Check for updates

Vehicle recognition pipeline via DeepSort on aerial image datasets

Muhammad Hanzla[1], Muhammad Ovais Yusuf[1], Naif Al Mudawi[2], Touseef Sadiq[3]*, Nouf Abdullah Almujally[4], Hameedur Rahman[1]*, Abdulwahab Alazeb[2] and Asaad Algarni[5]

[1]Faculty of Computing and AI, Air University, Islamabad, Pakistan, [2]Department of Computer Science, College of Computer Science and Information System, Najran University, Najran, Saudi Arabia, [3]Department of Information and Communication Technology, Centre for Artificial Intelligence Research, University of Agder, Grimstad, Norway, [4]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia, [5]Department of Computer Sciences, Faculty of Computing and Information Technology, Northern Border University, Rafha, Saudi Arabia

Introduction: Unmanned aerial vehicles (UAVs) are widely used in various computer vision applications, especially in intelligent traffic monitoring, as they are agile and simplify operations while boosting efficiency. However, automating these procedures is still a significant challenge due to the difficulty of extracting foreground (vehicle) information from complex traffic scenes.

Methods: This paper presents a unique method for autonomous vehicle surveillance that uses FCM to segment aerial images. YOLOv8, which is known for its ability to detect tiny objects, is then used to detect vehicles. Additionally, a system that utilizes ORB features is employed to support vehicle recognition, assignment, and recovery across picture frames. Vehicle tracking is accomplished using DeepSORT, which elegantly combines Kalman filtering with deep learning to achieve precise results.

Results: Our proposed model demonstrates remarkable performance in vehicle identification and tracking with precision of 0.86 and 0.84 on the VEDAI and SRTID datasets, respectively, for vehicle detection.

Discussion: For vehicle tracking, the model achieves accuracies of 0.89 and 0.85 on the VEDAI and SRTID datasets, respectively.

KEYWORDS

deep learning, remote sensing, object recognition, unmanned aerial vehicles, DeepSort, dynamic environments, path planning

1 Introduction

Rapid economic and demographic expansion generate a dramatic surge in vehicle numbers on highways. Hence, complete road traffic monitoring is necessary for acquiring and evaluating data, essential for comprehending highway operations within an intelligent, autonomous transportation framework (Dikbayir and İbrahim Bülbül, 2020; Xu et al., 2022; Yin et al., 2022). Consequently, there's a compelling need to automate traffic monitoring systems. While various image-based solutions have been developed, obstacles exist in expanding their capabilities, especially in dynamic contexts where backdrop and objects are in flux (Weng et al., 2006; Di et al., 2023; Dai et al., 2024). Traditional approaches like background removal and frame differencing struggle when used to photographs acquired from mobile platforms owing to background motion, blurring the boundaries between background and foreground objects. Hence, improvements in computer vision and image processing, covering disciplines such as intelligent transportation, medical imaging, object

identification, semantic segmentation, and human-computer interaction, present promising paths (Angel et al., 2003; Cao et al., 2021; Ding et al., 2024).

Semantic segmentation, defining and identifying pixels by class, provides a sophisticated method (Schreuder et al., 2003; Sun et al., 2020; Ren et al., 2024). Unlike current systems confined to binary segmentation (e.g., vehicle vs. backdrop), our suggested technique utilizes multi-class segmentation, expanding scene knowledge (Ding et al., 2021; Gu et al., 2024). Moreover, utilizing aerial data offers enormous promise in boosting traffic management. However, obstacles such as varying item sizes, wide non-road regions, and different road layouts need efficient solutions to exploit mobile platform-derived data effectively (Najiya and Archana, 2018; Sun et al., 2018; Omar et al., 2021).

In this study, a unique approach for the identification and tracking of vehicles is based on aerial images. In our approach, aerial films are first transformed into frames for images (Sun et al., 2023). Defogging and gamma correction methods are then used for noise reduction and bright-ness improvement, respectively, while pre-processing is being done on those frames (Qu et al., 2022; Chen et al., 2023a; Zhao X. et al., 2024). After that, Fuzzy C Mean and DBSCAN algorithm is used for segmentation to decrease the background complexity. YOLOv8 is applied for recognition of automobiles in each extracted frame as it can detect tiny objects successfully. To track several cars inside the image's frames, all identified vehicles have been allocated an ID based on ORB attributes. Also, to estimate the traffic density on the roadways, a vehicle count has been kept throughout the picture frames. The tracking has been done using the DeepSort with Kalman filter. Moreover, the provided traffic monitoring systems were verified by the tests done on VEDAI, and SRTID datasets. The studies have exhibited amazing detection and tracking precision when compared to other state-of-the-art (SOTA) approaches.

Some of the prominent contributions of this work include:

- Our model reduces model complexity by combining pre-processing methodologies with segmentation techniques for the preparation of frames prior to detection phase.
- Evaluation of unsupervised segmentation strategies, specifically Fuzzy C-Mean (FCM) algorithm and Density-Based Spatial Clustering of Applications with Noise (DBSCAN), was undertaken, boosting segmentation effectiveness.
- Significantly enhanced accuracy, recall, and F1 Score in vehicle recognition and tracking compared to earlier techniques have been obtained.

Implementation of vehicle tracking leveraging the DeepSort algorithm, reinforced by an ID assignment and recovery module based on ORB, has been successfully accomplished, exhibiting remarkable performance proven across two publicly available datasets.

The article is structured into the following sections: Section 2 dives into the literature on traffic analysis. Section 3 goes over the proposed technique in great depth. Section 4 describes the experimental setting, offering empirical insights into the system's performance. Section 5 reviews the system's performance and considers its advantages and disadvantages. Section 6. Discuss the work's limitations. Section 7 is the conclusion, which summarizes the main results and proposes future research and development goals.

2 Literature review

Over the last several years, academics have aggressively excavated into constructing traffic monitoring systems. They have examined the behaviors of their systems utilizing multiple picture sources, including static camera feeds, satellite images, and aerial data (Li J. et al., 2023; Wu et al., 2023). Typically, the full photos undergo first preprocessing to exclude non-essential components beyond cars, followed by feature extraction (Hou et al., 2023a). Different strategies depend on techniques such as image differencing, foreground extraction, or background removal, especially when the Region of Interest (ROI) is well-defined and suitably sized within the images (Shi et al., 2023; Zhu et al., 2024). Aerial imaging can cause the size of vehicles to vary based on the height of image acquisition. Because of this, semantic segmentation techniques have become popular for detection and tracking applications. It is also common to use additional stages such as clustering and identifier assignment to improve results. Deep learning algorithms have become popular in recent years for object recognition, showing better performance in handling complex situations (Wang et al., 2024; Yang et al., 2024). To provide an overview of current models and approaches, the linked research is classified into machine-learning and deep learning-based traffic system analyses.

2.1 Machine learning-based traffic scene analysis

Machine learning has been extensively used in computer vision-related jobs for a long time, particularly in traffic control and monitoring. To find the cars in the images (Rafique et al., 2023), introduced a vehicle recognition model based on haar-like characteristics with an AdaBoost classifier. In Drouyer and de Franchis (2019), a method for monitoring traffic on highways using medium resolution satellite images is shown. The backdrop image difference approach was used to identify the items in motion after a median filter was applied to the images after road masking for the elimination of irrelevant regions. Next, the gray level of the resultant image was computed. The last phase used a thresholding strategy to identify large, bright spots as autos. According to the authors in Hinz et al. (2006), motion detection algorithms are in-effective because aerial images include motion in both the foreground and background. Therefore, an approach based on morphological operations, the Otsu partitioning method, and bottom-hat and top-hat transformations was applied for detection. After extracting the Shi Tomasi features, clusters were formed based on displacement and angle trajectories. The automobiles vanished behind the backdrop clusters. Each vehicle's robust feature vector was used for tracking. To achieve excellent precision, they used several feature maps. Vehicle detection has been accomplished utilizing two distinct methods in separate research (Chen and Meng, 2016). While the other approach employed HSV color characteristics in conjunction with the Gray Level Co-occurrence Matrix (GLCM) to identify cars, the first methodology used features from the Accelerated Segment Test (FAST) and HOG features. Vehicle tracking is achieved via the use of Forward and Backward Tracking (FBT).

The background subtraction approach is used by Aqel et al. (2017) to identify moving automobiles. Morphological adjustments are carried out to reduce the incidence of false positives. Ultimately, the

invariant Charlier moments are used to achieve categorization. The method's applicability to a variety of traffic circumstances is limited by the usage of standard image processing methods. Additionally, the automobiles that are not moving will be removed using the background subtraction approach, which will lower the true positives. Another traffic monitoring strategy has been provided by Mu et al. (2016). The model selected the area with a high Absolute Difference (SAD) value as a moving vehicle after computing the image difference. Vehicles have been found and matched across many picture frames using SIFT. The authors of Teutsch et al. (2017) used a novel technique for stacking images. The image registration process was limited to tiny autos, and the warping approach was used to blur any stationary backdrops close to moving vehicles. The main goal of this algorithm is to remove distracting backdrop features from images so that only the vehicle is visible when the surrounding region is smoothed out. These systems have a high temporal complexity, and these approaches were distinguished by their complicated properties. These methods incur high computational costs. Furthermore, the model's generalizability is weakened as scene complexity rises.

2.2 Deep learning-based traffic scene analysis

Traffic monitoring has always included manual techniques and in-car technology. Nonetheless, deep learning is more effective than traditional methods when it comes to image processing. An automobile recognition method based on the You Look Only Once (YOLOv4) deep learning algorithm has been presented by Lin and Jhang (2022). Another study Bewley et al. (2016) employed the Faster R-CNN as the target detector and developed a tracking method (SORT) for real-time systems based on the Hungarian matching algorithm and the Kalman filter to track several targets at once. The SORT tracker does not take the monitored object's appearance characteristics into account. A technique for detecting automobiles using an enhanced YOLOv3 algorithm is proposed by Zhang and Zhu (2019). To increase the detection method's accuracy, a new structure is added to the pre-trained YOLO network during training. YOLOv3, on the other hand, is among the most ancient. Using the most recent designs may enhance the detection result. Miniature CNN architecture, as described by Ozturk and Cavus (2021), is a vehicle identification model that relies on Convolutional Neural Networks (CNNs) in conjunction with morphological adjustments. The computational cost of this post-processing is high. Moreover, different aerial image databases show different levels of accuracy. A method for real-time object tracking and detection was reported by the authors in Alotaibi et al. (2022). An enhanced RefineDet-based detection module is included in the model. Additionally, the twin support vector machine model and the harmony search technique are employed for classification. Pre-processing of the data is absent from the model, which might lower the model's total computing complexity. A vehicle identification model based on deep learning is shown in Amna et al. (2020). Convolutional Neural Networks (CNN) are used by the model to recognize vehicles, while radar data is used to determine the target's location. A two-stage deep learning model is developed in different research (He and Li, 2019). In addition to detecting cars, the model also recognizes them again in subsequent frames, which is a crucial component of tracking. As opposed to traditional appearance and

motion-based characteristics, the re-identification is mostly reliant on vehicle tracking context information.

