Deep Reinforcement Learning With Visual Attention for Vehicle Classification

Dongbin Zhao, *Senior Member, IEEE*, Yaran Chen, and Le Lv

***Abstract*—Automatic vehicle classification is crucial to intelligent transportation system, especially for vehicle-tracking by police. Due to the complex lighting and image capture conditions, image-based vehicle classification in real-world environments is still a challenging task and the performance is far from being satisfactory. However, owing to the mechanism of visual attention, the human vision system shows remarkable capability compared with the computer vision system, especially in distinguishing nuances processing. Inspired by this mechanism, we propose a convolutional neural network (CNN) model of visual attention for image classification. A visual attention-based image processing module is used to highlight one part of an image and weaken the others, generating a focused image. Then the focused image is input into the CNN to be classified. According to the classification probability distribution, we compute the information entropy to guide a reinforcement learning agent to achieve a better policy for image classification to select the key parts of an image. Systematic experiments on a surveillance-nature dataset which contains images captured by surveillance cameras in the front view, demonstrate that the proposed model is more competitive than the large-scale CNN in vehicle classification tasks.**

***Index Terms*—Convolutional neural network (CNN), reinforcement learning, vehicle classification, visual attention.**

I. Introduction

IN MODERN transportation systems, the images of vehicles on the road are available with low cost due to the popularity of cameras. Therefore an automatic vehicle recognition and classification system based on images captured by cameras is a highly practical and essential technique, which has been applied to intelligent transportation systems, including traffic measurement, traffic management, retrieving, and public security [1]. For example, vehicle classification can be exploited to inexpensively automatic and expedite paying tolls from the lanes, based on different rates for different types of vehicles. Vehicle classification also can help police to track suspected vehicles with fake license plates, by finding the same vehicle makers and models with the suspected vehicles. However, vehicle classification in real-world environments is still a challenging task, specially under complex lightings and video surveillance, the performance is far from being satisfactory. In this paper, we focus on the automatic vehicle classification based on images captured by traffic cameras.

Automatic vehicle recognition can be naturally formulated as an image classification problem. Given an image of a vehicle, usually captured by a traffic camera, a typical vehicle classification system uses a model to determine the category. Image classification has been extensively studied for decades. Many impressive methods have been proposed and achieved great success. Traditional image classification systems use a shallow classification model, such as Bayesian [2], support vector machine [3]–[5], random forest [6], and boosting [7], [8], to classify an image based on the features extracted from the whole image, such as scale-invariant feature transform [9], histogram of oriented gradients [10], and local binary patterns features. On one hand, these methods heavily rely on hand-designed features. On the other hand, shallow models which are trained by original training data sometimes have limited capability in representation learning [11]. Recently, convolutional neural network (CNN) is wide applied to image classification which makes a huge breakthrough. Some methods even exceed the human-level performance on several tasks, such as image classification and face recognition [12]–[16]. Compared to traditional image classification methods, CNN can automatically learn the feature representation associated with classification target from the raw image. Although CNN has achieved great success in image classification, CNN still models the overall image and the different parts of an image are treated without distinctions, which could cause that CNN has limited capability in capturing and highlighting the key areas of an image for the classification. However, in the fine-grained classification, it is essential for the classification to highlight the nuances in the key areas.

In comparison to typical image classification, vehicle classification has its unique characteristics. There are large quantities of vehicle models, among which a lot of similarities exist. It is difficult to distinguish them with an overall view, especially for fine-grained vehicle classification, which is more challenging. Meantime, vehicles present many unique properties, such as unique vehicle parts, including headlights,

Manuscript received April 6, 2016; revised July 3, 2016 and August 11, 2016; accepted September 24, 2016. Date of publication September 30, 2016; date of current version December 7, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 61273136, Grant 61573353, and Grant 61533017, and in part by the National Key Research and Development Plan under Grant 2016YFB0101000.

D. Zhao is with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: dongbin.zhao@ia.ac.cn).

Y. Chen and L. Lv are with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: chenyaran2013@ia.ac.cn; iamlvle@126.com).

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/TCDS.2016.2614675

doors, wheels, and logos. These parts yield slight differences in appearance among different models. Therefore, exploiting these local differences and highlighting the key parts of a vehicle image will probably improve the classification performance, especially for the challenging fine-grained vehicle classification, which needs to classify more kinds of vehicle models.

Owing to the multiglimpse and visual attention mechanism, human vision system shows remarkable capability in exploiting these local differences and highlighting the key parts of an image [17]. Visual attention means that a person only focuses attention on one area when looking at an image and multiglimpse refers that human can quickly scan the whole image and find multiple key areas during the image recognition process [18]. In this process, the internal relation of different parts would be built up, to guide eye movement for finding the next area where to look. Focusing on the relevant parts of the image and ignoring the irrelevant parts make it easier to learn in the presence of clutter. Therefore, human vision can achieve incomparable performance in object recognition. Many researches have been studied in the combination of the visual attention and computer vision system. For example, cognition robotics with human visual attention have been studying in order to perceive the world and act up on it in a human-like fashion [19], [20].

Another advantage of multiglimpse and visual attention mechanism is robustness. If our eyes stare at an object in real-world scene, we will ignore the parts away from the stared object. Therefore, the background away from the stared object does not interfere with us to recognize the object. However, most current image classification methods, including CNN-based image classification, equally treat every part of an image. Therefore, the redundant parts and the irrelevant parts probably confuse the classification, resulting that the classification is sensitive to the visual environments, including the background, the viewpoint of the camera and the lighting condition.

The multiglimpse and visual attention mechanism helps human to execute action, plan, reason, and make a decision, which is essentially a sequential decision processing [17] and can be well solved by reinforcement learning for its powerful decision-making ability [21]. Inspired by the mechanism, we propose a novel CNN model based on visual attention for vehicle classification (denoted as "CNNVA"). The model uses reinforcement learning algorithm to exploit visual attention which can capture key areas of a vehicle image through multiple guided searches and visual attention. In detail, we first input a vehicle image into an attention module to produce a focused image with the key part clear and others fuzzy. Then the focused image is input into a CNN classifier to predict its label. According to the classification probability distribution, we compute the information entropy of the focused image, which can judge the discrimination of the focused image and the classification confidence. The information entropy can be used as the reward of reinforcement learning algorithm to guide the agent to learn good policies of selecting another key area. After several steps, the agent can almost find all key areas that are helpful to the vehicle classification. Finally, incorporating all key parts into an image, we can recognize the vehicle.

We summarize our contributions as follows.

1) We propose a novel method to integrate multiglimpse and visual attention mechanism into CNN using reinforcement learning algorithm, which can find the key areas of an image to help vehicle classification.
2) We use deep reinforcement learning to select the key area automatically, for finding fewer and useful key areas which are enough for the classification task.
3) We combine the information entropy and reinforcement learning to find the next viewpoint. The information entropy of the classification probability distribution is considered as metrics to judge the discrimination of the focused image and the classification confidence, which is used as a reward to guide the reinforcement learning.

The rest of this paper is organized as follows: the related work is presented in Section II. In Section III, we give an overview of the architecture of the proposed CNNVA. Sections IV–VI describe the CNNVA model, visual attention module, evaluation network and viewpoint selection module, respectively, of CNNVA. The classification and optimization are presented in Section VII. Section VIII presents the experiments and results. Finally, we conclude this paper in Section IX.

II. Related Work

CNN has been extensively studied for decades in the computer vision community, and many practical methods have been proposed. It was first proposed by LeCun *et al.* [22] and applied to the handwritten digit recognition with a small scale. Due to the limited computation capability, CNN has been only applied in small-scale image classification, such as Mixed National Institute of Standards and Technology database [23], CIFAR10/100 [24], and NYU object recognition benchmark [25]. With the improvement of computation capability, especially the large-scale application of graphics processing unit, deep CNN with millions of parameters can be effectively optimized using the large-scale data, and achieve remarkable performance in some tasks, such as Imagenet [11] and scene labeling [26], [27].

