2015 Sixth International Conference on Intelligent Systems Design and Engineering Applications

Vehicle Detection in Open Parks Using a Convolutional Neural Network

Haihui Xie, QingXiang Wu*, Binshu Chen, Yanfeng Chen, Sanliang Hong

Key Laboratory of OptoElectronic Science and Technology for Medicine of Ministry of Education, College of Photonic and Electronic Engineering,
Fujian Normal University, Fujian, Fuzhou, China
qxwua@fjnu.edu.cn

Abstract—**This paper proposed a new vehicle detection algorithm based on a CNN (convolutional neural network), which dedicates to detect and localize vehicles in an open park. After an off-line training the network can fast respond to an input image so that it is suitable for real-time applications and has the potential to use in vehicle park management systems. Firstly, the trained CNN with a defined sliding window is used to search and identify vehicles in open parks. Secondly, a distribution matrix is defined to reflect the density of vehicle distribution, and it is used to remove redundant windows of vehicles to locate a position of vehicle accurately. Compared to other approaches for vehicle detection, the CNN-based approach does not require any engineered features. The proposed algorithm has combined a CNN with the distribution matrix so that the accuracy of the position location has been improved.**

Keywords-Vehicle detection; vehicle localization; convolutional neural network; distribution matrix; sliding window

I. Introduction

In the machine vision domain, many technologies were used in the vehicle detection. Firstly, some researchers introduced the Multi-Classifier [1] based on multi-features to detect vehicles in a vehicle park. The method above has high complexities in calculation, and the system becomes tedious and it is difficult to deal with these algorithms in the real-time. Secondly, pixel detection [2] with a suitable threshold to differentiate between vehicles and empty slots had also been used to detect vehicles. However, the method does not work well when the camera is far away from the vehicle park or the vehicles have few pixels in the image or the environment changes severely, or the color of vehicles is close to the background. Thirdly, there was a method which uses trained neural networks to determine occupancy states based on visual features extracted from parking spots [3]. However, some features are difficult to be extracted from the vehicle park accurately. In addition, some features are too complex to obtain it.

In this paper, the convolutional neural networks (CNNs) [4, 17] are applied to detect vehicles in open parks. Recently, CNNs have been demonstrated as an effective model, especially in analyzing the content of images and videos [7]. The keys of these applications were techniques for scaling up the networks to tens of millions of parameters and massive labeled data sets that should be used in the learning process. For actions recognition, CNNs could be regarded as a large-scale video classifier to obtain complex evolution of human actions [7]. On the other hand, for object segmentation, recognition, detection and so on, CNNs have also been demonstrated as an effective model [8-14].

This paper proposed a new vehicle detection algorithm based on a CNN, which dedicates to detect and localize vehicles in an open park. The algorithm is suitable for real-time applications and has the potential to benefit vehicle park management systems. Our contributions can be summarized as follows: firstly, we apply CNNs and sliding window techniques to identify vehicles in open parks. Secondly, we have proposed distribution matrix which reflects the density of vehicle distribution to remove redundant windows of vehicles to locate a position of each vehicle accurately. Compared to other neural networks in application of vehicle detection [1, 3, 5], the CNN does not require any engineered features. Compared to other algorithms based on CNNs [16-17], in this paper distribution matrix has been combined with the CNN to improve the accuracy of the vehicle position.

II. Convolutional Neural Network

In the paper, the CNN [4] is trained to identify two different types of categories. The CNN is one of the deep learning algorithms [6], and it is made up of multi-layer neural networks and it can be used to extract features automatically. Each layer is composed of a group of two-dimensional neuron arrays, and each neuron array is composed of a group of independent neurons. Each neuron array stands up for a kind of features. More layers there are in the net, more abstract the features are. With the increase of layers, the sizes of feature maps are smaller and smaller and the dimensions become more and more. The work flow is shown in Figure 1.

