차량 번호판 검출을 위한 2단계 합성곱 신경망 접근법*

크리스찬 거버* 정목동*
*부경대학교 컴퓨터공학과
christian.gerber@sunrise.ch, mdchung@pknu.ac.kr

Number Plate Detection with a 2-step Neural Network Approach for Mobile Devices

Christian Gerber* Mokdong Chung*
*Dept. of Computer Engineering, Pukyong National University

Abstract

A method is proposed to achieve improved number plate detection for mobile devices by applying a two-step convolutional neural network (CNN) approach. Supervised CNN-verified car detection is processed first. In the second step, we apply the detected car regions to the second supervised CNN-verifier for number plate detection. Since mobile devices are limited in computing power, we propose a fast method to detect number plates. We expect to use in the field of intelligent transportation systems (ITS).

1. Introduction

Acquiring real-time number plate localization with a high detection rate in a natural traffic environment is still a widely researched area in computer science. Recent research results [1, 2, 3] show that CNNs provide a high detection - and a low false positive rate among classifying images. Chen et al. [1] proposed a CNN-based verifier for number plate detection by processing a small, number plate conformed sliding sub window on the whole input image. Instead of detecting license plates directly, we reduce the neural network computations by applying a bigger sliding window to localize cars first. The detected car regions are the input for the second supervised CNN to detect number plates. Our approach, compared to single CNN-based number plate detectors, provides a high detection rate by reducing the overall neuron calculations. Therefore, number plates can be detected on mobile devices, based on neural network classifiers, within a fast changing real world environment. Intelligent transportation systems (ITS) are becoming more important since the amount of traffic is growing. Number plate detection is important for modelling and tracking the traffic flow. In section 2, we compare our proposed method to similar approaches. Section 3 shows the method and in section 4 are the results. The discussion is in section 5 and the final section is the conclusion.

2. Related Work

Chen et al. [1] proposed a number plate detection using a single-stage, single-scale CNN. This approach is looking for text features by applying a square shaped sliding windows over the full input image. The sliding window is looking for two full characters, which requires a minimum resolution and a small sliding window step size (vertical, horizontally). Approaches using a sliding window for object detection are slow due to the evaluation of overlapping image regions. Our method uses a single-scale CNN for both car and plate detection. The searching area for license plate is reduced to a car detected image region by applying car detection first. The car detection uses a bigger sliding window step size as well as a bigger input image for convolution, which reduces the overall classification amount. Li et al. [2] proposed a multi-scale CNN architecture approach. The classifier can be fed with the features extracted by multiple stages. This gives the advantage for feeding the classifier with different scales of receptive fields. The whole input image has to be divided into small steps sliding windows to detect the license plate. With our approach we reduce the input image classification by applying a CNN for car detection first (Table 1).

<Table 1> Comparison of similar plate detection approaches

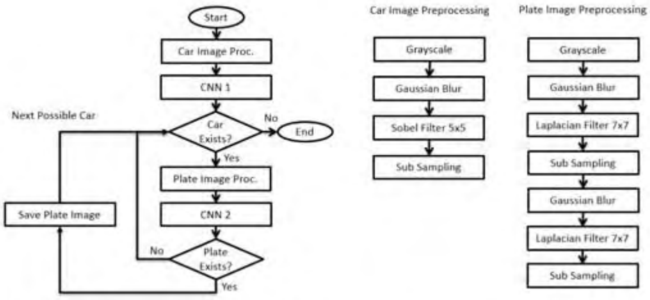
	Li et al. [2]	Chen et al. [1]	2-step CNN
CNN	multi-scale, multi-stage	single-stage, single-scale	multi-stage, single-scale
Adv.	reduced classification steps	fast learning	reduced classification steps, fast learning
Disadv.	slow learning	requires min. resolution, small sliding window steps	plate detection depends on car detection results

3. Methods

3.1 Design

The license plate is detected by a two-step CNN approach. We first perform a supervised CNN for car detection (CNN1) and second a supervised CNN for number plate detection (CNN2). CNN2 takes the detected car regions as input. The detected car plate is saved and we apply CNN2 for the next car-containing region. If there are no more existing car regions, the algorithm will terminate (Fig.1).

· This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2010-0024053).



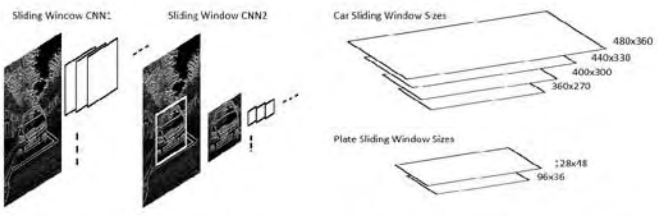
(Fig. 1) 2-step CNN number plate detection flow diagram

3.2 Convolutional Neural Network

A CNN consists of one or multiple stages of image processing and a neural network as classifier. One stage consists of a convolution step followed by a subsampling step. The convolution step usually convolutes the input data with multiple different filters to extract features. The sub sampling layer summarizes detected features into a feature map and reduces the dimension of the convoluted images from the previous step.

