FAST DETECTION OF TARGET BASED ON SSD DATASET TRAINING ALGORITHM

C.H. CHEN†, Z.Q. HOU†‡, W.S. YU†, X.F. LIAO†, and J.Y. WANG†

†Air Force Engineering University, Xi’ an, China
1327067049@qq.com
‡Xi’an University of Posts and Telecommunications, Xi’an, China
13279492656@163.com

Abstract—Edge detection is one of the most basic contents of image processing and analysis. The edge of the image contains the position and contour of the image, which is one of the basic features of the image. The accuracy of the traditional algorithm is not high because of the strong jitter of the target and the larger interference. The key to target edge detection lies in the extraction of effective features, and this can be properly realized with a feature extraction based on the depth-learning algorithm. In this paper, a method of sample synthesis is proposed, which is used for network training and can be used to detect small-scale moving targets in a limited sample space. A large number of experimental tests show that the algorithm can detect small moving target edges, showing high accuracy, real-time performance and strong robustness.

Keywords—deep learning; SSD model; data set training; target; edge detection.

I. INTRODUCTION

Deep learning is a learning method based on multilayer neural network, includes a number of important network: convolutional neural network, automatic encoder, sparse encoding, restricted Boltzmann machine, deep belief network and multi feedback recurrent neural network. For different problems, we need to choose different network models to achieve better results (Taimori et al., 2015). Therefore, the depth model is the means, and the characteristic named ‘learning’ is the purpose. Shallow learning is different from the traditional one, and deep learning is different in turn. Specifically, for emphasis on the structure of the model depth, which usually has 5 and 6 hidden nodes or even hundreds of layers. It clearly highlights the characteristic of learning. That is to say, by feature transforming, the sample features in the original space representation transform a new feature space. So, the classification or prediction become eased (Wu, 2015). Compared with the artificial rules, the use of large data to learn features can embody the rich intrinsic information of the data.

In order to detect and identify targets in complex scenes, first of all, select some simple and obvious features, such as colour and direction, to realize the feature extraction and matching location of interested regions or objects, that is, the separation of targets and scenes, and lay the foundation for subsequent scene analysis (Wu et al., 2018a). However, when the target is fast, it is easy to appear empty phenomenon and it is sensitive to noise (Qishou et al., 2018). The optical flow method uses the velocity vector of each pixel and the optical flow constraint equation to detect the target pixels (Abdelqader et al., 2015; Dollár et al., 2014). The calculation is complex and cannot meet the real-time requirement. In this paper, a deep learning based video indoor target edge detection algorithm is trained based on a model based on big data and finally applied for real scene detection.

II. INTRODUCTION OF DEEP LEARNING SSD NETWORK MODEL

A. SSD Model Network Structure

There are many methods for target detection based on deep learning, such as R CNN, Fast RCNN, Faster RCNN, YOLO and Single Shot Multi-box Detector. For example, the classic RCNN first uses Selective Search to generate about 2000 candidate boxes, then the candidate box is sent to RCNN for feature extraction, and finally the extracted features are classified. Method of pre-frame is based on the biggest problem of the processing speed of a piece of slow. It is because the need to go through the deformation after processing by CNN prior to the calculation of a network characterized by about 2000 candidate boxes. Selective Search obtain a picture frame, which covers the repeated calculation of repeat region in a picture (Wu et al., 2018b). The SSD method used in this paper directly enters a whole image into the neural network, so that the neural network can judge where and where it may be, and at the same time, it also reduces the number of possible image blocks, which improves the speed of the algorithm.

Each target box is fixed to the location of its corresponding feature map cell. In every feature map cell, we have to predict offsets between box and default box, and score in every box contains every class probability. Therefore, for every box in K boxes at a location, we need to compute C classes, score of each class, and 4 offset values of this box relative to its default box. Then, in the feature map in each feature map cell, you need to have (c+4) * k filters for m x n size feature map is generated (c+4) * k * m * n output.
III. SSD DATA SET TRAINING AND CORRECTION

A. SSD Network Training

In training, the SSD and those with region proposals and pooling method difference is SSD ground truth, the training images need to give to those fixed output boxes. As mentioned earlier, the SSD output is defined in advance, a series of fixed-size bounding boxes. The purpose of training is to match the pre-prepared label files with the location of the model prediction (Lopez-Molina et al., 2013). From Fig. 1(a) dog ground truth white bounding boxes, was on the label, to a white ground truth box given in Fig. 1(c) a series of fixed output boxes can be seen. Thus, rough dotted line in Fig. 1(c) box to make a cat with black ground truth box to Fig. (b) in a series of fixed output boxes, i.e. rough dotted line in Fig. 1(b) box.