CNN has powerful ability to representation learning, which can automatically learn the feature representation associated with training objective from the raw data. Through the visualization technology, Zeiler and Fergus [28] showed that some convolutional layers of CNNs can extract features that are similar to hand-designed features. In fact, each convolutional map of the convolutional layer performs like a filter designed by human. Moreover, owing to multiple nonlinear processing layers (convolutional and pooling layers), CNN can extract abstract features. Generally speaking, low-level features are parts-based representations, such as corners, lines, and edges, which usually have nothing to do with the specific classification task and are shared by all kinds of categories. And high-level features are more global and more invariant, which are usually related to a specific classification task. Although deep architectures show prominent superiorities compared

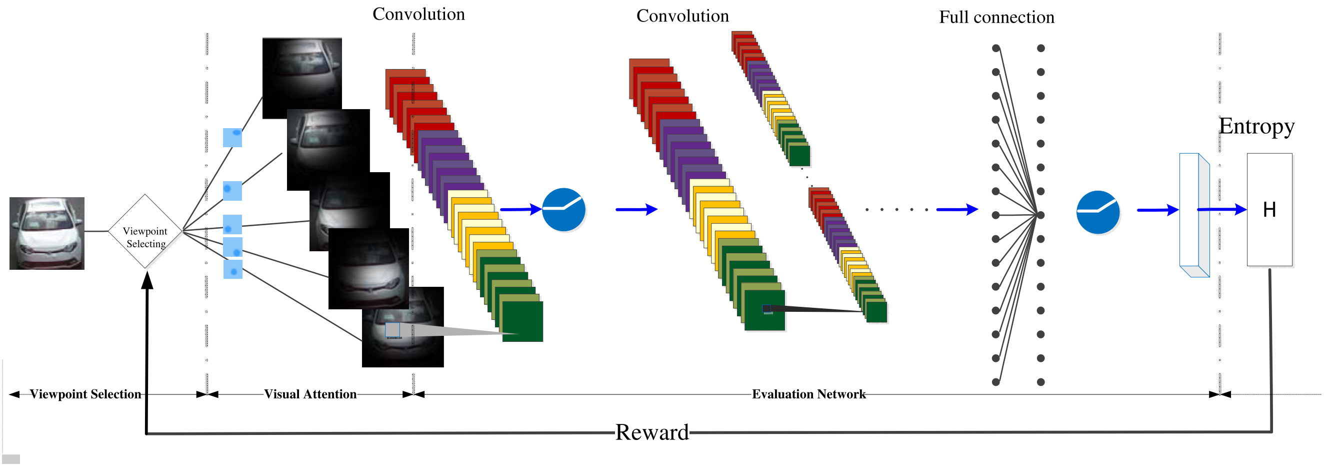

Fig. 1. Architecture of the proposed CNNVA.

to shallow architecture, especially in the challenging task, it is difficult to train a deep model with a large number of parameters. Fortunately, a set of good initialization parameters can reduce the difficulty of learning. Therefore, we can initialize our CNN model using some low layer parameters of the CNN model trained on other data in advance, such as Oxford VGG (trained on Imagenet) [29], which probably improves the performance of vehicle classification.

When we are looking at something, the viewpoint of our eyes usually focuses on one part of the picture at one glimpse, and through the multiple glimpses, the viewpoint will pay attention to the different parts along with the movement of eyes. Integrating these glimpses, we understand the picture completely. This process is the vision mechanism of multiglimpse and visual attention, which is extensively studied for decades in cognitive science and neuroscience community. Treisman and Gelade [30] explored the factors that influence people to search for an object in a simple set of stimuli. Itti *et al.* [31] proposed a saliency-driven focal visual attention model. Underwood found that bottom-up visual characteristics and the visual saliency play an important role in guiding eye movement [32]. Foulsham and Underwood [33] studied how a human moves eyes to fast find the interested object when they first see an image.

Some of these research findings have been applied to the fields of computer vision. Lukic *et al.* [34] developed a humanoid robot with visual attention by using attentional landscape for driving visual attention. Yücel *et al.* [35] presented a task-independent gaze-fixation, cognitively-inspired and object segmentation mechanism for robotic joint attention. Xu *et al.* [36] applied visual attention to recognize the content of an image. Recently, visual attention is used in image classification for some simple tasks. Ranzato [37] proposed a method to learn where to look sequentially in images through many glimpses and applied this method to the handwriting recognition. The method used a neural network consisting of simple fully connected layers mainly, to select a location. Google DeepMind proposed a recurrent model of visual attention [21]. This paper used a recurrent neural network to select where to look next. These methods did not take spatial location into account and do not use the convolutional layers to extract image features. Xiao *et al.* [38] used deep CNN to realize a two-level attention model for fine-grained image classification. While this paper did not use reinforcement learning to choose the viewpoint. In this paper, we propose a novel CNNVA-based and use reinforcement learning to guide where to look. Compared with this paper [38], our model can find fewer key areas due to reinforcement learning.

III. CNNVA Architecture

In this section, we present an overview of the CNNVA architecture, consisting of a visual attention module, an evaluation network and a viewpoint selection module, as shown in Fig. 1. The visual attention module highlights one part of the image and weakens other parts. It can transform a raw image into a focused image with one part clear and others fuzzy through a mapping function. The evaluation network computes the probabilistic prediction of the classification of the focused image. In this paper, we take CNN as the evaluation network. Its probabilistic outputs can be used to judge the discrimination of the focused image and the classification confidence. According to the classification probability distribution, the viewpoint selection module can select another attention area.

Given an image $\mathbf{X}$, we randomly select an attention viewpoint from the input image, and the visual attention module transforms $\mathbf{X}$ into a focused image $\mathbf{X}_f$ according to the selected viewpoint. Then the evaluation network predicts the label of the focused image $\mathbf{X}_f$. According to the probabilistic outputs of evaluation network, the viewpoint selection module selects a new attention viewpoint to produce another focused image for $\mathbf{X}$. Repeating this process several times, we can almost find all key areas that are helpful to the classification. In fact, viewpoint selection can be regarded as a sequential decision processing, as shown in Algorithm 1. After several repeats, the key areas of an image that are helpful to the classification are almost found. Finally, incorporating the key areas, the image can be recognized, as shown in Fig. 2. The viewpoint selection algorithm in Algorithm 1 aims to select a new viewpoint through the reward of the current viewpoint.

Algorithm 1 CNNVA Architecture

```
for episode=1,M do
    for each image do
        initialize a viewpoint $(u, v)$;
        repeat
            generate a focused image with $(u, v)$ and compute the reward $r$
            execute the viewpoint selection algorithm to generate a new viewpoint
                $(u, v) \leftarrow (u', v')$
        until reach the goal
    end for
end for
```

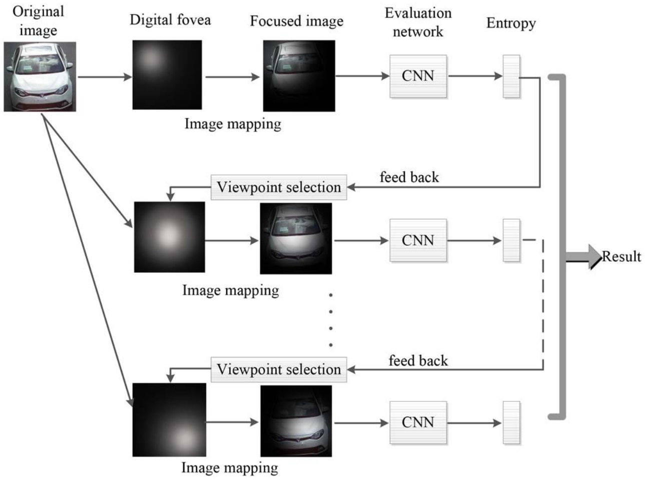

Fig. 2. Flow chart of the proposed CNNVA algorithm.

