vehicles

MDPI

Article

Unauthorized Expressway Parking Detection Based on Spatiotemporal Analysis of Vehicle–Structure Distances Using UAV Aerial Images

Xiaolong Gong, Haiqing Liu *, Yuehao Wang, Yaxin Wei and Guoran Shi

School of Transportation and Logistic Engineering, Shandong Jiaotong University, Jinan 250357, China;
xiaolonggong0219@gmail.com (X.G.); yuehaowang66@gmail.com (Y.W.); yaxinwei21@gmail.com (Y.W.);
guoranshi580@gmail.com (G.S.)
* Correspondence: hqliu@sdjtu.edu.cn

Abstract

Owing to their high-altitude vantage point and maneuverability, unmanned aerial vehicles (UAVs) have emerged as an effective technical solution for real-time parking detection in expressway scenarios. Using UAV cruise-perspective images, this paper proposes an unauthorized parking detection method by analyzing the time-series variations in the relative distances between the moving vehicle and static structure as a reference. Firstly, vehicle and static structure targets are recognized and tracked by the DeepSort, and a Vehicle–Structure (V-S) distance matrix is further constructed to describe their frame-wise relative positions in the pixel coordinate system. Then, to eliminate the radial scale errors caused by perspective distortion, a scale factor (SF) index is introduced to correct the original V-S matrix and provide a more accurate spatiotemporal representation. Finally, the stationarity of the distance series in the V-S matrix is tested using the Augmented Dickey–Fuller (ADF) test, and a parking detection method is proposed by introducing the parking support ratio (PSR) to establish a multi-structure joint decision scheme. Experimental results show that the corrected V-S matrix can faithfully describe the spatial positional relationship between road vehicles and static structures. With the optimal PSR threshold ψ_0 and time window T, the proposed method achieves better overall parking-detection performance in terms of accuracy, precision, recall, and F1-score in comparison with a traditional speed threshold approach.

Keywords: expressway parking detection; UAV; vehicle–structure distance matrix; time series analysis

Academic Editor: Tuan Nguyen Gia

Received: 11 December 2025
Revised: 26 February 2026
Accepted: 4 March 2026
Published: 6 March 2026

1. Introduction

Expressways are a critical component of the transportation network. The operational safety and efficiency of expressways have become a paramount concern in both academia and the industry. Due to the enclosed road structure and high-speed vehicle motion, parking on expressways will lead to serious safety issues, such as severe vehicle collisions and secondary accidents. Hence, accurate and reliable parking detection in real time on expressways is fundamental for implementing effective vehicle management strategies and improving road safety. According to the types of sensors used, parking detection methods can be broadly categorized into non-vision-based approaches, like geomagnetic sensors [1–3], LiDAR sensors [4,5], ultrasonic sensors [6], and vision-based approaches. Owing to their intuitive imaging capability, rich feature representation, and the high reusabil-

https://doi.org/10.3390/vehicles8030049

ity of existing video surveillance devices, vision-based approaches have become a research hotspot in parking detection.

Traditional vision-based parking detection methods mainly rely on basic image-processing techniques, such as gray-level analysis and background modeling. In Ref. [7], to address illumination changes caused by shadows and lighting variations in complex road scenarios, the robustness of the HSV color model can be exploited to construct a stable background, and vehicle parking states can be detected using frame differencing. The authors in Ref. [8] presented a method for identifying parking-space occupancy based on the analysis of pixel-intensity variations in low-quality grayscale images, yielding a cost-effective solution for parking detection with roadside cameras. In Ref. [9], parking-cell regions were segmented according to gray-level density and the distribution of segment areas was further analyzed for parking detection using a roadside camera. However, these methods excessively rely on prior knowledge and require manual configurations of intensity statistics, color rules, or gray-level thresholds. Deep-level image features cannot be fully exploited to describe vehicle states, which leads to performance degradation in dynamic complex road scenarios. Consequently, in addition to gray and color features, more cross-scale and cross-domain features need to be further exploited for parking detection.

Since deep learning (DL) approaches can effectively extract the hierarchical and context-aware features of targets, they can address the limitations of traditional image-processing-based methods and have become a crucial research direction for parking detection. In Ref. [10], an ensemble DL-based parking detection model was proposed. It combined HSV, discrete wavelet transform (DWT), discrete cosine transform (DCT), and gradient features in a CNN model to detect the parking occupancy status under various weather conditions. In Ref. [11], a DL-based parking-space detection method was proposed that took 360 degree panoramic images as its input and employed Faster R-CNN to localize parking spaces. Leveraging the powerful deep-level feature representation and self-learning capabilities of DL methods, parking detection research has also shifted from traditional static parking-space occupancy state recognition to dynamic parking events. Based on the basic road structure, the methods in Refs. [12,13] detected and tracked multiple vehicles targets (with Ref. [13] using YOLO-based detectors) and further recognized driving behaviors such as illegal parking and temporary stops. However, the aforementioned research has mainly focused on fixed roadside surveillance viewpoints, where the parking area has been static in the image. When the camera drifted or the road structure changed, the performance tended to degrade significantly.

Recently, unmanned aerial vehicles (UAVs) have been extensively applied in road monitoring [14,15]. Equipped with multi-sensors and edge computing devices [16–18], the UAVs can detect and track vehicle targets and further analyze real-time traffic events [19–21]. For static parking-space detection, the authors of [22,23] used UAVs to collect aerial perspective images of parking lots and applied feature matching or CNN-based methods to recognize the parking-space occupancy. For UAV-based dynamic parking detection, the authors in Ref. [24] developed SmartUAV-PS, a multi-UAV parking supervision system that used a pre-trained CNN-based detector to recognize improperly parked vehicles in real urban scenes. In Ref. [25], a parking-violation detection (PVD) framework that followed a suspect-and-investigate paradigm was proposed, which integrated an efficient unsupervised optical-flow CNN, a flow-guided detection RCNN, and a visual SLAM for illegally parked car identification. Although these studies have demonstrated the great potential of UAV platforms, many existing methods do not explicitly address the challenges introduced by UAV motion under realistic flight conditions, such as motion blur and drastic viewpoint changes. These issues remain key bottlenecks that affect detection accuracy and overall system reliability.

To address the aforementioned issues, this paper proposes a parking detection method using UAV cruise-perspective images for expressway scenarios. The main contributions of this work include two aspects: (1) A three-dimensional tensor based on the V-S distance matrix is built in the pixel coordinate system to describe the relative positional relationship between vehicles and static structures. (2) The stationarity of each distance series in the V-S matrix is analyzed using the Augmented Dickey–Fuller (ADF) test, and a parking detection method is presented referring to the parking support ratio (PSR), which is introduced to fuse the multi-structures decisions.

2. Modeling of the V-S Distance Matrix

The overall architecture of the proposed unauthorized parking detection method is illustrated in Figure 1.

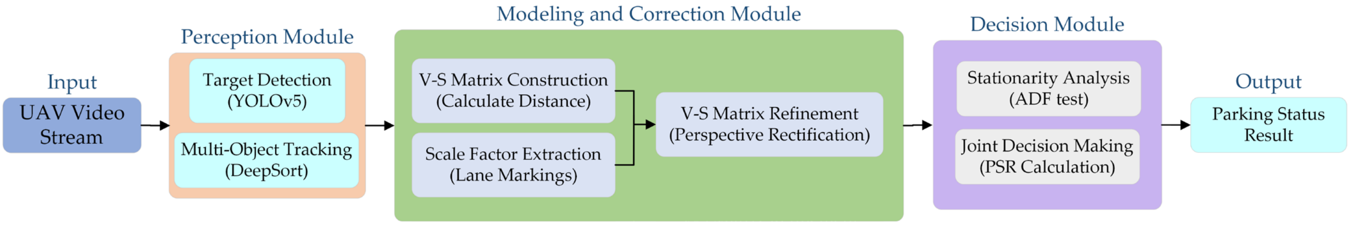


Figure 1. The overall system architecture.

2.1. Vehicle and Structure Detection and Tracking

The cruise UAV is used to capture videos of road traffic conditions and the surrounding environment in expressway scenarios. In this paper, YOLOv5 is used to extract multiple categories of road traffic targets. To meet the modeling requirements of parking detection, a labeling framework containing the vehicles and structures is presented, and the details are elaborated in Table 1. In this paper, static roadside structures (e.g., speed limit signs, gantries, and guidance arrows) are utilized as the references for unauthorized parking detection.

Table 1. Labeling of targets.

