Sensors and Materials, Vol. 38, No. 11 (2026) 6539–6556
MYU Tokyo

S & M 4694

Sensor Signal Processing and System Integration Framework for Scale-invariant Vehicle Detection in Aerial Urban Imagery for Intelligent Automation Systems

Abdulwahab Alazeb,[1,2] Fatimah Alhayan,[3] Muhammad Waqas Ahmed,[4] Hadeel Alsolai,[3] Bayan Alabdullah,[3] Mohammed Alonazi,[5] and Ahmad Jalal[6,7*]

[1]Department of Computer Science, College of Computer Science and Information System, Najran University, Najran, Saudi Arabia
[2]Science and Engineering Research Center, Najran University, Najran, Saudi Arabia
[3]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia
[4]Department of Computer Games Development, Air University, E-9, Islamabad 44000, Pakistan
[5]Department of Information Systems, College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al-Kharj, Saudi Arabia
[6]Department of Computer Science, Air University, Islamabad 44000, Pakistan
[7]Department of Computer Science and Engineering, College of Informatics, Korea University, Seoul 02841, South Korea;

(Received May 13, 2026; accepted August 10, 2026)

Keywords: sensor signal processing, system integration, vehicle detection, aerial imagery, smart cities, EfficientNet, DenseNet-121, remote sensing, intelligent transportation systems

The increasing use of aerial sensing systems in intelligent transportation and automation environments has enabled the large-scale acquisition of urban imagery for vehicle monitoring and analysis. However, reliable vehicle detection using aerial sensor data remains challenging owing to degraded signal quality and complex scene structures in urban environments. To address these issues, we propose a sensor signal processing and automation framework for robust vehicle detection and classification in aerial imagery. The framework begins with Wiener filtering for noise suppression and image enhancement, followed by Markov random field-based segmentation to improve spatial consistency and object delineation. For feature representation, the EfficientNet deep learning model is employed to extract discriminative features with computational efficiency suitable for automation systems. In addition, a hybrid feature extraction strategy combining Haar wavelet transform-based scale-invariant representations with the histogram of oriented gradients is introduced to capture complementary structural and texture information from aerial sensor signals. The resulting features are classified using the DenseNet-121 model to improve feature propagation and classification performance within an integrated processing pipeline. Experimental results on the Vehicle Aerial Imagery and Roundabout Aerial Images datasets demonstrate the effectiveness of the proposed framework, achieving precision scores of 97.6 and 95.1%, respectively. Overall, the proposed framework provides an effective solution for sensor signal collection, processing, and system integration in intelligent transportation systems and smart city automation applications.

*Corresponding author: e-mail: ahmadjalal@mail.au.edu.pk
https://doi.org/10.18494/SAM6434

ISSN 0914-4935 © MYU K.K.

1. Introduction

The rapid increase in vehicle production globally highlights the importance of advanced vehicle classification systems within intelligent transportation systems (ITSs). These systems are particularly important for optimizing traffic flow, implementing automated toll systems, and improving detection for self-driving cars. Traditional methods that include laser and loop induction sensors have been employed for vehicle classification, but they often suffer from limitations in accuracy owing to environmental factors and road surface conditions.[1,2]

Several computer vision and machine learning techniques have enabled the formulation of image-processing-based vehicle classification systems. These systems rely on a two-step approach: feature extraction and classification. Visual data are processed to create handcrafted features in order to capture the most important characteristics, and then vehicle categories are classified using machine learning classifiers. While these methods can effectively implement real-time solutions, this approach has issues linked to the scalability of small datasets and accuracy that largely relies on subject-matter expertise.[3]

Currently, the advancement of deep learning methods has contributed to the ability to categorize vehicles using better algorithms and improved flexibility in the field. Among them, convolutional neural networks (CNNs) have shown the best practicability owing to the complexity of their network structures in dealing with large-scale visual datasets. This has also been improved with the help of evolution in GPU technology, facilitating the handling of high-dimensional visual data.[4] However, a great deal of labeled data is used and there is no unified benchmark dataset, which creates difficulties in achieving consistent performance under different conditions.[5] The limitations of the current vehicle identification systems discussed here can be addressed by using deep learning techniques and generating large and diverse sample sets of different vehicle models. These advancements are aimed at enhancing the efficiency, flexibility, and relevance of ITSs across the complex and changing dynamics of the real-world environment.[6]

The key contributions of our paper are as follows.

1. We propose a novel feature extraction method that integrates the histogram of oriented gradients (HOG) with the Haar wavelet transform (HWT). This approach performs multi-level Haar decomposition to isolate spatial features and reduce noise, followed by computing HOG descriptors for high-frequency components. The resulting combined feature vector, enriched with gradient-based texture and multi-resolution frequency information, enhances robustness, scale invariance, and efficiency in vehicle detection within aerial imagery.
2. We design a comprehensive vehicle detection and classification framework by leveraging state-of-the-art deep learning models, including EfficientNet for precise detection and DenseNet-121 for accurate classification, ensuring superior performance in terms of both accuracy and computational efficiency.

The rest of the paper is structured as follows. Section 2 of this paper provides a review of the literature on target detection. The selected methodology is explained in detail in Sect. 3, and its benefits and drawbacks are briefly discussed. In Sect. 4, the experimental environment along

with results is explained. Finally, Sect. 5 concludes the article with a summary of the key findings and suggestions for directions in future research and development.

