2022 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA) | 978-1-6654-9991-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICAICA54878.2022.9844630

Image Recognition in Autonomous Driving Based on Improved Swin Transformer

Yequan Bie*
School of Automation Science and Engineering
South China University of Technology
Guangzhou, China
yequanbie@gmail.com

Hao Tan*
School of Automation Science and Engineering
South China University of Technology
Guangzhou, China
th101292@163.com

*Equal contribution.

***Abstract*—Traffic image recognition is one of the most important phases in the field of autonomous driving, including the classification of real-time periods and the detection of pedestrians, vehicles, etc. on the road. In this paper, we proposed an end-to-end classifcation and object detection method based on Swin Transformer with improved cascade RoI heads. Our method focuses on the scale problem from language to vision field in traditional Transformer model and the mismatch problem of bounding box regression in previous object detection methods (e.g. Faster R-CNN). A modified Swin Transformer architecture with multiple RoI heads is adopted in the proposed model to perform classification and object detection, meanwhile improved optimization strategies are used. We applied the model to SODA10M, an autonomous driving dataset released by Huawei, and finally attained a classification accuracy of 95.3% and a detection mAP of 91.9%, both achieving state-of-the-art.**

***Keywords*—Autonomous Driving, Object Detection, Swin Transformer, Cascade RoI Head**

I. Introduction

In the field of deep learning, especially computer vision, convolutional neural networks have dominated various tasks for many years, covering image recognition, object detection, semantic segmentation, instance segmentation, etc. The convolutional neural network applied to the field of computer vision was proposed by Yann LeCun in the 1980s and was successfully applied to the task of handwritten digit recognition [4]. In the 21st century, the ImageNet Large Scale Visual Recognition Challenge initiated by Feifei Li and others set off another wave of artificial intelligence algorithms. Starting from AlexNet [2], the deep learning model based on convolutional neural network has been used as a powerful target for solutions to vision problems, developed rapidly and widely. The backbone networks that emerged later include: VGG [3] with more model parameters, GoogLeNet [5] with Inception structure, and ResNet [6] with more network layers. In addition, object detection is also one of the most important tasks in the field of computer vision. There are also many excellent works in this field, including SSD [7] and YOLO [8] which are one-stage algoithms, and Faster R-CNN [9], SPPNET [10], etc. which are two-stage.

However, recently, the classic model Transformer [17], which should belong to the field of natural language processing, has been introduced into the field of computer vision and has gained huge success, including in the area of image classification [11] and joint vision-language modeling [12]. One of the most successful models at present is the Swin Transformer [1], in which sliding window technology is mainly used to limit the complexity of self-attention and at the same time supports cross-window connections, successfully allowing researchers to see that the transformer can be used in vision field and achieve immense success. Besides, various models with the Swin Transformer as their backbone have also made breakthroughs in different vision tasks [22]–[24].

In this paper, we propose a more innovative method for image classification and object detection on the dataset SODA10M [13]. For the classification problem, we use an improved Swin Transformer to classify the period of the day in which the image scene is located. For the object detection problem, we aim at the scale problem from langauge to vision field, that is, tokens are of a fixed scale but vision applications like object detection are not, combine the two models of Swin Transformer and Cascade R-CNN [14] with appropriate modification, and successfully achieve high performance.

II. SODA10M Dataset

A. Description

SODA10M is a new large-scale 2D dataset released by HUAWEI Noah's Ark Lab in 2021, which includes 10M images without annotations and 20k images with exact bounding boxes belonging to 6 object categories. This dataset is the largest 2D autonomous driving dataset until now which is ten times larger than Waymo dataset [26]. The rich diversity ensures its generalization performance as self-supervised pre-training data set and semi-supervised additional data in downstream autonomous driving tasks.

The task of collecting images is assigned to many ride-hailing drivers. They are supposed to use mobile phones and dashcams to obtain a large number of images. The visual angle must be maintained in the middle of the image, and

Fig. 1. Examples of SODA10M dataset

the contents of the car must not exceed 15 % of the picture. To achieve more diversity, suppliers are required to obtain images in various weather conditions, periods, places and cities. Some examples of SODA10M dataset are shown in Fig. 1.

