RGB-D Indoor Object Recognition Algorithm Based on Fusion Convolutional Neural Network

Decheng Wang1,*, Hui Yi1 and Feng Zhao1,2

1 Graduate School, Space Engineering University, Beijing, China
2 61618 Troops, Beijing, China

*Corresponding E-mail: 823062375@qq.com

Abstract. Aiming at the problems of insufficient network fusion and low detection efficiency in current object recognition using RGB-D images, a recognition algorithm based on the medium-level layer-by-layer fusion of dual-channel networks is proposed. First of all, the RGB and Depth networks are trained with ten labelled RGB-D indoor objects respectively, and then determine the fusion coefficients according to the identify accuracy of two types networks. Finally, two kinds of features are merged in convolutional layers step by step to obtain the final weights. By testing on the challenging NYU Depth v2 dataset, we found that the recognition accuracy of our method is 92.85%, and average detection time is 61.03ms per image. Through comparison experiments, we got the conclusion that average accuracy of the RGB-D layer-by-layer fusion network is 5.22% higher than that of the RGB network.

1. Introduction

In recent years, with the emergence of new RGB-D sensors (such as Kinect and Xtion, etc.), the application of RGB-D images in related fields has been greatly promoted. The combination of image depth information and RGB information opens a new window for solving basic problems of computer vision. Depth information provides useful additional information for scenes and target classifications of complex problems, and does not change with brightness and colour, making objects be better identified from the background. At present, the target recognition of fusing depth information has become a hot research topic in the field of computer vision. It is widely used in indoor robot visual recognition systems to realize the understanding and interaction of indoor complex scenes[1].

Socher et al.[2] first proposed a convolutional recurrent neural network (CNN-RNN) fusion model to learn RGB-D features in 2012. The original RGB-D images were directly put into CNN to obtain low-level features, and then these low-level features were put into RNN to obtain the effective high-level features, and finally the softmax method is used to obtain better classification performance. In the RGB-D target recognition algorithm based on hybrid structure proposed by [3], using the deep convolutional neural network (VGG-16) and the improved HONV descriptor Fisher Vector coding to respectively extract feature expression for the RGB and depth image of each frame, and achieved superior recognition performance. Hoffman et al.[4] proposed a medium-level fusion CNN of RGB and Depth information, and given a few types of mark depth images, realized the detection of RGB-D objects under weak supervision conditions. Xu et al.[5] proposed a new method for target recognition in RGB-D scenes. By using the shared weight strategy and the parameter-free correlation layer to jointly perform RGB-D object detection and region object recognition, through post-fusion, multi-
modal RGB-D object recognition effect is effectively improved. In [6], a new dual-stream convolutional network is proposed. After inputting RGB and depth images into two convolutional networks with the same structure to extract their respective features, the convolutional layer based on the optimal weights fused features of two networks, and finally the output is obtained through the fully connected layer.

In view of the current insufficient RGB-D network fusion and low detection efficiency, this paper proposes a two-channel network with layer-by-layer fusion. RGB and Depth CNN are trained respectively on the NYU Depth v2 dataset[7], and test the network after fusion. Finally we compared and evaluated quantitatively for the results.

2. Method

2.1. Convolutional neural network model structure
Convolutional neural network is excellent deep learning models in the field of image recognition and detection[8]. The paper Based on the current popular Yolo v3[9] target detection framework, in order to reduce network complexity and reduce redundant data, the original Darknet53 network is replaced by VGG16[10] to extract RGB image features. For a single Depth image, the paper designed with a convolutional network of 10 convolutional layers and 6 maximum pooling layers, so that the output feature dimension before each pooling layer corresponds to VGG16, for the two kinds of features can be effectively merged.

The training process of the network can be divided into two stages: forward propagation and backward propagation. In the forward propagation phase, information is propagated layer by layer from the input layer. The calculation formula of the output of the $j$ neuron in the $l$ convolution layer is:

$$a_j^l = f_c[w^l(t \sum_{i=0}^{M} a_i^{l-1} * k_j^l) + b_j^l]$$

(1)

$a_i^{l-1}$ is the output of the previous layer, $f_c(\cdot)$ is ReLU activate function, $k$ is convolution kernel, $M$ is the set of input feature maps, $w^l$ is weight of the $l$ layer of the convolutional network, $b$ is the bias of $l$ layer, * represents the process of convolution. The essence of backward propagation is to iteratively adjust the weight and bias of the network according to the change of the loss function, so as to obtain the optimal network parameters. The loss function in the Yolo algorithm is defined as the sum of the coordinate positioning error, the IoU error and the class error, and calculated for each prediction data and the calibration data. The formula is as follows:

$$loss = \lambda_{coord} \sum_{i=0}^{B} \sum_{j=1}^{N} I_{ij}^{obj}[(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2] + \lambda_{coord} \sum_{i=0}^{B} \sum_{j=1}^{N} I_{ij}^{obj}[(\sqrt{w_i} - \sqrt{w_{\hat{i}}})^2 + (\sqrt{h_i} - \sqrt{h_{\hat{i}}})^2] + \lambda_{obj} \sum_{i=0}^{B} \sum_{j=1}^{N} I_{ij}^{obj} \sum_{classes} (p_{\hat{i}}(c) - p_i(c))^2$$

(2)

$\lambda$ represents weight, $\lambda_{coord} = 5$, $\lambda_{obj} = 0.5$, $x, y, w, h, c, p$ are network predictive value, $x, y, z, h, c, p$ are calibration value, $I_{ij}^{obj}$ represents objects in the $i$ grid, $I_{ij}^{obj}$ and $I_{ij}^{noobj}$ represents objects in and out the $j$ bounding box of the $i$ grid respectively.