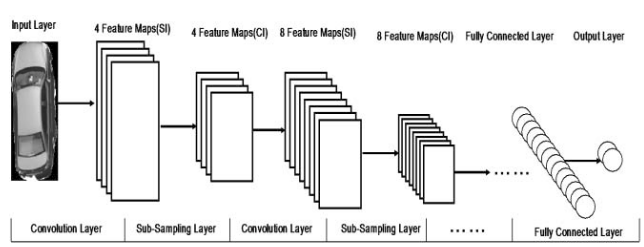


Figure 1. Flow diagram of CNN

A. The Convolution Layer

In the convolution layers, the fore feature maps are convoluted with the convolution kernels which can be learned automatically, and the result passes through the activation functions, and the network can get the output feature maps.

B. The Sub-Sampling Layer

For the sub-sampling layers, if there are N input maps, there are N output maps. And the size of the output map is less than the size of the input map, the dimensions of the output map is the same as the dimensions of the input map.

C. The Fully Connected Layer

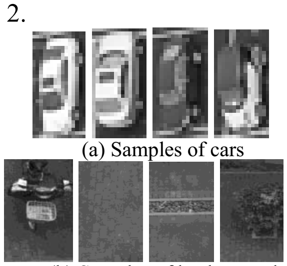
The fully connected layer is a last layer of the CNN, and it is the same as BP neural network [6].

III. THE METHOD OF THE TRAINING

The program of the training is run in the environment of the Matlab. The steps of the program are as follows.

A. Collection of the Training Sample

In the experiment, there are two distinguished results, the class of the cars and the class of the background of the parking space. In Table I, two kinds of samples are listed. The training samples should include all possible samples which may appear in the car park. In the training process, 1230 samples of the cars are used, 5640 samples of the background are collected. Some training maps are as follows in figure 2.



(a) Samples of cars

(b) Samples of background

Figure 2. Some of positive samples and negative samples

B. Normalize the image

The input interface of the CNN requests that the image should be normalized into an uniform block. In this experiment, the size of the block is set to 80×32. Ratio of width to height of the block is similar to a car, so it could be used to avoid distortion of samples.

C. Generate the training data and the label

In the experiment, (1, 0) expresses a car, and (0, 1) expresses the background of an empty parking space.

D. Set the network parameters

In the experiment, the learning rate is 0.01 and the number of iterations is 10000. From the result of detection, it can be proved that these parameters are suitable in this training.

E. Train the network

In the process of the training, error curve is shown in Figure 3. From Figure 3, it can be found that the training error tend to zero with the increasing of the number of iterations.

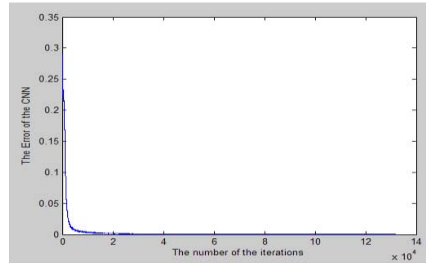


Figure 3. Training error curve of CNN

IV. THE METHODS LOCATING POSITIONS OF VEHICLES

After training the CNN, it is necessary to search the whole image of a park to identify whether there exists a car with a sliding window. The results of searching are as follows:

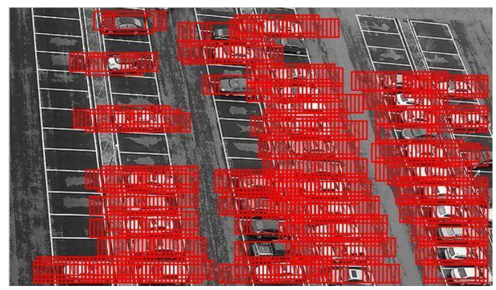


Figure 4. Results of searching

From Figure 4, it can be seen that many windows are pointed to the same vehicle. To locate the accurate position of a car, we propose a new algorithm. After searching the whole image of a parking, a label matrix $L(x,y)$ is generated, which reflects the situation whether there exists a car at (x,y).

$$L(x,y)=\{_{0}^{1 \quad if\ f_1>f_2}, \tag{1}$$

where f_1 · f_2 are outputs of a CNN, and can be regarded as probabilities of vehicle and background. They can be used to judge whether there exists a car at (x,y).

To describe a distribution of vehicles in an open park, we define a distribution matrix $D(x,y)$. $W_{(x,y)}$ is a sliding window which is used to search $L(x,y)$ at (x,y). Figure 5 shows the situation of segmentation for $W_{(x,y)}$. The heights and widths of S,S_0,S_1,S_2,S_3 are halves of the window's and there are symmetric distributions in the window $W_{(x,y)}$.