B. Our Tasks & Data Augmentation

In this papar, our tasks are to perform classification and object detection on images in the SODA10M dataset, where for the classification problem, we need to distinguish the time period in which the scene in the image is located, including Daytime, Night and Dawn/Dusk. The objective of this task is to allow the autonomous driving system to correctly distinguish the current time period so as to adapt to the current environment and adjust the brightness of the lights automatically. For object detection task, the model needs to predict the exact locations of cars, trucks, pedestrians, trams, cyclists and tricycles in the form of tightly-fitting 2D bounding boxes.

In the field of deep learning, the number of parameters of model will be significantly increased due to the deepening of the number of network layers. As a result, there will be an over-fitting problem when data is not sufficient. The over-fitting problem will make the network more focus on memorizing the features of the training set rather than learning general laws, which leads to a bad performance on test data. Therefore, in order to alleviate the over-fitting problem of our model and improve the generalization ability, we take the following measures for data augmentation:

- Random flip
- Resize
- Random crop

An example of horizontal filp is shown in Fig. 2.

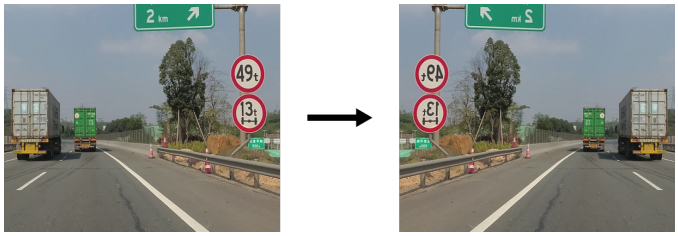


Fig. 2. Horizontal flip

III. Method

A. Architecture

An overview of the architecture of our model is presented in Fig. 3. We first split RGB images of SODA10M dataset into patches which are treated as "tokens" in transformer model by patch partition module. Then we use a linear embedding layer to change the dimension. Several Swin Transformer blocks are applied on these tokens together with Patch Merging module. Finally, we propose an architecture with improved cascade RoI heads to help performing object detection.

Patch Partition Our model first partitions an input image into $\frac{H}{K} \times \frac{W}{K}$ patches by a patch partition module, when using a patch size of $K \times K$. Each patch is treated as a so-called "token" often mentioned in the field of natural language processing [17]. The patch partition example with 3×3 patches is presented in Fig. 4.

Linear Embedding Following [11], there is a linear embedding layer applying on the raw-valued feature which is gained from patch partition, and this embedding layer project patches to an arbitray dimension in preparation for passing into the swin transformer blocks.

