

Research Article

Automatic Outer Contour Detection and Quantification of Vehicles Using Monocular Vision

Dayong Han,¹ Chaodong Zhang,¹ Liang Wang,¹ Xianglong Xu,¹ and Yingkai Liu ²

¹PowerChina Road Bridge Group Co., Ltd., Changsha 410082, China

²College of Civil Engineering, Hunan University, Changsha 410082, China

Correspondence should be addressed to Yingkai Liu; lyk199343@hnu.edu.cn

Received 8 July 2023; Revised 16 November 2023; Accepted 3 January 2024; Published 23 January 2024

Academic Editor: Łukasz Jankowski

Copyright © 2024 Dayong Han et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Oversized vehicles have the potential to collide with walls or ceilings when passing through tunnels and bridges, posing a serious threat to the health of transportation infrastructure and public safety. Hence, it is crucial to accurately and immediately detect vehicle dimensions, including length, width, and height, to avoid such accidents. Using computer vision and view geometry, this study presents a framework for automatically detecting and quantifying the outer contours of vehicles through monocular vision. First, traffic scene images are captured which are then used to create a transformation matrix of the ground surface. Second, a modified Mask region-based convolutional neural network (Mask R-CNN) is constructed to detect and segment the vehicle instances from the video frames. Finally, a view geometry-based algorithm was developed to detect the outer contours of passing vehicles and quantify their dimensions. In the field test, the accuracy of the vehicle segmentation and the identified vehicle dimensions was validated. In addition, the proposed method's superiority was confirmed by comparing it with two other existing approaches. The comparison results show that the proposed method has better accuracy and is more convenient to use since it does not require a premeasured reference. In addition, the developed method can accurately identify not only the dimensions of vehicles parallel to the road but also vehicles that are changing lanes or making a U-turn.

1. Introduction

Over the past decades, the increasing number of vehicles on the road has caused a rising global concern regarding traffic accidents caused by unregulated transportation [1]. In an attempt to increase their profits, some transporters unlawfully modify their vehicles to significantly expand their carrying capacity. Nonetheless, the modified vehicles usually pose hidden dangers, such as excessive length, width, and height, which are the primary causes of most traffic crashes [2]. With the development of large-scale engineering projects in power construction, metallurgy, petrochemicals, and other fields, there is an increasing demand for the transportation of ultralarge pieces of equipment [3]. When maneuvering through curves, tunnels, and bridges, these oversized vehicles have a significantly higher probability of colliding with walls or ceilings [4]. The rising number of oversized vehicles on the road is significantly endangering

the structural integrity of bridges, tunnels, and other infrastructure. In Beijing, for instance, 50% of bridges have suffered collisions with oversized vehicles, causing over 20% of total damage to bridges [5]. Furthermore, these collisions can result in catastrophic casualties, thereby underscoring the seriousness of the issue. For example, four passengers died as a result of a collision between a double-decker bus and a railroad bridge in Onondaga, New York [6]. In 2014, there were over 34 serious collisions between vehicles and bridges in Texas, and it is estimated that repairing a damaged bridge will cost between \$200,000 and \$300,000 [6]. In 2012, 28 passengers died in a tour bus collision with a tunnel wall on Swiss Route 9 [7]. Direct collisions between large vehicles and tunnel walls constitute 42.3% of all tunnel traffic accidents, according to statistical evidence [8]. Hence, averting collisions between vehicles and traffic infrastructure on road sections with restricted passing dimensions is of utmost importance for ensuring traffic safety.

In real traffic conditions, collisions between oversized vehicles and transportation infrastructure are typically prevented in the following three ways:

- (1) *Passive Warning*. The most common and cost-efficient method of reducing the risk of traffic accidents is to provide drivers with warning signs [9]. Traditional traffic signs are typically placed along roads to alert drivers of upcoming turns, vertical clearances, and horizontal clearances for bridges or tunnels. However, drivers often tend to underestimate the dimensions of their vehicles, which leads to the ineffectiveness of warning signs in preventing vehicle collisions (only 10–20% effectiveness) [10].
- (2) *Sacrificial Protection*. Height limit beams and width limit piles are common sacrificial protection facilities used to limit the size of passing vehicles [11, 12]. Indeed, the height limit beams and width limit piles can effectively halt oversized vehicles and protect important traffic structures such as bridges and tunnels. However, there is a potential risk of injury or death to the driver and passengers of the stopped vehicle and subsequent vehicles.
- (3) *Proactive Monitoring*. In addition to manual measurements, common commercial proactive monitoring methods include infrared measurements and laser measurements. Manual measurement is accurate but can interrupt traffic. Infrared and laser measurements can accurately capture the outer contour of the vehicle without interrupting traffic [13]. Once a vehicle is detected to be oversized, it will be warned and directed off the road.

Of the three methods mentioned above, proactive monitoring is deemed the most efficient measure to prevent collisions by detecting the dimensions of vehicles in a noncontact manner and alerting the driver of an oversized vehicle in advance [14]. Nevertheless, laser and infrared-based devices come with a hefty cost for installation and maintenance, which hinders their widespread use. In a study conducted by Cawley [10], it was observed that installation costs for a single laser or infrared-based vehicle dimension inspection system can be as high as \$135,000 and possibly more when postinstallation maintenance is taken into account. Thus, there is a pressing need to propose a real-time vehicle dimension measuring method that is more economical and precise.

In the past decade, image measurement techniques have become popular as a new method for obtaining vehicle shape information [15–18]. Using this technology, the entire system for detecting oversized vehicles can be implemented with less hardware and at a reduced cost, with simpler installation and better real-time performance. Rezaei et al. [15] and Lu et al. [16] accurately identified vehicles from videos and obtained their true dimensions. However, in these studies, the camera had to be calibrated using satellite images and feature point matching, which significantly increased the complexity of the method. Furthermore, the identification accuracy of the vehicle's dimensions can be greatly

impacted when the satellite images are blurred. To eliminate the reliance on satellite imagery, Zhang et al. [17] developed a multitarget tracking and image calibration method that uses two cameras on bridge pylons to measure vehicle length and position. Even without the need for satellite imaging, the camera should still undergo precalibration using a standard vehicle. Likewise, Lu et al. [18] established a vision-based algorithm for defining the 3D bounding box of a vehicle and determining whether the vehicle is overheight, and Liu et al. [19] proposed an algorithm to detect the wheelbase of a vehicle using a monocular camera. Similarly, the camera must be calibrated by finding a reference with a known height.

In the current study, based on computer vision and view geometry, a framework is developed for detecting and quantifying vehicle outer contours automatically using monocular vision. The first step involves proposing an automatic camera calibration algorithm, which does not depend on satellite images or references, to achieve a partial transformation relationship between 2D images and 3D environments. Then, a modified target detection network based on Mask R-CNN [20] is developed to accurately segment road vehicles at the pixel level. Finally, a novel 3D vehicle bounding box identification algorithm is proposed to determine whether the length, width, and height of a vehicle exceed the threshold values. The proposed method can identify and segment vehicles automatically, without the need for manual point selection and without the assumption that the vehicle must be parallel to the road direction (as assumed in the studies of Zhang et al. [17] and Lu et al. [18]). Thus, the proposed method is applicable to vehicles changing lanes or making a U-turn. The study presents a comparative analysis of the proposed method against other established techniques and investigates the impact of vehicle driving angle, camera position, and camera resolution separately. The positive results of the study underscore the accuracy and efficiency of the proposed method, showcasing its potential in detecting oversized vehicles and preventing infrastructure collisions.

2. Architecture of the Proposed Method

In this study, a monocular vision-based method for automatic detection and quantification of vehicle exterior contour is proposed. Figure 1 illustrates the specific flowchart of this method, which consists of four primary steps:

- (1) *Calibration of the Camera*. First, a novel algorithm is proposed to automatically calibrate the camera by capturing an image of the road surface and determining the vanishing point in the camera's field of view.
- (2) *Segmentation of the Vehicle in Pixel Level*. Second, the vehicle instances are segmented from each frame of the video using the modified Mask R-CNN.
- (3) *Construction of Vehicle's 3D Bounding Box*. Subsequently, the vehicle's 3D bounding box is precisely constructed at varying angles in each video frame.

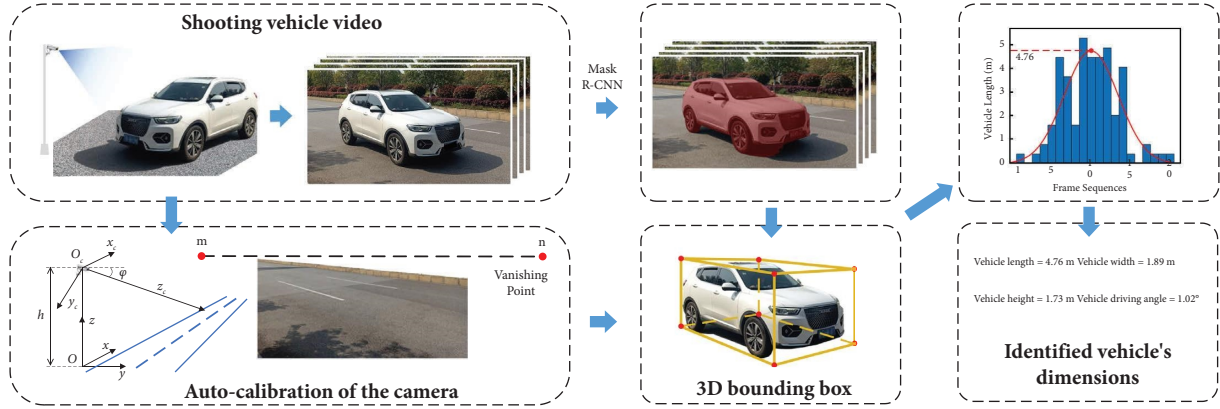


FIGURE 1: Flowchart of the proposed method.

- (4) *Identification of Vehicle's Outer Contour Dimensions.* Finally, the dimensions of the vehicle, i.e., length, width, height, and driving angle, can be determined using the proposed method.

In the above steps, calibration with standard references and manual selection of key points in the video is unnecessary. Therefore, the proposed method offers enhanced convenience in identifying vehicle dimensions and determining if it exceeds the threshold. In Section 2.1–2.3, each step will be elaborated in detail.

2.1. Automatic Calibration of the Road Surface in the Image.

In order to ascertain the precise dimensions of an object from 2D image, it is first necessary to establish an accurate transformation relationship between the 2D image and the 3D world [21], i.e., camera calibration. Traditional methods typically involve premeasuring the precise dimensions of a reference object [18, 22]. However, these approaches have limited applicability in road sections where finding a reference object or measuring its dimensions is challenging. Moreover, the methods require repeated manual measurements in case of any changes in camera position or referential object dimensions, thereby proving to be time-consuming and labor-intensive. Therefore, this study presents a reference-free algorithm to determine the measurements of the vehicle based on the camera's height and angle. As there are no reference objects, it is only possible to establish a transformation relationship between the image's road surface and the actual road surface, which constitutes an incomplete calibration. Based on this incomplete calibration, the dimensions of the vehicle can still be determined, and the detailed methodology is presented in the following subsections.

