IEEE Open Journal of
Intelligent Transportation
Systems

Received 25 May 2023; revised 5 July 2023; accepted 11 July 2023. Date of publication 25 July 2023; date of current version 17 August 2023.

Digital Object Identifier 10.1109/OJITS.2023.3298031

Robust Perception and Visual Understanding of Traffic Signs in the Wild

RODOLFO VALIENTE[1], DARREN CHAN[1], ALAN PERRY[1], JOSHUA LAMPKINS[1], SASHA STRELNIKOFF[1], JIEJUN XU[1], AND ALIREZA ESNA ASHARI[2]

[1]ISL, HRL Laboratories, LLC, Malibu, CA 90265, USA
[2]Research and Development, General Motors, Warren, MI 48092, USA

CORRESPONDING AUTHOR: R. VALIENTE (e-mail: rvalienteromero@hrl.com)

This work was supported by General Motors.

ABSTRACT As autonomous vehicles (AVs) become increasingly prevalent on the roads, their ability to accurately interpret and understand traffic signs is crucial for ensuring reliable navigation. While most previous research has focused on addressing specific aspects of the problem, such as sign detection and text extraction, the development of a comprehensive visual processing method for traffic sign understanding remains largely unexplored. In this work, we propose a robust and scalable traffic sign perception system that seamlessly integrates the essential sensor signal processing components, including sign detection, text extraction, and text recognition. Furthermore, we propose a novel method to estimate the sign relevance with respect to the ego vehicle, by computing the 3D orientation of the sign from the 2D image. This critical step enables AVs to prioritize the detected signs based on their relevance. We evaluate the effectiveness of our perception solution through extensive validation across various real and simulated datasets. This includes a novel dataset we created for sign relevance that features sign orientation. Our findings highlight the robustness of our approach and its potential to enhance the performance and reliability of AVs navigating complex road environments.

INDEX TERMS Autonomous systems, sign detection, sign recognition, sign relevance.

I. INTRODUCTION

As SELF-DRIVING cars become more widespread, they will require new capabilities that allow them to navigate more ambiguous or difficult driving situations such as construction zones or accident sites.

Traffic signs play a crucial role in ensuring reliable and smooth traffic flow. They provide essential information to drivers, including warnings, regulations, and directions. Understanding traffic signs can be challenging for autonomous vehicles (AVs) due to various factors such as robustness and sign relevance, as illustrated in Figure 1. These challenges stem from the dynamic and complex nature of real-world driving environments, and addressing them is critical for ensuring the efficient operation of AVs. One significant limitation of current self-driving technology is that it lacks the ability to detect, recognize, and interpret uncommon or temporary traffic signs. This limitation restricts their application to more limited environments that have already been mapped or modeled.

Precise and robust traffic sign detection and recognition are essential for sign understanding. However, this presents a challenge due to factors such as the varied nature of signs and unpredictable scenarios. For instance, traffic signs are exposed to various environmental conditions, including changing lighting and weather, that can affect clarity and visibility. Moreover, signs can be partially occluded, vandalized, or damaged.

Aside from detecting and recognizing traffic signs, the AV must determine whether a traffic sign is relevant to its planned path. To address this challenge, we introduce a novel approach for estimating the sign's relevance to the AV, enabling the AV to prioritize detected signs.

Despite significant advances in recent years, the development of robust sign perception systems remains a challenging task [1], [2], [3], [4]. An AV that fails to recognize a relevant stop sign, for example, can cause fatal accidents. As

The review of this article was arranged by Associate Editor Xin Xia.

FIGURE 1. Illustration of some of the challenges of understanding traffic signs for autonomous vehicles.

such, it is crucial to develop a perception system capable of robustly detecting, recognizing, and estimating the relevance of various traffic signs. To address this need, we propose a unified approach that integrates sign detection, and sign relevance together with text extraction and recognition, along with comprehensive experimental analyses across multiple datasets. We focus on robust traffic sign perception and visual understanding, to improve the performance of AVs navigation in complex environments.

Figure 2 showcases a sample of the datasets employed to evaluate our pipeline and an overview of the essential perception tasks, highlighting the diversity of traffic signs. The examples illustrate traffic signs found on both highways and urban streets. Signs may contain symbols, text, or both. Certain signs convey information about current driving conditions (such as speed limit signs), whereas others provide information irrelevant to AVs (such as signs to encourage seatbelt use). This wide array of scenarios presents significant challenges, which we address with the following contributions:

- We introduce a comprehensive and robust sign perception system that seamlessly incorporates sign detection, sign relevance, text extraction, and recognition.
- We propose a novel approach for estimating the relevance and 3D direction of traffic signs from 2D images, which allows AVs to prioritize relevant signs.
- We validate the robustness of our perception pipeline across different real and simulated datasets. In particular, we created a comprehensive dataset with a novel sign orientation feature, which provides the angular orientation of traffic signs relative to the ego vehicle. To the best of our knowledge, we are the first to create a dataset specifically designed to address sign orientation, enabling a more accurate assessment of our perception pipeline.

Current solutions have demonstrated feasibility in detecting and understanding traffic signs. However, they are often limited to a specific subset of signs, predominantly stop signs and traffic lights [5]. In contrast, our research aims to develop a system that can accurately perceive and interpret arbitrary traffic signs, an objective that extends significantly beyond current systems. Our model incorporates not only detection and recognition but also sign relevance estimation and text extraction, thus providing a more comprehensive understanding of traffic signs and their implications. Our experiments demonstrate that our approach outperforms state-of-the-art solutions across multiple datasets, which can significantly improve the performance and reliability of AVs.

II. RELATED WORK

A. SIGN DETECTION

Traffic sign detection is essential for AVs, as it enables them to recognize and interpret traffic signs, thereby improving their overall driving performance. There has been a significant amount of research on traffic sign detection over the years, with various techniques proposed to improve accuracy and speed [1], [6]. Popular object detection algorithms such as Fast-RCNN [7], Mask R-CNN [8], ViT [9] and YOLO [10] have been re-purposed for sign detection. Recent advancements in deep learning have significantly improved the accuracy of these techniques, enabling them to achieve state-of-the-art performance on various datasets. In that direction, Cao et al. [11] proposed a sparse R-CNN utilizing residual connections in the ResNet backbone and a self-attention mechanism to handle foggy, frosty, and snowy images. Meanwhile, Zhang et al. [12] introduced a cascaded R-CNN with multiscale attention, which employs data augmentation to balance the class prevalence of small signs often missed. The method is specifically designed to reduce false detections caused by illumination variation and adverse weather conditions. These detectors are trained on a small set of common signs, and they assign observed signs to their best match within this set. Consequently, they cannot identify signs outside this fixed, predetermined set. This is a severe limitation in real-world driving scenarios, as traffic signs can exhibit countless shapes and forms not accounted for during training.

Several studies have explored ways to enhance the accuracy of traffic sign recognition systems, particularly focusing on error detection. Hacker and Seewig [13] proposes a comprehensive approach that combines various techniques. Their method, tested on a traffic sign recognition task using self-created 3D driving scenarios, successfully handled Deep Neural Network (DNN) errors associated with in-distribution, out-of-distribution, and adversarial data. A method for traffic sign recognition that is robust against camera failures is proposed in [14]. Through experimental evaluation using three public datasets and artificially injected camera failures, they demonstrate that using a sliding window of frames, rather than a single frame, significantly enhances the robustness of DNN classifiers against compromised frames. They further corroborate their findings using explainable AI techniques to understand the variable performance of different classifiers under camera failure conditions. Similarly, Geissler et al. [15] introduces a plausibility-based fault detection method for high-level fusion perception systems in autonomous driving applications. Although these studies substantially improve system robustness, they primarily focus on bolstering robustness

