This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Fast Automatic Vehicle Annotation for Urban Traffic Surveillance

Yi Zhou, Li Liu, Ling Shao, *Senior Member, IEEE*, and Matt Mellor

***Abstract*— Automatic vehicle detection and annotation for streaming video data with complex scenes is an interesting but challenging task for intelligent transportation systems. In this paper, we present a fast algorithm: detection and annotation for vehicles (DAVE), which effectively combines vehicle detection and attributes annotation into a unified framework. DAVE consists of two convolutional neural networks: a shallow fully convolutional fast vehicle proposal network (FVPN) for extracting all vehicles' positions, and a deep attributes learning network (ALN), which aims to verify each detection candidate and infer each vehicle's pose, color, and type information simultaneously. These two nets are jointly optimized so that abundant latent knowledge learned from the deep empirical ALN can be exploited to guide training the much simpler FVPN. Once the system is trained, DAVE can achieve efficient vehicle detection and attributes annotation for real-world traffic surveillance data, while the FVPN can be independently adopted as a real-time high-performance vehicle detector as well. We evaluate the DAVE on a new self-collected urban traffic surveillance data set and the public PASCAL VOC2007 car and LISA 2010 data sets, with consistent improvements over existing algorithms.**

***Index Terms*— Vehicle detection, attributes annotation, latent knowledge guidance, joint learning, deep networks.**

I. Introduction

INTELLIGENT traffic surveillance is being widely explored since the number of vehicles is ever-increasing and large-scale streaming video data become available. Among many traffic surveillance techniques, computer vision-based methods have attracted a great deal of attention and made significant contribution to practical applications such as vehicle counting, target vehicle retrieval, and behavior analysis. Particularly, efficient and accurate vehicle detection and attributes recognition are highly important components in these applications.

Vehicle detection is a fundamental problem in traffic surveillance. Vision-based approaches [1]–[5] can usually extract semantic visual features such as color, shape and texture. However, the challenge is that vision-based vehicle detectors always become unsteady caused by severe illumination and orientation variations, complicated backgrounds and occlusions. Therefore, vehicle detection by machine vision has been extensively investigated in recent years.

In urban traffic surveillance, a more interesting and valuable task is to extract diverse semantic information from detected vehicles, called vehicle attributes learning. Each vehicle on the road has its special attributes: travel direction (i.e. pose), inherent color, type and other more fine-grained information on headlight, grille and wheel. It would be extremely beneficial for identifying a target vehicle if its attributes could be annotated accurately. In general, traditional vehicle attributes recognition problems such as pose estimation, color recognition and type classification are usually treated separately [6]–[8]. However, separate independent analysis makes visual information utilized inefficiently. There exist strong correlations between these vehicle attributes learning tasks. For example, vehicle type classification based on visual appearance is highly dependent on the viewpoint. Therefore, we believe adopting multi-task learning can be helpful since jointly training implicitly learns the common features shared by correlated tasks. Moreover, a unified multi-attributes inference model can significantly improve the efficiency.

In this paper, we propose a fast framework DAVE, illustrated in Fig. 1, for vehicle detection and attributes annotation in urban traffic surveillance. As the deep convolutional neural network (CNN) has been widely and successfully applied to many vision tasks [9]–[15], we adopt its great advantages to build our models. The DAVE consists of two CNNs: fast vehicle proposal network (FVPN) and attributes learning network (ALN). The FVPN is a shallow fully convolutional network which aims to predict all the bounding-boxes of vehicles in real-time. The latter ALN configured with a very deep structure can precisely verify each detection and simultaneously infer pose, color and type information for positive vehicles. Although the ALN architecture is very deep, it can process 2 full high definition (FHD) resolution frames per second with GPU acceleration. Moreover, since highly descriptive features can be learned from the deep ALN, we adopt these features as latent data-driven knowledge to guide training the shallow FVPN so that the proposal network can even achieve competitive performance compared to other state-of-the-art detection frameworks. Furthermore, our DAVE also contributes to vehicle re-identification, to which little work has been devoted within the computer vision community.

In the experiments, we adopt the large-scale CompCars dataset [16] to train our models. We evaluate our models for vehicle detection on three datasets: a self-collected and manually-labeled Urban Traffic Surveillance (UTS) dataset

Manuscript received August 21, 2016; revised March 20, 2017; accepted August 11, 2017. The Associate Editor for this paper was N. Papanikolopoulos. *(Corresponding author: Ling Shao.)*

Y. Zhou, L. Liu, and L. Shao are with the School of Computing Sciences, University of East Anglia, Norwich NR4 7TJ, U.K. (e-mail: yizhou.szcn@gmail.com; liuli1213@gmail.com; ling.shao@ieee.org).

M. Mellor is with Createc, Cockermouth CA13 0HT, U.K. (e-mail: matt.mellor@createc.co.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2017.2740303

1524-9050 © 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

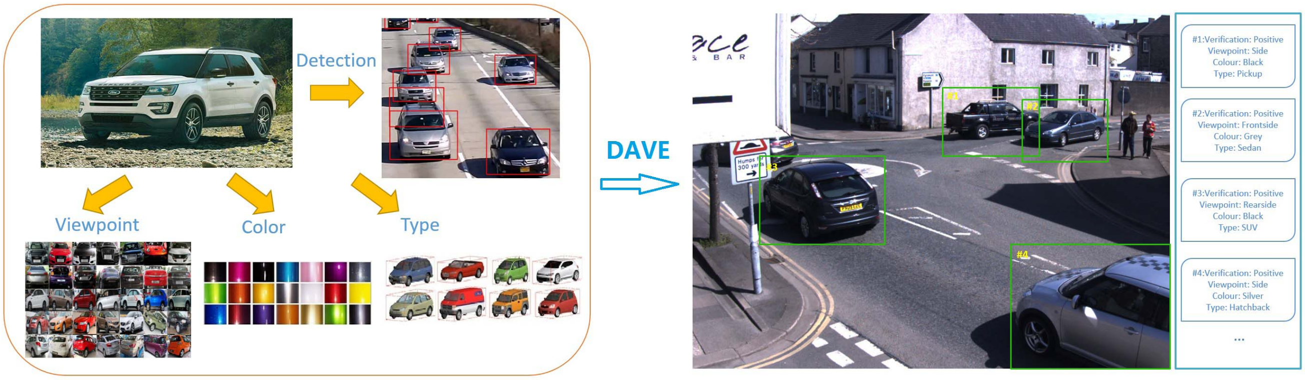


Fig. 1. **Illustration of DAVE**. A vehicle has many semantic attributes that can be applied to intelligent transportation systems, as shown in the left sub-figure. Given numerous surveillance videos, human labeling is expensive and time-consuming. The motivation of our proposed DAVE is to annotate the location, pose, type and color of all the vehicles on the raw videos automatically.

containing six 1920 × 1080 (FHD) resolution videos with different illumination and viewpoints, the PASCAL VOC2007 car dataset [17], and the LISA 2010 dataset [18]. Experimental results show our framework can efficiently detect various vehicles and outperforms four state-of-the-art detection methods: DPM [19], RCNN [20], Fast RCNN [21] and Faster RCNN [22]. Besides, evaluation of vehicle attributes annotation is carried out on the CompCars dataset and the UTS dataset.

This paper aims to unify multiple vehicle-related tasks into one vehicle annotation framework DAVE and makes three main contributions as follows.

- It proposes a deep vehicle Attributes Learning Network (ALN) to verify vehicles and annotate their pose, color and type simultaneously. Vehicle attributes can be further applied to vehicle re-identification tasks.
- It proposes a Fast Vehicle Proposal Network (FVPN) to predict all the vehicles' positions at real-time speed. Since the FVPN is trained together with the deeper ALN, latent data-driven knowledge learned from the ALN enables that the FVPN can be deployed as an independent, highly efficient vehicle detector.
- It introduces a new Urban Traffic Surveillance (UTS) vehicle dataset consisting of six 1920 × 1080 (FHD) resolution videos with different illumination and viewpoints.

Our preliminary work was presented in [23]. Compared to the previous work, in this paper, we first enhance the FVPN introduced in [23] to get better proposal performance by hard negative mining and carefully adjusting the training hyper-parameters such as weight decay and base learning rate. The current FVPN can be deployed as a successful individual vehicle detector without verification by the ALN. In addition, we explore the superiority and necessity of our method compared to the state-of-the-art one-net pipeline. Furthermore, through experiments, we prove our DAVE can contribute to the vehicle re-identification task which is hugely neglected by the current computer vision community.

II. Related Work

Vehicle-related systems usually adopt sensor-based approaches which are robust against illumination and viewpoint variations, to efficiently and stably detect and annotate vehicles. Sonar sensors [24] are configured in the front and rear of vehicles to help detection. Wireless magneto-resistive sensors [25] are adopted to test whether there are vehicles passing by. Strain gauge sensors are introduced in [26] to automatically classify vehicle types, and dead-reckoning/GPS sensors are exploited to estimate the pose of a driverless vehicle in [27].

