RESEARCH ARTICLE

A Novel Framework for Vehicle Detection and Tracking in Night Ware Surveillance Systems

NOUF ABDULLAH ALMUJALLY[1], ASIFA MEHMOOD QURESHI[2], ABDULWAHAB ALAZEB[3], HAMEEDUR RAHMAN [2], TOUSEEF SADIQ [4], (Graduate Student Member, IEEE), MOHAMMED ALONAZI[5], ASAAD ALGARNI [6], AND AHMAD JALAL [2]

[1]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia
[2]Faculty of Computer Science and AI, Air University, Islamabad 44000, Pakistan
[3]Department of Computer Science, College of Computer Science and Information Systems, Najran University, Najran 55461, Saudi Arabia
[4]Centre for Artificial Intelligence Research, Department of Information and Communication Technology, University of Agder, 4879 Grimstad, Norway
[5]Department of Information Systems, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj 16273, Saudi Arabia
[6]Department of Computer Sciences, Faculty of Computing and Information Technology, Northern Border University, Rafha 91911, Saudi Arabia

Corresponding authors: Hameedur Rahman (hameed.rahman@mail.au.edu.pk) and Touseef Sadiq (touseef.sadiq@uia.no)

ABSTRACT In the field of traffic surveillance systems, where effective traffic management and safety are the primary concerns, vehicle detection and tracking play an important role. Low brightness, low contrast, and noise are issues with low-light environments that result from poor lighting or insufficient exposure. In this paper, we proposed a vehicle detection and tracking model based on the aerial image captured during nighttime. Before object detection, we performed fogging and image enhancement using MIRNet architecture. After pre-processing, YOLOv5 was used to locate each vehicle position in the image. Each detected vehicle was subjected to a Scale-Invariant Feature Transform (SIFT) feature extraction algorithm to assign a unique identifier to track multiple vehicles in the image frames. To get the best possible location of vehicles in the succeeding frames templates were extracted and template matching was performed. The proposed model achieves a precision score of 0.924 for detection and 0.861 for tracking with the Unmanned Aerial Vehicle Benchmark Object Detection and Tracking (UAVDT) dataset, 0.904 for detection, and 0.833 for tracking with the Vision Meets Drone Single Object-Tracking (VisDrone) dataset.

INDEX TERMS Defogging, deep learning, yolov5, feature fusion, vehicle detection and tracking, image normalization.

I. INTRODUCTION

Vehicle recognition in aerial images is crucial for both military and civilian applications. Military target strikes and traffic control can both benefit from the use of this technology. Researchers have proposed various techniques for object recognition in aerial photos in the daytime with sufficient lightning, producing remarkable results [1]. However, vehicle detection of objects in low light conditions is a challenging and significant issue with surveillance

The associate editor coordinating the review of this manuscript and approving it for publication was Ghufran Ahmed .

camera applications. In low-illumination conditions there is less information available and difficulty extracting enough useful features, such as at night time there is background light interference, the object is underexposed, and brightness and contrast are poor which results in low-image quality [2].

The generic object detection method has poor accuracy and a limited ability to extract the intended objects. Consequently, in low brightness, capturing every detail of a scene is impossible, and a lot of detailed information, such as colour and texture, is lost. This is especially true in distant view or aerial images, where the objects are frequently far away and small, having low contrast against the background [3].

One of the solutions to this problem is to use specialized and improved hardware, which might get very costly. Therefore, more algorithmic solutions are focused on by the researchers.

Recent studies on low lighting concentrate on image enhancement to improve basic visual properties in the pre-processing steps [4], [5], [6]. These methods include global and local enhancement techniques. The global enhancement techniques when applied to night-time images may over-expose already bright parts of the images, however, local contrast enhancement methods focus on image details, but it increases noise when contrast gain is high [4]. However, deep learning models can produce better results to enhance image contrast. By applying a simple histogram equalization method or gamma correction, the contrast of road and vehicle headlights was increased that decreases the image quality for vehicle detection. Therefore, we applied the low light enhancement model miRNet on nighttime traffic sequences as it produces robust results and prevents over-exposure of car and road lights. Our proposed model consists of the following steps: all the extracted nighttime image sequences are first subjected to defogging and then fed into miRNet model. The enhanced images are then passed onto the YOLOv5 object detection algorithm to locate vehicles in each image frame. For each detection, SIFT features and templates are extracted, based on which IDs are assigned to each one. The extracted templates are used to find the possible matches in succeeding image frames which are filtered to get the best possible match by feature matching. In the last, the trajectories of all the tracked vehicles are drawn by plotting the centroid points.

The main contribution of our work is as follows:

- An efficient and computationally lightweight vehicle detection and tracking algorithm for night-time aerial image sequences is established.
- We used the deep learning model YOLOv5 to detect objects in night-time aerial images containing dense scenes to enhance the detection rate.
- A simple and efficient multi-vehicle tracking approach that uses SIFT features which are robust to noise, light transformation, and angle view for identifier allocation, and a template-matching model is proposed.

The night sequences from the publicly available datasets UAVDT [5] and VisDrone [6] were used to evaluate our vehicle recognition and tracking system. Our proposed model produces efficient results on both datasets as compared to other techniques.

The remainder of the paper is structured as follows: Section II presents related work. Our proposed architecture is thoroughly explained in Section III. Benchmark datasets and the experimental findings are described in Section IV. Section V presents the conclusion and suggested next steps.

II. LITERATURE REVIEW

Numerous researchers have focused on object detection in low-light conditions. In most cases, the images are first subjected to pre-processing to enhance brightness levels. Also, traffic monitoring work has been done, including vehicle detection and tracking as the core steps [7], [8], [9], [10]. Therefore, this section is divided into two categories: object detection in low-light conditions and vehicle detection and tracking methodologies.

A. OBJECT DETECTION IN LOW-LIGHT CONDITIONS

One of the most extensively used machine learning approaches for image enhancement includes histogram equalization, which is easy to implement and consumes low computational power [11]. However, due to excessive gray merging, gray levels are easily lost in γ-correction. It is predicated on the hypothesis that the sensitivity of the human eye to ambient light is exponentially related to the input light intensity. Human eyes can more easily detect changes in low illumination, but it becomes more difficult for them to perceive brightness fluctuations as illumination levels increase. Gamma adjustment increases the visibility of the contrast effect of image illumination. However, it can be difficult to automatically determine a suitable gamma value while performing image processing on the source image. Reference [12] developed an improved SSD-based low-illumination image object detection technique, and the Retinex theory-based image enhancement algorithm was used to improve the original low-illumination image. In another study [13], a real-time object recognition method for nighttime monitoring is presented. The detection algorithm is built on methods for contrast analysis. The model inputs two image frames and calculates the change between the contrast of both images to detect object masks. However, this model can only detect moving objects. Tian et al. [14] proposed a statistical modelling technique for photos with low illumination and uneven illumination based on image wavelet coefficients. Tian et al. [14] has proposed an efficient pre-processing technique that uses a dynamic function of the pixel values in spatial neighbourhoods to improve underexposed, low dynamic range videos. This method has a very high level of computational complexity.