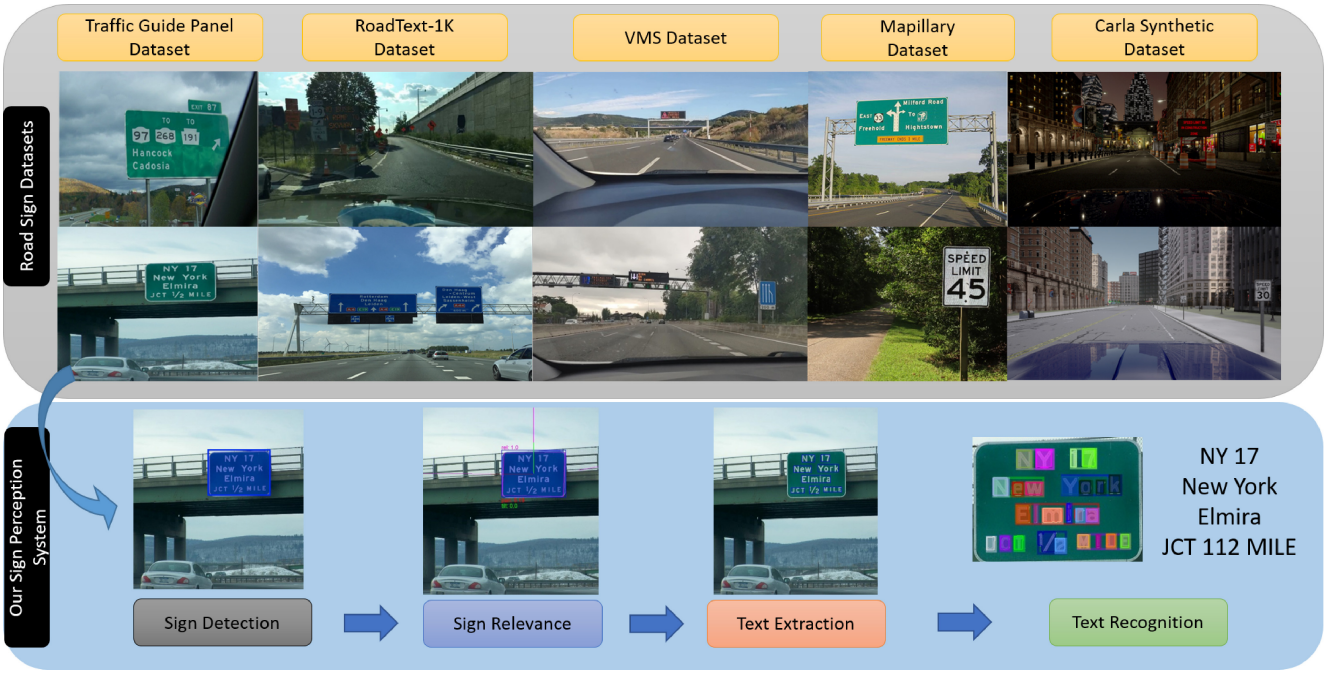


FIGURE 2. Illustration of diverse traffic sign images from the datasets tested and an overview of the components of our method.

through error detection and do not fully consider broader scenarios or diverse datasets.

B. SIGN RELEVANCE

Sign relevance is a critical visual processing task that enables AVs to prioritize detected signs based on their significance to driving. AVs must accurately determine which signs are relevant to the current driving conditions and filter out any irrelevant information. In this study, we specifically focus on sign orientation and introduce a novel method for visually estimating the 3D direction and relevance of traffic signs from 2D images. While the majority of existing work focuses on sign detection and recognition, estimating the relevance of signs has received less attention. Several studies have shown that visual features such as color, contrast, shape complexity, and size, can influence the salience and detectability of traffic signs, but algorithms that estimate sign relevance are limited. For instance, Greer et al. [16] conducted a preliminary study on traffic sign salience recognition and introduced a novel dataset that considers sign salience. Their CNN predicts the sign salience with 76% accuracy. However, the study is limited to a small map area and does not consider the orientation of the sign.

Some AV systems try to match perceived signs with geolocated signs from a map to estimate the sign relevance for the ego vehicle. However, this approach is limited if communication with the map is lost, if the region is not covered by the map, or if the environment undergoes changes that are not reflected in the map. For example, incidents such as road construction or accidents temporarily alter the environment and it is infeasible to update the map in real time.

C. TEXT EXTRACTION

Image text extraction involves locating text within images. Numerous techniques have been proposed to address this problem, ranging from traditional feature-based methods to more advanced deep learning-based approaches. However, it remains a challenging problem because of the varying quality and complexity of real-world road images, e.g., blurring artifacts due to motion, lighting that affects text contrast, etc. Some approaches are character-based and usually involve complicated character detectors such as Stroke Width Transform (SWT) and Maximally Stable Extremal Regions (MSER) followed by filtering [17], [18]. However, these methods require elaborate design, involve multiple stages of processing, and are time-consuming, leading to suboptimal performance due to error accumulation. Recent advancements in deep learning have shown significant promise in addressing these challenges, enabling more accurate and robust text extraction. In particular CNN [19] and instance-segmentation-based [20], [21] methods have been proposed. In this context, Ye et al. [21] proposed TextFuseNet, which exploits richer features fused for text detection by perceiving texts from different levels of feature representations, achieving robust arbitrary text detection. The multi-level feature representation adequately describes texts by dissecting them into individual characters while maintaining their general semantics.

D. TEXT RECOGNITION

Text recognition (TR) refers to reading text in images, converting text images to machine-readable strings, and it has been an important task in a wide range of applications such as digitizing printed books and documents, processing bank

FIGURE 3. Importance of text recognition in traffic signs for autonomous vehicles, highlighting the critical role it plays in ensuring reliable navigation, compliance with traffic regulations, and informed decision-making.

checks, recognizing license plates, and assisting visually impaired individuals [22], [23]. TR is especially important for AVs as it enables them to interpret crucial information, (shown in Figure 3). This ensures that AVs can navigate securely, adhere to traffic regulations, and make informed decisions in diverse situations. Furthermore, text recognition improves navigation and route guidance, as it allows AVs to identify street names, highway exit numbers, and other pertinent information.

One of the most common approaches is optical character recognition (OCR) [24], which involves segmenting the image into individual characters and identifying them with pattern recognition algorithms. These techniques achieve high accuracy in extracting text from high-quality images, e.g., images with high resolution and controlled lighting. However, they often struggle with low-quality images and handwritten text. Recent advancements have shown significant promise in addressing these challenges, enabling more accurate and robust text recognition. Baek et al. [25] introduce a unified scene text recognition (STR) approach, along with a text recognition benchmark including a new dataset and metrics. Luo et al. [26] leverage the improved results from using text rectification and combine the multi-object rectification network (MORN) and an attention-based sequence recognition network (ASRN) for general scene text recognition. The image is rectified by the MORN and given to the ASRN improving the final TR. For a comprehensive review, readers are encouraged to refer to [3], [4].

Despite their success, most existing methods for sign perception focus on detection and recognition. While this approaches suffices for common road signs, they fail in unexpected or uncommon situations involving complex signs or variable message signs containing customized text [2], [27]. Moreover, existing approaches are inadequate for real-world scenarios, where road signs can have countless shapes and forms not accounted for during training [1]. In that direction, authors Lampkins et al. [1] proposed MOSER (Multimodal rOad Sign intERpretation System for Autonomous Vehicles), a scalable solution that interprets arbitrary road signs using multimodal techniques. We build on top of that, improving robustness and considering sign relevance. In contrast with previous solutions, our method can estimate the relevance of various signs using 2D images, making it more versatile and adaptable to a broader array of situations.

While individual aspects of our research have been examined in isolation in prior work, one distinctive value of our study lies in the integration of all these components into one unified system, which we argue has not been sufficiently explored in the existing literature. We develop a robust sign perception system that seamlessly incorporates sign detection, sign relevance, text extraction, and recognition. Another important novelty of our work is our approach for estimating the relevance of traffic signs, and we created a unique dataset that includes a novel sign orientation feature that indicates the angular orientation of traffic signs relative to the ego vehicle. To the best of our knowledge, no other study has developed a dataset specifically designed to evaluate sign orientation.

III. PERCEPTION PIPELINE

The proposed architecture integrates sign detection, sign relevance, text extraction, and text recognition, as parts of our perception process. The system is fully modular, making it flexible and customizable. It accepts an image as input and outputs the detected sign locations along with their text and relevance. Our pipeline has the advantage of being able to process any type of text without making any assumptions about the specific sign categories that might be encountered, making it highly adaptable to a diverse range of driving scenarios. In the remainder of this section, we discuss each of these steps (illustrated in Figure 4), which are summarized as follows:

- Sign detection: The sign detector processes RGB images received from the camera and outputs sign instances (sign boxes and segmentation mask).
- Sign relevance: The sign relevance module processes the sign instances to estimate their relevance for the AV, which helps the AV to prioritize the detected signs.
- Text extraction: Each sign instance is fed into the text extraction module, which extracts regions of interest that contain text.
- Text recognition: Finally each text instance is fed to the text recognition module that converts the text instances to strings.
- The output is post-processed to obtain the final textual information. The text instances that lie within the bounding box of a sign instance are combined and rearranged into logical reading order (top-down, left-right) [1].