The flowchart of the incomplete calibration is illustrated in Figure 2. In the figure, $O-xyz$ is the world coordinate system, O is the origin, and x , y , and z are the three axes. $O_c-x_c y_c z_c$ is the camera coordinate system, O_c is the origin, and x_c , y_c , and z_c are the three axes. α refers to the conversion factor between the image coordinate system and the spatial coordinate system, with $\alpha \neq 0$. Camera parameters f , h , φ , and θ represent the focal length, height, pitch, and deflection

angle of the camera, respectively. The projection formula from world coordinates onto camera coordinates can be obtained by a simple derivation [23]:

$$\begin{cases} u = \frac{\alpha u}{\alpha} = \frac{f x}{y \cos(\varphi) - z \sin(\varphi) + h \sin(\varphi)}, \\ v = \frac{\alpha v}{\alpha} = \frac{f h \cos(\varphi) - f y \sin(\varphi) - f z \cos(\varphi)}{y \cos(\varphi) - z \sin(\varphi) + h \sin(\varphi)}. \end{cases} \quad (1)$$

In order to identify the vehicle's dimensions in the world coordinate system, it is recommended to first rotate the y -axis of the world coordinate system in the same direction as the road direction. Next, the roadside camera coordinate system is converted to the world coordinate system by rotation and translation. The rotation matrix \mathbf{R} contains two components: the rotation angle $\varphi + \pi/2$ around the x -axis and the rotation angle θ around the z -axis, which can be expressed as follows:

$$\mathbf{R} = \mathbf{R}_x\left(\varphi + \frac{\pi}{2}\right) \mathbf{R}_z(\theta)$$

$$= \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ -\sin(\varphi)\sin(\theta) & -\sin(\varphi)\cos(\theta) & -\cos(\varphi) \\ \cos(\varphi)\sin(\theta) & \cos(\varphi)\cos(\theta) & -\sin(\varphi) \end{bmatrix}, \quad (2)$$

and the translation matrix \mathbf{T} is as follows:

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -h \end{bmatrix}. \quad (3)$$

Therefore, the homogeneous coordinate form of (1) can be expressed as follows:

$$\begin{aligned} (u, v, 1)^T &= \mathbf{KRT}(x, y, z, 1)^T \\ &= \mathbf{H}(x, y, z, 1)^T, \end{aligned} \quad (4)$$

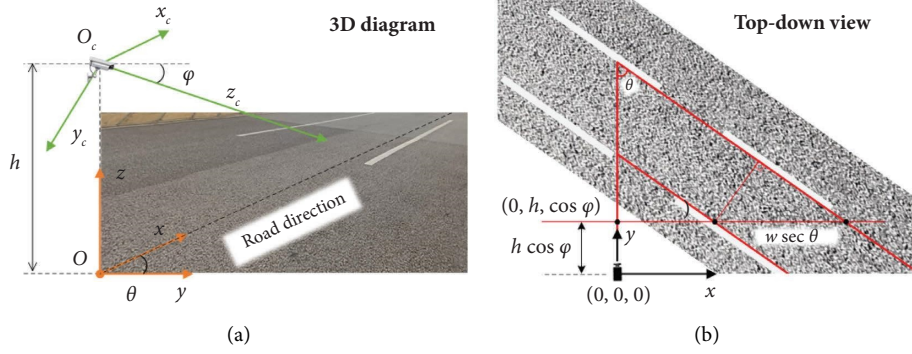


FIGURE 2: Diagram of the real world and camera coordinate systems.

where \mathbf{K} is the intrinsic matrix of the camera and \mathbf{H} is the transformation matrix, $\mathbf{H} = [h_{ij}]$, where $i = 1, 2, 3$ and $j = 1, 2, 3, 4$. After expanding the formula, equation (4) can be rewritten as follows:

$$\begin{cases} u = \frac{h_{11}x + h_{12}y + h_{13}z + h_{14}}{h_{31}x + h_{32}y + h_{33}z + h_{34}}, \\ v = \frac{h_{21}x + h_{22}y + h_{23}z + h_{24}}{h_{31}x + h_{32}y + h_{33}z + h_{34}}. \end{cases} \quad (5)$$

It can be seen from (5) that when the z value of a spatial point is known, the 3D coordinates of that point can be uniquely determined. This incomplete calibration is one of the foundations for determining the dimensions of the vehicle. In addition, it is also necessary to determine the locations of the vanishing points, i.e., the intersection of parallel lines in the 2D image. In the present study, an algorithm to automatically identify three major, mutually orthogonal vanishing points in an image is proposed. The three vanishing points are as follows: the first vanishing point v_1 located in the direction of the road, the second vanishing point v_2 located in the direction perpendicular to v_1 and parallel to the road surface, and the third vanishing point v_3 located in the direction perpendicular to the road surface. In the general camera view, the position of the third vanishing point v_3 tends to be at infinity [24]. Therefore, only v_1 and v_2 need to be derived in the present study.

The method for determining the vanishing point is shown in Figure 3, and the specific processes are as follows:

(i) Step 1:

Take photos of the road and segment the road surface as a region of interest (ROI) using a well-trained model [25], as shown in Figures 3(b) and 3(c). Segmentation of the raw photos can exclude the effect of background and reduce the computational cost [26]. Note that one is actually free to choose other well-developed semantic segmentation models and embed them in the method to achieve the same purpose.

(ii) Step 2:

The Canny operator [27] then finds the obvious edgelets in the ROI, as shown in Figure 3(d). In general traffic scenarios, the edgelets pointing to

vanishing points v_1 and v_2 are mainly distributed on the road surface, such as lane edges and traffic index lines [28]. Each edgelet has three characteristic parameters, i.e., position x_i , direction d_i (perpendicular to the edge gradient), and intensity s_i (gradient magnitude). The line passing through each edgelet and parallel to its direction is represented as a vector l_i .

(iii) Step 3:

All edge pixel points x_i are collected into the edgelet set \mathbf{E} . Take any two edgelets x_j and x_k from \mathbf{E} and calculate the intersection of l_j and l_k as a candidate vanishing point $v_{jk} = l_j \times l_k$, as shown in Figure 3(e). Then, all other edgelets x_i in \mathbf{E} are traversed, and it is determined separately whether the angle β between l_i and the line $x_i v_{jk}$ is less than the specified threshold β_{th} . When $\beta < \beta_{th}$, x_i votes one score to the candidate vanishing point v_{jk} , which can be expressed as follows:

$$\text{score}(v_{jk}, x_i) = \begin{cases} s_i \left(\frac{1 - e^{-\cos\beta_i}}{1 - e^{-1}} \right), & \beta_i < \beta_{th} (i \in \mathbf{E}), \\ 0, & \text{otherwise,} \end{cases} \quad (6)$$

where s_i is the intensity value of each edgelet. The reason for using s_i as the weighting factor is that edgelets with larger gradients are more likely to be real edges rather than noise. From (6), it can be seen that the score of the candidate vanishing point v_{jk} depends on its consistency with the direction of the remaining edgelets. The directional consistency reaches a maximum value of s_i when l_i passes through the candidate vanishing point v_{jk} , while it drops to 0 when the deviation angle β exceeds β_{th} . When traversing all edgelets, the candidate vanishing point with the highest score is selected as the first vanishing point v_1 , and the edgelets supporting v_1 will be removed from \mathbf{E} . Then, the above procedure is repeated to find the second vanishing point v_2 . The two vanishing points identified in the case are plotted in Figure 3(f), and the red crosses in the figure indicate the unselected candidate vanishing points.

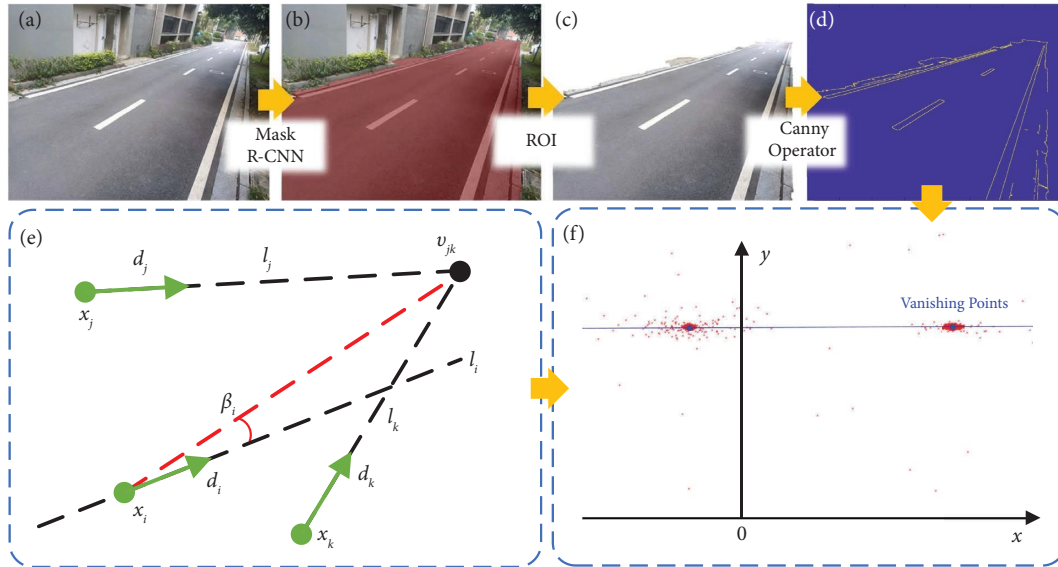


FIGURE 3: Automatic determination of vanishing points: (a) photographing the road surface; (b) segmenting the road surface with well-trained Mask R-CNN; (c) region of interest; (d) identifying edge pixels in the range of ROI with the Canny operator; (e) illustration of candidate vanishing point estimation; (f) identifying vanishing points v_1 and v_2 .

It can be seen that a higher value of β_{th} will result in a higher number of false vanishing points being misidentified as real vanishing points in each iteration, thus requiring repeated computations and reducing computational efficiency. Conversely, a lower β_{th} value will cause candidate vanishing points located in close proximity to the same vanishing point to be falsely classified as two primary vanishing points. Therefore, this study determined a suitable β_{th} of 10° after conducting trial calculations. This ensures both sufficient computational efficiency and accurate identification of the vanishing point.

2.2. Pixel-Level Vehicle Segmentation with Modified Mask R-CNN. The Mask R-CNN model is an instance segmentation framework that was proposed by He et al. [24] in 2017. It is simple, flexible, and versatile, providing precise mask and bounding box coordinates for multiple objects simultaneously. Given that the present study involves the detection and segmentation of numerous vehicles in monocular images, this feature is particularly important. Furthermore, it is worth noting that the implementation of Mask R-CNN has yielded outstanding outcomes in multiple benchmarks like COCO and Pascal VOC [24]. Therefore, it guarantees both robustness and accuracy when employed in this research.

Based on the Faster R-CNN model [29], the Mask R-CNN can perform object detection and segmentation by adding binary mask branches to each ROI, as shown in Figure 4. In the previous subsection, a well-trained Mask R-CNN model is used to effectively detect the ground surface. Given that the accuracy of the ROI boundaries will not affect the detection of small edges in the ROIs, it is sufficient to use the original Mask R-CNN alone. However, when segmenting vehicle instances from traffic scene images, the accuracy of the original Mask R-CNN will become

unsatisfactory. This is because there are three problems in this process: first, the complex background in real traffic scenes significantly disrupts the segmentation of foreground vehicles; second, the computational efficiency is crucial for the real-time detection of passing vehicles; and third, there is a large amount of unfavorable training data in the vehicle segmentation dataset, which is extremely difficult to discriminate and extremely easy to discriminate, which consumes a large amount of training resources. In this study, the following adjustments to enhance the training performance were implemented: (1) adding a Squeeze and Excitation module to improve the model's attention to foreground vehicles and reduce the interference of complex background; (2) adding a Tree module to improve the backbone of the network to reduce the complexity of information transfer and improve the overall efficiency; and (3) adding a gradient harmonizing mechanism to improve the gradient density and focal loss-based loss function. By suppressing both easy and challenging samples, the samples are harmonized across the gradient to improve training performance.

