2018 24th International Conference on Pattern Recognition (ICPR)
Beijing, China, August 20-24, 2018

Attention-based Neural Network for Traffic Sign Detection

Jing Zhang, Le Hui, Jianfeng Lu, Yuhua Zhu
School of Computer Science and Engineering
Nanjing University of Science and Technology
Nanjing, China
Email: {jing.zhang, le.hui, lujf, zhuyh}@njust.edu.cn

***Abstract*—Existing object detection pipelines can show superior performance for large objects with high resolution but fail to detect very small objects such as traffic signs. So, detecting traffic signs is a proverbially challenging problem. In this paper, we propose a novel end-to-end architecture that improves small object detection by combining Faster R-CNN with the attention mechanism. Specifically, we focus on channel-wise features and utilize the attention mechanism to enhance the feature responses by explicitly modeling the interdependencies between channel-wise features. Finally, the regression of bounding boxes and the classification of traffic signs are generated after selecting the discriminative features by the attention mechanism. Extensive evaluations of the largest traffic sign dataset demonstrate that the attention mechanism improves the performance of detecting objects, especially the small targets. For traffic sign detection task, our method achieves better performance compared with many state-of-the-art approaches on the largest traffic sign detection dataset, Tsinghua-Tencent 100K.**

I. Introduction

Recent advances in object detection are owed to the success of the convolutional neural network (CNN). Through a series of convolutions interleaved with non-linearities and down-sampling, CNN is capable of learning complex hierarchical feature representations of images. The variants of the CNN based approaches to object detection can be divided into two types. One of those methods proposes region proposals in the first stage followed by a second stage of classifying and refining these region proposals. The other treats object detection as a single shot problem, which straight localizes and classifies the object from image pixels. Existing object detection pipelines show superior performance for large objects, but the reduction in resolution often leads to failure to detect small objects. Unfortunately, small objects are very common in many real-world applications.

Traffic sign detection is an inevitable and arduous task in many real-world applications. In particularly, accurately localizing and classifying traffic signs is highly related to the safety in the field of advanced driver assistance systems (ADAS). However, due to environmental complexity, occluded region, small-scale property and so forth, this task is more challenging than normal object detection.

Over the past few years, some efforts [3], [4], [5], [6], [7] have been devoted to small object detection problems via two ways. The first way handles objects of small size independently by increasing the size of the input image and then producing high-resolution feature maps for detection, such methods probably have the higher recall in general, but they may suffer from high computation cost. Extra training and testing time will bring the high burden to real-time applications. Another way focuses on developing network variants to generate multi-scale representation and can improve the capacity of the model, but the computational burden is still high for real-time applications.

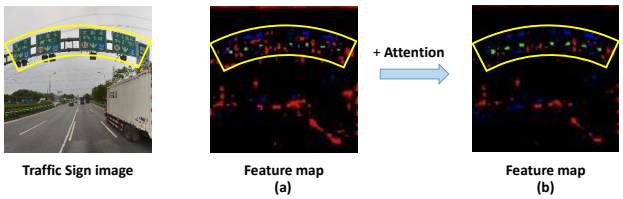

Fig. 1. The visualization of features learned from Faster R-CNN [1] with non-attention model vs. attention model. Specifically, we extract the features of the conv5 layer of VGG [2]. (a) the features learned from the non-attention model, and (b) the features learned from attention model. The **yellow** areas represent the regions of traffic signs. The **green** spots represent the responses of traffic signs.

Notwithstanding their demonstrated success, the detection of small targets also needs to be strengthened. After a careful examination of existing detection networks, we argue that a simple increase in input scale will bring an extra time burden for training and testing, and the complex network structures are hard to train for this problem. For addressing the problems of the traffic sign detection, we introduce the attention mechanism to improve the response of the network to small targets without significantly increasing the network complexity. In order to capture the channel-wise dependencies to generate the discriminative feature for object detection, we utilize attention mechanism to enhance the feature responses by explicitly modeling the interdependencies between channel-wise features. By selecting more discriminative features based on the attention mechanism, the regression of bounding boxes and the classification of traffic signs have achieved better results.