![Figure 1. Training picture matching.](image)

(a) Image with GT boxes (b) 8x8 feature map (c) 4x4 feature map

B. Training process

The convolution neural network is actually a mapping relationship between input and output (Lopez-Molina et al., 2013). Because of the complexity of the mapping, the precise mathematical expression relationship that is defined by human is not able to meet its requirements. Therefore, the network will be able to get the mapping relationship that meets the needs of the people only by training the large number of known data to set up the set of network models. SSD integrates detection and classification to achieve end to end training (Hu et al., 2015). In order to verify the feasibility of the model, MSCOCO data set is adopted, and the training process includes 5 steps.

1. The basic features of the image are obtained by the forward propagation of the input image.
2. The candidate region extraction multi-level feature map and the location of each of these features in the map on the selection of different size and different aspect ratio;
3. The coordinate position offset and class score of each candidate region are calculated.
4. The final area is calculated according to the offset of candidate area and coordinate position, then the loss function of candidate area is calculated based on category score, and the final loss function is accumulated.
5. The weight value of each layer is corrected by the last loss function through the reverse propagation process.

In order to further verify the SSD model of this article, we carry out training tests on SSD300 and SSD500 on the MS COCO data set. Because the detection goals of the COCO dataset are smaller, we use a smaller default boxes on all layers. Here, it is also compared with the ION detection method. The overall results are as following Table 1:

<table>
<thead>
<tr>
<th>Method</th>
<th>Data</th>
<th>Average precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fast R-CNN</td>
<td>Train</td>
<td>42.3</td>
</tr>
<tr>
<td>YOLO</td>
<td>Train</td>
<td>43</td>
</tr>
<tr>
<td>SSD300</td>
<td>Trainval135k</td>
<td>39</td>
</tr>
<tr>
<td>SSD500</td>
<td>Trainval135k</td>
<td>44.8</td>
</tr>
</tbody>
</table>

IV. FAST DETECTION OF TARGET EDGE BASED ON SSD

Edge is one of the most important characteristics of objects. Only the edge of objects can describe the characteristics of objects. For pattern recognition and computer vision, edge is the most important feature of image, and only one edge can recognize an object. Edge detection is an important step before feature extraction, edge detection to extract the contour from the original image, retaining most information of the original image, can remove many false information for subsequent identification work, reduce the amount of information...
and improve the recognition speed and accuracy. Because of the importance of edge detection, there are many edge detection algorithms (Tang et al., 2016).

A. Traditional Method of Target Edge Detection
The essence of edge detection is to use some algorithm to extract the boundary between objects and objects, objects and background in the image. The edge of the image can be defined as a regional boundary with a sharp change in the grey level of the image. The change of image grey level can be reflected by the gradient of the image grey distribution. Therefore, the edge detection operator can be obtained by the local image differential technique. The process is shown in Fig. 2. First, smoothing is used to filter the noise in the image, then first order differential or two order differential operation is applied to get the zero crossing point of the gradient maximum or the two order derivative. Finally, an appropriate threshold is selected to extract the boundary.

Figure 2. The process of image edge detection.

The local edge of image is defined as the transition between two regions with different intensities. The gradient function of image, namely the rate of the change of image grey scale, will have the maximum value on these transition boundaries. Early edge detection is to estimate the gradient direction of the image grey level gradient operator by the detector or the first derivative, these changes enhance regions in the image, and then the threshold operation on the gradient, if the gradient value is greater than a given threshold, there is edge (Akula et al., 2013). The corresponding greyscale change curve is shown in Fig. 3.

First order differential is the most basic method of image edge and line detection. The gradient of the image function \( f(x, y) \) in the point \((x, y)\) is a vector with direction and size.

\[
\nabla f(x, y) = \left[ G_x \ G_y \right]^T = \left[ \frac{\partial f}{\partial x} \  \frac{\partial f}{\partial y} \right]^T
\]

(1)

The amplitude of \( \nabla f(x, y) \) is:

\[
g(x, y) = |\nabla f(x, y)| = \sqrt{\left( \frac{\partial f}{\partial x} \right)^2 + \left( \frac{\partial f}{\partial y} \right)^2}
\]

(2)

Direction angle:

\[
\theta(x, y) = \arctan \left( \frac{\frac{\partial f}{\partial x}}{\frac{\partial f}{\partial y}} \right)
\]

