Neurocomputing 243 (2017) 60–68

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Location-aware fine-grained vehicle type recognition using multi-task deep networks

Bin Hu[a,b], Jian-Huang Lai[a,c,*], Chun-Chao Guo[a,b]

[a] Sun Yat-sen University, Guangzhou, 510006, P.R. China
[b] Guangdong Key Laboratory of Information Security, Guangzhou, 510006, P.R. China
[c] XinHua College, Sun Yat-sen University, Guangzhou, P.R. China

ARTICLE INFO

Article history:
Received 3 November 2016
Revised 26 February 2017
Accepted 27 February 2017
Available online 8 March 2017

Communicated by Dr Jiwen Lu

Keywords:
Vehicle type recognition
Fine-grained vehicle
Unconstrained-view vehicle
Vehicle localization
Multi-task CNNs

ABSTRACT

Distinct from conventional approaches to generic vehicle type recognition, this paper proposes an approach to fine-grained vehicle type recognition that leverages multiple cues jointly to convolutional neural networks (CNNs). Fine-grained vehicle recognition is inherently difficult with many challenges, including (i) how to incorporate multiple cues through joint optimization instead of separately handling, (ii) how to learn data-driven representation for vehicles instead of hand-crafted features and (iii) how to perceive and localize the vehicle region instead of using the whole image. Our multi-task CNNs localize vehicles in the first stage and recognize subclasses in the second stage, allowing us to handle samples with cluttered backgrounds and those where vehicles do not fill most of the image. Through collaborative feature learning from multiple tasks in each stage, our approach can handle subtle inter-class variations at the subordinate level. We also develop new vehicle-oriented data augmentation strategies in CNN training. To advance research on vehicle-centered tasks, we release a new vehicle dataset that provides both semantic labels and bounding boxes. This dataset can be used in vehicle localization, recognition, viewpoint estimation, and so on. Extensive experiments on two benchmark vehicle datasets demonstrate that our approach outperforms state-of-the-art algorithms.

1. Introduction

Vehicle type recognition is becoming increasingly important for a wide range of vehicle-centered applications, such as advanced driver assistance systems (ADAS), large-scale vehicle searches, automatic traffic flow monitoring, and so on. Notable progress has been achieved in generic vehicle type recognition that aims to categorize coarse-grained types. However, we have not witnessed rapid advances in fine-grained vehicle type recognition, which can be even more significant and challenging. To further benefit vehicle-oriented tasks, this paper advocates recognizing fine-grained types instead of coarse-grained labels.

The existing vehicle type recognition solutions are insufficient, and many challenges still remain. First, most approaches [1–3] in this field use traditional hand-crafted descriptors such as SIFT [4], PHOG [5], Gabor filters, etc. Despite the poor generalization of many hand-crafted features, they require expensive domain or even scene knowledge, making it difficult to fully mine the discriminative cues hidden in data. Second, the need to localize vehicle regions is often ignored, despite being of equal importance when compared with the stage of vehicle classification. For multi-shot vehicle recognition [6,7], vehicle regions are usually delineated through background subtraction across video frames. However, for single-shot vehicle recognition [8–10], images with pre-labeled position are usually cropped straightly. Although the idea of sweeping off the background is reasonable, it does not achieve good performance if there are no pre-labeled positions. Another reason may be that it is rather difficult to train a generic vehicle detector to recognize all vehicle types. Nevertheless, when a vehicle occupies a small ratio in an image, or the background is cluttered, the performance of many approaches can degrade dramatically. Third, many vehicle recognition systems [1,3,11] focus on exploiting multiple cues, but they do so by separately extracting patterns and directly concatenating them for the final recognition. Joint optimization of complementary cues has not been comprehensively explored for vehicle-centered tasks.

Compared with coarse-grained vehicle type recognition, fine-grained recognition is much more difficult. Hundreds of subclasses must be categorized despite sharing global similarities in shape and structure. This requires that subtle differences should be found between subclasses of vehicles to overcome large intra-class and

* Corresponding author at: School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, PR China.
E-mail addresses: huglenn232@gmail.com (B. Hu), stsljh@mail.sysu.edu.cn (J.-H. Lai), chunchaoguo@gmail.com (C.-C. Guo).

http://dx.doi.org/10.1016/j.neucom.2017.02.085
0925-2312/

small inter-class variations. We propose a unified framework to deal with fine-grained vehicle recognition based on a single shot. Our framework comprises two deep convolutional neural networks (CNNs) used for vehicle localization and fine-grained recognition, both of which use multi-task optimization. We also develop new vehicle-oriented data augmentation strategies in CNN training to further improve the performance.

The three main contributions of this work are as follows. First, we advocate recognizing vehicles in a fine-grained manner and develop a unified framework comprising two deep CNNs. The data-driven features learned from the CNNs are then used for vehicle representation. Second, we exploit multiple discriminative cues in both stages of our approach via multi-task learning. With the help of our joint optimization scheme across the two stages, our approach can localize vehicle regions in the first stage and leverage more patterns of representation learning in the second, both of which benefit the final recognition. Third, we release a new vehicle dataset named SYSU-Vehicle to advance research on vehicle-centered tasks, such as vehicle localization and type recognition. Bounding boxes, viewpoint labels, and type labels are provided. As we know, there are some public datasets for vehicle type recognition collected from video sequences, but those images always capture each vehicle from a single viewpoint. Our released dataset enriches these video-based datasets by providing more cues for vehicle classification.

The remainder of this paper is organized as follows. Section 2 reviews closely related works on vehicle type recognition. Sections 3 and 4 describe our location-aware multi-task deep learning framework. In Section 5, we present the experimental results on the CompCars and SYSU-Vehicle datasets. Finally, our conclusions are drawn in Section 6.

2. Related work

Recent research on vehicle images has received great interest in computer vision, especially vehicle detection [12,13], tracking [14,15], pose estimation [16–18], and attribute analysis [19,20]. However, the hottest research area is classification [1–3,21]. There are two primary steps in this task: feature extraction and an efficient classification strategy. This section focuses on these two steps.

2.1. Feature representation

Robust vehicle representation, as the significant procedure for vehicle image classification, concentrates on catching discriminative information for every class. Generally, the Texton [4,22] and edge features are the top two most popular features for this task. For example, Chen et al. [9] used multiple shape and size information such as aspect-ratio, silhouette and perimeter to construct vehicle representation. Zhang et al. [1], however, focused on Texton features. In their algorithms, Gabor filters and PHOG [5] were exploited. SIFT [4] is another widely used and discriminative feature due to its robustness on image scale. Ma and Grimson [2] proposed a modified-SIFT feature. They first searched edge key points with a Canny descriptor and then extracted the SIFT feature. Finally, a mean-shift technique [23] was used to set up the feature pool. In [24], Wen et al. proposed an ROI-based Haar-Like feature. They used a new feature selection method based on Adaboost to extract the more discriminative feature. Pearce and Pears [3] used canny edges, Harris corner, and SMG, a new feature descriptor to classify vehicle type. However, their method needs human annotation for ROI selection and only works for frontal-view vehicle images. Furthermore, some other features used in vehicle detection and attribute analysis could also be used in classification. Lee et al. [20] focused on discovering style-aware mid-level representation of vehicle images. Through a matching technique, they constructed many feature clusters, Then looked inside each cluster to mine style-sensitive vision elements using instance distribution. This method could also be used in classification.

