End-to-End View-Aware Vehicle Classification via Progressive CNN Learning

Jiawei Cao, Wenzhong Wang, Xiao Wang, Chenglong Li, and Jin Tang(✉)

School of Computer Science and Technology,
Anhui University, No. 111 Jiulong Road, Hefei 230601, China
ahujwcao@foxmail.com, wenzhong@ahu.edu.cn, wangxiaocvpr@foxmail.com,
lcl1314@foxmail.com, jtang99029@foxmail.com

Abstract. This paper investigates how to perform robust vehicle classification in unconstrained environments, in which appearance of vehicles changes dramatically across different angles and the numbers of viewpoint images are not balanced among different car models. We propose a end-to-end progressive learning framework, which allows the network architecture is reconfigurable, for view-aware vehicle classification. In particular, the proposed network architecture consists of two parts: a general end-to-end progressive CNN architecture for coarse-to-fine or top-down fine-grained recognition task and an end-to-end view-aware vehicle classification framework to combine vehicle classification and viewpoints recognition. We test the technique on a large-scale car dataset, "CompCars", and experimental results show that our framework can significantly improve performance of vehicle classification.

Keywords: Deep learning · Vehicle classification
Viewpoint recognition · Convolutional neural networks

1 Introduction

Vehicle classification is one of the most important tasks for video surveillance, especially for intelligent transportation system. It has a wide range of applications for our daily life. For example, the highway toll system can charge by the vehicle types. For another example, effective vehicle classification can play a very important supplementary role in vehicle retrieval. But it's a challenging task for fine-grained vehicle classification because of viewpoint variation: cars are 3D objects whose appearances change dramatically across different angles.

Recently, Duan et al. [4] propose a method based on K-means clustering for estimate vehicle viewpoints in vehicle recognition. The authors of [2,3,17] all present that visual based vehicle type classification task should be studied from frontal view and side view. But the only two viewpoints are not enough and almost all of current methods cannot play well in practical traffic scenes because of various views as seen in Fig. 1. In fact, if our designed vehicle classification

J. Yang et al. (Eds.): CCCV 2017, Part I, CCIS 771, pp. 729–737, 2017.
https://doi.org/10.1007/978-981-10-7299-4_61

Fig. 1. The vehicles in practical scenes have various viewpoints.

algorithm can be truly applied into the intelligent transportation system, we should deal with the different views together with only a united method.

To handle above issues, we propose an expressive method to address the above mentioned issues for vehicle classification in the practical complex scenes. Our method mainly contains the following two parts: a progressive CNN Architecture and an end-to-end view-aware vehicle classification framework. At the first step, we develop a progressive CNN Architecture for learning the two-level hierarchical and multi-task CNN that can be used for a coarse-to-fine or top-down recognition task. At the second step, we propose an end-to-end view-aware vehicle classification framework for combining vehicle classification with viewpoints recognition. In our hierarchical framework, we first perform viewpoints recognition in our base network and then the data will be passed into corresponding sub-networks according to the viewpoint. Then the sub-networks will classify the categories of cars. The base network and sub-networks are two configurable components in our framework. We deploy different state-of-the-art convolutional neural networks of image classification as two components and the experimental results show that our framework can significantly improve performance of vehicle classification in various viewpoints.

In summary, this paper has the following contributions. First, we introduce a general progressive CNN architecture for coarse-to-fine or top-down recognition task. Second, we develop a scheme to achieve end-to-end training and testing for above progressive CNN architecture through setting run level for layers in CNN. Third, we propose an end-to-end view-aware vehicle classification framework. We have performed evaluations on the large-scale car dataset CompCars, and our method has achieved state-of-the-art performance.