2.2.1. Add Squeeze and Excitation Attention Gate. The accurate segmentation of the vehicle directly determines the accuracy of vehicle outer contour dimension identification. However, the original Mask R-CNN network can hardly fully extract vehicle features, so it is often a challenge to distinguish vehicles from other moving objects. When vehicle shadows and complex backgrounds are considered, the final identified dimensions may have significant errors. Therefore, the feature extraction capability of Mask R-CNN needs to be further enhanced. In this study, a Squeeze and Excitation (SE) attention mechanism [30] is used in the feature extraction module, as shown in Figure 5. The SE attention mechanism consists of three main procedures:

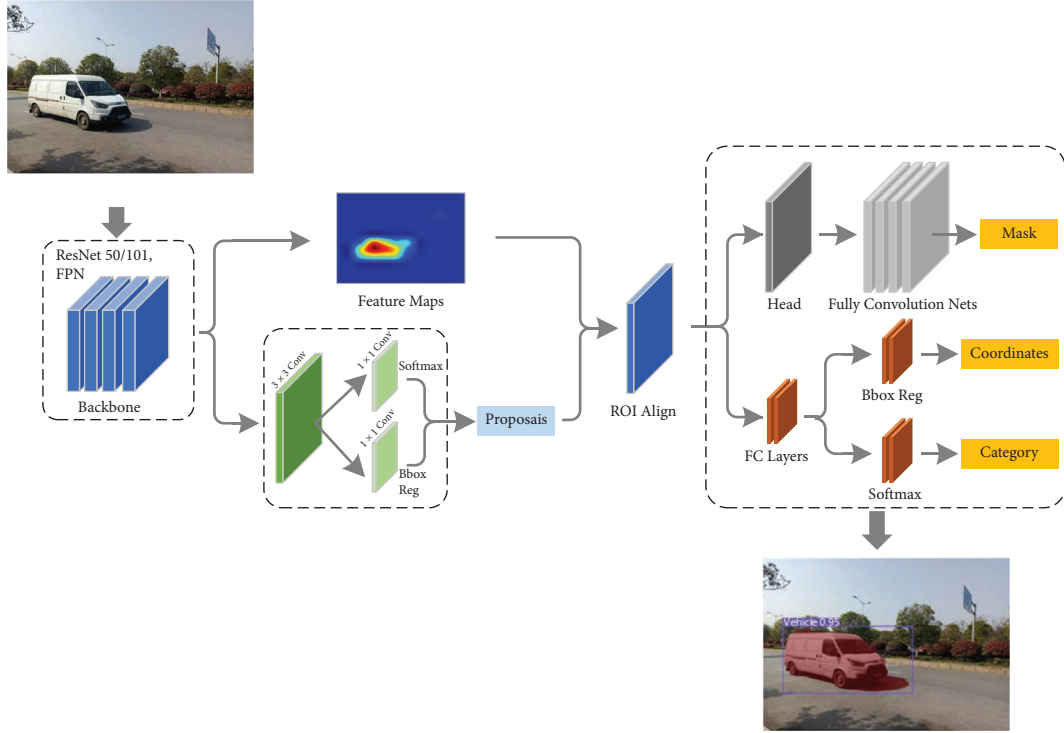


FIGURE 4: Mask R-CNN network structure.

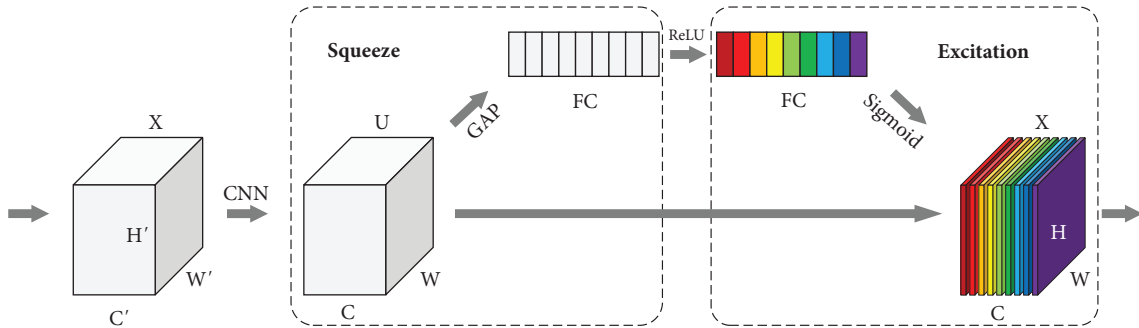


FIGURE 5: Architecture of the SE module.

(i) Step 1:

Squeezing the original $H \times W \times C$ feature map into $1 \times 1 \times C$ dimensions by global average pooling:

$$\begin{aligned} z_c &= \mathbf{F}_{sq}(\mathbf{u}_c) \\ &= \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W u_c(i, j). \end{aligned} \quad (7)$$

\mathbf{u}_c in the equation can be expressed as follows:

$$\mathbf{u}_c = \mathbf{v}_c \otimes \mathbf{X} = \sum_{s=1}^{C'} \mathbf{v}_c^s \otimes \mathbf{x}^s, \quad (8)$$

where \mathbf{u}_c represents the convolution result for the convolution kernel of group c with channel

number s and the feature map with channel number s , \mathbf{v} denotes the learned set of filter kernels, \mathbf{v}_c refers to the parameters of the c th filter, and \otimes denotes the convolution operation. The purpose of the first step is to improve the correlation between the individual channels rather than merely summarizing them.

(ii) Step 2:

The $1 \times 1 \times C$ feature map obtained by squeezing is passed through the fully connected layer, and the importance of each channel is predicted. Then, the predicted values are normalized to a range of 0-1 using the Sigmoid activation function to obtain the weights of different channels. Finally, the weights are applied to the original feature map:

$$\begin{aligned}
\mathbf{s} &= \mathbf{F}_{ex}(\mathbf{z}, \mathbf{W}) \\
&= \sigma(g(\mathbf{z}, \mathbf{W})) \\
&= \sigma(\mathbf{W}_2 \delta(\mathbf{W}_1 \mathbf{z})),
\end{aligned} \tag{9}$$

where δ is the Rectified Linear Unit (ReLU), \mathbf{W}_1 is the first fully connected operation for dimensionality reduction, and \mathbf{W}_2 is the second fully connected operation that can restore dimension to the input dimension. After two fully connected layers, the weights of each feature map are normalized with the sigmoid activation function (σ).

(iii) Step 3:

Multiply the initial feature map element-wise with the weight matrix:

$$\begin{aligned}
\bar{\mathbf{x}}_c &= \mathbf{F}_{scale}(\mathbf{u}_c, s_c) \\
&= s_c \mathbf{u}_c.
\end{aligned} \tag{10}$$

The convolution operation in Step 1 amplifies both the useful and useless feature information. However, the key aspect to consider is that the excitation is not solely dependent on the convolutional output. Instead, it combines the squeezed feature map with the output of the convolutional layers using a weighted sum. The weights are learned during the training process and adaptively adjust the importance of each spatial location in the input feature map. The sigmoid activation function is applied to the output of the excitation, which serves as a gate mechanism to control the amount of information flowing through the channel. The sigmoid function calculates the probability of each spatial location being selected, and the weighted sum of the input feature maps is computed. This process results in a new feature representation that emphasizes the most relevant spatial locations and suppresses the less important ones. Thus, the SE attention module can effectively amplify the useful feature information and reduce the useless feature information.

2.2.2. Modification of the Backbone. The backbone of the original Mask R-CNN is ResNet [31], which can convert information from the previous layer to the next layer through skip connections. Generally, adding too many layers to a residual block may impede the flow of information in the network and reduce computational efficiency [32]. However, computational efficiency is crucial to the real-time detection of passing vehicles. Therefore, in this study, a more lightweight Tree module is employed [33], as shown in Figure 6. As can be seen from the figure, the 1×1 layer in the Tree module can increase the dimensionality, while the 3×3 layer is further processed by subsequent convolutional layers. Finally, the outputs of all 1×1 layers and 3×3 layers are connected and converted through a transition layer. Compared to ResNet, the Tree module has a deeper network structure, while the model is less complex and more efficient.

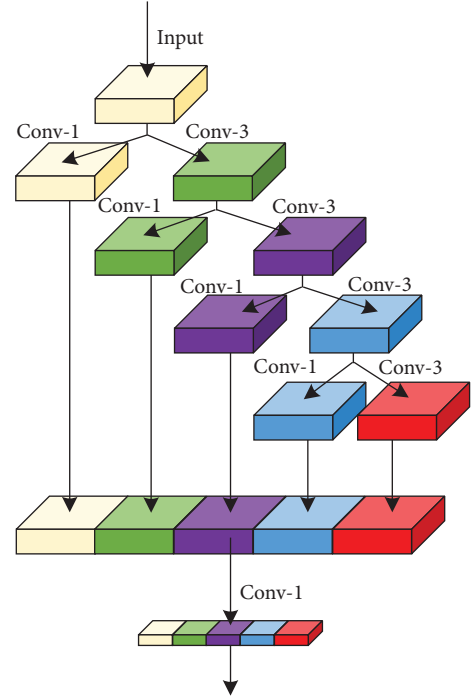


FIGURE 6: Architecture of the Tree module.

2.2.3. Focal Loss Function with Gradient Harmonizing Mechanism. The loss function of the original Mask R-CNN consists of three components: the loss of Fast R-CNN, the loss of Region Proposal Network (RPN), and the loss of Mask. Furthermore, the loss of Fast R-CNN is composed of the loss of classification and the loss of prediction box regression. The loss function for classification is the cross-entropy (CE) loss function:

$$\text{CE}(p, y) = \begin{cases} -\log(p), & \text{if } y = 1, \\ -\log(1-p), & \text{otherwise.} \end{cases} \tag{11}$$

In equation (11), y takes the value 1 or -1 , representing the foreground or background, respectively. p denotes the probability that the model classifies a pixel as a foreground, and its value ranges from 0 to 1. Here, a function of p is defined as follows:

$$p_t = \begin{cases} p, & \text{if } y = 1, \\ 1-p, & \text{otherwise.} \end{cases} \tag{12}$$

Combining with equation (12), equation (11) can be simplified as follows:

$$\begin{aligned}
\text{CE}(p, y) &= \text{CE}(p_t) \\
&= -\log(p_t).
\end{aligned} \tag{13}$$

It can be seen that CE represents the difference between the real probability distribution and the predicted probability distribution. The smaller the value of CE, the better the model prediction. However, under complex traffic conditions, the positive and negative samples in the vehicle

detection dataset are extremely unbalanced. In addition, most of the vehicles to be detected are irregular in shape and belong to the difficult samples. Therefore, directly using the CE loss function for object classification can result in significant errors. Therefore, the Focal Loss [34] (FL) function, which can automatically assign weights based on the difficulty of the sample, is employed in this study:

$$\text{FL}(p_t) = -\alpha(1 - p_t)^\gamma \log(p_t), \quad (14)$$

where α is the weighting factor for regulating the positive and negative sample imbalance, $(1 - p_t)$ is used to adjust the weights of difficult and easy samples, a small p_t indicates a poor classification and a difficult sample, a large p_t indicates a good classification and an easy sample, and γ is a moderator; that is, by increasing or decreasing γ , easy or difficult samples can be better trained. In order to determine the distribution of easy or difficult samples in the dataset, a gradient norm g is defined:

$$g = \begin{cases} |p - p_{\text{gt}}| \\ = \begin{cases} 1 - p, & p_{\text{gt}} = 1, \\ p, & p_{\text{gt}} = 0, \end{cases} \end{cases} \quad (15)$$

where p is the probability of model prediction and p_{gt} is the label of ground truth. It can be found that g is proportional to the difficulty of the sample, and the larger g is, the harder it is to detect. In this study, the distribution between the gradient norms and the fraction of samples is plotted in Figure 7.