The above mentioned features have three main drawbacks. First, many of them are not robust enough as they only describe targets in certain ways. For example, edge and shape descriptors cannot demonstrate concrete Texton information. Second, the extraction rate of some is slow, such as HOG and denseSIFT. Third, many have a complex extraction process because they need to combine multi-cue information and make selections.

In recent years, CNNs have attracted attention for their capabilities in many fields, such as image classification [37], object detection [41], person re-identification [43], etc. The reason for this lies in their ability to learn discriminative features from raw data inputs. Currently, deep learning methods have also been used in vehicle image classification. Dong et al. proposed an unsupervised [21] and a semisupervised [25] method to extract deep features. Zhang et al. [8] used CNNs with a proposed data enhancement technique and pre-trained method. Moreover, Krause et al. [11] focused on mining discriminative part expression. Like [20], they used a clustering technique to localize useful vision patches, then ensembled all of the patches and global deep feature to construct the discriminative description.

2.2. Classification strategies

Classification strategies aim to output a reliable vehicle label given a feature signature. There are substantial basic classification techniques, such as SVM, AdaBoost, LDA and Bayesian. Zhang [1] used Adaboost to assemble multiple strong classifiers. In every stage of their framework, maximum voting was used to combine the results of all of the classifiers, including SVM, kNN (*k*-nearest neighbors), and NN (neural network). Kuo and Nevatia [26] used a tree-structured classifier to cascade Adaboost. Fu et al. [27] proposed a hierarchical multi-SVMs for vehicle classification, in which they decomposed the task by two SVMs. The first-order SVM is to classify the region that contains single target while the second-order SVM aims at multi-target region. Zhu et al. [28] also focused on SVM. They proposed a linear programming formulation for υ-nonparallel support vector machine, termed as υ-LPNPSVM. In [19], an extended MI-SVM with pairwise constraints was used to catch vehicle viewpoints and discriminative vision patches. Hu et al. [10] proposed a dual-layer classification framework. They formally split an image into several parts, then trained an LDA detector for each part. Finally, all of the LDA detectors' scores were integrated into a feature vector and put into SVM. There are also some methods [2,3] that have used the Bayesian technique for its simple model and strong theoretical foundation. Moreover, Felzenszwalb et al. [29] proposed DPM, a popular part-based framework for object detection that is also widely used in vehicle classification.

As for traditional methods, it is necessary to search robust classifiers to compensate for the features' insufficiencies. However, with the development of CNNs, an increasing number of discriminative features have been mined. Therefore, light-weighted classifiers such as softmax and logistic regression have been favored by deep learning researchers [8,30].

2.3. Fine-grained vehicle type recognition

To our knowledge, there are only a few works that recognize vehicles at the fine-grained level. Krause et al. [31] proposed a 3D representation with spatial pyramid and locality-constrained linear coding [32]. However, it needed to construct a 3D model for every 2D image. Liao et al. [33] proposed a DPM-based method with

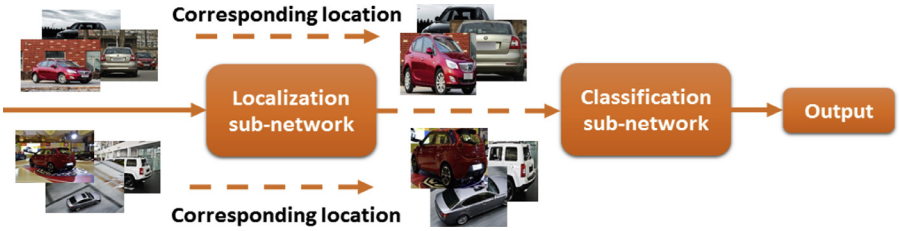

Fig. 1. Our two-stage deep learning framework, comprising two deep networks, that are used for vehicle localization and subcategory classification, respectively.

part-level supervision called SSDPM. Although this model was simple and designed specifically for vehicle classification, it required additional part-level labels, which are time-consuming to obtain. In [34], multi-class SVMs with designed loss function was used, but the classification ability was limited to three subcategory types. Obviously, deep learning has also been used in this domain. Sochor et al. [35] made full use of the geometric information of a vehicle, such as width, height and 3D mask. Although their CNN did improve the performance, it is tough to obtain this info without catching the camera parameter before. Zhang et al. [36] concentrated on embedding label structures. They used CNN to jointly optimize classification and similarity constraints which showed an obvious improvement in performance. Moreover, [8,11,30] are all deep learning methods that have been introduced before.

The aforementioned models may offer partial solutions to a challenging task, but many of them do not consider the following two problems. First, fine-grained vehicle images always contain similar backgrounds such as lawns, highway and parking lots. Unlike in category-level classification, these scene contexts do not facilitate the fine-grained task. The classifier would also not work well on images with sophisticated backgrounds or tiny vehicle targets, so localization is necessary. Second, they only consider the vehicle subcategory label, regardless of other cues such as viewpoint, which may be useful for sub-classification. This paper proposes a two-stage framework that integrates vehicle localization and sub-classification. Moreover, we jointly consider multi-cues in the classification stage.

3. Our method

To categorize vehicle types on a fine-grained level, we design a two-stage framework that comprises deep convolutional networks, as shown in Fig. 1. It is noteworthy that each deep convolutional network has integrated multiple tasks to collaboratively boost the performance in each stage.

The first network is designed to localize foreground vehicles, which can alleviate the influence of the images' backgrounds. It makes our method effective even when the background is cluttered or the vehicle only occupies a small ratio of the image. Two useful cues in a vehicle image help in designing the network. A typical cue is the vehicle's bounding box which indicates its location. Besides, considering the fact that the vehicle is usually salient in an image, we incorporate the visual saliency into the network. The collaboration between the two tasks in our CNN performs particularly well when handling many challenging samples.

The network in the second stage is designed to recognize types at the subordinate level. To achieve this goal, the network requires the ability to capture subtle differences across fine-grained types. Therefore, we introduce viewpoint cues to our CNN, as shown in Fig. 2.

Furthermore, we develop a multi-scale augmentation technique to alleviate the inadequacy of annotated fine-grained data. Our augmentation strategy is customized for vehicle-oriented tasks.