IV. Visual Attention Module

A human eye has a foveate structure called "fovea centralis" which has higher resolution than other portions [18]. At a glimpse, the viewpoint of our eyes usually focuses on one point of the picture. Due to the fovea centralis of human eyes, the focused part has higher resolution than the parts away from the focused point. In order to mimic this characteristic of human vision system, we construct an attention weight matrix (AWM) to map a raw image into the focused image with one part clear and others fuzzy, as the following:

$$\mathbf{X}_f = f_d(\mathbf{X}, \mathbf{\Phi}) = \mathbf{\Phi} \odot \mathbf{X} \tag{1}$$

where $\mathbf{X} \in \mathbb{R}_+^{M\times N}$ and $\mathbf{X}_f \in \mathbb{R}_+^{M\times N}$ are the pixel matrices of the raw image and the focused image, respectively, and M and N are the height and the width of the image, respectively. $\mathbf{\Phi} \in \mathbb{R}_{[0,1]}^{M\times N}$ is the AWM and $\odot$ denotes an element-wise multiplication. By analogy with the human fovea centralis, we call AWM "digital fovea" and the mapping function is denoted as "digital fovea operator."

The AWM $\mathbf{\Phi}$ are associated with the given focused point (u, v) and the distance r between (u, v) and another point (i, j) in the image. Its element ϕ_{ij} is regarded as an attention rate on the pixel of the image at the location (i, j), and can be computed using a sigmoid function, as follows:

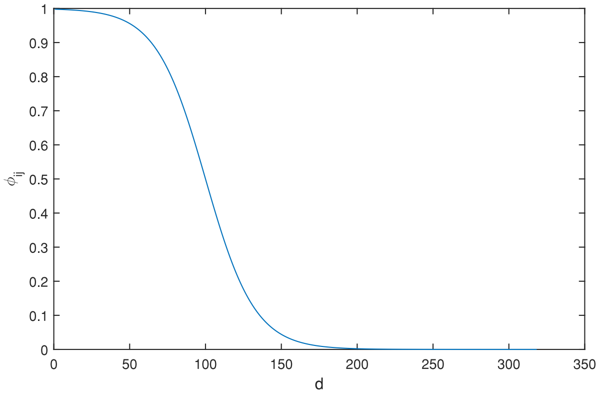

Fig. 3. Mapping function: the attention weight ϕ_{ij} decreases by increasing the distance d.

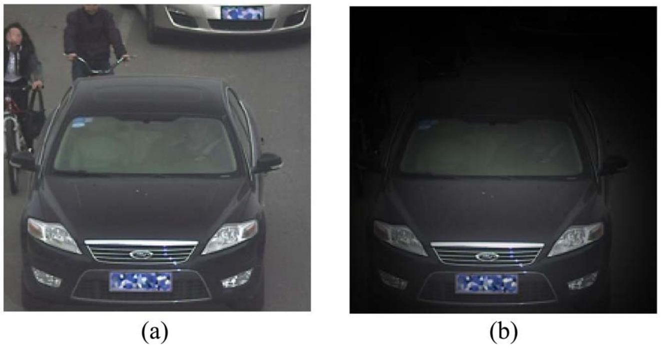

Fig. 4. Example of a focused image. (a) Raw image. (b) Focused image with a viewpoint.

$$\phi_{ij} = \text{sigm}(d) = \frac{1}{1 + \exp(\alpha(d + \beta))} \tag{2}$$

where $d = \sqrt{(i-u)^2 + (j-v)^2}$, and the adjustable parameters α and β decide the shape of the sigmoid function, which are fixed by the classification performance in the training phase. We set $\alpha = -0.0446$ and $\beta = 60$ for these values providing the best validation accuracy and a proper training time. Intuitively, the closer the pixel is to the focused position (u, v), the more attention it gets, as shown in Fig. 3.

After computing each element of the AWM $\mathbf{\Phi}$ by (2), we can then obtain the focused image by (1). Fig. 4 gives an example of the focused image with a center viewpoint. We can see that the attention areas in the focused image are highlighted and the others are weakened, which probably reduces the interferences of the background to the classification.

Given a focused position, the visual attention module can transform a raw image into a focused image, and the focused image highlights the attention areas of the image and weakens the others. However, for the focused image, we do not know whether attention areas are essential for classification. Therefore, we need to evaluate the selected viewpoint and then make a decision about the next viewpoint according to the evaluation result. Then, we will introduce the evaluation network.

V. Evaluation Network

In this paper, we take CNN as the evaluation network. With the multiple convolutional layers and down-sampling layers, CNN can automatically learn the useful representation for the classification target. Then through several nonlinear fully connected layers, CNN can classify the focused image. In fact, CNN outputs the probabilistic prediction of the classification. Here, we denote $\mathbf{P} \in \mathbb{R}_{+}^{C\times 1}$ as the probabilistic output of CNN for the classification of an input image, where C is the number of classes or the number of output notes. Each of its elements p_c means the probability that the input image $\mathbf{X}_f$ belongs to the class c, which is mathematically formulated as $p(c|\mathbf{X}_f)$. We take the class with the maximum probability as the final classification of the input image, as follows:

$$y = \arg\max_{c\in[1,C]}(p(c|\mathbf{X}_f)). \tag{3}$$

The probabilistic output $\mathbf{P}$ of CNN can be computed by a forward propagation algorithm, as follows:

$$\mathbf{P} = \text{cnnfp}(\mathbf{X}_f, \theta_{\mathbf{cnn}}) = \text{cnnfp}(\mathbf{\Phi} \odot \mathbf{X}; \theta_{\mathbf{cnn}}) \tag{4}$$

where “cnnfp” denotes the forward propagation function of CNN. $\theta_{\mathbf{cnn}}$ is the learning parameters of CNN, including the coefficients of convolutional maps, the connection weights and the bias of fully connected layers. $\mathbf{\Phi}$ is associated with the spatial locations of the pixels in an image, which means that CNN considers the spatial location information of the pixels.

$\mathbf{P}$ is the probability distribution of a classification. Its concentration reflects the confidence of CNN to the classification of the input image. The more flat the probability distribution of a classification is, the more difficult it is to distinguish the input image. When $\mathbf{P}$ is a uniform distribution, it means that the focused image can not been distinguished by the CNN. Information entropy can quantitatively measure the amount of the uncertainty of random variables. The less the information entropy of a random variable is, the less its uncertainty is. In this paper, we take information entropy of the classification probability $\mathbf{P}$ as the evaluation metric to judge the discrimination of the focused image and the classification confidence, which is calculated as follows:

$$H(P) = E[-\log(p_c)] = -\sum_{c=1}^{C} p_c \log(p_c), \quad \sum_{i=1}^{C} p_i = 1 \tag{5}$$

where $E[\cdot]$ is the expected value operator and log is the base 2 logarithm. To simplify, we use a const factor to normalize the information entropy, as follows:

$$H'(P) = \frac{H(P)}{\log(C)} = -\frac{\sum_{c=1}^{C} p_c \log(p_c)}{\log(C)}. \tag{6}$$

For the different focused positions, there are different attention areas. So we obtain different information entropy. A small $H'(P)$ means it is easy to discriminate the image with more confidence, which can guide us to find the key areas of the image to the classification.

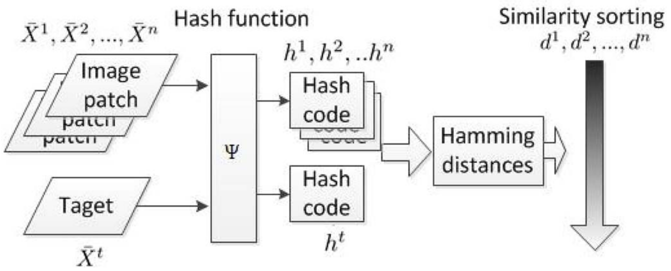

Fig. 5. Flowchart of hashing-based image retrieval.