2. Related Work

The detection of vehicles in aerial images has been a subject of research for decades, mainly for traffic control, parking management, and security applications. Unlike detecting vehicles from close-range or ground viewpoints, aerial vehicle detection presents unique challenges. The targets are smaller and less distinct than the environment, and it is more difficult to detect them accurately. Furthermore, the images are of poor quality because of the long-range acquisition in aerial environments. In the years that have passed, improvements have been made in a number of techniques, not only for the detection of objects on the ground. This has enabled the computer vision community to create and improve vehicle detection algorithms for airborne imagery. [7,8]

Deep feature-based systems have become a more effective solution to overcome the drawbacks of handcrafted feature-based classifiers. A CNN-based semi-supervised classification method for real-time vehicle classification extracts relevant vehicle features using a sparse-Laplacian filter, and a softmax layer is trained to calculate the probability of each vehicle class. This method enhances the accuracy and adaptability of vehicle detection in dynamic and diverse environments.[9] To solve the multiscale vehicle detection problem, another novel deep learning method was proposed, which was based on a region-based CNN and integrated with a feature pyramid network (FPN). The FPN allowed the model to leverage features at different scales, which helped in improving the detection of small vehicles in complex urban environments.[10] A novel multimodal fusion framework for vehicle detection in aerial imagery was based on RGB images and thermal IR data. This method employed a two-stream CNN architecture, where one stream was used for RGB data and the other for IR data. A fusion of features from both modalities was achieved by using an attention mechanism where the contribution of each modality was dynamically adjusted in accordance with the input characteristics.[11] The use of attention mechanisms to enhance vehicle detection in complex aerial scenes has also been explored. The attention-guided network was designed to comprise spatial and channel-wise attention modules that focus on the relevant features and suppress the background noise. The authors showed that this method led to a substantial increase in detection accuracy, particularly in cluttered background and partial-occlusion cases.[12] A multi-order descriptor that served to extract vehicle features from aerial images is also explored for classification. By combining superpixel segmentation with patch orientation, their approach outperforms high-resolution images compared with basic approaches, such as histogram of oriented gradients, support vector machine, and local binary patterns with partial least squares, and standard sparse representation methods. Their method also improves the reliability and precision of recognizing vehicles in a cluttered aerial environment.[13] The authors considered integrating LiDAR point cloud data with RGB images for the accurate and reliable detection of vehicles on roads in urban centers. In their method, they used a voxelization process to convert the 3D PC data into a format more suitable for 2D convolution operation. The voxelized LiDAR features were again combined with

RGB features in a cross-modality attention module where the methods are built to incorporate information from both sensors.[14] Cao *et al.*[15] proposed a deep learning framework for vehicle detection in remote sensing images, which was based on multitask learning. Their method integrates multiscale feature fusion, hard example mining, homography augmentation, and super-resolution to improve the detection of small vehicles in aerial imagery. The framework optimizes the image enhancement and vehicle detection process jointly, which enables the features of the image to be represented more completely and improves the accuracy of vehicle detection from the challenging remote sensing data. Cao *et al.* [16] presented a weakly supervised vehicle detection system for satellite images, which only needs region-level annotations instead of manually labelled vehicle bounding boxes. Their approach involves a density estimation method to create vehicle density maps and a multi-instance support vector machine (MI-SVM) for vehicle localization and detection. This approach significantly reduces annotation effort while maintaining competitive detection performance across images with different resolutions and imaging conditions.

3. Data, Materials, and Methods

We present an advanced GeoAI-based system for vehicle detection and classification in aerial urban images for intelligent transportation monitoring and smart city sensing applications. The proposed methodology combines several deep learning and image processing methods to facilitate the comprehensive analysis of aerial sensor data in complex urban environments. It starts with the preprocessing of aerial images by applying a Wiener filter to reduce noise and improve the quality of the image for subsequent analysis. Then, image segmentation is carried out by the Markov random field (MRF) approach, which allows for precise object delineation and spatial region refinement in heterogeneous urban scenes. The proposed framework uses EfficientNet for vehicle detection because it is a well-balanced model that is efficient in computation and has good detection performance, making it suitable for large-scale geospatial sensing environments. A hybrid feature extraction approach comprising Haar transform and HOG descriptors is employed to extract discriminative structural and texture features from aerial images to improve feature representation. Lastly, multivehicle classification is performed by DenseNet-121, a densely connected deep network that enhances feature propagation and classification ability. The performance of the proposed integrated framework is assessed by segmentation accuracy, detection accuracy, precision, and classification performance metrics. The proposed system is shown to be suitable for smart city applications based on GeoAI, such as urban monitoring, traffic analysis, and multisource aerial sensing. The proposed system is shown in the overall architectural framework in Fig. 1.

3.1 Image preprocessing

In image processing, using the Wiener filter is an important preprocessing method that aims to minimize noise while retaining the important information in the image. Its effectiveness is particularly notable when the characteristics of the noise are known or can be accurately

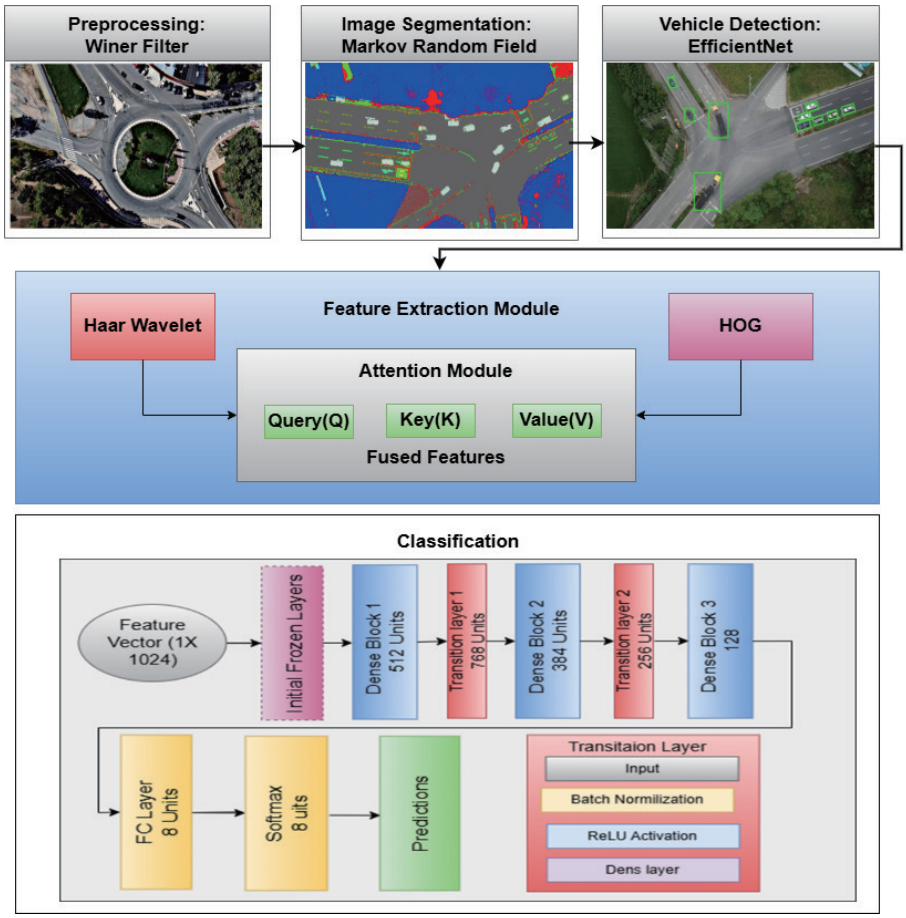

Fig. 1. (Color online) Architectural framework of our proposed model for vehicle classification.