2 Related Work

Vehicle classification has achieved great development in recent years and many algorithms has been proposed for vehicle classification. These algorithms can be roughly fall into two categories: *image-based methods* and *video-based methods*. Image-based methods directly process with the handcrafted or detected vehicle images. Video-based methods will consider the corresponding relationships

between the frames. The authors of [8,11,16] both use video information for vehicle type classification task. But they don't take full advantage of the corresponding relationships between the frames well. There exists several types of imaged-based methods, namely, model reconstruction method, feature representation method and convolutional neural network. Model reconstruction methods [8,14,20] estimate 3D measurement parameters for the size of a vehicle to reconstruct a three dimensional model of the vehicle. Feature representation approaches [2,4,5,15,17] extract features to represent the vehicle appearance for subsequent classification. CNN-based models [7,12,19] hold state-of-the-art performance in various computer vision tasks and the authors of [3,10,18,21] utilize CNN instead of the classical methods which depend on hand-crafted features and achieved new state-of-the-art performance in classification problem.

As for Vehicle viewpoint recognition, Chen and Lu [1] propose a discriminative viewpoint-specific component model to estimate viewpoint. Ricardo *et al.* [6] combine vehicle detection with viewpoint estimation based on the motion model. Little work has explored combining vehicle classification with viewpoints recognition and most of methods about viewpoint recognition are based on unsupervised clustering. The estimated viewpoints are rough with not high enough accuracy.

3 The Proposed Approach

Our proposed approach for vehicle classification and viewpoints recognition contains two parts: a progressive CNN architecture and an end-to-end view-aware vehicle classification framework for combining vehicle classification and viewpoints recognition. First, we develop a general and progressive CNN architecture. The general architecture is two-level and hierarchical for progressive recognition tasks. Then we design a scheme to achieve end-to-end training and testing. At last, we propose a model for end-to-end view-aware vehicle classification. We will introduce them in the following subsections, respectively.

3.1 Progressive CNN Architecture

We develop a general scheme for learning the two-level hierarchical CNN as illustrated in Fig. 2. It's designed for progressive recognition tasks. There is a top category CNN classifier for task A with K categories in base network. The next are K sub-networks for task B corresponding to the K top categories in task A. Thus, it's a two-level and hierarchical architecture. The general architecture can apply in many object recognition tasks. For example, we can use the architecture for fine-grained classification. We can take task A in base network as a coarse classifier to separate easy classes from one another. More challenging classes are routed downstream to fine category classifiers for task B in sub-networks that focus on confusing classes.

The architecture should be end-to-end in training and test phrases. Thus, we need a CNN architecture where data can flow dynamically and some layers in

neural network will switch activation and deactivation accordingly. Caffe [9] is a deep learning framework made with expression, speed, and modularity in mind. Lots of researchers and engineers have made Caffe models for different tasks with all kinds of architectures and data. Thus we choose to realize our architectures based on Caffe. But Layers of CNN in the Caffe architecture can't play a selective role dynamically. Hence, if we deploy some branch networks, the data will flow through each sub-network. At the same time, the backward propagation will work in each sub-network. We need the neural network could automatically decide whether to be activated or inactive according to the classification results in base network.

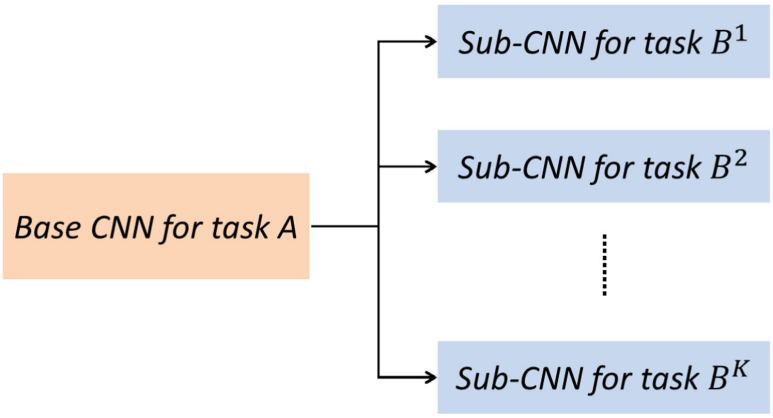

Fig. 2. The architecture of two-level hierarchical CNN.