As can be seen from Figure 7, the fraction of samples with a gradient norm close to 0 is the largest. The proportion of samples decreases rapidly as the gradient norm increases but is also large as the gradient norm approaches 1, which means that the number of both easy and particularly difficult samples is large. During the training process, easy samples and extremely difficult samples may present additional challenges. This is because the network is unable to learn new features from very easy samples and adjust the weights effectively. Furthermore, the gradient norm “ g ” for these extremely difficult samples is significantly larger than the norm for an average sample. As a result, the accuracy of the model may be reduced. Therefore, both easy and particularly difficult samples should be suppressed in training. In this study, the gradient density (GD) is defined to measure the fraction of samples within a certain gradient range and is used to balance the easy and particularly difficult samples, which can be expressed as follows:

$$\text{GD}(g) = \frac{1}{l_\varepsilon} \sum_{k=1}^N \delta_\varepsilon(g_k, g), \quad (16)$$

where

$$\begin{cases} \delta_\varepsilon(x, y) \begin{cases} 1, & y = \frac{\varepsilon}{2} \leq x + \frac{\varepsilon}{2}, \\ 0, & \text{otherwise.} \end{cases} \\ l_\varepsilon = \min\left(g + \frac{\varepsilon}{2}, 1\right) - \max\left(g - \frac{\varepsilon}{2}, 0\right). \end{cases} \quad (17)$$

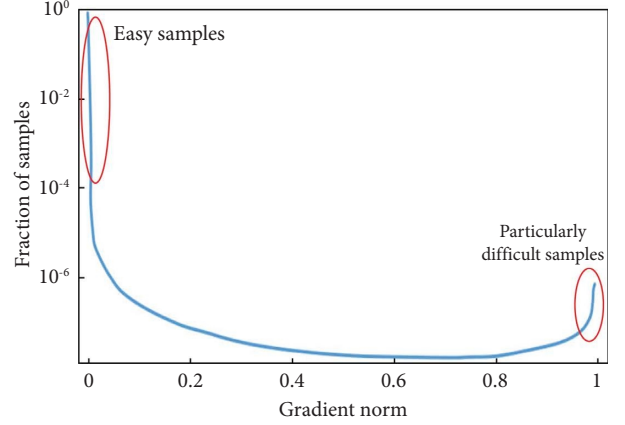


FIGURE 7: Gradient norm and fraction of examples.

In this study, a gradient harmonizing mechanism (GHM) is proposed to modify the FL function based on GD. Specifically, by dividing FL by GD, the loss of large GD is also suppressed. This way, both easy and particularly difficult samples are suppressed simultaneously so that the samples are harmonized in gradients:

$$L_{\text{GHM-C}} = \sum_{i=1}^N \frac{\text{FL}(p_i)}{\text{GD}(g_i)}. \quad (18)$$

2.3. Detection and Quantification of the Vehicle’s Outer Contour. In the preceding section, the incomplete calibration of the camera, the determination of the vanishing point, and the segmentation of vehicle pixels were presented. Subsequently, based on the view geometry, a vehicle’s outer contour dimension detection and quantification method is developed. The algorithm has two major steps: (1) detect the 3D bounding box of the vehicle in the image and (2) determine the actual length, width, height, and driving angle of the vehicle based on the pixel coordinates of the 3D bounding box. Compared with some existing approaches, the proposed algorithm can accurately identify the 3D bounding box of the vehicle at any driving angle. To demonstrate this superiority, a detailed comparison will be presented in Section 4. After detecting the 3D bounding box of the vehicle in the image, it is easy to directly obtain the dimensions of the vehicle in pixels. Next, based on the incomplete calibration introduced in Section 2.1, the real dimensions of the vehicle can be identified. The details of the proposed method will be presented separately in the following sections.

2.3.1. Detection of the Vehicle’s 3D Bounding Box. In Section 2.1, a method for vanishing point determination is presented. By connecting these two vanishing points, the vanishing line, which is a line consisting of vanishing points in the horizontal direction, can be obtained. Based on vanishing points and vanishing lines, the vehicle’s 3D bounding box can then be detected in each image frame. In some existing studies [18],

the 3D bounding box of a vehicle can be determined by two fixed vanishing points. However, there must be a strict assumption that the vehicle must be perfectly parallel to the direction of the road. When the vehicle travels at an angle to the direction of the road, the results will have significant errors. To demonstrate this, two top-down views of the vehicle with different driving angles are compared in Figures 8(a) and 8(b). As can be seen from Figures 8(a) and 8(b), when using two fixed vanishing points to construct the vehicle's 3D bounding box, the results change significantly with the change of the vehicle's driving angle. Therefore, for each vehicle, it is necessary to determine the vanishing point only from the edgelets corresponding to each vehicle (referred to as the vehicle's vanishing point). In addition, when detecting the edgelets of each vehicle, the area of the vehicle in the image needs to be selected as the ROI using the network introduced in Section 2.2, as shown in Figure 8(c). After obtaining two vehicle's vanishing points, the 3D bounding box of the vehicle at any driving angle can then be constructed using the proposed method, as shown in Figure 8(d).

The central task of constructing a 3D bounding box for a vehicle is to determine the pixel coordinates of the 8 vertices of this box. The procedure to determine the pixel coordinates of these 8 vertices is illustrated in Figure 9. Figure 9(a)–9(c) represent the three calculation steps, and Figure 9(d) shows the final results. In Figure 9, the red line indicates the new boundary line added at each step, the blue line indicates the boundary line of the final constructed 3D bounding box, and the yellow points indicate the vertices of the 3D bounding box. The details of the three steps illustrated in Figure 9 are as follows:

- (i) *Step 1.* Through the two vanishing points of the vehicle, three lines tangent to the vehicle instance are first determined (L1~3 in Figure 9(a)). Since the third vanishing point is at infinity, it is also possible to determine two vertical lines tangent to the head and tail of the vehicle (L4-5 in Figure 9(a)). Depending on the intersection of the five lines, four vertices can be determined (P1-4 in Figure 9(a)).
- (ii) *Step 2.* Next, L6, L7, and L8 can be determined by connecting the two vehicle vanishing points and the vertices P3, P1, and P6, respectively. L9 can then be determined by making a vertical line through the vertex P2. L6 and L7 intersect at the vertex P5, L8 and L9 intersect at the vertex P6, and L4 and L8 intersect at the vertex P7.
- (iii) *Step 3.* L10 and L11 can be determined by connecting the two vehicle vanishing points and the vertices P4, P7, respectively. L10 and L11 intersect at the vertex P8. L12 can be determined by connecting the vertices P5 and P8.

After these three steps, the pixel coordinates of the eight vertices (P1-8) can be determined. In order to determine the real length, width, height, and driving angle of the vehicle, it is necessary to further convert these pixel dimensions to real dimensions, the details of which will be presented in the next subsection.

2.3.2. Identification of the Vehicle's Dimensions. After constructing the 3D bounding box, the dimensions of the vehicle can be constructed based on the view geometry. According to the previous section, the pixel coordinates of the vanishing points m and n , the pixel coordinates of the vehicle's vanishing points m_v and n_v , and the pixel coordinates $P_i\{(u_i, v_i)|i = 1, 2, \dots, 8\}$ of the eight vertices can be obtained from the image. The real spatial shape and top-down view of the vehicle are plotted in Figure 10. Note that the numbering order of the vertices in Figure 10 is different from that in Figure 9. In Figure 10, L_v , W_v , and H_v represent the true dimensions of the vehicle, respectively. $P_i\{(x_i, y_i, z_i)|i = 1, 2, \dots, 8\}$ are the coordinates of the vehicle's 3D bounding box vertices in the world coordinate system. $P_1, P_2, P_3,$ and P_4 are located at the road surface, and therefore the corresponding $z_i = 0$ ($i = 1, 2, 3, 4$). The distances of $P_5, P_6, P_7,$ and P_8 from the ground are the vehicle height and the corresponding $z_i = H_v$ ($i = 5, 6, 7, 8$). The dimensions of the vehicle include the length, width, height, and driving angle of the vehicle, which are determined as follows:

- (i) Determine the vehicle's length and width:

As shown in Figure 10, the vertices $P_1, P_2, P_3,$ and P_4 are located at the surface of the road. Therefore, they have $z_i = 0$ ($i = 1, 2, 3, 4$). Then, substituting their pixel coordinates into (5) yields

$$\begin{cases} u = \frac{h_{11}x + h_{12}y + h_{14}}{h_{31}x + h_{32}y + h_{34}}, \\ v = \frac{h_{21}x + h_{22}y + h_{24}}{h_{31}x + h_{32}y + h_{34}}. \end{cases} \quad (19)$$

By solving the above equation set, the real coordinates $(x_{1\sim 4}, y_{1\sim 4})$ of $P_{1\sim 4}$ can be obtained from their pixel coordinates $(u_{1\sim 4}, v_{1\sim 4})$. Based on a simple geometric relationship, the length L_v and width W_v of the vehicle can be obtained as follows:

$$\begin{aligned} L_v &= \sqrt{(x_2 - x_3)^2 + (y_2 - y_3)^2}, \\ W_v &= \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}. \end{aligned} \quad (20)$$

- (ii) Determine the vehicle height:

When there is no reference with a known height, H_v cannot be determined directly from the 2D image. In this study, H_v is determined with the help of geometric constraints. Because of the large number of geometric constraints available, it is necessary to establish optimization equations to obtain the optimum result. Specifically, the coordinates of $P_{5\sim 6}$ are as follows:

$$\begin{aligned} P_j &= (x_j, y_j, H_v), \\ j &= 5, 6, 7, 8, \end{aligned} \quad (21)$$

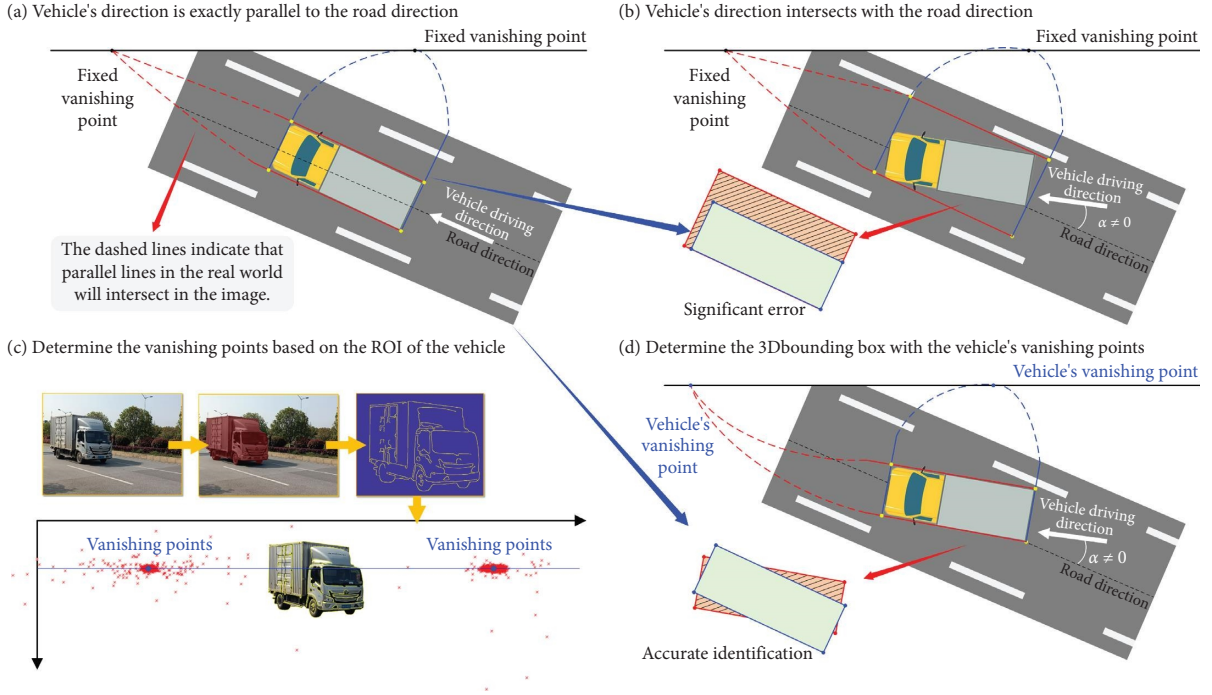


FIGURE 8: Comparison of identifying the vehicle's 3D bounding box in different conditions.