estimated. The Wiener filter is a filter that minimizes the mean square error between the estimated and actual signals.[17] Its formulation in the frequency domain is expressed as

$$g(x,y) = \frac{|F(u,v)|^2 \cdot H(u,v)}{|F(u,v)|^2 + Sn/Sf} F(u,v), \tag{1}$$

where $H(u,v)$ is the frequency response of the filter and $F(u,v)$ is the Fourier transform of the noisy image. Sn and Sf represent the noise and signal power spectra, respectively.

3.2 Segmentation

Two segmentation methods, dynamic adaptive Gaussian mixture model (DAGMM) and MRF, were used to find the most appropriate segmentation method for aerial vehicle images. Both methods were tested to determine their effectiveness in accurately segmenting the regions

of the vehicle while maintaining the boundaries of the objects and reducing background interference. The segmentation performance of each method was quantitatively evaluated, and the method with the best performance was then applied in the preprocessing stage for the proposed vehicle classification framework.

3.2.1 DAGMM for segmentation

The first segmentation technique we applied is the modified Gaussian mixture model (GMM) known as DAGMM for segmentation. The main difference between the two is that DAGMM is more adaptable and dynamic in nature. Both methods incorporate dynamic covariance updates, adaptive component management, and adaptive learning rates. However, DAGMM further integrates spatiotemporal modeling, outlier detection, and regularization, making it more suitable for segmenting complex and dynamic data distributions. In GMM, covariance matrices are static for each iteration, whereas, in DAGMM, these covariance matrices are recomputed on the basis of the distribution of data around each Gaussian distribution. This dynamic adaptation enables the algorithm to identify changes in shapes and orientations of data clusters more effectively. The major difference of DAGMM from traditional approaches is that the dynamics in the parameters of the Gaussian components allow changes during the segmentation.[19] The probability density function for DAGMM can be expressed as

$$P(x)=\sum_{k=1}^{L}\pi_k(t)\cdot\frac{1}{\sqrt{2\pi\sigma_k^2(t)}}\cdot\exp\left(-\frac{(x-\mu_k)^2}{2\sigma_k^2}\right), \tag{2}$$

where π_k is the mixing coefficient of the kth Gaussian component, μ_k is the mean, and σ_k is the standard deviation of the kth component. The modification is that the covariance is adapted in real time in accordance with the local environment of the Gaussian mixture model. The components are added when new clusters appear and removed when clusters are no longer significant, which helps to achieve better segmentation. The components are added on the basis of the Mahalanobis distance of a data point to the nearest component, which is estimated as

$$D_M(x)=\sqrt{(x-\mu)^T\Sigma^{-1}(x-\mu)}. \tag{3}$$

Figure 2 shows representative segmentation results obtained using the proposed DAGMM. The original aerial images are shown on the left, while the corresponding segmentation maps, where different colors represent different segmented regions, are shown on the right. The results demonstrate that DAGMM effectively separates major land-cover regions, including roads, vegetation, buildings, and urban structures, by dynamically adapting the Gaussian model parameters. The segmented images preserve important object boundaries and provide meaningful region separation for subsequent image analysis.

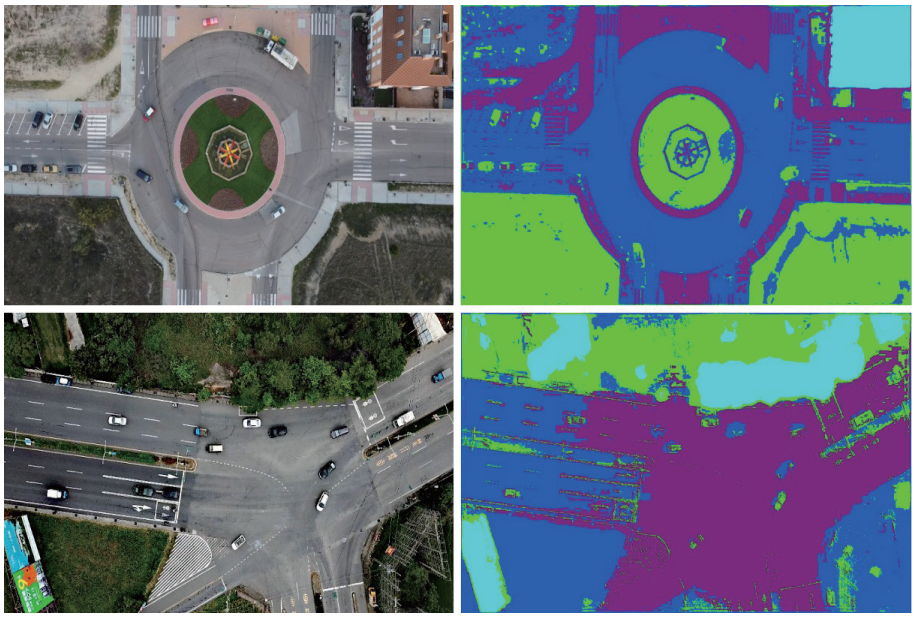

Fig. 2. (Color online) Segmentation results obtained using the proposed DAGMM. Different colors indicate different Gaussian clusters representing distinct image regions identified by the algorithm.

