Traffic Sign Interpretation via Natural Language Description

Chuang Yang, Kai Zhuang, Mulin Chen, Haozhao Ma, Xu Han, Tao Han, Changxing Guo, Han Han, Bingxuan, Zhao, and Qi Wang, *Senior Member, IEEE*

***Abstract*—Most existing traffic sign-related works are dedicated to detecting and recognizing part of traffic signs separately, which fails to analyze the global semantic logic among signs and may convey inaccurate traffic instruction information. Following the above issues, we propose a traffic sign interpretation (TSI) task, which aims to interpret global semantic interrelated traffic signs (e.g., driving instruction-related texts, symbols, and guide panels) into a natural language for providing complete traffic instruction support to autonomous or assistant driving. Meanwhile, considering the lack of an effective framework for the proposed TSI task in existing works, we design a multi-task learning architecture (TSI-arch) to detect and recognize various traffic signs with drastic changes in sizes and aspect ratios. Meanwhile interpreting these signs into a natural language like a human according to Chinese design criteria of road traffic signs. Furthermore, the absence of a public TSI available dataset prompts us to build a traffic sign interpretation dataset, namely TSI-CN. The dataset consists of real road scene images, which are captured from the highway and the urban way in China from a driver's perspective. It contains rich location labels of texts, symbols, and guide panels, and the corresponding natural language description labels. Experiments on our TSI-CN dataset demonstrate that the TSI task is achievable and the TSI architecture can interpret traffic signs from scenes successfully even if there is a complicated semantic logic among signs.**

***Index Terms*—Traffic sign detection, traffic sign recognition, traffic sign interpretation, autonomous driving, intelligent transportation.**

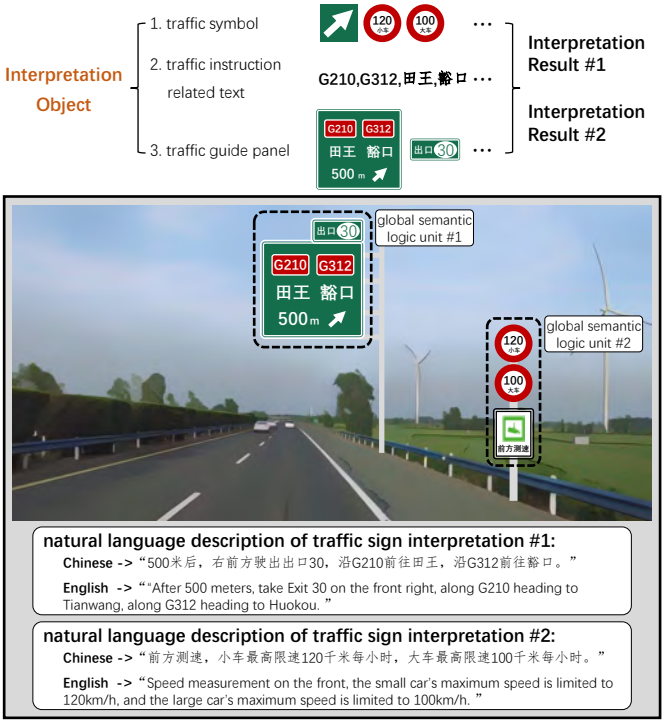


Fig. 1: The illustration of the TSI task. Given an RGB real road scene image, TSI detects and recognizes traffic texts, symbols, and guide panels at first. It then organizes the global semantic logic among these components to generate natural language descriptions with accurate traffic instruction information. The **global semantic logic unit** is a sign set, which includes multiple signs that need to be combined together to interpret complete traffic instruction information according to Chinese design criteria of road traffic signs.

I. Introduction

INTELLIGENT transportation has progressed in recent years with the rapid development of deep learning technology. As a fundamental task in the field of intelligent transportation, traffic sign understanding [1]–[5] has become a hot topic and attracted more attention.

Traffic signs consist of driving instruction-related symbols, texts, and guide panels (as shown in Fig. 1). Initially, researchers mainly focused on the recognition of traffic symbols individually [6], [7], which makes it hard to understand complete traffic sign information and even results in inappropriate driving behaviors (as shown in Fig. 2). For example, the detection and classification of the speed limit symbol, one of the most common traffic signs, has already played an important role in current autonomous or assistant driving systems. However, the symbol often appears along with other signs to convey road traffic information as a whole (e.g., the combination of speed limit symbol and vehicle type information). Therefore, driving according to the information obtained from speed limit signs alone may cause traffic jams or even accidents (as shown in Fig. 2). Though some works propose to recognize

This work was supported by the National Natural Science Foundation of China under Grant U21B2041.

Chuang Yang, Kai Zhuang, Xu Han, and Changxing Guo are with the School of Computer Science, and with the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, Shaanxi, P. R. China.

Mulin Chen, Haozhao Ma, Han Han, Bingxuan Zhao, and Qi Wang are with the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, P.R. China.

Tao Han is with Shanghai Artificial Intelligence Laboratory, Longwen Road 129, Xuhui District, 200232 Shanghai, China.

E-mail: cyang113@mail.nwpu.edu.cn, zhuangkai@mail.nwpu.edu.cn, chenmulin@mail.nwpu.edu.cn, haozhaoma@mail.nwpu.edu.cn, hxu04100@gmail.com, hantao10200@gmail.com, 15035187080@163.com, 1356376210@qq.com, bxuanzhao202@gmail.com, crabwq@gmail.com.

Chuang Yang and Kai Zhuang contributed equally to this work.

Qi Wang is the corresponding author.

Fig. 2: The essential difference between the information provided from the existing sign recognition task and the proposed traffic sign interpretation (TSI) task, respectively. It may lead to traffic accidents when conveying the information of limitation speed signs without scene additional information (such as the ramp in this example) to the autonomous driving system or drivers.

traffic texts [8], [9], considering all symbols, texts, and guide panels together for understanding road traffic information is still under researched. Recently, some researchers have tried to predict the relationships among traffic directional symbols and texts [10], [11]. Though they have achieved progress compared with previous related works, putting directional symbols and texts together simply without the analysis of global semantic logic among signs makes it still hard to interpret complete traffic instruction information from real road scenes (such as the example illustrated in Fig. 3).

Considering the above problems, the task of **traffic sign interpretation (TSI)** is proposed in this work. TSI aims to interpret all interrelated signs (including symbols, texts, and panels) within the same global semantic logic unit (such as the example shown in Fig. 1) into natural languages like a human, where **global semantic logic unit** is a sign set, which includes multiple signs that need to be combined together to interpret traffic instruction information. Different from previous works [10], [11] understanding symbol information separately (the first row in Fig. 2 first row), since the interpreted results of natural language forms are organized according to all interrelated signs jointly and the rules of Chinese design criteria of road traffic signs (the second row in Fig. 2), which ensure the integrality of the traffic instruction information that support autonomous and assistant driving systems. To achieve the interpretation task, we introduce a multi-task architecture, namely **TSI-arch**. Considering the drastic changes in sizes and aspect ratios of symbols, texts, and guide panels make existing traditional detectors hard to cover as many as possible, TSI-arch proposes to locate central regions of signs at pixel level at first and then expand the region to rebuild the final bounding boxes, which avoids the limitations of existing traditional detectors for various sings. After determining the location and category of each sign, the architecture next analyzes the global semantic logic based on the rules that are learned from the Chinese design criteria of road traffic signs combined with the detected signs. In the end, natural language that can convey traffic instruction information is organized to serve drivers and autonomous driving systems.

Furthermore, the absence of a public TSI available dataset prompts us to construct a related dataset for promoting the progress of the sign interpretation community. We build a traffic sign interpretation dataset from China, namely **TSI-CN**, to fulfill the research and evaluation in this field. Compared with the previous **traffic sign recognition dataset** [12]–[14] and **traffic sign understanding dataset** [10], [11], TSI-CN enjoys the following two main advantages over them: 1) Containing rich location and recognition labels of traffic symbols, texts, and guide panels simultaneously; 2) Natural language descriptions based on the analysis of global semantic logic among signs, where the semantic logic is organized according to Chinese design criteria of road traffic signs. The contributions of this work are summarized as follows:

1) The traffic sign interpretation (TSI) task is proposed to interpret interrelated signs into natural languages based on the analysis of global semantic logic among them. It helps understand signs and convey complete instruction information from real road scenes like a human.
2) A TSI task corresponding dataset (TSI-CN) is built based on plenty of images from real road scenes of China. It is the first traffic sign dataset that is equipped with location and recognition labels of various signs and natural language description labels of logic-interrelated signs simultaneously.
3) A TSI architecture is designed, which consists of the three sub-structures of detection, recognition, and interpretation. Experimental results on the created TSI-CN dataset demonstrate that the TSI task is achievable and the effectiveness of TSI architecture.
4) The TSI task and the corresponding dataset will promote the progress of the traffic sign recognition community and provide support for the development of autonomous and assistant driving systems.

The rest of the paper is organized as follows. Section II introduces the related works on traffic sign recognition method and dataset. Section III and Section IV describe the TSI task and TSI-CN dataset. Section V shows the details of the proposed TSI architecture. The experimental results are discussed in Section VI. Section VIII concludes the paper.

II. Related Work

The field of traffic sign recognition has attracted more attention recently with the rapid development of computer vision-based autonomous or assistant driving systems. Existing related works will be introduced from the two aspects of the method and dataset in this section.