Category	Target Label
Vehicle	Passenger car Medium bus Large bus Light truck Medium truck Heavy truck
Structure	Speed limit sign Channelization zone Gantry Signboard Speed bump Guidance arrow

Using YOLOv5, vehicle and structure targets are detected. Furthermore, the targets are tracked using the DeepSort model. The resulting output data structure is summarized in Table 2.

Table 2. Target tracking output data structure.

Column	Data Type	Description	Example
Target_ID	int	The unique ID for tracking the target	'15'
Class	string	Category name of the target	'Passenger car'
Confidence	float	Detection confidence of targets (in range [0, 1])	'0.92'
Frame_ID	int	Video frame sequence number	'120'
Bbox	tuple	The upper-left and lower-right coordinates of the boundary box	'[(125,80), (245,180)]'
Timestamp	string	Timestamp in "YYYY-MM-DD HH:MM:SS" format	'2025-11-22 13:45:12'

2.2. Construction of the V-S Matrix

In this paper, the relative positions between moving vehicle targets and static structure targets in the image coordinate system are used for parking detection. Referring to Table 2, the vehicle target can be described by Equation (1).

$$V(i) = \{(x_{i,\min}^{v}, y_{i,\max}^{v}), (x_{i,\max}^{v}, y_{i,\min}^{v}), P_i^v, I_i^v, C_i^v\} \tag{1}$$

where $(x_{i,\min}^{v}, y_{i,\max}^{v})$ and $(x_{i,\max}^{v}, y_{i,\min}^{v})$ denote the upper-left and lower-right coordinates of the boundary box, respectively; P_i^v is the confidence; I_i^v is the target ID; and C_i^v is the vehicle category.

Similarly, the structure target can be described by Equation (2).

$$S(j) = \left\{(x_{j,\min}^{s}, y_{j,\max}^{s}), (x_{j,\max}^{s}, y_{j,\min}^{s}), P_j^s, I_j^s, C_j^s\right\} \tag{2}$$

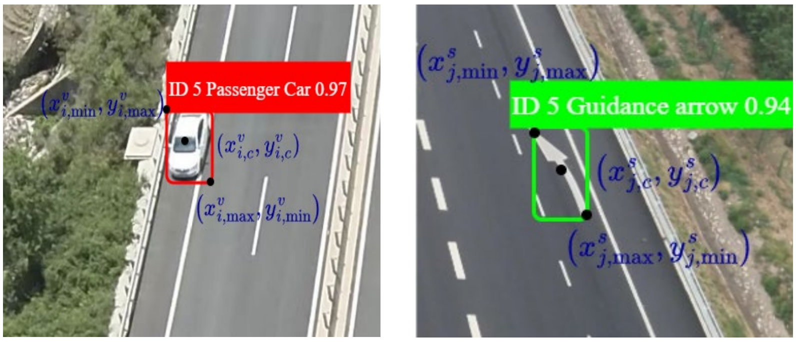
where $(x_{j,\min}^{s}, y_{j,\max}^{s})$ and $(x_{j,\max}^{s}, y_{j,\min}^{s})$ denote the upper-left and lower-right coordinates of the boundary box, respectively; P_j^s is the confidence; I_j^s is the target ID; and C_j^s is the structure category.

Referring to Equations (1) and (2), the center coordinates of the vehicle and structure targets in the image coordinate system can be expressed by Equations (3) and (4), respectively.

$$(x_{i,c}^{v}, y_{i,c}^{v}) = \left(\frac{x_{i,\min}^{v} + x_{i,\max}^{v}}{2}, \frac{y_{i,\min}^{v} + y_{i,\max}^{v}}{2}\right) \tag{3}$$

$$\left(x_{j,c}^{s}, y_{j,c}^{s}\right) = \left(\frac{x_{j,\min}^{s} + x_{j,\max}^{s}}{2}, \frac{y_{j,\min}^{s} + y_{j,\max}^{s}}{2}\right) \tag{4}$$

The center points of the detection boxes are illustrated in Figure 2.



(**a**) (**b**)

Figure 2. Position description of targets: (**a**) vehicle target; (**b**) structure target.

Supposing that there are M vehicles and N structures in the t-th sampling frame of the UAV video, the vehicle–structure distance matrix (V-S matrix) can be expressed by Equation (5).

$$D(t) = \begin{bmatrix} d_{11}(t) & d_{12}(t) & \dots & d_{1N}(t) \\ d_{21}(t) & d_{22}(t) & \dots & d_{2N}(t) \\ \vdots & \vdots & \ddots & \vdots \\ d_{M1}(t) & d_{M2}(t) & \dots & d_{MN}(t) \end{bmatrix} \tag{5}$$

where d_{ij} denotes the pixel Euclidean distance between the i-th vehicle and the j-th structure, which can be calculated by Equation (6).

$$d_{ij} = \sqrt{\left(x^v_{i,c} - x^s_{j,c}\right)^2 + \left(y^v_{i,c} - y^s_{j,c}\right)^2} \tag{6}$$

The V-S matrix describes the relative positions between static structures and moving vehicle targets in the image coordinate system, as shown in Figure 3.

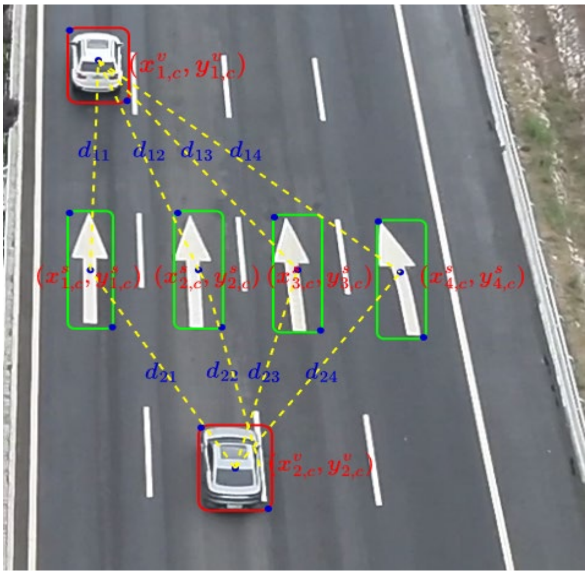


Figure 3. Relative position description by V-S matrix.

2.3. Correction of the V-S Matrix

Influenced by the forward pitch angle of the UAV camera during the cruise process, perspective distortion occurs in the captured road images and the pixel distance scale varies at different radial positions along the flight direction. As shown in Figure 4, as the UAV moves forward, the pixel distance between the sample vehicle and the guidance arrow increases, leading to errors of d_{ij} in the V-S matrix.

Referring to the perspective projection geometry illustrated in Figure 4, the mathematical model is constructed based on the standard pinhole camera model and the ground planar assumption [26,27]. This establishes the geometric mapping relationship between the 2D image coordinate system and the 3D world coordinate system. To explicitly describe this transformation, the pixel coordinate system o-xy and the world coordinate system O_w-$X_wY_wZ_w$ are defined as below.

Pixel coordinate system o-xy: the center of the lower edge of the image is set as the origin. The UAV flight direction is set as the y-axis and direction perpendicular to the y-axis is defined as the x-axis.

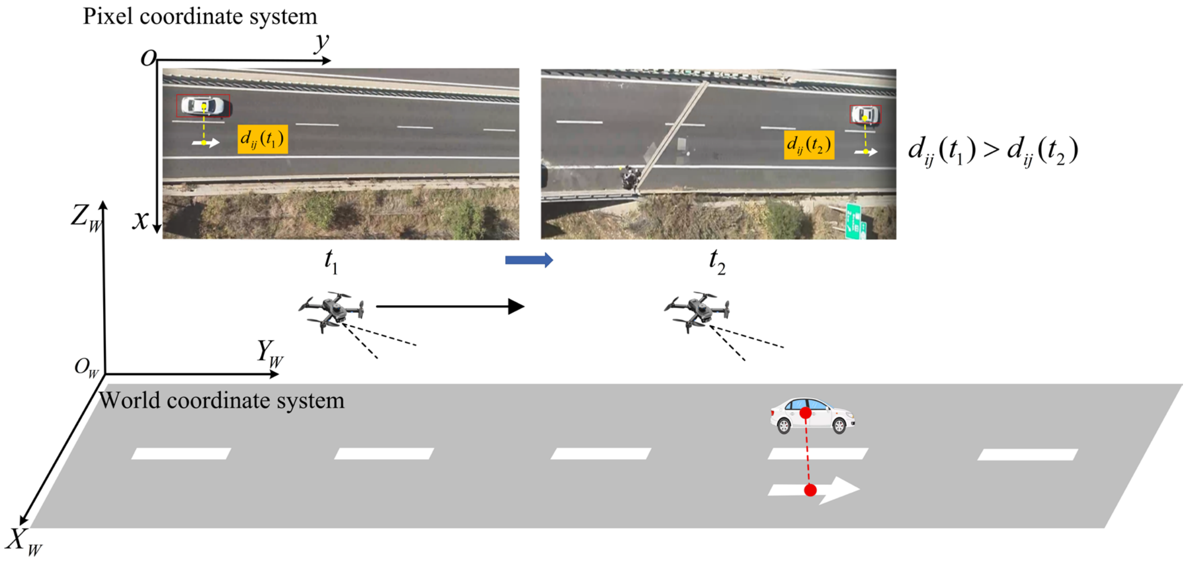


Figure 4. Perspective distortion of UAV-detected images.