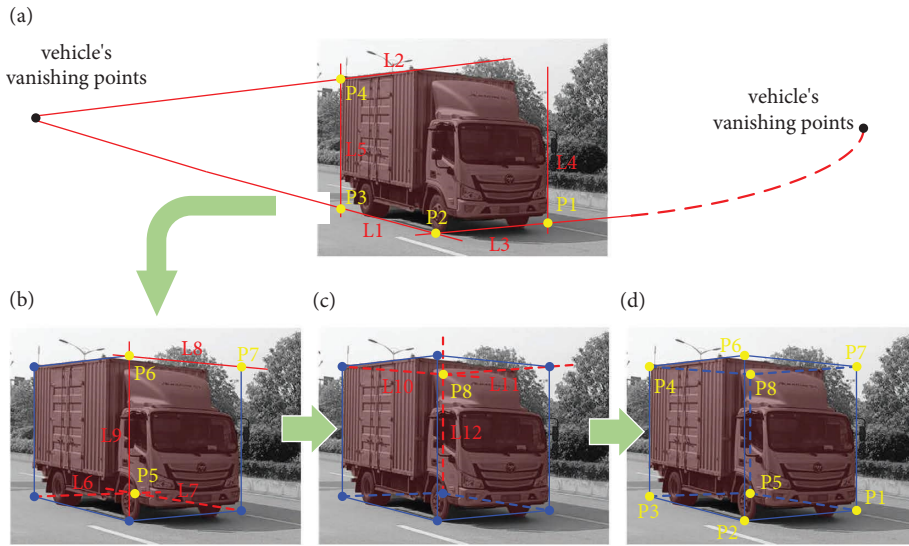


FIGURE 9: Three steps to construct the vehicle's 3D bounding box and the final result.

where $(x_{5\sim 6}, y_{5\sim 6}) = (x_{1\sim 4}, y_{1\sim 4})$. Then, the corresponding pixel coordinates are obtained by substituting (21) into (5):

$$\tilde{p}_j = (\tilde{u}_j, \tilde{v}_j), \quad (22)$$

$$j = 5, 6, 7, 8.$$

The pixel coordinates $\{P_j = (u_j, v_j) | j = 5, 6, 7, 8\}$ read from the image are subtracted from the above equation to form a nonlinear system of equations:

$$\mathbf{F}(H_v) = (u_5 - \tilde{u}_5, v_5 - \tilde{v}_5, \dots, u_8 - \tilde{u}_8, v_8 - \tilde{v}_8). \quad (23)$$

Thus, a least squares problem can be constructed:

$$\begin{cases} \arg_{H_v} \min \frac{1}{2} \|\mathbf{F}(H_v)\|^2, \\ \text{s.t. } H_{\min} \leq H_v \leq H_{\max}. \end{cases} \quad (24)$$

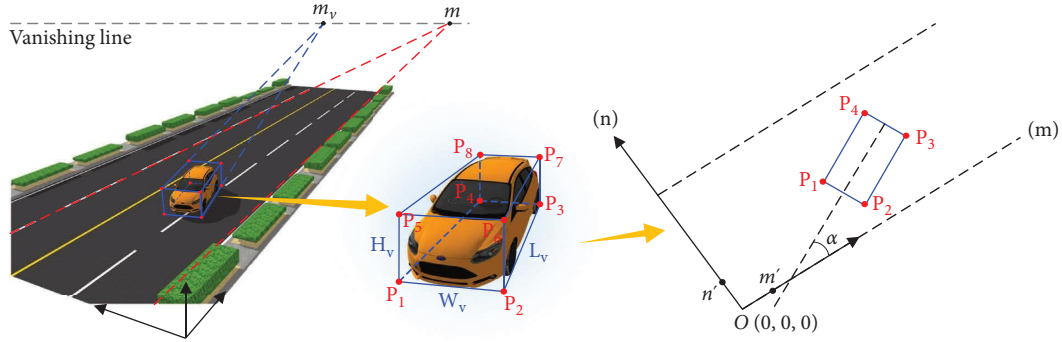


FIGURE 10: Vehicle's 3D bounding box diagram.

where H_{\min} and H_{\max} are the lower and upper limits of H_v , which can be determined empirically based on the upper and lower limits of local traffic statistics. The initial vehicle height H_v is the average of H_{\min} and H_{\max} .

(iii) Determine the vehicle angle:

First, according to Section 2.1, two vanishing points m and n in the road scene can be identified. Then, the two points m' and n' can be selected arbitrarily from the segments Om and On . The pixel coordinates of m' and n' can be converted to world coordinates using (5). Note that the z -coordinates of m' and n' are zero. Finally, the vehicle angle α_v can be determined as follows:

$$\alpha_v = \arctan \frac{\|y_3 - y_2\|}{\|x_3 - x_2\|} - \arctan \frac{y_{m'}}{x_{m'}}. \quad (25)$$

After the above steps, the length, width, height, and driving angle of the on-road vehicle can be determined. When applied to monitor the dimensions of passing vehicles in different traffic conditions, the thresholds for these four parameters can be set according to specific requirements. When a vehicle with dimensions exceeding the thresholds is detected, it can be immediately warned and directed off the road. The accuracy and effectiveness of the proposed method will be analyzed in the following section.

3. Experiments and Discussion

In this section, the performance of the proposed method is experimentally analyzed. First, the dataset establishment, model construction, and implementation details are presented. Then, the accuracy of the proposed method is verified in a field test. Subsequently, in real traffic conditions, the accuracy of the vehicle segmentation is validated. In addition, to demonstrate the superiority of the constructed model, two other advanced deep learning algorithms,

namely, You Only Look at CoefficientTs ++ (YOLACT++) [35] and DeepLabv3+ [36], are compared. Finally, the proposed method is compared in detail with two other typical vision-based methods to identify the dimensions of vehicle.

3.1. Dataset Establishment and Implementation Details

3.1.1. Datasets Establishment. The training and validation datasets used in this study are from the public dataset developed by Sochor et al. [37], while the test images are self-collected actual road traffic images [36]. The dataset contains images of vehicles with different environments, camera shooting angles, and weather conditions, and the details of the dataset assignment are shown in Table 1. As the network input size is 512 pixels \times 512 pixels, to prevent image distortion, the raw images are all scaled equally, and the excess is filled with gray pixels. The comprehensive dataset ensures the performance of the vehicle instance segmentation model and thus contributes to the accurate identification of the vehicle's dimensions. It is worth noting that the image is annotated with the VGG image annotator, i.e., a manual annotation software developed by the Visual Geometry Group (VGG) [38].

3.1.2. Training Configuration. In this study, the Ubuntu 20.04 operating system with Pytorch 1.9.1, CUDA 11.0, and the CUDNN 8.04 deep learning framework is used to implement the vehicle instance segmentation. The hardware configuration is a SuperCloud R8428 G11 and an Nvidia GeForce RTX 3060. The dataset is divided into training and validation sets in the ratio of 9 : 1, and the Adam optimizer is used during the training process. The hyperparameters considered include batch size, learning rate, and weight decay. The optimal hyperparameters are obtained by validation, and the specific values are listed in Table 2.

3.1.3. Evaluation Metrics. To evaluate the modified Mask R-CNN and the proposed vehicle's dimension identification method, the following four metrics are considered:

TABLE 1: Image dataset for vehicle detection and segmentation.

		Training	Validation	Testing
Modified Mask R-CNN	Number of images	6858	762	120
	Resolution (pixel)	512×512	512×512	1920×1080

TABLE 2: The well-tuned hyperparameters for modified Mask R-CNN.

	Batch size	Learning rate	Weight decay	Training epoch	Optimizer
Modified Mask R-CNN	64	0.0004	0.0005	100	Adam

(i) Mean average precision (mAP):

mAP is a performance metric that can indicate the accuracy of the vehicle localization and category prediction tasks, which can be expressed as follows:

$$\begin{aligned} \text{mAP} &= \frac{\text{AP}}{N} \\ &= \frac{\sum_1^N \int_0^1 P(R) dR}{N}, \end{aligned} \quad (26)$$

where R is the precision, P is the recall, and N is the number of vehicle categories. The closer the mAP is to 1, the better the model is.

(ii) Intersection over union (IoU):

IoU represents the similarity between the predicted and real regions of the vehicle in the image, defined by the following equation:

$$\text{IoU} = \frac{\text{area}(T_a \cap T_b)}{\text{area}(T_a \cup T_b)}, \quad (27)$$

in which T_a and T_b refer to the real vehicle pixels and the predicted vehicle pixels, respectively.

(iii) Vehicle's dimension evaluation index:

The identified and real vehicle's dimensions are $S_v = (W_v, L_v, H_v, \alpha_v)$ and $S_r = (W_r, L_r, H_r, \alpha_r)$, respectively. Then, the relative error (RE) of the identified vehicle's dimensions can be obtained as follows:

$$\left\{ \begin{array}{l} \text{RE}_L = \frac{|L_v - L_r|}{L_r} \times 100\%, \\ \text{RE}_W = \frac{|W_v - W_r|}{W_r} \times 100\%, \\ \text{RE}_H = \frac{|H_v - H_r|}{H_r} \times 100\%, \\ \text{RE}_\alpha = \frac{|\alpha_v - \alpha_r|}{\alpha_r} \times 100\%. \end{array} \right. \quad (28)$$

3.2. Accuracy Verification on Experimental Road. To verify the accuracy of the proposed method, the dimensions of the test vehicles were identified and compared with their true values. During the test, a fixed-focus camera with a fixed pitch angle was employed. As the test vehicle passes through

the camera's FOV at different driving angles, images are taken from different distances to determine the vehicle's dimensions.



The test vehicle and the test road are shown in Figure 11. For safety, the test was conducted on a closed experimental road. The test vehicle was a Tiguan L, with the measured dimensions of $4194 \text{ mm} \times 1760 \text{ mm} \times 1560 \text{ mm}$ (length \times width \times height). The rationale for employing a small car as a test vehicle, rather than a large one, is as follows: (1) The essence of the algorithm proposed in this study for identifying vehicles with dimensions exceeding the limit is precise identification of the vehicle's dimensions. The dimensions of the vehicle determine whether or not the vehicle is deemed oversized, based on a human-defined threshold. During the evaluation of the approach, it is critical to focus on the conformity between the dimensions identified and the actual dimensions of the vehicle. The size of the vehicle, whether it is "small" or "large," has no impact on the method's function or on the final results' precision. In case the threshold is set artificially smaller than the dimensions of a "normal" small car, the dimensions of the small car can still be precisely determined and classified as a vehicle whose dimensions surpass the threshold, namely, an "oversized" vehicle. (2) In this study, field tests were conducted on the experimental road and real roads, respectively. In the experimental road, due to the restriction of the closed test site, only small vehicles are allowed to enter, while large engineering vehicles are prohibited. In addition, the scheduling of the large vehicles would be very difficult to match the different photographing needs under different parameters. Therefore, only "normal" small cars were used in the experimental road. And in the next section on real roads, all types of vehicles passing through the measurement points will be considered, both "large" and "normal" vehicles.

In order to verify the effectiveness of the proposed method for vehicles under different driving conditions, three cases of straight ahead, U-turn, and lane change were considered. The real driving angles of the vehicles in the three cases were obtained based on the top-down photos taken by the unmanned aircraft. In addition, when the vehicle passes the FOV of the camera, the vehicle is photographed at 5 m, 20 m, and 35 m from the camera, respectively. The focal length of the camera is 8 mm, the pitch angle $\varphi = 12.9^\circ$, the shooting angle $\theta = 37.5^\circ$, and the camera height $h = 3.2 \text{ m}$. The specific models of the camera and lens are listed in Table 3. Based on the internal and external parameters of the camera, the transformation matrix \mathbf{H} of the camera can be described as follows:



FIGURE 11: When the test vehicle passes the FOV of the camera, photos are taken as it travels to different distances.

