusion relationship of objects, ambient lighting and other conditions. They, meanwhile, are limited by the shape of objects, lack generalization performance, and are not highly robust. On the other hand, with the development of deep learning, the advantages of convolutional neural networks in object detection and feature extraction are constantly highlighted. Compared with traditional method [5, 6], Convolutional neural networks can better adapt to different environments with strong robustness, and can adapt to various shapes of objects with strong
generalizability. Convolutional neural networks also do not need to pre-select the features to be learned, and more importantly, they greatly improve the accuracy of object recognition and grasping. With the creation of various types of networks, this accuracy is still improving and is increasingly becoming an important part of the grasping function.

Therefore, convolutional neural networks are gradually becoming the choice of many researchers when faced with object recognition grasping problems. In [7] on this learning recognition method was used to design a convolutional neural network based on dilated convolution and was able to perform tasks such as static and dynamic object recognition and multiple object recognition. However, this convolutional neural network also has many problems. First, this neural network doesn’t have a high accuracy, and the convergence speed is slow, and the robustness of object recognition in different environments is low. Secondly, this network still has more parameters with limited accuracy and bad real-time performance.

![Fig. 1 The general view of Real Time Light-Weight Hierarchical Fusion Convolutional Neural Network (HF-CNN)](image)

In this paper, a real-time lightweight layer fusion convolutional neural network (Fig.1.1) is proposed. The advantages of this convolutional neural network are: first, the network adopts a network structure of residual module and atrous spatial pyramid pooling, which greatly improves the generalization performance of the network and has good performance in multiple object recognition. Secondly, the key feature information in the image is further extracted by an encoding-decoding based feature fusion approach. This method is good at improving the accuracy of object recognition and ensuring that the number of parameters is maintained at a low level.

I evaluate the performance of the network with Cornell's dataset which include 249 test set and we achieve 91.1% on It., showing an extremely high robustness and generalization performance.

2. Related Work
Posture generation for grasping unknown objects, how to obtain the object's pose is always an important issue. The correct pose can largely reduce the difficulty of the grasping task and increase the success rate of grasping. Generally speaking, there are two methods of pose generation. One is a mathematical geometric model-based pose acquisition algorithm. The second is the method based on deep learning.

For mathematical geometric models algorithms are used to determine the grasping strategy by the shape of the objects acquired in real space. The most important of them is the method of object features obtained visually which 3D point clouds method is one of them. [3, 4] are using this way. The 3D model...
of the object is constructed by combining the visual and point cloud methods. However, this method of simply applying the object model and determining the poses by geometric relationship method does not establish a good bridge between the real situation and the simulated scene. Only local information of the object can be obtained in the face of unknown objects, such as when there are occlusion relations, ambient lighting, etc. The ability to model unknown objects is greatly affected and leads to a decrease in accuracy.

Another very important method in mathematical geometric models is the bit-pose generation by means of object edge detection [8]. This method is based on the knowledge of image processing morphology, which identifies the pixel gradients of the object edges and the environment to distinguish the two and generate the corresponding poses. It solves to some extent the influence of the environment on the accuracy rate. It is able to maintain good accuracy and high robustness without obtaining complete information about the object or being influenced by the environment. However, this method is difficult to generate effective grasping strategies for complex objects.

The deep learning based method is very good at improving the accuracy of the capture as well as the accuracy rate. Of course, this method, based on existing data sets, has a very high accuracy rate in the face of learning objects, using images or models as input to the convolutional neural network, and then acquiring recognition features from them through deep learning methods. While ensuring high robustness and accuracy, it is still able to face the situation of unknown objects and unknown environment very well.

Therefore, this paper also uses the deep learning method to use the depth image as the input of the convolutional neural network for extracting features, which greatly improves the accuracy of the grasp recognition. It also ensures robustness and stability and has better performance on unknown objects.

Deep learning for the methods in this paper the deep learning effect directly determines how well the features are extracted, which greatly improves the accuracy of the grasp recognition. It also ensures robustness and stability and has better performance on unknown objects.