Compared to the high costs of industrial grade sensors, computer vision methods only require low-cost cameras and attract increasingly more interests in intelligent surveillance applications in past decades. Most traditional vision-based vehicle detection works reviewed in [1] can be categorized into frame-based and motion-based approaches. For motion-based approaches, frames subtraction [28], adaptive background modeling [29] and optical flow [30] methods are often utilized, but the drawback is that less visual information is exploited so that any non-vehicle moving objects will be falsely detected. On the other side, conventional frame-based vehicle detection methods follow the sliding window fashion that is composed of appearance features extraction and classification. For instance, Histogram of Oriented Gradients (HOG) [31] and Haar-like features [32] are usually extracted, and SVMs [33], and AdaBoost [34] are adopted to discriminate whether each window with different scales and aspect ratios is a positive vehicle. The deformable part-based model (DPM) [19] successfully handles the detection of deformable objects but is not efficient due to the sliding window framework.

In recent years, with the great success of deep learning methods on image classification [9], Girshick et al. [20] proposed Region-based CNN which combines object proposal [35], [36], CNN learned features and SVM classifiers to perform detection. For increasing the detection speed and accuracy, Fast RCNN [21] adopts a region of interest (ROI) pooling layer and multi-task loss to estimate object classes while predicting bounding-box positions. Furthermore, Faster RCNN [22] employs initial layers with shared convolutional features to enable cost-free effective proposals. However, deep models deployed by these methods are designed for general object detection. Our work advances the idea of detection by focusing on one specific object: private motor vehicles, which

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

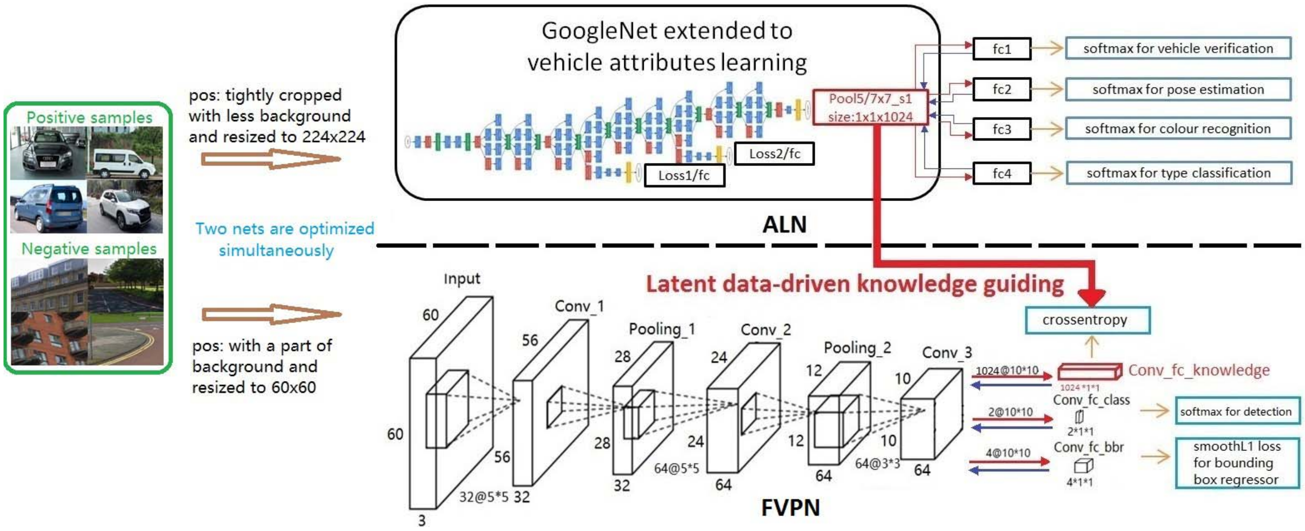

GoogleNet extended to vehicle attributes learning
Positive samples
Negative samples
pos: tightly cropped with less background and resized to 224x224
Two nets are optimized simultaneously
pos: with a part of background and resized to 60x60
Pool5/7x7_s1 size:1x1x1024
Loss1/fc
Loss2/fc
fc1
fc2
fc3
fc4
softmax for vehicle verification
softmax for pose estimation
softmax for colour recognition
softmax for type classification
ALN
Latent data-driven knowledge guiding
crossentropy
Input
60
60
3
Conv_1
56
56
32@5*5 32
Pooling_1
28
28
32
Conv_2
24
24
64@5*5
64
Pooling_2
12
12
64
Conv_3
10
10
64@3*3
64
1024@10*10
Conv_fc_knowledge
1024*1*1
Conv_fc_class
2@10*10
2*1*1
Conv_fc_bbr
4@10*10
4*1*1
softmax for detection
smoothL1 loss for bounding box regressor
FVPN

Fig. 2. **Training Architecture of DAVE.** The FVPN is a shallow fully convolutional network, which aims to precisely localize all the vehicles in real-time. The ALN is built by adding 4 fully connected layers to extend the deep GoogLeNet into a multi-attribute learning model. These two networks are simultaneously optimized in a joint manner by bridging them with latent data-driven knowledge guidance.

obtains effective performance with a very shallow and least-cost architecture. Detailed comparisons are illustrated in the experiments section.

Previous automatic vehicle annotation methods only focused on some single-purpose tasks such as color recognition [37] and coarse vehicle type classification [38]. Little work has been conducted for annotating different vehicle attributes simultaneously including pose, color and type information. We mainly review separate related work in this section. Lin et al. [6] presented an auto-masking neural network for vehicle detection and viewpoint estimation. In [7], an approach by vector matching of template was introduced for vehicle color recognition. In [8], an unsupervised convolutional neural network was designed for vehicle type classification from frontal view images. However, all these models were implemented on their own small datasets without any robust comparisons.

III. Detection and Annotation for Vehicle (DAVE)

In this paper, vehicle detection and annotation of pose, color and type are unified into one framework: DAVE. As illustrated in Fig. 2, DAVE consists of two convolutional neural networks called fast vehicle proposal network (FVPN) and attributes learning network (ALN), respectively. For training the models, FVPN and ALN are optimized together, while two-stage inference is performed in the test phase. FVPN aims to predict all the positions of vehicles in real-time. Afterwards, these vehicle candidates are passed to the ALN to simultaneously infer their corresponding pose, color and type, and verification is also operated to discard those false alarms by FVPN. Training our DAVE is inspired by Hinton [39] that knowledge learned from solid deep networks can be distilled to teach shallower networks. We design to apply latent data-driven knowledge from the deep ALN to guide training the shallow FVPN. This method is proved to be able to enhance the performance of the FVPN through experiments. The architecture of FVPN and ALN are described in the following subsections. More detailed training and inference methods are presented as well.

A. Fast Vehicle Proposal Network (FVPN)

Searching the whole image to classify whether each region is a vehicle in a sliding window fashion is prohibitive for real-time applications. Traditional object proposal methods are put forward to alleviate this problem, but thousands of proposals usually contain numerous false alarms and duplicate predictions which heavily lower the efficiency. Particularly for one specific object, we expect very fast and accurate detection performance can be achieved.

Our proposed fast vehicle proposal network (FVPN) is a shallow *fully convolutional network*, which aims to precisely localize all the vehicles in real-time. We are interested in exploring whether or not a small scale CNN is enough to handle the single object detection task. A schematic diagram of the FVPN is depicted in the bottom part of Fig. 2. The first convolutional layer (*conv_1*) filters the 60×60 resolution training images with 32 kernels of size 5×5. All the convolutional layers in FVPN are configured with stride parameter as 1 and padding as 0. The second convolutional layer (*conv_2*) takes as input the feature maps obtained from the previous layer and filters them with 64 kernels of size 5×5. Max pooling and Rectified Linear Units (ReLU) layers are configured after the first two convolutional layers. The third convolutional layer (*conv_3*) with 64 kernels of size 3×3 is branched into three sibling 1×1 convolutional layers transformed by traditional fully connected layers. In detail, *Conv_fc_class* outputs softmax probabilities of positive samples and the background; *Conv_fc_bbr* encodes bounding-box coordinates for each positive sample; *Conv_fc_knowledge* is configured for learning latent data-driven knowledge distilled from the ALN, which makes the FVPN be trained with more meticulous vehicle features. Inspired by [40], these 1×1 convolutional layers can successfully lead to differently purposed heatmaps

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

in the inference phase. This property can achieve real-time vehicle localization from whole images/frames by our FVPN.