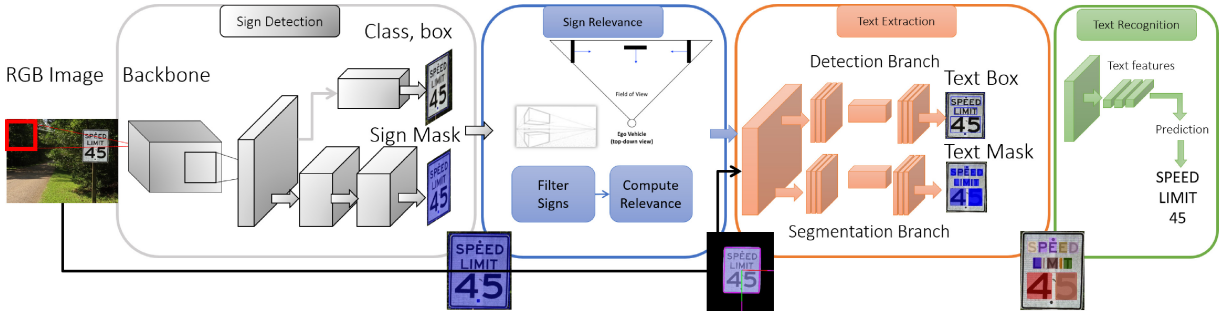


FIGURE 4. An illustration of our architecture's step-by-step process, including sign detection, sign relevance, text extraction, and text recognition.

FIGURE 5. Step-by-Step Example output: 1) displays the results after sign detection (sign instance masks, shown in the blue shaded region); 2) shows the outcomes of sign relevance, with the orientation of each extracted sign indicated by green and red line segments (green represents the y-directional eigenvector, and red represents the x-directional eigenvector); 3) illustrates the individual text extractor instances; and 4) demonstrate the processing of individual instances for recognition and the resulting text recognition outcomes.

A. SIGN DETECTION

Sign detection is the first step of the pipeline (Figure 4), which receives an image and generates sign instances, comprising sign boxes and segmentation masks (shown in Figure 5.1). Our sign detector is based on Mask R-CNN [8] which we modified and trained to detect generic sign shapes and text without any notion of specific sign categories. It simultaneously detects objects in the image while generating a high-quality segmentation mask for each instance. The first module of Mask R-CNN is the backbone, which functions as an image feature extraction network. The output of the feature extraction network is a set of feature maps, which are used to generate candidate regions in the next step. We use a ResNet50 architecture to extract features from the image and a Region Proposal Network (RPN) to generate candidate object bounding boxes. Once the candidate regions have been generated, the network is split into two heads: a classification and bounding box regression head, and a segmentation head. The classification and bounding box regression head classifies each region as containing a specific object and predicts the precise location of the object within the region. Finally, the segmentation head generates a mask for each detected object. For additional details, readers are encouraged to refer to He et al. [8].

To achieve scalability, we designed the sign detection network to extract generic signs, rather than those belonging to specific classes, e.g., stop, yield, etc. In other words, our method is not limited to the kinds of signs it can detect but instead extracts sign-like regions, characterized by bounding box locations, and a corresponding prediction confidence score. The model is trained to simultaneously optimize for object detection and instance segmentation by minimizing two losses: the RPN classification and regression loss, and the mask prediction loss.

FIGURE 6. Visually estimating the 3D direction of traffic signs from a 2D image mask.

The sign detector is a critical component for the sign relevance module described in the next section, as we require precise boundaries (segmentation mask) to compute the 3D orientation of the sign.

B. SIGN RELEVANCE

The sign relevance module processes the sign instances to compute the sign relevance and orientation, as shown in Figure 5.2. Our method introduces a novel approach for estimating the 3D direction and relevance of traffic signs from 2D images (shown in Figure 6), it estimates their orientation from the first-person perspective of the vehicle, which determines their relevance. It processes each individual sign instance and estimates their direction with respect to the vehicle's heading, then finally assigns a relevancy score (i.e., whether or not the sign is relevant to the ego vehicle).

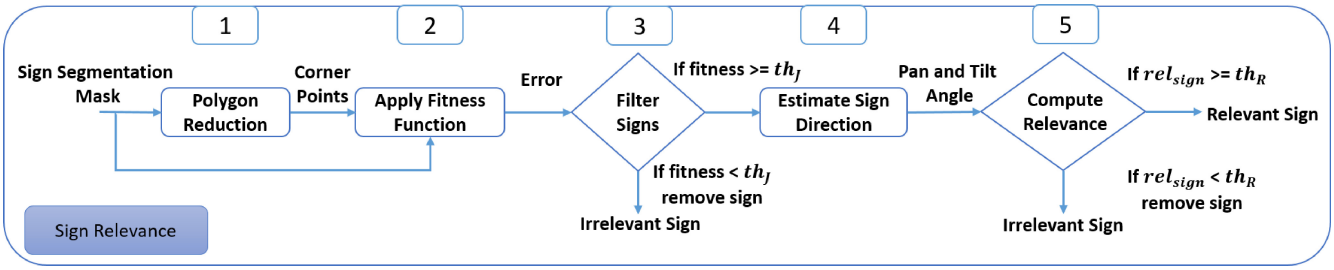


FIGURE 7. Flow chart of the sign relevance algorithm: 1) polygon reduction function, 2) apply fitness function, 3) filtering operation, 4) sign direction estimation from vanishing points, and 5) relevance threshold, to determine the relevance and orientation of a detected sign based on its segmentation mask.

Algorithm 1 Sign Relevance Algorithm

Input: s_m: Sign Mask; K: Intrinsic camera parameters; th_J: Fitness threshold;th_R: Relevance threshold

1-Polygon reduction
polygon $= f_{PolygonReduction}\ (s_m)$
2-Apply fitness function
Set A: sign mask, B: polygon mask
Compute $J(A, B) = \frac{|A \cap B|}{|A \cup B|}$
3-Filtering sign
if $J(A, B) >= th_J$ **then**
  Keep the sign
else
  Remove the sign → End
end if
4-Estimating the direction of the sign
$(V_x,\ V_y) = f_{VanishingPoints}$ (polygon, K) //compute the vanishing points $(V_x,\ V_y)$
Follow equations from 2-9 to compute pan (θ_x) and tilt angle (θ_y)
$(\theta_x,\ \theta_y) = f_{SignDirection}\ (V_x,\ V_y)$
5-Relevance Estimation
Compute relevance as $rel_{\text{sign}} = \cos(\theta_x)$
if $\text{rel}_{\text{sign}} >= th_R$ **then**
  The Sign is relevant → End
else
  The Sign is irrelevant → End
end if

The sign relevance algorithm receives the sign segmentation mask and determines whether the sign is relevant using the estimated 3D orientation (pan and tilt angle) of the sign. The algorithm consists of the following components (shown in Figure 7): 1) a polygon reduction function, 2) a polygon fitness function, 3) a filtering operation, 4) sign direction estimation from vanishing points, and 5) a relevance threshold. The pseudocode of our algorithm is presented in Algorithm 1.

1-Polygon reduction (Figure 7.1): After extracting the segmentation masks, they are fed into a contour-quadrilateral fit function, which reduces the number of contour points that form the sign boundaries to a simplified polygon. In essence, this removes noise and rough edges from the output of the segmentation algorithm. This step is essential for computing the direction of the signs since the contours form the lines that are used to compute vanishing points.

2-Apply fitness function (Figure 7.2): Occasionally, the segmentation masks are irregularly shaped, which often corresponds to false positives. In these situations, we measure the shape similarity between the fitted contour and the original contour to detect irregularly shaped objects. To determine the normalized fitness score $[0, 1]$ of two arbitrary masks, we compute their Jaccard Index, i.e., intersection over union of mask A and B as,

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

3-Filtering sign (Figure 7.3): After computing the Jaccard index between the segmentation masks and their fitted contours, we remove those with a low score using a fitness threshold. If $J(A, B) >= th_J$ we keep the sign, otherwise we remove it. The threshold (th_J) can be defined either empirically or with a data-driven approach. A higher similarity between two shapes (A and B) yields a higher Jaccard Index score. Irregular shapes will have a poor similarity to their fitted quadrilaterals.

4-Estimating the direction of the sign (pan and tilt) from two vanishing points (Figure 7.4): With the remaining contour points, we compute the horizontal and vertical vanishing points $(V_x,\ V_y)$ of the signs and use the intrinsic camera parameters K to recover the tilt (the elevation angle) and pan (the horizontal angle) [28].

To compute the vanishing point (V_i) using the edges, we first find the equations of the lines representing the edges and then compute the intersection point of the lines. From two edges, we first identify their endpoints (x_i, y_i). Then, we compute the line equations for both edges in slope-intercept form, $y = mx + b$, where m is the slope and b is the y-intercept. We calculate m_1, b_1 for the first edge and m_2, b_2 for the second edge. To compute the vanishing point $V_i = (V_i^x, V_i^y)$, in which (V_i^x, V_i^y) are the (x, y) coordinates for V_i vanishing point, we find the intersection of the two line equations:

$$m_1 x + b_1 = m_2 x + b_2 \tag{2}$$

When computing the vanishing point, if the slopes of the two edges, m_1 and m_2, are equal, the denominator becomes zero, leading to a singularity. To address this issue, we introduce a conditional check. If m_1 equals m_2, we can set the x-coordinate of the vanishing point (V_i^x) to a large predefined value that can represent infinity or consider this condition as an exception, handling it separately based on the constraints of the application. For instance, we can assign the relevance as unknown or a fixed value (if $m_1 = m_2$, we can assume that the sign is facing perpendicular to the vehicle, such that $\theta_x = 0$). This way, we avoid division by zero and accommodate the interpretation of parallel lines.