Deep learning for the methods in this paper the deep learning effect directly determines how well the features are extracted, which greatly improves the accuracy of the grasp recognition. It also ensures robustness and stability and has better performance on unknown objects.

Deep learning for the methods in this paper the deep learning effect directly determines how well the features are extracted, which greatly improves the accuracy of the grasp recognition. It also ensures robustness and stability and has better performance on unknown objects.

Deep learning for the methods in this paper the deep learning effect directly determines how well the features are extracted, which greatly improves the accuracy of the grasp recognition. It also ensures robustness and stability and has better performance on unknown objects.

Deep learning for the methods in this paper the deep learning effect directly determines how well the features are extracted, which greatly improves the accuracy of the grasp recognition. It also ensures robustness and stability and has better performance on unknown objects.

Deep learning for the methods in this paper the deep learning effect directly determines how well the features are extracted, which greatly improves the accuracy of the grasp recognition. It also ensures robustness and stability and has better performance on unknown objects.

3. Method

Grasp Point Definition
For the grasping problem we need to pay particular attention to the grasping pose, and in this paper the grasper is centered on the center point. For a grasper, its center point determines its relative position in 3D space, while the parameters of the grasper itself determine the grasping pose. Let \( L = (x, y, z) \) define a grasp, executed perpendicular to the x-y plane. The gripper’s center position \( p = (w, \phi) \), the opening and closing width of the gripper is w, and the rotation angle around the z-axis is \( \phi \). To evaluate the merit of a grasping pose, we define a contribution factor \( d \). The magnitude of \( d \) determines how much we approve of a grasping value. By adding the contribution coefficients to the functions corresponding to position and pose, we obtain the complete definition for a grasper \( (L, q, d) \in R^{3\times H\times W} \). In order to be able to capture the pose on the image, we translate the definition of the capture in 3D space to the plane. According to the basic principle of coordinate transformation. We obtain the basic internal and external parameters of the used camera. Two transformations are performed, namely, the transformation from the world coordinate system to the camera coordinate system, with the transformation matrix defined as \( T_{WC} \), and the transformation from the camera coordinate system to the image coordinate system, with the transformation matrix defined as \( T_{CT} \). Finally, for the purpose of pose identification at each point we then transform the image coordinate system to each pixel point, defining the transformation matrix as \( T_{IP} \). we set the plane pixel coordinates \( f = (u, v) \).

We refer to the set of grasps in the image space as the grasp map, which we denote

\[
\tilde{g} = (f, \tilde{p}, d)
\]

Also, we can obtain the transformation formula from the world coordinate system to the pixel coordinate system, which we denote

\[
\tilde{g}^T = T_{IP}T_{CT}T_{WC}g^T
\]

Further for a pixel in \( f \) we have \( P \) for the amount of pose \( \tilde{p} \) in the pixel coordinate system and \( D \) for the contribution factor \( d \) in the pixel coordinate system. This allows us to derive a pixel pose equation

\[
G = (P, D) \in R^{3\times H\times W}
\]

Let the input image be \( I \). We can determine that there is a certain correspondence between each input object and the coordinates \( G \) we get, and we define this correspondence as \( T \), then \( G = T(I) \). And the goal of our task is to find the \( G \) corresponding to the maximum value of \( D \) among all the pixel points.

Convolutional Neural Network

After determining the best pose we estimate this correspondence \( T \) by convolutional neural network. we can estimate the true value of the best pose \( G_t \) from this correspondence by the input image, the correspondence of the existing system is \( T_\theta \) and the true correspondence is \( T_t \), which we define

\[
G_t = \lim_{T \to \mathcal{N}_t} T(I)
\]

So the loss function is applied to approximate with the output, such that

\[
\theta = \arg\min_{\theta} \frac{1}{2} || G_t - T_\theta(I) ||^2
\]

Where \( G \) represents an estimate for the pixel pose. It contains two quantities in itself, the pose parameter \( P \) and the contribution coefficient \( D \) for each pixel point.