In this paper, we adopt VGG [2] and ResNet [8] as feature extraction networks. And we validate our proposed attention-based network on the largest traffic sign dataset Tsinghua-Tencent 100K. The experimental results show that our method explicitly improves the detection performance of the targets at

978-1-5386-3788-3/18/$31.00 ©2018 IEEE

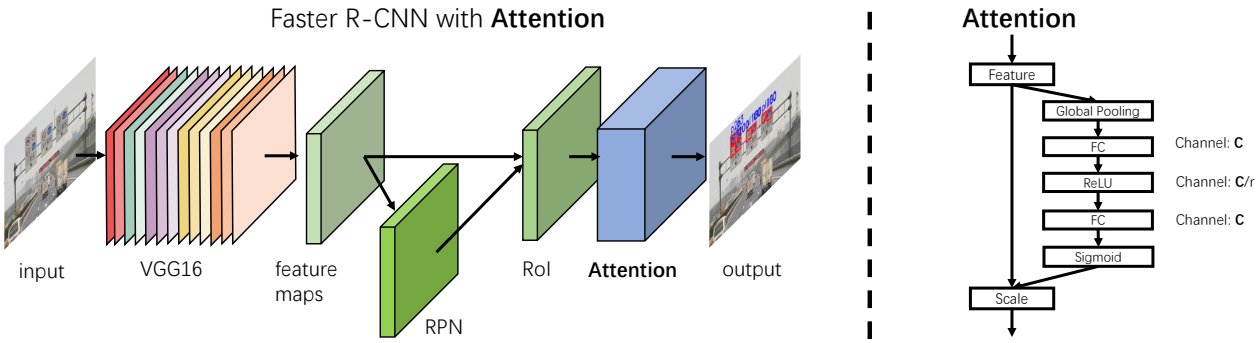

Fig. 2. An overview of our framework for attention-based traffic sign detection network. **Left:** The core structure of Faster R-CNN [1] object detection network by using VGG-16 [2] as the feature extraction network. Note that we apply the attention to the fully connected layers (In practice, we find that the attention applies to fully connected layer obtain the best results) after ROI pooling, and then regress the bounding boxes and classify the detected targets. **Right:** The detailed structure of attention module, which uses global information to selectively emphasize informative features and suppresses less useful ones.

different scales, especially for the small-scale targets. More notably, the results are superior to that of state-of-the-art by using the VGG-16-s8 model with the attention mechanism.

II. Related Work

A. General Object Detection

Over the past decade, Convolutional Neural Network (CNN) has been successfully applied to various image analysis tasks and gradually become one of the most powerful machines learning approaches. Especially, CNN has achieved state-of-the-art results for a wide range of challenging tasks such as object detection [9], [10], [11], [1], object recognition [12], [2], [8], [13], natural language processing [14], [15], etc. Girshick et al. [10] proposed a simple and scalable detection algorithm that greatly improved the performance of object detection, namely R-CNN which revived this research domain. More notably, it is the earliest end-to-end convolutional network to localize and classify the object simultaneously, and get impressive results on the PASCAL VOC [16] dataset. However, the computation of R-CNN is expensive so that a series of object detection networks were proposed to accelerate the speed of detection such as Fast R-CNN [11], Faster R-CNN [1], SSD [4], YOLO [17], etc., and improve the performance of detection and classification at the same time. All these networks can well cover conventional target detection and obtain the state-of-the-art at a wide range of detection tasks.

B. Traffic Sign Detection

Traffic sign detection has been a popular problem for intelligent vehicles. With the rapid development of unmanned technologies, the detection of traffic sign plays an important role. In the traditional methods, Wang et al. [18] proposed a HOG (histogram of oriented) [19] based coarse-to-fine sliding window detection framework. Escalera et al. [20] used color threshold and shape analysis for traffic sign detection at first, and then used a neural network for classification. However, there are still some problems about missing detection and wrong classification. Recently, deep learning based approaches have attracted increased attention from researchers of traffic sign detection. Zhu et al. [21] trained a fully convolutional neural network to simultaneously localize and classify traffic signs in the wild. Although this approach brought the great improvement over the traditional methods, the problem of missing detection and wrong classification persists for small-scale targets. To tackle this intractable issue, Liu et al. [22] proposed a Perceptual Generative Adversarial Network model for lifting representations of small traffic signs and the results were superior to that of all the state-of-the-art methods, but it had to face the heavy burden of training.

C. Attention and Gating Mechanisms

Attention can be broadly viewed as a tool to assign the available processing resources towards the most informative components of an input signal. It has been significantly interested in the last several years as a powerful addition to deep neural networks [23], [24]. Attention has shown its ability to improve the performance across a wide range of tasks, from localization and understanding in images [25], [26] to sequence-based models [27], [28]. It is worth noting that recent works have researched how to model spatial dependence [7], [29] better and incorporate spatial attention [26] by trial and error. Latterly, Hu et al. [30] proposed SENet that uses global information to selectively emphasize informative features and suppress less useful ones. It embeds the global distribution of channel-wise feature responses and enables information from the global receptive field of the network to be utilized by its lower layers. Our key observation is that the SENet proposed in [30] can be applied to object detection. Thus, we use this structure to help traffic sign detection task.

