Single Shot MultiBox Detector for Vehicles and Pedestrians Detection and Classification

Qiong WU, Sheng-bin LIAO

ABSTRACT

In recent years, with the rapid development of national economy and the acceleration of urbanization, the road traffic environment has become more and more complex, and the safety and security of people's daily travel are paid more and more attention. It’s worthy to detect vehicles and pedestrians by using deep learning algorithm. In this paper, a method using the Single Shot MultiBox Detector to detect vehicles and pedestrians is introduced and compared with the others. Furthermore, the development of vehicle and pedestrian detection in the field of deep learning is discussed.

INTRODUCTION

Object detection is one of the main topics in the field of computer vision. The core task of object detection is to locate the objects in the images, distinguish the object category and give the bounding box of each object. However, due to the distortion caused by the factors such as perspective, occlusion and posture, object detection has become a certain challenge. The traditional method for object detection can be mainly divided into six steps: denoising, window sliding, feature extraction, feature selection, feature classification, calculating the bounding box, of witch, feature extraction and feature classification are the key points. Because the traffic environment is so complex and changeable that even if the best nonlinear classifiers are used for feature classification, the accuracy of object detection and anti-interference ability will fail to meet the actual requirement.

In order to extract better features from the object, in 2006, Hinton put forward the conception of deep learning, which is implemented by using the deep neural networks (DNN) to automatically learn high-level features from a large amount of sample data. With the continuous development of deep learning, several deep learning networks with different functions have been proposed, such as Multi-layer Perceptron (MLP), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN) and Generative Adversarial Network (GAN). Researchers found that by using convolutional neural network to detect the object, the accuracy can be greatly improved and anti-interference ability will be much better. This is because the convolutional neural

Qiong WU1 Sheng-bin LIAO2
First author, Qiong WU, National Engineering Research Center For E-Learning of Central China Normal University, Lane 152, Luoyu Road, Hongshan District, Wuhan, China.
Second author, Sheng-bin LIAO, National Engineering Research Center For E-Learning of Central China Normal University, Lane 152, Luoyu Road, Hongshan District, Wuhan, China.
Corresponding author: wuqiong@mails.ccnu.edu.cn
network extracts high-level features and the expressive ability of the features is improved. What’s more, convolutional neural networks incorporate feature extraction, feature selection and feature classification together in the one same model, which will greatly increase the divisibility of the features and with end-to-end training, it optimizes the function of the network as a whole.

In recent years, the algorithm models constructed with convolutional neural network have get good competition results in Pascal Visual Object Challenge. Both YOLO\textsuperscript{7} algorithm and SSD\textsuperscript{8} algorithm use convolutional neural network as the basic network for feature extraction, so they are much better than traditional object detection methods in accuracy and anti-interference ability. The difference between YOLO and SSD is that YOLO algorithm uses multiple convolutional layers to extract features and then predict the output probability with a fully connected layer. But SSD algorithm uses different convolutional layers to extract features with different sizes, and then directly calculates the location loss, confidence loss and gives the bounding box. So, the SSD algorithm does not use the full connection layer which helps to make the detection speed faster, and the SSD algorithm uses the features obtained from multiple different convolution layers, so that fewer features are lost and the detection accuracy is improved.

**MODEL OF SINGLE SHOT MULTIBOX DETECTOR**

Object detection has always been one of the most important fields of computer vision. In recent years, convolutional neural network has been applied in many object detection algorithms and have helped to greatly improve both detection accuracy and detection speed.\textsuperscript{9} Single Shot MultiBox Detector is relatively faster and more robust among various object detection methods, because it uses multiple different convolution layers for object detection.\textsuperscript{8}

![Figure 1. The model structure of SSD algorithm.](image)

Single Shot MultiBox Detector is based on a feed-forward convolutional network which produces a series of fixed-size bounding boxes and scores for the presence of object class instances in those boxes, followed by a non-maximum suppression step to produce the final detections.\textsuperscript{8} The early network layers, which is also called base network, are based on a standard architecture constructed with VGG-16 and used for high quality image classification.\textsuperscript{10} After the base network, extra auxiliary network structures and extra prediction are added, such as: (1) additional convolution layers whose size is decreased layer by layer to make predictions at multiple scales, which helps to improve the prediction accuracy; (2) additional feature layers, for each added feature layer can produce a fixed set of detection predictions using a set of convolutional filters. (3) We associate a set of bounding boxes with each feature map cell, predict the offsets relative to
the ground truth box shapes in the cell, as well as the per-class scores that indicate the presence of a class instance in each of those boxes. The model of SSD algorithm is shown in Figure 1.