We employ different loss supervision layers for three corresponding tasks in the FVPN. First, discrimination between a vehicle and the background is a simple binary classification problem. A softmax loss layer is applied to predict vehicle confidence, $p^c = \{p^c_{ve}, p^c_{bg}\}$. Besides, each bounding-box is encoded by 4 predictions: x, y, w and h. x and y denote the left-top coordinates of the vehicle position, while w and h represent the width and height of the vehicle size. We normalize all the 4 values relative to the image width and height so that they can be bounded between 0 and 1. Note that all bounding boxes' coordinates are set as *zero* for background samples. Following [21], a smooth L1 loss layer is used for bounding-box regression to output the refined coordinates vector, $loc = (\hat{x}, \hat{y}, \hat{w}, \hat{h})$. Finally, for guiding with latent data-driven knowledge of an N-dimensional vector distilled from a deeper net, the cross-entropy loss is employed for $p^{know} = \{p^{know}_0 \dots p^{know}_{N-1}\}$.

We adopt a multi-task loss L_{FVPN} on each training batch to jointly optimize binary classification of the vehicle against background, bounding-box regression and learning latent knowledge from a deeper net as the following function:

$$L_{FVPN}(loc, p^{bic}, p^{know}) = L_{bic}(p^{bic}) + \alpha L_{bbox}(loc) + \beta L_{know}(p^{know}), \quad (1)$$

where L_{bic} denotes the softmax loss for binary classification of vehicle and background. L_{bbox} indicates a smooth ℓ_1 loss defined in [21] as:

$$L_{bbox}(loc) = f_{L1}(loc - loc_t),$$
$$\text{s.t. } f_{L1}(x) = \begin{cases} 0.5x^2, & \text{if } |x| < 1 \\ |x| - 0.5, & \text{otherwise} \end{cases} \quad (2)$$

Furthermore, the cross entropy loss L_{know} is to guide the training of the FVPN by a latent N-dimensional feature vector t^{know} learned from a more solid net, which is defined as:

$$L_{know}(p^{know}) = -\frac{1}{N}\sum_i^N t_i^{know} \log p_i^{know} + (1 - t_i^{know}) \log(1 - p_i^{know}). \quad (3)$$

It is noteworthy that a bounding-box for the background is meaningless in the FVPN back-propagation phase and will cause training to diverge early [41], thus we set $\alpha = 0$ for background samples, otherwise $\alpha = 0.5$, whiles β remains at a fixed weighting value of 0.5.

B. Attributes Learning Network (ALN)

Attributes learning is also an interesting task [42]. Modeling vehicles' pose, color and type information separately is less accurate and inefficient. Actually, relationships between these tasks can be explored, so that designing a multi-task network is beneficial for learning shared features which can lead to extra performance gains. The attribute learning network (ALN) is a unified network to verify vehicle candidates and annotate their poses, colors and types. The network architecture of the ALN is mainly inspired by the GoogLeNet [10] model. Specifically, we design the ALN by adding 4 fully connected layers to extend the GoogLeNet into a multi-attribute learning model. The reason to adopt such a very deep structure here is because vehicle annotation belongs to fine-grained categorization problems and a deeper net has the more powerful capability to learn representative and discriminative features. Another advantage of the ALN is its high-efficiency inherited from the GoogLeNet which has lower computation and memory costs compared with other deep nets such as the VGGNet [11].

The ALN is a multi-task network optimized with four softmax loss layers for vehicle annotation tasks. Each training image has four labels in V, P, C and T. V determines whether a sample is a vehicle. If V is a true vehicle, the remaining three attributes P, C and T represent its pose, color and type respectively. However, if V is the background or a vehicle with a catch-all[1] type or color, P, C and T are set as *zero* denoting attributes are unavailable in the training phase. The first softmax loss layer $L_{verify}(p^V)$ for binary classification (vehicle vs. background) is the same as $L_{bic}(p^c)$ in the FVPN. The softmax loss $L_{pose}(p^P)$, $L_{color}(p^C)$ and $L_{type}(p^T)$ are optimized for pose estimation, color recognition and vehicle type classification respectively, where $p^P = \{p^P_1, \dots, p^P_{np}\}$, $p^C = \{p^C_1, \dots, p^C_{nc}\}$ and $p^T = \{p^T_1, \dots, p^T_{nt}\}$. $\{np, nc, nt\}$ indicate the number of vehicle poses, colors and types respectively. The whole loss function is defined as follows:

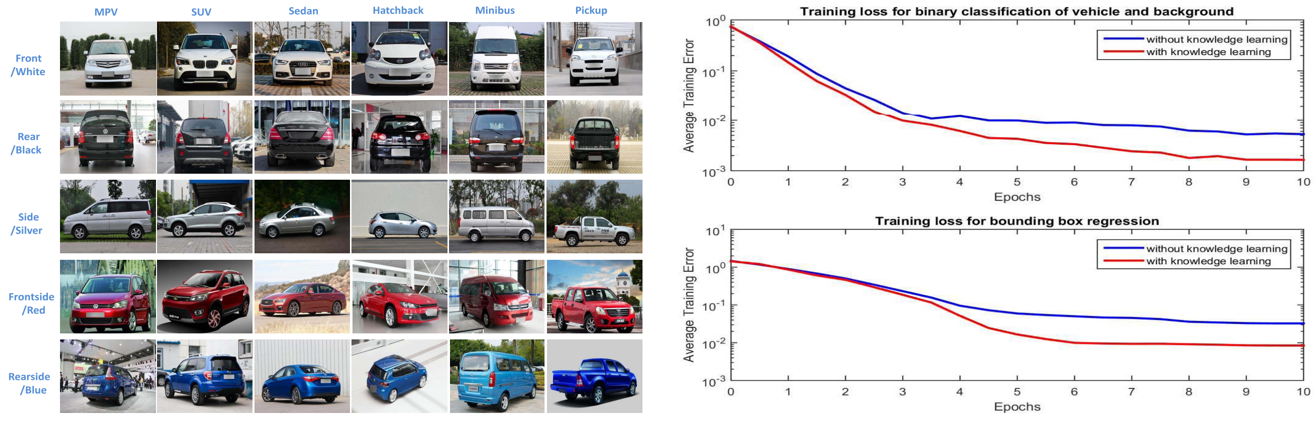
$$\begin{aligned} &L_{ALN}(p^V, p^P, p^C, p^T) \\ &\quad = L_{verify}(p^V) + \lambda_1 L_{pose}(p^P) \\ &\qquad + \lambda_2 L_{color}(p^C) + \lambda_3 L_{type}(p^T), \end{aligned} \quad (4)$$

where all the four sub loss functions are softmax loss for vehicle verification (***"verification" in this paper means confirming whether a detection is vehicle***), pose estimation, color recognition and type classification. Following the similar case of α in Eq. (1), parameters $\{\lambda_1, \lambda_2, \lambda_3\}$ all remain at a fixed weighting value of 1 for the positive samples, otherwise 0 for the background.

C. Deep Nets Training

1) Training Dataset and Data Augmentation: We adopt the large-scale CompCars dataset [16] with more than 100,000 web-nature data as the positive training samples which are annotated with tight bounding-boxes and rich vehicle attributes such as pose, type, make and model. In detail, the web-nature part of the CompCars dataset provides five viewpoints as *front*, *rear*, *side*, *frontside* and *rearside*, twelve vehicle types as *MPV, SUV, sedan, hatchback, minibus, pickup, fastback, estate, hardtop-convertible, sports, crossover* and *convertible*. To achieve an even training distribution, we discard less common vehicle types with few training images and finally select six types with all the five viewpoints illustrated in Fig. 3(a) to train our model. Besides, since color is another important vehicle attribute, we additionally annotated colors on more than 10,000 images with five common vehicle colors as *black, white, silver, red* and *blue* to train our final model.

[1] "Catch-all" indicates other undefined types and colors which are not included in our training model.



(a) (b)

Fig. 3. (a) Training data (columns indicate vehicle types, while rows indicate poses and colors), (b) Training loss with/without knowledge learning.

Apart from positive samples, about 150,000 negative samples (by hard negative mining) without any vehicles are cropped from Google Street View Images to compose our training data.

For data augmentation, we first triple the training data with increased and decreased image intensities for making our DAVE more robust under different lighting conditions. In addition, image downsampling up to 20% of the original size and image blurring are introduced to improve annotation precision and recall of detected vehicles that were small in scale within the image.

2) Jointly Training With Latent Knowledge Guidance: The entire training structure of DAVE is illustrated in Fig. 2. We optimize the FVPN and the ALN jointly but with different sized input training data at the same time. The input resolution of the ALN is 224 × 224 for fine-grained vehicle attributes learning, while it is decreased to 60 × 60 for the FVPN to fit smaller scales of the test image pyramid for efficiency in the inference phase. In fact, the resolution of 60 × 60 can well guarantee the coarse shape and texture of a vehicle is discriminative enough against the background. Besides, another significant difference between the ALN and the FVPN is that input vehicle samples for the ALN are tightly cropped, however, for the FVPN, uncropped vehicles are used for bounding-box (labeled as loc_t in Eq. (2)) regressor training.

The pre-trained GoogLeNet model for 1000-class ImageNet classification is used to initialize all the convolutional layers in the ALN, while the FVPN is trained from scratch. A 1024-dimensional feature vector of the *pool5/7×7_s1* layer in the ALN, which can exhaustively describe a vehicle, is extracted as the latent data-driven knowledge guidance to supervise the same dimensional *Conv_fc_knowledge* layer in the FVPN by cross entropy loss. We set the dimension of layer *Conv_fc_knowledge* in FVPN with the same value of 1024 correspondingly.