3.3 Car Localization

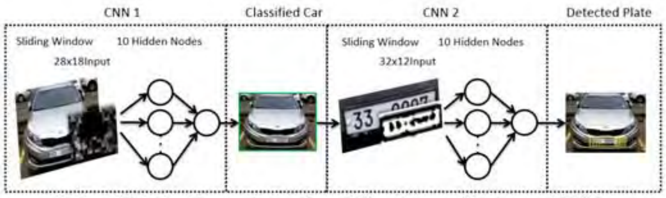
The video input stream is acquired using a session capture of a 640x480 pixels resolution color images. The images of the video stream are converted to gray and convoluted by a Sobel 5x5 kernel filter to extract edges and corners. A sliding window starts at the upper left corner and proceeds with a step of 20 pixels vertically and horizontally with four different resolutions (Fig.2). We acquire the input image for CNN1 by down scaling the convoluted sliding window output to a resolution of 28x18 pixels. Therefore, CNN1 is a single-stage CNN with only one convolution layer for feature extraction. The classification for CNN1 is a fully connected multilayer perceptron (MLP), with one hidden layer and 10 hidden nodes and trained by applying back propagation [4] with a given training set of 121 front car images and 2578 non-car images. We perform sliding windows of different resolutions to detect cars of variable sizes (Fig.2).



(Fig. 2) Two sliding windows to generate candidates for the classifiers

3.4 Plate Localization

Once the car image regions are detected, we apply CNN2 for number plate detection onto the car detected sub image from CNN1. A sliding window with a vertical and horizontal step of 4 pixels is used as for CNN2 with two different resolutions. For the single CNN plate classification we use sliding windows with 128x48 and 96x36 pixels resolutions, whereas the sizes of the candidates within the 2-step CNN approach are relativ to the size of the detected car sliding window (Fig.2). CNN2 is a two-stage CNN and in every stage there is a Gaussian blur followed by a Laplacian 7x7 filter convolution and a subsampling step. The classification layer for CNN2 is a supervised MLP, where the training is done by back propagation with a given training set of 20 narrow plate images and 1699 non-plate images (Fig.3).



(Fig. 3) Number plate classification with two CNNs

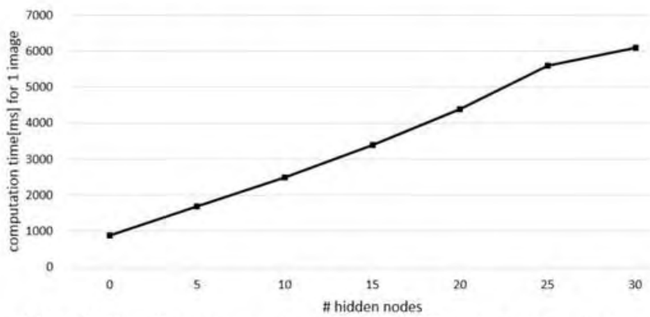
4. Results

Detecting the car number plate with only one CNN as a classifier, similar to the method by Chen et al. [1], results in computing 578560 neurons per image. With the 2-step CNN approach, the neuron calculation amount is 3660 24x18 input neurons and 48,200 32x12 input neurons per image, in the case of only one detected car region of size 480x360 pixels (Table 2).

<Table 2> Neuron calculations of single CNN, 2-step CNN

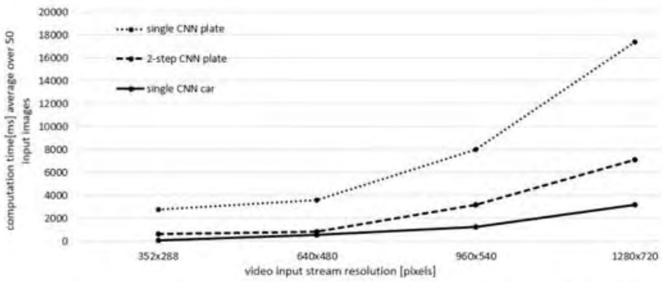
	Single CNN	2-step CNN
CNN1 24x18 Input Neuron	-	3660
CNN2 32x12 Input Neuron	578,560	48,200
Total Neuron Evaluation	578,560	51,860

Both, the car and the plate classifier, consist of one hidden layer with 10 hidden nodes. By increasing the amount of hidden neurons we would also increase the overall calculation time (Fig.4).