Where \( P \) is determined by the image input, which contains two quantities \( \Phi \) and \( W \in R^{H\times W} \). \( \Phi \) describes the relative angle of each pixel point. And \( W \) describes the width of each grabber at the time of grabbing.

And \( D \) is used to evaluate the contribution coefficient of each pixel capture, as long as it is a successful capture, the contribution coefficient is set to 1, and on the contrary if the capture fails, the contribution coefficient is set to 0.
Network structure the network (fig. 2) in this paper is mainly composed of two downsamples and two upsamples. In order to obtain a wider range of information at the target scale. We add a void pyramidal pooling structure in the middle of the residual link. In this structure, the input is subjected to a normal 3 × 3 convolution and three kinds of 3 × 3 null convolutions with sampling rates of \{6, 12, 18\}, respectively, in this paper. Each of them can capture information at different scales to further improve the network performance. The input is then subjected to one more adaptive pooling and convolution. Finally the features of the above outputs are fused by a 1 × 1 output convolution.

After that we further deepened the network by using a residual structure between each sampling \cite{13, 14}. Because as the number of layers of the network increases it will face many problems such as long and computationally intensive computation time and overfitting. Also even the problem of degeneracy phenomenon can occur. Therefore by adding the input value \(x_n\) with the output that goes through the network and taking it as the next input value, we can obtain the input-output relation for each time of the network residual structure as follows.

\[ x_{n+1} = x_n + F(x_n, W_n) \]

In this paper, we add a layer of convolution on the input side as well, so we can obtain further residual structure input-output relations.

\[ x_{n+1} = G(x_n, W_n) + F(x_n, W_n) \]

Where \(W_n\) represents the weight parameter of the residual network and \(\overline{W}_n\) is the parameter of the bypass network. \(x_n\) is the input of the previous session, and \(x_{n+1}\) is the output of this session and the input of the next session. \(f\) is the correspondence between the residual structure and the output, and \(g\) is the correspondence of the bypass (note the translation). Each residual structure contains two identical convolutional units, and the units consist of convolutional layers, normalization and activation functions. Each of these residual blocks has the same weight, while in this paper, the residual block of the network is set to 2.

This link solves the degradation problem well and improves the accuracy of the network at the same time. And this method increases the upper limit of the number of network layers that can be added.

4. Experimental Setup
The experimental environment of the computer used for training the network is Windows 10, with a CPU of i5 8300H and a graphics card NVIDIA GeForce GTX 1050ti to assist the computational process. The Python compiler is python 3.9.4, the acceleration library is CUDA 11.2, and the dependent deep
The learning framework is pytorch 1.8.1 (GPU version). In the training process, the Adam optimizer is used, the number of iterations is 50, and the batch size is set to 8.

The dataset used in this paper is the Cornell University dataset with 2049 data, of which 90% is chosen as the training set and 10% as the test set. We performed the same processing of the dataset as done in [11] on the dataset, transforming the images to a size of 300 × 300. And we increased the sample size by random rotation and cropping to generate a dataset containing 8840 depth maps.

We generate a rectangle in each image to calibrate the grasping device, and the pose of this rectangle represents the pose of the grasper. And we take one-third of the rectangle length as the valid region for grasping. Set the contribution factor $D$ of all image pixels in all valid regions to 1 and the others to 0. Set the angle $\Phi$ of the pixels in the valid region of the image to the angle $\Phi$ of the rectangle of the valid region with respect to the horizontal coordinates and set the value $W$ of the grabber width to the width $W$ of all pixels in the rectangle of the valid region. and use these three matrices as the labels of the dataset.

5. Experiments
We evaluated testing our network by designing several sets of related experiments. To ensure the validity of the experiments and the reliability of the experimental results. The experiments were all conducted in the same environment, using the same training set and dataset, for a total of 50 epoch.

