CAPTCHA RECOGNITION BASED ON CONVOLUTION NEURAL NETWORK

Ling-Feng Gao*, Inwhee Joe, Ph.D *
*Dept. of Computer Science, Hanyang University

컨볼루션 네트워크 기반의 캡차 인식

고릉풍*, 조인휘*
*한양대학교 컴퓨터소프트웨어학과
glf1024@naver.com, iwjoe@hanyang.ac.kr

Abstract

For a long time, CAPTCHA Recognition has been a major challenge in the field of artificial intelligence. Although there are many related technologies that can solve this identification problem, further breakthroughs are still needed. Based on the existing SSD network, this paper adds a non-block module. Compared with the original SSD network, the recognition rate of SSD + Non-local is improved from 86.12% to 88.47%.In addition, it is worth noting that the recognized character verification code consists of Arabic numerals, uppercase and lowercase English letters.

1. Introduction

CAPTCHA is a fully automatic public program that distinguishes whether a user is a computer or a human, which can avoid maliciously cracking passwords, ballot rigging, forum rigging, etc., so as to effectively prevent a hacker from constantly trying to log in to a specific registered user with a specific program brute force. CAPTCHA can be generated and judged by computers, but it must be answered by humans themselves. Since the computer cannot answer CAPTCHA's question, the user who answers the question can be considered human.

In view of the fact that CAPTCHA is widely used in the Internet, the recognition of CAPTCHA has been studied at home and abroad, and there are a large number of technologies to crack CAPTCHA. For CAPTCHA images with less noise and fixed number of characters, the Deep Neural Networks (DNNs) can achieve good results by learning multiple tags from the whole CAPTCHA image to complete the classification task. In addition, the combination of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) can also be used to realize end-to-end recognition of CAPTCHA images [1].

This paper first introduces the recognition content of CAPTCHA image which is composed of numbers and letters, and analyzes the deficiency of SSD network[2] to extract global information. Furthermore, this paper introduces non-local block on the basis of SSD network[4]. The experimental results show that the improved SSD network can improve the recognition accuracy of Captcha images with numbers and letters.

2. Networks and modules

2.1 CAPTCHA data [3]

The character type CAPTCHA identified in this paper is composed of Arabic numerals, uppercase and lowercase English letters and operators in different fonts. CAPTCHA consists of a picture with a length of 200 pixels and a width of 60 pixels, as well as five uppercase and lowercase English letters and numbers. This type of CAPTCHA is composed of 26 uppercase and lowercase letters and Arabic numerals from 0 to 9, with background interference of noise and lines, as shown in Figure 2-1.

Figure 2-1 Picture of CAPTCHA [3]

2.2 Single shot detector (SSD) [2]

Single shot detector (SSD) network is a single-step target detection network proposed in 2016. It achieves the end-to-end model structure of CNNs, which not only ensures the detection speed and accuracy, but also facilitates model training and optimization. SSD adopts VGG16 [5] as the basic model, and then adds a convolution layer on the basis of VGG16 to obtain more feature graphs for detection. The default box does not correspond to the receptive field of each layer, but corresponds to a specific convolution feature map, and the feature maps of different layers correspond to default boxes with different scaling ratios and different aspect ratios. SSD network does not have the stage of region generation, but carries out the prediction of object types and the regression of frame coordinates directly on the default frame. In order to introduce the concept of the default box more intuitively in the SSD network, the idea of the default box is introduced by

introducing the concept map of the default box in the SSD network article, as shown in Figure 2-2.

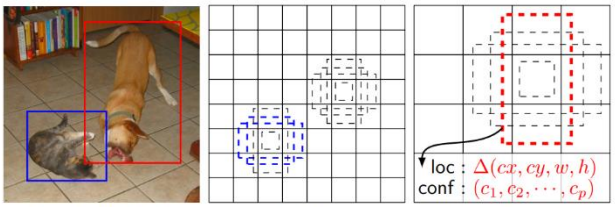

Figure 2-2 SSD network default frame structure diagram [2]

Figure 2-2 shows that the SSD network uses convolution feature graphs of different sizes to regress the location information (location) and the object category confidence value (confidence) through the default box. Figure 2 Sub-figure (a) is the input image with a marker box, the target in the blue rectangle is the cat, and the target in the red rectangle is the dog. Figure 2 Sub-figure (b) is the default box set on the convolution feature map of 8-8 size. The default box is a series of fixed-size rectangular boxes on a small grid, represented by dotted lines, and the blue default box at the lower left in the Sub-figure (b) in figure 2-2 corresponds to the target kitten in the (a) in figure 2-2. Figure 2-2 sub-figure (c) is the default box set on the convolution feature map of size 4*4, in which the default box of the red dotted line corresponds to the target puppy in figure 2-2 sub-figure (a), and the regression of location information (location) and object category confidence value (confidence) is carried out on the default box. SSD network adds six convolution layer modules with decreasing feature graph size after the traditional basic network (such as VGG network). The inputs of six feature graphs are convoluted by two different convolution kernels of 3x3, one output classification confidence value, and each default box generates 21 categories of confidence. One outputs the position information for regression, and each default box generates four coordinate values. Finally, the results on the five feature graphs are merged and sent to the loss function layer. The SSD network structure is shown in Figure 2-3.