There is always room for development in the field of automated traffic monitoring systems, despite the substantial research that has been done in this area. To get effective results, efficient and specialized designs are needed for the recognition of automobiles in aerial images, particularly in situations with heavy traffic. Machine learning techniques are insufficient to distinguish between objects whose pixels exhibit motion. As a result, we use deep learning strategies to raise the precision of vehicle tracking and detection.

3 Materials and methods

3.1 System methodology

This section details the planned traffic monitoring system. System architecture overview is provided in Figure 1. This work offers a vehicle recognition and tracking system based on semantic segmentation. Firstly, the videos are turned into frames and pre-processing processes, i.e., defogging for noise reduction are done to the images. Then Gamma correction is employed for adjustment of image intensity for enhanced detection. FCM and DBSCAN segmentation was done on the filtered images for separation of foreground and back-ground items. YOLOv8 is applied for vehicle detection. ORB attributes are used for the assignment of unique ID. Vehicles were traced over several frames of images using a Deepsort. For finding each tracked vehicle, ORB key point description combined with trajectories approximation are used to recover IDs. Further information on each module is given in the ensuing subsections.

3.2 Images pre-processing

To eliminate superfluous pixel information from the resulting image, noise reduction is necessary since the extra pixel's complicate recognition (Rong et al., 2022; Xiao et al., 2023). For best performance, any filter using defogging methods is applied to noise (Gao et al., 2020; Tang et al., 2024). The defogging technique measures the amount of noise in each pixel of the picture and then removes it in the following ways.

$$G(x) = X(x)Y(x) + Z(1 - p(x))$$

where pixel location is denoted by x, fog density by Z, and transmission map by $Y(x)$. Figure 2 represents defogged images:

The denoised image's intensity is then adjusted using gamma correction (Huang et al., 2018; Zhao L. et al., 2024) since a high brightness allows for the most effective detection of the area of interest. The gamma correction power-law is provided as follows:

$$Vo = TV_I^\gamma$$

where V_I is the non-negative value with power γ of the input, which may vary from 0 to 1, and T is a constant, usually equals to 1. V_o stands for the final image. The plotted denoised, intensity adjusted. Figure 3 shows the gamma-corrected images.
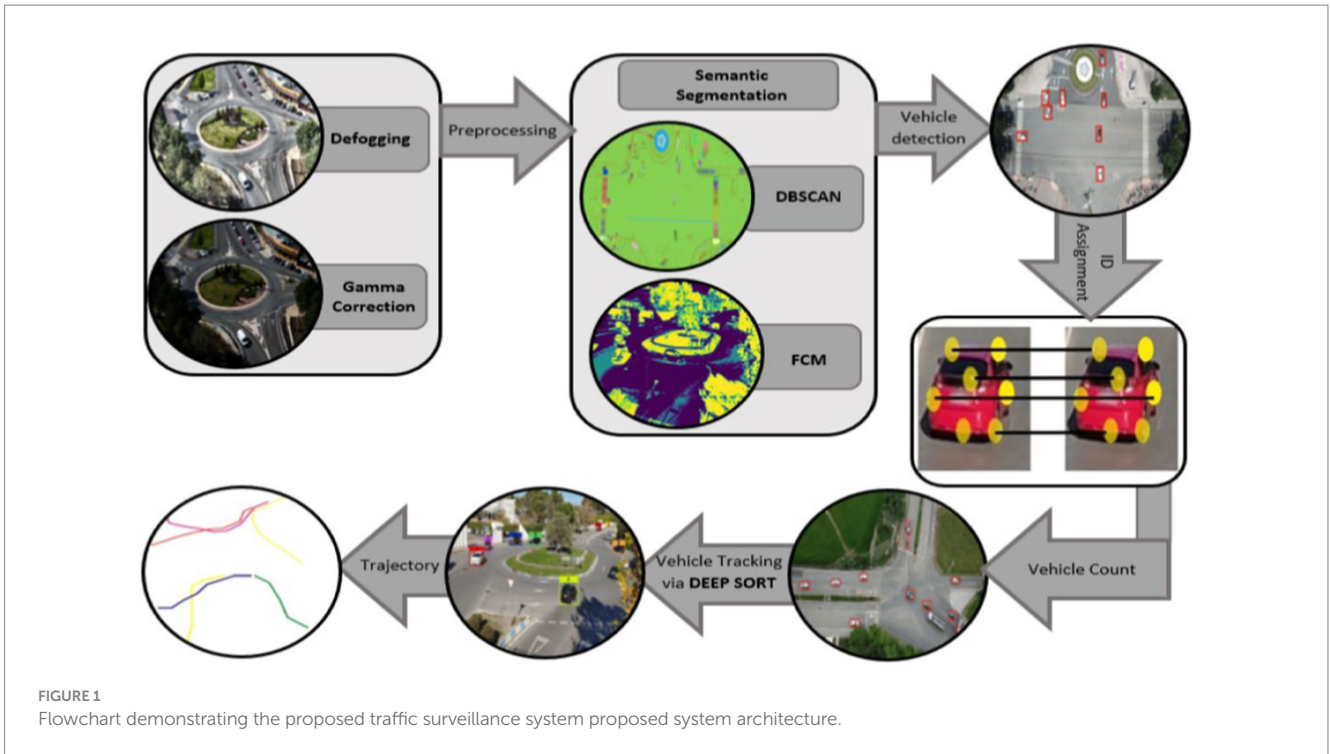
FIGURE 1
Flowchart demonstrating the proposed traffic surveillance system proposed system architecture.

FIGURE 2
Defogging results over the **(A)** original image of VEDAI dataset **(B)** defogged image **(C)** original image of SRTID dataset **(D)** defogged image.

3.3 Semantic segmentation

In many computer vision applications, including autonomous vehicles, medical imaging, virtual reality, and surveillance systems, image segmentation is essential. Images are divided into homogeneous sections using segmentation methods. Every area stands for a class or object. To improve item recognition on complicated backdrops, we compared two segmentation techniques.
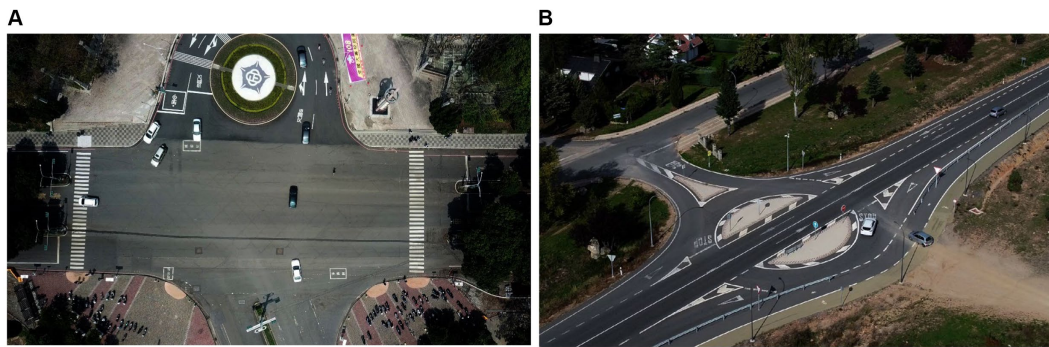
FIGURE 3
Pre-processed image using gamma correction over the **(A)** VEDAI dataset **(B)** SRTID dataset.

3.3.1 FCM segmentation

Segmentation is widely employed in a variety of computer vision applications. This is a fundamental stage. Segmentation methods separate images into homogeneous sections (Huang et al., 2019; Hao et al., 2024). Each area denotes an item or class. We used the Fuzzy C-Mean segmentation technique. FCM is a clustering method in which each picture pixel might belong to two or more groups. Fuzzy logic (Chong et al., 2023; Zheng et al., 2024) refers to pixels that belong to more than one cluster. Because we are working with many complicated road backdrops including several items and circumstances, segmentation approaches based on explicit feature extraction and training are unable to deliver a generic solution. For this purpose, we used FCM, a non-supervised clustering algorithm. During the FCM segmentation process, the objective function is optimized across numerous rounds. Throughout the iterations, the clustering centers and membership degrees were continually updated (Rehman and Hussain, 2018). The FCM method separates a finite collection of N items (S=$s1$, $s2$, sn) into C clusters. Each component of vi (i = 1, 2..., N) is a vector of d dimensions. We design a technique to divide s into C clusters using cluster centers $u1$, $u2$, and so on in the centroid set u (Xiao et al., 2024; Xuemin et al., 2024). The FCM approach uses a representative matrix (g) to represent the membership of each element in each cluster. The matrix g may be defined using equation:

$$g(i,z), 1 \leq i \leq N; 1 \leq z \leq C$$

where $g(i, z)$ represents the membership value of the element si having cluster center vz. While calculating performance index J_{fcm}, and it is used to calculate the weighted sum of the distance between cluster center and components of the associated fuzzy cluster.

$$Jfcm = (g,u) = \sum_{i=1}^{M} \sum_{Z=1}^{N} g_{iz}^{b} \|s_i - v_z\|^2, 1 < b < \infty$$

where m indicates the number of clusters, N signifies the number of pixels, si is the ith pixel, vz is the zth cluster center, and b represents the blur exponent. The degree of membership function must meet the conditions specified in the equation below.

$$0 \leq b_{iz} \leq 1 \sum_{z=1}^{r} b_{iz} = 1, \sum_{i=1}^{M} b_{iz} \leq M, i = 1, 2, ..M \text{ and } z = 1, 2, .., r$$

Each time the membership function matrix is updated using equation:

$$b_{iz}^{b} = \frac{1}{\sum_{j=1}^{c} \left(\frac{d_{iz}^2}{d_{jz}^2} \right)^{\frac{2}{b-1}}}$$

The membership matrix (b_{iz}^{b}) is between [0,1], and the distance between cluster centroid (vi) and pixel (sz) is supplied by $\left(d_{iz}^2\right)$. The cluster centroid is determined by equation:

$$v_j = \frac{\sum_{z=1}^{N} g_{iz}^{b} s_z}{\sum_{z=1}^{N} g_{iz}^{b}}$$

A pixel receives a high membership value as it gets closer to the belonging cluster center and vice versa. Figure 4 depicts the results of the FCM segmentation.

3.3.2 Density-based spatial clustering (DBSCAN)

DBSCAN, or density-based spatial clustering, is a popular method in machine learning and data analysis (Khan et al., 2014; Deng et al., 2022). In contrast to conventional clustering techniques that need preset cluster numbers, DBSCAN utilizes a data-centric methodology. It uses data density and closeness to its advantage to detect variable-sized and irregularly formed clusters within complicated datasets (Bhattacharjee and Mitra, 2020; Liu et al., 2023). Initially, core points are determined based on having the fewest surrounding data points within a certain distance. These core locations are then expanded into clusters by adding nearby data points that satisfy density requirements (Chen et al., 2022; Zhang et al., 2023). Noise is defined as any data point that does not fit into a designated cluster or core point.