3.1. The localization network

In the first stage, the proposed network generates the coordinates of the bounding box. From the perspective of multi-task learning, our network incorporates both visual saliency and object detection cues. Considering that fine-grained training samples do not provide ground-true saliency maps, we borrow auxiliary data from other datasets to train the localization network. Fig. 3 depicts the pipeline of the localization network. Two loss functions guide the entire training procedure for localization. The first is a salient loss function and the second is the position loss function. Loss of saliency is expressed as Eq. (1). Here, N_{sal} is the number of input auxiliary salient images, M represents the number of pixels in an image, M_{neg} indicates the number of background pixels in salient maps, M_{pos} is the number of salient pixels, s_i denotes the predicted value of pixel i, and gs_i is the ground-true value. The motivation behind the designing of the loss function is the need to balance the influence of the salient and non-salient pixels.

$$Loss_{sal} = \frac{1}{N_{sal}} \sum_{i=1}^{N_{sal}} \left(\frac{M_{neg}}{M} \sum_{j_{pos}=1}^{M_{pos}} smooth_{L1}(s_{j_{pos}} - gs_{j_{pos}}) \right. \tag{1}$$

$$\left. + \frac{M_{pos}}{M} \sum_{j_{neg}=1}^{M_{neg}} smooth_{L1}(s_{j_{neg}} - gs_{j_{neg}}) \right) \tag{1}$$

in which

$$smooth_{L1}(x) = \begin{cases} 0.5x^2 & , if\,|x| < 1 \\ |x| - 0.5 & , otherwise \end{cases} \tag{2}$$

Eq. (3) shows the position loss function, where N_v is the number of input vehicle images. It calculates the smooth difference between the predict value p_j and the ground-true location value gp_j. Finally, Eq. (4) expresses the total loss of the localization network, where λ is a super parameter.

$$Loss_{pos} = \frac{1}{N_v} \sum_{i=1}^{N_v} \sum_{j \in l,t,r,b} smooth_{L1}(p_j - gp_j) \tag{3}$$

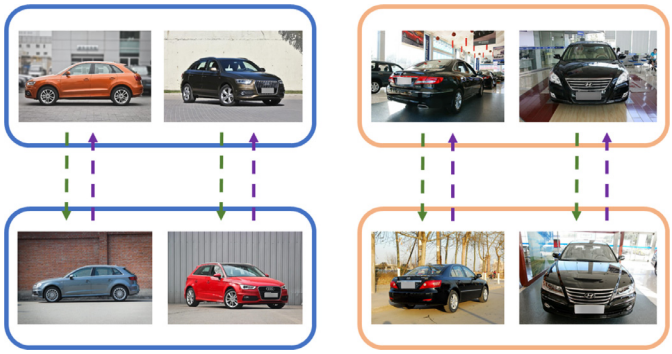

Fig. 2. Viewpoints play an important role in what the appearance exhibits. Among the four fine-grained types shown, without considering color, the vehicle pairs connected by dashed arrows share a similar appearance. Thus, the viewpoint is an important cue for vehicle representation.

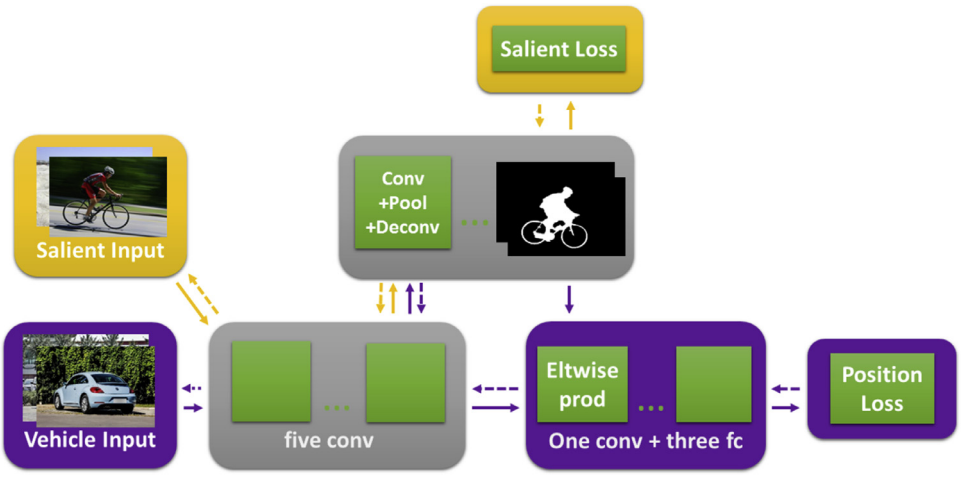

Fig. 3. Illustration of the localization network. We train for salient detection and vehicle localization simultaneously using the multi-task mode.

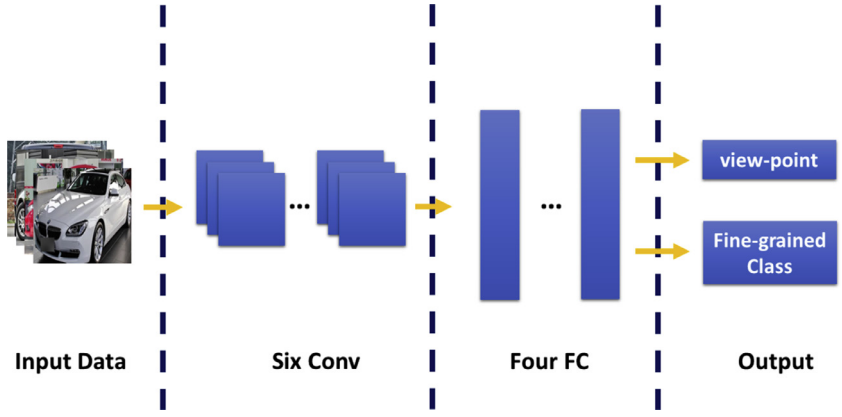

Fig. 4. The recognition network. Fine-grained recognition and viewpoint classification are jointly achieved through the multi-task learning.

$$Loss_{loc} = Loss_{sal} + \lambda Loss_{pos} \tag{4}$$

3.2. The recognition network

As Fig. 2 shows the same viewpoint creates a similar appearance, even among vehicles that belong to different types. This is mainly because vehicles are rigid objects with generally similar shapes. Thus, the network should be able to capture the viewpoint and subtle differences in appearance. Similar to the localization network, we also adopt a multi-task mechanism here. Fig. 4 demonstrates the pipeline, which comprises six convolutional layers and four fully connected (FC) layers. Of the FC layers, two are responsible for recognizing fine-grained types, while the remaining two depict viewpoints. The final objective function that guides the recognition stage is as follows.

$$\begin{aligned} Loss_{cls} &= Loss_t + \alpha Loss_v \\ &= \sum_{i=1}^{N}\sum_{c_t=1}^{C_t} \delta(y_t, c_t) Pr(y_t = c_t|X_i) \\ &\quad + \alpha \sum_{i=1}^{N}\sum_{c_v=1}^{C_v} \delta(y_v, c_v) Pr(y_v = c_v|X_i) \end{aligned} \tag{5}$$

where $Loss_t$ denotes the subcategory-level type prediction loss and $Loss_v$ indicates the viewpoint estimation loss. N depicts the number of input images, and c_t and c_v represent the fine-grained and viewpoint labels respectively. The final objective function in the recognition stage is the combination of two basic softmax with loss formulations.