To evaluate the basic performance of the network itself. We set the valid rectangle of the label of the number $i$ of image of the test set as $A_i$ and the valid rectangle generated by the recognition result as $B_i$, for the mIOU value of the model computed on the test set for each image.

$$mIOU = \argmax_i \frac{A_i \cup B_i}{A_i \cap B_i}$$

Where the $IOU$ value greater than 0.25 is recorded as a positive sample $P$ for a successful crawl recognition and vice versa as a negative sample $N$. Finally, the ratio of the number of successful crawls to the total number of crawls is taken as the accuracy $Acc$.

$$Acc = \frac{P}{P+N}$$

First we use GGCNN and GGCNN2 to compare with networks that have not undergone knowledge distillation. Secondly to verify that the network structure does not contain redundant links, we compare the performance of the base network containing four layers of convolutional two-layer upsampling with the residual structure, ASPP and both, respectively, and investigate the role each link plays in terms of performance as well as robustness.

5.1. Compare with the other network

Table 1. Compare of the accuracy rate and the network parameters on grasp detection based on different model.

| Method   | Parameters | mIOU  | Acc  |
|----------|------------|-------|------|
| GGCNN    | 62.42k     | 39.8% | 79.1%|
| GGCNN2   | 66.68k     | 43.2% | 84.3%|
| HF-CNN   | 60.69k     | 44.4% | 85.1%|
We compared our network with existing networks in terms of performance, etc. The GGCNN has a total of 62,420 parameters with a final maximum accuracy of 79.1% (197/249), which is low in relative accuracy despite the small number of parameters, while the GGCNN2 has a total of 66,676 parameters with a final maximum accuracy of 84.3% (210/249), while the network with the addition of residual structure, ASPP and feature fusion, the network has only 60,692 parameters and the final maximum accuracy is 85.1% (212/249).

It can be seen that our network has higher accuracy with fewer parameters in comparison, and converges faster and the loss decreases more rapidly. The advantages of the structure of the network proposed in this paper for hierarchical extraction of image features can be seen.

Also, after analysis, structurally speaking, the network in this paper has a smaller convolutional kernel, but at the same time has a deeper network and feature fusion. Therefore, the number of channels and the number of convolutional kernels are considered to contribute less in improving the performance of the network, and more parameters are added. Instead, the network depth and the breadth of feature fusion play a more critical role in this regard.

5.2. Compare with different structures

| Method                | Parameters | mIOU | Acc  |
|-----------------------|------------|------|------|
| Baseline              | 34.39k     | 37.7%| 71.1%|
| Baseline+Res          | 53.94k     | 43.1%| 82.7%|
| Baseline+ASPP+Res     | 60.69k     | 44.4%| 85.1%|

Fig. 3 the curve of the accuracy rate based on different model.

Fig. 4 the curve of the loss based on different model.

Fig. 5 the curve of the accuracy rate based on the baseline which combine with different module.

Fig. 6 the curve of the loss based on the baseline which combine with different module.
To illustrate the effectiveness of our network improvement links, we selected method 1: using a base network with a network structure containing four convolutions and two upsamplings, respectively. Method 2: Adding ASPP module to the network structure. Method 3: Adding a residual module to the network. Method 4: Adding both residual module and ASPP module to the network.

Observing the experimental results, we found that the base network parameters had a total of 34388, but the highest accuracy rate was 71.1% (177/249), and the network performance was very bad. We added the ASPP structure after we downsampled the network twice in (1) (Fig.). The resulting parameters totaled 41140, while the accuracy improved by 6.4% to 77.5% (193/249). In another experimental group (2) (Fig.), two layers of residual structure were added with the number of resblock in each layer, resulting in a total of 53,940 parameters, while the accuracy rate improved by 11.6% to 82.7% (206/249). The final network built in this paper, on the other hand, contains both the residual structure and the ASPP structure. The resulting parameters totaled 60692, while the accuracy improved by 14.0% to 85.1%.

By incorporating these structures, the network in this paper improves the overall performance of the network and improves the generalization and robustness of the network on the basis of increasing the number of few parameters. We can see that both this residual structure and ASPP module can improve the network performance very well. And in this paper, the hierarchical fusion of these two structures obtains a new network with better overall performance.