B. VEHICLE DETECTION AND TRACKING

Several studies are focusing on vehicle detection and tracking methods [16]. In [18], an effective Gaussian Mixture Model (GMM) based image segmentation technique is applied. This technique may identify different automobiles' frontal views. The Canny edge detector and Hough transform are used for spotting lanes to determine the vehicles' driving area. This work trains the Support Vector Machine (SVM) classifier using the Histogram of Gradient (HOG) features, colours, and Haar features of automobiles to increase the efficacy of the proposed technique further. Also, to detect vehicles, an upgraded You Look Only Once version 3 (YOLOv3) algorithm is created. The data collection is first clustered using a clustering analysis approach, and the network topology is then optimized to increase the number of final output grids and improve the relatively weak vehicle

prediction capacity. In another study [20], an intelligent transport system has been proposed which uses a Kalman filter and YOLO detector. The model also generates track IDs and uses a Hungarian algorithm to retrieve them. Similarly, [21] creates a Simple, Online, real-time tracking (SORT) technique based on the Kalman filter and the Hungarian matching algorithm, using the Faster Regional Convolutional Neural Network (R-CNN) algorithm as the target detector to track multiple targets concurrently. One drawback of the SORT algorithm is that it does not incorporate appearance features. Another study [22] presents a vehicle recognition and monitoring method using the Gaussian Mixture Model (GMM) and blob extraction method. Firstly, the background was estimated and subtracted from frames to extract the foreground objects. For further noise removal, morphological corrections were employed. Tracking of the vehicle was enhanced using the GMM algorithm. Ait Abdelali et al. [23] developed a vision-based traffic monitoring model. The detection of vehicles is done using the deep learning-based YOLO detector. For tracking, a particle filter is implemented. Mou et al. [24] proposed a detection method based on segmenting the aerial image into similar regions using a Convolutional Neural Network (CNN). Then, a trained SVM classifier was used to track and classify vehicles. The training of two different classifiers increases the complexity and computational cost of the model which limits its applicability to large datasets.

In this paper, we aim to introduce a lightweight vehicle detection and tracking approach which requires limited training. Also, we have combined deep learning and machine learning techniques to increase the model efficiency.

III. THE PROPOSED FRAMEWORK

Fig. 1 depicts the general architecture of the proposed model. The model mainly consists of five modules: (i) pre-processing steps to enhance the brightness level of nighttime images; (ii) vehicle detection using the deep learning model YOLOv5; (iii) SIFT feature extraction of each detected vehicle and identifier assignment; (iv) vehicle tracking using the template matching algorithm; and (v) drawing trajectories of each tracked vehicles. Each module of the framework is explained in detail in the following subsections. a machine inspection dataset, the suggested system is tested, and the findings demonstrate that it outperforms a number of cutting-edge object recognition.

A. IMAGE PRE-PROCESSING
1) DEFOGGING
The input images extracted from nighttime traffic videos at the rate of 8 FPS were first resized to 768×768 coordinates.

To denoise the image, we applied the defogging method [25], [26] which estimated the intensity of noise in each image pixel and then removed as follows:

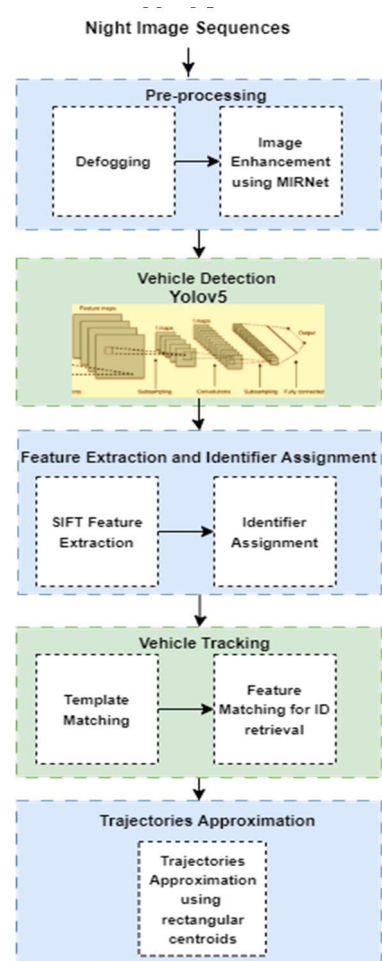$$I(x) = U(x) t(x) + K(1 - t(x)) \qquad (1)$$

FIGURE 1. The architecture of the proposed system for low-illumination conditions.

where x represents the location of the pixel, K is the density of the fog, and $t(x)$ is the transmission map [27]. The visualization of the defogging process is shown in Fig 2.

2) LOW-LIGHT ENHANCEMENT USING MIRNet
After denoising the images, the next step is to enhance the brightness level of the images to locate the objects easily. For this purpose, we used the pre-trained model MIRNet. All the images are passed onto the contrast enhancement module. MIRNet is a pre-trained fully convolutional deep learning architecture that retains spatially exact high-resolution representations over the whole network while receiving significant contextual information from the low-resolution representations [28]. The model consists of a feature extraction module that maintains the high-resolution original features to reserve fine spatial details while computing a complementary collection of features at various spatial scales [29]. Also, the characteristics from numerous multi-resolution branches are gradually integrated for better representation learning using a recurring information exchange mechanism [30]. It uses a technique for fusing features from different scales
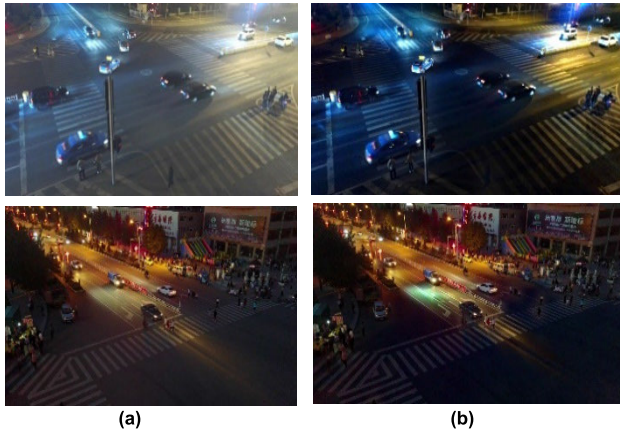
FIGURE 2. Defogging process over UAVDT and VisDrone datasets (a) original images and (b) defogged images.