3.2.2 MRF for segmentation

In image segmentation, especially when dealing with the fused images obtained from the RGB and depth modalities, the use of contextual information in the form of MRFs is useful for modeling spatial dependences. After fusing RGB and depth maps, MRFs keep the segmentation results smooth and consistent, which is very important for the correct differentiation of regions or objects in the image after RGB–depth map fusion. MRF defines the probability distribution of a set of stochastic variables, where each stochastic variable is assigned to a pixel or a region in an image.[20] This spatial dependence is helpful in achieving better outcomes of segmentation since the algorithm guarantees that the values of adjacent pixels are close to one another.[21] In our approach, the objective of MRF-based segmentation is to determine the labeling that minimizes the energy function, which quantifies the trade-off between data fidelity and spatial smoothness, the compatibility of the solution with the input data as well as the degree of smoothness. The energy function E for MRF-based segmentation can be defined as

$$E(L) = \sum_i D(i, L_i) + \sum_{(i,j)\in N} V(L_i, L_j), \tag{4}$$

where L is the label configuration for all pixels and $D(i, L_i)$ is the data term that indicates the cost of assigning label L_i to pixel i. This term ensures that the segmentation closely aligns with the features extracted from the image and is represented as

$$D(i, L_i) = -\log\left(\sum_{k=1}^{K} \pi_k \cdot \mathcal{N}(f_i \mid \mu_k, \Sigma_k)\right), \tag{5}$$

where f_i is the feature vector for pixel i and $\sum_{k=1}^{K}$ is the summation over all K Gaussian components in the mixture. $\mathcal{N}(f_i \mid \mu_k, \sum_k)$ is the multivariate Gaussian probability density function for pixel i. $V(L_i, L_j)$ is the smoothness term, which penalizes label inconsistencies between neighboring pixels i and j.

$$V(L_i, L_j) = \lambda \cdot \exp\left(-\frac{\|p_i - p_j\|^2}{\sigma^2}\right) \cdot \left(1 - \delta(L_i, L_j)\right) \tag{6}$$

Here, λ controls the strength of the smoothness penalty, p_i and p_j are the positions of pixels i and j, respectively, σ is a parameter controlling the range of spatial smoothness, and $\delta(L_i, L_j)$ is the Kronecker delta, which equals 1 when $L_i = L_j$ and 0 otherwise, ensuring that the penalty is applied only when adjacent pixels have different labels. MRF segmentation significantly improves feature extraction by enforcing spatial coherence, which results in cleaner, well-defined object boundaries. This boundary clarity enables more precise feature extraction as the segmented regions contain fewer background artifacts and more distinct object characteristics. Figure 3 shows segmentation results produced by the MRF model.

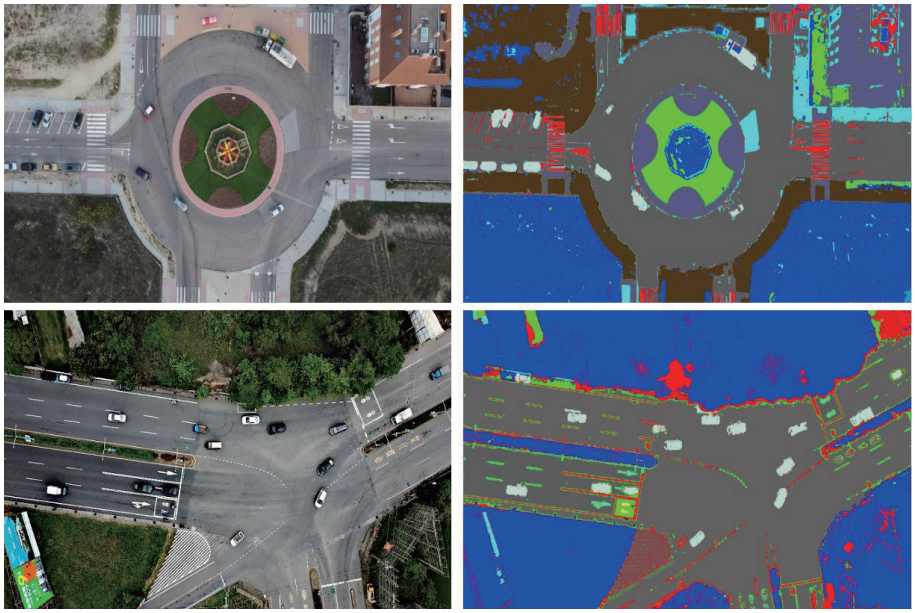

Fig. 3. (Color online) Segmentation results produced by the MRF model.

We have applied DAGMM and MRF for the segmentation. The effectiveness of MRF is evaluated using the Dice coefficient (DC) for each object, which indicates the similarity between the predicted segmentation and the ground truth, as shown below.

$$Dice = \frac{2 \times |X \cap Y|}{|X| + |Y|} \tag{7}$$

MRF is selected for further processing because of its better processing time and accuracy than DAGMM, as shown in Table 1.

Table 2 presents the object-wise DC scores obtained by the proposed DAGMM segmentation method from the Roundabout Aerial Images dataset (Roundabout) and Vehicle Aerial Imagery dataset (VAID). The DC score is a measure of the overlap between the segmented vehicle region and the corresponding ground truth, where a higher value indicates more accurate segmentation. The results show consistently high segmentation performance across different vehicle categories, with the highest DC score of 0.91 being achieved for Trailer in Roundabout and that of 0.87 for Truck and Bus in VAID. The proposed method achieves mean DC scores of 0.858 and 0.85 for Roundabout and VAID, respectively, demonstrating its ability to accurately segment different vehicle types and provide reliable inputs for the subsequent vehicle classification stage.

3.3 Vehicle detection

EfficientNet is a new and improved vehicle detection model for aerial imagery that embodies several new features. The algorithm is based on compound scaling, which optimizes three important parameters: the denser, wider, and higher the resolution of a network, the greater its

Table 1
Computational times and accuracies of DAGMM and MRF segmentations.

Dataset	Computational time (s)		Segmentation accuracy	
	DAGMM	MRF	DAGMM	MRF
Roundabout	210.15	187.2	81.3	84.3
VAID	180.35	171.7	82.1	85.2

Table 2
DC score for each object in the two datasets.