For solving this problem, we design a novel scheme that we set different run levels for layers in CNN based on Caffe. We set run level 0 for all layers in basic network and the layers with run level 0 will always execute forward and backward propagation along with flowing data. Then we set bigger run level such as 1, 2, 3 for corresponding sub-networks and the run levels of layers in same sub-network are identical. In our new architecture, the layers whose run level isn't 0 will execute forward and backward propagation selectively according to the output of classifier in basic network. For example, in test phase, if the classifier in basic network predict that the input car image belongs to the third top category, then all layers in third sub-network (the run level is 3) will transfer data and execute forward propagation. At the same time, layers in the others sub-networks (the run level is not 3 or 0) will be closed so these layers can't perform any actions or participate in the calculation.

It is also worthy to note that the whole architecture is a general pipeline and could be trained and tested in the end to end fashion.

3.2 End-to-End View-Aware Vehicle Classification

Overall Architecture. We design an end-to-end view-aware vehicle classification CNN which combines vehicle classification with viewpoints recognition based on above architecture. As illustrated in Fig. 3, the overall architecture consists of two components: a basic network for viewpoints recognition and a series of sub-networks for vehicle classification. The two components are independent and configurable.

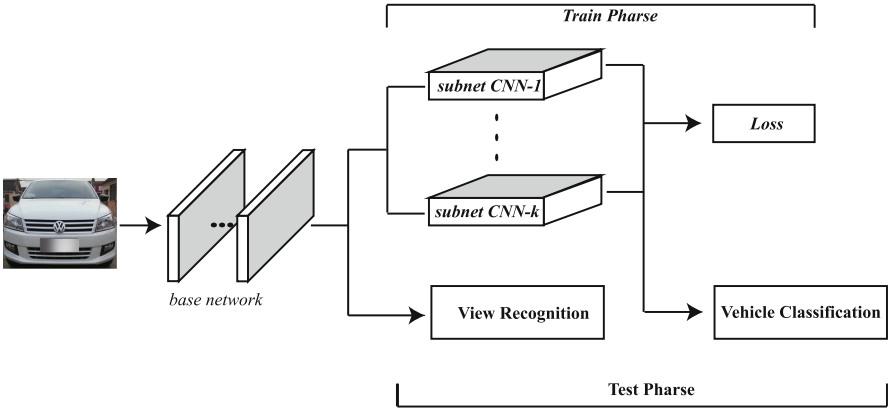

Fig. 3. Our progressive CNN framework.

Basic Network for Viewpoints Recognition. In training phrase, the data layers in this network take car images and labels as input. The labels in this network are annotations of viewpoints. The run level of this network is zero, hence it will be activated all the time. In another words, we will perfom viewpoints recognition for all car images. At last, there is a softmax layers which is not only a view classifier but also a selection switch to active corresponding branch network in sub-networks. We will switch corresponding sub-network in train phrase according to groundtruth of viewpoints. While testing, the network will active corresponding run level based on the output of view classifier. We choose the state-of-the-art CNN ResNet [7] in image classification task to build the base network.

Sub-networks for Vehicle Classification. There are a series of parallel sub-networks for vehicle classification following on the basic network. There is only one sub-network will be activated at the same time. We deploy the popular neural networks, including AlexNet, GoogleNet and ResNet-50 as sub-networks respectively. Thus, there are three CNN models and the sub-networks have same architecture in each model.

Training. The train phrase is end-to-end and we use stochastic gradient descent algorithm to train our network. The training images within the stochastic gradient descent minibatch are probabilistically routed to different sub-networks because of different viewpoints, we set minibatch size 1 and don't use batch gradient descent. The loss function we use is shown below.

$$Loss = \alpha Loss1 + \beta Loss2, \tag{1}$$

where α, β are tuning parameters. *Loss1* is the cross-entropy loss of viewpoint classifier in basic network's softmax layer and *Loss2* is the cross-entropy loss of cars classifier in each sub-network. In our network, we set α 0.3 and β 0.7.