We first jointly train ALN and FVPN for about 10 epochs on the selected web-nature data that only contains pose and type attributes from the CompCars dataset. In the next 10 epochs, we fine-tune the models by a subset with our complementary color annotations. Throughout the training process, we set the batch size as 64, and the momentum and weight decay are configured as 0.9 and 0.0005, respectively. Learning rate is scheduled as 10^{-3} for the first 10 epochs and 5×10^{-4} for the second 10 epochs. To make our method more convincing, we train two models with and without knowledge guidance, respectively. During training, we definitely discover that knowledge guidance can indeed benefit training the shallow FVPN to obtain lower training losses. Training loss curves for the first 10 epochs are depicted in Fig. 3(b).

D. Two-Stage Deep Nets Inference

Once the joint training is finished, a two-stage scheme is implemented for inference of DAVE. First, the FVPN takes as input the 10-level test image Gaussian pyramid. For each level, the FVPN is operated over the input frame to infer *Conv_fc_class* and *Conv_fc_bbr* layers as corresponding heatmaps. All the 10 *Conv_fc_class* heatmaps are unified into one map by rescaling all the channels to the largest size among them and keeping the maximum along channels, while the index of each maximum within 10 channels is used to obtain four unified *Conv_fc_bbr* heatmaps (10 levels by similar rescaling). After unifying different levels *Conv_fc_class* heatmaps into the final vehicle detection score map, we first filter the score map with threshold $thres$ to discard low hot spots, and then local peaks on the map are detected by a circle scanner with tuneable radius r. In all our experiments, $r = 8$ and $thres = 0.5$ are fixed. Thus, these local maximal positions are considered as the central coordinates of proposals, $(\hat{x}_i\ ,\hat{y}_i)$. Coarse width and height of each detection can be simply predicted based on the bounding-box of its corresponding hot spot centered on each local peak. If one hot spot contains multiple peaks, the width and height will be shared by these peaks (i.e. proposals). For preserving the complete vehicle body, coarse width and height are multiplied by fixed parameter $m = 1.5$ to generate $(\hat{w}_i^{nobbr}\ ,\hat{h}_i^{nobbr})$. Thus, a preliminary bounding-box can be represented as $(\hat{x}_i,\hat{y}_i,\hat{w}_i^{nobbr}\ ,\hat{h}_i^{nobbr})$. Finally, bounding-box regression offset values (within [0,1]) are extracted from four unified heatmaps of *Conv_fc_bbr* at those coordinates $(\hat{x}_i,\hat{y}_i)$ to obtain the refined bounding-box.

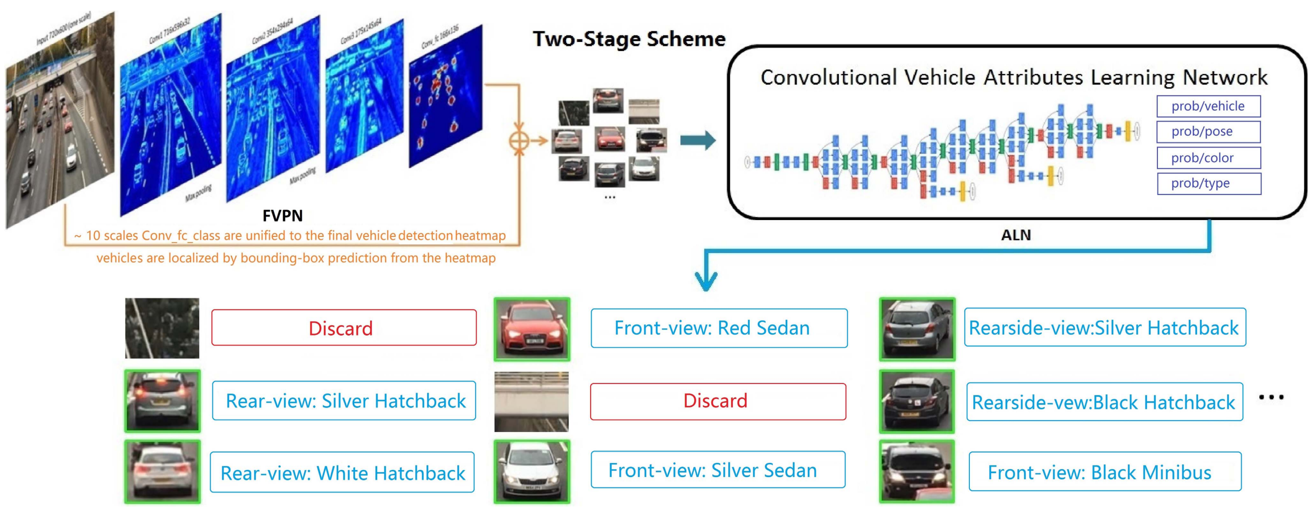

Input 720x600 (one scale)
Conv1 716x596x32
Conv2 354x294x64
Conv3 175x145x64
Conv_fc 168x136
Max pooling
Max pooling
FVPN
~ 10 scales Conv_fc_class are unified to the final vehicle detection heatmap
vehicles are localized by bounding-box prediction from the heatmap
Two-Stage Scheme
...
Convolutional Vehicle Attributes Learning Network
prob/vehicle
prob/pose
prob/color
prob/type
ALN
Discard
Front-view: Red Sedan
Rearside-view:Silver Hatchback
Rear-view: Silver Hatchback
Discard
Rearside-vew:Black Hatchback
...
Rear-view: White Hatchback
Front-view: Silver Sedan
Front-view: Black Minibus

Fig. 4. **A two-stage inference phase of DAVE.** Vehicle candidates are first obtained from FVPN in real-time. Afterwards, we use ALN to verify each detection and annotate each positive one with the vehicle pose, color and type.

Vehicle candidates inferred from the FVPN are taken as inputs into the ALN. Although verifying each detection and annotation of attributes are at the same stage, we assume that verification has a higher priority. For instance, in the inference phase, if a detection is predicted as a positive vehicle, it will then be annotated with a bounding-box and inferred pose, color and type. However, a detection predicted as the background will be neglected in spite of its inferred attributes. Finally, we perform non-maximum suppression as in RCNN [20] to eliminate duplicate detections. The full inference scheme is demonstrated in Fig. 4. At present, it is difficult to train a model that has the capability to annotate all the vehicles with enormously rich vehicle colors and types. During inference, a vehicle with untrained colors and types is always categorized into similar classes or a catch-all “others” class, which is a limitation of DAVE. In future work, we may expand our training data to include more abundant vehicle classes.

IV. Experiments and Results

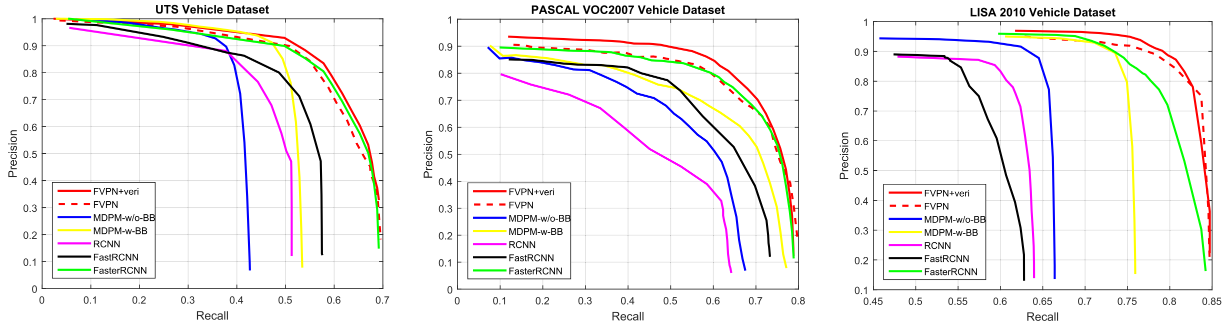
In this section, we evaluate our DAVE for detection and annotation of pose, color and type for each detected vehicle. Experiments are mainly divided into two parts: vehicle detection and attributes learning. In addition, we also explore the vehicle re-identification problem using the automatically annotated attributes. DAVE is implemented based on the deep learning framework Caffe [43] and run on a workstation configured with a NVIDIA TITAN X GPU.

A. Evaluation of Vehicle Detection

To evaluate vehicle detection, we train our models using the large-scale CompCars dataset as mentioned before, and test on three other vehicle datasets. We collect a full high definition (1920 × 1080) Urban Traffic Surveillance (UTS) vehicle dataset with six videos which were captured from different viewpoints and illumination conditions. Each video sequence contains 600 annotated frames. To be more convincing, we also compare our method on two other public datasets: the PASCAL VOC2007 car dataset [17] and the LISA 2010 dataset [18] with four competitive models: DPM [19], RCNN [20], Fast RCNN [21] and Faster RCNN [22]. These four methods obtain state-of-the-art performances on general object detection and the codes are publicly available. We adopted the trained car model from voc-release5 [44] for DPM, while the competitive NN models (VGG-16 based) were trained for this study using the CompCars dataset to implement vehicle detection. The vehicle detection evaluation criterion is the same as PASCAL object detection [17]. Intersection over Union (IoU) is set as 0.7 to assess correct localization.