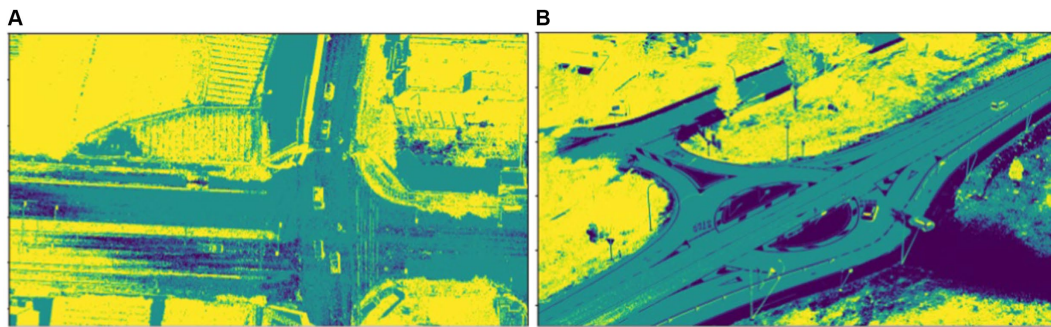$$N_\varepsilon(x_i) = \left\{ x_j \in X | dist(x_i, x_j) \leq \varepsilon \right\}$$

FIGURE 4
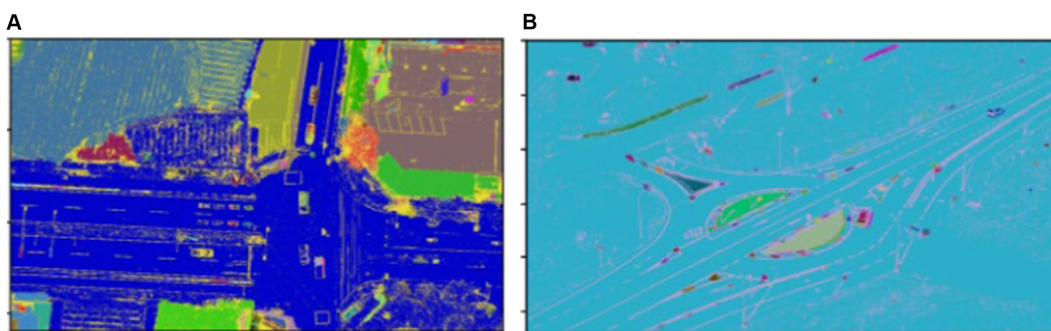Segmentation using FCM over **(A)** VEDAI dataset and **(B)** SRTID dataset.

FIGURE 5
Segmentation using DBSCAN over **(A)** VEDAI dataset and **(B)** SRTID dataset.

where $N_\varepsilon(x_i)$ represent the neighborhood of a point x_i, $x_j \in X$ denotes all points x_j belonging to the dataset X, $dist(x_i, x_j)$ calculates the distance between points, ε is a threshold distance parameter, defining the maximum distance for points to be considered neighbors (see Figure 5).

$$C = \left\{ x_i \in X \| N_\varepsilon(x_i) \geq MinPts \right\}$$

$$x_i \in X \| N_\varepsilon(x_j) \text{ and } x_j \in C, X = \left\{ x_1, x_2, \dots, x_n \right\}.$$

where x_i is the epsilon neighborhood of x_j and x_j is the core point.

The FCM and DBSCAN segmentation methods were evaluated in terms of computational cost and error rates determined using equations.

$$\text{Error Rate} = 1 - accuracy$$

FCM surpasses DBSCAN owing to its adeptness in managing datasets with varied cluster shapes and sizes. By adding fuzzy membership degrees, FCM addresses the ambiguity inherent in data point assignments, resulting in more adaptive and improved clustering. Furthermore, FCM enables increased control over cluster boundaries via parameterization, allowing for exact alterations to better fit the specific properties of the data. Table 1 exhibits FCM's

better efficacy and accuracy in picture segmentation on VEDAI and SRTID datasets. Considering both computation time and error rates, FCM shines, making its findings the preferable option for following tasks such as vehicle recognition, ID allocation, recovery, counting, and tracking.

3.4 Vehicle detection

YOLOv8 is utilized for vehicle recognition and radiates as an excellent single-shot detector capable of identifying, segmenting, and classifying with fewer training parameters (Chen et al., 2023b; Wang et al., 2023). According to the CSP principle, the C2f module replaces the C3 module to align with the YOLOv8 backbone, increasing gradient flow information while keeping YOLOv5 compliant. The C2f module combines C3 with ELAN in a unique manner, drawing on YOLOv7's ELAN methodology, ensuring YOLOv8 compatibility (). The SPPF module at the backbone's end employs three consecutive 5 × 5 Maxpools before concatenation in each layer to reliably identify objects of varied sizes with lightweight efficiency (Sun et al., 2019; Li S. et al., 2023; Yi et al., 2024).

YOLOv8 integrates PAN-FPN in its neck portion, which improves feature fusion and data use at different sizes (Mostofa et al., 2020; Xu et al., 2020). The neck module combines a final decoupled head structure, many C2f modules, and two up samplings (Song et al., 2022; Wu and Dong, 2023). YOLOv8's neck is like YOLOx's head idea,

which combines confidence and regression boxes to increase accuracy. It operates as an anchor-free model, detecting the object center directly, lowering box predictions, and speeding up the Non-Maximum Suppression (NMS) process, an important post-processing step (Li et al., 2024). Figure 6 shows automobiles spotted using YOLOv8.

3.5 ID allocation and recovery based on ORB features

Prior to tracking each identified vehicle in the subsequent image frames, an ID based on ORB traits was assigned to each detected vehicle. A quick and effective feature detector is ORB (Chien et al., 2016; Chen et al., 2022). FAST (Features from Accelerated Segment Test) key point detector is used for key-point detection. It is a more sophisticated version of the BRIEF (Binary Robust Independent Elementary Features) description. It is also rotationally and scale-invariant. Equation is used to get a patch moment (Luo et al., 2024; Yao et al., 2024).

$$n_{st} = \sum x^s y^t l(u,v)$$

where x and y are the image pixels' relative intensities, represented by the values s and t. These moments may be utilized to find the center of mass using equation:

$$N = \frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}$$

where the equation defines path orientation:

TABLE 1 Error rate comparison of DBSCAN and FCM.

Datasets	Error rate	
	DBSCAN	FCM
VEDAI	0.32	0.20
SRTID	0.37	0.23

$$\theta = atan\left(m_{01}, m_{10}\right)$$

The identified cars in the subsequent frames were compared using the extracted ORB features, and if a match was discovered, the ID was restored; if not, the vehicle was recorded in the system with a new ID (Cai et al., 2024). ID is restored across frames and ORB feature description is applied to the extracted cars; results are shown in Figure 7.

3.6 Vehicle counting

Using YOLOv8's vehicle detections, we incorporated vehicle counts in every image frame to conduct a thorough analysis of the traffic situation (Tian et al., 2022; Yang et al., 2023). Using a counter, each seen vehicle was painstakingly recorded under equation. Road traffic density at different times may be measured by counting the number of cars within each frame (Minh et al., 2023). This data is essential for enabling quick responses to unforeseen events like traffic jams or other circumstances that might impair traffic flow (Wu et al., 2019; Peng et al., 2023).

$$\text{Vehicle Count} = \sum_{i=1}^{N} T$$

where, T denotes the vehicle detections within a single frame, with the corresponding output visualized in Figure 8.

3.7 Vehicle tracking

We utilized the DeepSORT tracker to observe the movements of vehicles frame by frame. DeepSORT is a tracking approach that makes use of deep learning characteristics with the Kalman filter to track objects based on their appearance, motion, and velocity (Bin Zuraimi and Kamaru Zaman, 2021; Sun G. et al., 2022). Using the Mahalanobis

FIGURE 6
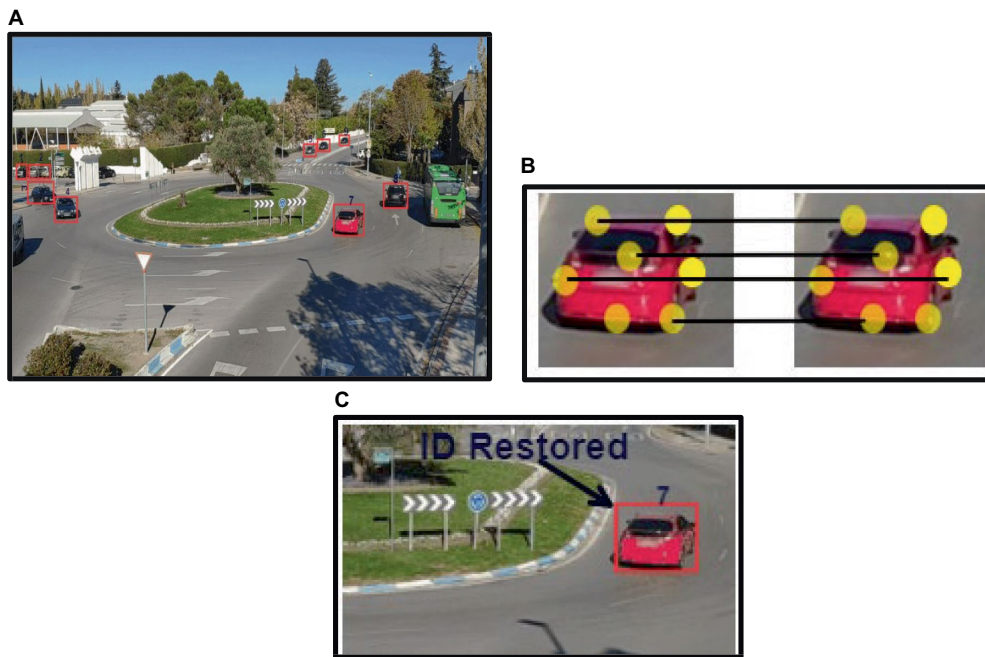Vehicle Detection over **(A)** VEDAI and **(B)** SRTID datasets marked with red boxes via the YOLOv8 algorithm.