A. Traffic Sign-Related Method

Initially, researchers focused on traffic symbol recognition individually. Saturnino *et al.* [15] classified symbols into different categories based on support vector machines (SVMs). Lu *et al.* [16] introduced local manifold structures into sign classification via graph embedding algorithm. With the development of deep learning, many Convolutional Neural Networks (CNNs)-based traffic symbol recognition methods [17]–[20] are proposed. Dan *et al.* [17] and Pierre *et al.* [18]

TABLE I: The essential differences between the existing sign datasets and TSI-CN. 'GSL' and 'NL' denote 'global semantic logic' and 'natural language', respectively. 'RRS' denotes real road scenes. symbol_a and symbol_o are arrow symbols and the others, respectively.

dataset	RRS	detection content				recognition content				interpretation content	
		symbol_a	symbol_o	text	panel	symbol_a	symbol_o	text	panel	GSL analysis	NL description
GTSRB [12]			✔				✔				
CTSD [13]	✔		✔				✔				
TT100K [6]	✔		✔				✔				
DFG [14]	✔		✔				✔				
CTSU [10]		✔	✔	✔		✔		✔			
RS10K [11]	✔	✔	✔	✔	✔	✔		✔			
TSI-CN (**Ours**)	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔

proposed to detect and classify symbols with an end-to-end CNN framework, which ran faster and recognized more accurately than previous traditional methods. Jin *et al.* [19] designed a hinge loss to encourage the model to focus on hard samples. Li *et al.* [20] utilized a Generative Adversarial Network (GAN) to represent small symbols to super-resolved ones to avoid challenges from low-resolution images. Except for the traffic symbol recognition, Rong *et al.* [8] combined CNNs and Recurrent Neural Networks (RNNs) to recognize traffic texts inspired by the scene text detection and recognition technique [21]–[23]. To better understand traffic signs, Guo *et al.* [10] recognized traffic symbols and texts simultaneously and combined recognition results simply. However, interpreting accurate instruction information according to the global semantic logic among signs from real road scenes is still under researched.

B. Traffic Sign-Related Dataset

To fulfill the research and evaluation of the traffic sign recognition field, some task-related datasets [6], [10], [12]–[14] are constructed. Johannes *et al.* [12] collected more than 50,000 images in Germany. The dataset contained 43 different traffic symbols with rectangle box labels. Yang *et al.* [13] created a China Traffic Sign Dataset (CTSD) with sparse symbol distribution. The images of different sizes, low-light scenes, and low resolution in CTSD encouraged researchers to focus on complex background analysis. Zhu *et al.* [6] constructed a large-scale traffic symbol dataset, namely Tsinghua-Tencent 100K, that was sampled from Tencent Street View panoramas. Considering the deficiency of the most normal symbols, Domen *et al.* [14] built a traffic-sign dataset with 200 symbol categories. Considering the recognition of symbols or texts individually could not convey integrity traffic instruction information, Guo *et al.* [10], [11] cropped traffic guide panels patches from images to construct a traffic sign understanding dataset. It is equipped with labeled arrows, texts, and the relationships among them. However, a traffic sign dataset that is captured from real road scenes while containing labeled symbols, texts, guide panels, and natural language descriptions of traffic instruction information based on the analysis of global semantic logic among signs simultaneously is needed. It will promote the progress of the traffic sign recognition community and provide support for the development of autonomous and assistant driving systems.

III. TSI Task Description

Different from previous works [10] that recognize traffic directional marks and predict the relationship among them simply, the TSI task aims to convey accurate traffic instruction information via natural language l by finding out all signs $q = \{s, t, p\}$ from real road scenes and organizing the **global semantic logic** h among them. The whole interpretation process of traffic signs can be described as:

$$h = \mathcal{H}(q), \tag{1}$$

$$l = \mathcal{T}(h, q), \tag{2}$$

where q is a collection with detected and recognized symbols (s), texts (t), and guide panels (p). h is the hidden state of the global semantic logic of q that generated from some nonlinear functions $\mathcal{H}(\cdot)$. $\mathcal{T}(\cdot)$ and l are the interpretation mapping function and output natural language strings l with traffic instruction information. The same as $\mathcal{H}(\cdot)$, $\mathcal{T}(\cdot)$ consists of nonlinear functions. l can be written as:

$$l = (c_0, c_1, ..., c_{n-1}, c_n), \tag{3}$$

where c_n is the n-th character of the string l.

IV. TSI-CN Dataset

To fulfill the research and evaluation of TSI task, the TSI-CN dataset is constructed. This section describes the dataset from four perspectives: data collection, data annotation, data analysis, data split and evaluation metric.

A. Data Collection

Our data are collected from self-shooting in the driver's perspective through the DJI OSMO Pocket2 camera, which helps simulate the locations of traffic recorders for the adaptability of subsequent algorithm deployment. The images in the TSI-CN Dataset are captured from traffic record videos related to some popular Chinese cities, including Xi'an, Xianyang, Baoji, Zhengzhou, and Zhoukou, containing some typical road scenes, such as highways, urban roads, urban streets, and rural roads. In the capturing process, we sample a key frame of a picture every 5 seconds from the recorded videos. Considering

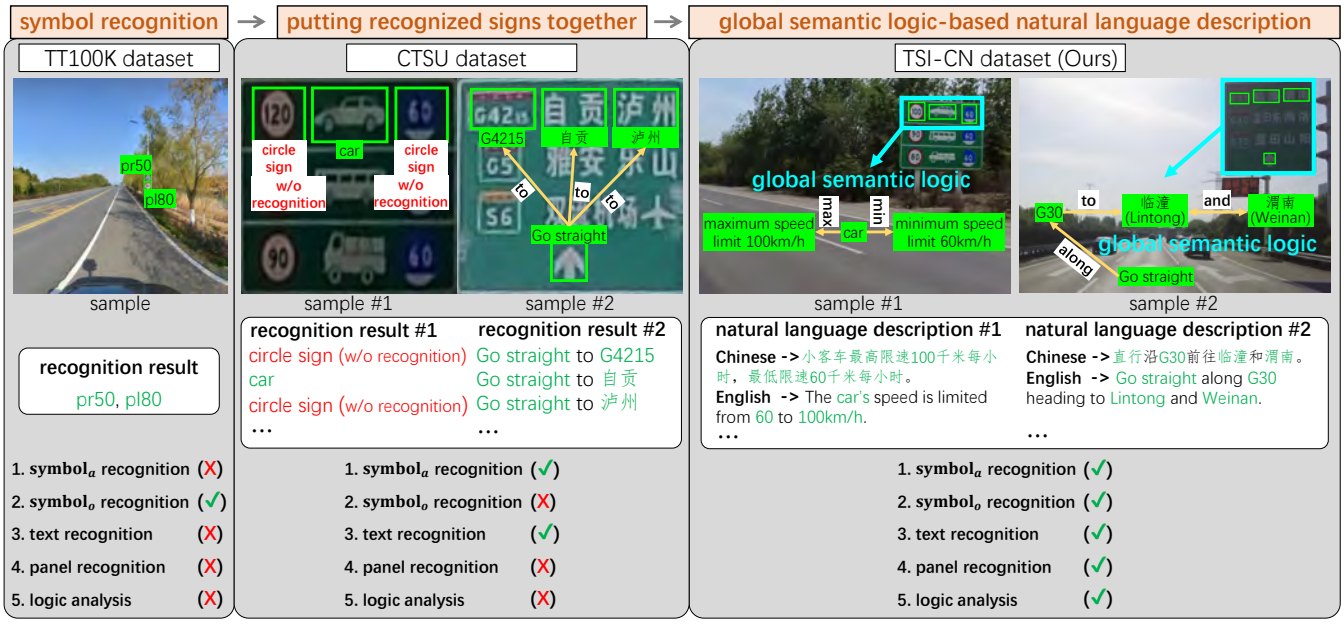


Fig. 3: The illustration of the essential differences (in Table I) between the existing representative sign recognition datasets and the sign interpretation dataset (Our TSI-CN).

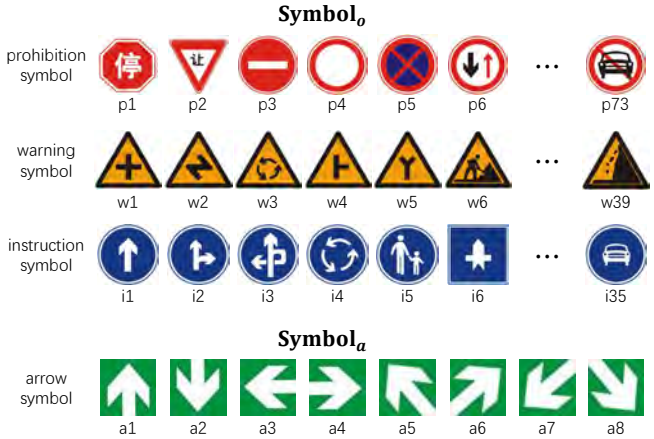


Fig. 4: The visualization of traffic symbols in TSI-CN dataset.

text type	example	statistics
Arabic numerals	'0', '1', '2',···, '9'	10
English character	'a', 'b', 'c', ···, 'Z'	52
Chinese character	'专', '世', '丘',···, '个'	995
Other special character	'(', '!', '+',···, '&'	28

Fig. 5: The visualization and statistics of traffic texts in TSI-CN dataset.

Fig. 6: The visualization of guide panels in TSI-CN dataset.

the traffic jam will lead to duplicate frames within a long time, the above-sampled frames per 5 seconds will be re-sampled to construct the TSI-CN dataset.