1) Testing on the UTS Dataset: We not only test our real-time and highly accurate FVPN independently, but also verify that, by the deeper ALN (i.e., FVPN+verify in Fig. 5), the detection performance can be further improved, because some false alarms by the shallow FVPN will be discarded after the more rigorous ALN. The detection accuracy as average precision (AP) and speed as frames-per-second (FPS) are compared in the left column of Table I. Our model outperforms all the other methods with obvious improvements. Specifically, the shallow FVPN obtains an increased AP of 2.11% compared to the best model Faster RCNN, while the detection speed is significantly improved from 4 fps to 30 fps which can be termed as real-time. After verification by the deep ALN, 1.1% AP increase can be further achieved compared to FVPN, but the efficiency superiority is lost due to the deep architecture of ALN. However, the more complicated ALN is designed for fine-grained vehicle attributes annotation. Therefore, if we only consider implementing vehicle detection tasks, our proposed FVPN is preferred to be independently adopted as a high-performance vehicle detector.

The other two deep models, RCNN and Fast RCNN, do not produce satisfactory results mainly due to the low-precision proposals extracted by Selective Search [35]. Mixture-DPM with bounding-box prediction (MDPM-w-BB [19]) significantly improve the performance compared to MDPM-w/o-BB [19] by 10.77%. In addition, the speed of all these baselines is slower than 1 fps which is far from real-time vehicle detection.



(a)

(b)

(c)

Fig. 5. **Precision-recall curves on three vehicle datasets** FVPN+veri illustrates the detection results after verification by the ALN. MDPM-w/o-BB and MDPM-w-BB denote Mixture-DPM without / with bounding-box prediction, respectively.

TABLE I

VEHICLE DETECTION AP (%) AND SPEED (fps) COMPARISON ON THE UTS, PASCAL VOC2007 AND LISA 2010 DATASETS

Methods	**UTS**		**PASCAL VOC2007**		**LISA 2010**	
	Average Precision (AP)	**Processing Speed (fps)**	**Average Precision (AP)**	**Processing Speed (fps)**	**Average Precision (AP)**	**Processing Speed (fps)**
MDPM-w/o-BB	41.96%	0.25	48.44%	1.25	63.61%	0.7
MDPM-w-BB	52.73%	0.2	57.14%	1.25	72.89%	0.7
RCNN	44.87%	0.03	38.52%	0.08	55.37%	0.06
FastRCNN	51.58%	0.4	52.95%	0.5	53.37%	0.5
FasterRCNN	59.82%	**4**	63.47%	**6**	77.09%	**6**
FVPN-w/o-knowledge guide	55.73%	30	60.27%	46	73.88%	42
FVPN-w/o-bbr	57.04%	30	60.81%	46	73.46%	42
FVPN	61.93%	**30**	65.12%	**46**	80.46%	**42**
FVPN+Verify	**63.03%**	2	**66.44%**	4	**81.10%**	4

"bbr" indicates the bounding-box regressor used in our model, while "BB" denotes bounding-box prediction used in DPM model. "w" and "w/o" are the abbreviations of "with" and "without", respectively. "Verify" denotes the vehicle verification in the ALN.

We also test the FVPN trained without knowledge guidance, with the AP decreased by 6.20%, which proves the significant advantage of knowledge guidance. Moreover, if FVDN-w/o-bbr is adopted for simplifying the algorithm, most predicted bounding boxes will get some offsets or include more backgrounds, which makes detection unsatisfactory. Corresponding experiments are carried out to demonstrate that bounding-box regression can be helpful with the AP increased by 4.89%.

2) Testing on the PASCAL VOC2007 Car Dataset and the LISA 2010 Dataset: To make our methods more convincing, we also evaluate on two public datasets. All the images containing vehicles in the trainval and test sets (totally 1434 images) in the PASCAL VOC 2007 dataset are extracted to be evaluated. In addition, the LISA 2010 dataset contains three video sequences with low image quality captured by an on-board camera. All the results are shown in the middle and right columns of Table I. For the PASCAL VOC2007 dataset, the FVPN achieves 65.12% in AP with high-speed of 46 fps, which outperforms MDPM-w-BB, RCNN, FastRCNN and Faster RCNN by 7.98%, 26.6%, 12.17% and 1.65%, respectively. Likewise, FVPN+veri can even obtain a higher AP of 66.44%. Similarly, for the LISA 2010 dataset, the highest accuracy of 81.10% by FVPN+veri and 80.46% by only FVPN beats all other methods as well. Therefore, it demonstrates that our method is able to stably detect vehicles with different viewpoints, occlusions and varied image qualities.

Fig. 5 presents the precision-recall curves of all the compared methods on UTS, PASCAL VOC2007 car and the LISA 2010 datasets, respectively. From all these figures, we can further discover that, for all three datasets, both FVPN+Verify and only FVPN-based system achieve better performance than other vehicle detection methods by comparing Area Under the Curve (AUC). Besides, some qualitative detection results including successful and failure cases are shown in Fig. 6. It can be observed that the FVPN cannot handle highly occluded cases at very small sizes, since local peaks within the corresponding FVPN heat map overlap. The similar situation also exists in most of the deep networks based detection approaches [20]–[22], [41].

3) PASCAL VOC2007 Car Dataset Error Analysis: To further examine the detection results, we look into an in-depth error analysis on the VOC2007 car dataset. We categorize all the positive predictions into four different types with their corresponding IoU settings:

- Correct localization, $IoU >= 0.7$
- Wrong localization due to occlusions, $0.3 <= IoU < 0.7$
- Wrong localization due to others, $0.3 <= IoU < 0.7$
- False alarm due to backgrounds, $IoU < 0.3$

Fig. 7 demonstrates the error type analysis of false positives by a pie chart. We find that the severe occlusion between

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

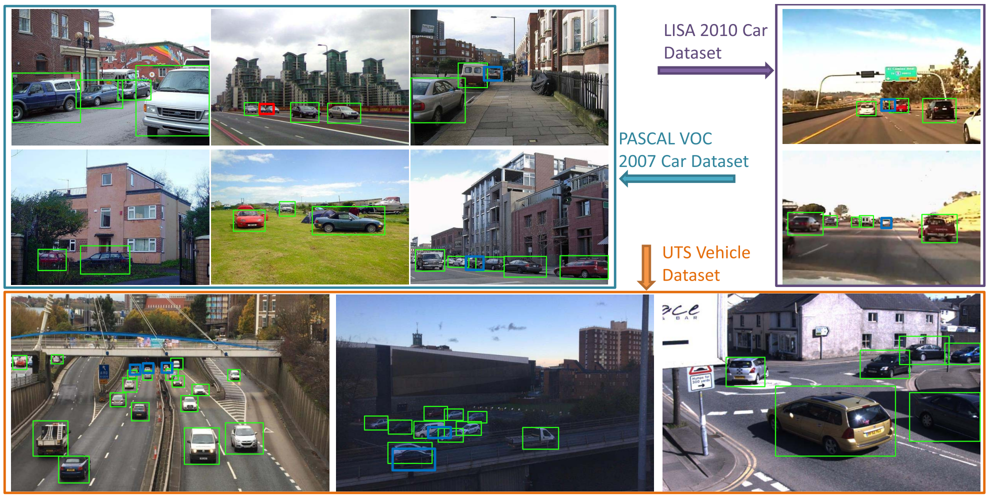


Fig. 6. **Examples of successful and failure cases for detection.** A green box denotes correct localization, a red box denotes false alarm and a blue box denotes missing detection.

TABLE II
EVALUATION (%) OF ATTRIBUTES ANNOTATION COMPARED TO ONE-NET PIPELINE ON THE UTS DATASET

Methods	Detection	Pose Estimation	12-type Classification	6-type Classification	Color Recognition
Faster RCNN VGG-16	59.82	91.30	58.19	88.43	69.44
DAVE	63.03	98.03	69.64	94.91	79.25

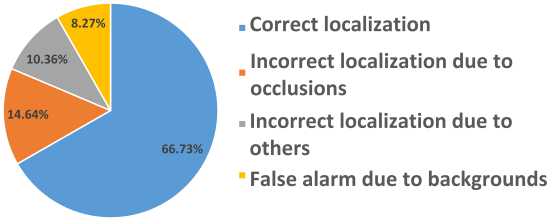


Fig. 7. **Error analysis for detection results on VOC2007 car dataset.** It shows the false positive detections are mainly due to the incorrect localization.

cars is the main factor for incorrect bounding-box prediction. Apart from false positives, the case of missing detection (i.e. false negatives) is always observed when the scale size is extremely small or a car body is occluded over 50%. Moreover, we can observe that vehicles in some dark colors cannot be detected under very low illumination. If we drop the data augmentation by adjusting image intensities for training, this situation will become more severe, decreasing the AP to 62.35%.