2.2. Dual channel network fusion method
When take joint detection of RGB-D images, the information of RGB and Depth images is mainly fused by a convolutional neural network. The current simple mode for RGB-D information fusion is early and late fusion [11], but the middle fusion before the fully connected layer will achieve better recognition. Based on the [12], this paper proposes a medium-level layer-by-layer fusion method (as shown in Figure 1). The output of each convolution layer of the Depth network is fused to the feature
layer of the corresponding RGB network with different weights to be the input of the next layer, and finally enter the fully connected layer of the RGB network to recognize the object category in the scene.

Figure 1. RGB-D medium-level layer-by-layer fusion network. $y_r$ and $y_d$ are output of RGB and Depth networks respectively, $\alpha$ and $\beta$ are fusion modulus of RGB and Depth network, $y$ is output of feature fusion.

We assume the $i$ Neurons of the $l$ convolution layer is $a_i^l$, $M_r$ and $M_d$ are collections of input feature maps respectively selected from the RGB network and the Depth network. According to the formula (1), the neuron calculation formula of the fusion layer can be obtained:

$$a_i^l = f_i\{\alpha[w_i^r \ast \sum (a_i^{l-1} \ast k_i^r) + b_i^r] + \beta[w_d^m \ast \sum (a_i^{m-1} \ast k_d^m) + b_d^m]\}$$

(3)

In the (3), $w_i^r$ and $b_i^r$ are weights and bias for the RGB network $l$ convolution layer respectively, $w_d^m$ and $b_d^m$ are weights and bias for the Depth network $m$ convolution layer respectively, $\alpha$ and $\beta$ are fusion modulus of RGB and Depth network respectively, calculation formula as follows [6]:

$$\frac{\alpha}{\beta} = \frac{R_{rgb}}{R_d}$$

(4)

$$\alpha + \beta = 1$$

(5)

Among them, $R_{rgb}$ and $R_d$ are average accuracy of detecting RGB images and Depth images separately.

3. Experiment process

3.1. Experimental environment and data preprocessing

The experiment uses open source deep learning framework Darknet based on C and CUDA for model training, feature fusion and object recognition, running on Titan X graphics card and CUDA 9.0 GPU driver. The experiment used the NYU Depth v2 dataset, which contained 26 kinds of indoor scene, more than 1000 categories, and 1449 complementary depth RGB-D images.

In order to achieve the target recognition task on the NYU Depth v2 dataset, marked the original data manually with the detection box to create training samples. For reduce the time cost, our research team marked a total of 10 categories of 300 RGB-D images, and selected 80 images as test sets. Figure
2 shows that mark training samples by the Labelimg annotation tool.

Figure 2. Mark the training samples.

3.2. Training process and visualization

The experiment used batch standardization operation (BN), which trained 64 pictures per iteration, and iterated 30200 times. In the training phase, we used stochastic gradient descent with a momentum term of 0.9. The initial learning rate of the weight is 0.001, and the attenuation modulus is set to 0.0005. The RGB and Depth network parameters are separately trained. The training process visualizes the loss, IoU and Recall indicators (as shown in Figure 3) to evaluate the performance of the model. The IoU and Recall calculation formulas are as follows:

$$\text{IoU} = \frac{\text{area}(BB_{dt} \cap BB_{gt})}{\text{area}(BB_{dt} \cup BB_{gt})} \quad (6)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (7)$$

In the (6), $BB_{gt}$ is Ground Truth Box, $BB_{dt}$ is Detection Truth Box, the TP (True Positive) in equation (7) represents the number that positive samples are also judged as positive samples; the FN (False Negative) indicates the number that positive samples are in fact judged as negative samples.

Figure 3. Visualization of the training process.
(a)(b)(c) are loss, average IoU and average Recall of RGB network respectively; (d)(e)(f) are loss, average IoU and average Recall of Depth network respectively.

In order to improve the visualization effect, the sampling rate of the loss curve in Figure 3 is 10%, and the sampling rates of IoU and Recall curves are 0.20% and 0.25% respectively. After 30200 iterations, the loss of RGB and Depth networks is finally stabilized at 0.05 and 0.10 respectively. The training indicators of the two networks are shown in Table 1. In the case of object annotation information sharing, the overall performance of the RGB network is better than Depth due to the Multidimensionality of data.

| Table 1. The training index of RGB and Depth network |
|-----------------------------------------------------|
| Network    | IoU /%  | Recall /% |
| RGB        | 85.60   | 96.48     |
| Depth      | 82.96   | 95.80     |

3.3. Evaluation and comparison of experimental results

3.3.1. Qualitative evaluation. In the experiment, 80 RGB and Depth test images were tested with two trained networks, and the average accuracy was calculated. Then, the RGB-D image joint detection was performed by the medium-level layer-by-layer fusion method proposed in this paper. Finally, the recognition results are output (as shown in Figure 4).