Roundabout		VAID	
Objects	DC score	Objects	DC score
Sedan	0.81	Car	0.85
Mini bus	0.83	Truck	0.87
Truck	0.89	Cycle	0.81
Pickup	0.86	Bus	0.87
Bus	0.83	—	—
Cement truck	0.88	—	—
Trailer	0.91	—	—
Mean	0.858	Mean	0.85

capacity to deliver reliable data for decision making. This results in a balanced performance of the overall process and also reduces the computational complexity. Regarding the architecture of the algorithm, a bidirectional feature pyramid network is used for feature fusion because vehicle size might be significantly different when analyzing aerial imagery. In theory, this fusion mechanism enables the network to efficiently pull apart and integrate data at various resolutions, which improves the detection. The focal loss is one of the key strengths of EfficientNet. This method also directly addresses the issue of class imbalance by giving weight to the difficult detection cases to improve the accuracy of the model. The experimental results further show that when the model is trained on specialized datasets such as VAID, the accuracy and recall rate of vehicle detection will be highly significant. The design is centered on deployment issues that are likely to be meaningful in the context of real-time unmanned aerial vehicle surveillance. This compound scaling method, rather, appropriately scales the width of the network and the number of layers to allow for the extraction of hierarchical features without incurring too large a computational cost. This is achieved through a layer-wise learning rate that controls the computational load despite the model growing in depth and width.[13]

3.4 Feature extraction: HWT

In this section, we explain the feature extraction process performed on images after vehicle detection with EfficientNet. A hybrid feature extraction method is proposed to obtain discriminative and robust feature representations by combining HWT and HOG.

HWT is used to divide the image into various frequency sub-bands, which allows for an efficient representation of both spatial and frequency domain information. HWT offers scale-invariance, which is essential in aerial imagery where vehicles are captured at different scales depending on the camera's angle and altitude. The wavelet transform is applied independently to each RGB color channel, decomposing the image into four sub-bands: approximation coefficients (Al)—low-frequency components capturing global structural information; horizontal detail coefficients (Hl)—high-frequency components capturing vertical edges; vertical detail coefficients (Vl)—high-frequency components capturing horizontal edges; and diagonal detail coefficients (Dl)—high-frequency components capturing diagonal textures.

Mathematically, HWT at level j is expressed as

$$W_{xyc}(m,n) = \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} I_{xyc}(x,y)\Psi\left(\frac{x-m}{2^j},\frac{y-n}{2^j}\right), \tag{8}$$

where $W_{xyc}(m, n)$ are the wavelet coefficients at scale j, $I_{xyc}(x, y)$ is the intensity at pixel (x, y) in channel c, and Ψ represents the Haar wavelet basis function.

After decomposition, high-frequency coefficients (H_l, V_l, D_l) provide detailed texture and edge-related features that are particularly useful for vehicle shape recognition.

$$W_{xyc}^{`}(m,n) = \begin{cases} W_{xyc}(m,n), & \text{if } \left|W_{xyc}(m,n)\right| > T \\ 0 & \text{Otherwise} \end{cases} \tag{9}$$

Here, $W_{xyc}^{`}(m,n)$ denotes the selected coefficients, $W_{xyc}(m, n)$ represents the original wavelet coefficients, and T is the threshold determined by the statistical analysis of the coefficient distribution. An adaptive approach was used, with the threshold T being dynamically determined by a percentile-based strategy, that is, only the most significant coefficients are kept. This method maximizes the trade-off between preserving features and suppressing noise. The effect of thresholding on performance was evaluated by changing T and measuring the detection accuracy.

3.5 HOG for texture and gradient features

Haar wavelets represent features in terms of frequency, but they do not represent local gradient structures, which are important for representing object shapes and contours. To address this, HOG is used. It is useful for identifying vehicle edges, orientations, and textures. The HOG feature extraction process starts with computing the gradient, which involves calculating the gradient magnitude and direction for each pixel, using the following formulas:

$$G_x = I(x+1, y) - I(x-1, y), \tag{10}$$

$$G_y = I(x, y+1) - I(x, y-1). \tag{11}$$

The gradient magnitude and orientation are then obtained as

$$G = \sqrt{G_x^2 + G_y^2}, \tag{12}$$

$$\theta = \tan^{-1}\left(\frac{G_y}{G_x}\right). \tag{13}$$

Next, the orientation binning step involves dividing the image into small cells and computing a histogram of gradient orientations within each cell, with each bin representing a particular orientation (e.g., 0–180°). Block normalization is performed across overlapping blocks by L2 normalization to ensure illumination invariance as follows:

$$H_{norm} = \frac{H}{\sqrt{\|H\|^2 + \varepsilon}}, \tag{14}$$

where ε is a small constant to avoid division by zero. This makes HOG a robust feature descriptor for vehicle classification as it is more robust to lighting changes and better at discriminating features.

An attention-based fusion mechanism is used to effectively fuse the multiresolution features of HWT with the gradient-based features of HOG. The fusion module learns adaptive weights for both types of features, rather than concatenation, to highlight the most relevant features. A self-attention mechanism is used, which dynamically assigns weights to HWT and HOG features. The attention mechanism is designed to emphasize the most discriminative features, enhancing the overall accuracy of vehicle detection and classification.

$$Q = W_Q . F_{haar} + b_q \tag{15}$$

$$K = W_k . F_{HOG} + b_k \tag{16}$$

$$V = W_v . \left(F_{haar} + F_{HOG} \right) + b_v \tag{17}$$

Here, Q, K, and V are the query, key, and value matrices used in attention computation. W_Q, W_k, and W_v are the learned projection matrices, and F_{haar}, and F_{HOG} represent the extracted features from *Haar* and *HOG*, respectively. The final feature is given by

$$F_{final} = \gamma A + (1 - \gamma)\left(F_{haar} + F_{HOG} \right), \tag{18}$$

where γ is a learnable parameter that balances attention-based selection and direct feature concatenation as shown in Fig. 4.

