EasyChair Preprint

№ 3787

An AI-based Visual Attention Model for Vehicle Make and Model Recognition

Xiren Ma and Azzedine Boukerche

July 7, 2020

An AI-based Visual Attention Model for Vehicle Make and Model Recognition

Xiren Ma, and Azzedine Boukerche
PARADISE Research Laboratory, EECS, University of Ottawa, Canada
Emails: xma026@uottawa.ca, boukerch@eecs.uottawa.ca

***Abstract*—With the increasing highlighted security concerns in Intelligent Transportation System (ITS), Vehicle Make and Model Recognition (VMMR) has attracted a lot of attention in recent years. The VMMR method can be widely used in suspicious vehicle recognition, urban traffic monitoring, and the automated driving system. With the development of the Vehicle-to-Everything (V2X) technology, the vehicle information recognized by the AI-based VMMR method can be shared among vehicles and other participants within the transportation system, and can help the police fast locate the suspicious vehicles. VMMR is complicated due to the subtle visual differences among vehicle models. In this paper, we propose a novel Recurrent Attention Unit (RAU) to expand the standard Convolutional Neural Network (CNN) architecture for VMMR. The proposed RAU learns to recognize the discriminative part of a vehicle on multiple scales and builds up a connection with the prominent information in a recurrent way. RAU is a modular unit. It can be easily applied to different layers of the vanilla CNN architectures to boost their performance on VMMR. The efficiency of our models is tested on three challenging VMMR benchmark datasets, i.e., Stanford Cars, CompCars, and CompCars Surveillance. The proposed ResNet101-RAU achieves the best recognition accuracy of 93.81% on the Stanford Cars dataset and 97.84% on the CompCars dataset.**

***Index Terms*—Intelligent transportation system, convolutional neural network, recurrent attention, visual attention, vehicle make and model recognition.**

I. Introduction

Vehicle Make and Model Recognition (VMMR) is an important subject among various Intelligent Transportation System (ITS) applications [1], [2]. The vision-based VMMR method can directly recognize the vehicle information (e.g., vehicle make, model and year) according to the appearance. This method will be promoting in solving security-related issues.

For example, when a vehicle is stolen or grabbed by a criminal, the police usually need to set up inspection posts to intercept passing vehicles or manually search for a vehicle with specific type, make and model through the surveillance camera pre-deployed around the city, which takes much time and is inaccurate in the end. With the help of the VMMR technology, once the vehicle image is captured by an on-board camera, the information of the stolen vehicle or the vehicle driven by the suspicious criminal can be fast and accurately identified. Then, relying on the vehicle-to-everything techniques (e.g., the application of 802.11p [3] Standard and Dedicated Short-Range Communication (DSRC) [4] Standard for Wireless Access in Vehicular Environment (WAVE)) the derived information can be forwarded to the nearby police officers or a remote central office [5], [6]. However, VMMR is challenging, primarily because vehicle models of the same make contain subtle visual difference, and different companies produce vehicles that have a similar shape or appearance. Moreover, the uncertainty of circumstances where the vehicle located (e.g., the unconstrained pose, different illumination, and cluttered background of the vehicle, etc.) would cause a failure when recognizing a vehicle.

Recently, Convolutional Neural Network (CNN) is widely used to solve various computer vision tasks such as vehicle detection and image classification. Also, there are various methods which are based on the standard CNN architectures (e.g., VGG [7], ResNet [8]) to solve the VMMR task. The deep learning-based VMMR method can be classified into three categories: part detector-based methods [9], Recurrent Neural Network (RNN)-based attention methods [10], and feed-forward attention methods [11].

To resolve the drawback of slow recognition speed of RNN-incorporated models and take advantage of attention mechanism [12], we propose the Recurrent Attention Unit (RAU), which is a modular attention unit that can be applied on different layers of standard CNN architectures to extract prominent information from different scales. Each RAU takes the feature maps generated by the convolutional layer and the attention state generated by the previous RAU as inputs. Afterward, the unit produces a new state for the next RAU. The mechanism is not only extracting discriminative information from different layers but also combining them. This process allows the model to recognize an object by evaluating the discriminative features of different resolutions.