World coordinate system O_w-$X_wY_wZ_w$: the vertical projection of the UAV onto the ground is set as the origin. The UAV flight direction is set as the Y_w-axis and direction perpendicular to the Y_w-axis is defined as the X_w-axis. The Z_w-axis is the upward direction perpendicular to the ground.

At a certain radial position, for a unit pixel length Δy in the o-xy pixel coordinate system, there exists a corresponding actual distance ΔY in the O_w-$X_wY_wZ_w$ world coordinate system. In order to quantitatively describe the correspondence between the two coordinate systems and further correct the errors of d_{ij} in the V-S matrix caused by perspective distortion, this paper introduces a SF using the physical length of dashed lane markings as a reference, where the lane markings can be extracted by image processing methods such as edge detection [28] or deep learning [29].

The direction of lane markings is relatively consistent with the UAV's flight direction. Lane markings at the same radial position are defined as one lane group. Supposing that there are P lane groups in the image and each lane group contains Q lane markings, the coordinates of the q-th lane marking in the p-th lane group can be expressed by Equation (7).

$$C_{p,q} = \left\{ (x_{p,q}^{up}, y_{p,q}^{up}), (x_{p,q}^{down}, y_{p,q}^{down}) \right\} \tag{7}$$

where $(x_{p,q}^{up}, y_{p,q}^{up})$ and $(x_{p,q}^{down}, y_{p,q}^{down})$ denote the upper and lower endpoints of the lane marking, respectively, as shown in Figure 5.

Referring to Figure 5, the average pixel length of all lane markings in the p-th lane group is calculated by Equation (8).

$$l_p^{avg} = \frac{1}{Q}\sum_{q=1}^{Q} \sqrt{\left(x_{p,q}^{up} - x_{p,q}^{down}\right)^2 + \left(y_{p,q}^{up} - y_{p,q}^{down}\right)^2} \tag{8}$$

The average distribution interval of all lane markings in the p-th lane group along the y-axis is expressed by Equation (9).

$$\left[y_p^{up}, y_p^{down} \right] = \left[\frac{1}{Q}\sum_{q=1}^{Q} y_{p,q}^{up}, \frac{1}{Q}\sum_{q=1}^{Q} y_{p,q}^{down} \right] \tag{9}$$

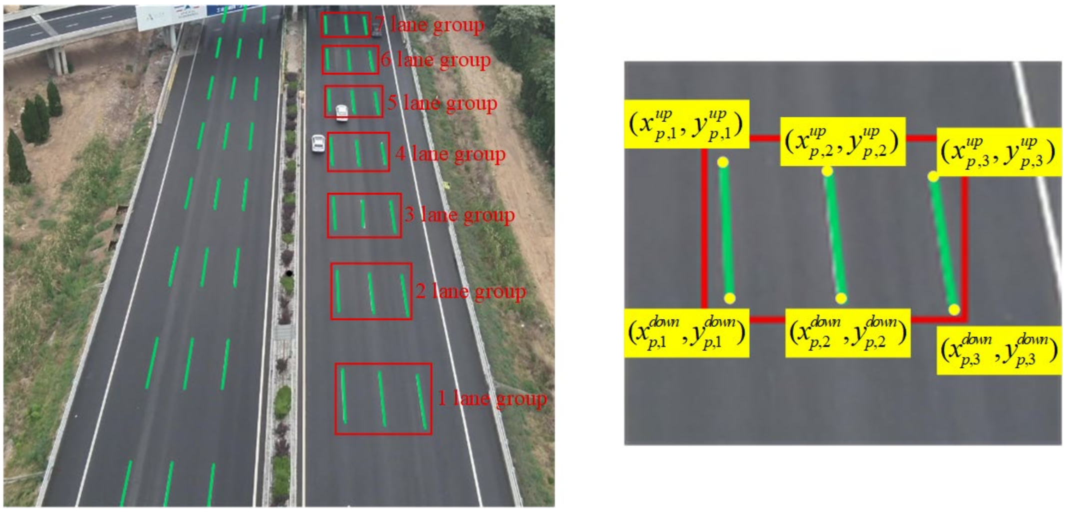


Figure 5. Description of lane groups and lane markings.

The ratio of the actual lane-marking length and the average pixel length in the same lane group is defined as the scale factor, as expressed by Equation (10).

$$\lambda_p = \frac{L}{l_p^{avg}} \tag{10}$$

where L denotes the actual length of a single lane-marking segment. According to relevant expressway standards and specifications in China, the length of a lane on an expressway is standardized to 4 m. To address the errors caused by road wear or partial occlusion, the SF is calculated based on the average pixel length of all lane markings within a lane group (as shown in Equation (8)). This mechanism effectively mitigates the impact of individual non-standard or incompletely detected markings.

When the average position of the lane markings' lower endpoints in the same lane group is located at the bottom edge of the image, the lane group is defined as the benchmark lane group and the calculated SF is defined as the basic scale factor (B-SF), as expressed by Equation (11).

$$\lambda_0 = \frac{L}{l_p^{avg}\Big|_{y_p^{down}=0}} \tag{11}$$

The B-SF is used as a benchmark to standardize the criteria for different SF values at different radial positions and to correct errors caused by perspective distortion.

The SF value increases with the radial pixel coordinate, reflecting that a unit pixel corresponds to a larger actual distance. To further describe this variation, the average y-coordinate of the lane markings' center points in each group is taken as the independent variable, and the corresponding scale factor is taken as the dependent variable, and a fitting function is proposed and shown by Equation (12).

$$\lambda(y_p) = k \cdot (y_p - y_0) + b \tag{12}$$

where y_0 is the average y-coordinate of the lane markings' center points in the benchmark lane group; and k and b are the slope and intercept to be estimated, respectively. Referring to Equation (12), the optimal values of k and b can be determined by linear regression, as expressed by Equation (13).

$$\min_{k,b} \sum_{u=0}^{U-1} \left[\lambda(y_p) - (k \cdot (y_p - y_0) + b)\right]^2 \tag{13}$$

Under different radial positions, the distribution of SF values and the fitting function are shown in Figure 6.

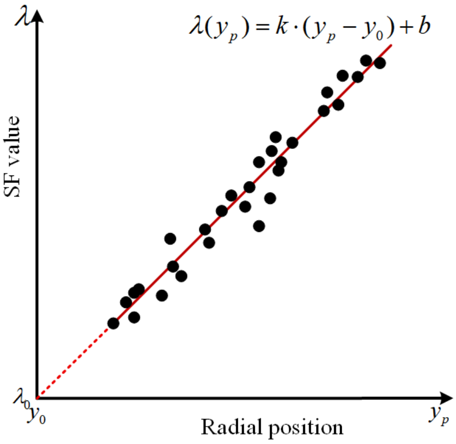


Figure 6. SF value distribution and the fitting function.

Referring to Equation (12), the SF is calculated with respect to the *y*-coordinate of the center point of the line between vehicle *i* and structure *j* as the independent variable, and is further used for the error correction of the pixel distance.

For each pixel distance value d_{ij} in the V-S matrix, the corrected value d^r_{ij} is expressed by Equation (14).

$$d^r_{ij} = d_{ij} \cdot \frac{\lambda(y_{ij})}{\lambda_0} \tag{14}$$

where

$$y_{ij} = \frac{y^v_{i,c} + y^s_{j,c}}{2} \tag{15}$$

In Equation (15), $y^v_{i,c}$ and $y^s_{j,c}$ are the vertical center coordinates of the bounding boxes of vehicle *i* and structure *j*, respectively.