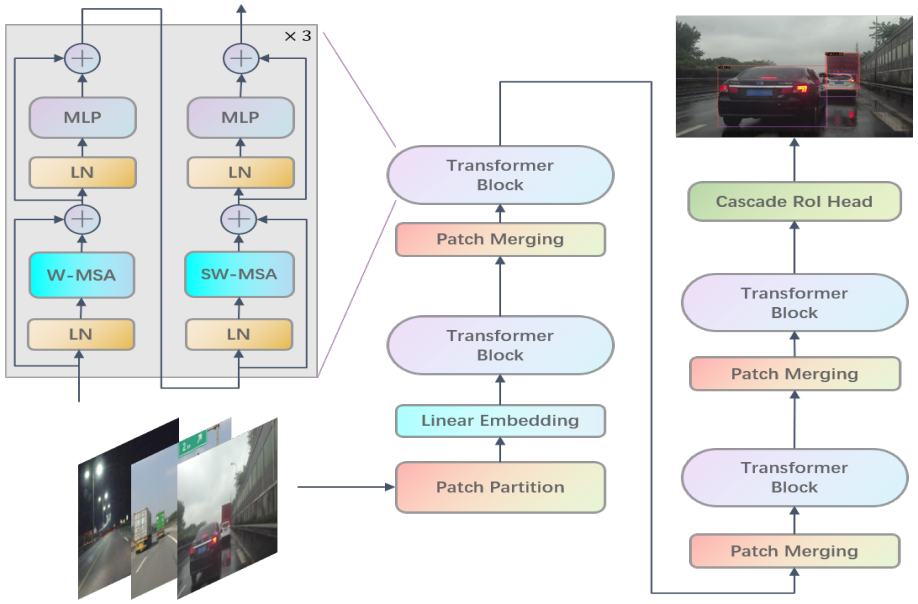


Fig. 3. Model architecture

Swin Transformer Block Each stage has several Swin Transformer blocks, and each swin transformer block is consisted of a 2-layer MLP (Multilayer Perceptron) with GELU [15] non-linearity in between as activation fuction. W-MSA is multi-head self attention modules with regular windowing configurations while SW-MSA is with shifted windowing scheme. A layer normalization layer is applied before each MSA module and MLP and there is a residual connection between different modules. Specifically, the working mechanism of self attention module is as follows:

$$\begin{aligned} \hat{x}^l &= W\text{-}MSA(LayerNorm(x^{l-1}) + x^{l-1}, \\ x^l &= MLP(LayerNorm(\hat{x}^l)) + \hat{x}^l, \\ \hat{x}^{l+1} &= SW\text{-}MSA(LayerNorm(x^l)) + x^l, \\ x^{l+1} &= MLP(LayerNorm(\hat{x}^{l+1})) + \hat{x}^{l+1} \end{aligned} \tag{1}$$

where $\hat{x}^l$ and x^{l-1} represent the output feature maps of the self-attention module and the multilayer perceptron module for block l, respectively.

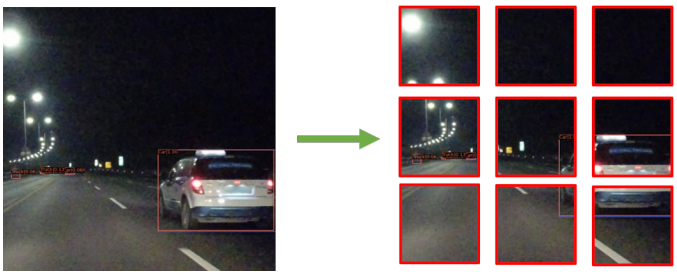


Fig. 4. Patch partition

In order to compute self-attention, we use the following fomular in the Swin Transform block:

$$Attention(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{d} + bias})V \tag{2}$$

where Q,K,V are the $query$, key and $value$ matrices in the pipeline of transformer, d is the dimension of $query$ and $bias$ represents the relative position bias [1].

Patch Merging A patch merging layer is applied before each Swin Transformer block and it concatenates the features of each group of neighboring patches as well as applies a linear layer on concatenated features. Patch merging is used to reduce the number of tokens and form a stage together with Swin Transformer block in order to produce a hierarchical representation.

B. Improved RoI Head

In the object detection task, we know that the Faster R-CNN has a wide range of applications as a classic model. However, it is extremely tough to ask a single regression head which is used in Faster R-CNN to perform consistently at every quality level. Therefore, we can decompose the difficult regression task into a series of simpler procedures, which is also in line with the idea of transformer. In order to apply swin transformer to task of object detection, we combine Cascade R-CNN with it. Specifically, in the downstream task of object detection, this paper uses the improved Swin Transformer as the backbone, and uses the cascade RoI head for object detection bounding box regression and object classification. In addition, different from the regression loss used by Cascade R-CNN, we use GIoU [16] Loss for bounding box regression optimization, which will be discussed in the next section. Our improved framework with cascade RoI head is presented in Fig. 5.

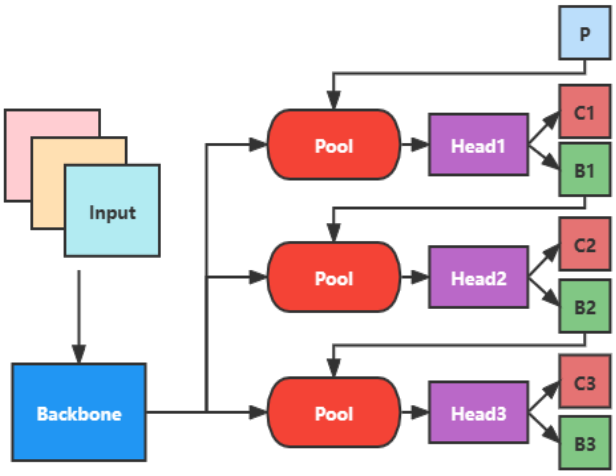


Fig. 5. Our improved net framework with cascade RoI head. B means bounding box and C is classification, P is proposals, $Backbone$ is Swin Transformer here.