Training our CNN is a tough task due to the constraint of an inadequate number of training samples with fine-grained labels. Therefore, data augmentation plays an important role in vehicle-centered applications. We propose a well-directed cropping scheme that merges three cropping methods. First, the image is not cropped, but rather scaled to a fixed size. Second, the foreground vehicle is cropped based on the given bounding box. Third, a multi-scale cropping method is used, that is, we make a slight shift along both the height and width on the basis of the original bounding box. To make the cropped patch meaningful, the range of shifting is restricted by [*low, high*]. During the training procedure, we use a weighted-selection approach to choose the cropping scheme for each image. The three weights are $w1$, $w2$, and $w3$. Our complete data augmentation strategy is described in Fig. 5.

4. Architecture

4.1. The localization network

In the localization network, we use an eltwise layer to combine salient detection and location regression. Both of the inputs for the two tasks go through a network with five convolutional layers, which are borrowed from the AlexNet [37]. After the convolutional layers, the salient branch is followed as shown in Fig. 6. We mainly use a tile layer and an eltwise layer to achieve the localization task. Finally, one convolutional layer and three fully connected layers are traversed as shown in Fig. 6.

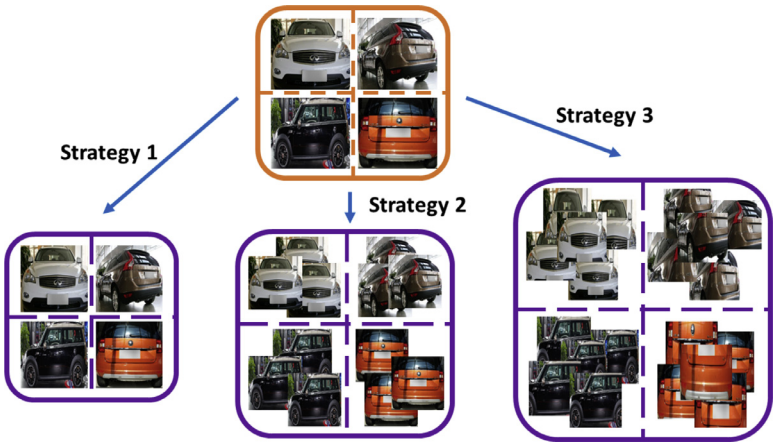

Fig. 5. Our data augmentation strategy.

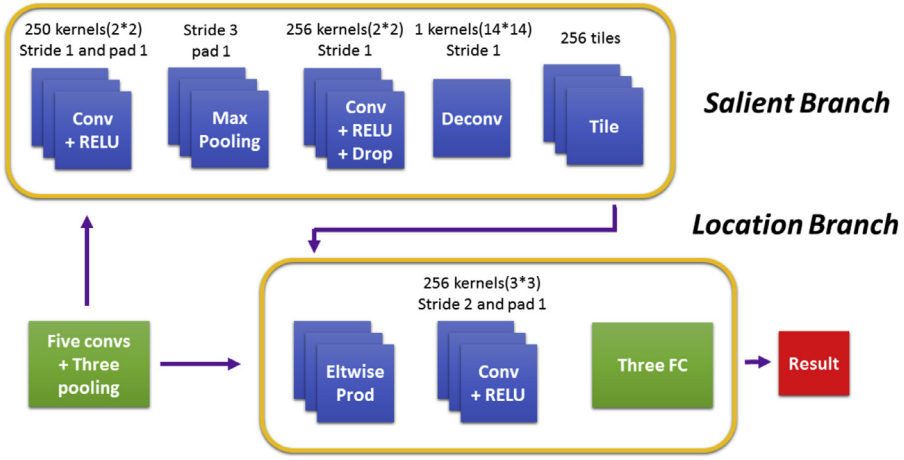

Fig. 6. The architecture of our localization network.

4.2. The recognition network

Increasing the depth of a network can improve the performance of a CNN in many situations. However, this strategy brings about a number of shortcomings, such as introducing more parameters to be learned, requiring more training data to obtain convergence, complicating tuning parameters, and so on. Given that we do not have enough data for the vehicle-centered task due to the huge manual annotation workload, we use data augmentation to obtain more training samples. We also try to decrease the number of parameters belonging to fully connected layers. Specifically, we add a convolutional layer to reduce the size of output feature maps, which are actually the input of the first fully connected layer. This can reduce the number of parameters in the first fully connected layer.

5. Experiment

In this section, we first introduce the datasets and our experimental settings. Then, the evaluation metrics and baseline methods are presented. Finally, we report the results on the benchmark datasets and show the analysis.

5.1. Datasets

We evaluate our approach using two datasets: CompCars and our newly released dataset named SYSU-Vehicle.

To our knowledge, CompCars [30] is currently the largest fine-grained vehicle dataset. It contains approximately 136,727 vehicle images from 1687 models and there is a single vehicle in each image. The authors provide the model label, maker label, viewpoint and the produced year for each vehicle in the image. Viewpoint is among the five labels including front, rear, side, front-side, and rear-side. Furthermore, there are two kinds of training/testing splitting provided by the authors. The first one is 50–50% splitting that results in 16,016 training and 14,939 testing images. The second splitting follows 70% versus 30%, which results in 36,456 training and 15,627 testing images. Both of the splitting schemes have 431 models and 75 makers, and these experiments are conducted based on both schemes.

SYSU-Vehicle, our new vehicle dataset, contains 5000 vehicle images, of which 4500 samples are used for training and the others for testing. In this dataset, images are divided into five classes: car, bus, truck, motorcycle and van. We provide the class label , the viewpoint (front, rear, side, front-side or rear-side), and bounding box annotation for each image. Fig. 7 demonstrates typical image samples from this dataset.