FIGURE 3. Low-light image enhancement using MIRNet over UAVDT and Visdrone datasets.

that correctly maintains the original information of the feature at each spatial level while dynamically combining varying receptive fields. To simplify the learning process, the recursive residual gradually decomposes the input image, enabling deep networks to be built [31].

The output of the MIRNet image enhancement is shown in Fig. 3. Also, the overall architecture of MIRNet is seen in Fig. 4.

B. YOLO v5-BASED VEHICLE DETECTION

Because of its high-performance capabilities, YOLO algorithms are frequently used in object detection systems, especially for vehicle detection tasks. YOLO sees an image as a regression problem with fast speed [32]. While training, YOLO takes the entire image as input training, paying more attention to global information for target detection and returns the position of the object bounding box [33], [34], [35].

YOLOv5 is a single-stage detector that considerably reduces the processing time of deeper networks. As we have already processed the images to make them viable for detection, therefore, we used YOLOv5 for detection purposes to keep the system lightweight as well as efficient. Also, it performs better in small target detection [35]. Four primary parts make up the construction of YOLOv5: input, backbone, neck, and head.

1) BACKBONE
The backbone selects the important component from the input image for further analysis. YOLOv5 uses spatial pyramid pooling (SPP) and cross-stage partial networks (CSP) as its main building blocks to extract rich, significant information from input images. SPP can be used for the same object detection in multiple sizes and scales enhancing the model's generalization.

2) NECK
It consists of a path aggregation network (PANet) and the Feature Pyramid Network (FPN). The primary function of PANet is to generate feature pyramids. FPN structure improves the bottom-up path and low-level feature propagation. Also, Localization features are sent from lower feature maps to higher feature maps via the PAN framework.

3) HEAD
The output layer consists of three convolution layers to predict the location of the object bounding box and scores. YOLOv5 uses the Sigmoid Linear Unit (SiLU) activation function in hidden layers and the Sigmoid activation function is utilized in the convolution operation of the output layer calculated as follows [36]:

$$SiLU\ (x) = x \times \sigma(x) \tag{2}$$

where $\sigma(x)$ is the logistic sigmoid.

$$S\ (x) = \frac{1}{1 + e^{-x}} \tag{3}$$

where $S(x)$ is the sigmoid function.

The loss function of the overall structure is calculated as given below:

$$Loss = \lambda_1 L_{cls} + \lambda_2 L_{obj} + \lambda_3 L_{loc} \tag{4}$$

where L_{cls}, L_{obj} and L_{loc} are the classes loss, objectness loss, and location loss respectively. The data was split into 70:30 ratios for training and testing respectively. The detailed configuration of the YOLOv5 model is given in Table 1.

The architecture of the YOLOv5 algorithm is shown in Fig. 5.

The image frames were divided into bursts of five images. The detection is performed on the first image of each burst while tracking was done on the next four images. The vehicle detection result using the YOLOv5 algorithm is visualized in Fig. 6.
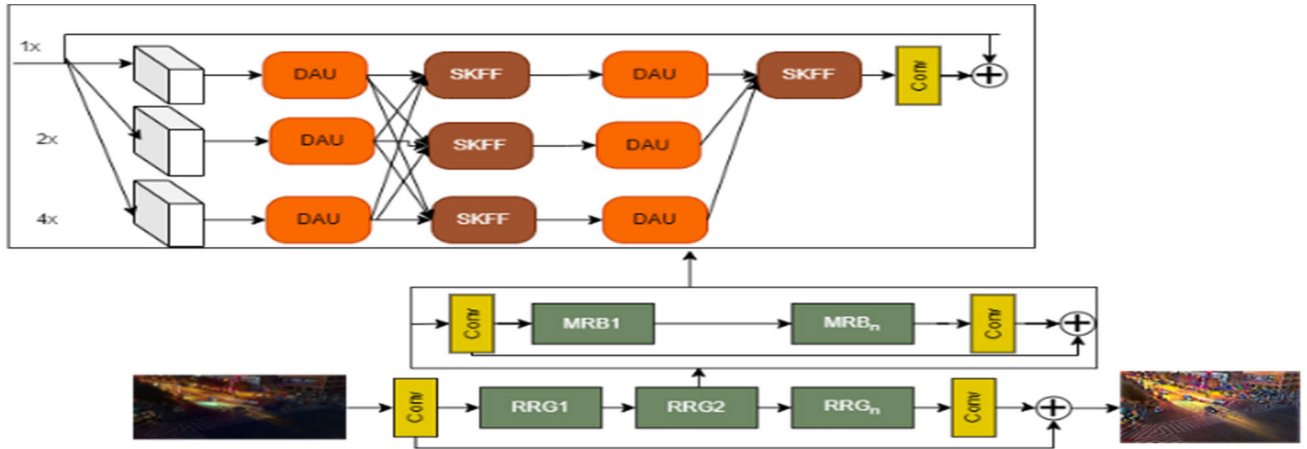
FIGURE 4. Architecture of MIRNet for image enhancement where RRG = Recursive Residual Group, MRB = Multiscale Residual Block, DAU = Dual Attention Unit, SKFF = Selective Kernel Feature Fusion.

TABLE 1. Parameter configuration for YOLOv5 algorithm.

Parameter Name	Value
Learning rate (initial)	0.0001
Epochs	60
IoU	0.4
Batch Size	5
Momentum	0.937
Input layer size	768×768
Weight decay	0.0005
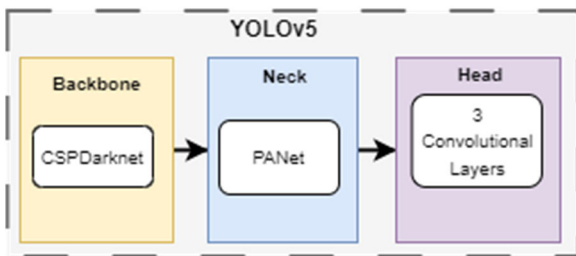
FIGURE 5. The architecture of the YOLOv5 model.

FIGURE 6. Vehicle detection using YOLOv5 over UAVDT and VisDrone datasets.

C. IDENTIFIER NUMBER ASSIGNMENT

As each image frame contains multiple vehicles to be tracked in the succeeding frames. Therefore, an identifier was required to locate each vehicle separately that should remain the same for a particular vehicle throughout the tracking. For this purpose, every detected vehicle was subjected to SIFT feature extraction [37], [38]. Based on this a unique identifier number was assigned to each car.