B. Evaluation of Vehicle Attributes Annotation

The experiments and analysis of the ALN are mainly based on the CompCars dataset and the UTS dataset. The web-nature data in the CompCars dataset are labeled with five viewpoints and twelve types about 136000 and 97500 images, respectively. We neglect those images without type annotation and randomly split the remaining data into the training and validation subsets as 7:3. In addition to pose estimation and type classification, we complement the annotation of five common vehicle colors on about 10000 images for evaluation of color recognition. Besides, for type classification, we compare the results of both the selected 6 common vehicle types and the total 12 types as mentioned in Section 3. 3. Vehicle verification (i.e., binary classification of vehicle and background) is evaluated in all the experiments as well.

In the following subsections, we first explain the superiority of our method compared to the state-of-the-art one-net pipeline. Then, we implement four different experiments to investigate the gain of the multi-task architecture, the accuracy by inputs with different image qualities, the effect of layer depths and the difficulty of fine-grained classification under different viewpoints.

1) Comparison to State-of-the-Art One-Net Pipeline: For investigating the necessity of our two-stage inference architecture for vehicle detection and attributes annotation, we compare it with the one-net pipeline FasterRCNN VGG-16 [22] trained with multi-attributes learning. We modify the last fully-connected layers of FasterRCNN to implement both bounding-box regression for detection and softmax for different attributes classification. Table II illustrates that the annotation accuracy of one-net FasterRCNN is much lower than that of our DAVE. The reasons are as follows. Vehicle attributes annotation is a fine-grained task, which requires relatively high-resolution vehicle images for better results, especially for vehicle type. Thus, the input frame of FVPN for test should be large (e.g. 1920 × 1080) to ensure all vehicle proposals inside are clear and informative to be fed into ALN for better annotation. However, one-net FasterRCNN has to take fixed size input (600 × 600) for initial layers (i.e. RPN), which leads to relatively small sized vehicle proposals (usually less than 100×100) that subsequently reduced the performance on later annotation due to lack of visual details. Although we can reconstruct FasterRCNN by uniformly amplifying the

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

TABLE III

EVALUATION (%) OF ATTRIBUTES ANNOTATION FOR VEHICLES ON THE UTS DATASET

Tasks	**Vehicle Verification**	**Pose Estimation**	**Vehicle Type Classification**		**Color Recognition**
			12 types	**6 types**	
Comparison of single-task learning (STL) and multi-task learning (MTL) for attributes prediction					
STL	98.73	96.94	60.37	88.32	78.33
MTL	**99.45**	98.03	69.64	94.91	**79.25**
STL feature+SVM	99.11	97.12	60.86	90.75	78.06
MTL feature+SVM	99.36	**98.10**	**69.86**	**95.12**	79.19
Comparison of Attributes prediction with different sizes of vehicle images					
28×28	90.45	83.49	37.52	53.66	49.73
56×56	98.12	91.33	52.02	77.02	66.14
112×112	99.37	96.56	63.41	90.67	**80.31**
224×224	**99.45**	**98.03**	**69.64**	**94.91**	79.25
Comparison of Attributes prediction with different deep models					
ALN based on FVPN ($depth = 4$)	95.96	81.21	27.26	43.12	65.12
ALN based on AlexNet ($depth = 8$)	**99.51**	95.76	66.01	89.25	77.90
ALN based on GoogLeNet ($depth = 22$)	99.45	**98.03**	**69.04**	**94.91**	**79.25**

TABLE IV

EVALUATION (%) OF FINE-GRAINED VEHICLE TYPE CLASSIFICATION ON THE UTS DATASET

Number of vehicle type	Front	Rear	Side	FrontSide	RearSide
12	58.02	60.37	66.73	61.28	64.90
6	79.42	84.60	92.93	86.77	92.53

spatial size of each layer to meet the requirement of resolution for good annotation, the computational and memory costs will dramatically increase and the detection speed will drop to 0.6 fps for FasterRCNN. Therefore, our two-stage inference architecture is necessary, and achieves significant advancement in real-world vehicle annotation tasks.

2) Single-Task Learning or Multi-Task Learning?: To explore this problem, we compare the multi-task ALN with the case of training networks for each attribute separately (i.e., single task). In addition, results by the combination of deep learned features and an SVM classifier are compared as well. All the model architectures are based on the GoogLeNet, and 1024-dimensional features are extracted from layer *pool5/7*$\times$7*_s1* to train the corresponding SVM classifier [45]. As shown in the top part of Table III, the multi-task model consistently achieves higher accuracies on four different tasks, which reveals the benefit of joint training. Although the combination of extracted features and SVM classifiers sometimes can lead to a small increase, we still prefer the proposed end-to-end model because of its elegance and efficiency.

3) How Small a Vehicle Size Can DAVE Annotate?: Since vehicles within surveillance video frames are usually in different sizes. Visual details of those vehicles far from the camera are significantly unclear. Although they can be selected by the FVPN with coarse requirements, after rescaling to 224×224, these vehicle proposals with low image clarity are hard to be annotated with correct attributes by the ALN. To explore this problem, we test vehicle images with original sizes of 224, 112, 56 and 28 using the trained ALN. The middle part of Table III illustrates that the higher resolution the original input size is, the better accuracy it can achieve.

4) Deep or Shallow?: How deep of the network is necessary for vehicle attributes learning is also worth to be explored. Since our ALN can be established on different deep models, we compare popular deep networks: AlexNet [9] and GoogLeNet with 8 layers and 22 layers, respectively. As VGGNet (16 layers version) [11] configured with numerous parameters requires heavy computation and large memory, we do not expect to employ it for our ALN. Besides, our proposed shallow FVPN with 4 layers is also used for attributes learning. From the bottom part of Table III, we can see that a deeper network does not obtain much better performance on vehicle verification compared to a shallow one. However, for pose estimation, type classification and color recognition, the deepest GoogLeNet consistently outperforms other nets with obvious gaps. Particularly for type classification which belongs to fine-grained categorization, the shallow FVPN gives extremely poor results. It illustrates that a deeper network with powerful discriminative capability is more suitable for fine-grained vehicle classification tasks.

5) Fine-Grained Categorization in Different Views: Finally, since vehicle type classification belongs to fine-grained categorization, we are interested in investigating its difficulty in different views due to its importance for our future work such as vehicle re-identification. As demonstrated in Table IV, for both 12-type and 6-type classification, higher precision is easier to be achieved from side and rearside views, while it is difficult to discriminate vehicle types from the front view. In other words, if we aim to re-identify a target vehicle from two different viewpoints, the type annotation predicted from a side view is more credible than that from a front view.

Fig. 8 shows some qualitative evaluation results of our DAVE on vehicle attributes annotation. It demonstrates that