4 Experiments

4.1 Experiment Settings

To implement our view-aware vehicle recognition system, a computer with Xeon E5-2620 v4 CPU, 64 GB memory and 8 GB memory GTX1080 GPU is employed. The program runs on a 64-bit Ubuntu LTS 14.04 operating system with CUDA-8.0, Python 2.7.6 installed. In train phrase, we set learning rate 0.0001 for the three CNN models.

Dataset Selection. We choose a large-scale public dataset CompCars [13] as our experimental data. In addition, we select 100 car models and take 61477 images as train set and 9808 images as test set. As for viewpoints of cars, we use five viewpoints, including front, rear, side, front-side and rear-side. The quantity distribution of the car images in different viewpoints is shown in Table 1.

Table 1. Quantity distribution of the car images of different viewpoints.

Viewpoint	Total images
Front	2616
Rear	2000
Side	1547
Front-side	1888
Rear-side	1757

4.2 Experimental Results

Comparison with State-of-the-art Methods. To compare our framework with state-of-the-art classification CNN models, we first train AlexNet, GoogleNet and ResNet-50 CNN models with car images in all viewpoints respectively. In these CNN models, a single category of the vehicle brand will contain

different viewpoints. There are total 100 car categories according to car brand. Then we deploy basic network of viewpoints recognition with ResNet-50 and the final testing accuracy of viewpoints recognition is 0.963. For the five sub-networks in different views, the AlexNet, GoogleNet and ResNet-50 CNN models are deployed respectively compared to above models. The performances of these models are summarized in Table 2. As can be seen from the Table 2, with decomposing the classification task in all views into several sub-tasks in a series of viewpoints, our CNN models significantly outperform the baseline models.

Table 2. Vehicles classification performance.

Models	Top-1 error	Compared to baseline
AlexNet	40.15%	-
GoogleNet	10.21%	-
ResNet-50	7.81%	-
Ours based on AlexNet	16.42%	−23.73%
Ours based on GoogleNet	6.89%	−3.32%
Ours based on ResNet-50	6.35%	−1.46%

Comparison over the Components. We analyze vehicles classification performance in different views using AlexNet and our CNN model based on AlexNet. As illustrated in Table 3, we can find the recognition accuracy in rear view is obviously lower than other views because of the different appearance variation in different views. For example, many car models are very similar in their side views but rather different in the front views as shown in Fig. 4. When we divide images into different views and perform car classification individually in our progressive architecture, the inter-class difference reduces and classification performance improves significantly in each viewpoints.

Table 3. Vehicles classification performance in different views.

Viewpoint	Alexnet top-1 error	Our method top-1 error
Front	29.59%	8.83%
Rear	39.2%	9.65%
Side	50.1%	26.96%
Front-side	42.32%	20.29%
Rear-side	45.87%	20.72%

Fig. 4. The two car models are more easily distinguished in their front views than side views.

5 Conclusion

In this paper, we proposed a general end-to-end deep neural networks for fine-grained vehicle classification. Aiming at jointly training the viewpoint classification and brand recognition issue, we deploy the base-network and following sub-CNNs for these two targets, respectively. Our neural network could not only utilized in vehicle classification, but also some other tasks. Experiments on the public benchmark validated the effectiveness of the proposed network compared with existing popular classification networks.

Acknowledgement. This work was supported in part by the National Natural Science Foundation of China under Grant 61671018 and Grant 61472002, in part by the Natural Science Foundation of Anhui Higher Education Institution of China under Grant KJ2017A017 and in part by the Co-Innovation Center for Information Supply & Assurance Technology of Anhui University under Grant Y01002449.