6. Conclusion
Deep learning is an important tool in the task of unknown object grasping. In this paper, the proposed layer fusion convolutional neural network algorithm successfully achieves the improvement of unknown object grasping recognition accuracy based on the low number of parameters by applying the residual structure, null space convolutional pyramidal pooling to 85.1%, and further applying the knowledge distillation method to increase the accuracy to 91.1%. And this method has high robustness and generalization performance. It provides an important idea of neural network light-weighting improvement, which has great value in practical applications.

References
[1] A.Makhal, F. Thomas and A. P. Gracia, "Grasping Unknown Objects in Clutter by Superquadric Representation," 2018 Second IEEE International Conference on Robotic Computing (IRC), 2018, pp. 292-299, doi: 10.1109/IRC.2018.00062.

[2] Y. Wang, J. Wang and W. Chen, "Grasp Planning Based On Scene Grasp Ability In Unstructured Environment," 2018 IEEE International Conference on Robotics and Biomimetics (ROBIO), 2018, pp. 1477-1482, doi: 10.1109/ROBIO.2018.8665076.

[3] Bohg J , Johnson-Roberson M , B León, et al. Mind the gap - robotic grasping under incomplete observation[C]/ 2011 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2011.

[4] Qujiang Lei, J. Meijer and M. Wisse, "A survey of unknown object grasping and our fast grasping algorithm-C shape grasping," 2017 3rd International Conference on Control, Automation and Robotics (ICCAR), 2017, pp. 150-157, doi: 10.1109/ICCAR.2017.7942677.

[5] Yang, J. Y. , et al. "A Novel Robotic Grasp Detection Technique by Integrating YOLO and Grasp Detection Deep Neural Networks." 2020 International Conference on Advanced Robotics and Intelligent Systems (ARIS) 2020.

[6] K. Du, J. Song, X. Wang, X. Li and J. Lin, "A Multi-Object Grasping Detection Based on the Improvement of YOLOv3 Algorithm," 2020 Chinese Control And Decision Conference (CCDC), 2020, pp. 1027-1033, doi: 10.1109/CCDC49329.2020.9164792.

[7] Morrison, D. , P. Corke , and J. Leitner , "Closing the Loop for Robotic Grasping: A Real-time, Generative Grasp Synthesis Approach." Robotics: Science and Systems 2018 2018.

[8] G. Huang, H. Lin and P. Chen, "Robotic arm grasping and placing using edge visual detection system," 2011 8th Asian Control Conference (ASCC), 2011, pp. 960-964.
[9] Y. Jiang, X. Li, M. Yu and Z. Bai, "Robotic Grasp Detection Using Light-weight CNN Model," 2020 Chinese Control And Decision Conference (CCDC), 2020, pp. 1034-1038, doi: 10.1109/CCDC49329.2020.9164710.

[10] N. Mustafa, J. Zhao, Z. Liu, Z. Zhang and W. Yu, "Iron ORE Region Segmentation Using High-Resolution Remote Sensing Images Based on Res-U-Net," IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium, 2020, pp. 2563-2566, doi: 10.1109/IGARSS39084.2020.9324218.

[11] Du X., Cai Y., Lu T., et al.,” A Robotic Grasping Method Based on Deep Learning”.” Jiqiren/Robot, 2017, 39(6):820-828 and 837.

[12] A. Jiang, N. Yan, F. Wang, H. Huang, H. Zhu and B. Wei, "Visible Image Recognition of Power Transformer Equipment Based on Mask R-CNN," 2019 IEEE Sustainable Power and Energy Conference (iSPEC), 2019, pp. 657-661, doi: 10.1109/iSPEC48194.2019.8975213.

[13] W. You, L. Chen and Z. Mo, "Soldered Dots Detection of Automobile Door Panels based on Faster R-CNN Model," 2019 Chinese Control And Decision Conference (CCDC), 2019, pp. 5314-5318, doi: 10.1109/CCDC.2019.8833343.

[14] Y. Liu, "An Improved Faster R-CNN for Object Detection," 2018 11th International Symposium on Computational Intelligence and Design (ISCID), 2018, pp. 119-123, doi: 10.1109/ISCID.2018.10128.