Generally, when an image contains multiple vehicles, the object detection method is first used to locate these vehicles. Then, these vehicles are cropped out and sent to the VMMR model. Since our work mainly focuses on recognizing fine-grained information of a vehicle, we assume that the image received by the VMMR model contains only one vehicle. Besides, our VMMR models have the ability to batch process images. Therefore, they can simultaneously identify the information of multiple vehicles.

The rest of the paper is organized as follows: We review the related work in Section II. Section III introduces the structure

This work is partially supported by NSERC-SPG, NSERC-DISCOVERY, Canada Research Chairs Program, NSERC-CREATE TRANSIT Funds.

of RAU. In Section IV, we evaluate our model and provide detailed analysis. Finally, we conclude our work in Section V.

II. Related Work

VMMR is a subcategory of fine-grained recognition, and it has been studied for many years. A variety of methods have been developed to distinguish fine-grained categories [13]. In this section, we mainly introduce RNN-based attention models and feed-forward attention models for fine-grained recognition because they are the most relevant work.

A. RNN-based attention models

Humans recognize an object through multiple glimpses because of the built-in biological attention mechanism [14]. Inspired by the biological attention mechanism, Recurrent Neural Networks (RNNs) such as vanilla RNN and Long-Short-Term Memory (LSTM) [15] are introduced to combine with the CNN, which enables models to focus on multiple discriminative parts of the object.

Diversified Visual Attention Network (DVAN) [10] forces the model to vary the attention regions. Attention canvases are used to select different regions of the original image. The generated attentive features are integrated by LSTM. The drawback of DVAN is that the whole model needs to be trained separately, and the parameters need to be carefully selected. Wu et al. [16] proposed a spatial LSTM to capture the spatial relationship of the local features. Also, an attention location matrix was introduced to weigh the importance of the bilinear features [17] in each location.

Fully Convolutional Attention Network (FCAN) was proposed in [18]. The attention procedure is formulated into a Markov Decision Process. At each time step, the attention network generates a single-channel confidence map. A part region is cropped from the feature maps based on the confidence map. Reinforcement learning is used to optimize the attention selection strategy. The processing procedure of their model is different from RAU. Their model generates different parts through multiple time steps. Our model only needs one feed-forward process to generate multiple attention regions.

B. Feed-forward attention models

Incorporating the feed-forward attention mechanism is another way to enhance the standard CNN architectures. These models expand the CNN architectures by generating the attention masks on several certain layers of the standard CNN architectures. These attention masks are then used to attend the feature maps to select the discriminative information of an image.

Bilinear CNN (BCNN) was proposed in [17], which uses two CNNs to generate image representations (one CNN is used to recognize the object parts). These two representations are combined by an outer product, generating an image descriptor. Inspired by BCNN, the Spatially Weighted Pooling (SWP) method was proposed in [19]. SWP consists of a set of predefined weighted masks. It directly pools the feature maps and outputs the image feature representations.

Rodriguez et al. [11] proposed a modular attention architecture. The attention model generates the attention map to select the discriminative part features and uses the proposed attention gate module to select the generated global and part features. The authors applied their proposed model to Wide Residual Network [20], and then obtained a new model, Wide Attentional Residual Network (WARN). WARN is the most relevant model among our proposed models, but the internal structure of their attention module and the working mechanism are different from RAU's.

The knowledge transfer method was revisited in [12]. In this work, the activation-based and gradient-based attention transfer methods are applied to a proposed teacher-student network. The authors obtained a significant improvement on the student network. They visualized several attention maps which are generated according to the feature maps. They found that mid-level attention maps have higher activation values in smaller regions (e.g., human eyes and nose), and high-level attention maps focus on larger regions (e.g., the whole face) [12]. The incentive of RAU is based on the observation of the attention mechanism. Therefore, an attempt to connect the features generated from different layers was made and was achieved by adopting the RNN structure. This allows discriminative features at different scales to be taken into consideration.

III. The Proposed Approach

RAU is designed as a modular attention unit that can be applied to different convolutional layers without changing the original structure of the standard CNN, which allows the transfer learning approach to be instantly applied to the proposed models. It seamlessly enhances the performance of standard CNN architectures such as ResNet on solving the VMMR task.