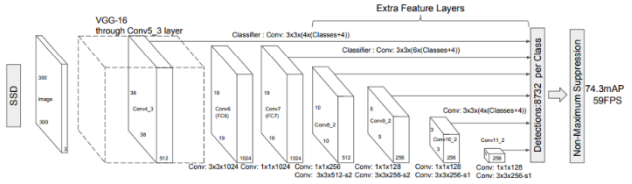

Figure 2-3 SSD network structure diagram [2]

2.3 Non-local block[4]

CNNs extracts features through convolution kernel, and the size of convolution kernel corresponds to the local receptive field of convolution layer, for example, the size of convolution kernel is3*3, which means that each position in the process of feature extraction can only interact with the features in the neighborhood of 3x3. Although the design of this structure can reduce the number of parameters and the amount of calculation of the network, it will make the network lose the ability to integrate global information. The non-local module is designed to improve the ability of the network to extract global information. The design idea of non-local is as follows:

$$\mathbf{y}_i = \frac{1}{\mathcal{C}(\mathbf{x})} \sum_{\forall j} f(\mathbf{x}_i, \mathbf{x}_j) g(\mathbf{x}_j).$$

i is one of the locations of the output feature map, which in general can be time, space, and space-time. j is the index of all possible locations, and x is the input signal, which can be images, sequences, and videos, usually feature maps. y is the same output graph as x scale, f is the pairing calculation function, calculating the correlation between the ith position and all other positions, g is the unary input function, the purpose is to carry out information transformation, C(x) is a normalized function to ensure that the whole information remains unchanged before and after the transformation. The above is a very general formula, details of which are shown below. In the local convolution operator, generally:

$$i - 1 \le j \le i + 1$$

Since both f and g are general formulas, the specific form of the neural network needs to be considered.

First of all, since g is a one-element output, it is relatively simple. 1x1 convolution can be used to represent linear embedding, and its form is:

$$g(\mathbf{x}_j) = W_g \mathbf{x}_j$$

For f, it is actually calculating the correlation between two positions, so the first very natural function is Gaussian.

(1) Gaussian

$$f(\mathbf{x}_i, \mathbf{x}_j) = e^{\mathbf{x}_i^T \mathbf{x}_j}$$

Do a dot multiplication on the two positions, and then use exponential mapping to amplify the difference.

(2) Embedded Gaussian

$$f(\mathbf{x}_i, \mathbf{x}_j) = e^{\theta(\mathbf{x}_i)^T \phi(\mathbf{x}_j)}$$

The previous gaussian form is directly calculated in the current space, while (2) is more general, calculating the Gaussian distance in the embedding space. Here:

$$\theta(\mathbf{x}_i) = W_\theta \mathbf{x}_i$$

$$\phi(\mathbf{x}_j) = W_\phi \mathbf{x}_j$$

The first two:

$$\mathcal{C}(\mathbf{x}) = \sum_{\forall j} f(\mathbf{x}_i, \mathbf{x}_j)$$

Observe carefully, if C(x) is taken into account, then it can be obtained that:

$$\frac{1}{\mathcal{C}(\mathbf{x})} f(\mathbf{x}_i, \mathbf{x}_j)$$

In fact, this is the softmax form, the complete consideration is:

$$\mathbf{y} = \text{softmax}\left(\mathbf{x}^T W_\theta^T W_\phi \mathbf{x}\right) g(\mathbf{x})$$

Design the non-local idea into a residual structure to be inserted into CNNs:

$$z_i = W_z \cdot y_i + x_i$$

According to the design idea of the non-local module expressed by the above formula, the network structure of the non-local module is shown in Figure 2-4.

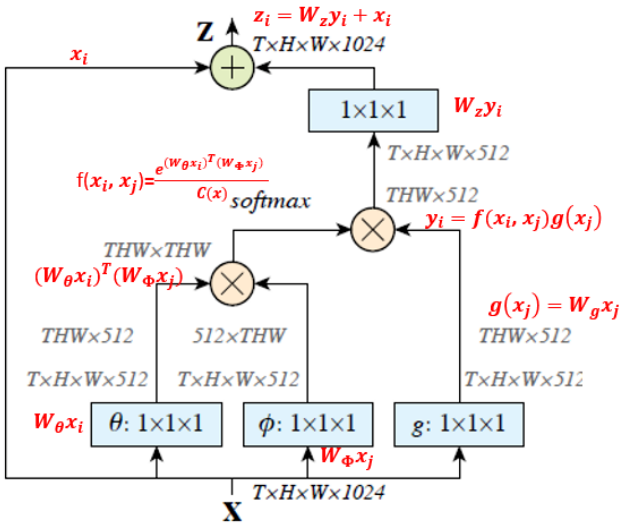

Figure 2-4 Non local block structure diagram [4]