III. Proposed Approach

An overview of the proposed network architecture is shown in Figure 2, which is specially designed for traffic sign detection task. In this section, we will first present the technical parts of our network, and then the details of detection network and training strategy are given.

A. Attention Module

An attention module is actually a computational unit which can be constructed for any given transform function $\mathbf{F_{tr}}$: $\mathbf{X} \rightarrow \mathbf{Y}, \mathbf{X} \in \mathbb{R}^{W^{'} \times H^{'} \times C^{'}}, \mathbf{Y} \in \mathbb{R}^{W \times H \times C}$. Let $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_C]$ denotes the learned set of kernels, where $\mathbf{v}_c$ refers to the parameters of the c-th filter. We define that

$$\mathbf{y}_c = \mathbf{v}_c * \mathbf{X} = \sum\nolimits_{i=1}^{C^{'}} \mathbf{v}_c^i * \mathbf{x}^i, \tag{1}$$

where $*$ denotes convolution, $\mathbf{v}_c = [\mathbf{v}_c^1, \mathbf{v}_c^2, ..., \mathbf{v}_c^{C^{'}}]$ and $\mathbf{X} = [\mathbf{x}^1, \mathbf{x}^2, ..., \mathbf{x}^{C^{'}}]$. Note that $\mathbf{v}_c^i$ is a 2D spatial kernel, and therefore represents a single channel of $\mathbf{v}_c$, which acts as the corresponding channel of $\mathbf{X}$. That is, the outputs of $\mathbf{F_{tr}}$ as $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_C,]$. The channel dependencies are implicitly embedded in $\mathbf{v}_c$ due to the output is produced by a summation through all channels.

As shown in Figure 2, we first utilize global average pooling to squeeze global information to channel-wise statics as

$$G = \frac{1}{W \times H} \sum\nolimits_{i=1}^{W} \sum\nolimits_{j=1}^{H} y(i, j), \tag{2}$$

and then a simple ReLU activation function and a sigmoid activation function are employed to fully capture channel-wise dependencies as

$$S = \sigma(\delta(G)), \tag{3}$$

where δ is the ReLU function and σ is the sigmoid function. The activations act as channel weights adapted to the input specific descriptor. The final output of the block is obtained by rescaling the transformation output Y:

$$\widetilde{\mathbf{x}}_c = \mathbf{F}_{scale}\left(\mathbf{y}_c \cdot s_c\right) = s_c \cdot \mathbf{y}_c, \tag{4}$$

where $\widetilde{\mathbf{X}} = [\widetilde{\mathbf{x}}_1, \widetilde{\mathbf{x}}_2, \cdots, \widetilde{\mathbf{x}}_c]$ and $\mathbf{F}_{scale}$ refers to channel-wise multiplication between the scalar s_c and feature map $\mathbf{y}_c \in \mathbb{R}^{W \times H}$. In this paper, the attention intrinsically introduce dynamics conditioned on the input, helping to boost feature discriminability. In practice, we find that the effect of attention on the last two fully connected layers is the best.

B. Network Structure

In general, a discriminative feature will explicitly boost the separability of the similar samples and achieve better performance on a variety of tasks. The traffic sign detection has always been a difficult task due to their small size when far away and complex environment, etc. To alleviate this problem, we propose to combine Faster R-CNN with the attention mechanism to improve the detection performance of objects at different scales, especially the small scale.

As shown in Figure 2, based on the Faster R-CNN, we apply the attention mechanism to the output of the ROI layer to boost the detection and classification performance.

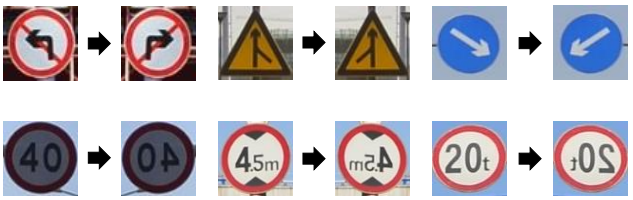

Fig. 3. The **ERROR** samples of horizontal flip augmentation of Tsinghua-Tencent 100K [21]. Therefore, the horizontal flip is fatal to traffic sign detection.