A. Recurrent Attention Unit

Based on the recently proposed Prototype Recurrent Unit (PRU) in [21], we build the Recurrent Attention Unit (RAU). PRU was proposed as a prototypical example for future study of LSTM-like recurrent networks [21]. PRU has a compact structure and captures the key component of LSTM and GRU [22]. PRU can only process the one-dimensional sequence vector. As an improvement, our RAU can directly process the feature maps extracted by the convolutional layers.

The detailed structure of RAU is illustrated in Fig. 1. The standard CNN takes the image as input to generate multi-scale feature maps along the feed-forward process. Then each RAU receives the feature maps and the former attention state generated by the previous RAU as inputs. The initial attention state is set to 0. After a series of operations on the feature maps and the attention state, RAU will output a new attention state. Each attention state represents the scores of vehicle models on multiple discriminative locations.

The internal structure of RAU can be concluded as three parts: feature integration, attention mask generation, and attention state generation. For the feature integration submodule, we

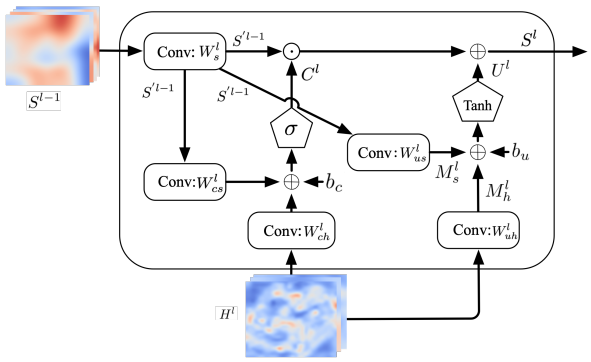


Fig. 1. The detailed structure of the Recurrent Attention Unit. The convolutional layer is denoted by $Conv$.

integrate the features from the feature maps and the attention state. First, the feature maps are processed by a convolutional layer to generate $M_h^l \in \mathbb{R}^{kn \times h^l \times w^l}$:

$$M_h^l = W_{uh}^l * H^l, \tag{1}$$

where $H^l \in \mathbb{R}^{c^l \times h^l \times w^l}$ are the feature maps for the lth RAU, $l \in [1, ..., L]$. The height and width of the feature maps are denoted by h^l and w^l, c^l is the channel of the feature maps. The number of attention masks is denoted by k, n is the number of vehicle model categories. The whole convolution process of the convolutional layer is denoted by '$*$'. W_{uh}^l represents the overall parameters of the convolutional layer.

To make the attention state $S^{l-1} \in \mathbb{R}^{kn \times h^{l-1} \times w^{l-1}}$ be compatible with the parameters of current RAU, a convolutional layer is utilized to reduce the spatial size of the attention state S^{l-1} and we use another convolutional layer to generate M_s^l:

$$S'^{l-1} = W_s^l * S^{l-1}, \tag{2}$$

$$M_s^l = W_{us}^l * S'^{l-1}, \tag{3}$$

where $S'^{l-1} \in \mathbb{R}^{n \times h^l \times w^l}$, $M_s^l \in \mathbb{R}^{kn \times h^l \times w^l}$. Then M_h^l and M_s^l are combined together to generate the feature representation U^l:

$$U^l = tanh(M_h^l + M_s^l + b_u), \tag{4}$$

where $U^l \in \mathbb{R}^{kn \times h^l \times w^l}$, $tanh$ introduces nonlinearity to the sum of M_h^l and M_s^l, b_u is the bias.

For the attention mask generation submodule, whether the attention mask is generated by considering features from the feature maps and the attention state is what we want to ensure. Therefore, H^l and S'^{l-1} are sent to two convolutional layers to generate k learnable single-channel attention masks respectively, and then they are added together:

$$C^l = \sigma(W_{ch}^l * H^l + W_{cs}^l * S'^{l-1} + b_c), \tag{5}$$

where $C^l \in \mathbb{R}^{k \times h^l \times w^l}$ will select the information of the attention state S'^{l-1}. Theses attention masks focus on the discriminative regions of S'^{l-1}. The sigmoid function is denoted by σ.