C. Loss Function Optimization

1) classification loss: We simply adopt the cross entropy loss as the classification loss of the model, cross entropy loss is a good mean of measurement of the difference between two probability distributions (p and q in the following formular) and is easy to optimize.

$$CrossEntropy(p,q) = -\sum_{x} p(x)log(q(x)) \tag{3}$$

2) regression loss: The model proposed in this paper uses GIoU Loss as the loss function for the regression of the bounding box. After experiments, we found that its performance is better than the loss function mentioned in the paper of Cascade R-CNN. GIoU Loss is effectively an improvement of IoU Loss. Compared with IoU Loss, GIoU Loss not only focuses on the overlapping area of the two bounding boxes, but also cares about the non-overlapping areas, which can better reflect the difference between the predicted bounding box and ground truth. $GIoULoss = 1 - GIoU$ and the calculation process of GIoU is as follows.

Algorithm 1 Generalized Intersection over Union

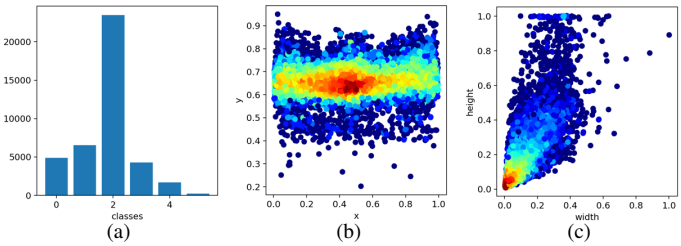
Input: Two arbitrary convex shapes: $A, B \subseteq \mathbb{S} \in \mathbb{R}^n$
Output: GIoU

1: For A and B, find the smallest enclosing convex object C, where $C \subseteq \mathbb{S} \in \mathbb{R}^n$
2: $IoU = \frac{|A \cap B|}{|A \cup B|}$
3: $GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$

IV. Experiments

A. Exploratory Data Analysis

We train and test the proposed model using the SODA10M dataset released by Huawei, which is designed to promote the development of industrial autonomous driving applications with 10k precisely annotated images provided. There are 5k in training set and 5k in validation set. In our experiments, we directly use the training set for training and test the model on the validation set. The distribution of the labeled boxes of the training data is shown in Fig. 6. It is obvious that there are very few samples of the "tricycle" category, meaning that there is a data imbalance problem, which will definitely affect the ability of the model. Thus we take appropriate resampling strategy on the "tricycle" category, and it is verified that the model mAP increases by about 2.6% after resampling. Learned from Fig. 6(b), the distribution of bbox positions is relatively uniform. Most of the objects are located in the middle of the picture. As can be seen from Fig. 6(c), there are a large number of small bounding boxes, indicating that there are many "far away" vehicles and even overlapping vehicles and pedestrians need to be detected in the task, which makes our task quite challenging.



(a) (b) (c)

Fig. 6. Distribution of SODA10M training set. (a) Index $0 \sim 5$ represent categories "pedestrain", "cyclist", "car", "truck", "tram" and "tricycle", respectively. (b) "x" and "y" represent the normalized center position of the bbox. (c) "width" and "height" represent the normalized width and height of the bounding box, respectively.

B. Evaluation Metrics

For the evaluation metrics of object detection, we just follow Faster-RCNN [9] to utilize the region-based mean average precision (AP) at IoU threshold 0.5 (AP_{50}) and 0.75 (AP_{75}). As for the period classification, we adopt top-1 accuarcy as our criterion.