VI. Viewpoint Selection Module

A. Artificial Rules-Based Method

Intuitively, if one area of an image is essential for its classification, the other areas similar to this area may also be the key areas. Based on this simple idea, we propose a rule-based viewpoint selection method, denoted as VS-Rule. First, we divide an image into several patches with $K \times K$. Then we randomly chose a patch and take its center point as the first focused position. According to the focused position, we can obtain the focused image with one attention area and its information entropy of the classification probability $H'(P)$. If $H'(P)$ is less than a given threshold H'_* or $H'(P)$ becomes smaller, we chose the most similar patch as the next one, otherwise, we chose the most dissimilar patch. And similarly we take the viewpoint of the chosen patch as the next focused position. Repeating this process until you can not find a better focused position, you will almost find all key areas that are helpful to the classification.

For the selection of similar patch, we use a hashing-based image retrieval method to compute the similarities between the patches [39], as shown in Fig. 5. In detail, we first construct a hashing function to transform each patch $\bar{\mathbf{X}}^{(i)}$ into a hashing code $\mathbf{h}_i$, as follows:

$$\mathbf{h}_i = \Psi\left(\bar{\mathbf{X}}^{(i)}\right) \tag{7}$$

where the hashing function$\Psi(\cdot)$ can be constructed with many methods [39] and we use the simple Google image selection method [40]. The similarities between the patches $\bar{\mathbf{X}}^{(i)}$ and $\bar{\mathbf{X}}^{(j)}$ can be quantitatively measured with the distance between $\mathbf{h}_i$ and $\mathbf{h}_j$, as follows:

$$d\left(\bar{\mathbf{X}}^{(i)}, \bar{\mathbf{X}}^{(j)}\right) = \left|\mathbf{h}_i - \mathbf{h}_j\right| \tag{8}$$

where $d(\bar{\mathbf{X}}^{(i)}, \bar{\mathbf{X}}^{(j)})$ is the similarities between $\bar{\mathbf{X}}^{(i)}$ and $\bar{\mathbf{X}}^{(j)}$, and the $|\cdot|$ denotes the norm operator of a vector.

B. Reinforcement Learning-Based Method

Although the rule-based viewpoint selection method is very simple and can find some key areas of an image, it heavily relies on the subjective experiences on vehicle classification and the careful observations of a variety of vehicles, which is high-cost. In addition, the rule-based viewpoint selection method has many limitations in dealing with the key areas that have complex relationships. For example, for a vehicle, the logo, headlights, doors, and wheels may be its key areas, but it is very hard for us to build a rule to guide the proposed model to find these key areas. The similarity-based rule can only deal with the simple situation, such as the symmetrical headlights.

Reinforcement learning is the well-known technique for sequential decision problems, which can automatically learn the optimal behaviors and has been widely applied to robotics, computer games, and network routing [41]. In fact, the viewpoint selection of the proposed CNNVA can be formulated as a sequential decision problem. We need to decide the next focused point according to the current state. In this paper, we propose to incorporate the reinforcement learning into the proposed CNNVA for the automatic viewpoint selection. We take the information entropy of the classification probability of a focused image as the reward to guide the agent of reinforcement learning to learn to find the next focused position.

1) Deep Reinforcement Learning: In this section, we will introduce the deep reinforcement learning. Reinforcement learning is learning how to map environment situations to actions, namely, knowing what to do by interacting with an environment at each of discrete time steps. Sutton and Barto [42] provided a typical reinforcement learning algorithm: temporal-difference algorithm and one important algorithm is Q-learning

$$Q(s,a) \leftarrow Q(s,a) + \alpha\left[r + \gamma \max_{a'} Q\big(s', \mathbf{a}'\big) - Q(s,a)\right] \tag{9}$$

where the action-value function, $Q(s,a)$, is the value of executing action a in state s, and α is the learning step size. At each time step, the agent makes decision to select an action a from the available actions set $\mathbf{a}$, then the environment presents a new state s' and also provides a reward r to the agent after executing action a.

In traditional reinforcement learning, the state s is hand-designed feature, obtained by specific sensors, such as speed, acceleration and displacement. However, in image classification task, the state s is a focused image with a viewpoint. Therefore, we need to use some methods to extract features of the focused image. Due to CNN showing remarkable capability of extracting abstract image features, this paper uses deep CNNs as Q-network to approximate the action-value function $Q(s,a)$, which is called deep reinforcement learning [43].

At step t, the current state s is the focused image $X_f(t)$. The action-value function Q in state s is computed by Q-network (deep CNN) F_Q

$$Q(s, \mathbf{a}) = F_Q(X_f(t); \theta_Q) \tag{10}$$

where $Q(s, \mathbf{a})$ represents the values in state s for all actions $\mathbf{a}$, and θ_Q represents the parameters of Q-network. In each step, we take the action a corresponding to the maximum $Q(s,a)$ of $Q(s, \mathbf{a})$. In this paper, the action set $\mathbf{a}$ includes sixteen actions (eight directions with two steps). That means the current focused point transfers a new focused point by moving a certain step in a direction.

a) Algorithm: In the proposed CNNVA model, the viewpoint selection module uses reinforcement learning (RL-based viewpoint selection module) to select key areas. It aims to train an agent that can automatically select key areas of an image. The learning process of the model is essentially the same as Q-learning process. For an image, the agent observes the environment (the focused image), takes an action a_t from action set $\mathbf{a}$ for getting another focused image and receives a reward at the current step t. The goal of the agent is also to maximize the cumulative rewards by selecting optimal policy of finding the next useful area.

In the model, a focused image with a focused point is a state s, which is generated by the visual attention module from Section IV. An action a represents selecting a specified area as the useful area (a focused point), which can be used to control the system state. By executing an action a, the system makes the state s transform into a new state s' (a focused image with a new focused point). r_t, the reward for doing the action a_t in the state s_t, is received from a reward network (detail is shown in the following). The value of taking the optimal action a in state s is corresponding to the optimal action-value function $Q^*(s,a)$. To achieve this, we use a neural network, named Q-network, to approximate the action-value function $Q(s,a)$. Finally, $Q(s,a)$ gradually approaches the optimal one $Q^*(s,a)$ by updating the Q-network.

b) Reward network: Reward is necessary in almost all reinforcement learning algorithms offering the goal of the reinforcement learning agent. The reward estimates how good the agent performs an action in a given state (or what are the good or bad things for the agent). In this paper, we design a reward network to map each state to a scalar, a reward, expressing the intrinsic expectation of the state.

For a classification task, an image is classified based on several key areas such as a headlight, a wiper, and a ventilation grille in a vehicle image. These features extracted from the key areas work together to determine the classification result.

In the RL-based viewpoint selection module, we use a focused image X_{mf} with several clear patches to put into the evaluate network (shown in Section V) and get an information entropy of the focused image to indicate how useful of these clear patches are

$$\mathbf{X}^*_{mf} = \max\big(\mathbf{X}_{f1}, \mathbf{X}_{f2}, \ldots, \mathbf{X}_{fn}\big) \tag{11}$$

where $[\mathbf{X}_{f1}, \mathbf{X}_{f2}, \ldots, \mathbf{X}_{fn}]$ are the focused images with only one clear patch which has proven to be useful for the task.

At step t, we select a new patch, and generate a new focused image $\mathbf{X}_{mf,t}$ which adds the selected new patch as another clear patch, and also retains the past clear patches

$$\mathbf{X}_{mf} = \max\Big(\mathbf{X}^*_{mf}, \mathbf{X}_{f,t}\Big). \tag{12}$$

So the information entropy $H'(\mathbf{X}_{mf,t})$ of the new focused image $\mathbf{X}_{mf,t}$, calculated by (6), can evaluate the value of these clear patches. If the predicted result of $\mathbf{X}_{mf,t}$ is right and the information entropy $H'(\mathbf{X}_{mf,t})$ is smaller than $H'(\mathbf{X}^*_{mf})$ of the focused image $\mathbf{X}^*_{mf}$, it means that the new selected patch is useful for our task. Then the agent would receive a positive

Algorithm 2 RL-Based Viewpoint Selection Algorithm

```
repeat
    get the maximum term $Q(s, a)$ from $Q(s, \mathbf{a})$
    get the best action $a$ through Q-learning network.
    execute the action $a$
    generate a focused image with the new viewpoint $(u, v) \leftarrow (u, v) + a$, which is the new state $s'$.
    receive the action-value function $Q(s', \mathbf{a}')$
    receive the reward $r_t$ of the new state $s'$.
    update $Q$ with the reward for the action and the maximum $Q(s', a')$ of the next state based on all actions. $Q(s, a) = r_t + \gamma \max_{a'} Q(s', \mathbf{a}')$
    update the state $s$:$s \leftarrow s'$
until reach the goal
```

reward and the selected patch will be retained

$$r_t = \begin{cases} -1, & y! = y* \\ 0, & y = y * \& H'(\mathbf{X}_{mf,t}) > H'\left(\mathbf{X}^*_{mf}\right) \\ 1, & y = y * \& H'(\mathbf{X}_{mf,t}) \leq H'\left(\mathbf{X}^*_{mf}\right) \end{cases} \tag{13}$$

where y is the prediction and $y*$ is the label of the image. The reward is 1 with a right classification result and an effective state. Otherwise, the reward is less than 1 in the condition of a noneffective state or a wrong prediction.