Solving for x:

if $m_1 = m_2$ **then**
  $V_i^x \leftarrow \infty$ //or handle according
else
  $V_i^x \leftarrow \frac{b_2 - b_1}{m_1 - m_2}$
end if

The value of V_i^x is substituted back into one of the line equations to find the corresponding y-coordinate V_i^y.

Next, the rotation matrix $R_{3\times3} = [r_1, r_2, r_3]$ can be computed using the intrinsic matrix ($K_{3\times3}$) and the 2 vanishing points (V_x, V_y). We then covert the vanishing points to camera coordinates (xc_{Vx},xc_{Vy}) with:

$$xc_{ximg} = K^{-1} \times \left[x_{img}^x, x_{img}^y, 1\right]^T \tag{3}$$

where x_{img}^x, x_{img}^y are the x,y location of the point in the image coordinate x_{img}. Then, $xc_{Vx} = K^{-1} \times [V_x^x, V_x^y, 1]^T$ and $x_{Vy} = K^{-1} \times [V_y^x, V_y^y, 1]^T$, with V_x^x, V_x^y been the x,y location of the horizontal vanishing point V_x and V_y^x, V_y^y the x,y location of the vertical vanishing point V_y. From that r_1, r_2 and r_3 are computed as follows,

$$r_1 = \frac{xc_{Vx}}{\|xc_{Vx}\|} = \frac{K^{-1}V_x}{\|K^{-1}V_x\|}; V_x = \left[V_x^x, V_x^y, 1\right]^T \tag{4}$$

$$r_2 = \frac{xc_{Vy}}{\|xc_{Vy}\|} = \frac{K^{-1}V_y}{\|K^{-1}V_y\|}; V_y = \left[V_y^x, V_y^y, 1\right]^T \tag{5}$$

$$r_3 = r_1 \times r_2 \tag{6}$$

From $r_3 = [r_{31}, r_{32}, r_{33}]^T$, we can compute the pan (θ_x) and tilt angle (θ_y) [28] as,

$$\theta_x = tan^{-1}(r_{31}, r_{33}) \tag{7}$$

$$\theta_y = sin^{-1}(r_{32}) \tag{8}$$

5-Relevance Estimation (Figure 7.5): Finally, we use the pan angle to estimate the relevance. We assume that signs are upright, so we disregard the tilt angle for relevancy estimation. We compute the cosine of the estimated pan angle, which yields a relevancy score rel_{sign} , i.e., normalized scalar in the interval [0, 1], corresponding to the direction of the sign with respect to the vehicle heading. For example, a relevancy score of 0 indicates that the sign is facing perpendicular to the vehicle (no relevance), while a relevancy score of 1 indicates that the sign is facing directly toward the vehicle (highest relevance). We then apply a relevance threshold (th_R) to the relevancy score to determine whether the sign is relevant; this threshold can be applied either empirically or using a data-driven approach:

$$\begin{aligned} rel_{\text{sign}} &= \cos(\theta_{\text{x}}) \\ \text{If } \text{rel}_{\text{sign}} >= th_R, &\quad \text{sign} \rightarrow \text{relevant} \\ \text{else}, &\quad \text{sign} \rightarrow \text{irrelevant} \end{aligned}$$

C. TEXT EXTRACTION

The text extraction module processes the sign instances to extract text instances and text regions of interest for each detected sign as shown in Figure 5.3. The text extractor is based on TextFuseNet [21], designed to efficiently detect and extract text from images. It employs three levels of feature representations: character, word, and global. This multi-level approach allows for a more comprehensive understanding of the text, while maintaining its overall semantics. It effectively aligns and merges features from different levels, producing a richer and more accurate representation of various text shapes. This reduces false positives and improves detection accuracy. The text extractor is specifically designed to segment text instances in images, identifying the precise coordinates that form the area of individual words. This precision is crucial for our system, as it enables us to accurately estimate logical text ordering.

D. TEXT RECOGNITION

The text recognition module receives text instances as input and outputs machine-readable strings as shown in Figure 5.4. Our text recognition module is based on the work of Baek et al. [25], which introduces a unified four-stage Scene Text Recognition (STR) framework. The STR framework involves four stages: spatial transformation, feature extraction, sequence modeling, and prediction. A Spatial Transformer Network (STN) normalizes the input text image to simplify downstream stages. Feature extraction maps the input image to a representation that emphasizes attributes relevant to character recognition while suppressing irrelevant features like font, color, size, and background. Sequence modeling captures contextual information within a sequence of characters, enabling more robust character prediction as opposed to independent predictions. Finally, prediction estimates the output character sequence based on the identified features of an image. In our implementation, image regions corresponding to individual words are fed into an STR model to convert each text instance into a machine-readable form (strings).

Following this, similar to Lampkins et al. [1], the sign text synthesizer integrates outputs from the sign detector, text extractor, and scene text recognizer into a unified structure. This module determines sign-text membership and governs the logical reading order of text through a five-step process. First, to establish sign-text membership, we compute

TABLE 1. Comparison of the various datasets used for evaluation, highlighting their distinct characteristics. Novel to our CARLA dataset, we include the camera parameters and annotations of sign orientations to evaluate sign relevance.

Dataset	Traffic Signs	Labeled Text	Complex Signs	Diverse Weather	Sign Orientation GT	Camera Parameters
TGP [29]	x	x				
RoadText-1K [30]	x	x	x	x		
VMS [31]	x	x				
Mapillary [32]	x		x	x		
CARLA (ours)	x			x	x	x

the overlapping region between text instances and sign-bounding boxes, with text instances fully encapsulated by a sign-bounding box being assigned as members of that sign. Second, to determine text order, we compute the orientation of each text instance using their image points and their x and y covariances, with estimated orientations obtained from the eigenvectors of those covariances. Third, the x-directional eigenvectors are extended to form line segments with endpoints intersecting their corresponding sign bounding box. Fourth, for any intersecting line segments, the corresponding text is appended to a list and reordered by increasing the x position, determining left-to-right ordering for a single text-readable line. Finally, if multiple text-readable lines exist within a sign instance, they are ordered by increasing the y-position, establishing a top-to-bottom order as shown in Figure 5.4.

IV. EXPERIMENTAL RESULTS

In this section, we detail the methodology for the evaluation of our perception system, focusing on sign detection, sign relevance, and text recognition. We begin by providing an overview of the datasets used for our experiments, followed by a description of the evaluation metrics and baseline models employed for comparison. Finally, we present and discuss our results.

A. DATASETS

We conducted our experiments on five diverse and challenging datasets, chosen to encompass a wide range of sign types to evaluate the robustness and performance of the perception pipeline in real-world situations (e.g., motion blur, lack of perspective capture bias, dynamic lighting conditions, etc). Table 1 summarizes the unique characteristics of each dataset.

Traffic Guide Panel (TGP) Dataset [29]: This dataset consists of 3,841 high-resolution images captured by car-mounted cameras in highway environments, with 2,315 images containing traffic guide panel annotations. The dataset includes various types of traffic guide panels like direction, toll plaza, destination distance, and exit indication. It contains mainly traffic panel signs collected in the U.S. and comes with text annotations.

RoadText-1K Dataset [30]: This dataset, designed for text detection in driving videos, consists of 1,000 unbiased video clips with annotated text bounding boxes and transcriptions. These 10-second video clips are sourced from the diverse and unconstrained BDD100K database, which includes 100K driving videos captured in various weather conditions, times of day, and locations across the United States.

VMS Dataset [31]: The Variable Message Signal (VMS) Spanish dataset consists of 1,216 instances, within 1,152 JPEG images, captured from inside a vehicle and focused on Spanish road Variable Message Signals. Annotations are provided in XML files in PASCAL VOC format, and a CSV file contains information about the geographic position, image location, and text annotations.

Mapillary Dataset [32]: This dataset is a large-scale, diverse collection of 25,000 high-resolution street-level images annotated into 66 object categories, with instance-specific labels for 37 classes. Captured using front-facing cameras mounted on moving vehicles, it encompasses a wide geographic range, including North and South America, Asia, and Europe.