5.2. Settings

Regarding the localization network, we use auxiliary salient data to achieve the multi-task strategy. The MSRA dataset [38] is our primary salient data source, which contains 10,000 images and the corresponding salient masks. When we run the experiments, we manually discard the images that have more than one salient

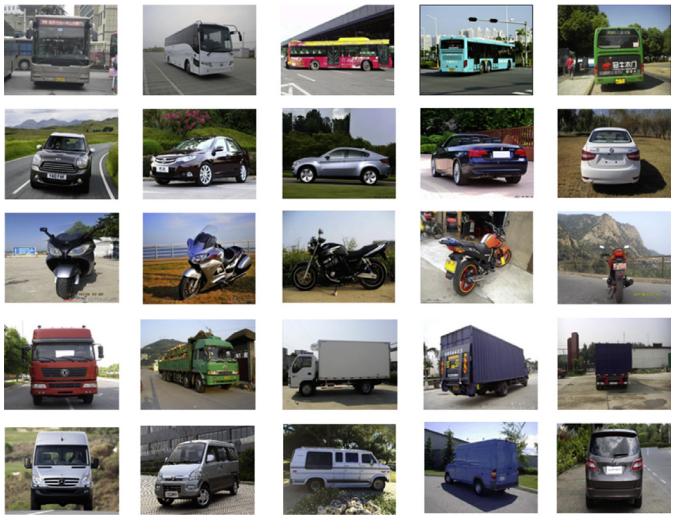

Fig. 7. Samples from SYSU-Vehicle dataset, collected from web images. It contains five classes: car, bus, truck, van and motor. For each class, we provide 180 images for training and 20 for testing per viewpoint. There are five viewpoints for each class. In the figure, each line of the image contains a single class with five viewpoints.

Table 1
Average overlap between the single-task and multi-task locations.

Method	Single-task	Multi-task
Average overlap (%)	90.89	**92.83**

Table 2
Recall under each overlap threshold. ST is the abbreviation for single-task and MT is the abbreviation for multi-task. Here, we only show the recall when the overlap threshold is between 0.7 and 0.9, as this interval is important for classification.

Threshold	0.7	0.75	0.8	0.85	0.9
ST recall (%)	99.26	98.77	97.50	92.06	68.53
MT recall (%)	99.48	99	97.84	95.26	86.17

Table 3
Fine-grained recognition accuracy for the AlexNet with and without our localization network. Loc ST/MT means that the position is obtained with our localization network, without/with our multi-task strategy.

	Top 1 (%)	Top 5 (%)	Makers (%)
Only AlexNet [37]	70.75	86.86	76.73
Loc ST+AlexNet	81.98	94.1	88.17
Loc MT+AlexNet	**83.23**	**94.29**	**89**

object, as our network only needs to localize a single target. There is one super parameter in this network, λ. We simply set it to 1 and keep it in all of the experiments. Additionally, all of the input images are normalized to a size of 400*400 pixels.

Before training or testing the recognition network in the second stage, all images should be fed into the localization network in the first stage to obtain the positions. Then, we crop out the target and resize it to 400*400 pixels, which is also the input size of the recognition network. Finally, we classify the preprocessed images. There are several parameters in the recognition network. The first is super parameter α. We also set it to 1 in all of the experiments. As to the weigths of the cropping method, we fix $w1$, $w2$, and $w3$ to 0.2, 0.4, and 0.4 respectively. Finally, the shifting range in multi-scale cropping method is set to [0.7, 1.0] without changed during the training process.

5.3. Evaluation metric

To evaluate the quality of our localization network, we use three evaluation metrics. The first is the average overlap between our location hypotheses and the ground truth. We follow the most widely accepted PASCAL protocol for the overlap calculation, as in Eq. (6), where g_i is the ground-true bounding box for the ith image and l_i is our prediction.

$$Overlap(g_i, l_i) = \frac{area(g_i) \cap area(l_i)}{area(g_i) \cup area(l_i)} \tag{6}$$

Second, the recall-overlap evaluation is used to concretely show the precision of our method. Finally, we compare the classification precision without localization, with single-task localization, and with multi-task localization to further demonstrate the necessity and improvement offered by our multi-task method. As for the classification metric, we simply follow the evaluation protocol of ImageNet challenge competition (ILSVRC) [39], which uses top 1 and top 5 accuracy.

5.4. Baselines

We compare our method with six approaches. First, the Bag-of-Words (BOW) [22] baseline is the most popular approach in image classification, and is also one of the representations of traditional methods. Second, Random Forest is a widely used ensemble classifier and has shown good performance in classification. We try to extract HOG feature and use Random Forest to solve the task. Third, AlexNet is a seminal method in deep learning. We fine-tune the pre-trained one on the fine-grained dataset. Fourth, the OverFeat [40] baseline, which is the CompCars dataset provided, has the same number of convolutional layers as our recognition network, but contains more convolutional kernels. It also integrates localization and classification like ours too. Fifth, Faster-RCNN is the state-of-the-art detection framework proposed by Ren et al. [41]. We use the code provided by the author and keep all of its parameters in training and testing. Sixth, The GoogleNet [42] baseline, which contains 22 layers, is the representation of deep and large networks.

5.5. Results and analysis

5.5.1. CompCars

This dataset provides two official splitting, with all of the experiments for both networks on 50–50% splitting. We compare our framework with state-of-the-art methods on both splitting.

We first evaluate our localization network. Table 1 shows that our multi-task method improves the precision of the predicted position by almost 2%. More concretely, Fig. 8 and Table 2 demonstrate that our multi-task method is useful for vehicle localization. It is obvious that our multi-task method improves the recall by more than 3% when the overlap threshold is set to 0.85, and this improvement ascends to near 20% if we set the threshold to 0.9.

Furthermore, Table 3 illustrates the classification impact given by the localization difference. Apparently, localization is helpful for vehicle fine-grained classification, improving it by approximately 11%. Our multi-task strategy also increases the performance by a minor margin. Fig. 9 shows some examples that are correctly classified by AlexNet with localization, but not by only AlexNet.

Next, we concentrate on the recognition network. Table 4 demonstrates the effects of our multi-task technique (1.5% improvement) and our multi-scale cropping strategy (1.4% improvement). To confirm the effects of our multi-task technique, we train the multi-task and single-task model without pre-trained on ImageNet. Table 5 summarizes these results. Obviously, the multi-task model achieves a much better result than the single-task one. Apparently, there is a huge difference in improvement between