With a positive reward, we store the current selected patch and update the focused image $X^*_{mf} \leftarrow X_{mf,t}$.

2) Learning: The RL-based viewpoint selection module uses Q-network to select key areas, of which parameters are updated by one-step Q-learning algorithm

$$Q(s_t, a_t) = r_t + \gamma \max_a Q(s_{t+1}, \mathbf{a}) \tag{14}$$

where r_t is the reward at the step t obtained from (13), the discount rate γ is set $0 \leq \gamma \leq 1$.

The RL-based viewpoint selection algorithm is shown as Algorithm 2.

The RL-based viewpoint selection algorithm is used to train the agent learning experience. The input of the algorithm is the current state (the focused image with the current viewpoint) and the reward, and the output is the next state (the focused image with the selected viewpoint). It is called in line 6 of Algorithm 1 (namely, the viewpoint selection algorithm). In each episode (training section), each image of the dataset goes through the RL-based viewpoint selection algorithm for once. For an image, the agent explores the environment and gets the reward until it receives a right prediction result with a high confidence. The purpose of the training is to enhance the brain of the agent that is represented by action-value function Q. More training will be given to the Q-learning network that can be used by the agent to select a better viewpoint.

The Q-network converges to the optimal action-value function $Q_t \to Q^*$ as $t \to \infty$. We train the Q-network by adjusting the parameters θ_Q at each step to maximize the mean-squared error $L(\theta_Q)$

$$L(\theta_Q) = (y - Q(s, a; \theta_Q))^2 \tag{15}$$

where $y = r + \gamma \max Q(s', \mathbf{a}'; \theta'_Q)$ can be considered as a target like traditional machine learning which is fixed during optimization and $Q(s, a; \theta_Q)$ is the predicted result. When optimizing θ_Q, we keep the θ'_Q fixed. So the differential equation of $L(\theta_Q)$ can be calculated by

$$\nabla L_i(\theta_i) = \left(r + \gamma \max_{a'} Q\left(s', a'; \theta_{i-1}\right) - Q(s, a; \theta_i) \right) \nabla Q(s, a; \theta_i) \tag{16}$$

where $\nabla Q(s, a; \theta_i)$ is the gradient of Q. We can use stochastic gradient descent to optimize the loss function of Q-network like the traditional machine learning.

VII. Classification and Optimization

A. Classification

After the previous modules, we get several key areas of an image. In this part, we will introduce how to provide a predicted class with high confidence based on the selected key areas.

For a classification task, an image is classified based on several important areas such as a headlight, a wiper, and a ventilation grille in a vehicle image. These features extracted from the important areas work together to determine the classification result. In the proposed CNNVA model, we observe the object with different viewpoints, and then integrate the images in different viewpoints to make a comprehensive decision on the classification.

In detail, we will select different key areas for the vehicle classification. After several repeats, we find almost all the key areas for the classification, and then we incorporate them into one image $\mathbf{X}^*_{mf}$ form (1) and (11) for the final classification (shown in Fig. 9). The $\mathbf{X}^*_{mf}$ is input into the classifier-evaluation network and a final classification y of the image is obtained from (3).

We will stop searching the key area until the evaluation network classifies the focused images with more confidence than the threshold or the number of found key areas is more than the given threshold $n^* = 5$. So if an image is clearer and easier to be predicted, the fewer key areas are to be found, otherwise, we will find more key areas for classification but no more than the given threshold $n^* = 5$.

B. Optimization

At the beginning, we use the raw images to train an initial evaluation network (CNNs). The initial evaluation network can get a relatively reliable classification and avoid that CNNVA-RL converges to the bad direction. For an image, we may find several key area candidates. The selection network (deep reinforcement learning) will update its parameters to find the more useful viewpoint. It does not stop until we find enough key areas to make the confidence of the result above the threshold H^*. Then we will update the parameters of the evaluation network (CNNs) on the focused image with all the found key areas.

In order to optimize the evaluation network, we present the input and output of each layer of the evaluation network.

Taking $\mathbf{X}_{mf}$ as input of the evaluation network, we obtain the presentation of each layer

$$\begin{aligned}\mathbf{z}_{l+1} &= f_{l+1}(\mathbf{a}_l; \mathbf{W}_l, \mathbf{b}_l)\\ \mathbf{a}_{l+1} &= f(\mathbf{z}_{l+1}) = \max(\mathbf{0}, \mathbf{z}_{l+1})\end{aligned} \tag{17}$$

where $\mathbf{a}_l$ is the output of the layer (l) and also the input of the layer $(l+1)$, f_l is the operation of the layer (l), which contains three types: 1) convolution; 2) pooling; and 3) full connection, and f is the activation function. $\mathbf{W}_l$ are the parameters and $\mathbf{b}_l$ denotes the bias term of layer l. If the class of the input image is y^*, our purpose is to maximize the p_{y^*}, so the objective function is the cross-entropy error defined by

$$J\big(\mathbf{X}, y^*\big) = -\log\big(p\big(y^*\big|\mathbf{X}_f\big)\big). \tag{18}$$

The error metric can be computed according to (18), and then we can use the backward propagation of the error to update the parameters of the network

$$\nabla\mathbf{W}_l = \frac{\partial J}{\partial \mathbf{W}_l} = \frac{\partial J}{\partial \mathbf{z}_{l+1}}\frac{\partial \mathbf{z}_{l+1}}{\partial \mathbf{W}_l} = \frac{\partial J}{\partial \mathbf{z}_{l+1}}(\mathbf{a}_l)^T = \delta_l(\mathbf{a}_l)^T \tag{19}$$

where the variable δ is different among different layers. In the output layer, δ is obtained from

$$\delta_{n_l} = \frac{\partial J}{\partial \mathbf{a}_{n_l}} \odot \frac{\partial \mathbf{a}_{n_l}}{\partial \mathbf{z}_{n_l}} = \frac{\partial J}{\partial \mathbf{a}_{n_l}} \odot f'(\mathbf{z}_{n_l}). \tag{20}$$

For the lth layer $(l = n_l - 1,\ n_l - 2, \ldots, 2)$, δ is

$$\delta_l = \big((\mathbf{W}_l)^T \delta_{l+1}\big) \odot f'(\mathbf{z}_l). \tag{21}$$

After getting all the δ, the derivatives of the objective function of each layer are calculated by

$$\begin{aligned}\nabla\mathbf{W}_l &= (\delta_{l+1})(\mathbf{a}_l)^T\\ \nabla\mathbf{b}_l &= (\delta_{l+1}).\end{aligned} \tag{22}$$

VIII. Experiment

In this section, we systematically evaluate the performance of the proposed model on a surveillance-nature dataset. The surveillance-nature dataset contains images from the natural scenarios, which are captured by surveillance cameras in the front view. To evaluate the proposed model, we systematically compare with the typical CNN on different vehicle classification tasks. Moreover, we compare the rule-based viewpoint selection method and the RL-based viewpoint selection method, denoted as "CNNVA-Rule" and "CNNVA-RL," respectively.