CARLA Dataset. Recognizing the absence of existing datasets for evaluating sign relevance, we developed the CARLA dataset consisting of 2000 synthetic image frames, each featuring traffic signs. Novel to our dataset, we annotated each traffic sign with bounding boxes and 3D orientation, which we use to evaluate relevance (i.e., their directional heading with respect to the vehicle). Another novelty of our dataset is that traffic signs are positioned at various angle orientations along the road, ensuring a diverse and comprehensive representation. The ground truth (GT) pose of each sign is provided, which encompasses their orientation (rotation angle) and global position (world coordinates). The camera is mounted on the dashboard facing forward and the camera parameters are known.

Finally, as adverse weather affects the quality of images and consequently, the performance of the detection framework, we investigate the influence of different weather conditions on our system. Using the CARLA simulator, we have created five additional datasets under diverse weather settings: normal, cloudy, foggy, rainy, and night. Figure 8 shows a sample image of each dataset. Each dataset consists of 1000 synthetic image frames, with the respective annotation data. We used these datasets to assess the performance of our framework in detecting traffic signs and determining their relevance under varied weather conditions.

To assess the performance of traffic sign detection, we utilize datasets that provide GT bounding boxes for signs, i.e., the TGP, VMS, Mapillary, and CARLA datasets (Table 1). To evaluate text recognition performance, we

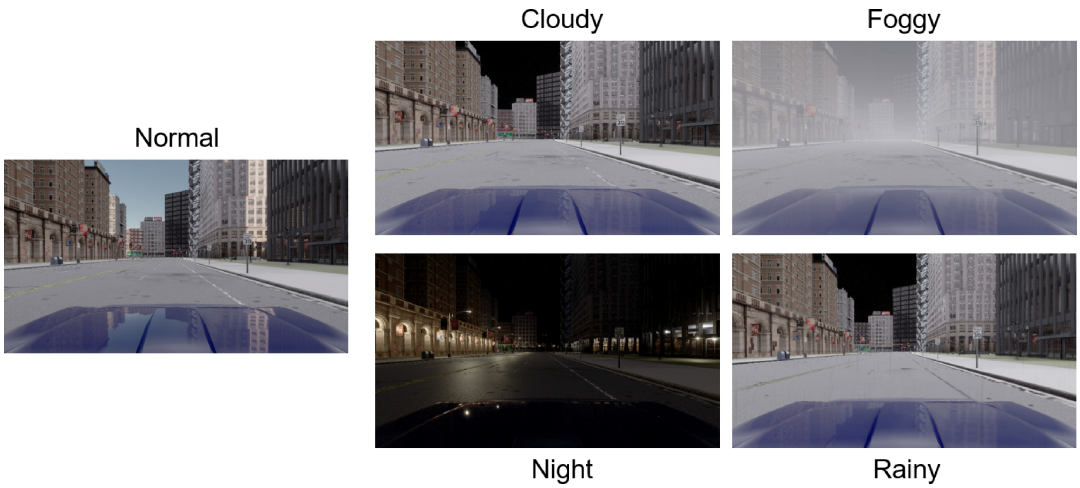


FIGURE 8. Sample images of the different CARLA datasets under diverse weather settings: normal, cloudy, foggy, rainy, and night.

employ the TGP, RoadText-1K, and VMS datasets because they contain text annotations. We evaluate sign relevance on the CARLA dataset, given that it is the sole testbed that includes GT sign orientations. By leveraging these datasets, we can effectively measure the performance of various aspects of traffic sign detection, sign relevance, and text recognition, paving the way for improvements in traffic sign interpretation.

B. EVALUATION METRICS

We evaluate the performance of our perception system based on:

- Sign detection, assessing the system's ability to identify and localize signs within an image.
- Sign relevance, assessing the system's ability to predict the sign relevance.
- Sign text recognition, assessing the system's ability to recognize text in the detected signs and convert it into machine-readable strings.
- Computation time, assessing the average time needed to process an input image from sign detection through text recognition.

Sign Detection: We assessed sign detection performance by measuring recall at different Intersections over Union (IoU) thresholds and computing the Area Under the Curve (AUC) [33]. Recall measures the proportion of positive cases (true positive instances of the sign) that the model correctly identified as positives out of all the actual positive cases:

$$\text{Recall} = \frac{\text{True Positives}}{(\text{True Positives} + \text{False Negatives})} \tag{9}$$

A larger AUC value indicates better sign detection, as it indicates that the model has high recall across different IoU thresholds or confidence scores. We do not evaluate precision because the datasets contain annotations for specific signs, while other signs are missing from the GT annotations. These missing signs, however, can be detected by our pipeline. For example, the dataset may include GT data for VMS signs but exclude GT information for all other sign types.

Sign text recognition: We assessed sign text recognition performance by measuring Word Recognition Rate (WER), Character Recognition Rate (CER), and Cosine Similarity. Because none of these metrics individually capture all the nuances of a model's performance we used multiple metrics. CER measures the ratio of the total number of character-level errors in the predicted text to the total number of characters in the reference (ground truth) text, i.e.,

$$\text{CER} = \frac{\text{Substitutions} + \text{Insertions} + \text{Deletions}}{\text{Total Reference Characters}} \tag{10}$$

WER measures the ratio of the total number of word-level errors in the predicted text to the total number of words in the reference (ground truth) text, i.e.,

$$\text{WER} = \frac{\text{Substitutions} + \text{Insertions} + \text{Deletions}}{\text{Total Reference Words}} \tag{11}$$

Finally, we use cosine similarity to measure the similarity between the two text strings by representing them as vectors and calculating the cosine of the angle between them. We convert the predicted text and the reference text into numerical vectors and compute their cosine similarity:

$$\text{Cosine Similarity} = \frac{\mathbf{A} \cdot \mathbf{B}}{||\mathbf{A}||\,||\mathbf{B}||} \tag{12}$$

A lower WER and CER value indicates better performance, as it implies that there are fewer character-level/word-level errors in the predicted text compared to the reference text. Similarly, a higher cosine similarity suggests improved performance.

Sign relevance: We assessed the performance of our sign relevance module by measuring sign angle estimation error, which is the difference between the GT sign angle and the angle estimated by the system. A smaller error value indicates that the estimated angle is closer to the ground truth, and thus, the system is more accurate in predicting the angle of the detected sign. In the context of sign detection, the

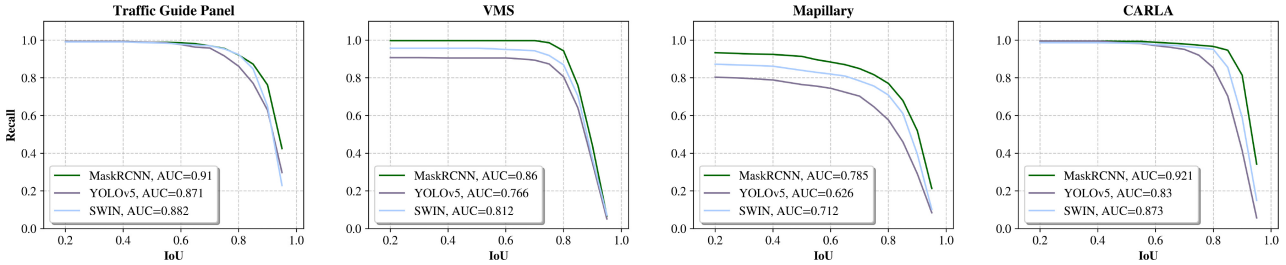


FIGURE 9. Comparison of Recall vs. IoU performance for three state-of-the-art object detectors on different datasets for the sign detection task, showcasing their respective effectiveness.

angle refers to the orientation of the sign in the image or the angle between the sign and the ego vehicle's line of sight.

Computation Time: The computation time consists of several sub-metrics that measure the average time needed to process different stages of an input image (1 to 6), from sign detection to text recognition. These sub-metrics are:

- 1) Sign Detection Time (s).
- 2) Sign Relevance Time (s).
- 3) Text Extraction Time (s).
- 4) Text Recognition Time (s).
- 5) Text-Sign Synthesizer Time (s): The average time required to synthesize the extracted text with the corresponding detected signs, creating a unified output.
- 6) Execution Time (s): The average time required to complete the entire process, from sign detection to text recognition and synthesis, for an input image.

These sub-metrics provide a comprehensive evaluation of the computational efficiency of our system.

C. BASELINES

In this section, we discuss the implementation and training of three state-of-the-art object detectors employed as sign detector baselines for our experiments: YOLO version 5 (YOLOv5), Mask R-CNN [8] (Mask R-CNN), and Mask R-CNN-SWIN [9] (SWIN). YOLOv5 is a popular real-time object detection model known for high accuracy and speed [10]. Mask R-CNN is a framework that extends Faster R-CNN [7] with a branch for predicting segmentation masks. The SWIN transformer is a hierarchical vision transformer that employs shifted windows to capture both local and global information.