The SIFT features are local making them robust against occlusion and clutter [35], [36], [37]. The feature extraction algorithm consists of the following four steps.

1) SCALE SPACE

This step includes selecting potential areas in the image to find features [38], [39]. The input image is convolved with a Gaussian kernel at various scales to produce the function $L(x, y, \sigma)$, which denotes the scale space [41] of the image given as.

$$L(x, y, \sigma) = G(x, y, \sigma)*I(x, y) \quad (5)$$

where I is the input image with x, y coordinates. σ represents the scale parameter and $G(x, y, \sigma)$ denotes the Gaussian blur operator which is calculated as follows:

$$G(x, y, \sigma) = \frac{1}{2\pi(\sigma)^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (6)$$

The size of the source image determines the number of octaves and scale in scale space. The Difference of Gaussian is further used to approximate Laplacian of Gaussian which is scale invariant.

2) KEY POINT LOCALIZATION

To update the keypoint location, the Taylor series expansion of scale space is used to locate the extrema with greater accuracy, if the intensity at the extrema is less than a certain threshold it is rejected.

3) ORIENTATION ASSIGNMENT

Each keypoint was assigned an orientation to make the extracted keypoint invariant to rotation. The neighbourhood around the keypoint position is chosen depending on the scale, and the gradient's amplitude and direction are defined as follows:

$$|I| = \sqrt{I_x^2 + I_y^2} \tag{7}$$

$$\Theta = tan^{-1}\left(\frac{I_y}{I_x}\right) \tag{8}$$

where I_x and I_y are x,y coordinates of the descriptors. A 360-degree orientation histogram with 36 bins is produced.

4) KEY POINT DETECTOR

To calculate the local image descriptor a 16×16 window is taken around the keypoint which is further separated into 16 subblocks of 4×4 size.

5) KEY POINT MATCHING

The matching between two images is obtained by identifying the nearest neighbour of key points using the formula given below:

$$(u, v) = \sqrt{\sum_{i=1}^{n} (v_i - v_i)^2} \tag{9}$$

where u and v are the key points descriptors.

If the number of matches exceeds threshold value 6, then the corresponding vehicle's identifier number is retrieved and assigned to the matched vehicle in the succeeding frame. If no match is found, the vehicle is added as a new entry with a unique number. the identifier assignment to each detected vehicle is shown in Fig. 7.

D. TEMPLATE MATCHING-BASED VEHICLE TRACKING

To lower the computational complexity of the model, we used a template-matching algorithm to avoid unnecessary feature extraction. A template model has been generated for every new vehicle registered in the system [43], [44]. This generated model was used to locate all the possible locations of the vehicle in the following frame. The template matching algorithm moves the template across the entire image and a similarity score is calculated between the area covered by the window and the template [45], [46], [47], [48]. The matching is implemented through a 2-dimensional convolution:

$$l(x, y) = f(x, y)^{\circ}g(x, y) \tag{10}$$

where $f(x,y)$ is the original image frame and $g(x,y)$ is the vehicle template.

FIGURE 7. Identifier assignment to each detected vehicle based on SIFT feature extraction over UAVDT and VisDrone datasets.

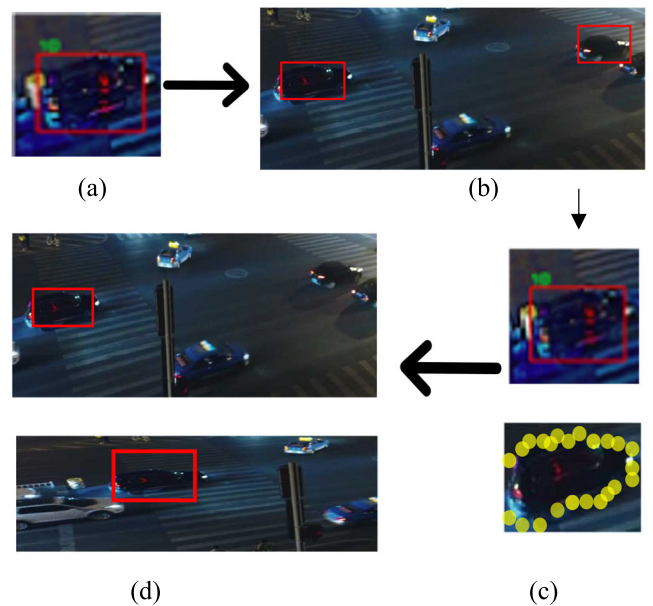(a)　　　　　　　　　(b)

(d)　　　　　　　　　(c)

FIGURE 8. Vehicle Tracking using Template Matching (a) vehicle model extracted from detection (b) number of template matchings is greater than 1 (c) SIFT feature extraction and matching with the template matches (d) best possible location retained across the image frames.

The extracted templates contain texture and appearance information which helps to find its match. If an image has more than one possible location detected, then it is subjected to SIFT feature matching to get the best match and the associated identifier number [49], [50]. Vehicle templates that are not found in the succeeding frames are retained and matched for the next 5 frames before deletion to handle the occlusion within the tracked images. The tracking results are shown in Fig. 8. The steps involved in template-based matching are given in Algorithm 1.

Algorithm 1 Vehicle Detection and Tracking

Input: vehicle detections $V = \{v1, v2, v3, \ldots \ldots vn\}$ where $vn = (x1,y1),(x2,y2)$, Input_image I,
Frames $F = \{f1, f2, \ldots \ldots, fn\}$
Output: The tracking results
1: Initilize feature_list =[], thresh =6, vehicle_model =[]
2: **for** *i in range (V)*
3: *x,y,w,h* ← *V* [*i*]
4: *ROI* = Extract Region_of_interest(*I(x,y,x+w,y+h)*)
5: *f* ← SIFT *(ROI)*
6: feature_list ← *f,i*
7: vehicle_model ← *ROI*
8: **while** F > 0
 9: **for** j in range (vehicle_model)
  10: matches = template matching(vehicle_model[j], F)
  11: **if** matches > 1 **then**
   12: *fm* = F eatureMatching(matches, feature_list)
    13: **if** *fm > thresh* **then**
    14: Retrieve and assign corresponding ID and discard other matched templates
    15: e**lse**
    16: Retrieve corresponding ID and assign to matched vehicle
     end if
   end if
  end
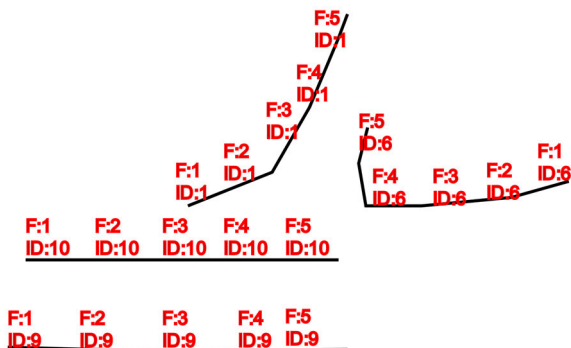
FIGURE 9. Vehicle trajectories approximation is estimated by joining the centroid of each vehicle location against the identifier number ID and Frame number.