FIGURE 7
ID assignment and restoration: **(A)** ID assigned to each vehicle based on ORB features; **(B)** features matching across frames; **(C)** ID restored for the same vehicle in succeeding frame.
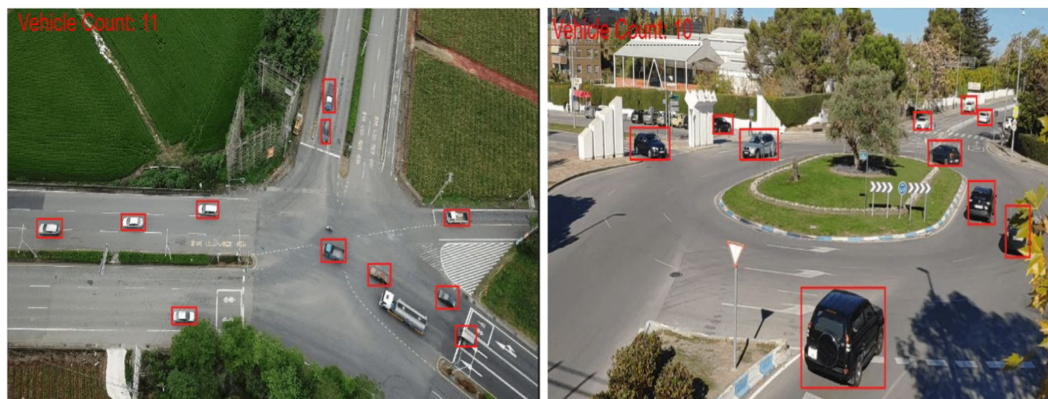

FIGURE 8
Density estimation by using vehicle count displayed at the left corner of each image.

distance metric between the Kalman state and the freshly obtained measurement, (Li et al., 2018; Sun Y. et al., 2022) the motion information is merged as described in equation:

$$k^{(1)}(i,j) = (k_j - v_i)^{\mathrm{T}} S_i^{-1}(k_j - v_i)$$

where k_j is the jth bounding box detection and (v_i, S_i) is the ith track distribution projection into space measurement. The appearance information has been computed using the smallest cosine distance, as

provided by equation, between the ith and jth detections in appearance space.

$$k^{(2)}(i,j) = \min\left\{1 - t_j^{\mathrm{T}} t_k^{(i)} \mid r_k^{(i)} \in \mathcal{R}_i\right\}$$

where t_j and $t_k^{(i)}$ represent the appearance and associated appearance descriptor, respectively. The extracted appearance and motion information is combined as given in equation:

ALGORITHM 1 Trajectory estimation of tracked vehicles

Input:
: $V = \{D^1, D^2, \ldots, D^z\}$
// rectangular coordinates of length $\{x_1, x_2, y_1, y_2\}$
Output: $\{P^r\}_{r=1}^R$
$R = \{R^1, R^2, \ldots, R^z\} \leftarrow$ DeepSORT(V)
feature vector = []
For i = 1 to the length of R
new_feature $\leftarrow$ ORB(R^i)
If feature vector == []
feature_vector $\leftarrow$ new_feature
Else
matches $\leftarrow$ (new_feature, *feature vector*)
If matches > 8
AssignID (R^i)
Else
feature_vector $\leftarrow$ new_feature
$x_1^i, x_2^i, y_1^i, y_2^i \leftarrow$ ExtractRectangularCoordinatesofVehicle ()

$$x_{center} \leftarrow \frac{(x_1^i + x_2^i)}{2}$$
$$y_{center} \leftarrow \frac{(y_1^i + y_2^i)}{2}$$

$R^r \leftarrow [x_{center}, y_{center}]$
end for
Return vehicle trajectories

$$c_{i,j} = \lambda k^{(1)}(i,j) + (1-\lambda)k^{(2)}(i,j)$$

where c is the corresponding weight. The appearance features are produced by a pre-trained CNN model that contains two convolution layers, six residual layers linked to a dense layer, one max pooling layer, and l2 normalization (Kumar et al., 2023; Mi et al., 2023). The DeepSORT algorithm's tracking mechanism is shown in Figure 9 (Singh et al., 2023). In Figure 10, the tracking result is shown.

3.8 Vehicle trajectory estimation

In addition to the previously computed density, we approximated the path traveled by each tracked vehicle. The trajectories taken by a vehicle may be utilized to construct vehicle detection (Adi et al., 2018; Bozcan and Kayacan, 2020). It may also be used to identify trajectory conflicts and accidents if it is further developed. The route is plotted if the vehicle is tracked (Chen and Wu, 2016; Wang et al., 2022). To approximate the trajectories, we used geometric coordinates from observed rectangular boxes. DeepSORT was used for location estimation and coordinate retrieval (Leitloff et al., 2014; Sheng et al., 2024). The center points of estimated locations, which represent individual vehicle IDs, were noted on a separate image, and then linked to construct trajectories.

The approach feeds detection coordinates into the DeepSORT tracker, which predicts vehicle placements in the following frame. Vehicle IDs are retrieved using ORB features; if the number of matches exceeds the threshold, relevant IDs are allocated, and new entries are assigned new IDs (Hou et al., 2023b). Rectangular coordinates and midpoints are used to trace vehicle routes. Algorithm 1 provides the exact processes for estimating the trajectory.

4 Experimental setup and datasets

4.1 Experimental setup

PC running x64-based Windows 11, with an Intel Core i5-12500H 2.40GHz CPU, 24GB RAM and other specifications is used to perform all the experiments. Spyder was used to acquire the results. The system employed two benchmark datasets, VEDAI and SRTID, to calculate proposed architecture's performance. In this section, concise discussion of the dataset used for vehicle identification and tracking system is done, as well as the results of several tests undertaken to examine the proposed system along with its assessment in comparison to numerous existing state-of-the-art traffic monitoring models.

4.2 Dataset description

In the subsequent subsection, we provide comprehensive and detailed descriptions of each dataset used in our study. Each dataset is thoroughly introduced, highlighting its unique characteristics, data sources, and collection methods.

4.2.1 VEDAI dataset

The VEDAI dataset (Sakla et al., 2017) is a standard point of reference for tiny target identification, specifically aerial images vehicle detection. This dataset comprises roughly 1,210 images of two distinct dimensions such as $1,024 \times 1,024$ pixels and 512×512 pixels. Both near-infrared and visible light spectra environment photos are acquired in this collection. The cars in acquired aerial shots feature incredibly tiny dimensions, lighting/shadowing shifts, various backdrops, multiple forms, scale variations, and secularities or occlusions. Moreover, it comprises nine separate kinds of automobiles, including aircraft, boats, camping cars, automobiles, pick-ups, tractors, trucks, vans, and other categories.

4.2.2 Spanish road traffic images dataset

The dataset consists of 15,070 images in.png format, followed by an equal number of files with the txt extension containing descriptions of the objects found in each image. There are 30,140 files including images and information. The images were shot at six separate places along urban and interurban highways, with motorways being deleted. The images include 155,328 identified vehicles, including automobiles (137,602) and motorbikes (17,726) (Bemposta Rosende et al., 2022).

4.2.3 VAID dataset

The VAID collection consists of six vehicle image categories: minibus, truck, sedan, bus, van, and automobile. The images were taken at a height of 90–95 meters above the ground by a drone under a variety of lighting circumstances. The photographs, which were captured at a resolution of $2,720 \times 1,530$ and at a frame rate of 23.98 frames per second, show the state of the roads and traffic at 10 locations in southern Taiwan, encompassing suburban, urban, and educational environments (Lin et al., 2020).

4.2.4 UAVDT dataset

UAVDT dataset: Comprising 80,000 representative frames, the UAVDT dataset (Du et al., 2018) includes UAV imagery of cars
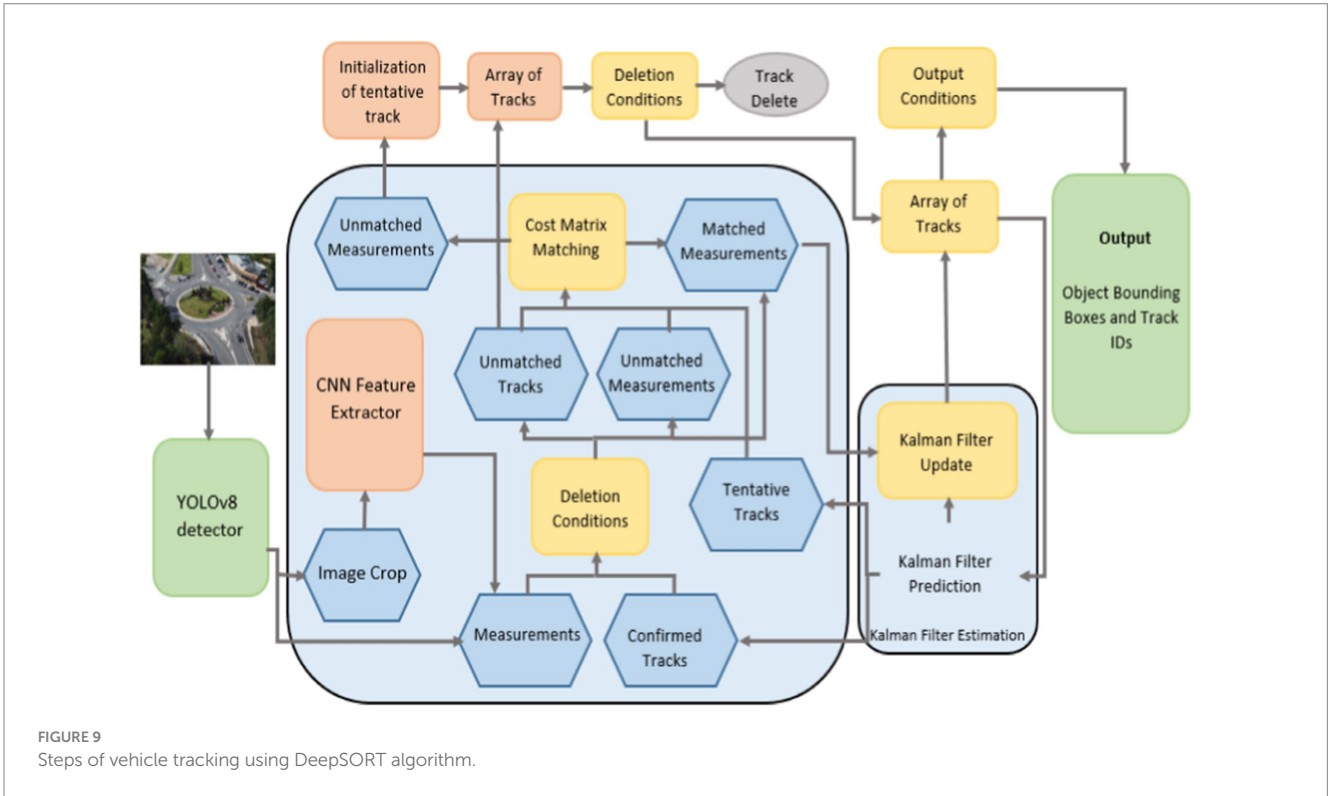
FIGURE 9
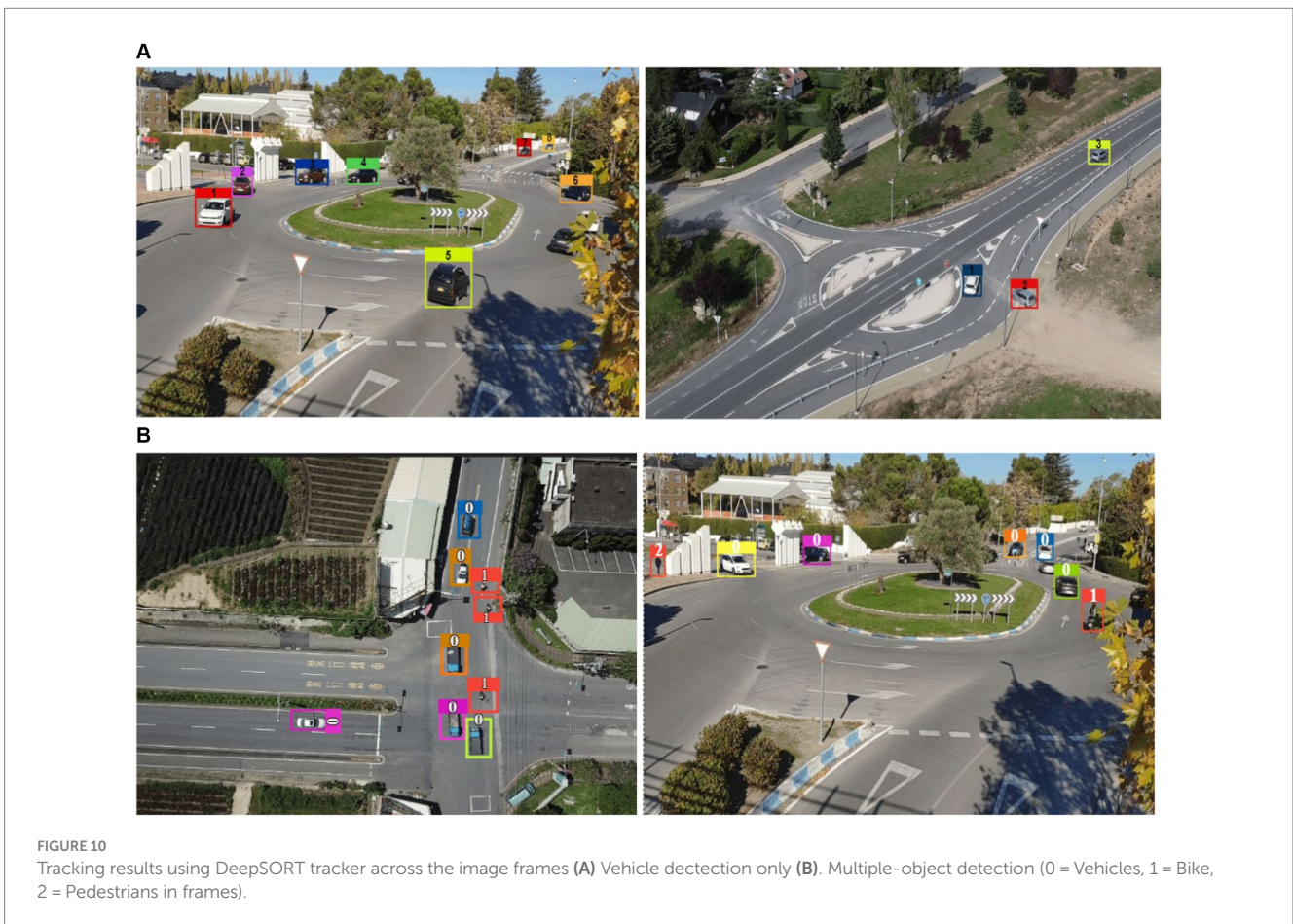Steps of vehicle tracking using DeepSORT algorithm.