In contrast to previous methods, we modified the models to detect generic sign shapes and text without the notion of specific sign categories. We achieve this by combining all classes containing sign-like objects to formulate a single metaclass, "sign", and fine-tuning all sign detection models to create a specialized sign detection model. Given an input image, the sign detector extracts location coordinates in the form of a bounding box or a segmentation mask. YOLOv5 and Mask R-CNN leverage a 50-layer residual network backbone (ResNet50), whereas SWIN consists of a 96-layer shifted window transformer network.

Model Training: Each model was trained for 30 epochs, utilizing the stochastic gradient descent learning algorithm with a starting learning rate of 0.02, a momentum of 0.9, and a weight decay of 0.0001. Throughout the training process, we implemented conventional image augmentation techniques to enhance their robustness [34].

D. ANALYSES AND DISCUSSION

In this section, we present the analysis of our sign perception pipeline with respect to the aforementioned metrics and datasets. We first present the quantitative analysis, then we discuss the qualitative results of our approach and their implications for traffic sign perception in the wild.

1) QUANTITATIVE ANALYSIS

Sign Detection: Our results as shown in Figure 9 indicate that the Mask R-CNN model outperforms YOLOv5 and SWIN at sign detection across all datasets, including TGP (AUC=0.91), VMS (AUC=0.86), Mapillary (AUC=0.78), and CARLA (AUC=0.92). This suggests that Mask R-CNN has greater effectiveness at detecting arbitrary signs. Our proposed system demonstrates considerable promise in detecting less common and complex signs present in the Mapillary and VMS datasets. This is a significant finding, as it suggests the system's ability to generalize to sign detection "in the wild" for autonomous driving applications.

Sign Relevance: Due to the unavailability of GT sign angles in existing datasets, we evaluated the sign relevance and angle estimation performance of our system on the CARLA dataset. We performed our assessment by calculating the error between the estimated angle and the GT angle. We observed that frame-by-frame sign relevance estimation from 2D images could introduce noise, as distant signs are difficult to segment and pixel-level errors in segmentation can result in inconsistent outcomes.

To improve sign direction estimation consistency, we average the pan angle across multiple frames. We calculated this average over 10 to 40 frames, yielding more accurate angle estimations. We found that pixel-level segmentation errors can make results less consistent. We observed that larger signs are less prone to noise, as the increased number of pixels results in a better estimation of vanishing points. Averaging the 10 frames within the closest view of the sign, we attained the lowest estimation error (shown in Table 2).

TABLE 2. Comparison of angle estimation errors for varying frame averages (Closest N Frame, from 10 to 40).

Number of frames	10	15	20	25	30	35	40
Average error	13.3	14.1	14.6	16.3	18.2	21.1	28.5
Median	12.4	13.7	13.9	15.2	16.6	19.8	24.3

TABLE 3. Average Cosine similarity (CosSim), Character Error Rate (CER) and Word Error Rate (WER) evaluated across different datasets.

	TGP	VSM	RoadText-1K
CosSim	0.84	0.59	0.46
CER	0.24	0.32	1.21
WER	0.33	0.46	1.03

To further refine the estimation method, we could employ filtering techniques to improve segmentation and integrate sign tracking to enhance accuracy and consistency.

The novel approach to estimating sign relevance plays a key role in prioritizing detected signs. This enables AVs to focus on the most critical information, facilitating more accurate and responsive navigation.

Sign Text Recognition: In order to evaluate the text recognition performance of our perception system, we conducted assessments using the TGP, VMS, and RoadText-1k datasets. We compared the text strings generated by our system to the annotated text using CER, WER, and Cosine similarity metrics on matched strings associated with signs. To perform a linear assignment between detected and annotated signs, we established the following process. Let A represent a bounding box for a detected sign and B represent a bounding box for a ground-truth sign. We then calculated a score based on the following rules:

- a) If text is present in A XOR B, the score is 0, indicating no similarity between the two signs when only one contains text
- b) If text is present in both A and B, we compute the CER, WER, and Cosine similarity metrics to quantify the degree of similarity
- c) If text is absent in both A and B, we ignore the pair and do not assign a score.

Our method achieved average cosine similarity scores of 0.84 for the TGP dataset, 0.59 for the VMS dataset, and 0.46 for the RoadText-1k dataset. The distribution of CER and WER between extracted and ground truth text from sign-text pairs across different datasets is presented in Figure 10 and Table 3 presents the average Cosine similarity, CER and WER across the different datasets.

Computation Time: To evaluate the computational efficiency of our sign perception pipeline, we measured the inference time, i.e., end-to-end runtime for processing one image; we also measured the runtime for each of the sub-components (i.e., sign detection, relevance estimation, text extraction, text recognition). The experiments were conducted on a machine with 32 Intel Xeon W3245 3.20GHz CPUs and 790GB of memory, with the Perception module utilizing a single Quadro RTX 8000 GPU. We observed

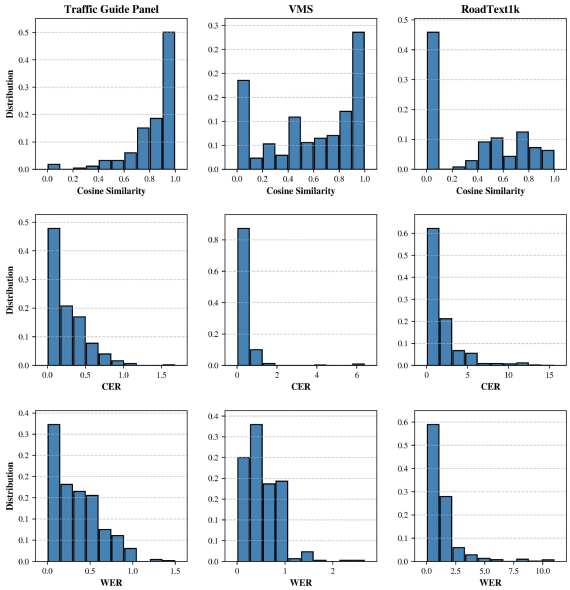


FIGURE 10. Distribution of Cosine Similarity, Character Error Rate (CER) and Word Error Rate (WER) between extracted and ground truth text from sign-text pairs, evaluated across different datasets.

that although Mask R-CNN achieved the highest recall (Figure 9), it also exhibited the highest computation time. As a result, practitioners aiming to deploy our approach on resource-limited computation platforms may need to weigh trade-offs between detection speed and recalls.

As previously discussed, the computation time metric comprises several sub-metrics that measure the average time needed to process different stages (1 to 6) of an input image, ranging from sign detection to text recognition. Tables 4, 5, and 6 provide a detailed breakdown of the results for each stage in the process.

By analyzing and optimizing these individual components, we can enhance the overall performance of the system in real-world scenarios. We conducted the following experiments for each dataset, reporting the individual sub-metrics for each experiment (presented in Tables 4, 5, and 6):

- Experiment 1 (Table 4): Text extraction is performed on individual sign instances, cropped from the image (zoom), with each cropped sign passed through the text extractor (as shown on the right side of Figure 11).
- Experiment 2 (Table 5): Text extraction is performed on individual sign instances (zoom) while filtering the signs based on their score (probability output of sign detector) and relevance value. For example, if the prediction score is less than 0.15 or the relevance score is less than 0.3, the sign is filtered and not passed through the text extractor, reducing computation time.
- Experiment 3 (Table 6): Text extraction is executed just once on the entire image, rather than on individual sign instances (as shown on the left side of Figure 11).

Our findings indicate that the average computation time per image is the lowest when text extraction is executed

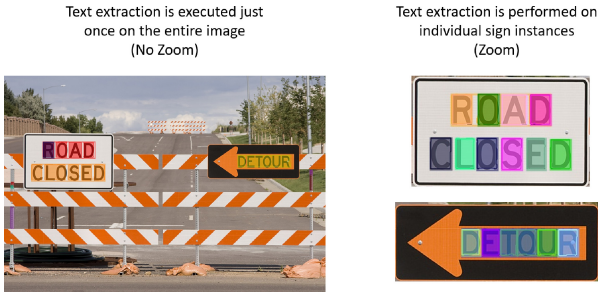


FIGURE 11. Side-by-side comparison of text extraction on the whole image (no zoom) vs. text extraction on individual sign instances, cropped from the image (zoom).

TABLE 4. Average inference time in seconds per image for different stages of the perception system (1 to 6), when the text extraction is performed on individual sign instances (Experiment 1).

	TGP	Mapillary	Road1k	VMS	CARLA
1	0.081	0.271	0.115	0.118	0.144
2	0.018	0.51	0.087	0.087	0.385
3	0.792	1.198	2.149	2.051	4.822
4	0.083	0.061	0.134	0.165	0.468
5	0.059	0.058	0.076	0.187	0.686
6	1.034	2.098	2.561	2.608	6.506

TABLE 5. Average inference time in seconds per image for different stages of the perception system (1 to 6), when the text extraction is performed on individual sign instances and filtering by score and relevance (Experiment 2).