By applying the above correction procedure frame by frame, the corrected V-S matrix is expressed by Equation (16).

$$D^r(t) = \begin{bmatrix} d^r_{11}(t) & d^r_{12}(t) & \dots & d^r_{1N}(t) \\ d^r_{21}(t) & d^r_{22}(t) & \dots & d^r_{2N}(t) \\ \vdots & \vdots & \ddots & \vdots \\ d^r_{M1}(t) & d^r_{M2}(t) & \dots & d^r_{MN}(t) \end{bmatrix} \tag{16}$$

3. Parking Detection Based on Time-Series Analysis of the V-S Matrix

The specific workflow of the proposed parking detection algorithm is presented in Figure 7.

The cruising UAV is used to record the road video, of which the frames are sampled at an interval of Δt. For each frame, the vehicle and structure targets are extracted and the V-S matrix is further constructed. To describe the temporal variation in the matrix, this paper introduces a structured three-dimensional spatiotemporal tensor that integrates consecutive discrete observations into a unified data model and is directly suitable for further time-series analysis.

The V-S matrices obtained at different sampled frames are stacked along the temporal dimension to form a three-dimensional time-series tensor, as expressed by Equation (17).

$$D = [D^r_1, \dots, D^r_t, \dots, D^r_T]^{\mathrm{T}} \tag{17}$$

where T denotes the length of the detection time window.

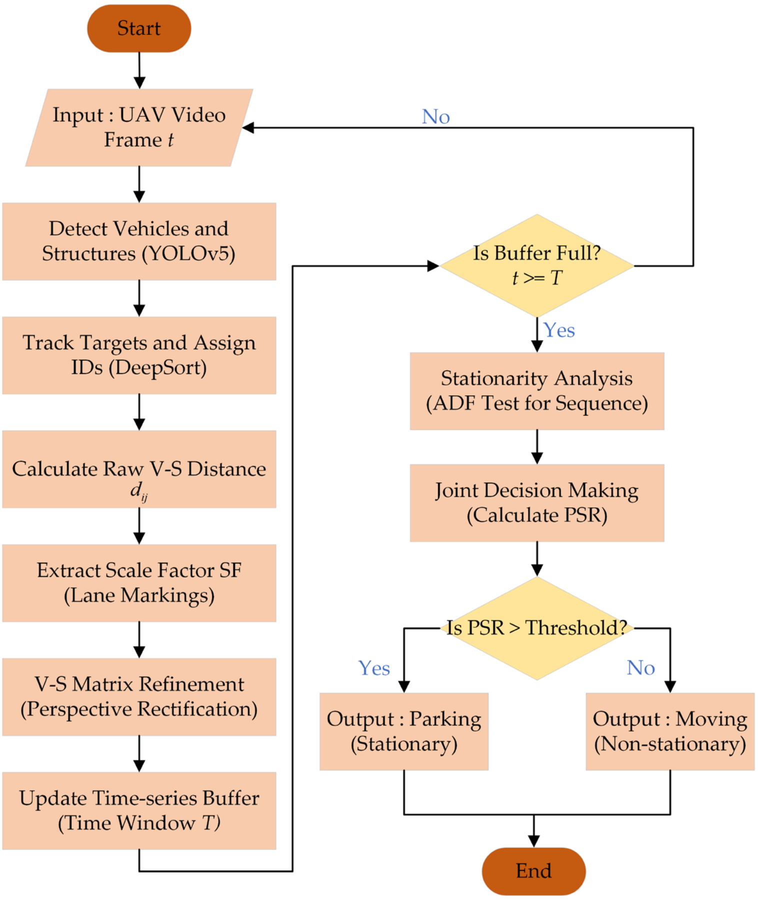


Figure 7. Flowchart of the detection process.

Within the time window T, O_s denotes the index set of structures that appear simultaneously in the image with vehicle i, as expressed by Equation (18).

$$O_s = \{o_1, o_2, \ldots, o_{N_s}\} \tag{18}$$

where N_s represents the total number of structures.

The sub-tensor corresponding to vehicle index i and structure indices belonging to O_s is extracted from the global tensor D, yielding the stationarity-analysis matrix U_i for vehicle i.

$$U_i = (d^r_{in_s}(t))_{t=1,\ldots,T;n_s=1,\ldots N_s} \in R^{T\times N_s} = \begin{bmatrix} d^r_{i1}(1) & d^r_{i2}(1) & \ldots & d^r_{iN_s}(1) \\ d^r_{i1}(2) & d^r_{i2}(2) & \ldots & d^r_{iN_s}(2) \\ \vdots & \vdots & \ddots & \vdots \\ d^r_{i1}(T) & d^r_{i2}(T) & \ldots & d^r_{iN_s}(T) \end{bmatrix} \tag{19}$$

where $d^r_{in_s}(t)$ denotes the corrected V-S distance between vehicle i and structure n_s at time t in tensor D.

In Equation (19), each column denotes the differing temporal variations in the spatial relationship between vehicle i and structure n_s. Each row denotes the spatial relationship between vehicle i and all structures at time t.

To determine whether there is relative position shift between the vehicle and the static structure, and thus to infer the parking status, the Augmented Dickey–Fuller (ADF) test is applied to analyze the stationarity of each column in U_i.

During the flight process, the relative distances between the parked vehicles and static structures remain relatively stable, manifesting as a time series oscillating around a constant mean value due to slight detection noise. Crucially, to effectively distinguish slow-moving vehicles from parked ones, the series must not exhibit a pronounced deterministic trend. Otherwise, including a trend term might erroneously interpret the systematic displacement as a trend-stationary process. Hence, an Augmented Dickey–Fuller (ADF) model with a constant term but without a deterministic trend term is adopted for the analysis, as expressed by Equation (20).

$$\Delta U_t = \alpha + \beta U_{t-1} + \sum_{k=1}^{\theta} \gamma_k \Delta U_{t-k} + \varepsilon_t \tag{20}$$

where ΔU_t denotes the first-order difference in the time series; α is the constant term; βU_{t-1} is the lagged level term, under the null hypothesis $H_0 : \beta = 0$; the series is considered to contain a unit root; the coefficients $\gamma_k (k = 1, \ldots, \theta)$ correspond to the θ lagged difference terms included to eliminate serial correlation—the lag order θ is automatically selected according to the series' length and its autocorrelation characteristics; and ε_t is the error term.

The null hypothesis is given as H_0; the time series has a unit root (i.e., it is non-stationary), and the alternative hypothesis is that the series is stationary.

For each column time series in U_i (i.e., each V-S distance series), the ADF test statistic is calculated and compared with the critical values at the predefined significance level. If the test statistic is smaller than the corresponding critical value, the null hypothesis is rejected and the series is regarded as stationary. Otherwise, the series is considered non-stationary. When the series is stationary, it means that the relative position between the vehicle and the structure does not change significantly in the time window T, indicating that the vehicle is in a parking status.

For vehicle i, the stationarity of all the time series to N_s structures is tested and the parking support ratio (PSR) is defined for the final parking status judgment, as shown in Equation (21).

$$\psi_i = \frac{1}{N_s} \sum_{n_s=1}^{N_s} I^{(\text{Reject } H_0 \text{ for } n_s\text{−th column series})} \tag{21}$$

where $I(\cdot)$ is an indicator function that equals one when the Augmented Dickey–Fuller (ADF) test rejects H_0 for the n_s-th column series in U_i, and 0 otherwise.

If the ψ_i exceeds a predefined threshold ψ_0, the vehicle is classified as being in a parking state during the time window T.

4. Experimental Results and Analysis

4.1. Experimental Scenario and Datasets

The parking experiment is carried out at the Intelligent and Connected Vehicle Test Base of Shandong Expressway, which is characterized by the most diverse scenarios and the most authentic testing environment in China, as shown in Figure 8. The experimental expressway features diverse road scenarios, including long straight sections, continuous curves, ramps, and bridges. In the experiment, a DJI Mavic 3T UAV is used to collect road traffic videos. The detailed environmental conditions, UAV hardware configurations, and flight parameters are elaborated in Table 3.

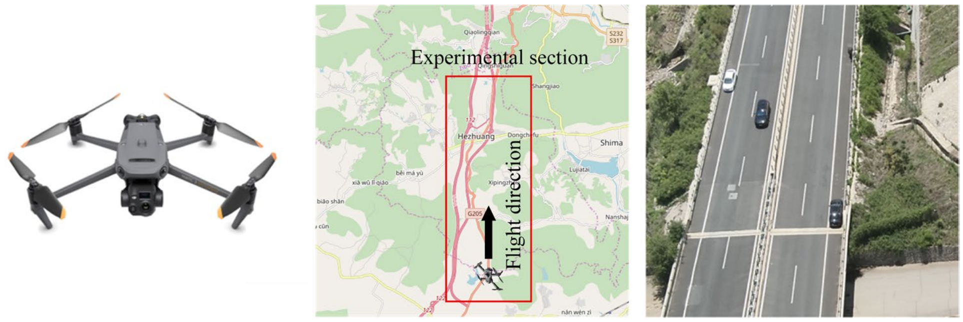


Figure 8. UAV and experimental scenario.