TABLE I
THE DETAILED INFORMATION OF THE WEIGHTS OF THE CONVOLUTIONAL LAYERS IN RAU

weight name	size of the kernel	stride	number
W_{uh}	1×1	1	kn
W_{ch}	3×3	1	k
W_s	3×3	2	n
W_{us}	1×1	1	kn
W_{cs}	3×3	1	k

The new attention state $S^l \in \mathbb{R}^{kn \times h^l \times w^l}$ is the sum of the selected features and U^l:

$$S^l = C^l \odot S'^{l-1} + U^l, \tag{6}$$

where $\odot$ denotes the element-wise product. The feature selection operation is done by repeating each mask for each of the feature channels of S'^{l-1} and conducting element-wise production. The attention state S^l represents the features of n vehicle models on k discriminative locations. In VMMR, the module should extract distinctive information of the object. We do not want to discard (weaken) the semantic information of the same region in the feature representation U^l. Therefore, U^l is directly added in Eq. (6), rather than adding the $(1-C^l) \odot U^l$ as in [21].

The difference between different vehicle models is quite subtle. Feature maps of different scales can provide different levels of information. Therefore, it is useful to collect and combine sufficient discriminative features from multi-scale regions. Therefore, we deploy the RAUs to different convolutional layers to receive the feature maps of different scales. The attention state S^l generated by the last RAU represents the integrated part features. The Global Average Pooling layer [8] is used to pool these part features, resulting a $kn \times 1 \times 1$ vector. The vector represents the scores of n vehicle models selected by k attention masks. Then, the vector is processed by the fully-connected (FC) layer to generate the classification scores.

The final recognition result is the average of the results of the standard CNN and the last RAU. The loss function is defined as:

$$L = \frac{1}{2} \sum_{i=1}^{n} \hat{Y}_i log Y_i + \frac{1}{2} \sum_{i=1}^{n} \hat{Y}_i log P_i, \tag{7}$$

where $P \in \mathbb{R}^n$ is the predicted classification probabilities from the last RAU, Y_i is the predicted probability for class i from the standard CNN, $\hat{Y}$ is the one-hot encoded ground truth label vector. This loss function simultaneously optimizes the global and part feature extraction abilities of the model. The detailed information about the weights of the convolutional layers in RAU are provided in Table I.

B. Using RAU with ResNet

RAU is compatible with most standard CNN architectures. We choose two strong baselines ResNet50 and ResNet101 [8] as our base models. By deploying RAUs to these two base models, we get two new models, ResNet50-RAU and

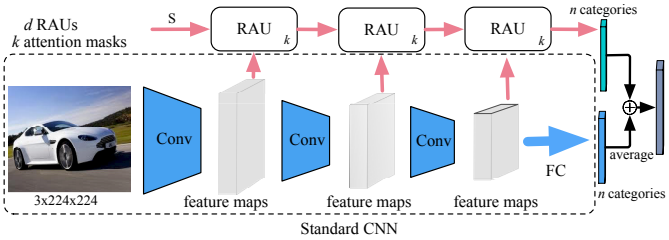
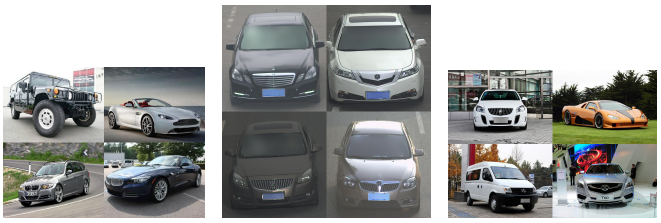


Fig. 2. The framework of the proposed model. Three Recurrent Attention Units (RAUs) are applied to the original CNN architecture at three different layer groups. Each RAU contains k attention masks. The final results come from the original CNN and the final RAU.



(a) Stanford Cars (b) CompCars SV (c) CompCars

Fig. 3. Samples from (a) Stanford Cars dataset, (b) CompCars Surveillance dataset and (c) CompCars dataset in our experiments.