B. Data Annotation

For providing supervision labels to detection, recognition, and interpretation tasks (as shown in Fig. 3), we annotate the collected images using the Labelme platform to generate the following three kinds of annotations: 1. Detection annotation; 2. Recognition annotation; 3. Natural language description annotation. The details of the annotation process and criterion will be illustrated following.

Detection and recognition annotation. Both the generation of detection and recognition labels are done together. Concretely, boxes with four corner points are drawn first to label sign (including symbol, text, and guide panel) locations to serve as the training supervision information for the detection task. Then, the recognition annotation can be labeled via the box name. Here, we design different naming strategies to the symbol, text, and guide panel for the following convenient labeling process.

For the **symbol**, we divide them into two types (as shown in Fig. 4), including symbol_a (***a***rrow) and symbol_o (***w***arning, ***i***nstruction, and ***p***rohibition), and naming symbol boxes via the combination string of the symbol type and serial number (e.g., 'w10' and 'a2'), where 'w' and 'a' represent the symbol

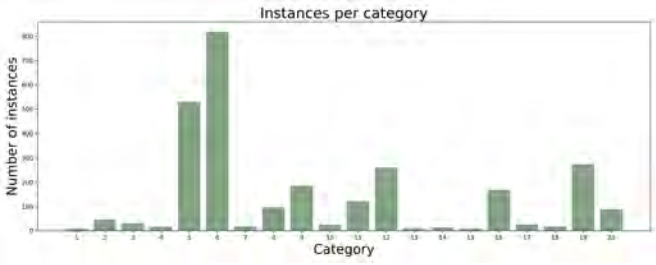


Fig. 7: The category statistics of traffic instruction in TSI-CN dataset.

number information : distance / speed / exit number			
location information : place name / road name			
direction information : go straight / towards the right front / turn right / towards the left front / turn left			
vehicle information : car / truck / bus			
#	instance category	#	instance category
1	currently traveling at [location]	11	[direction] reaching [location] and heading to [location]
2	take exit [number] on the front right	12	[direction] along [location] heading to [location]
3	heading to [location]	13	currently traveling at [location], after [number] meters, [direction] heading to [location]
4	after [number] meters, take exit [number] on the front right	14	currently traveling at[location], after [number] meters, along [location] heading to [location]
5	after [number] meters, heading to [location]	15	after [number] meters, [direction] reaching [location] and heading to [location]
6	[direction] heading to [location]	16	after [number] meters, [direction] along [location] heading to [location]
7	currently traveling at[location], after [number] meters, heading to [location]	17	[direction] along [location] heading to [location], along [location] heading to [location]
8	currently traveling at [location], [direction] heading to [location]	18	after [number] meters, [direction] along [location] heading to [location] along [location] heading to [location]
9	after [number] meters, [direction] heading to [location]	19	[vehicle] speed is limited from [number] to [number] km/h
10	after [number] meters, along [location] heading to [location]	20	other

Fig. 8: The visualization of traffic instruction in English.

type and the serial number refers to which particular category it is in a type.

For the **text**, the recognition annotation is the corresponding text string, which includes Arabic numerals, English, Chinese, and other special characters (as shown in Fig. 5). Particularly, considering some strokes of texts stick to each other making it difficult to distinguish them, they are labeled as '###' and ignored in both the training and inference processes by following the way of previous OCR datasets [24]–[26].

For the **guide panel**, it is divided into seven categories according to the content and basic visual features (as shown in Fig. 6). They are the prohibition, warning, normal road instruction, highway instruction, scenic area instruction, notice, and dynamic prompt panels, which are labeled with '1'∼'7' respectively. Meanwhile, considering there are lots of noises (such as architectural graffiti, billboards, etc) that enjoy highly similar visual features with panels, distinguishing them according to traffic-related symbols and texts is an effective and essential way. Therefore, a guide panel will be ignored if all symbols and texts within it are not clearly visible.

Natural language annotation. In traditional traffic symbol-related tasks, the recognition result is straightforward and distinct (such as 'pl80' and 'il80' in the TT100K dataset). However, traffic signs are inherently complex and varied, including guide panels, text, and symbols. Consequently, the interpretation of traffic signs produces continuous outputs in the form of natural language, making it impossible to directly classify the data in the TSI-CN dataset.

To facilitate the data classification, we follow the Chinese design criteria of road traffic signs to organize the global semantic logic among signs at first. Then, we put the signs

数字信息：距离/限速/出口编号			
位置信息：地名/路名			
方向信息：直行/右前方/右转/左前方/左转			
车辆信息：小客车/大客车/大货车			
#	instance category	#	instance category
1	当前行驶在[位置信息]	11	[方向信息]到达[位置信息]后前往[位置信息]
2	右前方驶出[出口编号]	12	[方向信息]沿[位置信息]前往[位置信息]
3	前方前往[位置信息]	13	当前行驶在[位置信息]，[数字信息]米后，[方向信息]前往[位置信息]
4	[数字信息]米后，请[方向信息]	14	当前行驶在[位置信息]，[数字信息]米后，沿[位置信息]前往[位置信息]
5	[数字信息]米后，前往[位置信息]	15	[数字信息]米后，[方向信息]到达[位置信息]后前往[位置信息]
6	[方向信息]前往[位置信息]	16	[数字信息]米后，[方向信息]沿[位置信息]前往[位置信息]
7	当前行驶在[位置信息]，[数字信息]米后，到达[位置信息]	17	[方向信息]沿[位置信息]前往[位置信息]沿[位置信息]前往[位置信息]
8	当前行驶在[位置信息]，[方向信息]前往[位置信息]	18	[数字信息]米后，[方向信息]沿[位置信息]前往[位置信息]沿[位置信息]前往[位置信息]
9	[数字信息]米后，[方向信息]前往[位置信息]	19	[车辆信息]最高限速[数字信息]，最低限速[数字信息]
10	[数字信息]米后，沿[位置信息]前往[位置信息]	20	其他

Fig. 9: The visualization of traffic instruction in Chinese.

that belong to the same semantic logic unit together and describe the traffic instruction information via natural language based on the global semantic logic. Next, the natural language interpretation is broken down into keyword k combinations (such as "currently traveling at - meters - reaching - go straight / towards the right front / turn right / towards the left front / turn left - along - heading to - along - heading to"). Following the principles of polynomial combinations, there are $\sum_{v=1}^{V} C_V^v$ categories theoretically, where V is category number of keyword. To avoid the impact of rare annotation errors, we have excluded categories with less than 10 instances, only counting those with 10 or more. As shown in Fig. 7, there are 20 types of interpretation categories in the TSI-CN dataset. The corresponding example of each category can be found in Fig. 8 and Fig. 9,

C. Data Analysis

The TSI-CN dataset consists of 2,682 images of 2160×3840 pixels. It covers various road scenarios (such as highways, urban roads, urban streets, and rural roads) and complex optical environments (such as rainy days, backlight, sunlight exposure, and occlusion), which ensures the scene diversity of the TSI-CN dataset. Meanwhile, in the aspect of sign diversity, the TSI-CN dataset includes all 7 kinds of traffic guide panels and 155 kinds of common traffic symbols. The above sample diversity in both optical environments and signs ensures the model's reliability and generalizability when assessing our framework via the TSI-CN dataset. Furthermore, the constructed dataset consists of 2,910 natural language description sentence annotations and provides 144,130 text characters to encourage the text recognition task. Compared with the existing sign datasets, it is the largest from the perspective of image size and annotation number. Existing datasets can be roughly divided into sign recognition and understanding datasets. We specifically compare our TSI-CN dataset with two representative (TT100K and CTSU) datasets to show the essential differences between them in Fig. 3 (RS10K has not been publicly available until submission):

1. **Detection and recognition content.** The proposed TSI-CN dataset provides the annotation of symbols, texts, and guide panels for the sign detection and recognition task simultaneously, which supports to analysis of traffic instruction information integrally.

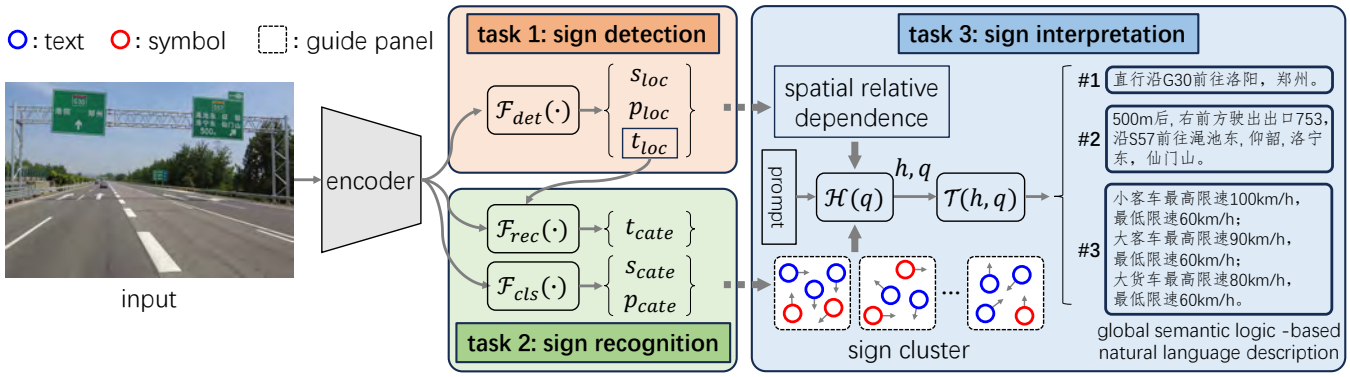


Fig. 10: The overall architecture of TSI-arch. It consists of sign detection, recognition, and interpretation modules, which are illustrated in detail in Fig. 11 and Fig. 12, respectively.