FIGURE 10
Tracking results using DeepSORT tracker across the image frames **(A)** Vehicle dectection only **(B)**. Multiple-object detection (0 = Vehicles, 1 = Bike, 2 = Pedestrians in frames).

chosen from 10-h long recordings. Bounding boxes with up to 14 different attributes (e.g., weather, flying altitude, camera view, vehicle category, occlusion, etc.) completely annotate the photos. Each of the three sets—training, val, and testing consists of 5,000, 1,658, and 3,316 images, all 1,024 × 540 pixels. The photographs from the same video have comparable backdrops, camera viewpoints, and lighting (for those recorded at the same time of day).

4.3 Experiment I: semantic segmentation accuracy

The DBSCAN and FCM algorithms were compared and assessed in terms of segmentation accuracy and computational time. DBSCAN requires training on a bespoke dataset, increasing the model's computing cost as compared to FCM. Furthermore, FCM produced superior segmentation results than DBSCAN, therefore we utilized the FCM findings for future investigation. Table 2 shows the accuracy of both segmentation strategies.

4.4 Experiment II: precision, recall, and F1 scores

The effectiveness of vehicle detection and tracking has been assessed using these evaluation metrics, namely Precision, Recall, and F1 score as calculated by using equations below:

$$\text{Precision} = \frac{\sum TP}{\sum TP + \sum FP}$$

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

$$F1\,\text{Score} = \frac{2\left(\text{Precision} \times \text{Recall}\right)}{\left(\text{Precision} + \text{Recall}\right)}$$

Table 3 shows vehicle detection's precision, recall, and F1 scores on the segmented images, while Table 4 shows vehicle detection's precision, recall and F1 scores on the raw images. True Positive indicates how many cars are effectively identified. False Positives signify other detections besides cars, whereas False Negatives shows missing vehicles count. The findings indicate that this suggested system can accurately detect cars of varying sizes.

TABLE 2 Accuracies comparison of DBSCAN and FCM segmentation.

Datasets	Segmentations accuracy	
	DBSCAN	FCM
VEDAI	0.65	0.83
SRTID	0.68	0.79
VAID	0.62	0.72
UAVDT	0.65	0.75

In case of tracking, the number of cars successfully tracked is indicated as True Positive, whereas False Positive is the vehicles count falsely recorded, and False Negative represents untracked vehicles count. Table 4 shows the vehicle tracking method's precision, recall, and F1 scores (Figure 11).

4.5 Experiment

4.5.1 ID assignment and ID recovery

We used two new metrics to assess the ID assignment and recovery module, as shown in equations. The AID is the accurate ID rate, which is the proportion of correct ID numbers assigned to automobiles (Table 5).

$$\text{AIDRate} = \frac{\sum_{i=1}^{N} AIDs_i}{\sum_{i=1}^{N} ID_i}$$

TABLE 3 Precision, recall, and F1 Score for vehicle detection via YOLOv8 over segmented and raw images.

Datasets	Precision	Recall	F_1 score
VEDAI (segmented)	0.86	0.84	0.85
SRTID (segmented)	0.84	0.83	0.83
VAID (segmented)	0.85	0.82	0.83
UAVD (segmented)	0.81	0.82	0.81
VEDAI (raw)	0.83	0.80	0.81
SRTID (raw)	0.79	0.81	0.79
VAID (raw)	0.81	0.78	0.79
UAVDT (raw)	0.76	0.77	0.76

TABLE 4 Precision, recall, and F_1 score for vehicle tracking via DeepSORT.

Datasets	Precision	Recall	F_1 score
VEDAI	0.87	0.88	0.87
SRTID	0.83	0.82	0.82
VAID	0.88	0.85	0.86
UAVDT	0.84	0.83	0.83

TABLE 5 Precision, recall, and F_1 score for vehicle tracking via DeepSORT and ByteTrack.

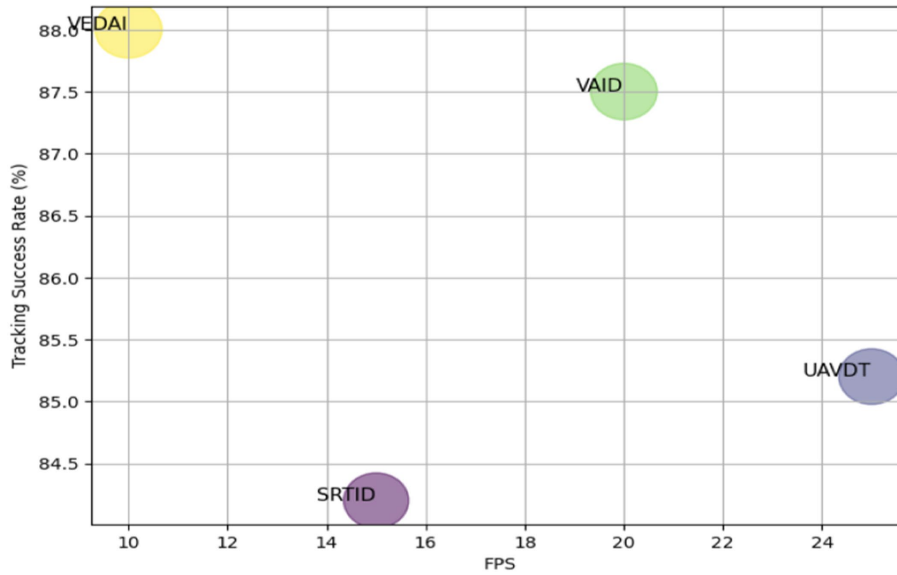| Datasets	Precision	Recall	F_1 score	Tracking success rate
VEDAI	0.89	0.90	0.89	88.1%
SRTID	0.85	0.84	0.84	84.2%
VAID	0.90	0.87	0.88	87.5%
UAVDT	0.86	0.85	0.85	85.2%

FIGURE 11
Tracking performance comparison of DeepSORT and ByteTrack across datasets.

TABLE 6 AIDRate and recovery rate for ID assignment recovery algorithm.

Datasets	AIDRate (%)	Recovery rate (%)
VEDAI	68	65
SRTID	63	59
VAID	59	55
UAVDT	65	60

where N is the total number of vehicles. $AIDs_i$ denotes the overall number of ID assignments made to the true vehicles, and ID_i denotes all of them. The Recovery Rate represents the percentage of true IDs recovered.

$$\text{Recovery Rate} = \frac{\sum_{i=1}^{N} TReCovers_i}{ReCovers}$$

where the total number of dissimilar vehicles is represented by $N. TReCovers_i$ represents the number of true recoveries and $ReCovers$ is the all-existing recoveries (Table 6).

4.6 Experiment IV: vehicle detection and tracking comparison with SOTA models

In this experiment, we have drawn a comparison of proposed model with other popular algorithms. Table 7 represents a comparison between our presented detection algorithm and other methods.

TABLE 7 Accuracy comparison of the proposed approach with SOTA vehicle detection models.

Methods	Accuracy %			
	VEDAI	SRTID	VAID	UAVDT
AVD NET (Mandal et al., 2020)	51.95	62.10	60.75	58.20
YOLOv5 (Hou S. et al., 2023)	75.54	73.20	74.30	72.10
Haar-like features (Nguyen and Tran, 2018)	77.0	65.00	64.50	62.00
D2Det (Cao et al., 2020)	73.40	56.92	68.10	64.30
R-FCN (Zhang, 2020)	68.90	73.0	69.80	70.20
SSD (Smith and Johnson, 2020)	71.00	70.30	81.0	74.50
R-FCN (Kim and Park, 2021)	72.50	70.90	75.0	73.20
NDFT (Cao et al., 2020)	63.50	62.80	64.00	52.03
YOLOv6 (Kumar and Singh, 2023)	74.07	71.20	72.80	74.07
YOLOv7 (Patel and Reddy, 2023)	76.80	74.50	75.60	72.0
Proposed method	79.4	77.7	83.1	77.2

TABLE 8 Accuracy comparison of the proposed approach with SOTA vehicle tracking models.