C. Training Strategy

The training of the proposed network is divided into two stages. First, we train the baseline models (including VGG-M-1024 [31], VGG-16 [2] and ResNet-50 [8]) by utilizing the pre-trained parameters on the ImageNet dataset [32]. Due to the difference between traffic sign dataset and general dataset, the feasible way for the above purpose is to learn the appropriate initialization parameters that baseline networks can adapt well to the traffic sign datasets. That is why there is no attention module in training model at the first stage. Specifically, we iterator 50K at first stage. Next, we train the attention-based model by utilizing the pre-trained model obtained at the first stage for additional 30K iterations. For a fair comparison, we add an additional 30K iterations when training the baseline models. Thus, both baseline models and attention-based models are 80K iterations.

Moreover, there are also many important tricks for training, which are the significant factors for experimental results. In deep learning, we know that the more the training data, the better the final result in general. It is generally true that the data augmentation will bring a steady boost to the result. However, the augmentation of the horizontal flip is fatal to the traffic sign datasets. As shown in Figure 3, the traffic sign usually has horizontal symmetry and they represent different implication and category, respectively. Once we use horizontal flip during training, the labels corresponding to those samples that cannot be horizontally flip will get confused. Our results show that the utilization of horizontal flip will reduce the overall performance by two percent. Therefore, none of our experiments adopt horizontal augmentation.

IV. Experiments

A. Datasets and Evaluation Metrics

The Tsinghua-Tencent 100K [21] is a new Chinese traffic sign dataset, which is the largest one in the world according to our knowledge. The images are with the resolution of 2,048 $\times$ 2,048 and cover various conditions in the real world, such as illumination, shelter, roads, and weather. It contains 30,000 traffic sign instances in 100 classes. Each instance is annotated with a classification label, a bounding box, and a pixel mask. Similar to [21], we ignore the classes whose instances are less than 100 and there are 45 classes left. The performance is evaluated by the same detection metrics as for the Microsoft COCO benchmark [33]. Specifically, we use the standard mean

TABLE I

THE **AP** (%) OF OUR METHOD COMPARED TO BASELINE ON TSINGHUA-TENCENT 100K [21]. THE NUMBERS IN **BOLD** ARE THE SUPERIOR RESULTS WHEN COMPARED WITH CORRESPONDING MODELS.

Class	p3	p5	p6	pg	ph4	ph4.5	ph5	pl100	pl120	pl20	pl30	pl40	pl5	pl50	pl60
VGG-M-1024	63.7	84.9	54.7	91.2	70.3	81.0	42.0	90.9	**88.5**	51.9	64.5	73.4	78.5	62.2	73.2
VGG-M-1024 + Attn (**ours**)	**73.1**	**87.5**	**61.8**	**92.0**	**75.9**	**81.7**	**55.1**	**92.3**	87.8	**59.6**	**69.8**	**78.0**	**78.7**	**71.1**	**78.0**
ResNet-50	91.1	**92.0**	**79.4**	82.5	**72.6**	86.3	56.7	**95.4**	94.7	56.5	78.7	84.2	**83.2**	79.3	81.3
ResNet-50 + Attn (**ours**)	**92.8**	91.6	78.9	**89.7**	**72.6**	**86.7**	**67.8**	94.7	**97.2**	**58.2**	**80.3**	**84.6**	**83.2**	**80.8**	**82.0**
VGG-16	**87.6**	**92.1**	**85.8**	90.5	**75.9**	**87.1**	**67.1**	**96.2**	92.9	74.5	84.4	88.2	83.4	79.3	87.7
VGG-16 + Attn (**ours**)	86.5	91.7	83.3	**91.6**	73.9	84.6	66.9	94.2	**96.6**	**83.3**	**87.1**	**89.9**	**89.4**	**82.6**	**88.7**
VGG-16-s8	**94.6**	94.5	91.3	**95.5**	89.7	87.9	81.3	99.2	98.7	88.0	92.9	94.9	93.7	94.3	94.0
VGG-16-s8 + Attn (**ours**)	92.6	**95.6**	**92.7**	94.7	**90.4**	**88.1**	**82.9**	**99.3**	**98.9**	**88.7**	**93.9**	**95.1**	**93.9**	**94.4**	**95.5**