2. **Interpretation content.** Different from all previous datasets, TSI-CN encourages to organization of the signs and the global semantic logic among them to express traffic instruction information via natural language, which ensures the accuracy and integrity of the interpretation of traffic instruction information and avoids misunderstanding to traff signs (such as the example in Fig. 2).

3. **Image resolution.** The images in our TSI-CN dataset are collected from high quality and high-resolution real road scenes. The resolution of TSI-CN image is 2160×3840, which is larger than that of other datasets.

D. Data Split and Evaluation Metric

Data split. TSI-CN Dataset is randomly split into two parts, namely training and test sets, which respectively contain 2,016 and 666 images. To ensure that the statistics (such as sign categories, the average value of sign resolutions) of the subset are almost the same.

Evaluation metrics. We use Precision (P), Recall (R), and Accuracy (Acc) for TSI-arch's detection and recognition performance evaluation, where Precision, Recall, and F-measure can be computed by the True Positive (TP), False Positive (FP), and False Negative (FN) samples as follows:

$$\begin{aligned} \mathrm{P} &= \mathrm{TP}/(\mathrm{TP}+\mathrm{FP}), \\ \mathrm{R} &= \mathrm{TP}/(\mathrm{TP}+\mathrm{FN}), \\ \mathrm{Acc} &= 2\times\mathrm{P}\times\mathrm{R}/(\mathrm{P}+\mathrm{R}). \end{aligned} \tag{4}$$

For the interpretation task, ROUGE [27] and Bleu [28] metrics are adopted to evaluate quality by comparing the similarity between automatically generated text and reference text. Specifically, ROUGE-1 and ROUGE-2 consider the overlap of single words (1-gram) and two consecutive words (2-gram), respectively. ROUGE-L uses the longest common subsequence (LCS) to assess the quality of summaries. It considers the order of sentences, measuring whether the order of sentences in the summary is consistent with the reference text. BLEU-4 calculates the overlap of 1-gram, 2-gram, 3-gram, and 4-gram between machine translation outputs and reference translations, and averages these overlaps with weights. Overall, ROUGE focuses on recall, i.e., how much of the reference text is covered by the automatically generated text, while BLEU focuses on precision, i.e., how much of the automatically generated text is accurate. Furthermore, to score the context and semantics of descriptions simultaneously, the GPT-4 metric is chosen to evaluate our model as a comprehensive measurement. This type of assessment is crucial for capturing the model's performance when dealing with complex semantic logic. It is particularly beneficial for traffic sign interpretation tasks where the understanding of complex semantic relationships is crucial for measuring the semantic closeness of the TSI predictions to their corresponding labels.

However, considering that the above metrics are hard to evaluate the relationship between the word order of keywords (such as place names, road names, etc.) and the descriptions (such as both "Go straight along G70 heading to Xi'an and Xianyang" and "Go straight along G70 heading to Xianyang and Xi'an" are correct descriptions, but the ROUGE and Bleu measure them with different scores), we design the Soft Accuracy metric (SA) to evaluate the semantic quality by assessing whether the syntactic categories of the descriptions are consistent. It will measure the above example with the same score. For instance, since "Go straight along G70 heading to Xi'an and Xianyang" and "Go straight along G70 heading to Xianyang and Xi'an" belong to the syntax category "Go straight - along - heading to", SA scores them as 1, else 0.

V. TSI Architecture

To achieve the TSI task, we constructed a multi-task learning architecture (namely TSI-arch). In this section, the structure of TSI-arch, training and inference process, and optimization function will be described in detail.

A. Overall Pipeline

As described in Section III, the proposed TSI is a visual-to-text task. To reduce training costs (including data and hardware resources) and force the interpretation process to focus on the valid sign regions, we decompose the TSI task into three sub-tasks: sign detection, sign recognition, and the aforementioned-tasks-based sign interpretation first. Then, a multi-task learning architecture is constructed based on a sign detection module, sign recognition module, and sign interpretation module, where each module can be trained with

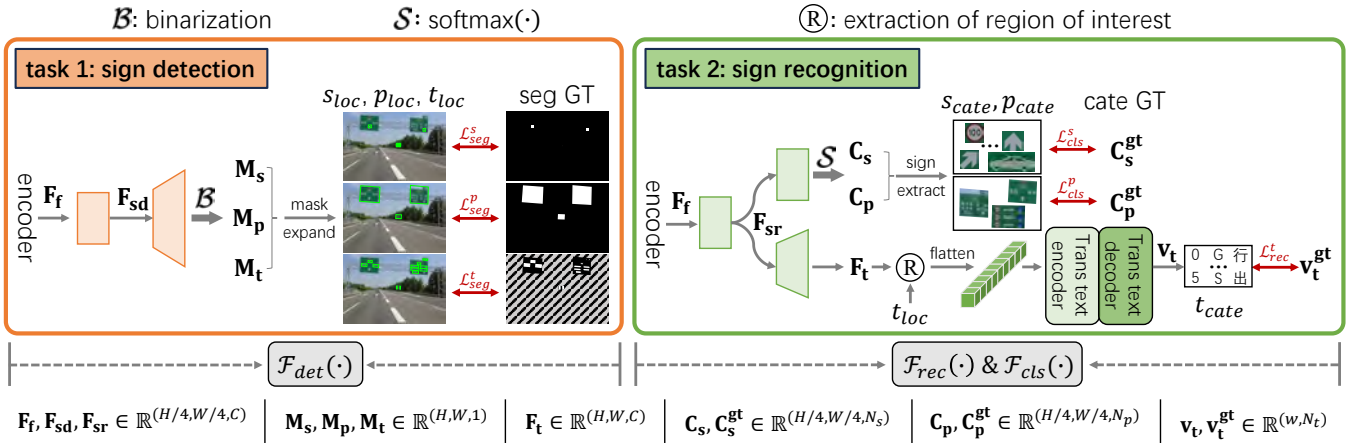


Fig. 11: The details of sign detection and recognition module structures.

the corresponding optimization function, which helps make full use of each sample in the TSI-CN dataset and brings improvements to the training process effectively.

As shown in Fig. 10, the whole framework of TSI-arch consists of an encoder, a sign detection module, a sign recognition module, and a sign interpretation module. As a visual-related framework, we follow the traditional object detection [29] and image segmentation [30] methods to construct the encoder based on the combination of backbone [31] and feature pyramid network (FPN) [32]. It is responsible for the extraction of the fused vision feature map $\mathbf{F}_f \in \mathbb{R}^{H/4,W/4,C}$ with multi-scaled context information, where H and W are the height and width of the input image I and C is the channel number of the feature map. The $\mathcal{F}_{det}(\cdot)$ and $\mathcal{F}_{cls}(\cdot)$ in sign detection and recognition modules take $\mathbf{F}_f$ as input to predict the sign locations $\{s_{loc}, p_{loc}, t_{loc}\}$ and the categories of symbols and guide panels $\{s_{cate}, p_{cate}\}$ in parallel, the process $\mathrm{Detection}(I)$ can be formulated as follows:

$$\begin{aligned} \mathrm{Detection}(I) &= \{s_{loc}, p_{loc}, t_{loc} | \mathrm{Classification}(I)\}, \\ \mathrm{Classification}(I) &= \mathrm{argmax}(P(cate_{\mathrm{Q}}|I)), \end{aligned} \tag{5}$$

where Q is the category number of the traffic symbol, guide panel, and text. It is worth noting that the whole text is considered as one category no matter what characters it includes in this recognition process. Therefore, the Q is set as 164 (including 155 symbol categories, 7 guide panel categories, text category, and background category) in this paper. $P(cate_{\mathrm{Q}}|I)$ is the probability of classifying the image into Q categories at pixel-level individually. $s_{loc}, p_{loc}, t_{loc}$ are the locations of text and different categories of symbols and guide panels that are extracted from pixel-level classified results $\mathrm{Classification}(I)$.

For the text recognition, $\mathcal{F}_{rec}(\cdot)$ in the sign recognition module recognizes text t_{cate} (including Arabic numeral, English, Chinese, and the other special characters) according to fused feature map $\mathbf{F}_f$ and text locations t_{loc} gradually, the corresponding process $\mathrm{Recognition}(I_{t_{loc}})$ can be expressed as:

$$\mathrm{Recognition}(I_{t_{loc}}) = \prod_{i=1}^{j} \mathrm{argmax}(P(c_i^{\mathrm{T}}|I_{t_{loc}})), \tag{6}$$

where $I_{t_{loc}}$ is the cropped text features that is extracted from $\mathbf{F}_f$ and the T denotes the number of character categories. T is set as 1085 (including 10 Arabic numerals, 52 English characters, 995 Chinese characters, and the other 28 special characters) in this paper. j is the number of characters in a specific $I_{t_{loc}}$, which varies for different $I_{t_{loc}}$. $P(c_i^{\mathrm{T}}|I_{t_{loc}})$ is the probability of classifying the i-th character in $I_{t_{loc}}$ into T categories.

With the sign detection and recognition results, the sign interpretation module first determines the spatial relative dependence of signs and groups symbols and texts that belong to the same sign together to generate chaotic sign clusters. It then organizes the internal semantic logic h of signs according to the spatial relative dependence and chaotic sign clusters via $\mathcal{H}(q)$. In the end, the module feeds h and the detected and recognized signs q into $\mathcal{T}(h, q)$ for generating natural language descriptions with integrity traffic instruction information. The sign detection, recognition, and interpretation modules will be described in detail following.