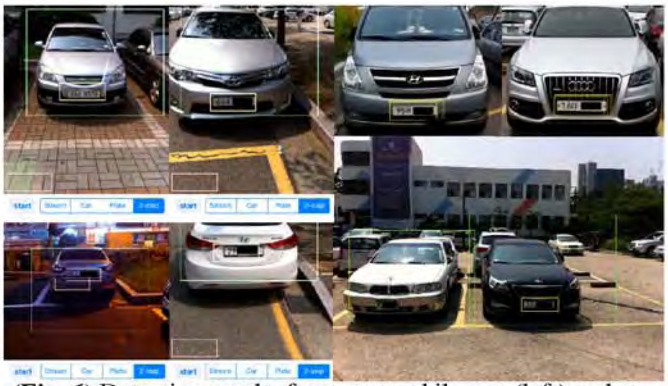
(Fig. 4) Car detection computation performance in relation with the number of hidden nodes (iPhone 4: 1GHz Cortex A-8 CPU, 512 Mb RAM [5], 640 x 480 pixels input)

We compared the 2-step CNN approach with our single CNN approach that is similar to the approach by Chen et al. [1] in terms of classification on real traffic environment front car images (Fig.5).



(Fig. 5) Car and plate detection performance of single and 2-step CNN approach in relation to the input size (iPhone 4: 1GHz Cortex A-8 CPU, 512 Mb RAM [5])

The detection rate of our single CNN Plate detector detects more than 90% of the occurring number plates on our car training data (Fig.6) with narrow number plates (Table 4).



(Fig. 6) Detection results from our mobile app (left) and our car training data (right)

The detection rate of the combined classifier is lower than the single CNN plate classifier because the car detector fails to detect all occurring cars (Table 4).

<**Table 4**> Detection rate of our method applied on different still image databases

Number Plate	**Detection Rate**	**False Positives**
Car Training Data (80 images)	91.25 %	84
Car		
Caltech [6] 99 (125 rear images)	96 %	6
Caltech [6] 01 (526 rear images)	88 %	26

However, real-time experiments in a natural traffic environment show that our car detector detects 98% of the front and back of a car, appearing on a video input stream of size 640x480 pixels, once it is within range of our sliding windows.

5. Discussion

With our proposed method, we trained the car detector with car front images only. Extending this method with an unsupervised CNN would allow us to detect other kind of vehicles, as well as detection of vehicles and plates with rotation or angle. Using the Laplacian and Sobel filters makes our detector insensitive to brightness variation, uneven illumination and low contrast. Reducing the searching area for number plates within the detected car region further, would reduce the overall classification steps. Cars that appear partially occluded within the video input image are not detected and therefore the algorithm will not search for a number plate. Input images of cars covered with shadows from the near environment have a higher false positive rate and a lower detection rate. Wider, more square-shaped plates and vehicles with a certain angle, rotation or with different shapes such as Sports Utility Vehicles (SUVs), mini-vans, trucks etc. have a low detection rate due to missing training data.

6. Conclusion

We proposed a neural network based method for real-time number plate localization on a mobile device. Real-time classification of image input data, classified by neural networks and sliding windows is costly and therefore not suited for mobile devices. By reducing the computation amount, we showed that our proposed method performed real-time license plate detection on a mobile device with a high detection rate. In the future, we are going to improve our classifiers in terms of rotation, angle and shape.

References

[1] Chen, Y.N., Han, C.C., Wang, C.T., Jeng, B.S., Fan, K.C., The Application of a Convolution Neural Network on Face and License Plate Detection. Proc. of Int. Conf. on Pattern Recognition, pp. 552–555 (2006).

[2] Jia Li, Changyong Niu, Ming Fan. J. Wang, G.G. Yen, and M.M. Polycarpou (Eds.), Multi-scale Convolutional Neural Networks for Natural Scene License Plate Detection. ISNN 2012, Part II, LNCS 7368, pp. 110–119, 2012. c_Springer-Verlag Berlin Heidelberg (2012).

[3] Ian J. Goodfellow, Yaroslav Bulatov, Julian Ibarz, Sacha Arnoud, Vinay Shet, Multi-digit Number Recognition from Street View Imagery using Deep Convolutional Neural Networks. Google Inc., Mountain View, CA arXiv: 1312.6082v3 [cs.CV] (2014).

[4] LeCun, Y., Bottou L., Bengio Y. Haffner P., Gradient-Based Learning Applied to Document Recognition. Proc. of the IEEE 86(11), 2278-2324 (1998).

[5] Apple Inc., Part of iOS 6 Developer Library.

[6]California Institute of Technology, Caltech: http://www.vision.caltech.edu/Image_Datasets