C. Model Settings

We compared models of different complexity and give the best model on SODA10M based on experimental results. The initial embedding dimension is set to 128 and 4 swin transformer blocks are stacked 2, 2, 16 and 2 times respectively. The number of heads in MSA module is 4, 8, 16 and 32. For the detection head, a total of 3 cascade RoI heads are used. The weights of loss are 1, 0.5 and 0.25 respectively. Moreover, we adopt GIoU Loss instead of smooth L1 Loss in bounding box regression.

D. Implementation Details & Result Analysis

Our network is implemented based on PyTorch. We complete all the experiments on Nvidia GeForce RTX 2080 Ti. Training for the proposed network takes about 9 hours.

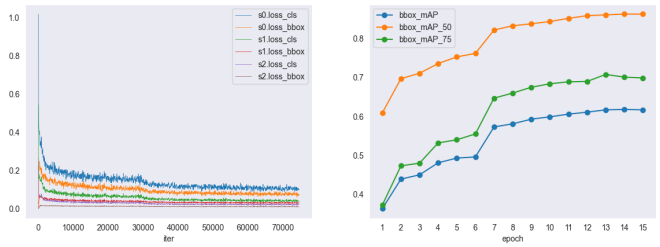
For classification task, we apply Swin Transformer as feature exactor with MLP head. The loss function we adopt is

TABLE I
COMPARISON BETWEEN DIFFERENT METHODS INCLUDING SOTA ON SODA10M VALIDATION SET.

Methods	pedestrain	cyclist	car	truck	tram	tricycle	mAP_{50}
RetinaNet [18]	58.92	61.83	66.14	72.43	62.92	38.82	60.2
YOLOv5 [19]	85.37	84.29	90.12	91.36	86.63	68.32	84.3
YOLOR [20]	83.92	87.48	92.36	91.85	87.32	64.92	84.6
YOLOX [21]	85.37	89.65	93.12	91.23	89.36	66.12	85.8
Ours	89.45	95.49	96.84	94.37	95.77	79.43	**91.9**

multi-class cross entropy in (3). And the optimization method is SGD. The weight decay and momentum are set to 10^{-4} and 0.9, respectively. The weights we use are pretrained from ImageNet-1K. After 10 epochs of training, the model converges rapidly. The top-1 accuracy achieves 95.3% in the end. It can be observed that using swin transformer as backbone, the convergence speed and model accuracy both are excellent on the classfication task.

As for the detection task, we apply the AdamW [25] optimization algorithm with learning rate policy of linear warmup and step decay. The learning rate will increase to 10^{-4} in 500 iterations and then divided by 10 after the 6th and the 12th epoch. For the best results, we ultimately decline the learning rate by 10 times in the last epoch. The weight decay is set to 0.05 to alleviate overfitting. On top of that, we adopt multi-scale training for better performance. The training of models based on CNN (i.e. YOLO and RetinaNet) follows the default settings. The training loss and validation mAP are shown in Fig. 7.



(a) Training Loss (b) Validation mAP

Fig. 7. The training loss(a) and the validation mAP(b). (a) $s0 \sim s2$ represents the three detection RoI heads. (b) mAP_{50} (follows the COCO evalutation metrics) achieves a high level on validation set.

As shown in table I, our model outperforms most popular one-stage object detection methods. The mAP of validation set reaches 91.9%. Our model performs particularly well in the recognition of "car" and "cyclist" (95.49% and 96.84%, respectively), especially cyclist category, which is significantly better than YOLO series. This may be due to the self-attention mechanism in Swin Transformer, which can quickly lock the region with obvious features in the image. The addition of the shifted window module ensures that there is continuity between the different windows, making it more helpful for big targets that can easily be shelled. The main constraint on the model performance is "tricycle" (only 79.43%AP). But even so, our method significantly outperforms RetinaNet by 40.61%. Fig. 8 and Fig. 9 show some examples of object detection results. They indicate that our method has a good performance on traffic detection regardless of the period (i.e. daytime, dawn/dusk or nignt). Also there are still some false detections in results which mainly contains two kinds of error: i) identify several pumps besides streets as pedestrains, ii) the recognition accuracy of overlapping vehicles in the long-sighted distance is relatively low.