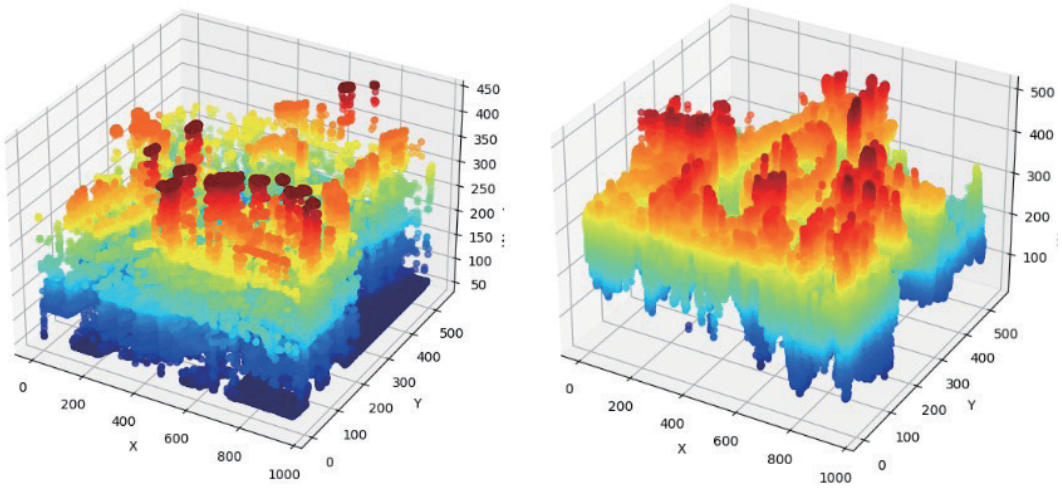

Fig. 4. (Color online) HWT-extracted features.

3.6 Classification: DenseNet

DenseNet is an advanced CNN architecture that introduces dense connections between layers, which helps to propagate features and gradients more effectively. In contrast to the conventional CNN architecture, which only connects the layers sequentially, DenseNet creates connections between all layers, enabling the efficient reuse of features and mitigating the vanishing gradient problem. Our implementation adapts the DenseNet-121 structure to work with pre-extracted feature vectors instead of raw images. The network processes 1024-dimensional input vectors, which contain precomputed features from aerial imagery, making the classification process more efficient by reducing redundant computations and memory overhead. The fundamental concept of DenseNet can be mathematically expressed as

$$xl = Hl\left(\left[x0,\ x1,\ldots,\ xl-1\right]\right), \tag{19}$$

where xl represents the output of the current layer and $[x0, x1,..., xl-1]$ indicates the concatenation of all preceding layer outputs. The composite function Hl consists of batch normalization, ReLU activation, and convolution operations. However, since we process 1D feature vectors (1×1024) instead of spatial feature maps, our architecture differs from conventional image-based DenseNet implementations. In our modified architecture, the 1024-dimensional input vector is directly passed to Dense Block 1, skipping the initial convolution and pooling layers designed for image-based input. Unlike standard DenseNet, which applies six convolution layers in Dense Block 1 for feature expansion in 2D spatial maps, our adaptation performs transformations in a 1D feature space using fully connected layers instead of convolutions. Each dense block applies feature transformations while maintaining dense connectivity, ensuring that information propagates efficiently across layers. Transition layers incorporating 1×1 convolutions and 2×2 average pooling are replaced with dimensionality reduction layers, which maintain computational efficiency in the 1D feature space. The ReLU activation function is employed throughout the network to introduce nonlinearity and prevent the vanishing gradient issue.

$$ReLU\left(x\right) = \max\left(0, x\right) \tag{20}$$

The network culminates in a final layer utilizing softmax activation, which produces probability distributions across our eight target classes.

$$Softmax\left(z\right)i = exp\left(zi\right) / \Sigma\left(exp\left(zj\right)\right) \tag{21}$$

It is an architectural modification of DenseNet that retains the good feature reuse properties of DenseNet, but is tailored to our aerial image classification task. Since we work in a space of pre-extracted feature vectors and not images, the network can be trained to learn high-level representations and decision boundaries that result in faster training and convergence[14].

Notably, our specific classification requirements imply the modification of the network to handle eight classes of outputs, which makes the network well suited to our case, and the input vector size provides a large feature space for learning discriminative patterns. It is also very connected, which facilitates feature re-use and gradient flow for training common deep neural network problems. In particular, this architecture is very successful for our task, as it takes advantage of precomputed features and retains all the powerful learning power of the original DenseNet structure as shown in Fig. 5.

4. Results and Analysis

Experiments were performed on the system equipped with an NVIDIA RTX 3090 GPU, 64 GB DDR4 RAM, and PyTorch 1.12.1 framework. This computational environment enabled us to process large aerial imagery datasets efficiently and train deep learning models in optimal performance. The evaluation is performed using two different aerial imagery datasets: Roundabout and VAID. These datasets offer a variety of environmental conditions and vehicle variations suitable for evaluating the generalization ability of our model. To evaluate the statistical reliability, we adopted the 5-fold cross validation method, splitting each of the datasets into five subsets. This approach provides a more holistic evaluation as each data point is utilized for training and testing in every fold.

4.1 Dataset description

Two publicly available aerial vehicle datasets, VAID and Roundabout, were used to evaluate the proposed vehicle classification framework. The datasets include aerial images taken under

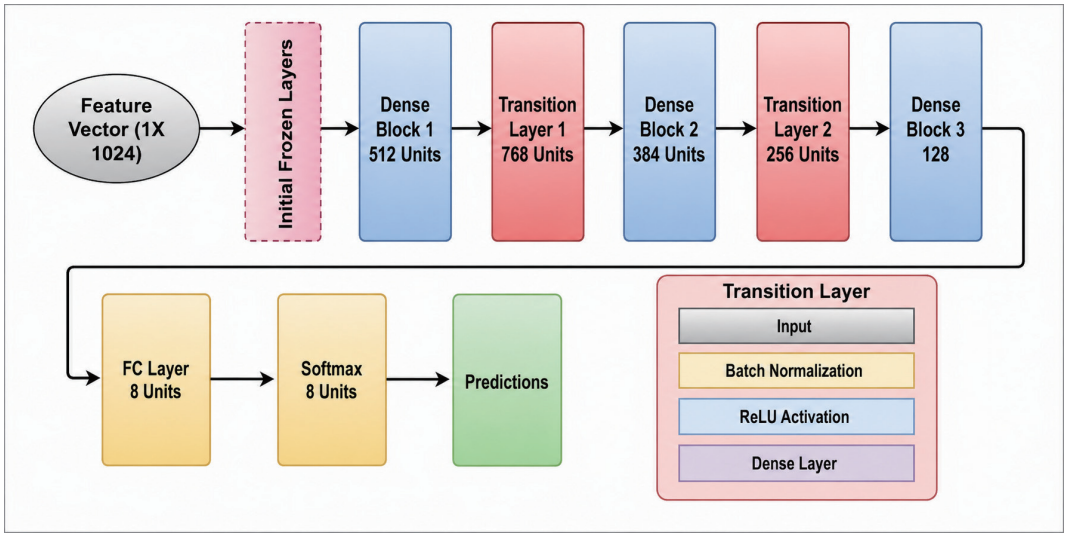

Fig. 5. (Color online) Modified DenseNet-121 architecture for aerial vehicle classification.

various environmental conditions and feature several types of vehicles, enabling us to evaluate the effectiveness and generalization potential.