B. Sign Detection Module

As illustrated before, the sign detection module takes the vision feature map $\mathbf{F}_f$ from the encoder (can be referred to Fig. 10 and Section V-A) as input for detecting traffic signs to help the model focus on the sign regions from road scenes in the interpretation process. Since the $\mathbf{F}_f$ is concatenated by multi-scaled feature maps from the backbone, there exists a feature discontinuity problem in $\mathbf{F}_f$. Therefore, the sign detection module is designed to smooth the concatenated feature to generate a smoothed feature map $\mathbf{F}_{sd}$ at first and then execute the sign detection task on it. The module structure will be introduced in detail next.

Module structure. The structure can be found in the left column of Fig. 11, which is designed for locating signs from road scene images. The function $\mathcal{F}_{det}(\cdot)$ in this module represents the detection mapping to be learned. It consists of a smooth convolutional layer and a segmentation head. The former is used for smoothing the gap between high-level and low-level features via the stack of a 3×3 filer, BN [33], and ReLU [34] to generate smooth feature map

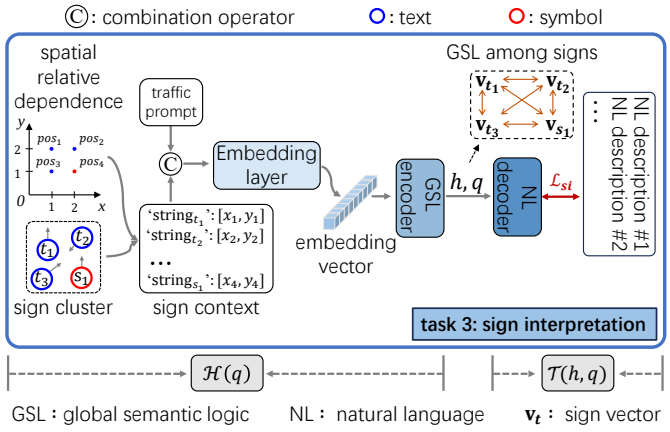


Fig. 12: The details of the sign interpretation module structure, where the sign cluster consists of recognized sign results.

$\mathbf{F}_{sd} \in \mathbb{R}^{H/4,W/4,C}$. The head takes $\mathbf{F}_{sd}$ to segment shrink-masks ($\mathbf{M}_s, \mathbf{M}_p$, and $\mathbf{M}_t \in \mathbb{R}^{H,W,1}$) of signs through two transposed convolutions with 2×2 filer and binarization operator $\mathcal{B}$. The final bounding boxes of signs are obtained by expanding shrink-masks to original regions.

Training stage. The label of shrink-mask is generated by shrinking the sign bounding box using the polar minimum distance algorithm [35]. The shrinking offset d_s is computed from the coordinates of the sign center point and a series of dense original sign box points:

$$d_s = \frac{1}{2}\min(\|p_{cp}, p_m\|_2^2), m = 1, 2, ..., M, \tag{7}$$

where p_{cp} and p_m are the coordinates of the ground-truth sign center point and the ground-truth sign dense box points. M is the number of ground-truth dense box points. Based on sign shrink-masks, the optimization function of the detection module can be formulated as follows:

$$\mathcal{L}_{det} = \mathcal{L}_{seg}^s + \mathcal{L}_{seg}^p + \mathcal{L}_{seg}^t, \tag{8}$$

where Dice loss [36] is adopted as $\mathcal{L}_{seg}$ in Eqn. 8.

Inference stage. The sign box can be obtained easily by expanding the predicted shrink-mask contour using the expanding distance d_e:

$$d_e = \min(\|p_{cp}, p_k\|_2^2), k = 1, 2, ..., K, \tag{9}$$

where p_{cp} and p_k are the coordinates of the predicted shrink-mask center point and the predicted shrink-mask dense contour points. K is the number of predicted dense contour points.

C. Sign Recognition Module

The same as the detection module, the sign recognition module generates a smoothed feature map $\mathbf{F}_{sr}$ at first. It then executes the sign classification task on $\mathbf{F}_{sr}$ via a convolutional layer-based multi-classification head. The lightweight convolutional head helps ease the optimization process while ensuring reliable classified results when training it with our TSI-CN dataset. Furthermore, considering the complexity of the text recognition task, a transformer-based encoder and decoder are adopted to translate visual features to the text string effectively with the help of the transformer's strong long-time contextual information extraction capability. The module structure is introduced in the following paragraphs.

Module structure. The structure can be found in the right column of Fig. 11, which is responsible for classifying the categories of symbols and guide panels and recognizing texts via $\mathcal{F}_{cls}(\cdot)$ and $\mathcal{F}_{rec}(\cdot)$, respectively. The same as $\mathcal{F}_{det}(\cdot)$, the function $\mathcal{F}_{cls}(\cdot)$ smooths multi-scaled features via a smooth layer and obtains the feature map $\mathbf{F}_{sr} \in \mathbb{R}^{H/4,W/4,C}$. It then predicts category probability maps of symbols $\mathbf{C}_s \in \mathbb{R}^{H/4,W/4,N_s}$ and panels $\mathbf{C}_p \in \mathbb{R}^{H/4,W/4,N_p}$ via two convolutions with 3×3 filer and softmax operator $\mathcal{S}$. Notably, the category classification and the sign detection are conducted in parallel, which helps simplify the framework's complexity. As for $\mathcal{F}_{rec}(\cdot)$, it is a function mapping for predicting character distribution probabilities. The function takes the flattened crop features that are extracted from $\mathbf{F}_{sr}$ according to text location information t_{loc} as input. It focuses on every single character via transformer encoder and decoding a series of character distribution probability vector $\mathbf{v}_t \in \mathbb{R}^{w,N_t}$ through transformer decoder, where the encoder and decoder consist of a multi-head attention layer [37] with 8 heads. The final recognized text can be transformed from $\mathbf{v}_t$ through the vec2word algorithm in [38].

Training stage. The label of the symbol $\mathbf{C}_s^{gt} \in \mathbb{R}^{H/4,W/4,N_s}$ and panel $\mathbf{C}_p^{gt} \in \mathbb{R}^{H/4,W/4,N_p}$ categories are composed of $\frac{H}{4} \times \frac{W}{4}$ one-hot classification vector, where N_s and N_s are the category number of symbol and panel, respectively. For the generation of a text label, we use the word2vec algorithm proposed in [38] to embed characters into vector representation $\mathbf{v}_t^{gt} \in \mathbb{R}^{w,N_t}$, where w and N_t are the lengths of the flatten features and vector dimension. For the recognition task of symbol, text, and panel, we use cross-entropy loss to formulate the optimization function:

$$\mathcal{L}_{rec} = \mathcal{L}_{ce}^s + \mathcal{L}_{ce}^p + \mathcal{L}_{ce}^t. \tag{10}$$

Inference stage. The symbol and panel category vectors are sampled from $\mathbf{C}_s$ and $\mathbf{C}_p$ by the symbol and panel box center positions at first. Then, category vectors are transformed into one-hot vectors via the softmax function and mapped to the corresponding categories with a category dict. The same as the symbol and panel's category prediction, a one-hot character vector is generated by the softmax function according to the outputted character distribution probability vector $\mathbf{v}_t$, and the character can be retrieved from a character dict.

D. Sign Interpretation Module

After obtaining the recognized signs, TSI-arch needs to combine them together to generate a natural language description of the signs for providing information support to drivers or autonomous driving systems. It is found according to Chinese design criteria of road traffic signs that the sign locations contain the corresponding semantic logic information, which is important for the sign interpretation. Therefore, the detected sign boxes are taken into the interpretation process with the recognized results together. Moreover, we follow the natural language processing to embed the detected and recognized

TABLE II: The detection performance of the models for dealing with traffic signs of different scales. '60' and '80' denote the input image is resized to 0.6 and 0.8 times the original size in the training process. 'S', 'M', and 'L' mean scaling the input image to 736, 1024, and 1280 along the short side in the inference stage.

Methods	symbol			text			panel		
	Precision	Recall	F-measure	Precision	Recall	F-measure	Precision	Recall	F-measure
TSI_{40-S}	60.82	45.62	52.13	66.10	53.01	58.83	89.33	90.28	90.07
TSI_{40-M}	76.04	63.55	69.23	75.02	66.66	70.59	82.73	93.43	87.76
TSI_{40-L}	76.60	67.83	71.95	76.18	72.37	74.23	78.61	84.68	85.09
TSI_{60-S}	63.97	48.11	54.92	67.44	54.22	60.11	88.70	89.26	88.98
TSI_{60-M}	74.58	65.74	69.88	76.27	69.05	72.48	82.26	91.87	86.80
TSI_{60-L}	80.29	77.09	78.66	76.46	76.90	76.68	74.57	92.91	85.14
TSI_{80-S}	66.37	43.82	52.79	66.11	50.89	57.51	90.08	85.19	87.57
TSI_{80-M}	75.69	60.16	67.04	73.69	65.31	69.25	86.75	91.45	89.04
TSI_{80-L}	80.11	71.81	75.74	74.59	71.57	73.05	81.77	92.60	86.85

results into vectors first. Then they are used to generate the corresponding descriptions by taking advantage of the large language model's language logic organization ability and generation ability. The interpretation module is introduced next in detail.