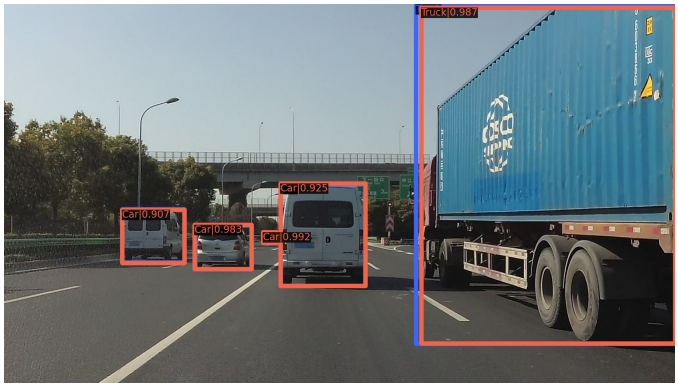


Fig. 8. The object detection results with confidence scores. *Red* is predictions while *blue* is the ground-truth.

V. CONCLUSION

In this paper, an end-to-end classifcation and object detection method based on Swin Transformer with improved cascade RoI head is proposed and is applied to autonomous driving dataset. Our model focuses more on the scale problem from language to vision field in traditional Transformer model through self-attention and shifted window scheme, and applied cascade RoI head with different weigthts to decompose the bounding box regression task into a sequence of smaller steps, which is also in line with the idea of Transformer as a sequence model. We also improve the optimization method by modifying the loss function and training strategy. Our model performs both classification and object detection tasks in SODA10M and achieves state-of-the-art, which attains 95.3% (top-1 accuracy) and 91.9% (mAP_{50}), respectively. We believe that our model can also be used and performs well in other fields besides traffic image recognition.