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

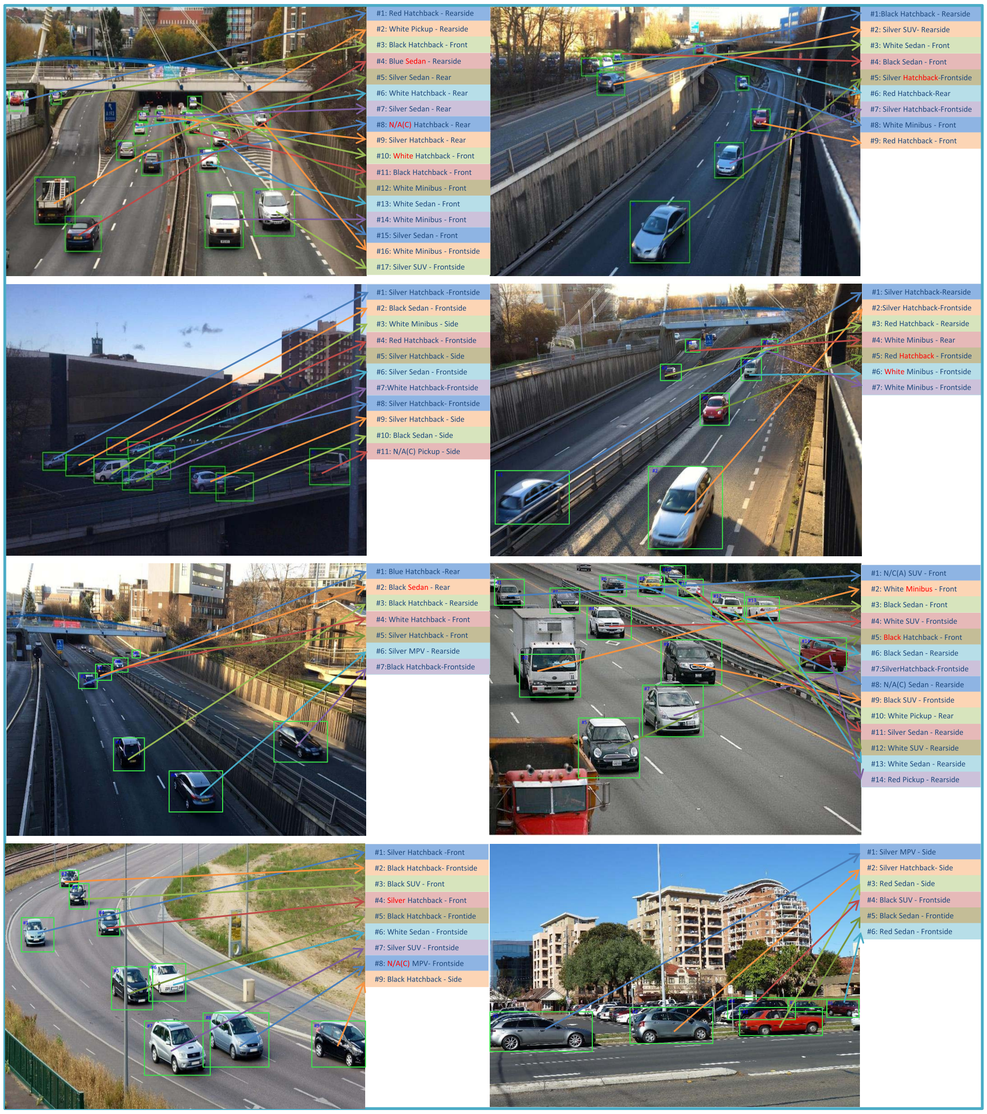

#1: Red Hatchback - Rearside
#2: White Pickup - Rearside
#3: Black Hatchback - Front
#4: Blue Sedan - Rearside
#5: Silver Sedan - Rear
#6: White Hatchback - Rear
#7: Silver Sedan - Rear
#8: N/A(C) Hatchback - Rear
#9: Silver Hatchback - Rear
#10: White Hatchback - Front
#11: Black Hatchback - Front
#12: White Minibus - Front
#13: White Sedan - Front
#14: White Minibus - Front
#15: Silver Sedan - Front
#16: White Minibus - Frontside
#17: Silver SUV - Frontside
#1:Black Hatchback - Rearside
#2: Silver SUV- Rearside
#3: White Sedan - Front
#4: Black Sedan - Front
#5: Silver Hatchback-Frontside
#6: Red Hatchback-Rear
#7: Silver Hatchback-Frontside
#8: White Minibus - Front
#9: Red Hatchback - Front
#1: Silver Hatchback -Frontside
#2: Black Sedan - Frontside
#3: White Minibus - Side
#4: Red Hatchback - Frontside
#5: Silver Hatchback - Side
#6: Silver Sedan - Frontside
#7:White Hatchback-Frontside
#8: Silver Hatchback- Frontside
#9: Silver Hatchback - Side
#10: Black Sedan - Side
#11: N/A(C) Pickup - Side
#1: Silver Hatchback-Rearside
#2:Silver Hatchback-Frontside
#3: Red Hatchback - Rearside
#4: White Minibus - Rear
#5: Red Hatchback - Frontside
#6: White Minibus - Frontside
#7: White Minibus - Frontside
#1: Blue Hatchback -Rear
#2: Black Sedan - Rear
#3: Black Hatchback - Rearside
#4: White Hatchback - Front
#5: Silver Hatchback - Front
#6: Silver MPV - Rearside
#7:Black Hatchback-Frontside
#1: N/C(A) SUV - Front
#2: White Minibus - Front
#3: Black Sedan - Front
#4: White SUV - Frontside
#5: Black Hatchback - Front
#6: Black Sedan - Rearside
#7:SilverHatchback-Frontside
#8: N/A(C) Sedan - Rearside
#9: Black SUV - Frontside
#10: White Pickup - Rear
#11: Silver Sedan - Rearside
#12: White SUV - Rearside
#13: White Sedan - Rearside
#14: Red Pickup - Rearside
#1: Silver Hatchback -Front
#2: Black Hatchback- Frontside
#3: Black SUV - Front
#4: Silver Hatchback - Front
#5: Black Hatchback - Frontide
#6: White Sedan - Frontside
#7: Silver SUV - Frontside
#8: N/A(C) MPV- Frontside
#9: Black Hatchback - Side
#1: Silver MPV - Side
#2: Silver Hatchback- Side
#3: Red Sedan - Side
#4: Black SUV - Frontside
#5: Black Sedan - Frontide
#6: Red Sedan - Frontside

Fig. 8. Qualitative results of attributes annotation. Red marks denote incorrect annotation, and N/A(C) means a catch-all color.

our model is robust to detect vehicles and annotate their poses, colors and types simultaneously for urban traffic surveillance. The failure cases mainly take place on incorrect colors and vehicle types.

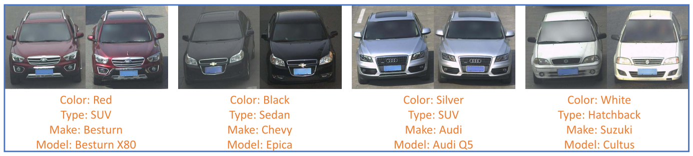


Fig. 9. Surveillance data pair samples for vehicle re-identification.

C. Evaluation of Vehicle Re-Identification

Compared to person re-identification [46]–[49], vision-based vehicle re-identification is a more challenging task, which is neglected by the computer vision community. Vehicle re-identification aims to identify a target vehicle in different cameras with disjoint views. In this experiment, we implement vehicle re-identification by exploiting semantic attributes learning. We adopt the attributes of vehicle type, color, make and model from the Compcars dataset to retrain the ALN with two extended softmax loss layers for learning make and model information, and then test on the surveillance-nature data. The surveillance data in Compcars consists of vehicles of 281 models in the front view. We select five positive pairs (i.e. same model and color) in each model to establish the ReID dataset (1405 IDs in total). Fig. 9 shows some positive pair samples and their corresponding attributes.

In the test phase, all the attributes can be inferred, and features of the *pool5/7×7_s1* layer are extracted for computing the Euclidean distances to get the final ranking of matching. Table V illustrates the classification results of each attribute and the final matching rate of top-k (k = 1, 5, 20 and 100) widely used in person re-identification. Fig. 10 also demonstrates some qualitative success and failure examples in the top-5 rank. We can observe that the color attribute is the most accurate factor for filtering candidates in the gallery

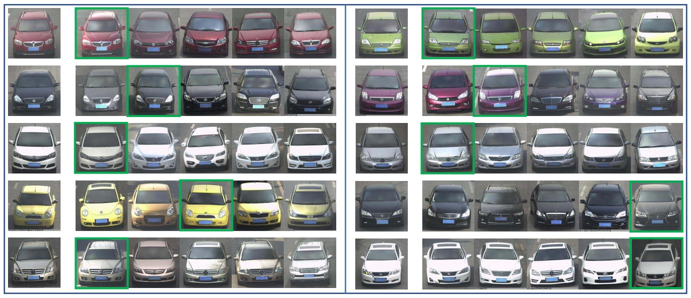


Fig. 10. **Qualitative results of top-5 rank by the ALN on the surveillance data.** The left column shows the query images, while the right column illustrates top-5 rank in the gallery set.

TABLE V

EVALUATION (%) OF ATTRIBUTES-BASED VEHICLE RE-IDENTIFICATION

Attributes	Type	Color	Make	Model
ALN	74.32	81.05	72.63	55.16
Human	61.54	96.82	87.43	-
Top-k	top-1	top-5	top-20	top-100
ALN	31.27	50.96	77.35	92.88
Human	86.25	100.00	-	-

set, although the illumination will make our model confused between similar colors such as black and gray, white and silver, and yellow and orange. Moreover, models with the same type and similar visual patterns also lead to failure cases.

To better understand the vehicle re-identification task, we introduce the human level performance for comparison. We invite 10 vehicle amateurs to annotate the type, color and make labels for the test images. Annotation on a large variety of models is hard to be included by non-experts. For re-identification, each human annotator is asked to re-identify 140 query vehicles and select 5 candidates in the gallery set by ranking as well. All the average results are compared in Table V. During the experiments, most people think the color is the easiest one to be identified, while the type is hard to be classified by the front view of a vehicle. The make recognition is mainly identified based on the logo, but failed in the low-resolution cases. Moreover, a very high accuracy of re-identification is achieved by humans since detailed visual textures can be carefully discriminated. However, for re-identifying one query vehicle, it averagely takes 220 seconds for a human to finish a ranking in the gallery set with the size of 1405. Although re-identification by humans is highly accurate, it is not acceptable in the real large-scale scenarios due to the poor efficiency. Therefore, the intelligent vehicle re-identification requires in-depth study.