Table 3. Experimental parameters.

Category	Parameter	Value
Environmental conditions	Wind scale	1–4 Bft
	Temperature	10–25 °C
	Visibility	4–10 km
	Weather	Clear/Partly cloudy, Daylight
UAV hardware configurations	Platform	DJI Mavic 3T (wide camera)
	Camera still resolution	Maximum 48 MP
	Video resolution/Fps	Max 3840 × 2160 p, 30 Fps
	Positioning	RTK ± 0.1 m
	Gimbal pitch	45° (oblique + nadir)
	Keyframe interval	1 Fps/10 s
Flight conditions	Altitude (AGL)	80–100 m
	Speed	12–15 m/s

Regarding data annotation, the LabelImg tool is employed to execute the manual annotation of vehicles and structures. Referring to the labeling framework proposed in Section 2.1, vehicles and structures are manually annotated and 30,000 labeled samples are generated. All the samples are divided into a training set, a validation set, and a test set with a ratio of 7:2:1. The proportions of vehicle and structure targets in each subset are shown in Table 4. The detailed label proportions of the targets are shown in Figure 9.

Table 4. Dataset target distribution.

Dataset	Train Set	Validation Set	Test Set
Vehicle	52.37%	53.60%	47.56%
Structure	47.63%	46.40%	52.44%

YOLOv5 and DeepSort are trained using the samples in the training set. As shown in Table 5, the detection results on the validation set show that the detector achieves an average precision of 0.79, a recall of 0.85, and an mAP@0.5 of 0.84. Part of the target detection and tracking results are shown in Figure 10.

Table 5. Detection performance.

Class	Precision	Recall	mAP@0.5
Passenger car	0.95	0.97	0.98
Medium bus	0.86	0.93	0.87
Large bus	0.83	0.91	0.85
Light truck	0.76	0.95	0.79

Table 5. *Cont.*

Class	Precision	Recall	mAP@0.5
Medium truck	0.96	0.73	0.76
Heavy truck	0.94	0.98	0.97
Gantry	0.93	0.69	0.72
Channelization zone	0.73	0.87	0.88
Signboard	0.94	0.84	0.90
Guidance arrow	0.85	0.88	0.92
Speed bump	0.79	0.73	0.72
Speed limit sign	0.82	0.75	0.76
Mean	0.79	0.85	0.84

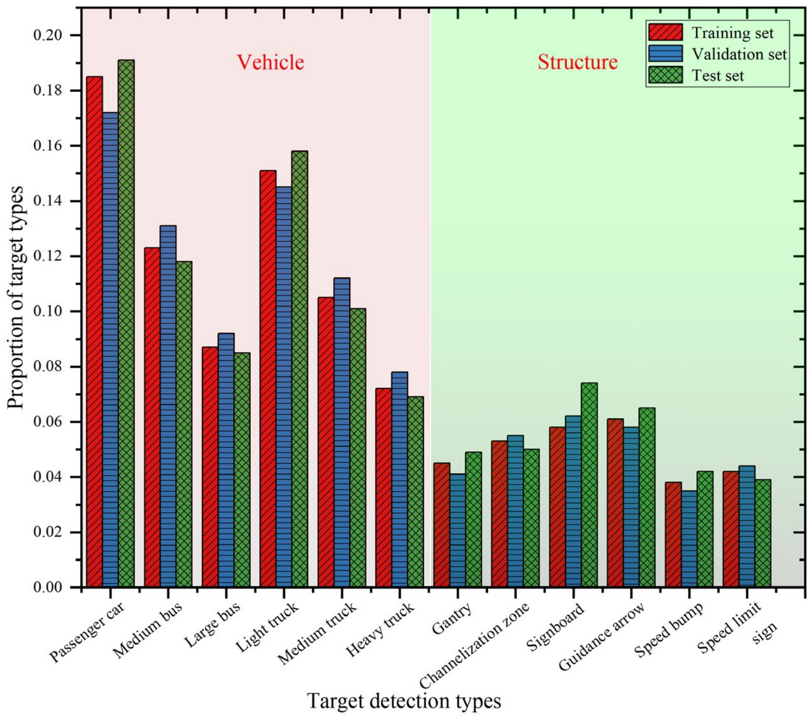


Figure 9. Detailed label proportions of targets.

4.2. Parking Detection Results and Analysis

To quantitatively evaluate the performance of the proposed method, four standard metrics are adopted: accuracy, precision, recall, and F1-score. The metrics are defined as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{22}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{23}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{24}$$

$$\text{F1} - \text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recal}} \tag{25}$$

In Equations (22)–(25), TP represents the number of correctly detected parked vehicles, FP denotes moving vehicles incorrectly identified as parked, TN refers to moving vehicles correctly identified as non-parking, and FN indicates parked vehicles missed.

Figure 10. Part of the target detection and tracking results. The non-English characters painted on the road surface in some image panels (e.g., '减速', '学校') are Chinese, which translate to 'Slow down' and 'School' in English.

(1) Analysis of the rolling detection time window T and PSR threshold ψ_0

The detection time window T is set to vary from 2 s to 6 s in steps of 1 s, and the parking support ratio (PSR) threshold ψ_0 is simultaneously varied from 0.1 to 1.0 in steps of 0.1. The accuracy, precision, recall, and F1-score of the parking detection results are presented in Table 6.

Table 6. Parking detection performance under different time windows T and PSR threshold ψ_0.

Time Window	Metrics	PSR Threshold 0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
$T = 2$ s	Accuracy	0.49	0.51	0.53	0.58	0.67	0.73	0.75	0.74	0.73	0.73
	Precision	0.43	0.48	0.49	0.50	0.58	0.63	0.69	0.63	0.69	0.71
	Recall	0.76	0.78	0.78	0.79	0.83	0.82	0.85	0.83	0.80	0.75
	F1-score	0.55	0.59	0.60	0.61	0.68	0.71	0.76	0.72	0.74	0.73
$T = 3$ s	Accuracy	0.53	0.56	0.59	0.65	0.73	0.79	0.89	0.89	0.88	0.87
	Precision	0.45	0.50	0.53	0.57	0.64	0.70	0.78	0.76	0.80	0.88
	Recall	0.82	0.82	0.82	0.82	0.85	0.84	0.87	0.85	0.80	0.76
	F1-score	0.58	0.62	0.64	0.67	0.73	0.76	0.82	0.80	0.80	0.82
$T = 4$ s	Accuracy	0.63	0.64	0.68	0.78	0.82	0.83	0.93	0.93	0.91	0.90
	Precision	0.48	0.53	0.59	0.65	0.71	0.78	0.82	0.79	0.83	0.93
	Recall	0.96	0.94	0.94	0.90	0.90	0.89	0.88	0.87	0.77	0.75
	F1-score	0.64	0.68	0.72	0.75	0.79	0.83	0.85	0.83	0.80	0.83