TABLE 3: Specifications of the employed camera.

Equipment	Model	Specifications
Camera	 MindVision/MV-XG1205GC/M	Maximum resolution: 4096 × 3072 Maximum frame rate: 409 fps Chroma: Bayer8 Lens mount: C-mount Sensitivity: 4050 mV 1/30 s
Lens	 MindVision/MV-LD-8-5M-C	Focal length: 8 mm Aperture: F1.6~F22 Mount: C-mount

$$\mathbf{H} = \mathbf{KRT}$$

$$= \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ -\sin\varphi\sin\theta & -\sin\varphi\cos\theta & -\cos\varphi \\ \cos\varphi\sin\theta & \cos\varphi\cos\theta & -\sin\varphi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -h \end{bmatrix}. \quad (29)$$

After obtaining the images of the test vehicle, the identified dimensions of the test vehicle in each case are obtained with the proposed method, and the results are summarized in Table 4. As can be seen from Table 4, when the test vehicle is at a distance of 20 m, the vehicle dimensions can be accurately identified under all three conditions (error <4%), which validates the accuracy and feasibility of the proposed method. However, there are significant errors when the vehicle is straight ahead (35 m and 5 m) or making a U-turn (5 m). Specifically, it can be seen from the table that

- (1) When the vehicle is turned around at 5 m, there is an obvious distortion in the 3D bounding box constructed with the proposed method (error = 57.39%, 19.20%, 29.35%, and 45.56%). This is because the side face or front face of the vehicle cannot be fully photographed at these camera angles, so the vanishing point corresponding to the edgelets on the front face cannot be accurately determined.
- (2) When the vehicle is travelling straight ahead at a distance of 5 m (angle = 0.62), there is also an obvious error for travelling angle identification

(error = 50%). This is due to the fact that the true value is very small (e.g., 0.62°) and that an absolute error of only 0.31° still leads to a larger relative error (50%).

- (3) When the vehicle is at 35 m, the limited resolution of the camera can also lead to a lack of clarity in the images. Blurred vehicles in the image can directly affect the accuracy of vehicle instance segmentation, which will thus reduce the accuracy of identifying vehicle dimensions (error = 14.43%, 22.10%, 17.95%, and 1076.00%).

Indeed, the use of ultrahigh-resolution cameras can be effective in increasing the measurement distance. However, this tends to increase the cost of the device significantly. In addition, it is possible to experimentally determine the optimum camera distance with the highest accuracy. However, in real-world traffic situations, the road may have three or more lanes. Once the optimum camera distance has been determined, this fixed value is hardly suitable for vehicles travelling in all lanes. And to determine the vehicle distance, other measuring devices such as laser rangefinders and RGBD cameras are needed. Therefore, it was not and is

TABLE 4: The vehicle dimensions under different conditions.

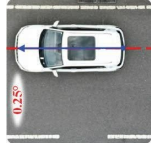



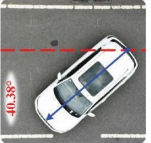

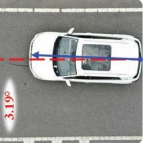

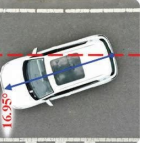



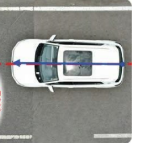



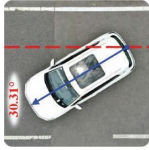
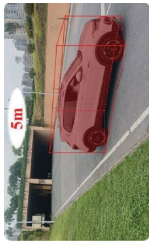
Angle	Camera distance	True dimensions	Identified dimensions	Relative errors (%)
		4194 mm	3589.00 mm	14.43
		1760 mm	1371.00 mm	22.10
		1560 mm	1840.00 mm	17.95
		4194 mm	3987.00 mm	4.94
		1760 mm	1587.00 mm	9.83
		1560 mm	1627.00 mm	4.30
		4194 mm	3992 mm	4.82
		1760 mm	1658 mm	5.80
		1560 mm	1485 mm	4.81
		4194 mm	4098.00 mm	2.29
		1760 mm	1698.00 mm	3.52
		1560 mm	1590.00 mm	1.92
		4194 mm	4123.00 mm	1.69
		1760 mm	1754.00 mm	0.34
		1560 mm	1551.00 mm	0.58
		4194 mm	4110.00 mm	2.00
		1760 mm	1699.00 mm	3.47
		1560 mm	1601.00 mm	2.63
		4194 mm	4107.00 mm	2.07
		1760 mm	1739.00 mm	1.19
		1560 mm	1540.00 mm	1.28
		0.62°	0.93°	50.00

TABLE 4: Continued.

Angle	Camera distance	True dimensions	Identified dimensions	Relative errors (%)
		4194 mm	4089.00 mm	2.50
		1760 mm	1801.00 mm	2.33
		1560 mm	1521.00 mm	2.50
		12.61°	12.03°	4.60
		4194 mm	6601.00 mm	57.39
		1760 mm	2098.00 mm	19.20
		1560 mm	2018.00 mm	29.35
		30.31°	44.12°	45.56

not necessary to obtain the optimum camera distance in this study. Considering that, on a normal road, the vehicle will always approach the roadside camera from far to near. In this study, a video of the passing vehicle is captured, and the vehicle's dimensions are identified frame by frame. When the identification result is stable, the mean value of the identification result of the next video frame is taken as the final result, and the detailed analysis will be presented in the next subsection.

3.3. Field Experiment on Real Road. To further verify the feasibility and accuracy of the proposed method in real traffic scenarios, field experiments were also conducted on a section of real road. The experimental site is at the Sanchaji Bridge over the Xiangjiang River in Changsha, Hunan Province, China. Videos of passing vehicles are captured by an industrial camera (MindVision/MV-XG1205GC/M) with a resolution of 1920×1080 pixels and a frame rate of 30 fps. The camera is located on the right side of the road, and the captured traffic scene is shown in Figure 12. The ground truth of vehicle length, width, and height is obtained by querying the vehicle types. Note that in the algorithm proposed in this study, it is not necessary to measure any reference object of known dimensions in advance. The camera height is 3 m, and the camera angles $\theta = 49.2^\circ$ and $\varphi = 19.3^\circ$.

It is obvious that the accuracy of the proposed method relies on the accurate identification of edgelets on the vehicle. However, edge detection from a single video frame often turns out to be unreliable in practice. This is due to the fact that the size of different vehicles varies greatly, and when small vehicles are too far away from the camera, the edgelet details in the image can be severely blurred. In addition, as noted in the previous section, the front or side of the vehicle may not be fully captured at certain camera angles. To address these problems, a simple and feasible strategy is to capture the entire video of a vehicle passing from far to near. That is, the vehicle's dimensions are identified frame by frame from the video, and the average values are determined as the final result. Figure 13 shows an example of identifying the vehicle's dimensions from a sequence of video frames. The variation of the four identified dimensions with increasing number of video frames is plotted in Figure 13(b). The vertical coordinate in the figure is the relative error of the value identified in each frame to the mean value after convergence. It can be seen that the values of the four identified parameters gradually converge and remain stable after 25 frames. Admittedly, the number of video frames required to achieve stabilization should be different for vehicles of different speeds. However, it has been found experimentally that the identified parameters of normal vehicles can all reach stability after 30 frames under general traffic conditions. Therefore, in this study, the video is recorded starting from the 30th frame after the vehicle is identified and continues to be recorded until the vehicle leaves the camera's FOV. The average value of the vehicle's dimensions determined in each frame of the recorded video is taken as the final result.

The accuracy of the proposed method for the final identification of the vehicle's dimensions also depends on the quality of the vehicle instance segmentation. Therefore, a comparison with the original Mask R-CNN is necessary to validate the superiority of the modified model. In addition, two other instance segmentation networks based on different architectures (YOACT++ and DeepLabv3+) are tested in this subsection, and the test results are compared with the proposed algorithm. The same 120 images are used in the test, and the test details and evaluation metrics are listed in Table 5. As can be seen from the table, the average 2D IoU of the segmentation results of the modified ResUNet reaches 93.35%, while the average IoU of the original Mask R-CNN, YOACT++, and DeepLabv3+ is only 77.91%, 68.21%, and 65.20%, respectively. The result indicates that the proposed method significantly outperforms the original Mask R-CNN, DeepLabv3+, and YOACT++ networks in vehicle detection and segmentation under real traffic conditions. For a more visual comparison, part of the vehicle segmentation results is also shown in Figure 14. As can be seen from Figure 14, for the modified Mask R-CNN, the interference of complex backgrounds in the images, as well as the interference of shadows, is effectively eliminated. In addition, the segmented vehicles have more accurate edges compared to the original Mask R-CNN. This is due to the fact that meaningless information in the background is more effectively suppressed by embedding attention blocks. In addition, the focal loss function with a gradient harmonizing mechanism can extract the vehicle features more efficiently. Overall, the modified Mask R-CNN can segment vehicle pixels from captured images more accurately than other models, which is an important foundation for accurate identification of a vehicle's dimensions in this study.

In the field experiment, a total of 10 minutes of video were collected to identify the dimensions of passing vehicles and to determine separately whether the dimensions of each vehicle exceeded the safety threshold. Since there is no actual dimensional limit required for the experimental road, artificial thresholds are set here for length, width, height, and driving angle: 5 m, 2 m, 3 m, and 5° , respectively. Note that these thresholds are only used to evaluate the proposed method and have no practical meaning. In practical applications, these thresholds need to be set according to the specific road condition requirements. After excluding the completely obscured vehicles, a total of 172 vehicles were counted in the video, and the results are plotted in Figure 15. In Figure 15, the length, width, height, and driving angle of the supervised vehicle are counted as bars. The manually set thresholds corresponding to these four parameters are drawn as four pink planes in the figure, i.e., vehicles exceeding the thresholds are considered oversized (highlighted in purple in the figure). As can be seen in Figure 15, the proposed method can visually detect potentially endangered vehicles when oversized vehicles are present. Road supervisors can warn the vehicle in the first instance and direct it off the road to avoid a serious collision.

To verify the accuracy of the proposed method under real traffic conditions, four vehicles are randomly selected from the results. The identification results of these vehicles are



FIGURE 12: The location of the camera on the Sanchaji highway bridge over the Xiangjiang River in Changsha, China.