4.1.1 VAID

VAID is the first dataset used for experimentation.[17] It includes drone aerial images of vehicles in urban environments throughout Taiwan. The dataset has seven different types of vehicles: sedans, buses, minibuses, trucks, pickup, cement truck, and trailer. The images are taken from an altitude between 90 and 95 m. The images are captured with a high spatial resolution of 2720 × 1530 pixels, allowing for a clear view of the vehicles and their surroundings. The variety of vehicles, opinions, and urban environments in VAID allows the assessment of the segmentation and classification accuracy of the proposed framework.

4.1.2 Roundabout

The second dataset is Roundabout,[18] which is widely used for vehicle detection and classification tasks. The dataset contains 61896 images captured by drones at eight different roundabouts in Spain. A total of 985260 labeled instances of different types of vehicles (cars, bicycles, tricycles, and buses) are included in the dataset. The confusion matrices for both datasets (VAID and Roundabout) show good overall performance of our vehicle categorization model. As presented in Tables 3 and 4, the high mean accuracies of these datasets (0.948–0.952) indicate the robustness and generalization ability of our vehicle categorization model.

Table 3
Confusion matrix of vehicle categorization on VAID.

Vehicle class	Sedan	Mini bus	Truck	Pickup	Bus	Cement truck	Trailer
Sedan	0.961	0.039	0	0	0	0	0
Mini bus	0.024	0.949	0	0.027	0	0	0
Truck	0	0	0.939	0.035	0	0	0.026
Pickup	0	0.349	0	0.951	0	0	0
Bus	0	0.030	0	0	0.942	0.028	0
Cement truck	0	0	0.049	0	0	0.951	0
Trailer	0	0	0.026	0	0	0	0.974
Mean: 0.952							

Table 4
Confusion matrix of vehicle categorization on Roundabout.

Vehicle class	Car	Cycle	Truck	Bus
Car	0.943	0	0.016	0.041
Cycle	0.069	0.931	0	0
Truck	0.017	0	0.956	0.027
Bus	0	0	0.037	0.963
Mean: 0.948				

The F1-score is a balanced evaluation metric that combines precision and recall into a single score. This is especially useful when there is a high cost to precision and recall, as it is the harmonic average of precision and recall. The higher the F1-score, the more effective the model is in balancing precision and recall, leading to accurate predictions and complete instance detection. The proposed model obtained 95.1% precision, 96.7% recall, and 93.5% F1-score on Roundabout, as shown in Fig. 6. Likewise, the model achieved 97.6% precision, 96.2% recall, and 96.9% F1-score on VAID, showcasing strong and consistent detection capabilities under various traffic conditions. The results of the proposed detection framework are generally good, with the precision, recall, and F1-score values being consistently high across both datasets, which validates the effectiveness and generalization capability of the proposed detection framework.

Table 5 shows the precision of the proposed vehicle detection model and state-of-the-art methods on Roundabout and VAID. The proposed model obtained the highest precision for both datasets, with 95.1% for Roundabout and 97.6% for VAID. The results show that the proposed framework, which uses MRF segmentation to enhance the quality of the vehicle regions detected

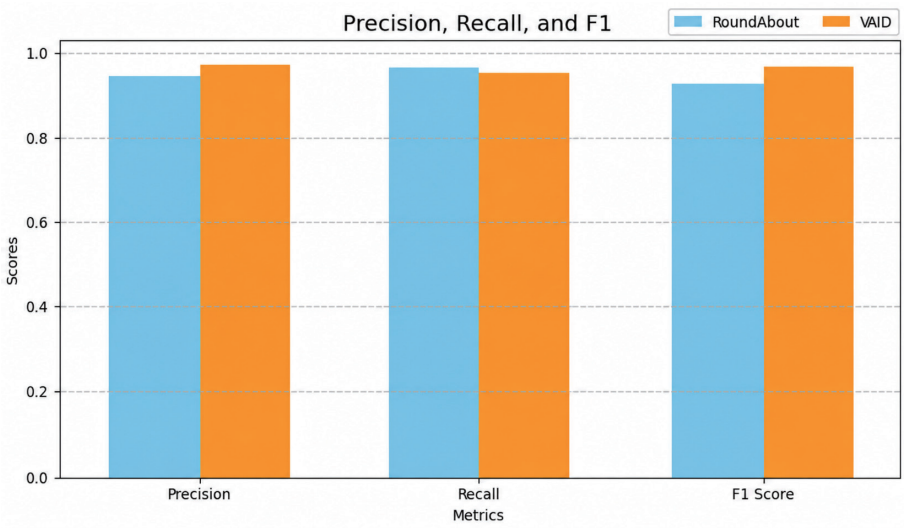

Fig. 6. (Color online) Precision, recall, and F1-score achieved by the proposed traffic object detection model on Roundabout and VAID.

Table 5
Vehicle detection precision values of our proposed method and existing methods.

Methods	Roundabout	VAID
YOLOv4[17]	—	0.940
YOLOv5[19]	0.941	0.963
EfficientDet[20]	—	0.970
Blob detection[21]	0.946	—
YOLOv8[22]	—	0.964
Proposed model	0.951	0.976

and minimize the background interference, is effective. The improved segmentation enables the network to learn more discriminative vehicle features, leading to higher detection precision. Moreover, the results obtained from various datasets show the robustness and generalization ability of the proposed method for vehicle detection in complex aerial scenes.