ResNet101-RAU. A three-RAU structure ResNet-RAU is presented in Fig. 2. ResNet contains several groups of convolutional layers, which are called layer groups. Each layer group contains a set of Residual Network [8], which contains convolutional layers with the same number of channels. ResNet50 and ResNet101 both consist of four convolutional layer groups, but each layer group contains a different number of convolutional layers. The sizes of the feature maps generated by theses four layer groups are $256 \times 56 \times 56$, $512 \times 28 \times 28$, $1024 \times 14 \times 14$, and $2048 \times 7 \times 7$. We deploy RAUs to the end of the last three layer groups to test the effectiveness of our proposed module. The recognition ability of each RAU is controlled by the number of attention masks k. It is worth mentioning that each attention mask can focus on different locations. The final classification result of the whole model is the average of the results from the global stream (standard CNN) and the part stream (RAUs). The related experiments are shown in Section IV-B.

IV. Experiments

We conduct experiments on three benchmark vehicle datasets, including Stanford Cars [23], CompCars [24] and CompCars Surveillance [24]. The detailed information of these datasets is provided in Table II. Samples from these datasets are shown in Fig. 3.

A. Implementation Details

Our method is implemented with PyTorch [25]. The two standard CNN architectures are first pre-trained on ImageNet dataset [26], later fine-tuned on the target datasets. For the original ResNet architecture, only the last FC layer is modified to adapt to each dataset. For all the experiments, the models

TABLE II
The statistics of three VMMR datasets. BBox stands for bounding box

Datasets	Category	Train	Test	BBox
Stanford Cars [23]	196	8,144	8,041	√
CompCars [24]	431	16,016	14,939	√
CompCars Surveillance [24]	281	31,148	13,333	

are trained using Stochastic Gradient Descent (SGD) with a weight decay of 0.0005 and a momentum of 0.9. Models are trained for 120 epochs. The initial learning rate is set to 0.01. After 40 epochs, the learning rate is reduced by a factor of 0.5 every 20 epochs. The image is resized to 224×224. Data augmentations such as random crop and horizontal flip are applied. For the last 40 epochs, images will be cropped with the help of the bounding box when training the model on the CompCars dataset. The weights of the convolutional layers in RAU are initialized by using the method presented in [8], biases are initialized to zero.

B. Ablation study

We evaluate the influence of two adjustable parameters of our model on the Stanford Cars dataset following a similar procedure presented in [11]. We conduct experiments to show the effect of the number of RAUs d and the number of attention masks k. To capture more target information contained in features of different scales, we need to deploy at least two RAUs to combine these features. The maximum number of RAUs that can be added is determined by the structure of the base model. For example, ResNet contains four layer groups, and three of them generate mid-level and high-level features, which are the main useful features for recognizing an object, since the information carried by the lower layers is very fundamental, and features of higher layers are more class-specific [27]. Therefore, the maximum value of d is 3 in our experiment. The value of k is set from 1 to 4.

The number of parameters and floating point operations (FLOPs) [28] are usually used to measure the computational complexity of deep learning models. The experimental results, including the accuracy and computational complexity of ResNet50-RAU with different d and k, are presented in Table III. We can observe that our model can achieve a better result than the previous state-of-the-art methods (shown in Table IV) when combining features extracted from feature maps of different scales. Moreover, when we increase k to 2 ($d = 2$), our model achieves the best recognition result 93.57%. In Table III, we can also observe that the computational complexity of the models has increased, but our models still have the ability to process images with real-time processing speed. For example, when running on a workstation with a GTX1080Ti GPU, the processing speed of ResNet50 ($d = 2, k = 2$) is 128 frames per second (FPS). For the next part of our experiments, we set d to 2 and k to 2 for ResNet50-RAU as these values give the best recognition result. Since ResNet50-RAU and ResNet101-RAU have a similar structure, we use the same parameter setting for ResNet101-RAU.

TABLE III
THE INFLUENCE OF THE NUMBER OF RAUS d, AND THE NUMBER OF ATTENTION MASKS k FOR EACH RAU ON THE STANFORD CARS DATASET

	ResNet50-RAU					
	$d = 2$			$d = 3$		
	A	P	F	A	P	F
$k = 1$	93.42	25.0	4.20	93.30	25.5	4.36
$k = 2$	**93.57**	26.0	4.28	93.40	27.0	4.59
$k = 3$	93.32	27.1	4.36	93.41	28.5	4.83
$k = 4$	93.26	28.1	4.44	93.35	30.1	5.06
baseline	ResNet50	91.78		P = 23.9	F = 4.12	

Abbreviations: A: Accuracy (%); P: The number of parameters ($\times 10^6$); F: The number of FLOPs ($\times 10^9$).