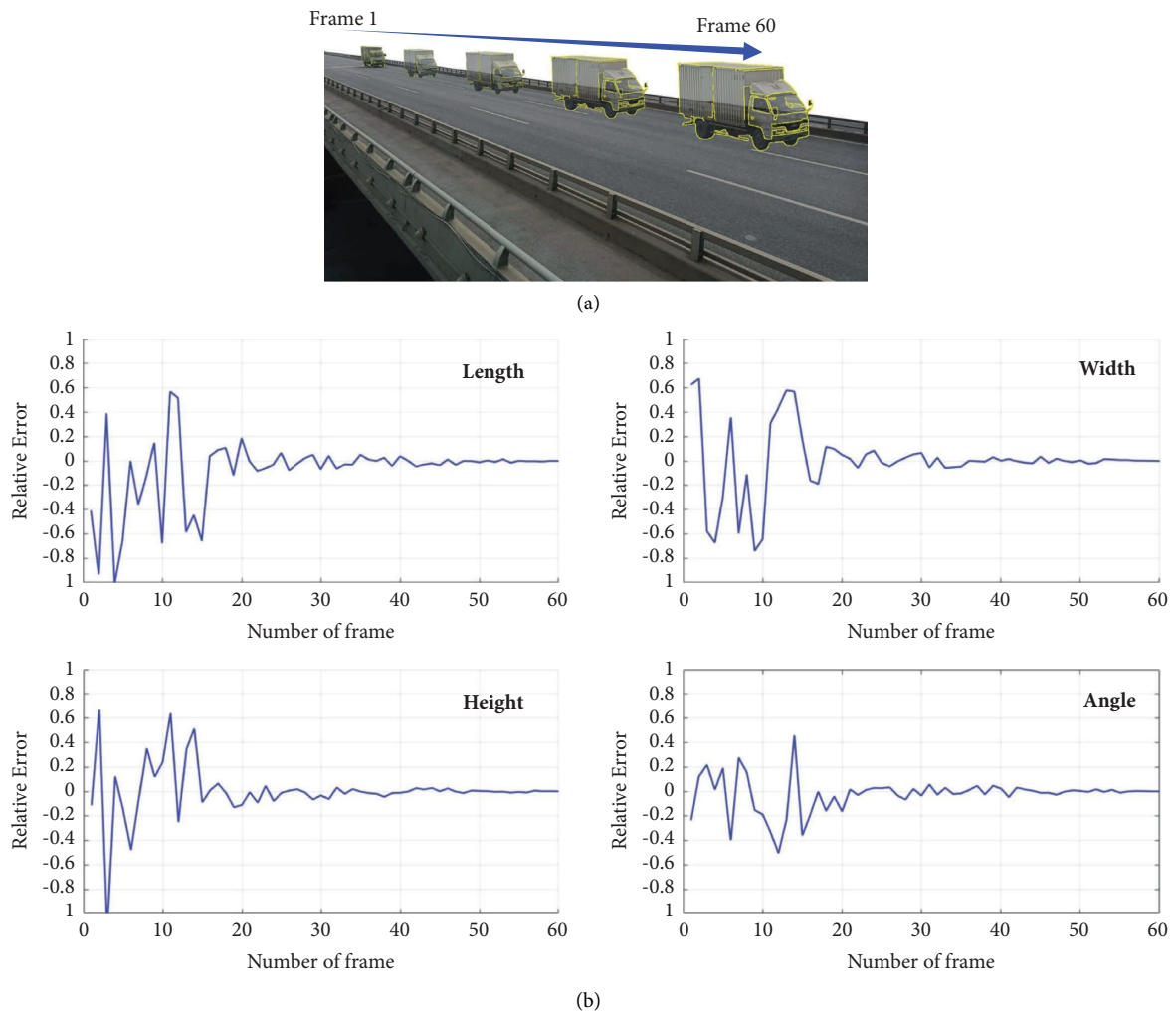


FIGURE 13: Frame-by-frame identification of the vehicle’s dimensions: (a) filming of passing vehicles; (b) variation of the identified vehicle’s dimensions with video frames.

illustrated in Figure 16. As can be seen from the figure, the determined vehicle’s dimensions can be approximated to follow a normal distribution. This is because, in the process

of photographing each vehicle, it is driven to the camera from far to near. When the vehicle is far from or very close to the camera, it can lead to large errors due to unclear vehicle

TABLE 5: Details and evaluation metrics for the four methods of training and testing.

	Mask R-CNN	YOLOACT++	DeepLabv3+	Modified Mask R-CNN
Label types	Mask	Mask	Mask	Mask
Training data	Open-source	Open-source	Open-source	Open-source
Testing data	Self-collection	Self-collection	Self-collection	Self-collection
Number of test images	120	120	120	120
Average 2D IoU (%)	77.91	68.21	65.20	93.35



FIGURE 14: Comparisons of the original Mask R-CNN, YOLOACT++, DeepLabv3+, and modified Mask R-CNN for segmenting vehicles from images with complex traffic backgrounds.

contours or lens distortion. Therefore, in this study, the average value of the identified vehicle's dimensions in a sequence of video images is obtained as the final result. From the identification results, it can be seen that the stability of identification can be effectively improved by calculating the average value of several consecutive frames of images. And the identification errors of the four selected vehicles are all within 4%.

3.4. Comparisons with Existing Vision-Based Vehicle's Dimensions Estimation Methods. In addition to the traditional methods mentioned in the introduction, a number of vision-based methods have now been developed for vehicle's dimensions identification. To further demonstrate the superiority of the proposed method, the results of the proposed method and other existing methods are compared in three typical traffic scenarios. The three typical scenarios include

vehicles making a U-turn or changing lanes, vehicles with occlusion, and vehicles with a shadow. In some studies [39–42], although the detection of vehicle length, width, or height is proposed, the establishment of a 3D bounding box is not involved and is not considered here. Therefore, the studies of Lu et al. [18] and Zhu et al. [22] are reasonably selected for comparison in this study.

Figure 17 compares the 3D bounding boxes of the vehicles (middle column) and their dimensions (right column) as determined using the three different methods for the three typical traffic scenarios. The figure also presents the ground truth of the vehicle dimensions and the vehicle type. For comparison, the identification results and limitations of the three methods are summarized in detail in Table 6. As can be seen from Figure 17 and Table 6, in all three typical scenarios, the vehicle dimensions estimated by the existing methods differ significantly from the ground truth. Particularly, the method proposed in the study by Lu et al. [18]

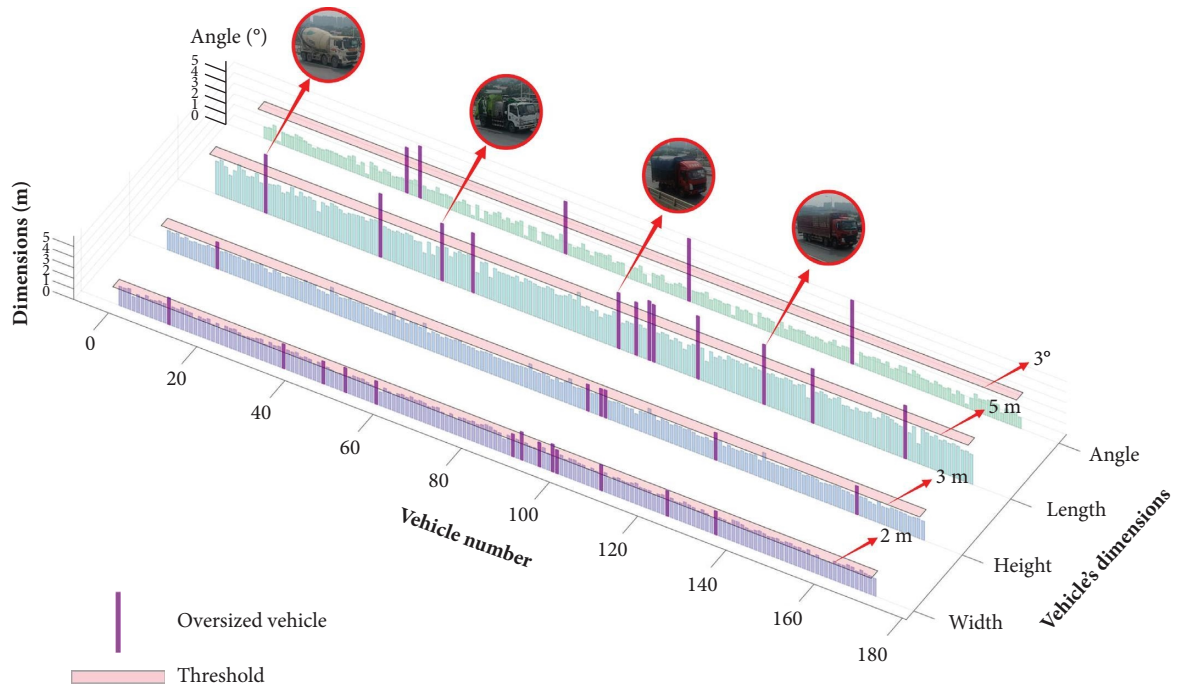


FIGURE 15: The identified length, width, height, and angle of 172 vehicles.

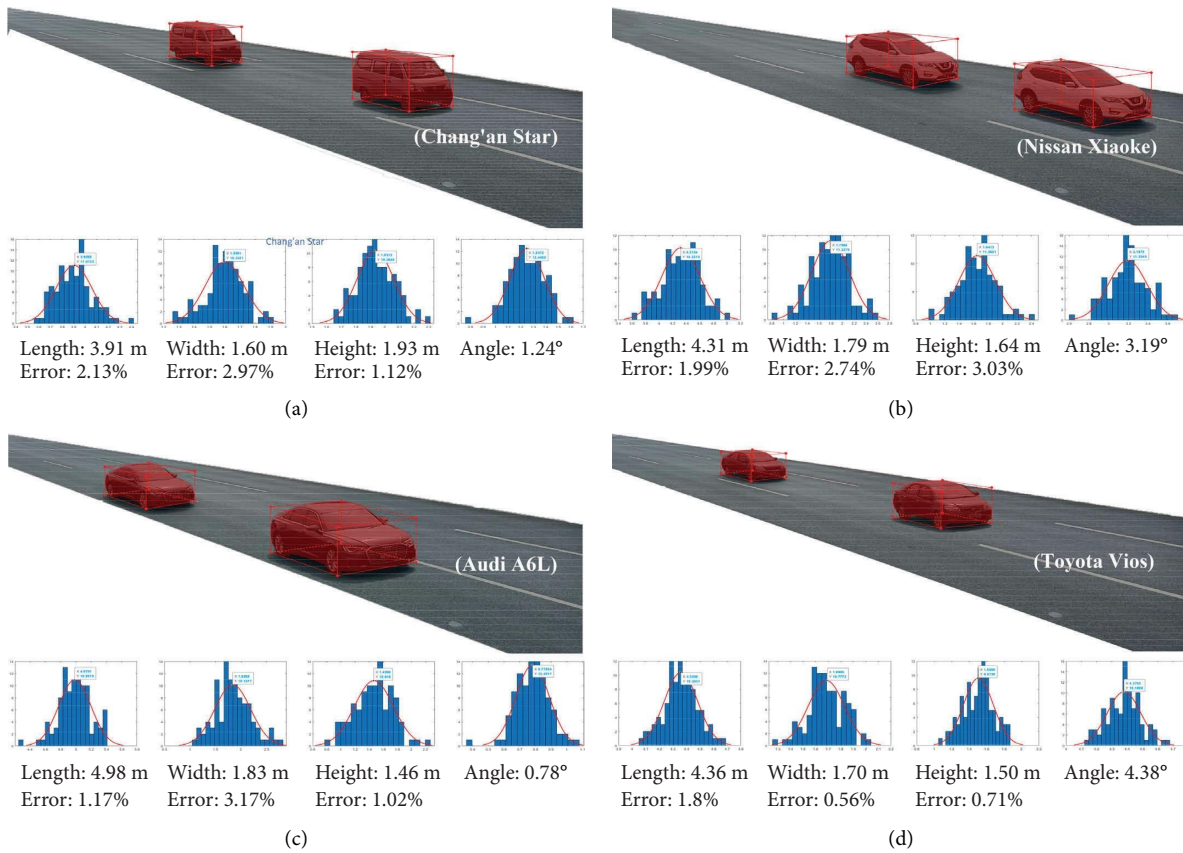


FIGURE 16: 3D bounding boxes of vehicles and histograms of dimensions identified from consecutive image frame sequences.

TABLE 6: Comparison of the three methods.