	TGP	Mapillary	Road1k	VMS	CARLA
1	0.08	0.265	0.117	0.118	0.142
2	0.011	0.326	0.06	0.059	0.162
3	0.422	0.513	1.384	1.245	1.472
4	0.081	0.045	0.127	0.16	0.171
5	0.056	0.053	0.066	0.161	0.143
6	0.649	1.202	1.754	1.743	2.09

TABLE 6. Average inference time in seconds per image for different stages of the perception system (1 to 6), when the text extraction is performed on the whole image just once (Experiment 3).

	TGP	Mapillary	Road1k	VMS	CARLA
1	0.079	0.266	0.116	0.118	0.145
2	0.018	0.484	0.085	0.085	0.385
3	0.294	0.551	0.374	0.357	0.464
4	0.086	0.085	0.19	0.171	0.144
5	0.067	0.045	0.098	0.19	0.139
6	0.545	1.432	0.863	0.921	1.277

just once on the entire image, as it does not require additional runs of the text extractor for individual signs. However, this approach reduces our system's performance for low-resolution and small signs. In contrast, zooming into each sign instance increases the image resolution, which improves text extraction performance. Therefore, it is crucial to balance computation time and extraction accuracy for efficient and reliable sign perception in AVs.

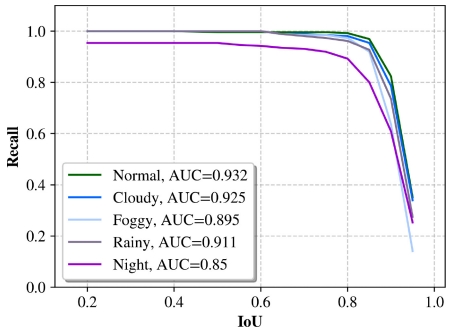


FIGURE 12. Comparison of Recall vs. IoU performance for different weather conditions.

To further reduce computation time we incorporate heuristics that allow our system to only process frames which contain signs with both high detection scores (greater than 0.7) and relevance scores (above 0.6). These score thresholds allow the system to ignore non-essential frames, thereby reducing unnecessary computations. Additionally, as text extraction and recognition tasks can be computationally expensive, we introduced a filter based on the resolution of the detected signs. We found that a higher resolution typically correlates with better readability. Therefore, we only process signs with a resolution exceeding 60 pixels in both width and height. Thus, each frame is evaluated based on these scores and resolution requirements. Frames that do not contain any signs meeting these criteria are skipped entirely. If a frame does contain signs that meet the criteria, we apply selective zoom to those signs and process them for text extraction and recognition. By doing this, we have managed to reduce computation time significantly, allowing our system to process around two images per second. We would like to highlight that these score thresholds were selected meticulously and optimized in a manner to ensure that the overall performance of the system did not degrade significantly. We experimented with a range of values to find the values that maximized speed without affecting the reliability of our system.

Influence of adverse weather: Figure 12 shows the comparison of Recall vs. IoU performance for different weather conditions. As expected, our system's best performance was observed under normal weather conditions. However, even under adverse weather conditions, the system demonstrated competent performance levels, suggesting its robustness to such scenarios.

In addition, Table 7 presents the evaluation results of sign angle estimation under various weather conditions. As anticipated, the best performance is observed under favorable weather conditions (normal weather). The impact of weather conditions becomes increasingly noticeable as we consider signs at greater distances (higher number of frames).

2) QUALITATIVE ANALYSIS

We show example outputs of our sign perception pipeline in Figure 13, which represents a diverse number of traffic

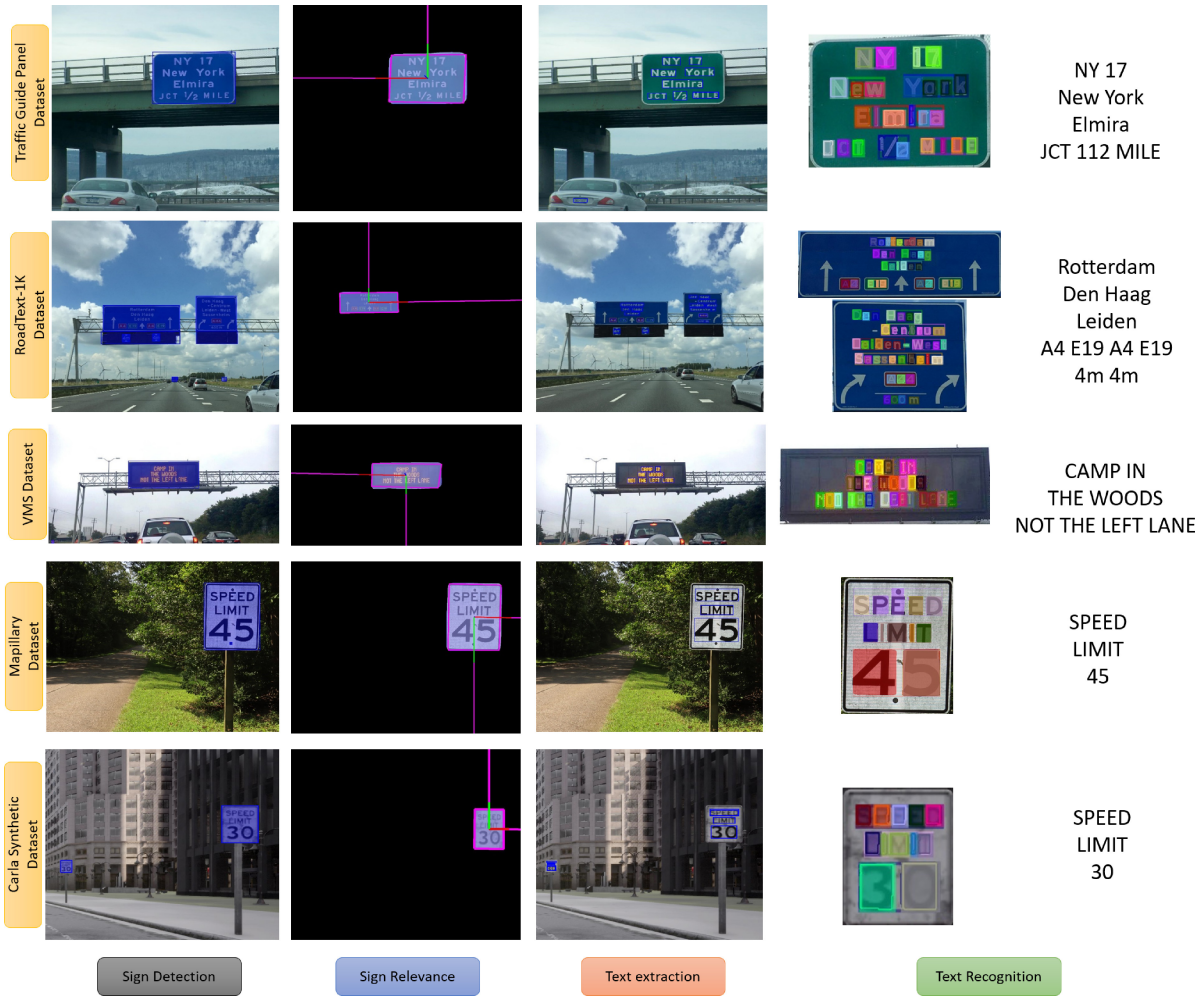


FIGURE 13. Illustration of diverse traffic sign images from the tested datasets, providing an overview of the components of our system. The figure showcases the outputs at different stages of the pipeline.

TABLE 7. Comparison of angle estimation errors for varying frame averages across different weather conditions.

Weather	Number of Frames (Closest N Frame)						
	10	15	20	25	30	35	40
Normal	13.1	14.3	14.7	16.5	18	21.4	28.3
Cloudy	13.6	14.6	15.2	17.1	18.7	22.3	29.1
Rainy	15.4	15.7	17.2	18.9	21.2	26.3	32.8
Foggy	17.8	18.4	20.2	21.8	24.6	27.7	36.4
Night	21.3	21.7	24.3	26.1	29.5	36	42.6

sign images. The figure highlights the robustness and accuracy of different modules in detecting signs and recognizing their text. Moreover, our system shows significant potential in detecting less common and complex signs found in the RoadText-1k, Mapillary, and VMS datasets. This underlines the system's capacity to generalize "in-the-wild" sign detection, laying the foundation for advancements in traffic sign understanding.