E. TRAJECTORIES APPROXIMATION

Each tracked vehicle's path was recorded and plotted against each video frame to understand the traffic flow conditions and routes. To estimate the trajectories [51], the final match obtained from the tracking algorithm for each vehicle was recorded by calculating the rectangular centroid of each vehicle against the frame number taken as a reference for the time stamp. The centroids are calculated as:

$$rectangular - centroids_{vehicles} = (\frac{x1 + x2}{2}, \frac{y1 + y2}{2}) \quad (11)$$

The points were plotted and joined with time information incorporated as shown in Fig. 9.

IV. EXPERIMENTS AND RESULTS

The experiments were conducted using a laptop with an Intel Core i5-8550U 1.80GHz processor, 6GB of Random Access Memory (RAM), Windows 10 running on the x64

FIGURE 10. Sample frames from the VisDrone dataset.
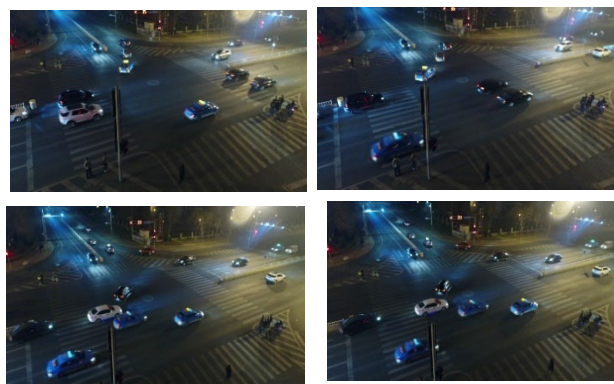
FIGURE 11. Sample frames from the UAVDT dataset.

architecture, and the Python tool. Also, to compare the performance of CPU and GPU. We ran the experiment on Tesla K80 GPU which is available free on Google Colab. The training time on the CPU was 1.3 hrs whereas it took 0.86 hrs to train on the GPU. However, there was no difference in the precision values. The proposed model produces remarkable results when tested on two benchmark datasets: UAVDT and VisDrone datasets.

A. DATASETs

1) VisDrone DATASET

The Vision Meets Drone Single Object-Tracking (VisDrone) dataset contains 288 clips of videos with a total of 261,908 frames and 10,209 still photos taken by several drones equipped with cameras and covering a variety of places. We used traffic image sequences taken at nighttime to test our model. Some of the sample images from the VisDrone dataset are displayed in Fig. 10.

2) VisDrone DATASET

The other dataset includes the Unmanned Aerial Vehicle Detection and Tracking (UAVDT) benchmark dataset. It consists of traffic sequences recorded using a UAV platform in various urban settings. Each frame is in jpg format with a 1080 × 540 pixels resolution. Sample images from the UAVDT dataset are shown in Fig. 11.

TABLE 2. Precision, Recall, and F1-score for the detection algorithm.

Datasets	Precision	Recall	F1-score
UAVDT	0.924	0.915	0.92
VisDrone	0.904	0.892	0.90

TABLE 3. Precision, Recall, and F1-score for the tracking algorithm.

Datasets	Precision	Recall	F1-score
UAVDT	0.861	0.881	0.87
VisDrone	0.833	0.830	0.83

FIGURE 12. Vehicles left undetected are marked in red circles.

B. EVALUATION OF DETECTION AND TRACKING ALGORITHM

We used three performance metrics to assess our proposed detection and tracking algorithm specially designed for low illumination conditions: Precision, Recall and F1-score. These parameters are calculated as follows:

$$Precision = \frac{TruePositive}{(TruePositive + FalsePositive)} \quad (12)$$

$$Recal = \frac{TruePositive}{(TruePositive + FalseNegative)} \quad (13)$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (14)$$

Table 2 shows the evaluation of the YOLOv5-based detection algorithm. Whereas the tracking algorithm evaluation is given in Table 3.

C. COMPARISION WITH OTHER METHODS

We compared our proposed model with other methods in terms of precision score. Our model outperforms other techniques for both vehicle detection and tracking. Table 4 demonstrates the contrast of our proposed detection model with other methodologies.

Table 5 shows the comparison of our proposed tracking algorithm with other methodologies. It can be seen that our model produces efficient results.

A comparison of detection and tracking techniques with state-of-the-art techniques has been demonstrated in Tables 6 and 7.

TABLE 4. Comparison of detection algorithm with other methods.

Datasets	Methods	Precision
UAVDT	YOLOv3	0.776
	YOLOv4	0.823
	YOLOv5 (nano)	0.920
	YOLOv5 (medium)	0.922
	YOLOv5	**0.924**
VisDrone	YOLOv3	0.760
	YOLOv4	0.842
	YOLOv5 (nano)	0.903
	YOLOv5 (medium)	0.904
	YOLOv5	**0.904**

TABLE 5. Comparison of tracking algorithm with other methods.

Datasets	Methods	Precision	Tracking speed(seconds/frame)
UAVDT	HOG+Template Matching	0.68	160
	SURF+Template Matching	0.75	
	SIFT+Template matching	**0.861**	**214**
VisDrone	HOG+Template Matching	0.70	173
	SURF+Template Matching	0.77	226
	SIFT+Template matching	**0.833**	**231**

TABLE 6. Comparison of detection algorithm with state-of-the-art techniques.

Datasets	Methods	Precision
UAVDT	NDFT [52]	0.520
	FPN [52]	0.490
	Our Model [nighttime sequences]	**0.924**
VisDrone	GA-FPN [53]	0.82
	MRCN [54]	0.81
	Our Model [nighttime sequences]	**0.904**

V. LIMITATIONS

The proposed method performs well for nighttime surveillance of road traffic. However, there are still some limitations of the model. The system can detect vehicles in case of partial occlusion or cluttering, but a separate method is required to eliminate the full occlusion or background cluttering problem due to low contrast as shown in Fig. 12. Moreover, the model does not take into account the pedestrians, bicycles or bikes.

TABLE 7. Comparison of tracking algorithm with state-of-the-art techniques.

Datasets	Methods	Precision
UAVDT	ASRDCT [55]	0.762
	Our Model [nighttime sequences]	**0.861**
VisDrone	Affinity Network [56]	0.74
	Our Model [nighttime sequences]	**0.833**

Moreover, diverse weather conditions such as images taken in cloudy, foggy or rainy weather require other pre-processing methodologies which are beyond the scope of our model.