References

1. Chen, T., Lu, S.: Robust vehicle detection and viewpoint estimation with soft discriminative mixture model. IEEE Trans. Circuits Syst. Video Technol. **27**(2), 394–403 (2017)
2. Dong, Z., Jia, Y.: Vehicle type classification using distributions of structural and appearance-based features. In: 2013 20th IEEE International Conference on Image Processing (ICIP), pp. 4321–4324. IEEE (2013)
3. Dong, Z., Wu, Y., Pei, M., Jia, Y.: Vehicle type classification using a semisupervised convolutional neural network. IEEE Trans. Intell. Transp. Syst. **16**(4), 2247–2256 (2015)
4. Duan, K., Marchesotti, L., Crandall, D.J.: Attribute-based vehicle recognition using viewpoint-aware multiple instance svms. In: 2014 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 333–338. IEEE (2014)

5. Fu, H., Ma, H., Liu, Y., Lu, D.: A vehicle classification system based on hierarchical multi-svms in crowded traffic scenes. Neurocomputing **211**, 182–190 (2016)
6. Guerrero-Gómez-Olmedo, R., López-Sastre, R.J., Maldonado-Bascón, S., Fernández-Caballero, A.: Vehicle tracking by simultaneous detection and viewpoint estimation. In: Ferrández Vicente, J.M., Álvarez Sánchez, J.R., de la Paz López, F., Toledo Moreo, F.J. (eds.) IWINAC 2013. LNCS, vol. 7931, pp. 306–316. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38622-0_32
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE CVPR, pp. 770–778 (2016)
8. Hsieh, J.W., Yu, S.H., Chen, Y.S., Hu, W.F.: Automatic traffic surveillance system for vehicle tracking and classification. IEEE Trans. Intell. Transp. Syst. **7**(2), 175–187 (2006)
9. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM international conference on Multimedia, pp. 675–678. ACM (2014)
10. Jiang, C., Zhang, B.: Weakly-supervised vehicle detection and classification by convolutional neural network. In: International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), pp. 570–575. IEEE (2016)
11. Jiang, L., Zhuo, L., Long, H., Hu, X., Zhang, J.: Vehicle classification for traffic surveillance videos based on spatial location information and sparse representation-based classifier. In: 2016 International Conference on Progress in Informatics and Computing (PIC), pp. 279–284. IEEE (2016)
12. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
13. Yang, L., Luo, P., Loy, C.C., Tang, X.: A large-scale car dataset for fine-grained categorization and verification. In: IEEE CVPR, pp. 3973–3981 (2015)
14. Ma, X., Grimson, W.E.L.: Edge-based rich representation for vehicle classification. In: Tenth IEEE International Conference on Computer Vision, ICCV 2005, vol. 2, pp. 1185–1192. IEEE (2005)
15. Manzoor, M.A., Morgan, Y.: Vehicle make and model classification system using bag of sift features. In: 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), pp. 1–5. IEEE (2017)
16. Mei, X., Ling, H.: Robust visual tracking and vehicle classification via sparse representation. IEEE TPAMI **33**(11), 2259–2272 (2011)
17. Peng, Y., Jin, J.S., Luo, S., Xu, M., Cui, Y.: Vehicle type classification using pca with self-clustering. In: 2012 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), pp. 384–389. IEEE (2012)
18. Yu, S., Wu, Y., Li, W., Song, Z., Zeng, W.: A model for fine-grained vehicle classification based on deep learning. Neurocomputing (2017)
19. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: IEEE CVPR, pp. 1–9 (2015)
20. Zhang, Z., Tan, T., Huang, K., Wang, Y.: Three-dimensional deformable-model-based localization and recognition of road vehicles. IEEE Trans. Image Process. **21**(1), 1–13 (2012)
21. Zhou, Y., Cheung, N.M.: Vehicle classification using transferable deep neural network features. arXiv preprint arXiv:1601.01145 (2016)