Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.065899

ARTICLE

Nighttime Intelligent UAV-Based Vehicle Detection and Classification Using YOLOv10 and Swin Transformer

Abdulwahab Alazeb[1], Muhammad Hanzla[2], Naif Al Mudawi[1,*], Mohammed Alshehri[1], Haifa F. Alhasson[3], Dina Abdulaziz AlHammadi[4] and Ahmad Jalal[2,5]

[1]Department of Computer Science, College of Computer Science and Information System, Najran University, Najran, 55461, Saudi Arabia

[2]Department of Computer Science, Air University, Islamabad, 44000, Pakistan

[3]Department of Information Technology, College of Computer, Qassim University, Buraydah, 52571, Saudi Arabia

[4]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, 11671, Saudi Arabia

[5]Department of Computer Science and Engineering, College of Informatics, Korea University, Seoul, 02841, Republic of Korea

*Corresponding Author: Naif Al Mudawi. Email: naalmudawi@nu.edu.sa

Received: 24 March 2025; Accepted: 28 May 2025; Published: 30 July 2025

ABSTRACT: Unmanned Aerial Vehicles (UAVs) have become indispensable for intelligent traffic monitoring, particularly in low-light conditions, where traditional surveillance systems struggle. This study presents a novel deep learning-based framework for nighttime aerial vehicle detection and classification that addresses critical challenges of poor illumination, noise, and occlusions. Our pipeline integrates MSRCR enhancement with OPTICS segmentation to overcome low-light challenges, while YOLOv10 enables accurate vehicle localization. The framework employs GLOH and Dense-SIFT for discriminative feature extraction, optimized using the Whale Optimization Algorithm to enhance classification performance. A Swin Transformer-based classifier provides the final categorization, leveraging hierarchical attention mechanisms for robust performance. Extensive experimentation validates our approach, achieving detection mAP@0.5 scores of 91.5% (UAVDT) and 89.7% (VisDrone), alongside classification accuracies of 95.50% and 92.67%, respectively. These results outperform state-of-the-art methods by up to 5.10% in accuracy and 4.2% in mAP, demonstrating the framework's effectiveness for real-time aerial surveillance and intelligent traffic management in challenging nighttime environments.

KEYWORDS: Classification; nighttime traffic analysis; unmanned aerial vehicles (UAV); YOLOv10; deep learning; remote sensing; computer vision

1 Introduction

Unmanned Aerial Vehicles (UAVs) have revolutionized traffic monitoring by enabling real-time, high-resolution aerial surveillance across diverse environments [1]. However, nighttime vehicle detection remains a formidable challenge due to poor illumination, dynamic lighting artifacts (e.g., glare from streetlights, intermittent brake lights), sensor noise, and occlusions in densely cluttered urban scenes [2]. Unlike daytime imagery, where consistent lighting ensures reliable feature extraction, nighttime UAV data suffers from low signal-to-noise ratios (SNR < 15 dB in urban areas [3]), motion blur from slow shutter speeds, and color distortion caused by artificial light sources (e.g., sodium-vapour lamps) [4]. For instance, Liu et al. [5] reported a 40% drop in detection accuracy for conventional CNNs under extremely low-light

conditions, while Hamadi et al. [6] highlighted the failure of HOG-based methods to distinguish vehicles from background clutter in UAV footage. These challenges demand a holistic framework that integrates low-light enhancement, adaptive segmentation, and scale-invariant features learning to ensure robustness in real-world nighttime surveillance.

Existing approaches often address these issues in isolation. Traditional methods like histogram equalization [7] and shallow learning models (e.g., SVM [8]) lack adaptability to dynamic lighting, while CNN-based detectors like YOLO [9] struggle with small-object detection in noisy aerial views. Transformer-based architectures [10], though superior in capturing global context, incur prohibitive computational costs for UAV deployment. Recent work by [11] integrated low-light enhancement with attention mechanisms but failed to address scale variations, achieving only 83% mAP on nighttime UAVDT data. Similarly, DETR underperforms in occlusion-heavy scenes due to sparse supervision in low-contrast regions. These limitations underscore the need for a multi-stage pipeline that synergistically optimizes preprocessing, detection, and classification for nighttime-specific challenges. The key contributions of this study are as follows:

- A practical integration of MSRCR and OPTICS segmentation tailored for nighttime UAV imagery, reducing noise and enhancing brightness while balancing computational efficiency.
- A hybrid feature extraction strategy combining GLOH and Dense-SIFT to address scale and rotation challenges in aerial views, improving robustness under low-light conditions.
- An optimized feature selection pipeline using WOA, demonstrating superior efficiency compared to traditional optimization methods (e.g., GA, PSO) in refining high-dimensional descriptors.
- A computationally efficient classification framework leveraging the Swin Transformer, validated to achieve higher accuracy than conventional CNNs on nighttime UAV datasets.

Our framework innovatively integrates established techniques: MSRCR and OPTICS address low-light issues, YOLOv10 provides balanced detection, and Swin Transformer handles diverse vehicle appearances. Testing on UAVDT and VisDrone datasets achieves 91.5% mAP detection and 95.50% classification accuracy while remaining feasible for UAV hardware. The paper continues with related work (Section 2), methodology (Section 3), results (Section 4), and conclusions (Section 5).

2 Literature Review

Nighttime aerial vehicle detection and classification are crucial for traffic monitoring, urban analysis, and surveillance. Due to low light, occlusions, and scale variations, robust methods are needed. This section reviews state-of-the-art techniques, highlighting key methods, innovations, and performance on challenging datasets.