V. CONCLUSION

In this paper, we presented a robust sign perception pipeline for autonomous vehicles that incorporates sign detection, sign relevance, text extraction, and recognition as essential components of the perception process. The comprehensive analysis of our system demonstrates its effectiveness in sign detection and text recognition across various datasets, contributing to more reliable autonomous vehicle navigation. Our system shows promise in detecting uncommon and complex signs, which is crucial for addressing the diversity and complexity of traffic signs, as well as the sign relevance, paving the way for improvements in traffic sign interpretation. Our novel approach to estimating sign relevance allows the autonomous vehicle to prioritize detected signs based on their orientation, thereby enabling more responsive navigation.

ACKNOWLEDGMENT

The authors would like to thank Hyukseong Kwon, Priyantha Mudalige, and Paul Krajewski for their constructive feedback and discussions.

REFERENCES

[1] J. Lampkins, D. Chan, A. Perry, S. Strelnikoff, J. Xu, and A. E. Ashari, "Multimodal road sign interpretation for autonomous vehicles," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, 2022, pp. 5979–5987.

[2] J. Li and Z. Wang, "Real-time traffic sign recognition based on efficient CNNs in the wild," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 975–984, Mar. 2019.

[3] D. Tabernik and D. Skočaj, "Deep learning for large-scale traffic-sign detection and recognition," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1427–1440, Apr. 2020.

[4] Y. Taki and E. Zemmouri, "An overview of real-time traffic sign detection and classification," in *Proc. 5th Int. Conf. Smart City Appl. Innov. Smart Cities Appl.*, 2021, pp. 344–356.

[5] M. Pressman, "Understanding Tesla's different autopilot software packages." Accessed: Jun. 2023. [Online]. Available: https://evannex.com/blogs/news/understanding-teslas-different-autopilot-software-packages

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

[7] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.

[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2961–2969.

[9] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 10012–10022.

[10] 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," 2022, *arXiv:2207.02696*.

[11] J. Cao, J. Zhang, and X. Jin, "A traffic-sign detection algorithm based on improved sparse R-CNN," *IEEE Access*, vol. 9, pp. 122774–122788, 2021.

[12] J. Zhang, Z. Xie, J. Sun, X. Zou, and J. Wang, "A cascaded R-CNN with multiscale attention and imbalanced samples for traffic sign detection," *IEEE Access*, vol. 8, pp. 29742–29754, 2020.

[13] L. Hacker and J. Seewig, "Insufficiency-driven DNN error detection in the context of SOTIF on traffic sign recognition use case," *IEEE Open J. Intell. Transp. Syst.*, vol. 4, pp. 58–70, 2023.

[14] M. Atif, A. Ceccarelli, T. Zoppi, M. Gharib, and A. Bondavalli, "Robust traffic sign recognition against camera failures," *IEEE Open J. Intell. Transp. Syst.*, vol. 3, pp. 709–722, 2022.

[15] F. Geissler, A. Unnervik, and M. Paulitsch, "A plausibility-based fault detection method for high-level fusion perception systems," *IEEE Open J. Intell. Transp. Syst.*, vol. 1, pp. 176–186, 2020.

[16] R. Greer, J. Isa, N. Deo, A. Rangesh, and M. M. Trivedi, "On salience-sensitive sign classification in autonomous vehicle path planning: Experimental explorations with a novel dataset," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2022, pp. 636–644.

[17] M. Donoser and H. Bischof, "Efficient maximally stable extremal region (MSER) tracking," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, 2006, pp. 553–560.

[18] R. Valiente, M. T. Sadaike, J. C. Gutiérrez, D. F. Soriano, G. Bressan, and W. V. Ruggiero, "A process for text recognition of generic identification documents over cloud computing," in *Proc. Int. Conf. Image Process., Comput. Vis., Pattern Recognit. (IPCV)*, 2016, p. 142.

[19] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao, "Detecting text in natural image with connectionist text proposal network," in *Proc. 14th Eur. Conf. Comput. Vis.*, 2016, pp. 56–72.

[20] D. Deng, H. Liu, X. Li, and D. Cai, "Pixellink: Detecting scene text via instance segmentation," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, 2018, pp. 6773–6780.

[21] J. Ye, Z. Chen, J. Liu, and B. Du, "TextFuseNet: Scene text detection with richer fused features," in *Proc. IJCAI*, vol. 20, 2020, pp. 516–522.

[22] X. Chen, L. Jin, Y. Zhu, C. Luo, and T. Wang, "Text recognition in the wild: A survey," *ACM Comput. Surveys*, vol. 54, no. 2, pp. 1–35, 2021.

[23] R. Valiente, J. C. Gutiérrez, M. T. Sadaike, and G. Bressan, "Automatic text recognition in Web images," in *Proc. 23rd Brazillian Symp. Multimedia Web*, 2017, pp. 241–244.

[24] R. Mithe, S. Indalkar, and N. Divekar, "Optical character recognition," *Int. J. Recent Technol. Eng.*, vol. 2, no. 1, pp. 72–75, 2013.

[25] J. Baek et al., "What is wrong with scene text recognition model comparisons? Dataset and model analysis," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 4715–4723.

[26] C. Luo, L. Jin, and Z. Sun, "MORAN: A multi-object rectified attention network for scene text recognition," *Pattern Recognit.*, vol. 90, pp. 109–118, Jun. 2019.

[27] E. H. Chen et al., "Investigating binary neural networks for traffic sign detection and recognition," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2021, pp. 1400–1405.

[28] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[29] X. Rong, C. Yi, and Y. Tian, "Recognizing text-based traffic guide panels with cascaded localization network," in *Proc. Comput. Vis. Workshops*, 2016, pp. 109–121.

[30] S. Reddy, M. Mathew, L. Gomez, M. Rusinol, D. Karatzas, and C. Jawahar, "RoadText-1K: Text detection & recognition dataset for driving videos," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2020, pp. 11074–11080.

[31] E. Puertas, G. De-Las-Heras, J. Sánchez-Soriano, and J. Fernández-Andrés, "Dataset: Variable message signal annotated images for object detection," *Data*, vol. 7, no. 4, p. 41, 2022.

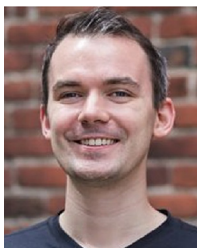
[32] G. Neuhold, T. Ollmann, S. R. Bulo, and P. Kontschieder, "The Mapillary vistas dataset for semantic understanding of street scenes," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 4990–4999.

[33] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. 13th Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[34] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, pp. 1–48, 2019.

RODOLFO VALIENTE received the Ph.D. degree in computer engineering from the University of Central Florida and the M.Sc. degree from the University of Sao Paulo. He is a Research Scientist with HRL Laboratories. His research interests include computer vision, autonomous vehicles, robotics, reinforcement learning, and artificial intelligence.

DARREN CHAN received the Ph.D. degree in computer science and engineering from the University of California San Diego. In 2021, he joined HRL Laboratories as a Research Scientist, where he currently serves as a Principal Investigator on several projects in the area of advanced manufacturing systems. He is also an Avid Hardware Designer with professional consulting experience in embedded systems and electronics. His activities also include research in autonomous vehicles with specializations in the areas of computer vision, robotics, and machine learning.



ALAN PERRY received the master's degree in data science from the University of San Francisco. He is a Research Scientist with HRL Laboratories. With research experience ranging from Semantic Segmentation to NLP Classification, he has addressed varied problems in the domains of autonomous vehicles, national security, and healthcare.

JOSHUA LAMPKINS received the master's degree in mathematics from the University of California, Los Angeles. He is a Research Scientist with HRL Laboratories. His research areas include cryptography, robotics, and autonomous vehicles.

SASHA STRELNIKOFF received the Bachelor of Science degree in mathematics from the University of California, Los Angeles, and the Master of Science degree in machine learning from the University of California, San Diego. He is a Research Scientist with HRL Laboratories whose primary research interests lie in causality-based methods, knowledge-infused learning, and natural language understanding.

JIEJUN XU received the Ph.D. degree from the Computer Science Department, University of California, Santa Barbara. He was affiliated with the Vision Research Lab, the Center for Bio-Image Informatics, and the Information Network Academic Research Center. He is a Senior Scientist with HRL Laboratories, where he is currently the Head of the Knowledge Navigation Center. His research interests include graph machine learning, multimodal deep learning, and natural language understanding.

ALIREZA ESNA ASHARI received the B.S. and M.S. degrees from the University of Tehran, Tehran, Iran, and the Ph.D. degree from the University of Paris-Est, working at Inria (the French national institute for research in computer science and control), France. He worked as a Research Faculty Member with the Georgia Institute of Technology, Atlanta, Georgia, USA. He is currently a Senior Researcher with the General Motors, Global Research and Development Center, Warren, MI, USA. His research interests include autonomous driving, reinforcement learning, computer vision, natural language processing, and multimodal deep learning.