C. Comparison with State-of-the-Art Methods

1) Performance on Stanford Cars: We compare the proposed models with recent state-of-the-art methods. The results are listed in Table IV. In this table, the results of the *previous* part are cited from the original papers, and the results of *baseline* models and our models are derived from the experiments that are conducted based on the experimental setting we mentioned before. Without adopting the bounding box annotation at test time, our model ResNet101-RAU achieves 91.47% recognition accuracy, which outperforms the baseline ResNet101 with a clear margin (4.14% relative gains), and it is also a comparable result with M-CAN (91.5%), which adopted the Class Activation Map (CAM) to select object features [29]. RA-CNN [30] and MA-CNN [31] crop out the part regions of the object according to their attention modules, and these models are trained with images of size 448×448. These part regions and large scale images provide more detailed information, which allows these models to produce higher results of 92.5% and 92.8%, respectively. PA-CNN [9] adopts the co-segmentation and alignment methods and tests the model by adopting the bounding box annotation, which achieves 92.8% accuracy. Under a similar experiment setting, the baseline ResNet50 and ResNet101 achieve 91.78% and 92.45%, respectively. By adopting our attention selection method, the recognition accuracies are improved from 91.78% to 93.57% for ResNet50-RAU and from 92.45% to 93.81% for ResNet101-RAU. The evaluation shows that the proposed RAU advances the performance of ResNet for the VMMR task. The previously reported best recognition accuracy 93.1% is achieved by ResNet101-SWP [19] and FCAN [18], ResNet101-RAU achieves a better recognition result with 0.71% relative improvement.

2) Performance on CompCars: The recognition results on the CompCars dataset are summarized in Table V. The two baselines ResNet50 and ResNet101 obtain 96.73% and 97.41% accuracy, respectively. By adopting our RAU, the values of the accuracy of the proposed ResNet50-RAU and ResNet101-RAU are boosted to 97.60% and 97.84%, respectively. The improvement shows our models have a stable performance on recognizing vehicle models. The recently proposed ResNet101-SWP [19] achieves 97.6% accuracy on this dataset, ResNet101-RAU surpasses it by a small margin of 0.24%, achieving the best recognition result on the CompCars dataset.

TABLE IV
COMPARISON OF THE RECOGNITION RESULTS ON THE STANFORD CARS DATASET, $\surd$ MEANS TESTING WITH BOUNDING BOX ANNOTATION, N/A INDICATES THAT BOUNDING BOX ANNOTATION IS NOT USED

Type	Model	Test Anno.	Accuracy (%)
Previous	DVAN [10]	N/A	87.1
	WARN [11]	N/A	90.0
	BCNN (448) [17]	N/A	91.3
	M-CAN [32]	N/A	91.5
	RA-CNN (448) [30]	N/A	92.5
	MA-CNN (448) [31]	N/A	92.8
	ResNet50-SWP [19]	$\surd$	92.3
	PA-CNN [9]	$\surd$	92.8
	FCAN [18]	$\surd$	**93.1**
	ResNet101-SWP [19]	$\surd$	**93.1**
Baseline	ResNet50	N/A	86.78
	ResNet101	N/A	87.33
	ResNet50	$\surd$	91.78
	ResNet101	$\surd$	92.45
Ours	ResNet50-RAU	N/A	90.30
	ResNet101-RAU	N/A	91.47
	ResNet50-RAU	$\surd$	93.57
	ResNet101-RAU	$\surd$	**93.81**

TABLE V
COMPARISON OF THE RECOGNITION RESULTS ON THE COMPCARS DATASET

Type	Model	Accuracy (%)
Previous	AlexNet [24]	81.9
	OverFeat [24]	87.9
	GoogLeNet [24]	91.2
	ResNet50-SWP [19]	97.5
	ResNet101-SWP [19]	**97.6**
Baseline	ResNet50	96.73
	ResNet101	97.41
Ours	ResNet50-RAU	97.60
	ResNet101-RAU	**97.84**