A. Image Dataset

The vehicle dataset contains images from the natural scenarios, which are captured by surveillance cameras in the front view. Some examples of the images are illustrated in Fig. 6, which are unclear or incomplete, because of the impact of the bad weather (heavy fog and dim lights), poor photographic equipment and obstructions (pedestrian and vehicle). Even some images are too dim to recognize for human, shown in Fig. 7.

The vehicles can be organized in a hierarchical structure, consisting of two layers: 1) vehicle type and 2) vehicle maker depicted in Fig. 8. We define five types of vehicles, which are sedan, van, truck, SUV, and coach. For the vehicle maker, there are 58 auto makers. Each vehicle type can be produced in different vehicle makers, some of which are similar only with subtle difference in their appearances. For instance, two makers of "sedan" are displayed in the last line of Fig. 8, which are "MAZDA" and "HAIMA," respectively. They are similar in appearance and only have small difference in headlights. So other information of the two images is useless or even harmful

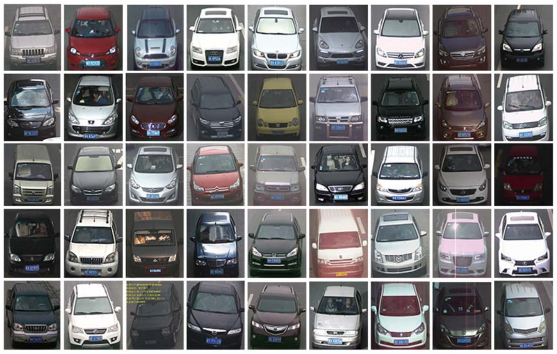

Fig. 6. Some vehicle images of the surveillance-nature.

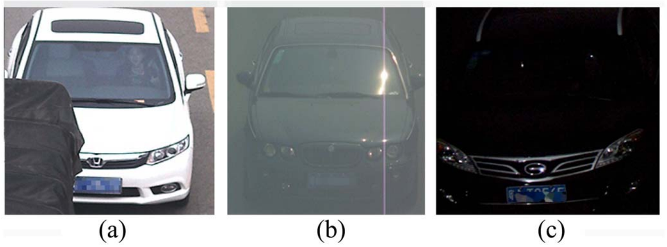

Fig. 7. Images of poor quality. (a) Being occluded. (b) With a poor photographic equipment. (c) With a dim light.

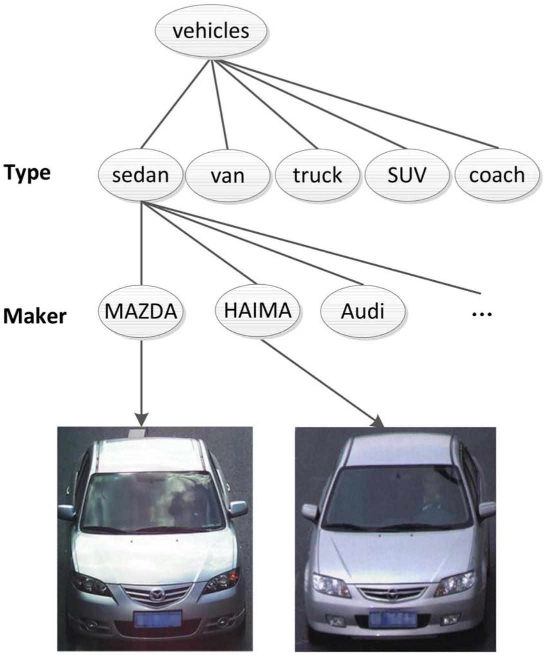

Fig. 8. Tree structure of vehicle hierarchy.

TABLE I
CLASSIFICATION PERFORMANCES OF DIFFERENT METHODS

Task	Method	Test accuracy rate	Objective
Vehicle-5	simple CNN	87.26%	-
	large-scale CNN	90.11%	4.83
	VGG5	94.81%	3.67
	CNNVA-Rule	97.12%	2.58
	CNNVA-RL	97.93%	2.12
Vehicle-58	simple CNN	-	-
	large-scale CNN	88.46%	7.23
	VGG58	91.8%	6.14
	CNNVA-Rule	95.13%	4.51
	CNNVA-RL	96.41%	2.84

to discriminate them. Thus, the second more challenging task needs more efficiency method to find the subtle difference.

In this paper, we select two tasks from the dataset according to the hierarchy to evaluate our methods. The first task is to divide vehicles into five categories according to their types and it consists of 2200 training images, 800 validation images, and 800 test images, labeled with five vehicle types, named Vehicle-5. The other task, named Vehicle-58, is vehicle classification discriminated from different vehicle makers, is vehicle classification discriminated from different vehicle makers which contains 13 000 training images, 2000 validation images, and 2000 test images. All the images are collected in the real traffic scenarios and divided randomly into training dataset, validation dataset, and test dataset. However, the number of images is not balanced among different vehicle makers, which scales from 20 to many thousands, due to the difficulty of collecting less popular vehicles, which also increases the difficulty of the task.

In the two tasks, every image has a different size and different length-width ratio, such as 478 × 631 and 596 × 744. However, our model's inputs require that images are squares. We compress all images into squares 224 × 224, because CNN has no requirement for images compression ratio consistency.

B. Results

In this section, we do some experiments to validate the quantitative effectiveness of our attention-based structures on task Vehicle-5 and Vehicle-58 and give a summary of the result in Table I.

1) Vehicle-5: We use five methods: 1) a simple CNN with fewer parameters than large-scale CNN; 2) large-scale CNN; 3) VGG5 which is a large-scale CNN and pretrained on ImageNet; 4) CNNVA-Rule; and 5) CNNVA-RL to complete the first task Vehicle-5. The simple CNN, consists of four convolutional layers followed by four pooling layers and 1 fully connected hidden layer and 1 soft-max output layer, which can be expressed by a short notation: $C(11, 4, 22) - R - N - P - C(7, 1, 22) - R - N - P - C(5, 1, 26) - R - C(5, 1, 26) - R - FC(120) - R - S(5)$, where $C(f, s, d)$ expresses that a convolutional layer concludes d filters and the spatial size is $f \times f$ with stride s. $FC(n)$ is the fully connected layer of n nodes. R is the activation function layer with rule function. N is the normalized layer. P is the pooling layer with size 3×3, stride 2 and max method. The last layer is the softmax output S, with 5 nodes as same as the classes we should predict. The large-scale CNN consists of five convolution layers and three fully connected layers: $C(5, 4, 64) - R - N - P - C(5, 1, 256) - R - N - P - C(3, 1, 256) - R - C(3, 1, 256) - R - C(3, 1, 256) - R - P - C(6, 1, 4096) - R - FC(4096) - R - S(5)$. The VGG5 method has the same layers with the large-scale CNN: $C(5, 4, 64) - R - N - P - C(5, 1, 256) - R - N - P - C(3, 1, 256) - R - C(3, 1, 256) - R - C(3, 1, 256) - R - P - C(6, 1, 4096) - R - FC(4096) - R - S(5)$. The difference between the large-scale CNN and VGG5 is that the parameters of the large-scale CNN are random initialization while the parameters of VGG5 are initialized by the parameters of Oxford VGG [29].

In our CNNVA models (CNNVA-Rule and CNNVA-RL), we use VGG5, as our evaluation network (shown in Section V) for providing an accurate information entropy of the current focused image, because the large-scale CNN has higher accuracy than the simple CNN which is proven in our experiments: from the first three lines of Table I, we find that the large-scale CNN has a better performance and higher accurate rate than simple CNN which means that large-scale CNN can extract more global, invariant and unique features among categories which are useful for the task. VGG5 shows a better performance and a shorter training time than the large-scale CNN with random initialization, meaning that the pretraining method can have a better initialization parameters trained on ImageNet and effectively save time in calculation.

We divide an image into $N = 25$ parts with size 45×45 of each patch. In viewpoint selection module of CNNVA-Rule, when we choose one part using (8), candidates are no more than 24. In CNNVA-RL, the selection network, Q-network, consists of four convolutional layers followed by 4 pooling layers and 1 fully connected hidden layer and 1 soft-max output layer: $C(11, 4, 22) - R - N - P - C(9, 1, 22) - R - N - P - C(7, 1, 26) - R - C(5, 1, 26) - R - FC(120) - R - S(16)$.