Module structure and inference stage. The structure can be found in Fig. 12, which is used for organizing the global semantic logic between detection and recognition signs q. This module combines spatial relative dependence $\{s_{loc}, t_{loc}, p_{loc}\}$ and sign cluster $\{s_{cate}, t_{cate}, p_{cate}\}$ for generating sign context first. Here, the spatial relative dependence is obtained from sign locations, and the distances between signs are normalized as 1. The sign cluster consists of recognition strings of signs (including symbols and texts) within the same guide panel. Then, the traffic prompt and sign context are embedded into vectors via the embedding layer that consists of a standard embedding operator implemented by Pytorch and an LSTM [39] layer. Next, $\mathcal{H}(\cdot)$, a stacked GLM block [40], takes the embedding vectors and sign context as input to analyze the global semantic logic among signs and to organize the logic as a hidden state h. In the end, $\mathcal{T}(\cdot)$ decodes the hidden state h to a series of word vectors through the combination of GLM blocks and a fully connected layer, and the final natural language description is generated by the vec2word algorithm.

Training stage. The data label is a natural language sentence that is organized based on signs within a guide panel according to the Chinese design criteria of road traffic signs. The optimization function $\mathcal{L}_{int}$ is formulated by cross-entropy loss just like the recognition loss function. The pre-trained sign interpretation module in [40] is fine-tuned on TSI-CN data to interpret traffic instruction as the natural language, where the likelihood $p(q, l)$ of generating a particular interpretation natural language sequence $l = (c_0, c_1, ..., c_{n-1}, c_n)$ based on recognized signs q can be represented as the following formula:

$$p(q, l) = \prod_{i=1}^{n} p(c_i \mid \mathcal{H}(q), c_0, c_1, \ldots, c_{i-1}). \quad (11)$$

Here, the output from $\mathcal{H}(q)$ is a semantic logic hidden state, closely resembling the natural language interpretation l. This means that the conditional information $\mathcal{H}(q)$ and the generated interpretation l are structurally similar, leading to a smaller difference between them. This similarity helps to facilitate the training of the model, making it more effective.

VI. Experiments

As described in Section V, the designed multi-task framework involves the three aspects of the detection, recognition, and interpretation of traffic signs. In this section, we verify the effectiveness of the framework on the TSI-CN dataset via the evaluation of all three aspects.

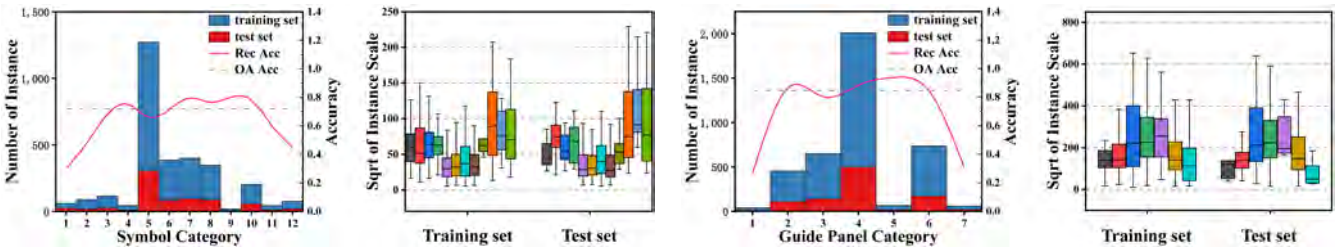
A. Implementation Details

In the experiments, the combination of ResNet-18 [31] and FPN is adopted as the encoder of TSI-arch, where the channel number of outputted fused vision feature map $\mathbf{F}_f$ is set to 64. In the data pre-processing stage, the training samples are increased by the following augmentation strategies: (1) random scaling (including image size and aspect ratio); (2) random rotating in the range of (-10°, 10°); (3) random cropping and padding. To optimize the proposed network, the Adam algorithm [41] is deployed to optimize the model. For learning rate, it is initialized as 0.001 and adjusted through the 'polylr' strategy [42]. All experimental results are obtained through training TSI-arch by 200 epochs and 24 batch sizes.

B. Results Analysis on Sign Detection

Here, we show the detection performance of TSI-arch on signs in Table II. As mentioned in Section IV, the TSI-CN dataset consists of large-scale images, which is beneficial to the effective detection of some small traffic signs (such as distant symbols and texts). However, the guide panel always enjoys a larger scale than the symbol and text (it even can be up to a quarter of the input image size), which makes it hard for TSI-arch to locate panels integrally and accurately. Therefore, the overall performance of the models for dealing with different scaled images are listed to explore the influence brought by image and instance scales.

Concretely, the input is resized to 0.4, 0.6, and 0.8 times the original size in the training process. For the inference stage, we scale the image to 736 (S), 1024 (M), and 1280 (L) along the



(a) symbol recognition performance (b) guide panel recognition performance

Fig. 13: The illustration of sign recognition performance and the distribution of instance number, and the distribution of instance scale.

TABLE III: The recognition performance of the model (TSI_{60-L}). Considering there are lots of different symbols in the TSI-CN dataset, some representative symbols are picked up to show the model performance (e.g., i78, i54, etc.) conveniently. 'OA' denotes the overall accuracy of all kinds of components. 'N', 'C', and 'E' are Arabic numerals, Chinese, and English characters.

symbol	i78	i54	i81	i79	a1	a3	a5	a7	p4	p34	p39	p36	OA
Acc	29.99	48.00	67.92	74.99	66.18	71.42	79.41	76.56	79.99	78.04	59.99	44.44	71.82
panel	0	1	2	3	4	5	6	OA	**text**	N	C	E	OA
Acc	26.67	86.46	80.24	88.35	93.75	84.35	30.76	84.75	**Acc**	60.19	62.15	60.90	62.93

short side. We observe that the detection performance of the model on symbols and texts gains improvements continually by tuning the image size larger in the inference process and keeping it fixed in the training stage, while it shows an opposite trend on the detection performance of the guide panel. It can be found that TSI_{60-L} achieves the best results on the detection of symbols and texts (78.66% and 76.68% in F-measure) and TSI_{40-S} achieves the best results on guide panel (90.05% in F-measure). The above experiments verify the proposed TIS-arch can detect different signs effectively and it is important for exploring the influence brought by image scales to different signs.

C. Results Analysis on Sign Recognition

Table III lists the recognition results of our model on signs. To show the performance of symbols conveniently, some representative of them (e.g., i78, i54, etc.) are picked out in this experiment. It is found that TSI-arch achieves superior performance of 71.82% in average accuracy on symbols, while the model performs badly on some of them (such as i78, i54, and p36). For the guide panel, a similar conclusion can be observed. To explore the reason behind this phenomenon, we analyze the relationship between the recognition performance and the distributions of instance number and scale in Fig. 13. It can be observed that large instance numbers can bring performance gains. Meanwhile, a big instance scale also can ensure our model achieves a competitive recognition accuracy on symbols even though there are few training samples in the TSI-CN dataset. For text recognition, we list the accuracy of different kinds of characters in Table III, since the complex stroke and categories, the recognition accuracy in Chinese is behind the numbers and English characters.

D. Results Analysis on Sign Interpretation

Different from previous traffic sign recognition works, the proposed TSI aims to interpret signs into natural languages with accurate and integrity instruction information, which provides support for the development of autonomous and assistant driving system. In the following experiments, we show the feasibility of the TSI task and the effectiveness of the proposed TSI-arch on the TSI-CN dataset.

Quantitative results. We explore the performance of the interpretation modules with different settings in Table IV. As described in Section V-D, it can be found that a suitable traffic prompt ("You are a text assembler with traffic experience. Please describe the traffic instruction based on the text, symbol, and their spatial relative dependence. The output must include all the input information, and can not be omitted.") brings gains in SA (1.70%). Meanwhile, it is observed that the combination of 'TP' and 'SRD' brings improvement in all metrics (Rouge (up to 2.0%), Bleu (up to 1.9%), SA (3.6%), and GPT-4 (0.68%)), where the performance gains in Bleu demonstrate the generated natural language descriptions enjoy high syntactic accuracy in all character grained ranging from 1 to 4. The improvements in Rouge verify the order of the generated natural language descriptions is consistent with the corresponding ground-truth sentences. The performance gains in the aspect of GPT-4 that can capture semantic equivalence and the order of words further verify the results in the measurement of Rouge and Bleu. The above results prove that 'SRD' can encourage the TSI-arch to organize the global semantic logic among signs and interpret them as natural language descriptions more accurately. This is mainly because the traffic text and symbol in the guide panel are distributed regularly according to Chinese design criteria of road traffic signs, which makes 'SRD' help our model to learn the global semantic logic among signs effectively.

Visualization results. As described in Section IV, the guide

Fig. 14: The visualization of the interpretation process and the global semantic logic-based natural language description of the instruction guide panel.

TABLE IV: Performance of the models with different settings based on the detected signs. 'R-1', 'R-2', 'R-*l*', 'B-1', 'B-2', 'B-3', 'B-4', and 'SA' are Rouge-1, Rouge-2, Rouge-*l*, Bleu-1, Bleu-2, Bleu-3, Bleu-4, and Soft Accuracy, respectively. 'SRD' and 'TP' are spatial relative dependence and traffic prompt.