	Lu and Dai [18]	Zhu et al. [22]	The proposed method
Maximum error	9.85	28.3%	2.8%
Camera calibration method	9.6%	9.0%	4.05
Vehicle segmentation algorithm	16.8%	13.3%	3.6%
Applicable (✓) to special traffic situations or not (✗)?	Calibration board YOLOv4	Calibration bar Mask R-CNN	Modified Mask R-CNN
Vehicle changing lanes or making a U-turn	✓	✗	✓
Vehicles with occlusion	✓	✓	✓
Vehicle with shadow	✗	✗	✓



FIGURE 17: The results of identifying vehicles' dimensions in special traffic situations using three methods: (a) vehicles are changing lanes or making a U-turn; (b) there is occlusion between vehicles; (c) there is shadow interference around vehicles.

shows glaring flaws in constructing the 3D bounding box of the vehicle when it is making a U-turn or lane change. This is because when the vehicle is not parallel to the road direction, assuming the vanishing point of the vehicle as the vanishing point of the road will cause a considerable error. Both existing methods and the method proposed in this study can identify the dimensions of a vehicle when the vehicle is obscured or has vehicle shadows. However, the vehicle instance segmentation algorithm used in this study adds an attention mechanism and a gradient harmonizing mechanism that can detect vehicle pixels more accurately in complex backgrounds. As a result, the identification error of

the method proposed in this study is greatly reduced. In particular, when vehicle shadows exist, the vehicle instance segmentation algorithm used in the existing method may not distinguish the vehicle from the vehicle shadow, which will result in a large error in the identification of the dimensions of the vehicle. In addition, after obtaining the 3D bounding box of the vehicle, the existing algorithms must have a reference of known size to calibrate the camera when determining the true size of the vehicle. However, such a procedure not only increases the complexity of the method but is also simply unusable in some road sections where it is difficult to find a reference. The existence of these situations

greatly hinders the widespread application of these two existing methods in real traffic. On the contrary, in three typical cases, the vehicle length, width, and height obtained by the proposed algorithm agree well with the ground truth (error <4%). This is mainly attributed to its accurate segmentation of vehicle contours using the modified Mask R-CNN and the accurate recovery of the vehicle's dimensions by the proposed algorithm. In addition, the proposed algorithm for recovering the vehicle's dimensions does not require the camera to be calibrated by a reference in the scene. This makes the proposed method more practical in real-life traffic scenes.

4. Conclusions

This study proposed a computer vision-based vehicle size detection method to detect and quantify the external contour dimensions of a vehicle using a roadside camera. The research methodology consists of the following steps: acquisition of traffic scenes to calibrate the camera, detection of the target vehicle, segmentation of vehicle instances using a modified Mask R-CNN, construction of the 3D bounding box of the vehicle based on the view geometry, and determination of the vehicle's length, width, height, and angle. To assess the precision and efficacy of the method, field experiments were conducted and the results were compared with other existing methods. The study reveals that

- (1) The improved Mask R-CNN provides higher accuracy in vehicle pixel segmentation (average 2D IoU >0.93%) compared to the original model and other commonly available models.
- (2) The method proposed is capable of precisely determining the dimensions, comprising length, width, height, and driving angle, of a moving vehicle at a reasonable shooting distance (error <5%).
- (3) The proposed technique can accurately detect vehicle dimensions even when the vehicle is at a nonright angle to the road, when the vehicle is obscured, when there is shadow interference, etc. In addition, the method eliminates the need to fully calibrate the camera by measuring a known-sized reference object.

Overall, the proposed method exhibits excellent performance and is expected to be a cost-effective alternative to conventional vehicle size detection systems. In practical applications, the method can be used to detect oversized vehicles, thus leading to a considerable reduction in possible traffic accidents.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to acknowledge the financial support provided by PowerChina Road Bridge Group Co., Ltd.

References

- [1] D. Feng and M. Feng, "Output-only damage detection using vehicle induced displacement response and mode shape curvature index," *Structural Control and Health Monitoring*, vol. 23, no. 8, pp. 1088–1107, 2016.
- [2] Y. Yu, X. Zhao, Y. Shi, and J. Ou, "Design of a real-time overload monitoring system for bridges and roads based on structural response," *Measurement*, vol. 46, no. 1, pp. 345–352, 2013.
- [3] G. Wiesaw and G. Anna, *Oversize Cargo Transport in the Polish Part of South Baltic Region*, Institute of Aviation, Szczecin, Poland, 2011.
- [4] J. Petru and V. Krivda, "The process of setting the parameters for ensuring passage of oversized cargos," *The Baltic Journal of Road and Bridge Engineering*, vol. 14, no. 3, pp. 425–442, 2019.
- [5] Y. S. Zhang, X. Z. Lu, J. Ning, and J. J. Jiang, "Computer simulation for the impact between over-high truck and composite viaduct," *Traffic and Computer*, vol. 25, no. 3, pp. 65–69, 2007.
- [6] M. Kozman and R. Stevens, *Overheight Vehicle Detection System (OVDS)*, Texas Department of Transportation, Austin, TX, USA, 2014.
- [7] G. McLauchlan and A. Belch, "Emergency management of a patient following a road traffic accident," *Companion Animal*, vol. 19, no. 10, pp. 512–516, 2014.
- [8] D. X. Wu, W. P. Li, and G. P. Zheng, "Survey analysis and countermeasure research on traffic and fire accidents in expressway tunnels of Zhejiang Province," *Highways*, vol. 8, p. 75, 2011.
- [9] B. Nguyen and I. Brilakis, "Understanding the problem of bridge and tunnel strikes caused by over-height vehicles," *Transportation Research Procedia*, vol. 14, pp. 3915–3924, 2016.
- [10] P. Cawley, *Evaluation of Overheight Vehicle Detection/warning Systems*, Today's Transportation Challenge: Meeting Our Customer's Expectations, Palm Harbor, FL, USA, 2002.
- [11] A. Ozdagli, F. Moreu, D. Xu, and T. Wang, "Experimental analysis on effectiveness of crash beams for impact attenuation of overheight vehicle collisions on railroad bridges," *Journal of Bridge Engineering*, vol. 25, no. 1, Article ID 4019133, 2020.
- [12] M. Yang and P. Qiao, "Analysis of cushion systems for impact protection design of bridges against overheight vehicle collision," *International Journal of Impact Engineering*, vol. 37, no. 12, pp. 1220–1228, 2010.
- [13] C. Smith, M. Rowley, C. Dvonch, and M. Fulton, "Non-destructive evaluation of metal and composite targets using an infrared line-scanning technique," in *Proceedings of the Society of Photo-Optical Instrumentation Engineering (SPIE)*, Orlando, FL, USA, March 2005.
- [14] B. Nguyen and I. Brilakis, "Real-time validation of vision-based over-height vehicle detection system," *Advanced Engineering Informatics*, vol. 38, pp. 67–80, 2018.
- [15] M. Rezaei, M. Azarmi, and F. Mir, "Traffic-net:3D Traffic Monitoring Using a Single Camera," 2022, <https://arxiv.org/abs/2109.09165>.
- [16] D. Lu, V. C. Jammula, S. Como, J. Wishart, Y. Chen, and Y. Yang, "Carom-vehicle localization and traffic scene

- reconstruction from monocular cameras on road infrastructures,” in *Proceedings of 2021 IEEE International Conference on Robotics and Automation, ICRA*, Xi’an, China, June 2021.
- [17] B. Zhang, L. M. Zhou, and J. Zhang, “A methodology for obtaining spatiotemporal information of the vehicles on bridges based on computer vision,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 34, no. 6, pp. 471–487, 2019.
- [18] L. J. Lu and F. Dai, “Automated visual surveying of vehicle heights to help measure the risk of overheight collisions using deep learning and view geometry,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 38, no. 2, pp. 194–210, 2023.
- [19] Y. Liu, D. Han, R. Cao, J. J. Guo, and L. Deng, “Automated vehicle wheelbase measurement using computer vision and view geometry,” *Measurement Science and Technology*, vol. 34, no. 12, Article ID 125051, 2023.
- [20] Q. Q. Zhu, S. Liu, and W. M. Guo, “Research on vehicle appearance component recognition based on mask R-CNN,” in *Proceedings of 3rd International Conference on Computer Graphics and Digital Image Processing (CGDIP)*, Rome, Italy, July 2019.
- [21] N. K. Kanhere and S. T. Birchfield, “A taxonomy and analysis of camera calibration methods for traffic monitoring applications,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 2, pp. 441–452, 2010.
- [22] J. S. Zhu, X. T. Li, C. Zhang, and T. Shi, “An accurate approach for obtaining spatiotemporal information of vehicle loads on bridges based on 3D bounding box reconstruction with computer vision,” *Measurement*, vol. 181, Article ID 109657, 2021.
- [23] W. Wang, C. Y. Zhang, X. Y. Tang, H. Song, and H. Cui, “Automatic self-calibration and optimization algorithm of traffic camera in road scene,” *Journal of Computer-aided Design & Computer Graphics*, vol. 31, no. 11, pp. 1955–1962, 2019.
- [24] R. C. Zhang, Q. Y. Du, Z. L. Yu, H. Liu, and K. Zhang, “A calibration method for road monitoring cameras exploiting reference images and roadway information,” *Journal of Highway and Transportation Research and Development*, vol. 31, no. 11, pp. 137–141, 2014.
- [25] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” in *Proceedings of the IEEE International Conference on Computer Vision*, Marrakesh, Morocco, October 2017.
- [26] H. T. Li, Z. Todd, N. Bielski, and F. Carroll, “3D lidar point-cloud projection operator and transfer machine learning for effective road surface features detection and segmentation,” *The Visual Computer*, vol. 38, no. 5, pp. 1759–1774, 2021.
- [27] R. Maini and H. Aggarwal, “Study and comparison of various image edge detection techniques,” *International Journal of Image Processing*, vol. 3, no. 1, pp. 1–11, 2009.
- [28] M. Dubská, A. Herout, R. Juránek, and J. Sochor, “Fully automatic roadside camera calibration for traffic surveillance,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1162–1171, 2015.
- [29] 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.
- [30] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, “Squeeze and excitation networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 8, pp. 2011–2023, 2020.
- [31] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, Las Vegas, NV, USA, June 2016.
- [32] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4700–4708, Honolulu, HI, USA, July 2017.
- [33] L. Rar, “TreeNet: a lightweight one-shot aggregation convolutional network,” 2021, <https://arxiv.org/abs/2109.12342>.
- [34] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 318–327, 2020.
- [35] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, “YOLACT++: better real-time instance segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 2, pp. 1108–1121, 2022.
- [36] L. C. Chen, Y. Zhu, and G. Papandreou, “Encoder-decoder with atrous separable convolution for semantic image segmentation,” in *Proceedings of the European Conference on Computer Vision*, Munich, Germany, September 2018.
- [37] J. Sochor, R. Juránek, J. Špaňhel et al., “Comprehensive data set for automatic single camera visual speed measurement,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 5, pp. 1633–1643, 2019.
- [38] A. Dutta and A. Zisserman, “The VIA annotation software for images, audio and video,” in *Proceedings of the 27th ACM International Conference on Multimedia*, Nice, France, October 2019.
- [39] J. Shao, S. K. Zhou, and R. Chellappa, “Robust height estimation of moving objects from uncalibrated videos,” *IEEE Transactions on Image Processing*, vol. 19, no. 8, pp. 2221–2232, 2010.
- [40] B. Nguyen, I. Brilakis, and P. A. Vela, “Optimized parameters for over-height vehicle detection under variable weather conditions,” *Journal of Computing in Civil Engineering*, vol. 31, no. 5, Article ID 04017039, 2017.
- [41] V. Khorramshahi, A. Behrad, and N. K. Kanhere, “Over-height vehicle detection in low headroom roads using digital video processing,” *International Journal of Transport and Vehicle Engineering*, vol. 2, no. 3, pp. 681–685, 2008.
- [42] F. Dai, M. W. Park, M. Sandidge, and I. Brilakis, “A vision-based method for on-road truck height measurement in proactive prevention of collision with overpasses and tunnels,” *Automation in Construction*, vol. 50, pp. 29–39, 2015.