(3)

Based on the above theories, people put forward many algorithms (Butanda et al., 2017). The commonly used methods are differential edge detection operator; Roberts edge detection operator, Sobel edge detection operator, Prewitt edge detection operator, Laplace edge detection operator and so on. The detection effect of Canny operator is better and the edge location is accurate, but the programming of the Canny algorithm is complex and the operation is slow (Slawiński et al., 2006).
B. SSD Data Set Training and Detection Method

Because the network structure of the convolution neural network is complex and the computation is huge, the algorithm is placed on the robot or intelligent terminal, which requires expensive hardware (Tucker et al., 2014). In order to reduce the hardware requirements of the robot terminal, you can use the following steps: pre-processing of image data; using TCP Socket multi-thread communication based on the data into the cloud, obtained the result on the target detection algorithm using SSD cloud network model, return to the intelligent terminal realize online detection (Burgos et al., 2017).

1) Preprocessing of image data.
In order to reduce the time spent in uploading pictures, we need to pre-process the pictures before uploading them to the server, that is, image dimensionality reduction operation, divided into trimming and zooming, as shown in Fig. 4.

(2) On-line detection
Socket is a universal API for Internet network programming and has a corresponding implementation version in windows and Linux platforms. The Client/Server model is the most basic Socket communication model, the research on the server at two Socket: R_Socket for the robot terminal access to connect through the public network; W_Socket for computer server access to connect through the LAN, in order to speed up the processing speed of each Socket, using multithreading non-blocking mode processing (see Fig. 5).

![Figure 4. Picture preprocessing.](image)

![Figure 5. Dynamic changes of the network training error](image)

All Client ends are connected in real time and immediately connected to the server when the robot or processing computer begins to run. After preprocessing, the robot terminal uploads the picture to the server through the R_Socket connection, adds the picture to the task list of the server, and checks the processing state of the task interval (see Fig. 6).

For server side, when we detect the new tasks in the task list, we retrieve a processing computer in idle state in the save processing computer’s status list, and send the task to the processing computer (Wang et al., 2017). After the processing computer receives the task, the image file is read through the LAN share and processed. After the processing is completed, the processing results are sent to the server by W_Socket.

After receiving the processing results, the server puts the result to the corresponding position in the task list and changes the processing state of the task. When the thread that detects the task is detected that the task has been processed, the result of the task is taken out and sent to the intelligent robot terminal by means of R_Socket. The online detection flow chart is shown in Fig. 7.

(3) Analysis of experimental results
In order to improve the accuracy of the algorithm, three experiments were done in this paper. For the first time, 2000 samples were used, and the final accuracy rate was 34.0523%, which basically realized the detection of vehicle edge. In the second experiment, the sample size was increased to 20000, the accuracy rate has been greatly improved, reaching 64.3873%, and the side vehicle can be detected. In order to further improve the accuracy, this paper will further increase the training
samples to 200000, in the use of 200000 training samples, the accuracy rate from 80000 iterations of the model test can be found in some extent with the increase of training samples, the accuracy will be improved rapidly. In order to improve the detection of small target rate, we use the sample synthesis method; the accuracy rate will definitely improve the accuracy, here refers to the average rate of accuracy, precision and recall rate respectively, all test results and accuracy as shown in Table 2 and Fig. 8.

Figure 6. Training effects of various samples obtained by SSD optimizing algorithm SSD.

Figure 7. Online detection flow chart.
V. CONCLUSIONS

From the final experimental results, it can be seen that the algorithm proposed in this paper can accurately detect the edge of urban road vehicles, no matter the side, rear or even half side occlusion can be accurately detected, and basically meet the detection of road cars. Through Table 1, it can be found that the increase in the number of training samples can significantly improve the accuracy. In the experiment, we also found the following factors that would affect the final accuracy rate: the accuracy of the samples, the normalization of the sample labeling, the size of parameter settings, such as learning rate, and the number of samples.

REFERENCES


Table 2. Experimental result.

<table>
<thead>
<tr>
<th>Sample size</th>
<th>2000</th>
<th>20000</th>
<th>200000</th>
<th>Sample synthesis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training: test sample</td>
<td>1:1</td>
<td>1:1</td>
<td>1:1</td>
<td>1:1</td>
</tr>
<tr>
<td>Accuracy rate</td>
<td>34.0523%</td>
<td>64.3873%</td>
<td>95.7974%</td>
<td>96.9253%</td>
</tr>
</tbody>
</table>

Figure 8. Experimental result.