Methods	R-1(%)↑	R-2(%)↑	R-*l*(%)↑	B-1(%)↑	B-2(%)↑	B-3(%)↑	B-4(%)↑	SA(%)↑	GPT-4(%)↑
Baseline	69.52	46.86	59.21	59.27	53.73	48.92	44.28	44.70	69.87
Baseline + TP	69.39	47.15	60.20	60.88	55.25	50.35	45.62	46.40	70.04
Baseline + TP + SRD	71.11	48.27	61.30	61.17	55.42	50.80	46.20	47.46	70.55

panel can be classified into seven categories (including highway instruction, urban way instruction, warning, prohibition, notice, scenic area, and emergency panels), which provides traffic instruction information with different types to drivers. In particular, warning and prohibition panels aim to regulate driving behavior and restriction of special roads and vehicles respectively, which relates to road safety and is important to every driver. Different from the above panels, the scenic area panel conveys scenic area information along the way, which keeps different importance for drivers with different purposes of driving. Considering the above situations, we interpret different kinds of panels separately and make it possible for drivers or autonomous driving systems to receive the information selectively that they need.

We show the interpretation process and the final results of the instruction guide panel in Fig. 14. Different from the other panels, the instruction guide panel enjoys complex semantic logic between multiple symbols and texts, which makes it hard to analyze the logic and interpret the traffic instruction via language description way. Considering this issue, TSI-arch follows the Chinese design criteria of road traffic signs to organize the logic combined with signs' spatial relative dependence, which helps to interpret the instruction guide panel accurately. We further visualize the interpretation results of warning, prohibition, notice, scenic area, and emergency, and combination panels in Fig. 15. It can be observed that the prohibition and emergency panels enjoy a very large aspect ratio, which is hard to detect and recognize integrally. Benefiting from the advantages of the pixel-level sign location process, our TSI-arch still be able to interpret them effectively.

E. Comparison with Related Methods

Considering TSI is a novel multi-task that can be divided into sign detection, text spotting, and language interpreting. To show the superior performance of the TSI-arch, we show the performance comparison with traditional object detection methods (e.g., Faster-RCNN [43] and YOLO [29]) on the detection task of symbol, text, and panel. Meanwhile, for the text spotting task, the proposed TSI-arch is compared with the current OCR methods (e.g., DB++ [44] and LeafText [46]).

Comparison with object detection and recognition methods. Our TSI-arch achieves superior comprehensive performance on most traffic sign objects. Specifically, as the introduction in Section IV, the TSI task needs to detect traffic symbols, text, and guide panels to provide essential information for subsequent language interpreting. Since there is a dramatic scale range among the above three kinds of traffic signs, it is important for the model to recognize them simultaneously. However, for Faster-RCNN [43] and Mask-RCNN [30], they search for those objects rely on predefined anchors, which makes it hard to match signs with different scales. Moreover, the large aspect ratio of the traffic symbol and text accelerates the decline of the recall rate of these methods. Different from them, TSI-arch segments sign regions at the pixel level to locate them, which helps our model be sensitive to different scaled and shaped signs. It ensures a high recall rate and promotes TSI-arch to outperform Faster-RCNN and Mask-RCNN a lot. Though YOLO [29] introduces a more effective feature extraction network (FEN) and feature pyramid network (FPN), the anchor mechanism limits this advanced detector to achieve superior accuracy on our TSI-CN dataset with a dramatic range of sign scales and shapes.

Comparison with sign detection and recognition meth-

TABLE V: Performance comparison with related methods on different sub-tasks. 'F' and 'Acc' are F-measure and accuracy metrics.

Type	Methods	symbol		text		panel	
		Det-Acc(%)↑	Rec-Acc(%)↑	Det-Acc(%)↑	Rec-Acc(%)↑	Det-Acc(%)↑	Rec-Acc(%)↑
object detection and recognition	Faster-RCNN [43]	49.08	43.86	74.78	–	60.21	59.38
	Mask-RCNN [30]	54.31	52.88	76.20	–	76.92	77.83
	YOLO7 [29]	69.65	70.02	80.38	–	82.81	83.90
sign detection and recognition	FM [3]	51.75	44.80	–	–	–	–
	DFR-TSD [5]	48.35	39.09	–	–	–	–
	VATSD [2]	53.66	52.17	–	–	–	–
text spotting	DB++ [44]	–	–	59.96	40.22	–	–
	ZTD [45]	–	–	61.78	43.95	–	–
	LeafText [46]	–	–	68.04	46.45	–	–
multi-task	TSI (**Ours**)	78.66	71.82	76.68	62.93	85.14	84.75

Fig. 15: The visualization of the interpretation process and the global semantic logic-based natural language description of the guide panels of warning, prohibition, notice, scenic area, emergency, and combination.

ods. The proposed TSI-arch outperforms the latest sign detection and recognition methods a lot in both detection and recognition accuracy simultaneously. Note that, considering the lack of open-source model codes, the sign detection and recognition models used to compare with our method in Table V are accomplished by ourselves. FM [3] is constructed based on YOLO [47] and VGG [48]. The same as traditional object detection frameworks, the anchor mechanism limits FM to detect and recognize symbols with varied aspect ratios and scales accurately. DFR-TSD [5] and VATSD [2] focus on small sign detection problems and image enhancement respectively. They still are not good at dealing with the symbols with varied aspect ratios and scales that occur in real road scenes. Different from them, our model is optimized by the detection and recognition of signs with different sizes, which helps TSI-arch learn multi-scaled features simultaneously and achieves a superior performance.

Comparison with text spotting methods. The proposed multi-task method outperforms existing methods 8% and 16% in the accuracy of detection and recognition (can be observed in Table V), respectively. To accomplish the text recognition task and ensure a fair comparison environment, we combine the latest text detectors (including DB++ [44], ZTD [45], and LeafText [46]) and the transformer-based text recognizer that designed in this paper. Since there are lot of text-related symbols (like speed limitation symbol, etc) in TSI-CN dataset, the recognition of them helps our method learn more text information than above text spotting only methods, which makes the proposed multi-task TSI-arch performs better for text spotting.

F. Model Analysis

Considering there lack of an effective architecture among existing works, the multi-task framework (TSI-arch) is designed for the proposed TSI task. To analyze the performance of TSI-arch comprehensively, we show the model performance with different settings in the following experiments.

Influence of the shrinking ratio. Different from existing anchor-based detectors, to cover traffic signs with a dramatic range of sizes and aspect ratios as many as possible (the results

TABLE VI: The sign detection performance of TSI_{60-L} with different mask shrinking ratios.

shrinking ratio	symbol-Acc(%)↑	text-Acc(%)↑	panel-Acc(%)↑
0.3	78.20	75.52	83.93
0.4	78.66	76.68	85.14
0.5	75.06	73.24	85.55
0.6	71.33	68.35	87.36

TABLE VII: The sign detection performance of TSI_{60-L} with different training tasks.

Training task			Det-Acc(%)↑		
symbol	text	panel	symbol	text	panel
✔	✔	✘	75.73 (2.93%↓)	75.16 (1.52%↓)	–
✔	✘	✔	76.54 (2.12%↓)	–	83.98 (1.16%↓)
✘	✔	✔	–	76.12 (0.56%↓)	83.31 (1.83%↓)
✔	✔	✔	78.66	76.68	85.14

in Table V verifies the effectiveness of shrink-mask strategy), TSI-arch adopts shrink-mask (can be referred in Section V-B) expanding strategy for rebuild sign bounding boxes. As shown in Table VI, we can find that the detection accuracy on the panel keeps rising with the growth of the shrinking ratio. This is mainly because the original guide panel sizes are too large. However, since the sizes of symbols and texts are smaller than guide panels, the model is hard to recognize them according to less information when the shrinking ratio is tuned from 0.3 to 0.6. It leads to our model achieving the best detection accuracy of symbols and texts when the shrinking ratio is set as 0.4 and the performance outperforms the others a lot. To pursue an superior comprehensive performance on all kinds of signs, we set the shrinking ratio to 5 in this paper.

Influence of the training task. As we described in Section V, considering the interrelated spatial relationship between traffic symbols, texts, and guide panels (such as texts always exist in symbols and guide panels or symbols occur in guide panels), we design TSI-arch to detect and recognize traffic symbols, texts, and guide panels simultaneously for reinforcing the performance of these sub-tasks each other. To analyze the performance gains brought by different tasks, we show the sign detection results of our model with different training tasks in Table VII. It can be observed from Table VII first row that the detection of guide panels brings significant improvements for detecting symbols and texts. This is mainly because of their spatial dependency on panels. In Table VII second and third rows, we can find that detecting symbols and texts can also enhance the model performance, which demonstrates the effectiveness and necessity of TSI-arch to integrate the detection tasks of symbols, texts, and guide panels into a unified architecture.

VII. Future Plan

As we introduced in Section III, different from previous sign recognition works, the proposed TSI task aims to convey accurate traffic instruction information via natural language, which helps understand traffic signs more comprehensively. The experiments in Section VI verify the TSI task is achievable and that the TSI framework can be well-trained based on the constructed TSI-CN dataset. Considering that the sign interpretation for autonomous driving is a dynamic process, where the information of the vehicle's location [49]–[51] and the road lane needs to be taken into the sign interpretation process to provide real-time traffic driving instructions, we plan to combine the proposed TSI task with the recognition of road lanes and egocentric locations for dynamically analyzing signs that are related to ourselves in the future, which is a more complicated task and needs a larger dataset. Therefore, we will enlarge the TSI-CN dataset in the aspect of the image number and labels simultaneously and try to augment the dataset via some existing augmentation techniques.