RELATED WORK

SSD algorithm is a kind of deep learning algorithm, it only needs dataset images with corresponding labels when training. In order to make the weights more fitting, a large number of dataset has to be provided. Dataset collection and tagging labels are tedious and time-consuming processes. When everything is ready, it is crucial to choose an appropriate deep learning network to conduct the experiment.

Single Shot MultiBox Detector framework

For the SSD algorithm, before start the training phase, the only requirement is to input the image samples with ground truth boxes, a good image sample is shown like figure 2(a), we can evaluate a series of default boxes with different aspect ratios and different sizes at each location in several feature maps with different scales, which is shown like figure 2(b) and figure 2(c). For each default box, we predict both the shape offsets and the confidences for all object categories. At training time, the first thing is to use these default boxes to match the ground truth boxes, as shown in figure 2(b), the car matches the default box is a red box with dot line, and for figure 2(c), pedestrian matches the default box is the blue box with dot line. The model loss is the weighted sum of the location loss (such as Smooth L1) and the confidence loss (such as Softmax).

Data collecting and processing

All of the datasets used in this paper are from the road pictures collected by its author, and one of the original pictures is shown like figure 3(a). Each interested object in the picture should have the corresponding ground truth box as shown in figure 3(b). It is necessary to tag all the collected pictures with proper corresponding ground truth boxes. To make sure the machine knows where is the box, we need write the object category and the object location into the txt document as shown in figure 4, the first number on each line in the txt document file represents the object type, followed by several figures on behalf of the object’s ground truth boxes information. In total, 16,000 images are collected and tagged, 80% of the total is randomly selected for training set, 10% for the verification set, the remaining 10% for the test set.
Selection of deep learning framework

With the deep learning technology getting widely accepted in academia and industry, more and more people begin to participate in the related research and practice of deep learning. However, considering the existence of a certain technical threshold, it is not an easy task to start a deep learning study quickly. One of the most important reasons is that many problems in deep learning depend on practice, so an excellent framework designed for deep learning that is characterized by good performance, easy expansion and quick start-up is urgently needed by beginners.

In 2013, a deep learning framework called Caffe was posted on GitHub by Yangqing Jia of the University of California, Berkeley. From the beginning of the release, the Caffe framework has been receiving widespread attention. The Caffe framework takes 'layer' as unit to build algorithm network, by using some sophisticated designs, the efficiency of execution is greatly optimized without losing the flexibility. No matter in terms of structure, performance or code quality, Caffe is an excellent open source framework, and more importantly, it reveals every detail of deep learning for researchers to study and practice.\textsuperscript{12}

EXPERIMENTAL PROCESS AND RESULTS

The experimental process includes training, validating and testing. The training phase is to use the training set to train the algorithm network and get the weight model, then, the back propagation is used to update the weights. During validating time, we use the weights to detect objects with validating set and get the mean average precision. When testing, we use the final weights to detect object with testing set and get the test results. Of course, there are more than one algorithm which is suitable for vehicle and pedestrian detection. Such as YOLO algorithm proposed by Joseph Redmon can also be used for the detection.\textsuperscript{7} The experimental results of the two algorithms will be compared later.

Training, validating and testing

The training process is shown in figure 5(a). The data layer is used for data augmentation and initializing the training set. The convolution and pool module performs
the feature sampling and dimensionality reduction on the training set. The feature maps segmentation and calculation module is used for calculating the location loss, confidence loss and the bounding box of the obtained feature maps. And the loss calculation module calculates the error between the training result and the actual result. If the times of training does not reach the given maximum one, the error will be back-propagated to update the weights, and when the times of training reached 120,000, training will be ended. During training time, after every ten thousand times the validating set is used to validate the weights acquired currently. In the validating phase, it mainly calculates the mean average precision, validating process is shown like figure 5(b) below.