Class	pl70	pl80	pm20	pm30	pm55	pn	pne	po	pr40	w13	w32	w55	w57	w59	wo
VGG-M-1024	68.7	71.0	70.4	64.9	79.4	75.1	81.7	60.7	96.1	43.3	62.2	60.5	75.7	63.6	39.4
VGG-M-1024 + Attn (**ours**)	**74.9**	**76.7**	**74.0**	**65.4**	**84.5**	**77.0**	**83.3**	**63.1**	**97.4**	**55.9**	**63.4**	**67.0**	**81.2**	**63.9**	**47.3**
ResNet-50	**78.1**	81.7	78.7	**81.8**	89.0	83.6	87.6	67.7	98.7	66.7	**78.6**	75.4	83.5	74.6	**49.3**
ResNet-50 + Attn (**ours**)	76.1	**84.1**	**79.4**	77.5	**91.4**	**86.2**	**88.2**	**76.0**	**98.8**	**74.8**	77.2	**77.2**	**86.8**	**80.3**	44.4
VGG-16	84.8	88.9	84.6	83.1	**93.2**	81.1	91.8	72.5	99.0	55.1	83.9	81.5	**90.4**	70.4	51.8
VGG-16 + Attn (**ours**)	**85.6**	**91.5**	**91.3**	**89.1**	93.1	**84.9**	**93.3**	**76.9**	**99.3**	**67.1**	**88.3**	**86.3**	89.0	**80.3**	**56.5**
VGG-16-s8	91.6	95.7	90.7	93.3	94.5	**94.9**	**96.5**	85.8	**99.8**	80.7	90.6	94.8	94.6	89.7	59.4
VGG-16-s8 + Attn (**ours**)	**92.0**	**96.3**	**91.6**	**93.5**	**94.6**	94.7	96.4	**86.2**	**99.8**	**88.0**	**93.4**	**95.8**	**95.3**	**90.4**	**64.8**

Class	i2	i4	i5	il100	il60	il180	io	ip	p10	p11	p12	p19	p23	p26	p27
VGG-M-1024	65.9	73.8	81.1	80.1	82.1	80.3	75.8	70.0	62.1	57.7	53.7	70.6	83.7	77.4	86.2
VGG-M-1024 + Attn (**ours**)	**68.2**	**78.8**	**85.1**	**82.1**	**87.7**	**87.3**	**78.6**	**70.9**	**66.7**	**67.6**	**65.1**	**77.5**	**88.3**	**81.8**	**88.9**
ResNet-50	75.9	87.7	88.4	**88.3**	91.1	92.5	81.8	77.6	74.6	77.8	81.9	93.7	**92.3**	84.8	**95.7**
ResNet-50 + Attn (**ours**)	**82.5**	**88.9**	**91.0**	87.5	**94.2**	**94.3**	**82.7**	**81.5**	**77.4**	**83.5**	**82.0**	**95.9**	**92.3**	**89.0**	93.2
VGG-16	**81.7**	**90.4**	**92.1**	92.0	93.1	91.2	82.9	**85.4**	**84.6**	73.2	76.1	93.6	**94.3**	88.7	91.6
VGG-16 + Attn (**ours**)	79.5	88.3	90.3	**94.6**	**98.8**	**95.9**	**84.0**	83.5	83.1	**85.0**	**87.0**	**95.9**	**94.3**	**90.4**	**96.3**
VGG-16-s8	**88.5**	**96.1**	**96.0**	**99.9**	**98.6**	**98.8**	**90.9**	90.4	**87.0**	94.4	96.9	97.0	94.9	**94.5**	99.6
VGG-16-s8 + Attn (**ours**)	87.4	95.0	95.2	99.8	98.2	98.3	90.8	**91.4**	85.6	**94.9**	**97.0**	**100.0**	**97.2**	94.4	**99.7**

average precision (mAP) at 0.5 IoU threshold as the evaluation metric. We report the detection performance on different size of objects, including small objects (area $< 32 \times 32$ pixels), medium objects ($32 \times 32 <$ area $< 96 \times 96$) and large objects (area $> 96 \times 96$). The number of instances corresponding to the three kinds of objects is 3270, 3829 and 599, respectively. In addition to comparison with [21], we also report recall and accuracy of different scales.

TABLE II

COMPARISONS OF DETECTION **mAP** (%) OF ALL CATEGORIES UNDER DIFFERENT MODELS ON TSINGHUA-TENCENT 100K [21] BY USING FASTER R-CNN.

Model	Baseline	+Attn (ours)
VGG-M-1024	70.8	**75.4**
ResNet-50	81.3	**83.5**
VGG-16	84.4	**86.9**
VGG-16-s8	92.6	**93.7**

B. Feature Extraction Networks

We mainly adopt two networks for feature extraction, namely, VGG [2] and ResNet [8] .