https://doi.org/10.3390/vehicles8030049

Table 6. *Cont.*

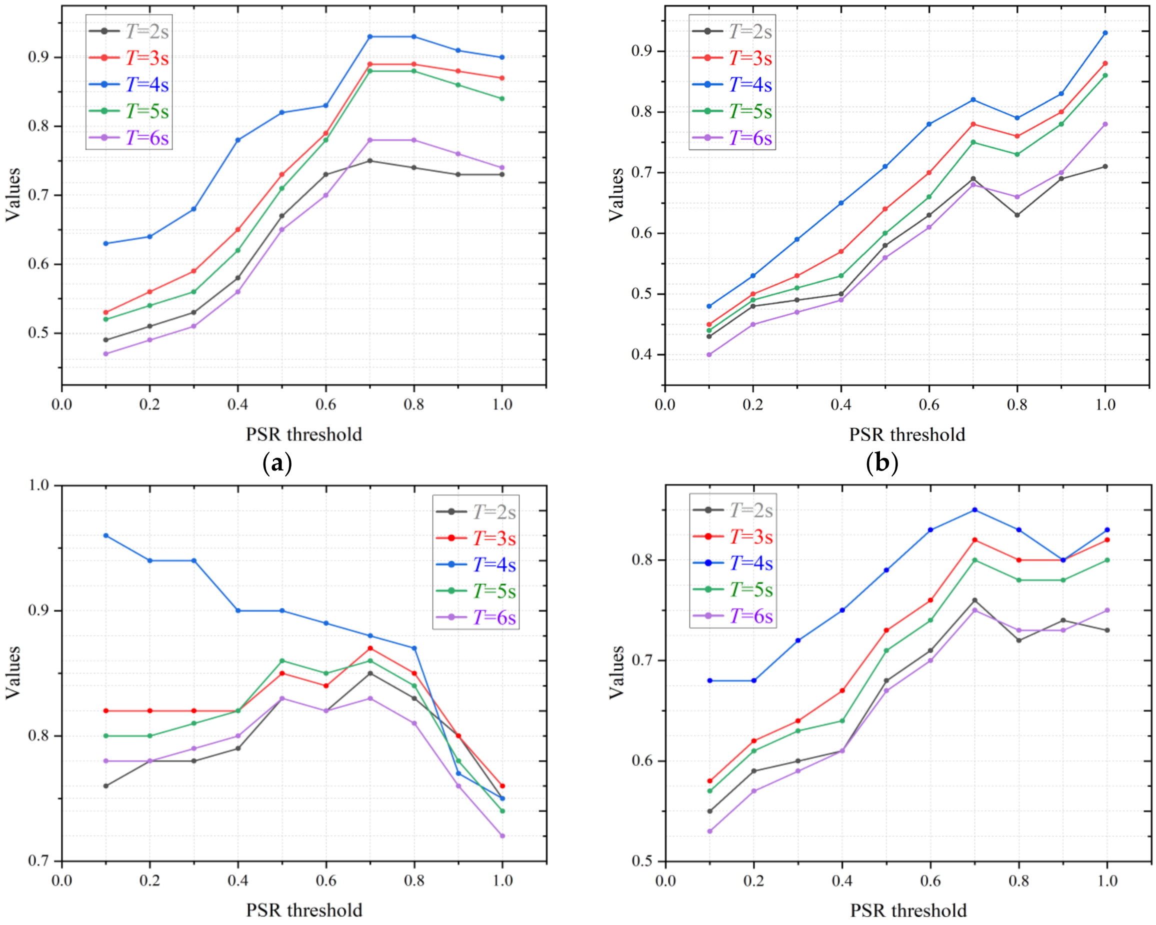
Time Window	Metrics	PSR Threshold									
		0.1	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1.0**
$T = 5$ s	Accuracy	0.52	0.54	0.56	0.62	0.71	0.78	0.88	0.88	0.86	0.84
	Precision	0.44	0.49	0.51	0.53	0.60	0.66	0.75	0.73	0.78	0.86
	Recall	0.80	0.80	0.81	0.82	0.86	0.85	0.86	0.84	0.78	0.74
	F1-score	0.57	0.61	0.63	0.64	0.71	0.74	0.80	0.78	0.78	0.80
$T = 6$ s	Accuracy	0.47	0.49	0.51	0.56	0.65	0.70	0.78	0.78	0.76	0.74
	Precision	0.40	0.45	0.47	0.49	0.56	0.61	0.68	0.66	0.70	0.78
	Recall	0.78	0.78	0.79	0.80	0.83	0.82	0.83	0.81	0.76	0.72
	F1-score	0.53	0.57	0.59	0.61	0.67	0.70	0.75	0.73	0.73	0.75

As shown in Table 6 and Figure 11, both the parking support ratio (PSR) threshold ψ_0 and the time window T have a significant impact on the parking detection performance. Under a fixed time window T, with the increase in the parking support ratio (PSR) threshold from 0.1 to 0.7, the accuracy, precision, and F1-score significantly improve, while the recall decreases slightly. In this paper, the parking support ratio (PSR) denotes the proportion of structural sequences that reject H_0 for the n_s-th column series in the Augmented Dickey–Fuller (ADF) test. When the parking support ratio (PSR) threshold is small, as long as the time series of a small number of structures exhibit stationarity, the vehicle will be classified as parked. In this case, the parking events are almost detected (high recall), while part of the moving vehicles are misclassified as parked due to the lack of enough structure references, leading to lower precision and F1-score values. When the parking support ratio (PSR) threshold ψ_0 is larger than 0.7, the performance degrades since a higher parking support ratio (PSR) threshold makes the decision rule much stricter. When there are fewer structures, even parked vehicles may fail to satisfy such a strict parking support ratio (PSR) threshold condition and be misclassified as non-parking, leading to a reduction in recall and F1-score.

When the parking support ratio (PSR) threshold is fixed and the time windows T takes small values, each window contains only a limited number of samples, and the parking detection is more sensitive to transient motion or jitter, leading to unstable decisions and degraded accuracy. Conversely, when the time window T is set too large, part of targets fail to enter a valid detection window when tracking drift occurs. Hence, some parking vehicles are misclassified as non-parking, leading to missed detections. An appropriate time window T can provide enough samples for a reliable Augmented Dickey–Fuller (ADF) test and simultaneously reduce the probability that the vehicle target disappears caused by tracking drift. Referring to Table 6 and Figure 11, when $T = 4$ s and $\psi_0 = 0.7$, the proposed method achieves the best performance.

(2) Ablation analysis of V-S matrix correction

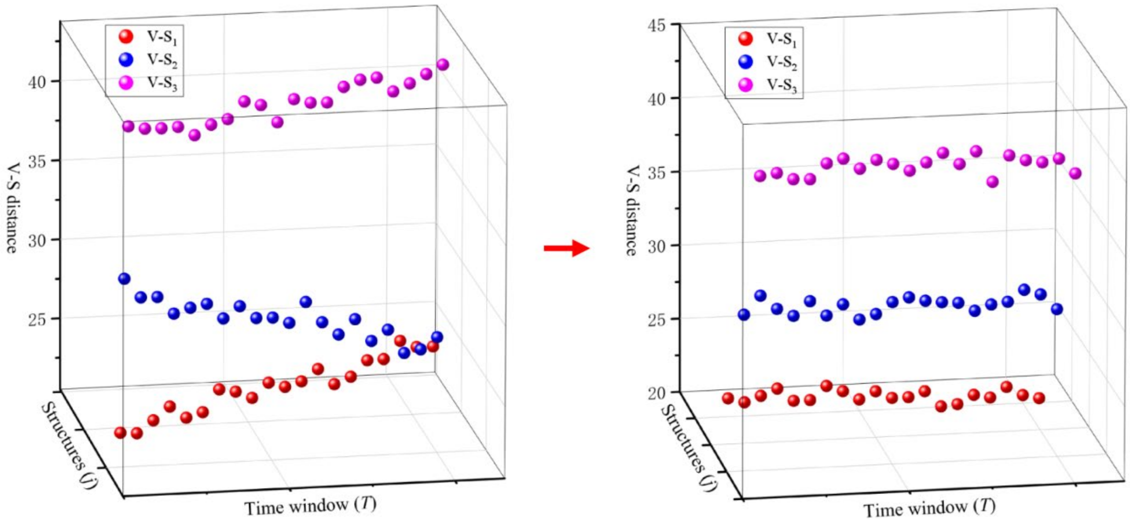
To provide an intuitive illustration of the correction effect of V-S matrix by the SF, the V-S distance tensor for a parking vehicle with three static structures is shown in Figure 12. In Figure 12a, the distance values for each vehicle in the original V-S matrix exhibit an obvious structure-dependent trend and scale discrepancy. After the correction, the distance values become much more uniform and approximately horizontal in Figure 12b, making the description of the relative position between the vehicle and the structure more objective and authentic.



(a) (b)

(c) (d)

Figure 11. Effect of PSR threshold ψ_0 and time window T on parking detection metrics: (**a**) accuracy; (**b**) precision; (**c**) recall; and (**d**) F1-score.



(**a**) (**b**)

Figure 12. V-S matrix correction effect by SF: (**a**) original V-S matrix; (**b**) corrected V-S matrix.

To analyze the parking detection performance using corrected V-S matrix by SF compared to the original matrix, experiments are designed and conducted for four cases, as follows:

Case A: the original V-S matrix is used and the nearest structure to the vehicle is applied for the only reference.

Case B: the original V-S matrix is used and multi-structures around the vehicle are applied for references.

Case C: the corrected V-S matrix is used and the nearest structure to the vehicle is applied for the only reference.

Case D: the corrected V-S matrix is used and multi-structures around the vehicle are applied for references.

Referring to the experimental results presented in Table 6, the detection time window T is set to 4 s and the parking support ratio (PSR) threshold ψ_0 is set to 0.7. The accuracy, precision, recall, and F1-score of the parking detection results under different cases are presented in Table 7.

Table 7. Parking detection performance under different V-S matrix and referenced structure cases.