![Diagram of training and validation process](image)

Figure 5. Training and validating.

When the training and validating are finished, we use the final weights to detect the objects with testing set. Figure 6 shows the detecting result of the interested object with the testing pictures.

![Testing results](image)

(a) Cars and pedestrian (b) Truck and pedestrians

Figure 6. Testing results.
Comparison of SSD and YOLO

YOLO algorithm is also widely used in the object detection, for YOLO algorithm, the input image will be divided into $n \times n$ grid, for each grid, two default boxes is predicted, so that there are $n \times n \times 2$ windows for just one image, and all the windows will be calculated the confidence score and category probability to achieve the purpose of the object detection. Compared with SSD algorithm, YOLO algorithm does not use multi-scale prediction, and only predicts two default boxes, so the detection accuracy is lower; and YOLO algorithm uses the full connection layers to predict the probability of category, compared to the SSD algorithm in the training process, the training time is much more longer. The structural comparisons between SSD and YOLO are shown in Table 1.

**TABLE 1. STRUCTURAL COMPARISONS BETWEEN SSD AND YOLO.**

<table>
<thead>
<tr>
<th>Method</th>
<th>Convolution layers</th>
<th>Default boxes</th>
<th>Multi-scale prediction</th>
<th>Full connection</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSD algorithm</td>
<td>24</td>
<td>2</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>YOLO algorithm</td>
<td>26</td>
<td>Multiple</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

In order to compare the accuracy of SSD algorithm and YOLO algorithm, the mean average precision is calculated for the two algorithms after every ten thousand times of training. The result is shown as figure 7 after trained for 120,000 times. From figure 7, as the number of training increase, mean average precision is continually increasing, when the number of training reaches 90,000 times, the accuracy rate tends to be saturated, when the number of training reaches 120,000 times, the mean average precision of SSD algorithm is about 65%, for YOLO algorithm, the mean average precision is about 62%.

![Figure 7. Line chart of mean average precision.](image)

At last, we use the weighted-models coming from the SSD and YOLO algorithm respectively to detect the objects with the same testing set and get the detecting accuracy of the two algorithm, as is shown in Table 2. When using the same video as testing set, we get the detected speed (frames per second) of the two algorithm, as is also shown in Table 2.
TABLE 2. COMPARISON RESULT OF ACCURACY AND FPS BETWEEN YOLO AND SSD.

<table>
<thead>
<tr>
<th>Method</th>
<th>Test accuracy</th>
<th>FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSD algorithm</td>
<td>0.6503</td>
<td>53</td>
</tr>
<tr>
<td>YOLO algorithm</td>
<td>0.6215</td>
<td>42</td>
</tr>
</tbody>
</table>

ACKNOWLEDGEMENT

This work was supported by the National Key Technology Research Program of the Ministry of Science and Technology of China[grant number:2015BAK33B02].

CONCLUSION

In this paper, we propose a method using deep-learning algorithm (SSD) to detect vehicles and pedestrians, what’s more, experiments were conducted using the 16000 images as data sets. To make comparison, similar experiments were conducted using Yolo algorithm with the same data sets. According to the analysis and the results of the experiments, we find that there is not much difference in the number of convolution layers between SSD and Yolo algorithms, but the prediction of more default boxes and the calculation by combining with multi-scales feature maps can improve the accuracy. Furthermore, fully-connected layer contains a large amount of data, and if the output probability is calculated by using the fully-connected layer such as YOLO, then the detection speed will not be as fast as the SSD algorithm.

In the future, to realize more accurate object detection, dynamic object tracking technology can be added to improve the overall accuracy, that is to say, the same object in multiple-continuous images can be tracked by detecting its trajectory when the object is further and smaller or is blocked.

REFERENCES