From the results, our proposed visual attention-based structures, CNNVA-Rule and CNNVA-RL, have boosts in performance than using VGG5, further increasing the accurate rate due to visual attention, which can remove the interference of useless information to make prediction only with effective information (extracted from images).

The objective, calculated from (18), presents the confidence of the prediction, namely, in the predicted result, the probability of the predicted label affects the credibility of the result, and the larger the probability is the more reliable the result should be. From the fourth column, we can see our visual attention-based structures have higher confidence of the prediction than VGG5. In visual attention architectures, CNNVA-RL accuracy results are significantly higher than CNNVA-Rule, and CNNVA-RL objective results are lower than that CNNVA-Rule, showing the strong decision and analysis capacity of deep reinforcement learning.

2) Vehicle-58: In the challenging task Vehicle-58, we compare the performance among large-scale CNN, VGG58, CNNVA-Rule, and CNNVA-RL shown in Table I. The large-scale CNN consists of five convolutional layers followed by 4 pooling layers and 1 full connected hidden layer and 1 soft-max output layer: $C(5, 4, 64) - R - N - P - C(5, 1, 256) - R - N - P - C(3, 1, 256) - R - C(3, 1, 256) - R - C(3, 1, 256) - R - P - C(6, 1, 4096) - R - FC(4096) - R - FC(58) - S$. The VGG58 also have the same network structure: $C(5, 4, 64) - R - N - P - C(5, 1, 256) - R - N - P - C(3, 1, 256) - R - C(3, 1, 256) - R - C(3, 1, 256) - R - P - C(6, 1, 4096) - R - FC(4096) - R - FC(58) - S$, where the parameters of the convolutional layers are similar to VGG5, initialized by the parameters of Oxford VGG [29]. The visual attention-based architectures, CNNVA-Rule and CNNVA-RL also use VGG58 as the evaluation network in task Vehicle-58.

The Vehicle-58 task is more challenging than Vehicle-5 for distinguishing categories only by some unique car parts, such as headlights and logos. So the accurate rate with VGG58 (91.8%) is lower than that with VGG5 (94.81%). For the difficult task, our methods (CNNVA-Rule and CNNVA-RL) outperform the large-scale CNN (VGG58) in accurate rate and in objective. And CNNVA-RL can also improve the performance on Vehicle-58 task compared to CNNVA-Rule.

Through introducing additional reinforcement learning network (Q-network) for choosing the next viewpoint, CNNVA-RL can outperform the large-scale CNN. On the surface, CNNVA-RL has more parameters (parameters of evaluation network and reinforcement learning network) than the lager-scale CNN. In fact, the size of reinforcement learning network is far less than that of the large-scale CNN, so CNNVA-RL pays a small cost for an apparent performance improvement. In addition, the reinforcement learning can find key areas, training the large-scale CNN (evaluation network) in these focused images with strengthening key areas.

For example, for the fuzzy images (shown as Fig. 7) and the similar images (as Fig. 8), our proposed methods can give correct classification results, "Jeep," "MG (morris garages)," "GAC (Guangzhou automobile group)," MAZDA, and HAIMA, while the predicted results of these fuzzy images using VGG58 are unsatisfactory with some results wrong.

From Table I, the VGG58 has a better performance than the large-scale CNN, which proves the CNNs pretrained on ImageNet have better initialization parameters. We can also see that the boosts on performance of our visual attention-based methods on Vehicle-58 (4%) are larger than that on Vehicle-5 task (3%), meaning that visual attention-based structure is more effective in challenging task, especially in the task with many categories discriminated only by subtle and local differences.

C. Analysis

In CNNVA-Rule, we select the next useful area at the principle of similarity. If the current selected area is proven to be useful by having low information entropy, we select the next area which is the most similar to the current selected area. By the principle, we get some key areas and highlight them in a focused image shown in Fig. 9. Fig. 9(a) is the raw data and Fig. 9(b) is the selected key areas by the principle of similarity. The first red box tagged by "1" is the first key area, the red box tagged by "2" is the next selected key area, the third selected key area is the one tagged by "3," and the last one is tagged by "4." We use this focused image to put into the CNN network to recognize its category. However, there are some useless areas for the classification, such as the box tagged by 2 and the box tagged by 3. The box tagged by 2 has the same effect as the one tagged by 1, and the one tagged by 3 has nothing to do with the classification and even has interference on classification.

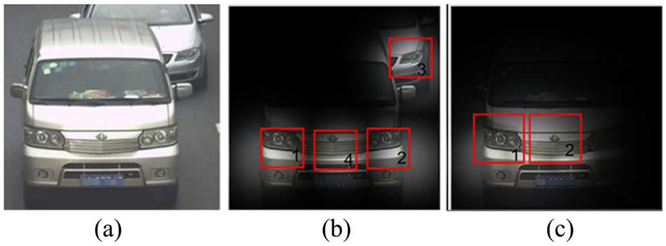

Fig. 9. Selecting results of CNNVA-Rule and CNNVA-RL model. (a) Raw data. (b) Selected key areas by the principle of similarity. (c) Selected key areas based on reinforcement learning.

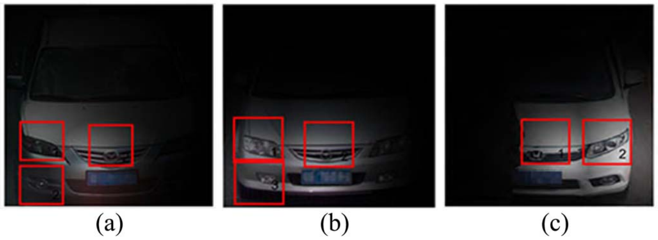

Fig. 10. Selecting results of CNNVA-RL model. Vehicle of (a) MAZDA and (b) HAIMA shown in Fig. 8. (c) Image with obstacles shown in Fig. 7(a).

CNNVA-RL can avoid selecting the redundant areas which have the same effect on the task, reducing the computation cost. We add a new selected area as another clear area with high resolution on the focused image. If the information entropy of the focused image after adding a new clear area, becomes high, we mark the new area as a useful area and give a positive reward. With this principle, we train an agent to select useful areas automatically and the result is shown in Fig. 9(c). The first key area is the left-headlight surrounded by a red box tagged by 1, and the next key area is tagged by 2. The Fig. 9(c) combines the two key areas which are used to predict its category. Compared with CNNVA-Rule, this viewpoint selection module selects two key areas, while the CNNVA-Rule selects four key areas which contain two useless patches for the classification. So the CNNVA-RL can use reinforcement learning to search the key areas efficiently to improve the classification accuracy and decrease the objective.

CNNVA-RL can give the correct labels of the similar images (as Fig. 8). The selecting results of the two images are shown in Fig. 10(a) and (b), representing MAZDA and HAIMA, respectively. The two makers (MAZDA and HAIMA) have some differences in headlights and logo. Our method can find these differences which are marked in red-box and highlighted by visual attention module. Therefore, the two makers can be distinguished by extracting features from the three key areas. Similarly, images where object vehicles are occluded by obstacles can be recognized their labels through finding and highlighting key areas shown in Fig. 10(c) [its raw image shown in Fig. 7(a)]. For some dim images, our method highlights the key areas and dim others, meaning that the key areas become clear relatively, which can also be recognized.

IX. Conclusion

We propose the visual attention-based deep reinforcement learning model for image classification. A visual attention module is presented to mimic this characteristic of human visual attention, by formulating focused images with key areas clear. Then we use two methods to select the key areas: 1) artificial rule and 2) reinforcement learning. The artificial rule-based selecting viewpoint aims to find key areas by selecting similar patches with hashing-based image retrieval method, and the reinforcement learning can train an agent to search key areas automatically from experience. The proposed methods are tested on surveillance-nature dataset. The accurate rate of the proposed methods is increased about 3% compared with large-scale CNN, which shows that the proposed methods can boost performance for image classification by visual attention.