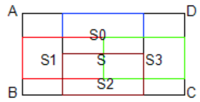


Figure 5. Searched window at a label matrix

The formulas are as follows:

$$D(x,y)=\sum_{i=1}^{w}\sum_{j=1}^{h}S(i,j) \tag{2}$$

$$D(x,y-1)=\sum_{i=1}^{w}\sum_{j=1}^{h}S_0(i,j) \tag{3}$$

$$D(x-1,y)=\sum_{i=1}^{w}\sum_{j=1}^{h}S_1(i,j) \tag{4}$$

$$D(x+1,y)=\sum_{i=1}^{w}\sum_{j=1}^{h}S_2(i,j) \tag{5}$$

$$D(x,y+1)=\sum_{i=1}^{w}\sum_{j=1}^{h}S_3(i,j) \tag{6}$$

where w, h are width and height of the corresponding region. According to the formula (2-6), results of $D(x,y)$ are showed in Figure 6.

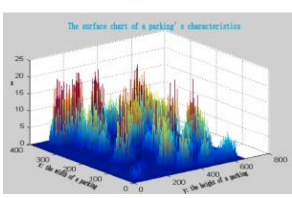


Figure 6. Characteristics surface of a parking

From Figure 6, it can be inferred that a value of the distribution matrix exists a maximum value of the local area at the position of a car. According to the characteristic, a new label matrix $L(x,y)$ is generated.

$$L(x,y)=\{_{0}^{1\ \ if\ D(x,y)>D(x-1,y),D(x,y)>D(x+1,y),D(x,y)>D(x,y-1),D(x,y)>D(x,y+1)} ,(7)$$

In order to locate the car more accurately, some measures should be implemented. Code structure is as follows:

for i=1:size(L).width

for j=1:size(L).height

if $L(i,j)==1$

calculate the gravity (PI,PJ) of $L(i-w/4:i+w/4,j-h/4:j+h/4)$;

$L^{'}(PI,PJ)=1$;

end

end

end,

where w and h are width and height of a sliding window $W_{(x,y)}$. According to $L^{'}$, the position of a car could be located. The result is showed in Figure7.

V. TEST PROCESS IN THE EXPERIMENT

The program runs in the environment of the Matlab. The step of the program is as follows.

1. Train the CNN net and load the CNN structure trained.
2. Search the whole image of a park using a sliding window with a size of a car and identify where there is a car based on the formula (1).
3. Remove extra windows around vehicles to locate positions of vehicles accurately with distribution matrix.

VI. THE EXPERIMENT RESULTS AND ANALYSIS

In this experiment, although the process of training the CNN costs much time, the trained CNN classifier can fast identify the car in the parking space accurately.

A. The result of testing

In the testing process of the CNN Classifier in this paper and a Soft-Cascade AdaBoost Classifier [6] based on Haar-like, 858 samples of the cars, and 330 samples of the background are used, and the result is listed in the Table I.

TABLE I. RESULT OF TESTING

Types of Testing	Cars	Background	All samples
Numbers of Samples	858	330	1188
Correct rate of CNN	96.0%	97.5%	96.42%
Correct rate of Adboost based on Haar-like	82%	86%	83.11%

From the Table I, it can be seen that the correct rate of CNN has higher than Adboost's.

B. The Results of the Vehicle Localization

In the experiment, cars are detected and located in a large car park in Figure 7 using CNN net and in Figure 8 using Adboost approach mentioned in [5].

Figure 7. Results of car detection and localization of CNN classifier

Figure 8. Results of car detection and localization of Adaboost classifier based on Haar-like

Seen in Figure 7 and Figure 8, the CNN classifier can achieve a better result than the Adboost classifier [5].

C. The Analysis of the Results

From Figure 7, it can be seen that two cars which are similar to the background, are missed by the classifier. From other papers [7-17], it could be concluded that a reason of missing is that samples are too small. However, it is difficult to collect millions of samples due to environmental restrictions. In the experiment, to improve detection rate of cars which are similar to the background, it is necessary to increase the proportion of cars which are similar to the background in the training samples.