5. Conclusions

Breaking new ground in aerial surveillance capabilities, we demonstrated how the intelligent integration of classical and deep learning approaches can solve the long-standing challenges of reliable vehicle detection in complex aerial imagery. The integrated framework with Wiener filtering, MRF for segmentation, EfficientNet for detection, HWT for feature extraction, and DenseNet-121 for classification effectively overcomes the challenges of detecting aerial vehicles in crowded environments. The experimental results on VAID and Roundabout validate the robustness and reliability of the model. The achieved precision scores of 97.6 and 95.1%, respectively, represent a substantial improvement over existing approaches. The proposed framework demonstrates particular strength in handling challenging scenarios common in aerial imagery, such as varying lighting conditions, different vehicle orientations, and partial occlusions. For future work, the plan is to incorporate temporal information for tracking moving vehicles across consecutive frames. This would enhance robustness against occlusions by leveraging motion patterns. Another promising direction is integrating multimodal data, such as LiDAR or thermal imagery, which could improve detection under low-light conditions. Additionally, exploring self-supervised or semi-supervised learning approaches could reduce the dependence on large labeled datasets, making the model more adaptable to different geographical regions without requiring extensive manual annotations.

Acknowledgments

The authors are thankful to the Deanship of Graduate Studies and Scientific Research at Najran University for funding this work under grant code NU/CPI/SERC/14/313-1. The study is financially supported by Princess Nourah bint Abdulrahman University Researchers Supporting Project (PNURSP2026R719), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. This study is supported via funding from Prince sattam bin Abdulaziz University project number (PSAU/2026/R/1447).

References

1 A. Asvadi, L. Garrote, C. Premebida, P. Peixoto, and U. J. Nunes: Pattern Recognit. Lett. **115** (2018) 20. https://doi.org/10.1016/j.patrec.2017.09.038
2 N. Al Mudawi, M. W. Ahmed, H. F. Alhasson, N. S. Alshassari, A. Alazeb, M. Alshehri, and B. Alabdullah: PeerJ Comput. Sci. **11** (2025) e3270. https://doi.org/10.7717/peerj-cs.3270
3 B. Wang and Y. Gu: Sensors **20** (2020) 4709. https://doi.org/10.3390/s20174709
4 Q. Song and X. Chen: IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. **15** (2022) 7988. https://doi.org/10.1109/JSTARS.2022.3206036
5 N. U. A. Tahir, Z. Zhang, M. Asim, J. Chen, and M. ELAffendi: Algorithms **17** (2024) 103. https://doi.org/10.3390/a17030103

6 X. Guo, J. Yang, X. Jia, C. Zang, Y. Xu, and Z. Chen: Sci. Rep. **14** (2024) 2027. https://doi.org/10.1038/s41598-024-52225-x
7 Q. Wang, Y. Chi, T. Shen, J. Song, Z. Zhang, and Y. Zhu: Remote Sens. **14** (2022) 2020. https://doi.org/10.3390/rs14092020
8 T. Liu, R. Zhao, and K.-M. Lam: Proc. SPIE **11766** (2021) 1176603. https://doi.org/10.1117/12.2590661
9 J. Dudczyk, R. Czyba, and K. Skrzypczyk: Sensors **22** (2022) 4323. https://doi.org/10.3390/s22124323
10 H. Wang, M. Wang, Z. Che, Z. Xu, X. Qiao, M. Qi, F. Feng, and J. Tang: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (2022) 6209. https://doi.org/10.1109/CVPR52688.2022.00611
11 P. Sun, W. Zhang, H. Wang, S. Li, and X. Li: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (2021) 1407. https://doi.org/10.1109/CVPR46437.2021.00146
12 M. Bakirci, P. Dmytrovych, I. Bayraktar, and O. Anatoliyovych: Proc. IEEE 7th Int. Conf. Actual Probl. Unmanned Aerial Veh. Dev. (APUAVD) (2024) 191. https://doi.org/10.1109/APUAVD63440.2024.10765862
13 N. Ahmed, A. A. Almujally, A. Alazeb, A. Algarni, and J. Park: Comput. Mater. Continua **79** (2024) 3315. https://doi.org/10.32604/cmc.2024.046501
14 A. Soumya, A. K. Dubey, C. K. Mohan, and L. R. Cenkeramaddi: J. Ambient Intell. Humaniz. Comput. **17** (2026) 413. https://doi.org/10.1007/s12652-026-05037-y
15 M. Cao, H. Ji, Z. Gao, and T. Mei: ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. V-2-**2020** (2020) 797. https://doi.org/10.5194/isprs-annals-V-2-2020-797-2020
16 L. Cao, F. Luo, L. Chen, Y. Sheng, H. Wang, C. Wang, and R. Ji: Pattern Recognit. **64** (2017) 417. https://doi.org/10.1016/j.patcog.2016.10.033
17 H.-Y. Lin, K.-C. Tu, and C.-Y. Li: IEEE Access **8** (2020) 212209. https://doi.org/10.1109/ACCESS.2020.3040290
18 E. Puertas, G. De-Las-Heras, J. Fernández-Andrés, and J. Sánchez-Soriano: Data **7** (2022) 47. https://doi.org/10.3390/data7040047
19 A. M. Qureshi, A. H. Butt, A. Alazeb, N. A. Mudawi, M. Alonazi, N. A. Almujally, A. Jalal, and H. Liu: Comput. Mater. Continua **80** (2024) 3315. https://doi.org/10.32604/cmc.2024.052582
20 M. O. Yusuf, M. Hanzla, N. Al Mudawi, T. Sadiq, B. Alabdullah, H. Rahman, and A. Algarni: Front. Neurorobot. **18** (2024) 1448538. https://doi.org/10.3389/fnbot.2024.1448538
21 A. M. Qureshi, A. Algarni, H. Aljuaid, M. H. Alatiyyah, K. Alnowaiser, and A. Jalal: SN Comput. Sci. **6** (2025) 30. https://doi.org/10.1007/s42979-024-03586-7
22 N. Al Mudawi, A. M. Qureshi, M. Abdelhaq, A. Alshahrani, A. Alazeb, M. Alonazi, and A. Algarni: Sustainability **15** (2023) 14597. https://doi.org/10.3390/su151914597