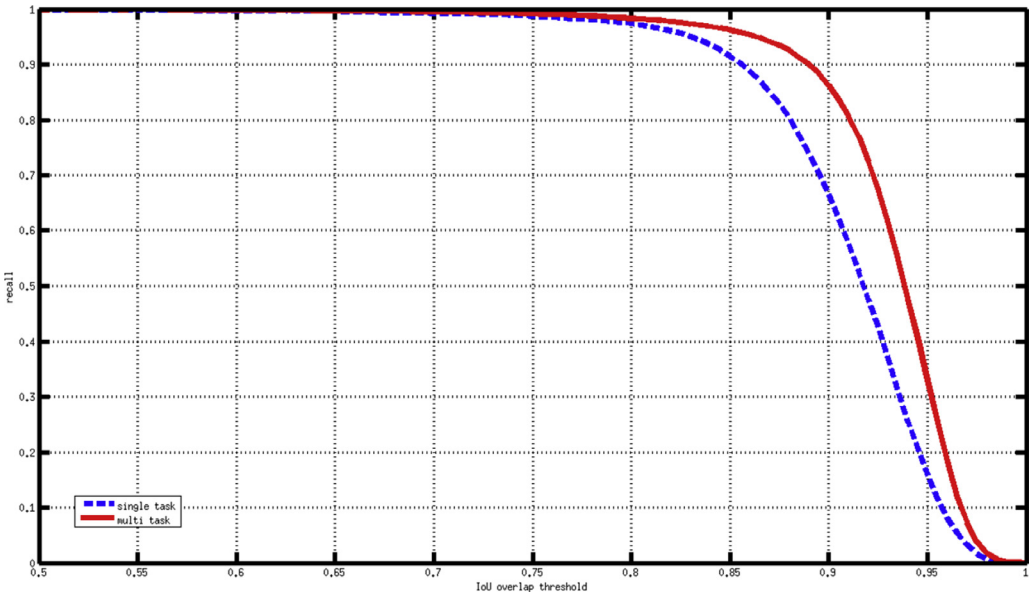

Fig. 8. Recall-overlap threshold curve. The blue dashed and red solid lines are the single-task and multi-task location results respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Fig. 9. Samples that are correctly classified by using our localization network, but mis-classified without localization.

Table 4
Comparison of our three methods. MS means the multi-scale cropping strategy is used and MT means the multi-task technique is used.

Method	Top 1 (%)	Top 5 (%)	Makers (%)
Ours+MT	89.6	97.55	94.12
Ours+MS	89.52	97.36	93.96
Ours+MS+MT	**91**	**97.77**	**95.14**

Tables 4 and 5. As ImageNet contains multi-view vehicles, the pre-trained model has already learned the viewpoint information, thus our multi-task strategy only helps the model learn better in a certain fine-tuned dataset. However, if we do not use the pre-trained model, some attributes of a vehicle can make the recognition network converge to a better result. And the viewpoint plays an important role among those attributes.

Finally, we compare our whole framework with other state-of-the-art methods in the dataset. Table 6 demonstrates the result for 50–50% official splitting, while Table 7 shows the comparison for

Table 5
Comparison between our single-task and multi-task classification models. NO_FT means the model is straightly trained in the fine-tuned dataset without pre-trained on ImageNet.

Method	Top 1 (%)	Top 5 (%)	Makers (%)
Ours+MS+ST (NO_FT)	57.02	75.77	63.17
Ours+MS+MT (NO_FT)	**77.19**	**91.44**	**84.38**

Table 6
Comparison of our method and other state-of-the-art methods with 50–50% splitting. Loc+AlexNet means first to get the location of the vehicle using our localization network, then using AlexNet for classification.

Method	Top 1 (%)	Top 5 (%)	Makers (%)
BOW [22]	8.07	–	14.61
HOG+RF	12.64	24.83	19.9
OverFeat [40]	76.7	91.7	82.9
Faster RCNN [41]	86.80	96.90	92.46
BoxCars [35]	84.8	95.4	–
Label tructure [36]	84.72	96.06	90.74
Loc+AlexNet	83.23	94.29	89
Ours+MS+MT	**91**	**97.77**	**95.14**

70–30% splitting. Both of the tables illustrate that our method outperforms all of the other methods, with our result reaching 90% accuracy. More concretely, as for 50–50% splitting, there is a 3% improvement between our method and Faster-RCNN [41], and a 13% promotion compared with OverFeat [40]. Especially, label structure [36] is a deep learning method which jointly optimizes classification and similarity constraints. Our classification accuracy is almost

Table 7
Comparison of our method and other state-of-the-art methods with 70–30% splitting.

	Top 1 (%)	Top 5 (%)
AlexNet [37]	81.9	90.4
OverFeat [40]	87.9	96.9
GoogleNet [42]	91.2	98.1
Ours+MS+MT	**94.3**	**98.9**

Table 8
Comparison of our method and other methods in the SYSU-Vehicle dataset. This table only shows the accuracy for each vehicle class, and – means that the corresponding vehicle class is ignored by the method. As our test data are uniform for every class, it is simple to calculate the average accuracy for every method. The average accuracy values for these methods are 69.5%, 79.5%, **93.8%**, 86.6%, 88.7%, and 90.8% from the top to the bottom of the table.

	Car (%)	Bus (%)	Motor (%)	Van(%)	Truck (%)
BOW [22]	67	66	81	–	64
HOG+RF	85	71	89	–	73
ICIG [10]	**93**	91	99	–	92
AlexNet [37]	91.4	**93.8**	97.7	62.6	87.4
Ours+MT	84.8	93.5	**99**	71.9	**94.4**
Ours+MS+MT	82.3	92.6	98.9	**88.1**	92

7% better than it. Moreover, our method also gets a better result than the very deep network, GoogleNet [42], on 70–30% splitting.

5.5.2. SYSU-Vehicle

In this dataset, the classification task is relatively simple, as it just category-level classification, so we do not use the localization network. Furthermore, to illustrate the performance of our multi-scale cropping strategy and show the robustness of our recognition network to the small dataset, we train our model on the dataset without pre-trained on ImageNet [39]. All of the results are summarized in Table 8. It can be observed that with the multi-scale cropping strategy, almost 2% improvement is made. Our method also does a better job of solving the classification task between cars and vans, which are very similar in appearance. There may be two reasons for this. First, our multi-scale cropping strategy solves the lack of training data in certain ways. Second, as our input images are resized to 400*400 pixels, which are larger than the input of AlexNet, our model can catch more discriminative information. It is necessary to illustrate that although the ICIG [10] method shows a state-of-the-art result in the dataset, it only classifies four classes, as do all the other traditional methods. That means there is no comparability between them and the deep learning methods.

6. Conclusion

We have presented an approach to vehicle type recognition at the fine-grained level. Multiple tasks are jointly optimized in two deep convolutional neural networks. This scheme can produce data-driven features in addiction to implicitly integrating complementary patterns into both stages. We also develop new vehicle-oriented data augmentation strategies in CNN training. Note that our approach is location-aware to handle the often-ignored vehicle localization, which is different from the weakly supervised image classification. The ability of localization is extremely useful in situations where vehicles occupy a small ratio of the whole image, or the background is cluttered. Furthermore, we release a new vehicle dataset named SYSU-Vehicle that contains semantic labels and bounding boxes, which is suitable for vehicle localization, recognition, viewpoint estimation, and so on. This dataset provides more cues than video-based datasets and can be used as an auxiliary dataset for vehicle classification on video sequence. Our future work will incorporate more structural constraints between vehicle parts to learn the data-driven high-level attributes of vehicles for classification.

Acknowledgment

This work was supported by the NSFC (U1611461, 61573387).