3) Performance on CompCars Surveillance: For this dataset, we only trained our models 50 epochs. The initial learning rate is set to 0.001. The learning rate is divided by 0.5 at 30th and 40th epoch. We use this dataset to test the performance of our models in a practical environment. The recognition results on the CompCars Surveillance dataset are summarized in Table VI. A coarse-to-fine (CF) [33] method was proposed to iteratively extract global and local vehicle features, achieving 98.63% accuracy. Lightweight Convolutional Neural Network (LWCNN) [34] is a simplified VGG model and is trained with a combined training strategy. The lightweight VGG achieves 98.71% accuracy. FF-CMNET [35] uses two separate networks to extract the upper part and down part features of a vehicle, and it achieves 98.89% accuracy. Our proposed models ResNet50-RAU and ResNet101-RAU achieve results of 98.83% and 98.90%, respectively, which demonstrate our models can still obtain excellent results on the surveillance dataset.

V. CONCLUSION

In this paper, we propose the Recurrent Attention Unit (RAU) to enhance the performance of standard CNN architec-

TABLE VI
COMPARISON OF THE RECOGNITION RESULTS ON THE COMPCARS SURVEILLANCE DATASET

Type	Model	Accuracy (%)
Previous	AlexNet [24]	98.0
	OverFeat [24]	98.3
	GoogLeNet [24]	98.4
	CF [33]	98.63
	LWCNN [34]	98.71
	FF-CMNET [35]	**98.89**
Ours	ResNet50-RAU	98.83
	ResNet101-RAU	**98.90**

tures for the VMMR task. RAU can help the model extract the most discriminative features of an object. RAU is easy to be added to most standard CNN architectures and can be trained end-to-end. Extensive experimental results demonstrate the effectiveness of fusing prominent features of different scales. With the help of the Vehicle-to-Everything (V2X) technology, the accurately identified vehicle information can be transmitted to the agents, which will help the police find the criminal and increase the region's security. In the future, we will try to simplify the structure of RAU and reduce the size of the model, so that the proposed models can have excellent performance under different hardware conditions. We will also test the performance of RAU when combining it with other models.

REFERENCES

[1] A. N. Assuncao, F. O. de Paula, and R. A. Oliveira, "Methodology to events identification in vehicles using statistical process control on steering wheel data," in *Proc. ACM MobiWac*, 2015, pp. 1–4.

[2] H. Huang and S. Lin, "Widet: Wi-fi based device-free passive person detection with deep convolutional neural networks," in *Proc. ACM MSWiM*, 2018, pp. 53–60.

[3] IEEE Standards Association, "Ieee 802.11p-2010 - ieee standard for information technology– local and metropolitan area networks– specific requirements– part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 6: Wireless access in vehicular environments," [Online]. Available: https://standards.ieee.org/standard/802_11p-2010.html, accessed on: Nov, 2019.

[4] SAE International, "Dedicated short range communications (dsrc) message set dictionary J2735," [Online]. Available: https://www.sae.org/standards/content/j2735_201603/, accessed on: Nov, 2019.

[5] T. Begin, A. Busson, I. Guérin Lassous, and A. Boukerche, "Video on demand in ieee 802.11 p-based vehicular networks: Analysis and dimensioning," in *Proc. ACM MSWiM*, 2018, pp. 303–310.

[6] A. Díaz Zayas, D. Rico, B. García, and P. Merino, "A coordination framework for experimentation in 5g testbeds: Urllc as use case," in *Proc. ACM MobiWac*, 2019, p. 71–79.

[7] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," [Online]. Available: https://arxiv.org/abs/1409.1556, accessed on: Nov., 2017.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. ECCV*, 2016, pp. 770–778.

[9] J. Krause, H. Jin, J. Yang, and L. Fei-Fei, "Fine-grained recognition without part annotations," in *Proc. IEEE CVPR*, Jun 2015, pp. 5546–5555.

[10] B. Zhao, X. Wu, J. Feng, Q. Peng, and S. Yan, "Diversified visual attention networks for fine-grained object classification," *IEEE Trans. Multimedia*, vol. 19, no. 6, pp. 1245–1256, 2017.

[11] P. Rodríguez, J. M. Gonfaus, G. Cucurull, F. XavierRoca, and J. Gonzalez, "Attend and rectify: a gated attention mechanism for fine-grained recovery," in *Proc. ECCV*, 2018, pp. 349–364.