Methods	Accuracy %			
	VEDAI	SRTID	VAID	UAVDT
Faster R-CNN (du Terrail and Jurie, 2018)	83.50	78.0	81.0	79.5
Correlation filter tracking (Liu et al., 2019)	76.0	72.5	70.0	74.0
SIFT features (Mu et al., 2016)	72.5	75.10	73.0	71.0
HIOU (Hua and Anastasiu, 2019)	70.0	77.0	69.5	71.5
Kalman filter (Poostchi et al., 2017)	68.0	66.5	65.0	67.5
CNN (Alotaibi et al., 2020)	69.4	71.0	82.0	70.0
Affinity network (Cao et al., 2022)	73.2	74.0	71.5	74.0
MaSiamRPN (Sun et al., 2023)	82.0	79.1	83.0	84.0
Proposed method	88.6	82.2	84.6	86.1

Table 8 depicts the comparison of proposed tracking algorithm. Proposed model model performs better than other state-of-the-art methods.

5 Discussion/research limitation

For smart traffic monitoring based on aerial images, the suggested model is an efficient solution. While catering to high-definition aerial images, object detection is one of the most difficult problems. To get efficient results, we devised a technique that combines multi-label semantic segmentation with deepsort tracking. However, the suggested technique has significant limitations. First and foremost, the system has only been evaluated with RGB shots acquired during the daytime. Analyzing video or pictorial datasets in low-light conditions or at night can further confirm this proposed technique as a lot of researchers already have succeeded with such datasets. Furthermore, our segmentation and identification system have problems with partial or complete occlusions, tree-covered roadways, and similar items.

6 Conclusion

This study presents a novel approach to recognizing and tracking vehicles in aerial image sequences. Before proceeding with the detection phase, the model preprocesses aerial images to remove noise. To decrease complexity, the FCM approach is used for segmentation of all the images. The YOLOv8 algorithm is used for vehicle detection. It identifies vehicles by giving them a unique ID that contains ORB elements to aid recovery. DeepSORT tracks cars across frames and predicts their travel patterns. The suggested approach generated encouraging results across both datasets. The suggested system must be trained with additional vehicle classes. In addition, further elements may be added to increase vehicle recognition and tracking accuracy. In the future, we want to add additional features and dependable algorithms to the proposed model system to boost its efficiency and make it standard for all traffic scenarios.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found at: https://github.com/mr8bit/vedai https://www.kaggle.com/datasets/javiersanchezsoriano/traffic-images-captured-from-uavs.

Author contributions

MH: Methodology, Writing – original draft. MY: Data curation, Writing – original draft. NA: Investigation, Writing – review & editing. TS: Formal Analysis, Writing – review & editing. NAA: Project administration, Writing – review & editing. HR: Supervision, Writing – review & editing. AbA: Investigation, Writing – review & editing. AsA: Resources, Writing – review & editing.

Funding

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

Adi, K., Widodo, A. P., Widodo, C. E., Pamungkas, A., and Putranto, A. B. (2018). Automatic vehicle counting using background subtraction method on gray scale images and morphology operation. *J. Phys.* 1025:012025. doi: 10.1088/1742-6596/1025/1/012025

Alotaibi, M. F., Omri, M., Abdel-Khalek, S., Khalil, E., and Mansour, R. F. (2022). Computational intelligence-based harmony search algorithm for real-time object detection and tracking in video surveillance systems. *Mathematics* 10:733. doi: 10.3390/math10050733

Alotaibi, M., Alhussein, M., Alqhtani, A., and Alghamdi, M. (2020). CNN: vehicle tracking in aerial images using a convolutional neural network. *IEEE Access* 8, 164725–164734. doi: 10.1109/ACCESS.2020.3026861

Amna, S., Jalal, A., and Kim, K.. (2020). An accurate facial expression detector using multi-landmarks selection and local transform features. In: IEEE ICACS Conference.

Angel, A., Hickman, M., Mirchandani, P., and Chandnani, D. (2003). Methods of analyzing traffic imagery collected from aerial platforms. *IEEE Trans. Intell. Transp. Syst.* 4, 99–107. doi: 10.1109/TITS.2003.821208

Aqel, S., Hmimid, A., Sabri, M. A., and Aarab, A. (2017). Road traffic: vehicle detection and classification. In Proceedings of the 2017 Intelligent Systems and Computer Vision (ISCV), Venice, pp. 17–19.

Bemposta Rosende, S., Ghisler, S., Fernández-Andrés, J., and Sánchez-Soriano, J. (2022). Dataset: traffic images captured from UAVs for use in training machine vision algorithms for traffic management. *Data* 7:53. doi: 10.3390/data7050053

Bewley, A., Ge, Z., Ott, L., Ramos, F., and Upcroft, B.. (2016). Simple online and real-time tracking. In Proceedings of the International Conference Image Processing, ICIP, 2016-August, pp. 3464–3468.

Bhattacharjee, P., and Mitra, P. (2020). A survey of density-based clustering algorithms. *Front. Comput. Sci.* 15:151308. doi: 10.1007/s11704-019-9059-3

Bin Zuraimi, M. A., and Kamaru Zaman, F. H. (2021). Vehicle detection and tracking using YOLO and DeepSORT. In: 11th IEEE symposium on computer applications & industrial electronics (ISCAIE), Penang, Malaysia, 2021, pp. 23–29.

Bozcan, I., and Kayacan, E. (2020) AU-AIR: a multi-modal unmanned aerial vehicle dataset for low altitude traffic surveillance. In: 2020 IEEE ICRA, pp. 8504–8510.

Cai, D., Li, R., Hu, Z., Lu, J., Li, S., and Zhao, Y. (2024). A comprehensive overview of core modules in visual SLAM framework. *Neurocomputing* 590:127760. doi: 10.1016/j.neucom.2024.127760

Cao, J., Cholakkal, H., Anwer, R. M., Khan, F. S., Pang, Y., and Shao, L. (2020). "D2DET: towards high quality object detection and instance segmentation" in Proceedings of the IEEE/CVF Conf. On computer vision and pattern recognition (Seattle, WA), 11482–11491.

Cao, B., Li, M., Liu, X., Zhao, J., Cao, W., and Lv, Z. (2021). Many-objective deployment optimization for a drone-assisted camera network. *IEEE Trans Netw Sci Eng* 8, 2756–2764. doi: 10.1109/TNSE.2021.3057915

Cao, X., Yang, X., and Xu, W. (2022). Affinity network for multi-view object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* 44, 1702–1715. doi: 10.1109/TPAMI.2021.3067218

Chen, X., and Meng, Q.. (2016). Robust vehicle tracking and detection from UAVs. In Proceedings of the 7th International Conference on Soft Computing Pattern Recognition, SoCPaR 2015, pp. 241–246.

Chen, J., Wang, Q., Cheng, H. H., Peng, W., and Xu, W. (2022). A review of vision-based traffic semantic understanding in ITSs. *IEEE Trans. Intell. Transp. Syst.* 23, 19954–19979. doi: 10.1109/TITS.2022.3182410

Chen, J., Wang, Q., Peng, W., Xu, H., Li, X., and Xu, W. (2023a). Disparity-based multiscale fusion network for transportation detection. *IEEE Trans. Intell. Transp. Syst.* 23, 18855–18863. doi: 10.1109/TITS.2022.3161977

Chen, Y., and Wu, Q.. (2016). Moving vehicle detection based on optical flow estimation of edge. In: Proceedings of the International Conference on Computing, 2016, pp. 754–758.

Chen, J., Xu, M., Xu, W., Li, D., Peng, W., and Xu, H. (2023b). A flow feedback traffic prediction based on visual quantified features. *IEEE Trans. Intell. Transp. Syst.* 24, 10067–10075. doi: 10.1109/TITS.2023.3269794

Chien, H.-J., Chuang, C.-C., Chen, C.-Y., and Klette, R. (2016). When to use what feature? SIFT, SURF, ORB, or A-KAZE features for monocular visual odometry. In: 2016 international conference on image and vision computing New Zealand (IVCNZ), New Zealand, pp. 1–6.

Chong, Q., Jindong, X., Ding, Y., and Dai, Z. (2023). A multiscale bidirectional fuzzy-driven learning network for remote sensing image segmentation. *IJRS* 44, 6860–6881. doi: 10.1080/01431161.2023.2275326

Dai, X., Xiao, Z., Jiang, H., and Lui, J. C. S. (2024). UAV-assisted task offloading in vehicular edge computing networks. *IEEE Trans. Mob. Comput.* 23, 2520–2534. doi: 10.1109/TMC.2023.3259394

Deng, Z. W., Zhao, Y. Q., Wang, B. H., Gao, W., and Kong, X. (2022). A preview driver model based on sliding-mode and fuzzy control for articulated heavy vehicle. *Meccanica* 57, 1853–1878. doi: 10.1007/s11012-022-01532-6

Di, Y., Li, R., Tian, H., Guo, J., Shi, B., Wang, Z., et al. (2023). A maneuvering target tracking based on fastIMM-extended Viterbi algorithm. *Neural Comput. Appl.* 35, 1–10. doi: 10.1007/s00521-023-09039-1

Dikbayir, H. S., and İbrahim Bülbül, H. (2020). "Deep learning based vehicle detection from aerial images" in 19th IEEE international conference on machine learning and applications (ICMLA) (Miami, FL), 956–960.

Ding, C., Li, C., Xiong, Z., Li, Z., and Liang, Q. (2024). Intelligent identification of moving trajectory of autonomous vehicle based on friction Nano-generator. *IEEE Trans. Intell. Transp. Syst.* 25, 3090–3097. doi: 10.1109/TITS.2023.3303267

Ding, Y., Zhang, W., Zhou, X., Liao, Q., Luo, Q., and Ni, L. M. (2021). FraudTrip: taxi fraudulent trip detection from corresponding trajectories. *IEEE Internet Things J.* 8, 12505–12517. doi: 10.1109/JIOT.2020.3019398

Drouyer, S., and de Franchis, C. (2019). Highway traffic monitoring on medium resolution satellite images. In IGARSS IEEE International Geoscience and Remote Sensing Symposium, pp. 1228–1231.