- **VGG**. Simonyan et al. [2] introduced the deeper convolutional networks for large-scale image recognition. Since VGG employs smaller convolution kernels and smaller strides, the accuracy of this network is higher. Due to the high accuracy of the network, VGG is often used as the underlying network in other network frameworks. We adopt medium VGG-M-1024 network and large VGG-16 network for feature extraction in the experiments.
- **ResNet.** Residual Network is first proposed by He et al. [8]. Fleetly, it has been extended to various fields and has obtained a lot of new state-of-the-art. Extreme depth makes ResNet possess a very strong representation ability. As a result, ResNet is usually nested as a basic network in a variety of different tasks. Therefore, in this paper, we adopt ResNet-50 as one of the feature extraction networks for Faster R-CNN.

C. Implementation Details

For traffic sign detection, we use the pre-trained VGG-M-1024, VGG-16 and ResNet-50 adopted in [21], [34], [22] to initialize our network respectively. The training strategy has been described in the proposed approach section.

The whole network is trained with Stochastic Gradient Descent (SGD) with the momentum of 0.9, and weight decay of 0.0005 on a single NVIDIA GeForce GTX TITAN X GPU with 12GB memory. For the dataset of Tsinghua-Tencent 100K [21], the training parameters are as follows: mini-batch size is 256 for 80K iterations, and the initial learning rate is set to 0.001, in addition, the size of the original image is adjusted from $2{,}048 \times 2{,}048$ to $1{,}600 \times 1{,}600$ for the feature extraction networks. In the structure of VGG-16, the localization error is one kind of the main errors for the detection due to the small size of the traffic sign. As a result, we remove the pool4 layer and use the stride of 8 to reduce localization error as VGG-16-s8. Except for the VGG-16-s8, the structure of VGG-M-1024, VGG-16 and ResNet-50 has not been changed. For testing,

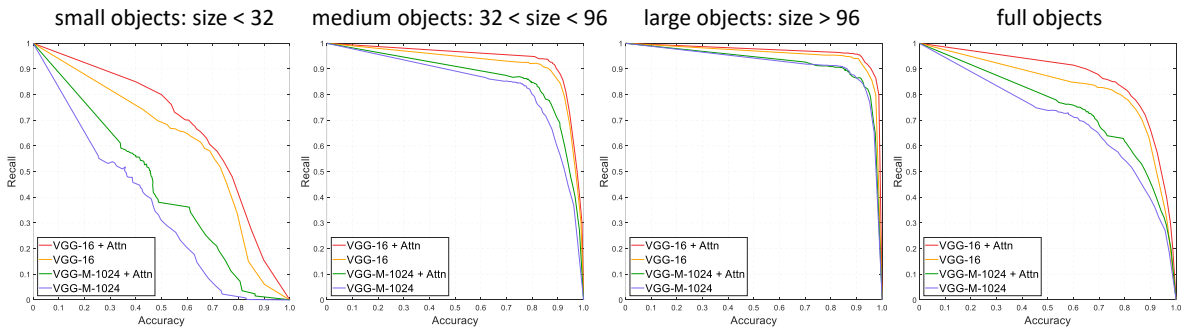

Fig. 4. Objects proposal results for traffic-signs for various object location methods on Tsinghua-Tencent 100K [21], for small, medium, large and full signs.