First, the network input is X= (batch, h, w, 1024), after two embedding weight transformations W_θ and W_ϕ in Embedded Gaussian, (batch, h, w, 512) and (batch, h, w, 512) are obtained. In fact, the purpose of this step is to reduce the number of channels and reduce the amount of calculation; reshape the two outputs separately to become (batch, hw, 512); perform matrix multiplication on these two outputs (one of them needs to be transposed), calculate the similarity, and get (batch, hw, hw); perform the softmax operation on the second dimension, the last dimension, to get (batch, hw, hw), this step aims to obtain spatial attention, which is equivalent to finding the normalized correlation between each pixel in the current picture or feature map and all other position pixels; Then, perform the same operation on g, that is, channel dimension reduction, and then reshape; then, multiply it with (batch, hw, hw) by matrix to obtain (batch, h, w, 512), which means that the spatial attention mechanism is applied to the corresponding position of each feature map of all channels, the essence is that each position value output is a weighted average of all other positions, and the commonality can be further highlighted through the softmax operation. Finally, the output channel is restored through a 1x1 convolution to ensure that the input and output scales are exactly the same.

3. SSD_NON_LOCAL Structure

Figure 3-1 shows the SSD_NON_LOCAL structure diagram of the improved SSD network, of which Conv7 represents that the convolution module is the seventh convolution module of the SSD network. The module consists of only one convolution layer and contains 1024 convolution cores. Since the module is located in the first module of the continuous convolution module, the size of the feature map is the largest and the prediction is the most accurate, so the non-local module is selected after the conv7 convolution layer.

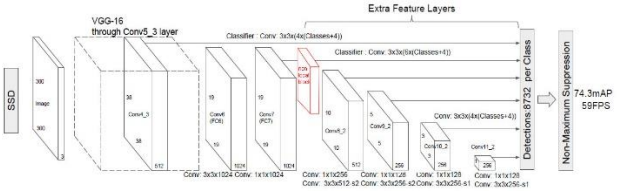

Figure 3-1 SSD_NON_LOCAL structure diagram

4. Experimental results

4.1 Experimental data set

The CAPTCHA picture data is the China University Student Service Outsourcing Innovation Competition, identified by A16CAPTCHA. 10,000 training set pictures and 5,000 test pictures [3].

4.2 Training parameters

parameter name	Setting parameters
Optimization Strategy	SGD + Momentum
Momentum	0.9
Learning rate	Multistep [80000, 100000, 120000]
Learning rate decay coefficient	0.1
Batch size	8
Number of iterations	120000

4.1 Recognition result

(a)SSD recognition result

(b) SSD_NON_LOCAL recognition result

Figure 4-1 Comparison of CAPTCHA recognition results

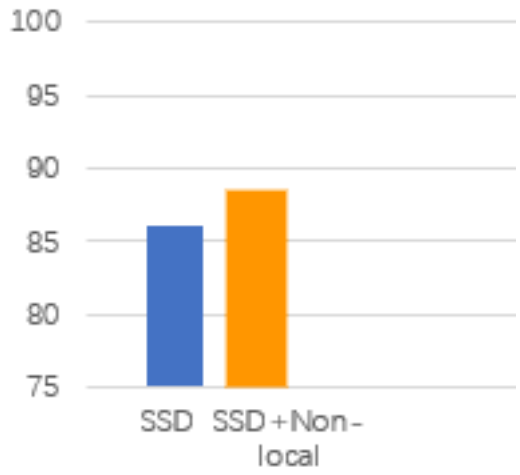

Table 4-2 Comparison of CAPTCHA recognition accuracy

It can be seen from Table 4-2 that compared with the original SSD network, the recognition accuracy rate of SSD_NON_LOCAL has increased from 86.12% to 88.47%.

5. Conclusion

This paper introduces the character CAPTCHA pictures composed of numbers and letters, points out that the SSD network will reduce the ability of extracting global information in the process of continuous stacking convolution layer, and improves the SSD network by introducing Non local block. The experimental results show that the recognition rate of SSD_NON_LOCAL on CAPTCHA pictures composed of numbers and letters is higher than that of SSD network.

References

[1] XU Xing;SONG Xiaopeng;DU Chunhui(School of Information and Communication Engineering, North University of China, Taiyuan 030051, China;Taiyuan Institute of China Coal Technology and Engineering Group, Taiyuan 030006, China)

[2] Liu W, Anguelov D, Erhan D, et al. SSD: single shot multibox detector [C]. 2016 14th European Conference on Computer Vision (ECCV). Amsterdam, Netherlands: Springer, 2016: 21-37.

[3] China University Student Service Outsourcing Innovation Competition, identified by A16CAPTCHA.https://github.com/SaulZhang/Tensorflow-CAPTCHA-Recognition

[4] Wang X, Girshick R, Gupta A, et al. Non-local neural networks[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 7794-7803.

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