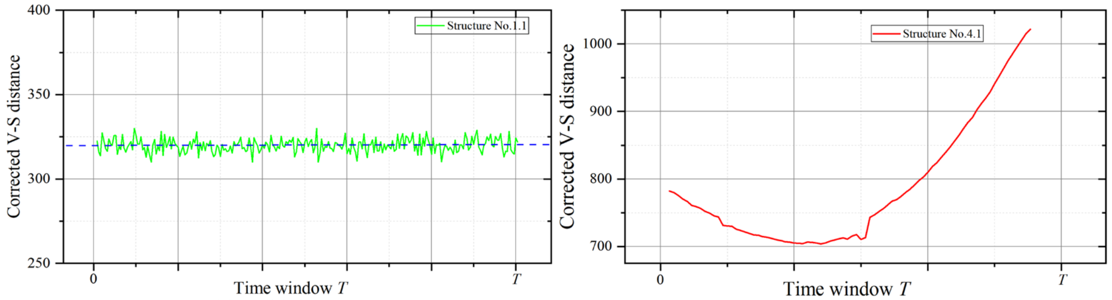
Case	Accuracy	Precision	Recall	F1-Score
A: Original V-S matrix, nearest structure as only reference	0.78	0.54	0.97	0.69
B: Original V-S matrix, multi-structures as references	0.84	0.62	0.93	0.74
C: Corrected V-S matrix, nearest structure as only reference	0.89	0.68	0.88	0.77
D: Corrected V-S matrix, multi-structures as references	0.93	0.82	0.88	0.85

As shown in Table 7, when the nearest structure is used as the only reference (Case A and Case C), the parking detection performance is the worst compared to the cases using multi-structures as references (Case B and Case D). By adopting the parking support ratio (PSR) to aggregate the detection results over multiple distance–time series between the vehicle and all structures, false positives caused by noise in any single distance–time series can be effectively suppressed. Hence, the accuracy, precision, and F1-score can be significantly improved while the recall is slightly reduced.

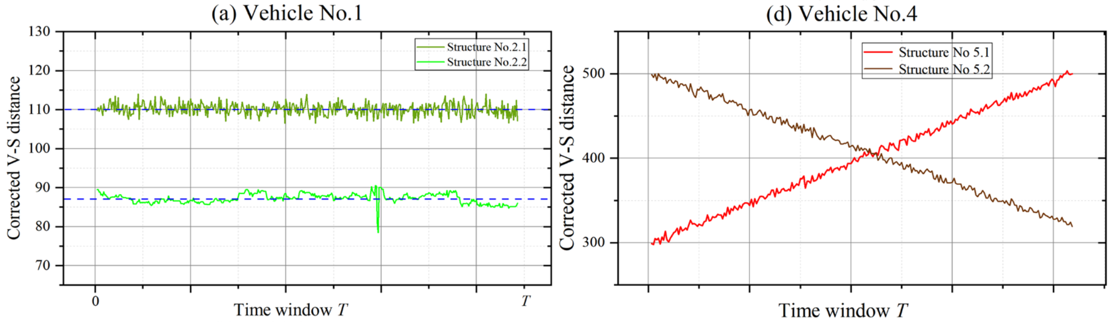
After the correction of the V-S matrix by the SF, perspective distortion and radial scale inconsistency under the UAV view are effectively compensated, making the distance–time series between the vehicles and the structures more discriminative. Under the same number of referenced structures, the accuracy, precision and F1-score values of Case C and Case D are superior to those of Case A and Case B, respectively. In particular for Case D, the integration of the corrected V-S matrix and multi-structures as references reduces both false positives and missed detections, and therefore achieves the most balanced performance in parking detection.

(3) Parking detection results and analysis

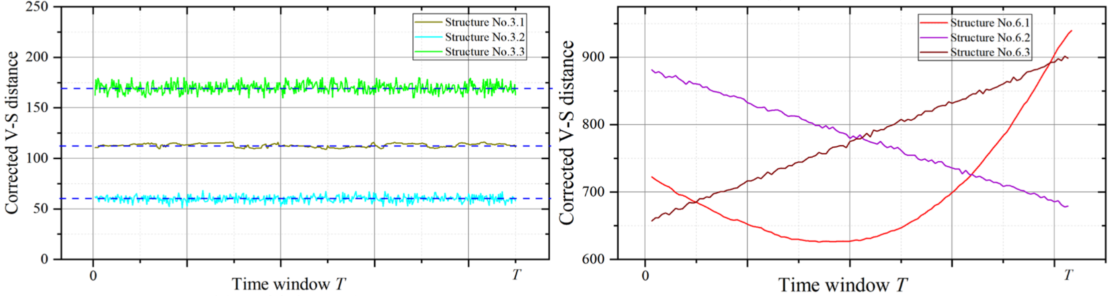
Taking $T = 4$ s and $\psi_0 = 0.7$, the performance of the parking detection based on time series analysis is evaluated. Part of the time series of the V-S distance values for both parking and non-parking vehicles are shown in Figure 13. For parking vehicles, it can be observed that the corrected V-S distance series fluctuate around a nearly constant level. The ADF test with a constant term and without a deterministic trend term is carried out referring to Equation (20). The detailed ADF test results are presented in Table 8. It shows that the t-statistics are strongly negative and p-values are close to zero, indicating that the unit-root null hypothesis is rejected and the series are regarded as stationary. In contrast, the non-parking samples exhibit obvious increasing or decreasing trends, and the ADF tests fail to reject the null hypothesis, indicating their non-stationarity characteristics. These results verify that the proposed method can effectively distinguish between parking and non-parking behaviors.



(a) Vehicle No.1 (d) Vehicle No.4



(b) Vehicle No.2 (e) Vehicle No.5



(c) Vehicle No.3 (f) Vehicle No.6

Figure 13. V-S distance series and ADF test results.

Table 8. ADF test results of V-S distance–time series for typical parking and non-parking cases.

Case	Vehicle No.	Structure No.	Lag *p*	*t*-Stat	*p*-Value	Stationary
Parking	1	1.1	0	−15.3788	0.0000	stationary (reject H_0)
	2	2.1	0	−21.3487	0.0000	
	2	2.2	2	−15.1082	0.0000	
	3	3.1	0	−4.1629	0.0008	
	3	3.2	0	−22.4153	0.0000	
	3	3.3	1	−21.5282	0.0000	
Non-parking	4	4.1	1	2.9773	1.0000	non-stationary (accept H_0)
	5	5.1	5	−0.0656	0.9505	
	5	5.2	10	−0.2451	0.9293	
	6	6.1	3	−0.9794	0.7588	
	6	6.2	1	0.0866	0.9635	
	6	6.3	6	0.1836	0.9707	

To evaluate the performance of the proposed algorithm, a parking detection method based on a speed threshold [30] is selected for comparison. By calculating vehicles' instantaneous speeds based on multi-object tracking and spatial coordinate transformation [31], the parking state is determined when the vehicle's speed remains below a predefined threshold for a specified number of consecutive frames. Specifically, the speed threshold is set as 5.38 km/h. The duration threshold of the optimal time window is set as 4 s. Both methods are evaluated on the test set presented in Section 4.1, and the detailed results are shown in Table 9.

Table 9. Performance of different parking detection methods.

Method	Accuracy	Precision	Recall	F1-Score
Speed threshold	0.74	0.72	0.96	0.83
The proposed method	0.93	0.82	0.88	0.85

As shown in Table 9, the proposed method in this paper achieves better performance compared with the speed threshold method. In the speed threshold method, the parking detection performance relies on the accuracy of vehicles' continuous speed variation in the world coordinate system. However, influenced by localization noise, calibration errors during the coordinate transformation process, UAV pose jitter, and frame-rate fluctuations, the errors can be amplified when estimating vehicles' real-time speeds, producing false positives and consequently degrading accuracy and precision.

The proposed method presents better performance in terms of accuracy (0.93), precision (0.82), and F1-score (0.85), while maintaining a relatively high recall (0.88). This is because the parking detection does not directly rely on the vehicles' speed values, but on their relative positions to static structures in the pixel coordinate system. By performing parking analysis directly within the pixel coordinate system, the proposed method bypasses the systematic errors that traditional speed threshold method can often introduce when calculating vehicle speeds in the world coordinate system. As long as the relative position between the vehicle and at least one structure does not exhibit a persistent trend during certain time, the V-S distance series can provide sufficient statistical evidence (via the ADF test and PSR) to support a parking detection, yielding greater accuracy.