VI. CONCLUSION AND FUTURE WORK

In this study, we propose a lightweight and efficient vehicle detection and tracking algorithm specially designed for low-illumination conditions. First of all, we pre-processed the nighttime traffic scenes to adjust the brightness level of the image. Then, we applied semantic segmentation based on FCM clustering to segment the image into multiple uniform regions to reduce the overall complexity. For detection, we used YOLOv5 which can detect small objects precisely. We assign identifiers based on SIFT features to track multiple vehicles within a single image frame. Then, template matching was employed to get each vehicle's possible location and its corresponding identifier was retrieved by SIFT feature matching. The evaluation experimentation on public datasets demonstrates that our proposed framework can efficiently detect and track automobiles and outperforms other methods. In the future, we aim to enhance vehicle monitoring techniques to adapt to more complex traffic scenarios.

REFERENCES

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

[2] Y. Chen, N. Li, D. Zhu, C. C. Zhou, Z. Hu, Y. Bai, and J. Yan, "BEVSOC: Self-supervised contrastive learning for calibration-free BEV 3-D object detection," *IEEE Internet Things J.*, vol. 11, no. 12, pp. 22167–22182, Jun. 2024, doi: 10.1109/jiot.2024.3379471.

[3] J. Li, C. Zhang, Z. Liu, R. Hong, and H. Hu, "Optimal volumetric video streaming with hybrid saliency based tiling," *IEEE Trans. Multimedia*, vol. 25, pp. 2939–2953, 2022, doi: 10.1109/TMM.2022.3153208.

[4] L. Toloşi and T. Lengauer, "Classification with correlated features: Unreliability of feature ranking and solutions," *Bioinformatics*, vol. 27, no. 14, pp. 1986–1994, Jul. 2011.

[5] J. Li, L. Han, C. Zhang, Q. Li, and Z. Liu, "Spherical convolution empowered viewport prediction in 360 video multicast with limited FoV feedback," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 19, no. 1, pp. 1–23, Jan. 2023, doi: 10.1145/3511603.

[6] R. Zhang, L. Li, Q. Zhang, J. Zhang, L. Xu, B. Zhang, and B. Wang, "Differential feature awareness network within antagonistic learning for infrared-visible object detection," *IEEE Trans. Circuits Syst. Video Technol.*, p. 1, 2023, doi: 10.1109/TCSVT.2023.3289142.

[7] D. Li, "An interactive teaching evaluation system for preschool education in universities based on machine learning algorithm," *Comput. Hum. Behav.*, vol. 157, Aug. 2024, Art. no. 108211, doi: 10.1016/j.chb.2024.108211.

[8] A. M. Qureshi and A. Jalal, "Vehicle detection and tracking using Kalman filter over aerial images," in *Proc. 4th Int. Conf. Advancement Comput. Sci. (ICACS)*, Feb. 2023, pp. 1–6, doi: 10.1109/ICACS55311.2023.10089701.

[9] D. Cheng, L. Chen, C. Lv, L. Guo, and Q. Kou, "Light-guided and cross-fusion U-Net for anti-illumination image super-resolution," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 12, pp. 8436–8449, Dec. 2022, doi: 10.1109/TCSVT.2022.3194169.

[10] H. Liu, H. Yuan, Q. Liu, J. Hou, H. Zeng, and S. Kwong, "A hybrid compression framework for color attributes of static 3D point clouds," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 3, pp. 1564–1577, Apr. 2021, doi: 10.1109/TCSVT.2021.3069838.

[11] L. Zhou, X. Sun, C. Zhang, L. Cao, and Y. Li, "LiDAR-based 3-D glass detection and reconstruction in indoor environment," *IEEE Trans. Instrum. Meas.*, vol. 73, pp. 1–11, 2024, doi: 10.1109/TIM.2024.3375965.

[12] J. Chen, Q. Wang, W. Peng, H. Xu, X. Li, and W. Xu, "Disparity-based multiscale fusion network for transportation detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 18855–18863, Oct. 2022, doi: 10.1109/TITS.2022.3161977.

[13] C. Fu, H. Yuan, H. Xu, H. Zhang, and L. Shen, "TMSO-Net: Texture adaptive multi-scale observation for light field image depth estimation," *J. Vis. Commun. Image Represent.*, vol. 90, Feb. 2023, Art. no. 103731, doi: 10.1016/j.jvcir.2022.103731.

[14] T. Guo, H. Yuan, L. Wang, and T. Wang, "Rate-distortion optimized quantization for geometry-based point cloud compression," *J. Electron. Imag.*, vol. 32, no. 1, Feb. 2023, Art. no. 013047, doi: 10.1117/1.jei.32.1.013047.

[15] E. P. Bennett and L. Mcmillan, "Video enhancement using per-pixel virtual exposures," in *Proc. ACM SIGGRAPH Papers*, Jul. 2005, pp. 845–852, doi: 10.1145/1186822.1073272.

[16] J. Xing, H. Yuan, R. Hamzaoui, H. Liu, and J. Hou, "GQE-Net: A graph-based quality enhancement network for point cloud color attribute," *IEEE Trans. Image Process.*, vol. 32, pp. 6303–6317, 2023, doi: 10.1109/tip.2023.3330086.

[17] J. Shen, N. Liu, and H. Sun, "Vehicle detection in aerial images based on lightweight deep convolutional network," *IET Image Process.*, vol. 15, no. 2, pp. 479–491, Feb. 2021, doi: 10.1049/ipr2.12038.

[18] K. Uesugi, H. Mayama, and K. Morishima, "Analysis of rowing force of the water strider middle leg by direct measurement using a bio-appropriating probe and by indirect measurement using image analysis," *Cyborg Bionic Syst.*, vol. 4, p. 61, Jan. 2023, doi: 10.34133/cbsystems.0061.

[19] A. Hbaieb, J. Rezgui, and L. Chaari, "Pedestrian detection for autonomous driving within cooperative communication system," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2019, pp. 1–6, doi: 10.1109/WCNC.2019.8886037.

[20] D. Ma, H. Fang, N. Wang, H. Lu, J. Matthews, and C. Zhang, "Transformer-optimized generation, detection, and tracking network for images with drainage pipeline defects," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 38, no. 15, pp. 2109–2127, Oct. 2023, doi: 10.1111/mice.12970.

[21] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 3464–3468, doi: 10.1109/ICIP.2016.7533003.