References

[1] B. Zhang, Reliable classification of vehicle types based on cascade classifier ensembles, IEEE Trans. Intell. Transp. Syst. 14 (1) (2013) 322–332, doi:10.1109/TITS.2012.2213814.
[2] X. Ma, W.E.L. Grimson, Edge-based rich representation for vehicle classification, in: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV 2005), Beijing, China, October 17-20, 2005, pp. 1185–1192, doi:10.1109/ICCV.2005.80.
[3] G. Pearce, N. Pears, Automatic make and model recognition from frontal images of cars, in: Proceedings of the Eighth IEEE International Conference on Advanced Video and Signal-Based Surveillance, (AVSS 2011), Klagenfurt, Austria, September 30, 2011, pp. 373–378, doi:10.1109/AVSS.2011.6027353.
[4] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110, doi:10.1023/B:VISI.0000029664.99615.94.
[5] A. Bosch, A. Zisserman, X. Muñoz, Representing shape with a spatial pyramid kernel, in: Proceedings of the Sixth ACM International Conference on Image and Video Retrieval, (CIVR 2007), Amsterdam, The Netherlands, July 9-11, 2007, pp. 401–408, doi:10.1145/1282280.1282340.
[6] A. Varghese, G. Sreelekha, Background subtraction for vehicle detection, in: Proceedings of the 2015 Global Conference on Communication Technologies (GCCT), Thuckalay, Kanya Kumari District, India, April 23-24, 2015, doi:10.1109/GCCT.2015.7342688.
[7] J. Hsieh, S. Yu, Y. Chen, W. Hu, Automatic traffic surveillance system for vehicle tracking and classification, IEEE Trans. Intell. Transp. Syst. 7 (2) (2006) 175–187, doi:10.1109/TITS.2006.874722.
[8] F. Zhang, X. Xu, Y. Qiao, Deep classification of vehicle makers and models: The effectiveness of pre-training and data enhancement, in: Proceedings of the 2015 IEEE International Conference on Robotics and Biomimetics, (ROBIO 2015), Zhuhai, China, December 6-9, 2015, pp. 231–236, doi:10.1109/ROBIO.2015.7418772.
[9] Z. Chen, T. Ellis, S.A. Velastin, Vehicle type categorization: a comparison of classification schemes, in: Proceedings of the IEEE Fourteenth International Conference on Intelligent Transportation Systems, (ITSC 2011), Hamburg, Germany, September 27-29, 2011, pp. 74–79, doi:10.1109/ITSC.2011.6083075.
[10] X. Hu, B. Hu, C. Guo, J. Lai, Fast unconstrained vehicle type recognition with dual-layer classification, in: Proceedings of the Eighth International Conference on Image and Graphics (ICIG 2015), August 13-16, 2015, pp. 275–283, doi:10.1007/978-3-319-21963-9.
[11] J. Krause, T. Gebru, J. Deng, L. Li, F. Li, Learning features and parts for fine-grained recognition, in: Proceedings of the Twenty-second International Conference on Pattern Recognition (ICPR 2014), Stockholm, Sweden, August 24-28, 2014, pp. 26–33, doi:10.1109/ICPR.2014.15.
[12] R.S. Feris, B. Siddiquie, Y. Zhai, J. Petterson, L.M. Brown, S. Pankanti, Attribute-based vehicle search in crowded surveillance videos, in: Proceedings of the First International Conference on Multimedia Retrieval (ICMR 2011), Trento, Italy, April 18-20, 2011, p. 18, doi:10.1145/1991996.1992014.
[13] B. Tian, B. Li, Y. Li, G. Xiong, F. Zhu, Taxi detection based on vehicle painting features for urban traffic scenes, in: Proceedings of the 2013 IEEE International Conference on Vehicular Electronics and Safety (ICVES 2013), Dongguan, China, July 28-30, 2013, pp. 105–109, doi:10.1109/ICVES.2013.6619612.
[14] B.C. Matei, H.S. Sawhney, S. Samarasekera, Vehicle tracking across nonoverlapping cameras using joint kinematic and appearance features, in: Proceedings of the Twenty-fourth IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2011), Colorado Springs, CO, USA, June 20-25, 2011, pp. 3465–3472, doi:10.1109/CVPR.2011.5995575.
[15] Y. Xiang, C. Song, R. Mottaghi, S. Savarese, Monocular multiview object tracking with 3D aspect parts, in: Proceedings of the Thirteenth European Conference Computer Vision (ECCV 2014), Zurich, Switzerland, September 6-12, 2014, pp. 220–235, doi:10.1007/978-3-319-10599-4.
[16] P.F. Felzenszwalb, R.B. Girshick, D.A. McAllester, Cascade object detection with deformable part models, in: Proceedings of the Twenty-third IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2010), San Francisco, CA, USA, June 13-18, 2010, pp. 2241–2248, doi:10.1109/CVPR.2010.5539906.
[17] K. He, L. Sigal, S. Sclaroff, Parameterizing object detectors in the continuous pose space, in: Proceedings of the Thirteenth European Conference Computer Vision (ECCV 2014), Zurich, Switzerland, September 6-12, 2014, pp. 450–465, doi:10.1007/978-3-319-10593-2.
[18] L. Yang, J. Liu, X. Tang, Object detection and viewpoint estimation with auto-masking neural network, in: Proceedings of the Thirteenth European Conference Computer Vision (ECCV 2014), Zurich, Switzerland, September 6-12, 2014, pp. 441–455, doi:10.1007/978-3-319-10578-9.
[19] K. Duan, L. Marchesotti, D.J. Crandall, Attribute-based vehicle recognition using viewpoint-aware multiple instance SVMs, in: Proceedings of the 2014 IEEE Winter Conference on Applications of Computer Vision (WACV 2014), Steamboat Springs, CO, USA, March 24-26, 2014, pp. 333–338, doi:10.1109/WACV.2014.6836081.
[20] Y.J. Lee, A.A. Efros, M. Hebert, Style-aware mid-level representation for discovering visual connections in space and time, in: Proceedings of the 2013 IEEE International Conference on Computer Vision (ICCV 2013), Sydney, Australia, December 1-8, 2013, pp. 1857–1864, doi:10.1109/ICCV.2013.233.
[21] Z. Dong, M. Pei, Y. He, T. Liu, Y. Dong, Y. Jia, Vehicle type classification using unsupervised convolutional neural network, in: Proceedings of the Twenty-second International Conference on Pattern Recognition (ICPR 2014), Stockholm, Sweden, August 24-28, 2014, pp. 172–177, doi:10.1109/ICPR.2014.39.