5. Conclusions and Future Work

This paper investigates a vehicle parking detection method using UAV cruise-perspective images for expressway scenarios. By constructing a V-S distance matrix, the method describes the relative positional relationship between vehicles and structures, applies SF-based correction to these distances, and then determines parking states using time-series analysis. In the pixel coordinate system, a PSR-based multi-structure joint decision strategy is further introduced to achieve accurate parking detection. Experimental results demonstrate that the proposed approach attains a more balanced performance in terms of accuracy, precision, recall, and F1-score compared with the traditional speed threshold method.

Future work will focus on conducting systematic experiments in more complex road expressway scenarios (e.g., nighttime and rainy conditions) to further evaluate the applicability of the proposed method, and more refined behavioral modeling strategies will be explored to enhance the parking detection performance in short stops and frequent stop-and-go behaviors.

Author Contributions: Conceptualization, X.G. and H.L.; methodology, H.L.; software, X.G.; validation, X.G., Y.W. (Yuehao Wang), Y.W. (Yaxin Wei) and G.S.; formal analysis, X.G.; investigation, Y.W. (Yuehao Wang); resources, X.G.; data curation, G.S.; writing—original draft preparation, X.G.; writing—review and editing, X.G.; visualization, X.G.; supervision, H.L.; project administration, H.L.; funding acquisition, H.L. All authors have read and agreed to the published version of the manuscript.

Funding: The research is partially supported by Jinan City's Self-Developed Innovative Team Project for Higher Educational Institutions (# 20233040) and Shandong Province Overseas High-Level Talent Workstation Project.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zhu, H.; Feng, S.; Yu, F. Parking detection method based on finite-state machine and collaborative decision-making. *IEEE Sens. J.* **2018**, *18*, 9829–9839. [CrossRef]
2. Zhang, Z.; He, X.; Huang, J.; Yuan, H. Parking detection using combined magnetic sensor and pulsed coherent radar. *IEEE Internet Things J.* **2022**, *9*, 17210–17219. [CrossRef]
3. Ren, C.; Lee, S.; Jeong, D.; Chen, H.; Xiao, Y. Parking guidance system based on geomagnetic sensors and recurrent neural networks. *J. Sens.* **2022**, *2022*, 7481064. [CrossRef]
4. Gong, J.; Raut, A.; Pelzer, M.; Huening, F. Marking-based perpendicular parking slot detection algorithm using LiDAR sensors. *Vehicles* **2024**, *6*, 1717–1729. [CrossRef]
5. Im, G.; Kim, M.; Park, J. Parking line based SLAM approach using AVM/LiDAR sensor fusion for rapid and accurate loop closing and parking space detection. *Sensors* **2019**, *19*, 4811. [CrossRef]
6. Sayeed, M.S.; Abdulrahim, H.; Razak, S.F.A.; Bukar, U.A.; Yogarayan, S. IoT Raspberry Pi based smart parking system with weighted K-nearest neighbours approach. *Civ. Eng. J.* **2023**, *9*, 1991–2011. [CrossRef]
7. Pratomo, A.H.; Kaswidjanti, W.; Nugroho, A.S.; Saifullah, S. Parking detection system using background subtraction and HSV color segmentation. *Bull. Electr. Eng. Inform.* **2021**, *10*, 3211–3219. [CrossRef]
8. Shaaban, K.; Tounsi, H. Parking detection method using quadtree decomposition analysis. *J. Traffic Transp. Eng. (Engl. Ed.)* **2022**, *9*, 645–653. [CrossRef]
9. Yamada, K.; Mizuno, M. A vehicle parking detection method using image segmentation. *Electron. Commun. Jpn. (Part III Fundam. Electron. Sci.)* **2001**, *84*, 25–34. [CrossRef]
10. Mago, N.; Kumar, S.; Goyal, L.M. Real-time fuzzy-based intelligent parking detection system using deep learning techniques. *Int. J. Fuzzy Syst.* **2022**, *24*, 2560–2568. [CrossRef]
11. Sayar, A.; Mustacoglu, A.F. Street-based parking lot detection with image processing and deep learning. *Signal Image Video Process.* **2024**, *18*, 945–952. [CrossRef]
12. Jo, K.H. Cumulative dual foreground differences for illegally parked vehicles detection. *IEEE Trans. Ind. Inform.* **2017**, *13*, 2464–2473.
13. Park, J.; Lee, J.; Park, Y.; Lim, Y. Deep learning-based stopped vehicle detection method utilizing in-vehicle dashcams. *Electronics* **2024**, *13*, 4097. [CrossRef]
14. Kabashkin, I.; Kulmurzina, A.; Nadimov, B.; Tlepiyeva, G.; Sansyzbayeva, Z.; Sultanov, T. Synchronized multi-point UAV-based traffic monitoring for urban infrastructure decision support. *Drones* **2025**, *9*, 370. [CrossRef]
15. Wu, H.; Niu, M.; Wang, B.; Yan, K.; Li, Y.; Pang, H. Fixed/mobile collaborative traffic flow detection study based on wireless charging of UAVs. *Drones* **2025**, *9*, 117. [CrossRef]
16. Zheng, B.; Zhou, J.; Hong, Z.; Tang, J.; Huang, X. Vehicle recognition and driving information detection with UAV video based on improved YOLOv5-DeepSORT algorithm. *Sensors* **2025**, *25*, 2788. [CrossRef] [PubMed]
17. Zhu, Y.; Wang, Y.; An, Y.; Yang, H.; Pan, Y. Real-time vehicle detection and urban traffic behavior analysis based on UAV traffic videos on mobile devices. *arXiv* **2024**, arXiv:2402.16246. [CrossRef]
18. Abughalieh, K.M.; Sababha, B.H.; Rawashdeh, N.A. A video-based object detection and tracking system for weight sensitive UAVs. *Multimed. Tools Appl.* **2019**, *78*, 9149–9167. [CrossRef]
19. Wang, L.; Chen, F.; Yin, H. Detecting and tracking vehicles in traffic by unmanned aerial vehicles. *Autom. Constr.* **2016**, *72*, 294–308. [CrossRef]
20. Walambe, R.; Marathe, A.; Kotecha, K. Multiscale object detection from drone imagery using ensemble transfer learning. *Drones* **2021**, *5*, 66. [CrossRef]
21. Yan, H.; Kong, X.; Wang, J.; Tomiyama, H. ST-YOLO: An enhanced detector of small objects in unmanned aerial vehicle imagery. *Drones* **2025**, *9*, 338. [CrossRef]
22. Kim, S.; Tak, Y.; Barmpounakis, E.; Geroliminis, N. Monitoring outdoor parking in urban areas with unmanned aerial vehicles. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 13393–13406. [CrossRef]
23. Chriki, A.; Touati, H.; Snoussi, H.; Kamoun, F. Deep learning and handcrafted features for one-class anomaly detection in UAV video. *Multimed. Tools Appl.* **2021**, *80*, 2599–2620. [CrossRef]

24. Jausevac, G.; Dobrilovic, D.; Brtka, V.; Jotanovic, G.; Perakovic, D.; Stojanov, Z. Multirole UAVs supported parking surveillance system. *Mob. Netw. Appl.* **2023**, *28*, 1246–1254. [CrossRef]
25. Wang, H.; Liu, Y.; Huang, H.; Pan, Y.; Yu, W.; Jiang, J.; Lyu, D.; Bocus, M.J.; Liu, M.; Pitas, I.; et al. ATG-PVD: Ticketing parking violations on a drone. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; pp. 541–557.
26. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*; Cambridge University Press: Cambridge, UK, 2003.
27. Bertozz, M.; Broggi, A.; Fascioli, A. Stereo inverse perspective mapping: Theory and applications. *Image Vis. Comput.* **1998**, *16*, 585–590. [CrossRef]
28. Cáceres Hernández, D.; Kurnianggoro, L.; Filonenko, A.; Jo, K.H. Real-time lane region detection using a combination of geometrical and image features. *Sensors* **2016**, *16*, 1935. [CrossRef]
29. Al Mamun, A.; Em, P.P.; Hossen, M.J.; Jahan, B.; Tahabilder, A. A deep learning approach for lane marking detection applying encode–decode instant segmentation network. *Heliyon* **2023**, *9*, e14212. [CrossRef]
30. Ding, B.; Yang, Z.; Ding, J.; Liu, J.; Yan, G. Expressway Tunnel Parking Detection Method Based on Improved YOLOv3. *J. Comput. Eng. Appl.* **2021**, *57*, 234–239.
31. Ali, A.; Hassan, A.; Ali, A.R.; Khan, H.U.; Kazmi, W.; Zaheer, A. Real-time vehicle distance estimation using single view geometry. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Snowmass Village, CO, USA, 1–5 March 2020; pp. 1100–1109.