[22] X. Zhao, Y. Fang, H. Min, X. Wu, W. Wang, and R. Teixeira, "Potential sources of sensor data anomalies for autonomous vehicles: An overview from road vehicle safety perspective," *Expert Syst. Appl.*, vol. 236, Feb. 2024, Art. no. 121358, doi: 10.1016/j.eswa.2023.121358.

[23] H. A. Abdelali, O. Bourja, R. Haouari, H. Derrouz, Y. Zennayi, F. Bourzex, and O. H. T. Rachid, "Visual vehicle tracking via deep learning and particle filter," in *Advances on Smart and Soft Computing*, vol. 1188. Singapore: Springer, 2021, pp. 517–526, doi: 10.1007/978-981-15-6048-4_45.

[24] J. Mou, K. Gao, P. Duan, J. Li, A. Garg, and R. Sharma, "A machine learning approach for energy-efficient intelligent transportation scheduling problem in a real-world dynamic circumstances," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–13, 2022, doi: 10.1109/TITS.2022.3183215.

[25] Y. Xu, J. Wen, L. Fei, and Z. Zhang, "Review of video and image defogging algorithms and related studies on image restoration and enhancement," *IEEE Access*, vol. 4, pp. 165–188, 2016, doi: 10.1109/ACCESS.2015.2511558.

[26] W. Gao, M. Wei, and S. Huang, "Optimization of aerodynamic drag reduction for vehicles with non-smooth surfaces and research on aerodynamic characteristics under crosswind," *Proc. Inst. Mech. Eng., D, J. Automobile Eng.*, vol. 2023, May 2023, Art. no. 095440702311734, doi: 10.1177/09544070231173471.

[27] W. Liu, F. Zhou, T. Lu, J. Duan, and G. Qiu, "Image defogging quality assessment: Real-world database and method," *IEEE Trans. Image Process.*, vol. 30, pp. 176–190, 2021, doi: 10.1109/TIP.2020.3033402.

[28] L. Yin, W.-T. Pan, J. Kuang, and M. Zhuang, "Application of bootstrap-DEA with fuzzy computing in performance evaluation of forklift leasing supplier," *IEEE Access*, vol. 8, pp. 66095–66104, 2020, doi: 10.1109/ACCESS.2019.2959598.

[29] C. Lu, J. Zheng, L. Yin, and R. Wang, "An improved iterated greedy algorithm for the distributed hybrid flowshop scheduling problem," *Eng. Optim.*, vol. 56, no. 5, pp. 792–810, May 2024, doi: 10.1080/0305215x.2023.2198768.

[30] K. Zhang, C. Yuan, J. Li, X. Gao, and M. Li, "Multi-branch and progressive network for low-light image enhancement," *IEEE Trans. Image Process.*, vol. 32, pp. 2295–2308, 2023, doi: 10.1109/TIP.2023.3266171.

[31] Z. W. Deng, Y. Q. Zhao, B. H. Wang, W. Gao, and X. Kong, "A preview driver model based on sliding-mode and fuzzy control for articulated heavy vehicle," *Meccanica*, vol. 57, no. 8, pp. 1853–1878, 2022.

[32] M. A. A. Al-qaness, A. A. Abbasi, H. Fan, R. A. Ibrahim, S. H. Alsamhi, and A. Hawbani, "An improved YOLO-based road traffic monitoring system," *Computing*, vol. 103, no. 2, pp. 211–230, Feb. 2021, doi: 10.1007/s00607-020-00869-8.

[33] R. Luo, Z. Peng, J. Hu, and B. K. Ghosh, "Adaptive optimal control of affine nonlinear systems via identifier–critic neural network approximation with relaxed PE conditions," *Neural Netw.*, vol. 167, pp. 588–600, Oct. 2023, doi: 10.1016/j.neunet.2023.08.044.

[34] D. Li, X. Dai, J. Wang, Q. Xu, Y. Wang, T. Fu, A. Hafez, and J. Grant, "Evaluation of college students' classroom learning effect based on the neural network algorithm," *Mobile Inf. Syst.*, vol. 2022, pp. 1–8, Oct. 2022, doi: 10.1155/2022/7772620.

[35] Y. Xu, E. Wang, Y. Yang, and Y. Chang, "A unified collaborative representation learning for neural-network based recommender systems," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 11, pp. 5126–5139, Nov. 2022, doi: 10.1109/TKDE.2021.3054782.

[36] J.-H. Kim, N. Kim, Y. W. Park, and C. S. Won, "Object detection and classification based on YOLO-V5 with improved maritime dataset," *J. Mar. Sci. Eng.*, vol. 10, no. 3, p. 377, Mar. 2022, doi: 10.3390/jmse10030377.

[37] P. Zhou, J. Qi, A. Duan, S. Huo, Z. Wu, and D. Navarro-Alarcon, "Imitating tool-based garment folding from a single visual observation using hand-object graph dynamics," *IEEE Trans. Ind. Informat.*, vol. 20, no. 4, pp. 6245–6256, Apr. 2024, doi: 10.1109/tii.2023.3342895.

[38] Y. Hua, J. Lin, and C. Lin, "An improved SIFT feature matching algorithm," in *Proc. World Congr. Intell. Control Autom. (WCICA)*, 2010, pp. 6109–6113, doi: 10.1109/WCICA.2010.5554659.

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

[40] J. Peng, N. Wang, A. A. A. El-Latif, Q. Li, and X. Niu, "Finger-vein verification using Gabor filter and SIFT feature matching," in *Proc. 8th Int. Conf. Intell. Inf. Hiding Multimedia Signal Process.*, Jul. 2012, pp. 45–48, doi: 10.1109/IIH-MSP.2012.17.

[41] D. Cai, R. Li, Z. Hu, J. Lu, S. Li, and Y. Zhao, "A comprehensive overview of core modules in visual SLAM framework," *Neurocomputing*, vol. 590, Jul. 2024, Art. no. 127760, doi: 10.1016/j.neucom.2024.127760.

[42] Y. Gu, Z. Hu, Y. Zhao, J. Liao, and W. Zhang, "MFGTN: A multi-modal fast gated transformer for identifying single trawl marine fishing vessel," *Ocean Eng.*, vol. 303, Jul. 2024, Art. no. 117711, doi: 10.1016/j.oceaneng.2024.117711.

[43] S. Li, J. Chen, W. Peng, X. Shi, and W. Bu, "A vehicle detection method based on disparity segmentation," *Multimedia Tools Appl.*, vol. 82, no. 13, pp. 19643–19655, May 2023, doi: 10.1007/s11042-023-14360-x.

[44] J. Chen, Y. Song, D. Li, X. Lin, S. Zhou, and W. Xu, "Specular removal of industrial metal objects without changing lighting configuration," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 3, pp. 3144–3153, May 2023, doi: 10.1109/TII.2023.3297613.