VII. CONCLUSION AND FUTURE WORK

This paper presents a car detection method based on the CNN technology. The experimental results show that this method can achieve a high accuracy of detection and location and that the algorithm is very efficient and can be used in real time. Information of parking cars and the locations can be used in car parking management systems in future work.

VIII. ACKNOWLEDGMENT

The authors gratefully acknowledge supports from Fujian Provincial Key Laboratory for Photonics Technology, and the fund from the Natural Science Foundation of China (Grant No.61179011) and Science and Technology Major Projects for Industry-academic Cooperation of Universities in Fujian Province (Grant No.2013H6008) , Fujian Province Department of Education(Grant:No.JA13073).

REFERENCES

[1] J.Z Liu, M. Mohandes, M. Deriche, “A Multi-Classifier Image Based Vacant Parking Detection System”, 2013 IEEE 20th International Conference on Electronics, Circuits, and Systems (ICECS), pp. 933-936, 2013.

[2] D. B. L. Bong, K. C. Ting, and K. C. Lai, “Integrated approach in the design of car park occupancy information system (COINS),” IAENG International Journal of Computer Science, vol. 35, pp. 7-14, 2008.

[3] Jermsak Jermsurawong, Mian Umair Ahsan, Abdulhamid Haidar, Haiwei Dong, Nikolaos Mavridis ”. Car Parking Vacancy Detection and Its Application in 24-Hour Statistical Analysis”. International Conference on Frontiers of Information Technology, 2012.

[4] Jake Bouvrie, “Notes on Convolutional Neural Networks,” Center for Biological and Computational Learning, Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, MA 02139, November 2006.

[5] Alberto Broggi, Elena Cardarelli, Stefano Cattani, Paolo Medici and Mario Sabbatelli, ”Vehicle Detection for Autonomous Parking using a Soft-Cascade AdaBoost Classifier”, 2014 IEEE Intelligent Vehicles Symposium (IV) Proceedings, pp. 912-917, 2014.

[6] Andrew Ng, Jiquan Ngiam, Chuan Yu Foo, Yifan Mai, Caroline Suen “UFLDL Tutoria”, April 2013.

[7] Karpathy, Andrej, Toderici, George, Shetty, Sanketh;Leung, Thomas, Sukthankar, Rahul;Fei-Fei, Li. “Large-Scale Video Classification with Convolutional Neural Networks”, Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, 2014.

[8] A. Krizhevsky, I. Sutskever, and G. Hinton. “Imagenet clas-sification with deep convolutional neural networks ”. In NIPS, 2012.

[9] L. N. Clement Farabet, Camille Couprie and Y. LeCun. “Learning hierarchical features for scene labeling”. PAMI, 35(8), 2013.

[10] D. Ciresan, A. Giusti, J. Schmidhuber, et al. “Deep neural net-works segment neuronal membranes in electron microscopy images”. In NIPS, 2012.

[11] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. “OverFeat: Integrated recognition, localization and detection using convolutional networks”. arXiv preprint arXiv: 1312.6229, 2013.

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik. “Rich fea-ture hierarchies for accurate object detection and semantic Segmentation ”. In CVPR, 2014.

[13] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carls-son. “CNN features off-the-shelf: an astounding baseline for recognition ”. arXiv preprint arXiv: 1403. 6382, 2014.

[14] Pedro H. O. Pinheiro, Ronan Collobert. “Recurrent Convolutional Neural Networks for Scene Parsing”. the 31st International Conference on Machine Learning, Beijing, China, 2014.

[15] Matthew D Zeiler, Rob Fergus. “Visualizing and Understanding Convolutional Networks”. ECCV 2014, Part I, LNCS 8689, pp. 818-833, 2014.

[16] Xueyun Chen, Shiming Xiang, Cheng-Lin Liu, and Chun-Hong Pan. “Vehicle Detection in Satellite Images by Hybrid Deep Convolutional Neural Networks”, IEEE GEOSCIENCE AND REMOTE SENSING LETTERS, VOL. 11, NO. 10, OCTOBER, 2014.

[17] Xueyun Chen, Shiming Xiang, Cheng-Lin Liu, and Chun-Hong Pan. “Vehicle Detection in Satellite Images by Parallel Deep Convolutional Neural Networks”, 2013 IEEE DOI 10.1109/ACPR. 2013.