![Figure 4](image)

**Figure 4.** The recognition effects of medium-level layer-by-layer fusion method. The top part indicates the recognition result by RGB network, the middle part expresses the recognition result by Depth network, and the bottom part shows the medium-level layer-by-layer fusion network recognition effect.

Figure 4 shows the recognition effect of different scenes in the test set. The comparison results show that the recognition effect of RGB images is excellent in most cases, but the detection effect of objects such as defects and overlap in the field of view is not ideal. In the detection results of the Depth network, the contour features of the objects in the near field of view are fully extracted, and the objects such as the missing and overlapping in the near field of view can be more accurately identified. For example, the lamp in (a), the tv in (b), the Whiteboard in (c), the desk in (d)(e)(f), etc., which are not detected in the RGB network, can all be correctly identified in the Depth network. The bottom part of Figure 4 is the recognition result using the medium-level layer-by-layer fusion method. It can be seen that the method combines the recognition characteristics of the two networks on the object, which reduces the missed detection rate of the object and improves the recognition accuracy. However, there are also missed detections and false detections. For example, in (c), the projection cloth is incorrectly recognized as Whiteboard, the door in (d) is not detected and recognized. In addition, due to the
limitation of the depth sensor’s working distance, the Depth network only has a high recognition rate for objects in the near field of view, and objects which far away in the field are still not recognized.

3.3.2. Quantitative analysis. In order to verify the recognition effect of the method, we not only compare the results of the individual recognition of the RGB and Depth networks trained in the experiment to our way, but also compare with the methods of paper [3] and [6] are used in the recognition accuracy and detection time, the results are shown in Figure 5. It can be seen that the RGB-D fusion method proposed by us is 5.22% more accurate than the recognition result based on RGB images, and is also superior to the methods proposed in paper [3] and [6]. However, while improving the recognition accuracy, due to the complex network structure, the time cost is high, and the average time required for detection reaches 61.03 ms, which is obviously higher than the detection time of other methods.

![Figure 5. Comparison histogram of test results with different methods.](image)

4. Conclusion
In the paper, we combined with the RGB and Depth images’ features which are benefit to recognize objects, trained two kinds convolutional neural networks respectively, and proposed the dual-channel networks method of medium-level layer-by-layer fusion. It could fuse the features extracted by each convolutional layer with different coefficients, and obtain the final weight of the two-channel network to recognize RGB-D objects effectively.

After comparison and analysis of the results, we found that our method can realize the effective recognition of RGB-D indoor objects in the near field of view, and the recognition accuracy and recall rate are obviously improved compared with the methods used in other papers. However, there are still two problems. First, due to the limitation of the depth sensor’s working distance, the Depth network has a high recognition rate only for the near-field objects of the depth image, and the distant pixel cannot be acquired. The second is complexity of the medium-level layer-by-layer fusion method is high, and the amount of its calculation is large, so that the recognition object is slower. The next work will focus on the above two aspects to research.

References
[1] Runshun L. 3D Object Recognition Algorithm Based on Deep Learning. Dalian: Dalian University of Technology, 2017: 1-4.
[2] Socher R, Huval B, Bath B, et al. Convolutional-recursive deep learning for 3d object classification. Advances in Neural Information Processing Systems, 2012: 665.
[3] Wei L. Research on RGB-D Object Recognition via Feature Learning. Wuhan: Huazhong University of Science and Technology, 2016: 25-36.
[4] Hoffman J , Gupta S , Leong J , et al. Cross-modal adaptation for RGB-D detection. IEEE International Conference on Robotics & Automation. IEEE, 2016.
[5] Xu X Y, Li Y C, Wu G S, et al. Multi-modal deep feature learning for RGB-D object detection. Pattern Recognition, 2017, 72: 300.

[6] Fan L, Pengyuan L, Junning Z, et al. Joint Detection of RGB-D Images Based on Double Flow Convolutional Neural Network. Laser & Optoelectronics Progress, 55, 021503(2018).

[7] Silberman N , Hoiem D , Kohli P , et al. Indoor Segmentation and Support Inference from RGBD Images. European Conference on Computer Vision. Springer, Berlin, Heidelberg, 2012.

[8] Lu H T, Zhang Q C. Overview of application of depth convolutional neural network in computer vision. Data Acquisition and Processing, 2016, 31(1): 1-17.

[9] Redmon J , Farhadi A . YOLOv3: An Incremental Improvement. 2018.

[10] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[11] Shuqin T, Yueju X. Yun Liang, et al. RGB-D image classification methods. Laser & Optoelectronics Progress, 2016, 53(6): 060003.

[12] Hazirbas C, Ma L , Domokos C , et al. FuseNet: Incorporating Depth into Semantic Segmentation via Fusion-Based CNN Architecture. 2016.