[45] H. Xu, Q. Li, and J. Chen, "Highlight removal from a single grayscale image using attentive GAN," *Appl. Artif. Intell.*, vol. 36, no. 1, Dec. 2022, Art. no. 1988441, doi: 10.1080/08839514.2021.1988441.

[46] H. Zhu, D. Xu, Y. Huang, Z. Jin, W. Ding, J. Tong, and G. Chong, "Graph structure enhanced pre-training language model for knowledge graph completion," *IEEE Trans. Emerg. Topics Comput. Intell.*, pp. 1–12, 2024, doi: 10.1109/tetci.2024.3372442.

[47] D. Li, "Machine learning based preschool education quality assessment system," *Mobile Inf. Syst.*, vol. 2022, pp. 1–8, Oct. 2022, doi: 10.1155/2022/2862518.

[48] X. Kong, Q. Chen, G. Gu, K. Ren, W. Qian, and Z. Liu, "Particle filter-based vehicle tracking via HOG features after image stabilisation in intelligent drive system," *IET Intell. Transp. Syst.*, vol. 13, no. 6, pp. 942–949, Jun. 2019, doi: 10.1049/iet-its.2018.5334.

[49] W. Zheng, S. Lu, Y. Yang, Z. Yin, and L. Yin, "Lightweight transformer image feature extraction network," *PeerJ Comput. Sci.*, vol. 10, p. e1755, Jan. 2024, doi: 10.7717/peerj-cs.1755.

[50] F. Song, Y. Liu, D. Shen, L. Li, and J. Tan, "Learning control for motion coordination in wafer scanners: Toward gain adaptation," *IEEE Trans. Ind. Electron.*, vol. 69, no. 12, pp. 13428–13438, Dec. 2022, doi: 10.1109/tie.2022.3142428.

[51] C. Mi, Y. Liu, Y. Zhang, J. Wang, Y. Feng, and Z. Zhang, "A vision-based displacement measurement system for foundation pit," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–15, 2023, doi: 10.1109/tim.2023.3311069.

[52] J. Cao, H. Cholakkal, R. M. Anwer, F. S. Khan, Y. Pang, and L. Shao, "D^{2}Det: Towards high quality object detection and instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11482–11491, doi: 10.1109/CVPR42600.2020.01150.

[53] Z. Wang, "CODAN: Counting-driven attention network for vehicle," in *Proc. 28th ACM Int. Conf. Multimedia*, 2020, pp. 73–82.

[54] W. Li, Z. Wang, X. Wu, J. Zhang, Q. Peng, and H. Li, "CODAN: Counting-driven attention network for vehicle detection in congested scenes," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 73–82, doi: 10.1145/3394171.3413945.

[55] Z. Chen, L. Liu, and Z. Yu. *ASRDCT: Automatic Spatial–Temporal Relevance Regularization With Derailment Distractor Constraint for UAV Object Tracking*. Accessed: May 10, 2024. [Online]. Available: https://ssrn.com/abstrac

[56] J. Leitloff, D. Rosenbaum, F. Kurz, O. Meynberg, and P. Reinartz, "An operational system for estimating road traffic information from aerial images," *Remote Sens.*, vol. 6, no. 11, pp. 11315–11341, Nov. 2014, doi: 10.3390/rs61111315.

NOUF ABDULLAH ALMUJALLY received the Ph.D. degree in computer science from the University of Warwick, U.K. She is currently an Assistant Professor of computer science with the Department of Information Systems, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University (PNU), Riyadh, Saudi Arabia. Her research interests include human–computer interaction (HCI), artificial intelligence (AI), machine learning, deep learning, and computer-based applications.

ASIFA MEHMOOD QURESHI received the M.S. degree in computer science from Air University, Islamabad. She is currently a Research Assistant with the Intelligent Media Center, Faculty of Computer Science and AI, Air University, Islamabad. Her research interests include machine learning, deep learning, camera and sensor-based gesture recognition, and virtual reality.

ABDULWAHAB ALAZEB received the B.S. degree in computer science from King Khalid University, Abha, Saudi Arabia, in 2007, the M.S. degree in computer science from the Department of Computer Science, University of Colorado Denver, USA, in 2014, and the Ph.D. degree in cybersecurity from the University of Arkansas, USA, in 2021. He is currently an Assistant Professor with the Department of Computer Science and Information Systems, Najran University. His research interests include cybersecurity, cloud and edge computing security, machine learning, and the Internet of Things. He received the Graduate Certificate from the University of Arkansas.

HAMEEDUR RAHMAN received the Ph.D. degree in computer vision. He is currently the Chair of the Department of Computer Games Development, Air University, Islamabad, boasts a robust research profile. He has expertise in augmented reality, virtual reality, and image processing. He demonstrates a commitment to cutting-edge technology. As a Pivotal Member, he has contributed to AI/Data Science, Cybersecurity, Computer Science, and Gaming Departments, mentoring students, and fostering interdisciplinary research, since 2018. Notably, his leadership in the Ignite (Pakistan) Project showcases practical applications of his research, emphasizing his dedication to knowledge dissemination and skill development in emerging fields.

TOUSEEF SADIQ (Graduate Student Member, IEEE) received the B.E. degree in computer engineering from Bahria University, Islamabad, Pakistan, and the M.S. degree in communications and computer networks engineering from the Polytechnic University of Turin, Italy. He is currently pursuing the Ph.D. degree with the University of Agder, Norway. His research interests include deep multimodal learning for descriptive object identification in urban environments, machine learning, computer vision, deep multimodal learning, and their applications.

MOHAMMED ALONAZI received the B.Sc. degree in computer science from King Saud University, Saudi Arabia, in 2008, the M.Sc. degree in computer science from Florida Institute of Technology, Melbourne, USA, in 2015, and the Ph.D. degree in informatics from the University of Sussex, U.K., in 2019. Currently, he is an Assistant Professor with the Department of Information Systems, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia. His research interests include human–computer interaction, UX/UI, digital transformation, cyber security, and machine learning.

ASAAD ALGARNI received the Ph.D. degree in software engineering from North Dakota State University, USA. He is currently an Assistant Professor with the Department of Computer Sciences, College of Computing and Information Technology, Northern Borders University, Saudi Arabia. His research interests include software engineering, computer vision applications, and machine learning.

AHMAD JALAL received the Ph.D. degree from the Department of Biomedical Engineering, Kyung Hee University, Republic of Korea. He was a Postdoctoral Research Fellow with POSTECH. He is currently an Associate Professor with the Department of Computer Science and Engineering, Air University, Pakistan. His research interests include multimedia content and artificial intelligence.

● ● ●