V. CONCLUSION

In this paper, we developed a unified framework for fast vehicle detection and annotation: DAVE, which consists of two convolutional neural networks FVPN and ALN. The detection and attributes learning networks predict bounding-boxes for vehicles and infer their attributes: pose, color and type, simultaneously. Extensive experimental results have shown that our method outperforms state-of-the-art frameworks and achieves a highly accurate vehicle attributes annotation system. In addition, we also integrated more vehicle attributes such as make and model into the ALN, and exploited these attributes for vehicle re-identification tasks.

REFERENCES

[1] S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1773–1795, Dec. 2013.

[2] N. Buch, S. A. Velastin, and J. Orwell, "A review of computer vision techniques for the analysis of urban traffic," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 3, pp. 920–939, Sep. 2011.

[3] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 5, pp. 694–711, May 2006.

[4] D. Zhang, D. Meng, and J. Han, "Co-saliency detection via a self-paced multiple-instance learning framework," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 865–878, May 2017.

[5] X. Wen, L. Shao, W. Fang, and Y. Xue, "Efficient feature selection and classification for vehicle detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 3, pp. 508–517, Mar. 2015.

[6] L. Yang, J. Liu, and X. Tang, "Object detection and viewpoint estimation with auto-masking neural network," in *Proc. ECCV*, 2014, pp. 441–455.

[7] X. Li, G. Zhang, J. Fang, J. Wu, and Z. Cui, "Vehicle color recognition using vector matching of template," in *Proc. Int. Symp. Electron. Commerce Security*, Jul. 2010, pp. 189–193.

[8] Z. Dong, M. Pei, Y. He, T. Liu, Y. Dong, and Y. Jia, "Vehicle type classification using unsupervised convolutional neural network," in *Proc. Int. Conf. Pattern Recognit.*, Aug. 2014, pp. 172–177.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.

[10] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. CVPR*, Jun. 2015, pp. 1–9.

[11] K. Simonyan and A. Zisserman. (2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556

[12] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.

[13] D. Zhang, J. Han, C. Li, J. Wang, and X. Li, "Detection of co-salient objects by looking deep and wide," *Int. J. Comput. Vis.*, vol. 120, no. 2, pp. 215–232, 2016.

[14] D. Zhang, J. Han, J. Han, and L. Shao, "Cosaliency detection based on intrasaliency prior transfer and deep intersaliency mining," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 6, pp. 1163–1176, Jun. 2016.

[15] X. Yao, J. Han, D. Zhang, and F. Nie, "Revisiting co-saliency detection: A novel approach based on two-stage multi-view spectral rotation co-clustering," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3196–3209, Jul. 2017.

[16] L. Yang, P. Luo, C. Change Loy, and X. Tang, "A large-scale car dataset for fine-grained categorization and verification," in *Proc. CVPR*, Jun. 2015, pp. 3973–3981.

[17] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Sep. 2009.

[18] S. Sivaraman and M. M. Trivedi, "A general active-learning framework for on-road vehicle recognition and tracking," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 267–276, Jun. 2010.

[19] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

[20] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. CVPR*, Jun. 2014, pp. 580–587.

[21] R. Girshick, "Fast R-CNN," in *Proc. Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1440–1448.

[22] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. NIPS*, 2015, pp. 91–99.

[23] Y. Zhou, L. Liu, L. Shao, and M. Mellor, "DAVE: A unified framework for fast vehicle detection and annotation," in *Proc. ECCV*, Oct. 2016, pp. 278–293.

[24] S. Kim *et al.*, "Front and rear vehicle detection and tracking in the day and night times using vision and sonar sensor fusion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Aug. 2005, pp. 2173–2178.

[25] S. Wei, W. Jian, C. Bai-Gen, and Y. Qin, "A novel vehicle detection method based on wireless magneto-resistive sensor," in *Proc. 3rd Int. Symp. Intell. Inf. Technol. Appl.*, Nov. 2009, pp. 484–487.

[26] P. Shin *et al.*, "Automatic vehicle type classification using strain gauge sensors," in *Proc. 5th Annu. IEEE Int. Conf. Pervasive Comput. Commun. Workshops (PerCom Workshops)*, Mar. 2007, pp. 425–428.

[27] V. Malyavej, P. Torteeka, S. Wongkharn, and T. Wiangtong, "Pose estimation of unmanned ground vehicle based on dead-reckoning/GPS sensor fusion by unscented Kalman filter," in *Proc. 6th Int. Conf. Elect. Eng./Electron., Comput., Telecommun. Inf. Technol. (ECTI-CON)*, vol. 1. May 2009, pp. 395–398.

[28] K. Park, D. Lee, and Y. Park, "Video-based detection of street-parking violation," in *Proc. Int. Conf. Image Process., Comput. Vis., Pattern Recognit.*, Nov. 2007, pp. 152–156.

[29] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. CVPR*, Jun. 1999 pp. 246–252.

[30] E. Martínez, M. Díaz, J. Melenchón, J. A. Montero, I. Iriondo, and J. C. Socoró, "Driving assistance system based on the detection of head-on collisions," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2008, pp. 913–918.

[31] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1. Jun. 2005, pp. 886–893.

[32] S. Han, Y. Han, and H. Hahn, "Vehicle detection method using Haar-like feature on real time system," *World Acad. Sci., Eng. Technol.*, vol. 59, no. 35, pp. 455–459, 2009.

[33] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining Knowl. Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[34] Y. Freund, R. Schapire, and N. Abe, "A short introduction to boosting," *J. Jpn. Soc. Artif. Intell.*, vol. 14, nos. 771–780, p. 1612, 1999.

[35] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 154–171, Apr. 2013.

[36] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Proc. ECCV*, 2014, pp. 391–405.

[37] M. Yang, G. Han, X. Li, X. Zhu, and L. Li, "Vehicle color recognition using monocular camera," in *Proc. Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Nov. 2011, pp. 1–5.

[38] Z. Chen and T. Ellis, "Efficient annotation of video for vehicle type classification," in *Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2013, pp. 59–64.

[39] G. Hinton, O. Vinyals, and J. Dean. (2015). "Distilling the knowledge in a neural network." [Online]. Available: https://arxiv.org/abs/1503.02531

[40] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. CVPR*, Jun. 2015, pp. 3431–3440.

[41] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. CVPR*, Jun. 2016, pp. 779–788.

[42] L. Liu, Y. Zhou, and L. Shao, "DAP3D-NET: Where, what and how actions occur in videos?" in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May/Jun. 2017, pp. 138–145.

[43] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. ACM Multimedia*, 2014, pp. 675–678.

[44] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester. *Discriminatively Trained Deformable Part Models, Release 5*. [Online]. Available: http://people.cs.uchicago.edu/~rbg/latent-release5/

[45] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27-1–27-27, 2011. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm

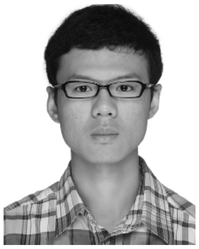
[46] S. Gong, M. Cristani, S. Yan, and C. C. Loy, *Person Re-Identification*, vol. 1. Springer, 2014.

[47] J. Chen, Y. Wang, J. Qin, L. Liu, and L. Shao, "Fast person re-identification via cross-camera semantic binary transformation," in *Proc. CVPR*, 2017.

[48] F. Zheng, Y. Tang, and L. Shao, "Hetero-manifold regularisation for cross-modal hashing," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published, doi: 10.1109/TPAMI.2016.2645565.

[49] F. Zheng and L. Shao, "Learning cross-view binary identities for fast person re-identification," in *Proc. IJCAI*, Jul. 2016, pp. 2399–2406.

Yi Zhou received the B.Eng. degree in electronic information engineering from Jiangsu University of Science and Technology, China, in 2012, and the M.Sc. degree in electronic and electrical engineering from University of Sheffield, U.K., in 2014. He is currently pursuing the Ph.D. degree with the School of Computing Sciences, University of East Anglia, Norwich, U.K. His research interests include computer vision and machine learning.



Li Liu received the B.Eng. degree in electronic information engineering from Xian Jiaotong University, Xian, China, in 2011, and the Ph.D. degree from the Department of Electronic and Electrical Engineering, University of Sheffield, Sheffield, U.K., in 2014. He is currently a Senior Research Associate with the School of Computing Sciences, University of East Anglia. His research interests include computer vision, machine learning, and data mining.

Ling Shao (M'09–SM'10) is currently a Professor with the School of Computing Sciences, University of East Anglia, Norwich, U.K. From 2005 to 2009, he was a Senior Scientist with Philips Research, The Netherlands. From 2009 to 2014, he was a Senior Lecturer with the University of Sheffield. From 2014 to 2016, he was a Professor with North Umbria University. His research interests include computer vision, image/video processing and machine learning. He is an Associate Editor of IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS and several other journals. He is a Fellow of the British Computer Society and the Institution of Engineering and Technology.

Matt Mellor received the Ph.D. degree in medical image analysis from University of Oxford. He is currently the Director of Createc, Cockermouth, U.K. His current research interests include computer vision, radiation imaging, and 3-D imaging.