Du, D., Qi, Y., Yu, H., Yang, Y., Duan, K., Li, G., et al. (2018). The unmanned aerial vehicle benchmark: object detection and tracking. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 370–386.

du Terrail, J. O., and Jurie, F. (2018). Faster RER-CNN: application to the detection of vehicles in aerial images. *arXiv:1809.07628*. doi: 10.48550/arXiv.1809.07628

Gao, T., Li, K., Chen, T., Liu, M., Mei, S., Xing, K., et al. (2020). A novel UAV sensing image defogging method. *IEEE J. Select. Top. Appl. Earth Observ. Remote Sens.* 13, 2610–2625. doi: 10.1109/JSTARS.2020.2998517

Gu, Y., Hu, Z., Zhao, Y., Liao, J., and Zhang, W. (2024). MFGTN: a multi-modal fast gated transformer for identifying single trawl marine fishing vessel. *Ocean Eng.* 303:117711. doi: 10.1016/j.oceaneng.2024.117711

Hao, J., Chen, P., Chen, J., and Li, X. (2024). Multi-task federated learning-based system anomaly detection and multi-classification for microservices architecture. *Futur. Gener. Comput. Syst.* 159, 77–90. doi: 10.1016/j.future.2024.05.006

He, Y., and Li, L.. (2019). A novel multi-source vehicle detection algorithm based on deep learning. In Proceedings of the International Conference on Signal Processing, ICSP, vol. 2018, August, pp. 979–982.

Hinz, S., Lenhart, D., and Leitloff, J. (2006). "Detection and tracking of vehicles low framerate aerial image GIS integration car detection car tracking calculation of traffic parameters" in Image, International Society for Photogrammetry and Remote Sensing (ISPRS) (Rochester, NY).

Hou, S., Fan, L., Zhang, F., and Liu, B. (2023). An improved lightweight YOLOv5 for remote sensing images. In: Proceedings of the 32nd international conference on artificial neural networks, Heraklion, Greece, 26–29, pp. 77–89.

Hou, X., Xin, L., Fu, Y., Na, Z., Gao, G., Liu, Y., et al. (2023b). A self-powered biomimetic mouse whisker sensor (BMWS) aiming at terrestrial and space objects perception. *Nano Energy* 118:109034. doi: 10.1016/j.nanoen.2023.109034

Hou, X., Zhang, L., Su, Y., Gao, G., Liu, Y., Na, Z., et al. (2023a). A space crawling robotic bio-paw (SCRBP) enabled by triboelectric sensors for surface identification. *Nano Energy* 105:108013. doi: 10.1016/j.nanoen.2022.108013

Hua, S., and Anastasiu, D. C. (2019). Effective vehicle tracking algorithm for smart traffic networks. In: 2019 IEEE international conference on service-oriented system engineering (SOSE), San Francisco, CA, pp. 67–6709.

Huang, Z., Fang, H., Li, Q., Li, Z., Zhang, T., Sang, N., et al. (2018). Optical remote sensing image enhancement with weak structure preservation via spatially adaptive gamma correction. *Infrared Phys. Technol.* 94, 38–47. doi: 10.1016/j.infrared.2018.08.019

Huang, H., Meng, F., Zhou, S., Jiang, F., and Manogaran, G. (2019). Brain image segmentation based on FCM clustering algorithm and rough set. *IEEE Access* 7, 12386–12396. doi: 10.1109/ACCESS.2019.2893063

Khan, K., Rehman, S. U., Aziz, K., Fong, S., and Sarasvady, S.. (2014). DBSCAN: past, present and future. In: The fifth international conference on the applications of digital information and web technologies (ICADIWT), India, pp. 232–238.

Kim, J., and Park, S. (2021). Hybrid approach for vehicle detection in VAID using YOLOv5 and R-FCN. *J. Adv. Transport. Syst.* 28, 412–423. doi: 10.1007/s11554-021-01078-5

Kumar, R., and Singh, A. (2023). Efficient vehicle detection in UAVDT dataset using YOLOv6 and deep SORT. *IEEE Trans. Intell. Transport. Syst.* 34, 405–417. doi: 10.1109/TITS.2022.3184512

Kumar, S., Singh, S. K., Varshney, S., Singh, S., Kumar, P., Kim, B.-G., et al. (2023). Fusion of deep Sort and Yolov5 for effective vehicle detection and tracking scheme in real-time traffic management sustainable system. *Sustain. For.* 15:16869. doi: 10.3390/su152416869

Leitloff, J., Rosenbaum, D., Kurz, F., Meynberg, O., and Reinartz, P. (2014). An operational system for estimating road traffic information from aerial images. *Remote Sens.* 6, 11315–11341. doi: 10.3390/rs61111315

Li, S., Chen, J., Peng, W., Shi, X., and Bu, W. (2023). A vehicle detection method based on disparity segmentation. *Multimed. Tools Appl.* 82, 19643–19655. doi: 10.1007/s11042-023-14360-x

Li, J., Han, L., Zhang, C., Li, Q., and Liu, Z. (2023). Spherical convolution empowered viewport prediction in 360 video multicast with limited FoV feedback. *ACM Trans. Multimedia Comput. Commun. Appl.* 19, 1–23. doi: 10.1145/3511603

Li, Z., Wang, Y., Zhang, R., Ding, F., Wei, C., and Lu, J. (2024). A LiDAR-OpenStreetMap matching method for vehicle global position initialization based on boundary directional feature extraction. *IEEE Trans. Intell. Vehicles* 9, 1–13. doi: 10.1109/TIV.2024.3393229

Li, F., Zhang, S., Wu, Y., and Huang, L. (2018). Yolov3: a real-time vehicle tracking system based on deep learning. *IEEE Access* 6, 9135–9143.

Lin, C.-J., and Jhang, J.-Y. (2022). Intelligent traffic-monitoring system based on YOLO and convolutional fuzzy neural networks. *IEEE Access* 10, 14120–14133. doi: 10.1109/ACCESS.2022.3147866

Lin, H. Y., Tu, K. C., and Li, C. Y. (2020) "VAID: An aerial image dataset for vehicle detection and classification." IEEE Access, vol. 8, pp. 212209–212219.

Liu, L., Zhang, S., Zhang, L., Pan, G., and Yu, J. (2023). Multi-UUV maneuvering counter-game for dynamic target scenario based on fractional-order recurrent neural network. *IEEE Trans. Cybern.* 53, 4015–4028. doi: 10.1109/TCYB.2022.3225106

Liu, Z., Wang, X., Yu, B., Xu, X., and Yang, Y. (2019). Vehicle detection in aerial images using RetinaNet and correlation filter tracking. *J. Remote Sens. Technol.* 10, 134–149. doi: 10.1016/j.jrst.2019.04.003

Luo, G., Shao, C., Cheng, N., Zhou, H., Zhang, H., Yuan, Q., et al. (2024). EdgeCooper: network-aware cooperative LiDAR perception for enhanced vehicular awareness. *IEEE J Sel Areas Commun* 42, 207–222. doi: 10.1109/JSAC.2023.3322764

Mandal, M., Shah, M., Meena, P., Devi, S., and Vipparthi, S. K. (2020). AVDNet: a small-sized vehicle detection network for aerial visual data. *IEEE Geosci. Remote Sens. Lett.* 17, 494–498. doi: 10.1109/LGRS.2019.2923564

Mi, C., Liu, Y., Zhang, Y., Wang, J., Feng, Y., and Zhang, Z. (2023). A vision-based displacement measurement system for foundation pit. *IEEE Trans. Instrum. Meas.* 72, 1–15. doi: 10.1109/TIM.2023.3311069

Minh, K. T., Dinh, Q.-V., Nguyen, T.-D., and Nhut, T. N.. (2023). Vehicle counting on Vietnamese street. In: IEEE statistical signal processing workshop (SSP), Hanoi, Vietnam, 2023, pp. 160–164.

Mostofa, M., Ferdous, S. N., and Nasrabadi, N. M. (2020). A joint cross-modal super-resolution approach for vehicle detection in aerial imagery. *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications II* 11413, 184–194.

Mu, K., Hui, F., and Zhao, X. (2016). Multiple vehicle detection and tracking in highway traffic surveillance video based on sift feature matching. *J. Inf. Process. Syst.* 12, 183–195. doi: 10.3745/JIPS.02.0040

Najiya, K. V., and Archana, M.. (2018). UAV video processing for traffic surveillance with enhanced vehicle detection. In Proceedings of the International Conference on Inventive Communication and Computational Technologies, pp. 662–668.

Nguyen, T., and Tran, P. (2018). A combined approach for vehicle detection using YOLOv2 and traditional feature-based methods. *Mach. Learn. Remote Sens.* 8, 89–100. doi: 10.1016/j.mlsr.2018.03.003

Omar, W., Oh, Y., Chung, J., and Lee, I. (2021). Aerial dataset integration for vehicle detection based on YOLOv4. *Kor. J. Remote Sens.* 37, 747–761. doi: 10.7780/kjrs.2021.37.4.6

Ozturk, M., and Cavus, E. (2021). "Vehicle detection in aerial imaginary using a miniature CNN architecture" in Proceedings of the 2021 international conference on innovations in intelligent systems and applications (INISTA) (Kocaeli), 1–6.

Patel, N., and Reddy, S. (2023). Real-time vehicle detection in UAVDT using YOLOv7 and SORT. *J. Aerial Robot.* 16, 120–135. doi: 10.1002/rob.22082

Peng, J. J., Chen, X. G., Wang, X. K., Wang, J. Q., Long, Q. Q., and Yin, L. J. (2023). Picture fuzzy decision-making theories and methodologies: a systematic review. *Int. J. Syst. Sci.* 54, 2663–2675. doi: 10.1080/00207721.2023.2241961

Poostchi, M., Palaniappan, K., and Seetharaman, G.., (2017). Spatial pyramid context-aware moving vehicle detection and tracking in urban aerial imagery. In: 2017 14th IEEE international conference on advanced video and signal based surveillance (AVSS), Lecce, pp. 1–6.

Qu, Z., Liu, X., and Zheng, M. (2022). Temporal-spatial quantum graph convolutional neural network based on Schrödinger approach for traffic congestion prediction. *IEEE Trans. Intell. Transp. Syst.* 24, 8677–8686. doi: 10.1109/TITS.2022.3203791

Rafique, A. A., al-Rasheed, A., Ksibi, A., Ayadi, M., Jalal, A., Alnowaiser, K., et al. (2023). Smart traffic monitoring through pyramid pooling vehicle detection and filter-based tracking on aerial images. *IEEE Access* 11, 2993–3007. doi: 10.1109/ACCESS.2023.3234281

Rehman, S. N., and Hussain, M. A. (2018). Fuzzy C-means algorithm-based satellite image segmentation. *Indones. J. Electr. Eng. Comput. Sci.* 9, 332–334. doi: 10.11591/ijeecs.v9.i2.pp332-334

Ren, Y., Lan, Z., Liu, L., and Yu, H. (2024). EMSIN: enhanced multi-stream interaction network for vehicle trajectory prediction. *IEEE Trans. Fuzzy Syst.* 32, 1–15. doi: 10.1109/TFUZZ.2024.3360946

Rong, Y., Xu, Z., Liu, J., Liu, H., Ding, J., Liu, X., et al. (2022). Du-bus: a realtime bus waiting time estimation system based on multi-source data. *IEEE Trans. Intell. Transp. Syst.* 23, 24524–24539. doi: 10.1109/TITS.2022.3210170

Sakla, W., Konjevod, G., and Mundhenk, T. N. (2017). Deep multi-modal vehicle detection in aerial ISR imagery. In: IEEE winter conference on applications of computer vision (WACV), Santa Rosa, CA, pp. 916–923.