References

[1] J. Douret and R. Benosman, “A multi-cameras 3D volumetric method for outdoor scenes: A road traffic monitoring application,” in *Proc. IEEE Conf. Pattern Recognit.*, Cambridge, U.K., 2004, pp. 334–337.

[2] D. Preotiuc-Pietro and F. Hristea, “Unsupervised word sense disambiguation with n-gram features,” *Artif. Intell. Rev.*, vol. 41, no. 2, pp. 241–260, 2014.

[3] O. Amayri and N. Bouguila, “A study of spam filtering using support vector machines,” *Artif. Intell. Rev.*, vol. 34, no. 1, pp. 73–108, 2010.

[4] B. Gu, V. S. Sheng, K. Y. Tay, W. Romano, and S. Li, “Incremental support vector learning for ordinal regression,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 7, pp. 1403–1416, Jul. 2015.

[5] B. Gu *et al.*, “Incremental learning for ν-support vector regression,” *Neural Netw.*, vol. 67, pp. 140–150, Jul. 2015.

[6] A. Liaw and M. Wiener, “Classification and regression by randomforest,” *R News*, vol. 2, no. 3, pp. 18–22, 2002.

[7] J. Shotton, J. Winn, C. Rother, and A. Criminisi, “Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context,” *Int. J. Comput. Vis.*, vol. 81, no. 1, pp. 2–23, 2009.

[8] X. Wen, L. Shao, Y. Xue, and W. Fang, “A rapid learning algorithm for vehicle classification,” *Inf. Sci.*, vol. 295, pp. 395–406, Feb. 2015.

[9] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[10] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, San Diego, CA, USA, 2005, pp. 886–893.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[12] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Pattern Recognit.*, Las Vegas, NV, USA, 2015, pp. 770–778.

[13] O. Russakovsky *et al.*, “Imagenet large scale visual recognition challenge,” *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[14] P. Sermanet *et al.*, “OverFeat: Integrated recognition, localization and detection using convolutional networks,” in *Proc. IEEE Int. Conf. Learn. Represent.*, Banff, AB, Canada, 2014.

[15] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, “Deepface: Closing the gap to human-level performance in face verification,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, 2014, pp. 1701–1708.

[16] Y. Sun, X. Wang, and X. Tang, “Deep learning face representation from predicting 10,000 classes,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, 2014, pp. 1891–1898.

[17] D. Drubach, *The Brain Explained*. Upper Saddle River, NJ, USA: Prentice-Hall, 2000.

[18] R. A. Rensink, “The dynamic representation of scenes,” *Visual Cogn.*, vol. 7, nos. 1–3, pp. 17–42, 2000.

[19] M. Begum and F. Karray, “Visual attention for robotic cognition: A survey,” *IEEE Trans. Auton. Mental Develop.*, vol. 3, no. 1, pp. 92–105, Mar. 2011.

[20] J. F. Ferreira and J. Dias, “Attentional mechanisms for socially interactive robots—A survey,” *IEEE Trans. Auton. Mental Develop.*, vol. 6, no. 2, pp. 110–125, Jun. 2014.

[21] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, “Recurrent models of visual attention,” in *Proc. Adv. Neural Inf. Process. Syst.*, Montreal, QC, Canada, 2014, pp. 2204–2212.

[22] Y. LeCun *et al.*, “Backpropagation applied to handwritten zip code recognition,” *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.

[23] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[24] L. Fei-Fei, R. Fergus, and P. Perona, “Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories,” *Comput. Vis. Image Understand.*, vol. 106, no. 1, pp. 59–70, 2007.

[25] Y. LeCun, F. J. Huang, and L. Bottou, “Learning methods for generic object recognition with invariance to pose and lighting,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Washington, DC, USA, 2004, pp. 97–104.

[26] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, “Learning hierarchical features for scene labeling,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, Aug. 2013.

[27] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, “CNN features off-the-shelf: An astounding baseline for recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, 2014, pp. 512–519.

[28] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in *Computer Vision—ECCV 2014*. Zürich, Switzerland: Springer, 2014, pp. 818–833.

[29] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *Comput. Sci.*, 2014.

[30] A. M. Treisman and G. Gelade, “A feature-integration theory of attention,” *Cogn. Psychol.*, vol. 12, no. 1, pp. 97–136, 1980.

[31] L. Itti, C. Koch, and E. Niebur, “A model of saliency-based visual attention for rapid scene analysis,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov. 1998.

[32] G. Underwood, “Cognitive processes in eye guidance: Algorithms for attention in image processing,” *Cogn. Comput.*, vol. 1, no. 1, pp. 64–76, 2009.

[33] T. Foulsham and G. Underwood, “If visual saliency predicts search, then why? Evidence from normal and gaze-contingent search tasks in natural scenes,” *Cogn. Comput.*, vol. 3, no. 1, pp. 48–63, 2011.

[34] L. Lukic, A. Billard, and J. Santos-Victor, “Motor-primed visual attention for humanoid robots,” *IEEE Trans. Auton. Mental Develop.*, vol. 7, no. 2, pp. 76–91, Jun. 2015.

[35] Z. Yücel *et al.*, “Joint attention by gaze interpolation and saliency,” *IEEE Trans. Cybern.*, vol. 43, no. 3, pp. 829–842, Jun. 2013.

[36] K. Xu *et al.*, “Show, attend and tell: Neural image caption generation with visual attention,” in *Proc. IEEE Conf. Int. Conf. Mach. Learn.*, Lille, France, 2015, pp. 2048–2057.

[37] M. Ranzato, “On learning where to look,” *Comput. Sci.*, 2014.

[38] T. Xiao *et al.*, “The application of two-level attention models in deep convolutional neural network for fine-grained image classification,” *J. Manag.*, vol. 40, no. 1, pp. 130–160, 2014.

[39] Y. G. Jiang, J. Wang, X. Xue, and S.-F. Chang, “Query-adaptive image search with hash codes,” *IEEE Trans. Multimedia*, vol. 15, no. 2, pp. 442–453, Feb. 2013.

[40] S. Baluja and M. Covell, “Similar image selection,” U.S. patent US8 611 617 P, 2013.

[41] M. Wiering and M. van Otterlo, *Reinforcement Learning: State-of-the-Art*. New York, NY, USA: Springer, 2012.

[42] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 1998.

[43] V. Mnih *et al.*, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

Dongbin Zhao (M'06–SM'10) received the Ph.D. degree from the Harbin Institute of Technology, Harbin, China, in 2000.

He was a Post-Doctoral Fellow with Tsinghua University, Beijing, China, from 2000 to 2002. Since 2002, he has been an Associate Professor with the State Key Laboratory of Management and Control for Complex Systems, Chinese Academy of Sciences, Beijing, China, where he is currently a Professor. Since 2012, he has been a Professor with the Institute of Automation, Chinese Academy of Sciences. He is also a Professor with the University of Chinese Academy of Sciences, Beijing. He has published four books, and over 50 international journal papers. His current research interests include computational intelligence, adaptive dynamic programming, deep reinforcement learning, robotics, intelligent transportation systems, and smart grids.

Dr. Zhao has been an Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, since 2012 and the *IEEE Computation Intelligence Magazine*, since 2014. He has been serving as the Chair of Adaptive Dynamic Programming and Reinforcement Learning Technical Committee since 2015, Multimedia Subcommittee of the IEEE Computational Intelligence Society, since 2015. He serves as a several guest editors of international journals. He is involved in organizing several international conferences.

Yaran Chen received the B.S. degree in automation from Harbin Engineering University, Harbin, China, in 2013. She is currently pursuing the Ph.D. degree in control theory and control engineering with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China.

Her current research interests include deep learning, computer vision, and reinforcement learning.

Le Lv received the B.S. degree from the School of Automation, University of Science and Technology at Beijing, Beijing, China, in 2011. He is currently pursuing the Ph.D. degree with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing.

His current research interests include computer vision, deep learning, and reinforcement learning.