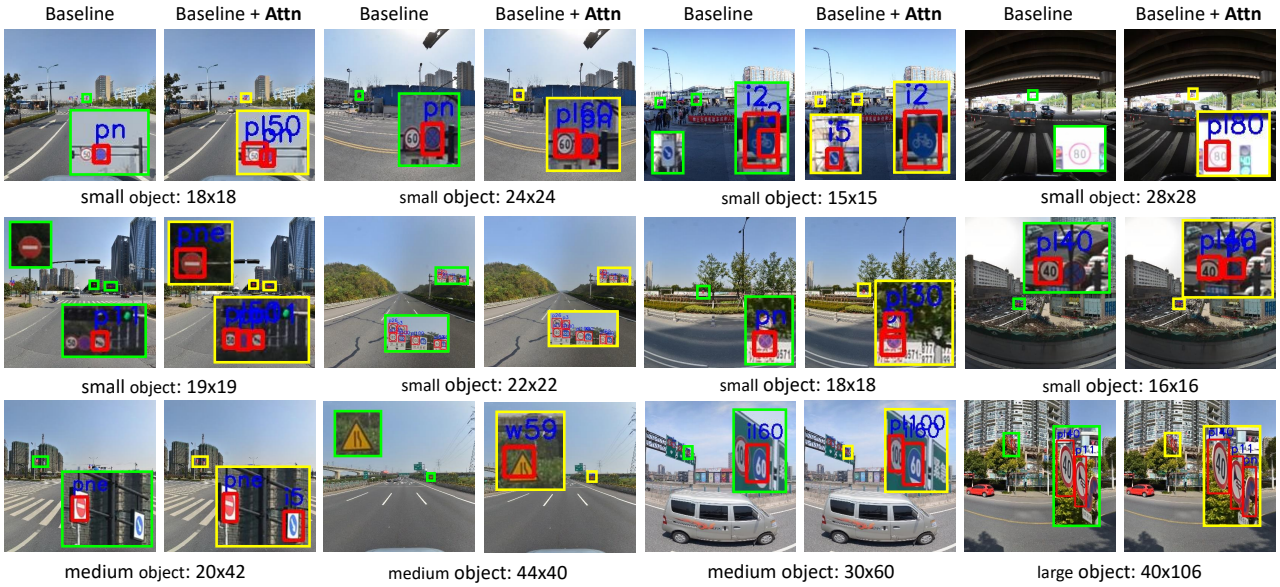

Fig. 5. Detection results of baseline and the proposed method on Tsinghua-Tencent 100K [21]. The proposed attention based network can successfully detect most small-size traffic signs which the baseline method has missed. Best viewed in color boxes.

on average, the attention based Faster R-CNN processes one image within 0.3 seconds (excluding object proposal time).

D. Performance Comparison

In order to evaluate the performance of the proposed model on the traffic sign detection, we adopt the dataset of Tsinghua-Tencent 100K, which is the most representative and largest dataset and covers a wide range of targets at different scales, and then we compare our performance with that of state-of-the-art models.

We adopt three widely used convolutional network architectures whose parameters are pre-trained on ImageNet as feature extractors for Faster R-CNN including VGG-M-1024, VGG-16, ResNet-50. Table I provides the comparison of our attention-based network with the corresponding baseline networks. To highlight general trends, we mark the numbers in **bold** to emphasize those results that are best. By applying attention mechanism to those networks, the overall performance of each network by mAP has been greatly raised by nearly 2%, which can be found in Table II. Notably, as shown in Table I, there are some categories even improved by nearly ∼5% such as i4, p6, w13, etc.

TABLE III
COMPARISONS OF DETECTION **Recall** AND **Accuracy** IN DIFFERENT SCALES WITH SEVERAL APPROACHES ON TSINGHUA-TENCENT 100K [21]. (R): RECALL, (A): ACCURACY.

Object size	Small	Medium	Large	All
Zhu et al. [21] (R)	0.87	0.94	0.88	-
Zhu et al. [21] (A)	0.82	0.91	0.91	-
Li et al. [22] (R)	0.89	-	-	0.93
Li et al. [22] (A)	0.84	-	-	0.88
VGG-16 (R)	0.55	0.88	0.94	0.76
VGG-16 + Attn (R) (**ours**)	**0.58**	**0.89**	**0.94**	**0.78**
VGG-16 (A)	0.70	0.88	0.89	0.82
VGG-16 + Attn (A) (**ours**)	**0.72**	**0.90**	**0.92**	**0.84**
VGG-16-s8 (R)	0.82	0.96	**0.95**	0.91
VGG-16-s8 + Attn (R) (**ours**)	**0.85**	**0.96**	0.94	**0.92**
VGG-16-s8 (A)	0.82	0.93	0.90	0.89
VGG-16-s8 + Attn (A) (**ours**)	**0.86**	**0.95**	**0.92**	**0.91**

Moreover, as shown in Table III, we present the average recall (R) and accuracy (A) of each model at three different scales for a fair comparison with the state-of-the-art. Note that we keep two decimal places of all model reported by us. From the overall point of view of Table III, our proposed approaches are all superior to the corresponding baselines. We obtain the best accuracies of 0.86, 0.95 and 0.94 on the three different

scales, respectively. Finally, from the whole, the results of our VGG-16-s8 model are superior to that of the state-of-the-art on accuracy.

E. The Effectiveness of Attention

To verify the superiority of the attention mechanism in detecting objects of different sizes, especially the small targets (area $< 32 \times 32$ pixels), we compare our method with several baseline models also.