Schreuder, M., Hoogendoorn, S. P., van Zulyen, H. J., Gorte, B., and Vosselman, G. (2003). Traffic data collection from aerial imagery. *IEEE Trans. Intell. Transp.* 1, 779–784. doi: 10.1109/ITSC.2003.1252056

Sheng, H., Wang, S., Chen, H., Yang, D., Huang, Y., Shen, J., et al. (2024). Discriminative feature learning with co-occurrence attention network for vehicle ReID. *IEEE Trans. Circuits Syst. Video Technol.* 34, 3510–3522. doi: 10.1109/TCSVT.2023.3326375

Shi, Y., Xi, J., Hu, D., Cai, Z., and Xu, K. (2023). RayMVSNet++: learning ray-based 1D implicit fields for accurate multi-view stereo. *IEEE Trans. Pattern Anal. Mach. Intell.* 45, 1–17. doi: 10.1109/TPAMI.2023.3296163

Singh, I. S., Wijegunawardana, I. D., Samarakoon, S. M. B. P., Muthugala, M. A. V. J., and Elara, M. R. (2023). Vision-based dirt distribution mapping using deep learning. *Sci. Rep.* 13:12741. doi: 10.1038/s41598-023-38538-3

Smith, K., and Johnson, L. (2020). Vehicle detection and tracking in VAID dataset using SSD and deep SORT. *Adv. Comput. Vis.* 12, 350–362. doi: 10.3390/s20154296

Song, F., Liu, Y., Shen, D., Li, L., and Tan, J. (2022). Learning control for motion coordination in water scanners: toward gain adaptation. *IEEE Trans. Ind. Electron.* 69, 13428–13438. doi: 10.1109/TIE.2022.3142428

Sun, R., Dai, Y., and Cheng, Q. (2023). An adaptive weighting strategy for multisensor integrated navigation in urban areas. *IEEE Internet Things J.* 10, 12777–12786. doi: 10.1109/JIOT.2023.3256008

Sun, G., Sheng, L., Luo, L., and Yu, H. (2022). Game theoretic approach for multipriority data transmission in 5G vehicular networks. *IEEE Trans. Intell. Transp. Syst.* 23, 24672–24685. doi: 10.1109/TITS.2022.3198046

Sun, G., Song, L., Yu, H., Chang, V., Du, X., and Guizani, M. (2019). V2V routing in a VANET based on the autoregressive integrated moving average model. *IEEE Trans. Veh. Technol.* 68, 908–922. doi: 10.1109/TVT.2018.2884525

Sun, G., Zhang, Y., Liao, D., Yu, H., Du, X., and Guizani, M. (2018). Bus-trajectory-based street-centric routing for message delivery in urban vehicular ad hoc networks. *IEEE Trans. Veh. Technol.* 67, 7550–7563. doi: 10.1109/TVT.2018.2828651

Sun, G., Zhang, Y., Yu, H., Du, X., and Guizani, M. (2020). Intersection fog-based distributed routing for V2V communication in urban vehicular ad hoc networks. *IEEE Trans. Intell. Transp. Syst.* 21, 2409–2426. doi: 10.1109/TITS.2019.2918255

Sun, Y., Zhao, Y., and Wang, S. (2022). Multiple traffic target tracking with spatial-temporal affinity network. *Comput. Intell. Neurosci* 2022:9693767. doi: 10.1155/2022/9693767

Tang, Q., Qu, S., Zhang, C., Tu, Z., and Cao, Y. (2024). Effects of impulse on prescribed-time synchronization of switching complex networks. *Neural Netw.* 174:106248. doi: 10.1016/j.neunet.2024.106248

Teutsch, M., Krüger, W., and Beyerer, J. (2017). Moving object detection in top-view aerial videos improved by image stacking. *Opt. Eng.* 56:083102. doi: 10.1117/1.OE.56.8.083102

Tian, J., Wang, B., Guo, R., Wang, Z., Cao, K., and Wang, X. (2022). Adversarial attacks and defenses for deep-learning-based unmanned aerial vehicles. *IEEE Internet Things J.* 9, 22399–22409. doi: 10.1109/JIOT.2021.3111024

Wang, G., Chen, Y., An, P., Hong, H., Hu, J., and Huang, T. (2023). UAV-YOLOv8: a small-object-detection model based on improved YOLOv8 for UAV aerial photography scenarios. *Sensors* 23:7190. doi: 10.3390/s23167190

Wang, R., Gu, Q., Lu, S., Tian, J., Yin, Z., Yin, L., et al. (2024). FI-NPI: exploring optimal control in parallel platform systems. *Electronics* 13:1168. doi: 10.3390/electronics13071168

Wang, S., Sheng, H., Yang, D., Zhang, Y., Wu, Y., and Wang, S. (2022). Extendable multiple nodes recurrent tracking framework with RTU++. *IEEE Trans. Image Process.* 31, 5257–5271. doi: 10.1109/TIP.2022.3192706

Weng, S. K., Kuo, C. M., and Tu, S. K. (2006). Video object tracking using adaptive Kalman filter. *J. Vis. Commun. Image Represent.* 17, 1190–1208. doi: 10.1016/j.jvcir.2006.03.004

Wu, T., and Dong, Y. (2023). YOLO-SE: improved YOLOv8 for remote sensing object detection and recognition. *Appl. Sci.* 13:12977. doi: 10.3390/app132412977

Wu, Z., Zhu, H., He, L., Zhao, Q., Shi, J., and Wu, W. (2023). Real-time stereo matching with high accuracy via spatial attention-guided Upsampling. *Appl. Intell.* 53, 24255–24274. doi: 10.1007/s10489-023-04646-w

Wu, W., Zhu, H., Yu, S., and Shi, J. (2019). Stereo matching with fusing adaptive support weights. *IEEE Access* 7, 61960–61974. doi: 10.1109/ACCESS.2019.2916035

Xiao, Z., Shu, J., Jiang, H., Min, G., Chen, H., and Han, Z. (2023). Overcoming occlusions: perception task-oriented information sharing in connected and autonomous vehicles. *IEEE Netw.* 37, 224–229. doi: 10.1109/MNET.018.2300125

Xiao, Z., Shu, J., Jiang, H., Min, G., Liang, J., and Iyengar, A. (2024). Toward collaborative occlusion-free perception in connected autonomous vehicles. *IEEE Trans. Mob. Comput.* 23, 4918–4929. doi: 10.1109/TMC.2023.3298643

Xu, H., Cao, Y., Lu, Q., and Yang, Q. (2020). "Performance comparison of small object detection algorithms of UAV based aerial images" in 2020 19th international symposium on distributed computing and applications for business engineering and science (DCABES) (Xuzhou), 16–19.

Xu, X., Liu, W., and Yu, L. (2022). Trajectory prediction for heterogeneous traffic-agents using knowledge correction data-driven model. *Inf. Sci.* 608, 375–391. doi: 10.1016/j.ins.2022.06.073

Xuemin, Z., Haitao, D., Zenggang, X., Ying, R., Yanchao, L., Yuan, L., et al. (2024). Self-organizing key security management algorithm in socially aware networking. *J. Signal Process. Syst.* 96, 369–383. doi: 10.1007/s11265-024-01918-7

Yang, D., Cui, Z., Sheng, H., Chen, R., Cong, R., Wang, S., et al. (2023). An occlusion and noise-aware stereo framework based on light field imaging for robust disparity estimation. *IEEE Trans. Comput.* 73, 764–777. doi: 10.1109/TC.2023.3343098

Yang, M., Han, W., Song, Y., Wang, Y., and Yang, S. (2024). Data-model fusion driven intelligent rapid response design of underwater gliders. *Adv. Eng. Inform.* 61:102569. doi: 10.1016/j.aei.2024.102569

Yao, Y., Zhao, B., Zhao, J., Shu, F., Wu, Y., and Cheng, X. (2024). Anti-jamming technique for IRS aided JRC system in Mobile vehicular networks. *IEEE Trans. Intell. Transp. Syst.* 25, 1–11. doi: 10.1109/TITS.2024.3384038

Yi, H., Liu, B., Zhao, B., and Liu, E. (2024). Small object detection algorithm based on improved YOLOv8 for remote sensing. *IEEE J. Select. Top. Appl. Earth Observ. Remote Sens.* 17, 1734–1747. doi: 10.1109/JSTARS.2023.3339235

Yin, Y., Guo, Y., Su, Q., and Wang, Z. (2022). Task allocation of multiple unmanned aerial vehicles based on deep transfer reinforcement learning. *Drones* 6:215. doi: 10.3390/drones6080215

Zhang, X. (2020). Advanced vehicle detection in aerial images using YOLOv3 and R-FCN. *J. Real-Time Image Process.* 17, 77–88. doi: 10.1007/s11554-019-00843-4

Zhang, Y., Li, S., Wang, S., Wang, X., and Duan, H. (2023). Distributed bearing-based formation maneuver control of fixed-wing UAVs by finite-time orientation estimation. *Aerosp. Sci. Technol.* 136:108241. doi: 10.1016/j.ast.2023.108241

Zhang, X., and Zhu, X. (2019). Vehicle detection in aerial infrared images via an improved Yolov3 network. In: Proceedings of the 2019 IEEE 4th international conference on signal and image processing (ICSIP), Wuxi, China, pp. 372–376.

Zhao, X., Fang, Y., Min, H., Wu, X., Wang, W., and Teixeira, R. (2024). Potential sources of sensor data anomalies for autonomous vehicles: An overview from road vehicle safety perspective. *Expert Syst. Appl.* 236:121358. doi: 10.1016/j.eswa.2023.121358

Zhao, L., Xu, H., Qu, S., Wei, Z., and Liu, Y. (2024). Joint trajectory and communication design for UAV-assisted symbiotic radio networks. *IEEE Trans. Veh. Technol.* 73, 8367–8378. doi: 10.1109/TVT.2024.3356587

Zheng, W., Lu, S., Yang, Y., Yin, Z., Yin, L., and Ali, H. (2024). Lightweight transformer image feature extraction network. *PeerJ Comput. Sci.* 10:e1755. doi: 10.7717/peerj-cs.1755

Zhu, H., Xu, D., Huang, Y., Jin, Z., Ding, W., Tong, J., et al. (2024). Graph structure enhanced pre-training language model for knowledge graph completion. *IEEE Trans. Emerg. Top. Comput. Intell.* 8, 2697–2708. doi: 10.1109/TETCI.2024.3372442