[12] S. Zagoruyko and N. Komodakis, "Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer," [Online]. Available: http://arxiv.org/abs/1612.03928, accessed on: Nov., 2018.

[13] A. Boukerche, A. J. Siddiqui, and A. Mammeri, "Automated vehicle detection and classification: Models, methods, and techniques," *ACM Comput. Surv.*, vol. 50, no. 5, p. 62, 2017.

[14] R. Desimone and J. Duncan, "Neural mechanisms of selective visual attention," *Annu. Rev. Neurosci.*, vol. 18, no. 1, pp. 193–222, 1995.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[16] L. Wu, Y. Wang, X. Li, and J. Gao, "Deep attention-based spatially recursive networks for fine-grained visual recognition," *IEEE Trans. Cybern.*, vol. 49, no. 5, pp. 1791–1802, 2019.

[17] T.-Y. Lin, A. RoyChowdhury, and S. Maji, "Bilinear cnn models for fine-grained visual recognition," in *Proc. IEEE ICCV*, 2015, pp. 1449–1457.

[18] X. Liu, T. Xia, J. Wang, Y. Yang, F. Zhou, and Y. Lin, "Fully convolutional attention networks for fine-grained recognition," [Online]. Available: https://arxiv.org/abs/1603.06765, accessed on: Sep., 2018.

[19] Q. Hu, H. Wang, T. Li, and C. Shen, "Deep cnns with spatially weighted pooling for fine-grained car recognition," *IEEE Trans. Intell. Transp. Syst*, vol. 18, no. 11, pp. 3147–3156, 2017.

[20] S. Zagoruyko and N. Komodakis, "Wide residual networks," [Online]. Available: http://arxiv.org/abs/1605.07146, accessed on: Nov., 2018.

[21] D. Long, R. Zhang, and Y. Mao, "Prototypical recurrent unit," *Neurocomputing*, vol. 311, pp. 146–154, 2018.

[22] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," [Online]. Available: https://arxiv.org/abs/1406.1078, accessed on: Nov., 2018.

[23] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3d object representations for fine-grained categorization," in *Proc. IEEE ICCVW*, 2013, pp. 554–561.

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

[25] A. Paszke, G. Chanan, Z. Lin, S. Gross, E. Yang, L. Antiga, and Z. Devito, "Automatic differentiation in pytorch," in *Proc. NIPS*, 2017, pp. 1–4.

[26] R. Socher, J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE CVPR*, 2009, pp. 248–255.

[27] Y. Wang, V. I. Morariu, and L. S. Davis, "Learning a discriminative filter bank within a cnn for fine-grained recognition," in *Proc. IEEE CVPR*, 2018, pp. 4148–4157.

[28] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient inference," [Online]. Available: http://arxiv.org/abs/1611.06440, accessed on: May, 2020.

[29] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE CVPR*, 2016, pp. 2921–2929.

[30] J. Fu, H. Zheng, and T. Mei, "Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition," in *Proc. IEEE CVPR*, 2017, pp. 4476–4484.

[31] H. Zheng, J. Fu, T. Mei, and J. Luo, "Learning multi-attention convolutional neural network for fine-grained image recognition," in *Proc. IEEE ICCV*, vol. 6, 2017.

[32] W. Zhong, L. Jiang, T. Zhang, J. Ji, and H. Xiong, "A multi-part convolutional attention network for fine-grained image recognition," in *Proc. ICPR*, 2018, pp. 1857–1862.

[33] J. Fang, Y. Zhou, Y. Yu, and S. Du, "Fine-grained vehicle model recognition using a coarse-to-fine convolutional neural network architecture," *IEEE Trans. Intell. Transp. Syst*, vol. 18, no. 7, pp. 1782–1792, 2017.

[34] Q. Zhang, L. Zhuo, S. Zhang, J. Li, H. Zhang, and X. Li, "Fine-grained vehicle recognition using lightweight convolutional neural network with combined learning strategy," in *Proc. BigMM*, 2018, pp. 1–5.

[35] Y. Yu, Q. Jin, and C. W. Chen, "Ff-cmnet: A cnn-based model for fine-grained classification of car models based on feature fusion," in *Proc. IEEE ICME*, 2018, pp. 1–6.