More comparisons of accuracy-recall curves in terms of different object sizes are provided in Figure 4, which can further demonstrate the effectiveness of the proposed attention-based network. This clearly indicates that our approach outperforms Faster R-CNN, especially when traffic signs are in small scale. In addition, we also note that there is an improvement for the medium and the large objects detection, although this is a slight promotion when compared with small object detection, which further manifests the effectiveness of our approach.

In order to see the trend more clearly, we provide the detection comparison between baseline and the proposed model at different scales under different networks in Figure 5. It can be seen clearly that the results with the attention module bring a steady improvement over the baseline results. Especially, the VGG-16 with attention module greatly boosts the detection performance for small targets. More intuitively, in Figure 1, we visualize the feature map of the network with non-attention model vs. attention model. The target is explicitly emphasized by attention mechanisms. The attention module further uses global information to selectively emphasize informative features and suppress less useful ones. The rich experiments manifest that our attention module actually boosts the performance of the original network.

V. Conclusion

In this paper, we introduced a novel network architecture for traffic sign detection by combining the Faster R-CNN with the attention mechanism. The key idea is to enhance the response to the feature of small-scale targets with a channel-wise attention without causing difficulties for training. The experimental results and the visualization can further indicate that the proposed network actually improves small-scale traffic sign detection. Notably, the VGG-16-s8 with our attention module can obtain better performance than the existing networks on Tsinghua-Tencent 100K dataset.

ACKNOWLEDGMENT

This work was supported by the National Key R&D Program of China Contract No 2017YFB1300205. We are thankful to anonymous reviewers for their constructive comments.

References

[1] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *NIPS*, 2015.

[2] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[3] X. Chen, K. Kundu, Y. Zhu, A. G. Berneshawi, H. Ma, S. Fidler, and R. Urtasun, "3d object proposals for accurate object class detection," in *NIPS*, 2015.

[4] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *ECCV*, 2016.

[5] F. Yang, W. Choi, and Y. Lin, "Exploit all the layers: Fast and accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers," in *CVPR*, 2016.

[6] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *CVPR*, 2015.

[7] S. Bell, C. Lawrence Zitnick, K. Bala, and R. Girshick, "Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks," in *CVPR*, 2016.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[9] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *NIPS*, 2013.

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *CVPR*, 2014.

[11] R. Girshick, "Fast r-cnn," in *ICCV*, 2015.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012.

[13] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten, "Densely connected convolutional networks," *arXiv preprint arXiv:1608.06993*, 2016.

[14] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.

[15] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *INTERSPEECH*, 2014.

[16] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[17] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *CVPR*, 2016.

[18] G. Wang, G. Ren, Z. Wu, Y. Zhao, and L. Jiang, "A robust, coarse-to-fine traffic sign detection method," in *IJCNN*, 2013.

[19] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.

[20] A. De La Escalera, L. E. Moreno, M. A. Salichs, and J. M. Armingol, "Road traffic sign detection and classification," *IEEE Transactions on Industrial Electronics*, vol. 44, no. 6, pp. 848–859, 1997.

[21] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-sign detection and classification in the wild," in *CVPR*, 2016.

[22] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan, "Perceptual generative adversarial networks for small object detection," in *ICCV*, 2017.

[23] H. Larochelle and G. E. Hinton, "Learning to combine foveal glimpses with a third-order boltzmann machine," in *NIPS*, 2010.

[24] V. Mnih, N. Heess, A. Graves *et al.*, "Recurrent models of visual attention," in *NIPS*, 2014.

[25] C. Cao, X. Liu, Y. Yang, Y. Yu, J. Wang, Z. Wang, Y. Huang, L. Wang, C. Huang, W. Xu *et al.*, "Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks," in *ICCV*, 2015.

[26] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *NIPS*, 2015.

[27] T. Bluche, "Joint line segmentation and transcription for end-to-end handwritten paragraph recognition," in *NIPS*, 2016.

[28] A. Miech, I. Laptev, and J. Sivic, "Learnable pooling with context gating for video classification," *arXiv preprint arXiv:1706.06905*, 2017.

[29] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *ECCV*, 2016.

[30] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," *arXiv preprint arXiv:1709.01507*, 2017.

[31] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *arXiv preprint arXiv:1405.3531*, 2014.

[32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.

[33] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014.

[34] H. Liu, Y. Tian, Y. Yang, L. Pang, and T. Huang, "Deep relative distance learning: Tell the difference between similar vehicles," in *CVPR*, 2016.