[22] F. Li, P. Perona, A Bayesian hierarchical model for learning natural scene categories, in: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), San Diego, CA, USA, June 20-26, 2005, pp. 524–531, doi:10.1109/CVPR.2005.16.
[23] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE Trans. Pattern Anal. Mach. Intell. 24 (5) (2002) 603–619, doi:10.1109/34.1000236.
[24] X. Wen, L. Shao, W. Fang, Y. Xue, Efficient feature selection and classification for vehicle detection, IEEE Trans. Circuits Syst. Video Technol. 25 (3) (2015) 508–517, doi:10.1109/TCSVT.2014.2358031.
[25] Z. Dong, Y. Wu, M. Pei, Y. Jia, Vehicle type classification using a semisupervised convolutional neural network, IEEE Trans. Intell. Transp. Syst. 16 (4) (2015) 2247–2256, doi:10.1109/TITS.2015.2402438.
[26] C. Kuo, R. Nevatia, Robust multi-view car detection using unsupervised subcategorization, in: Proceedings of the 2009 IEEE Workshop on Applications of Computer Vision (WACV 2009), Snowbird, UT, USA, December 7-8, 2009, pp. 1–8, doi:10.1109/WACV.2009.5403033.
[27] H. Fu, H. Ma, Y. Lin, D. Lu, A vehicleclassification system based on hierarchical multi-SVMs in crowded traffic scenes, Neurocomputing 211 (2016) 182–190, doi:10.1016/j.neucom.2015.12.134.
[28] G. Zhu, C. Yang, P. Zhang, in: Linear programming -nonparallel support vector machine and its application in vehicle recognition, Neurocomputing 215 (2016) 212–216, doi:10.1016/j.neucom.2015.07.159.
[29] P.F. Felzenszwalb, R.B. Girshick, D.A. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Trans. Pattern Anal. Mach. Intell. 32 (9) (2010) 1627–1645, doi:10.1109/TPAMI.2009.167.
[30] L. Yang, P. Luo, C.C. Loy, X. Tang, A large-scale car dataset for fine-grained categorization and verification, in: Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2015), Boston, MA, USA, June 7-12, 2015, pp. 3973–3981, doi:10.1109/CVPR.2015.7299023.
[31] J. Krause, M. Stark, J. Deng, L. Fei-Fei, 3D object representations for fine-grained categorization, in: Proceedings of the 2013 IEEE International Conference on Computer Vision Workshops, (ICCVW 2013), Darling Harbour, Sydney, Australia, December 3-6, 2013, pp. 554–561, doi:10.1109/ICCVW.2013.77.
[32] J. Wang, J. Yang, K. Yu, F. Lv, T.S. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: Proceedings of the Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2010), San Francisco, CA, USA, June 13-18, 2010, pp. 3360–3367, doi:10.1109/CVPR.2010.5540018.
[33] L. Liao, R. Hu, J. Xiao, Q. Wang, J. Xiao, J. Chen, Exploiting effects of parts in fine-grained categorization of vehicles, in: Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP 2015), Quebec City, QC, Canada, September 27-30, 2015, pp. 745–749, doi:10.1109/ICIP.2015.7350898.
[34] J. Zhan, H. Zhang, X. Luo, Fine-grained vehicle recognition via detection-classification-tracking in surveillance video, in: Proceedings of the International Conference on Digital Home (ICDH 2014), Guangzhou, China, November 28-30, 2014, pp. 14–19, doi:10.1109/ICDH.2014.10.
[35] J. Sochor, A. Herout, J. Havel, Boxcars: 3D boxes as CNN input for improved fine-grained vehicle recognition, in: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), Las Vegas, NV, USA, June 27-30, 2016, pp. 3006–3015, doi:10.1109/CVPR.2016.328.
[36] X. Zhang, F. Zhou, Y. Lin, S. Zhang, Embedding label structures for fine-grained feature representation, in: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), Las Vegas, NV, USA, June 27-30, 2016, pp. 1114–1123, doi:10.1109/CVPR.2016.126.
[37] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proceedings of the Twenty-sixth Annual Conference on Neural Information Processing Systems (NIPS 2012), Lake Tahoe, Nevada, United States, December 3-6, 2012, pp. 1106–1114. Advances in Neural Information Processing Systems 25.
[38] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, H. Shum, Learning to detect a salient object, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2) (2011) 353–367, doi:10.1109/TPAMI.2010.70.
[39] J. Deng, W. Dong, R. Socher, L. Li, K. Li, F. Li, Imagenet: a large-scale hierarchical image database, in: Proceedings of the 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), Miami, Florida, USA, June 20-25, 2009, pp. 248–255, doi:10.1109/CVPRW.2009.5206848.
[40] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, Y. LeCun, OverFeat: Integrated recognition, localization and detection using convolutional networks (2013) http://adsabs.harvard.edu/abs/2013arXiv1312.6229S.
[41] S. Ren, K. He, R.B. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, Proceedings of the 2015 Annual Conference on Neural Information Processing Systems (NIPS 2015), Montreal, Quebec, Canada, December 7-12, 2015, pp. 91–99. Advances in Neural Information Processing Systems 28, doi:10.1109/TPAMI.2016.2577031.
[42] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S.E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2015), Boston, MA, USA, June 7-12, 2015, pp. 1–9, doi:10.1109/CVPR.2015.7298594.
[43] S.-Z. Chen, C.-C. Guo, J.-H. Lai, Deep Ranking for Person Re-Identification via Joint Representation Learning, IEEE Trans. Image Process. 25 (5) (2016) 2353–2367, doi:10.1109/TIP.2016.2545929.

Bin Hu received the B.S. degree from the School of Information Science and Technology, Sun Yat-sen University, Guangzhou, China, in 2014, where he is currently pursuing the M.S. degree. His research focuses on computer vision and machine learning.

Jian-Huang Lai received his M.Sc. degree in applied mathematics in 1989 and his Ph.D. in mathematics in 1999 from SUN YAT-SEN University, China. He joined Sun Yat-sen University in 1989 as an Assistant Professor, where currently, he is a Professor in School of Data and Computer Science. His current research interests are in the areas of computer vision, pattern recognition and its applications. He has published over 250 scientific papers in the international journals and conferences on image processing and pattern recognition,e.g. IEEE TPAMI, IEEE TNN, IEEE TIP, IEEE TSMC (Part B), Pattern Recognition, ICCV, CVPR and ICDM. He serves as a deputy director of of the Image and Graphics Association of China and also serves as a standing director of the Image and Graphics Association of Guangdong. He is also the deputy director of Computer Vision Committee, China Computer Federation (CCF).

Chun-Chao Guo received the B.E. degree in communication engineering in 2010 from Lanzhou University and the Ph.D. degree in computer science in 2016 from Sun Yat-sen University, China. He is currently an algorithm engineer at Tencent. His research interests are in computer vision and pattern recognition, with a focus on human identity recognition, object tracking, object detection and visual surveillance. He is a recipient of the Excellent Paper Award at the 2014 National Conference on Image and Graphics. He won the first prize in the 2014 and 2015 National Graduate Contest on Smart-City Technology, and he was one of the winners in the 2014 Bocom Cup Contest on Video Analysis. He is a student member of CCF and IEEE.