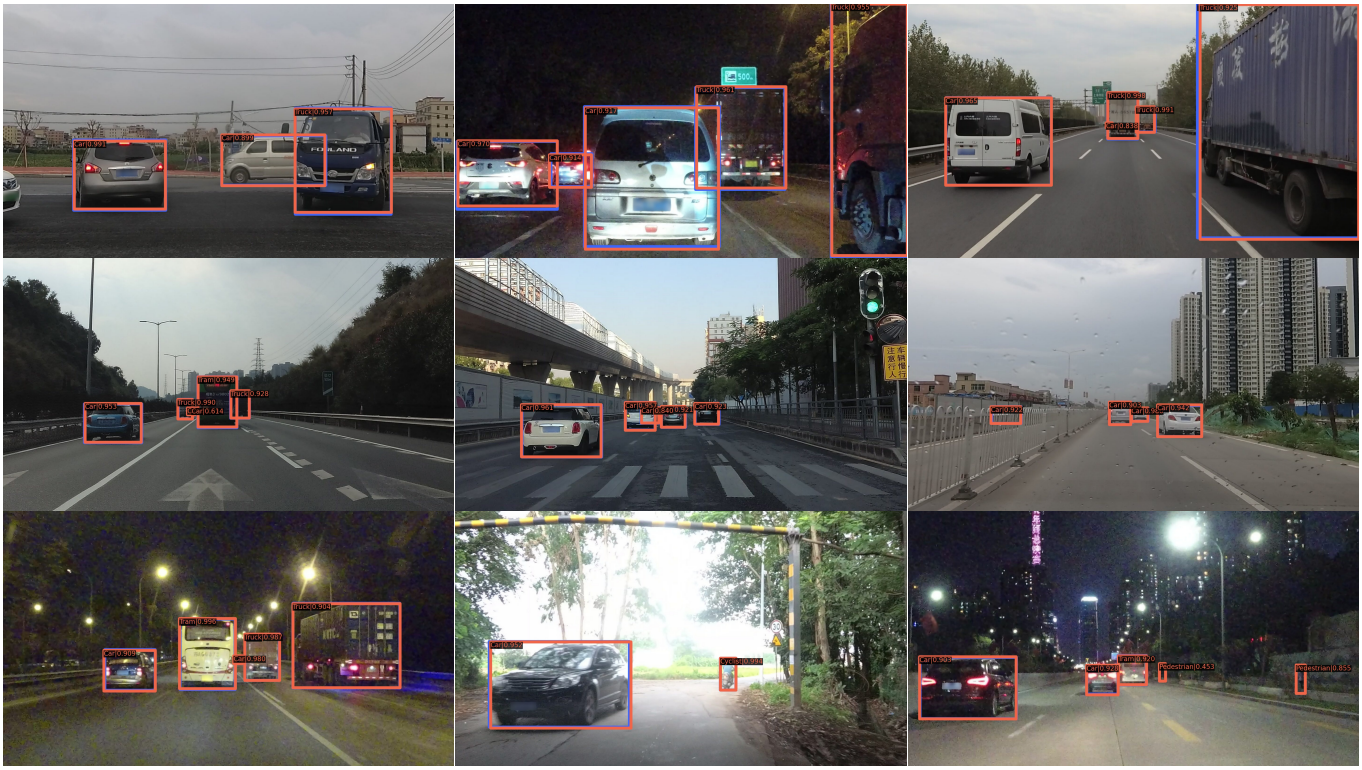


Fig. 9. Examples of object detection results on SODA10M validation set using the proposed model. *Red* is predictions while *blue* is the ground-truth. A score threshold of 0.5 is used to display these images.

REFERENCES

[1] Liu, Ze, et al. "Swin transformer: Hierarchical vision transformer using shifted windows." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

[2] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems 25 (2012).

[3] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).

[4] LeCun, Yann, et al. "Comparison of learning algorithms for handwritten digit recognition." International conference on artificial neural networks. Vol. 60. 1995.

[5] Szegedy, Christian, et al. "Going deeper with convolutions." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[6] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[7] Liu, Wei, et al. "Ssd: Single shot multibox detector." European conference on computer vision. Springer, Cham, 2016.

[8] Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[9] Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." Advances in neural information processing systems 28 (2015).

[10] Purkait, Pulak, Cheng Zhao, and Christopher Zach. "SPP-Net: Deep absolute pose regression with synthetic views." arXiv preprint arXiv:1712.03452 (2017).

[11] Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." arXiv preprint arXiv:2010.11929 (2020).

[12] Radford, Alec, et al. "Learning transferable visual models from natural language supervision." International Conference on Machine Learning. PMLR, 2021.

[13] Han, Jianhua, et al. "Soda10m: Towards large-scale object detection benchmark for autonomous driving." arXiv preprint arXiv:2106.11118 (2021).

[14] Cai, Zhaowei, and Nuno Vasconcelos. "Cascade r-cnn: Delving into high quality object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[15] Hendrycks, Dan, and Kevin Gimpel. "Gaussian error linear units (gelus)." arXiv preprint arXiv:1606.08415 (2016).

[16] Rezatofighi, Hamid, et al. "Generalized intersection over union: A metric and a loss for bounding box regression." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

[17] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).

[18] Lin, Tsung-Yi, et al. "Focal loss for dense object detection." Proceedings of the IEEE international conference on computer vision. 2017.

[19] YOLOv5, "Yolov5," https://github.com/ultralytics/yolov5.

[20] Wang, Chien-Yao, I-Hau Yeh, and Hong-Yuan Mark Liao. "You only learn one representation: Unified network for multiple tasks." arXiv preprint arXiv:2105.04206 (2021).

[21] Ge, Zheng, et al. "Yolox: Exceeding yolo series in 2021." arXiv preprint arXiv:2107.08430 (2021).

[22] Liang, Jingyun, et al. "Swinir: Image restoration using swin transformer." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

[23] Liu, Ze, et al. "Video swin transformer." arXiv preprint arXiv: 2106.13230 (2021).

[24] Xie, Zhenda, et al. "Self-supervised learning with swin transformers." arXiv preprint arXiv:2105.04553 (2021).

[25] Loshchilov, Ilya, and Frank Hutter. "Decoupled weight decay regularization." arXiv preprint arXiv:1711.05101 (2017).

[26] Sun, Pei, et al. "Scalability in perception for autonomous driving: Waymo open dataset." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.