VIII. Conclusion

In this paper, we have presented a novel task, namely traffic sign interpretation (TSI), to interpret accurate traffic instruction information via natural language like a human from real road scenes, which can promote the progress of the traffic sign recognition community while providing support for the development of autonomous and assistant driving systems. Meanwhile, we explore and design a multi-task framework to detect and recognize traffic signs, analyze the internal semantic logic among them, and interpret them as a natural language. Furthermore, a TSI-CN dataset is constructed to fulfill the research and evaluation of the TSI framework, which encourages more researchers to participate in the research of the TSI task. Experimental results demonstrate the TSI task is achievable and the framework has achieved superior performance in all aspects of traffic sign detection, recognition, and interpretation. In future work, we will concentrate on the practical application environment and requirement-related challenges (e.g. dark light, motion blur, and key information retrieval) in the research of TSI.

References

[1] N. Gray, M. Moraes, J. Bian, A. Wang, A. Tian, K. Wilson, Y. Huang, H. Xiong, and Z. Guo, "Glare: A dataset for traffic sign detection in sun glare," *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[2] J. Wang, Y. Chen, X. Ji, Z. Dong, M. Gao, and C. S. Lai, "Vehicle-mounted adaptive traffic sign detector for small-sized signs in multiple working conditions," *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[3] J. Yu, X. Ye, and Q. Tu, "Traffic sign detection and recognition in multiimages using a fusion model with yolo and vgg network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 16 632–16 642, 2022.

[4] W. Cao, Y. Wu, C. Chakraborty, D. Li, L. Zhao, and S. K. Ghosh, "Sustainable and transferable traffic sign recognition for intelligent transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[5] S. Ahmed, U. Kamal, and M. K. Hasan, "Dfr-tsd: A deep learning based framework for robust traffic sign detection under challenging weather conditions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 5150–5162, 2021.

[6] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-sign detection and classification in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2110–2118.

[7] W. Min, R. Liu, D. He, Q. Han, Q. Wei, and Q. Wang, "Traffic sign recognition based on semantic scene understanding and structural traffic sign location," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 15 794–15 807, 2022.

[8] X. Rong, C. Yi, and Y. Tian, "Recognizing text-based traffic guide panels with cascaded localization network," in *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part I 14*. Springer, 2016, pp. 109–121.

[9] J. Hou, X. Zhu, C. Liu, C. Yang, L. Wu, H. Wang, and X. Yin, "Detecting text in scene and traffic guide panels with attention anchor mechanism," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 11, pp. 6890–6899, 2021.

[10] Y. Guo, W. Feng, F. Yin, T. Xue, S. Mei, and C.-L. Liu, "Learning to understand traffic signs," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 2076–2084.

[11] Y. Guo, F. Yin, X.-h. Li, X. Yan, T. Xue, S. Mei, and C.-L. Liu, "Visual traffic knowledge graph generation from scene images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 21 604–21 613.

[12] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural networks*, vol. 32, pp. 323–332, 2012.

[13] Y. Yang, H. Luo, H. Xu, and F. Wu, "Towards real-time traffic sign detection and classification," *IEEE Transactions on Intelligent transportation systems*, vol. 17, no. 7, pp. 2022–2031, 2015.

[14] D. Tabernik and D. Skočaj, "Deep learning for large-scale traffic-sign detection and recognition," *IEEE transactions on intelligent transportation systems*, vol. 21, no. 4, pp. 1427–1440, 2019.

[15] S. Maldonado-Bascón, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gómez-Moreno, and F. López-Ferreras, "Road-sign detection and recognition based on support vector machines," *IEEE transactions on intelligent transportation systems*, vol. 8, no. 2, pp. 264–278, 2007.

[16] K. Lu, Z. Ding, and S. Ge, "Sparse-representation-based graph embedding for traffic sign recognition," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1515–1524, 2012.

[17] D. Cireşan, U. Meier, J. Masci, and J. Schmidhuber, "A committee of neural networks for traffic sign classification," in *The 2011 international joint conference on neural networks*. IEEE, 2011, pp. 1918–1921.

[18] P. Sermanet and Y. LeCun, "Traffic sign recognition with multi-scale convolutional networks," in *The 2011 international joint conference on neural networks*. IEEE, 2011, pp. 2809–2813.

[19] J. Jin, K. Fu, and C. Zhang, "Traffic sign recognition with hinge loss trained convolutional neural networks," *IEEE transactions on intelligent transportation systems*, vol. 15, no. 5, pp. 1991–2000, 2014.

[20] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan, "Perceptual generative adversarial networks for small object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1222–1230.

[21] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE, 2010, pp. 2963–2970.

[22] Q. Ye and D. Doermann, "Text detection and recognition in imagery: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 7, pp. 1480–1500, 2014.

[23] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Reading text in the wild with convolutional neural networks," *International journal of computer vision*, vol. 116, pp. 1–20, 2016.

[24] C. K. Ch'ng and C. S. Chan, "Total-text: A comprehensive dataset for scene text detection and recognition," in *Proceedings of the International Conference on Document Analysis and Recognition*, vol. 1, 2017, pp. 935–942.

[25] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1083–1090.

[26] D. Karatzas, L. Gomez, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. Chandrasekhar, and S. Lu, "Icdar 2015 competition on robust reading," in *Proceedings of the International Conference on Document Analysis and Recognition*, 2015, pp. 1156–1160.

[27] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*, 2004, pp. 74–81.

[28] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[29] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 7464–7475.

[30] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

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

[32] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[33] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015, pp. 448–456.

[34] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. AISTATS*, 2011, pp. 315–323.

[35] C. Yang, M. Chen, Z. Xiong, Y. Yuan, and Q. Wang, "Cm-net: Concentric mask based arbitrary-shaped text detection," *IEEE Transactions on Image Processing*, vol. 31, pp. 2864–2877, 2022.

[36] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 Fourth International Conference on 3D Vision*, pp. 565–571.

[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[38] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[39] A. Graves and A. Graves, "Long short-term memory," *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012.

[40] Z. Du, Y. Qian, X. Liu, M. Ding, J. Qiu, Z. Yang, and J. Tang, "GLM: general language model pretraining with autoregressive blank infilling," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, 2022, pp. 320–335.

[41] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[42] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 325–341.

[43] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[44] M. Liao, Z. Zou, Z. Wan, C. Yao, and X. Bai, "Real-time scene text detection with differentiable binarization and adaptive scale fusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 919–931, 2023.

[45] C. Yang, M. Chen, Y. Yuan, and Q. Wang, "Zoom text detector," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[46] ——, "Text growing on leaf," *IEEE Transactions on Multimedia*, 2023.

[47] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[48] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[49] A. K. Aggarwal, "Gps-based localization of autonomous vehicles," in *Autonomous Driving and Advanced Driver-Assistance Systems (ADAS)*. CRC Press, 2021, pp. 437–448.

[50] A. Kumar, T. Oishi, S. Ono, A. Banno, and K. Ikeuchi, "Global coordinate adjustment of 3d survey models in world geodetic system under unstable gps condition," in *20th ITS World CongressITS Japan*, 2013.

[51] A. K. Aggarwal, "A hybrid approach to gps improvement in urban canyons"," 2023.

Chuang Yang received the B.E. degree in automation and the M.E. degree in control engineering from Civil Aviation University of China, Tianjin, China, in 2017 and 2020 respectively. He is currently working toward the Ph.D. degree in the School of Computer Science and School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision, embodied AI, and intelligent transportation.

Kai Zhuang received the B.E. degree in computer science and technology and the M.E. degree in computer technology from Yanshan University, Qinhuangdao, China and Northwestern Polytechnical University, Xi'an, China, in 2020 and 2023 respectively. He is currently working toward the Ph.D.degree in the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision, and machine learning.

Mulin Chen received the B.E. degree in software engineering and the Ph.D. degree in computer application technology from Northwestern Polytechnical University, Xi'an, China, in 2014 and 2019 respectively. He is currently a researcher with the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His current research interests include computer vision and machine learning.

Haozhao Ma received the B.E. degree in computer science and technology from Hefei University of Technology, Hefei, China, in 2022. He is currently pursuing the M.E. degree in the School of Software and School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and machine learning.

Xu Han received the B.E. degree in information and computing sciences from Northeast Agricultural University, Harbin, China, in 2021. He is currently pursuing the Ph.D. degree with the School of Computer Science and School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision, pattern recognition and text detection.

Tao Han received a B.E. degree in transportation equipment and control engineering and an M.S. degree in computer science and technology from Northwestern Polytechnical University, Xi'an, China, in 2019 and 2022. He is currently pursuing a Ph.D. degree in computer science and engineering at the Hong Kong University of Science and Technology. His research interests include computer vision, ai4science, and AIGC.

Changxing Guo received the B.S. degree in computer science and technology from Taiyuan University of Technology, Taiyuan, China, in 2017 and the M.S. degree in computer science and technology from Northwestern Polytechnical University, Xi'an, China, in 2024. His research interests include computer vision, autonomous driving and traffic sign detection.

Han Han received the B.E. degree in computer science from Northeast Forestry University, Harbin, China in 2023. He is currently working toward the master degree at the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and deep learning.

Bingxuan Zhao received the B.Sc. degree in Information and Computing Science from Northwestern Polytechnical University, Xi'an, China, in 2024. He is currently working toward the Ph.D. degree in the School of Computer Science and School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and machine learning.

Qi Wang (M'15-SM'15) received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2005 and 2010, respectively. He is currently a Professor with the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision, pattern recognition and remote sensing.