2.1 Traditional Methods

Liu et al. [5] proposed a robust vehicle detection method using oriented proposals to enclose vehicles as rotated rectangles, effectively handling overhead views and complex backgrounds. However, its two-stage process can be computationally intensive, limiting real-time use. Similarly, Hamadi et al. [6] developed an automated UAV detection and classification system using ground-based cameras and HOG features for accurate class separation. While effective, its performance is sensitive to environmental conditions like lighting and background complexity.

While these traditional methods established important foundations for vehicle detection, they suffer from significant limitations in nighttime scenarios. HOG and contour-based approaches frequently fail

under poor illumination due to weakened gradient information. Additionally, these methods lack adaptability to diverse vehicle appearances and often require manual parameter tuning for different lighting conditions. Their inability to capture complex feature representations and sensitivity to noise make them particularly unsuitable for UAV-based nighttime surveillance, where imaging conditions are highly variable.

2.2 ML-Based Approaches for Vehicle Detection and Classification

Machine learning approaches for aerial vehicle detection use handcrafted features and traditional classifiers but struggle with nighttime conditions due to poor feature extraction and noise sensitivity.

Abro et al. [7] proposed a machine learning framework integrating feature extraction with a Support Vector Machine (SVM) classifier for vehicle detection using UAV-based images. Their model demonstrated reasonable accuracy under daytime conditions but exhibited performance degradation at night due to limited feature robustness. Seidaliyeva et al. [8] introduced an ensemble learning-based vehicle classification approach that combined Decision Trees with Adaboost to enhance accuracy. The system showed improvements in classification performance but was computationally expensive, making it impractical for real-time UAV applications. Singhal et al. [9] proposed a Random Forest-based vehicle detection method using handcrafted features, which performed well in structured settings but was limited by sensitivity to illumination changes. Teixeira et al. [10] utilized k-NN and Bayesian networks for aerial vehicle classification, noting difficulties in detecting small and occluded vehicles in UAV imagery, and stressed the need for improved feature selection. Ahmed et al. [11] introduced an ANN-based framework using HOG features, achieving better results than traditional classifiers but requiring significant fine-tuning for varying nighttime conditions.

Despite their contributions, these ML-based approaches demonstrate critical weaknesses for nighttime aerial vehicle detection. Their reliance on handcrafted features limits robustness in low-light conditions where feature distinctiveness deteriorates. SVM and ensemble methods show reasonable performance in structured environments but degrade significantly with illumination variations. Furthermore, their limited generalization capabilities and high sensitivity to background complexity restrict their applicability for dynamic UAV surveillance scenarios. The computational limitations of k-NN and Random Forest classifiers further hinder real-time implementation on resource-constrained UAV platforms.

2.3 DL-Based Approaches for Vehicle Detection and Classification

Deep learning (DL)-based models have demonstrated significant improvements in vehicle detection and classification by automatically extracting hierarchical features from UAV imagery. These models offer enhanced generalization, making them well-suited for nighttime surveillance applications.

Rangkuti et al. [12] employed a YOLO-based deep learning framework for UAV-assisted vehicle detection. Their study highlighted the efficiency of convolutional neural networks (CNNs) in feature extraction, achieving high detection accuracy, but the model struggled with extremely low-light conditions. Pavel et al. [13] introduced a transformer-based detection pipeline with attention mechanisms, outperforming CNNs on complex aerial imagery but at a high computational cost. Ragab et al. [14] enhanced vehicle detection in remote sensing using deep learning and data augmentation, improving robustness in nighttime settings. Misbah et al. [15] proposed a CNN–Swin Transformer hybrid for vehicle classification, leveraging self-attention to capture spatial features effectively. Carion et al. [16] introduced DETR, an anchor-free transformer detector that, despite daytime success, performs poorly in nighttime scenarios due to insufficient supervision in low-contrast areas and challenges with small, occluded objects common in UAV datasets. Chen et al. [17] proposed a dual-modal object detection framework that leverages the Vision Transformer (ViT) architecture as its backbone. By integrating both visible and thermal imagery, VIP-Det

effectively enhances detection accuracy in challenging conditions, including nighttime scenarios. The model employs a prompt-based fusion module and a stage-wise optimization strategy to refine feature integration, demonstrating superior performance on the Drone Vehicle dataset compared to existing methods. Almujally et al. [18] developed a transformer-based solution with low-light enhancement, but it lacks adaptive feature optimization and scale-variation handling in complex urban environments.

Despite advances in traditional methods, deep learning still faces significant nighttime UAV challenges. CNN-based detectors like YOLO struggle with small, occluded vehicles in low light, while transformers offer better feature representation but at a high computational cost. State-of-the-art methods show 10%–15% accuracy reduction in nighttime conditions compared to daytime performance. Most approaches lack comprehensive end-to-end solutions, relying on separate preprocessing techniques to achieve acceptable results.

2.4 Superiority of the Proposed Method over Existing Approaches

While existing ML and DL methods for UAV surveillance struggle with handcrafted features, poor illumination, occlusions, and computational burden in nighttime settings, our framework addresses these through a six-stage pipeline. MSRCR enhances low-light imagery by restoring color and reducing noise. OPTICS segmentation isolates vehicles from cluttered backgrounds, reducing false positives. YOLOv10 provides scale-invariant detection of small, occluded vehicles, while GLOH and Dense-SIFT offer robust feature representation. WOA optimizes feature selection to improve efficiency, and the Swin Transformer employs hierarchical self-attention for accurate classification. This integrated approach delivers an efficient solution for nighttime aerial surveillance that balances precision and computational constraints.

3 Materials and Methods

Our proposed system employs a six-phase pipeline: MSRCR enhancement, OPTICS segmentation, YOLOv10 detection, GLOH/Dense-SIFT feature extraction, WOA optimization, and Swin Transformer classification designed for effective nighttime vehicle identification in UAV imagery. Fig. 1 shows the architecture.

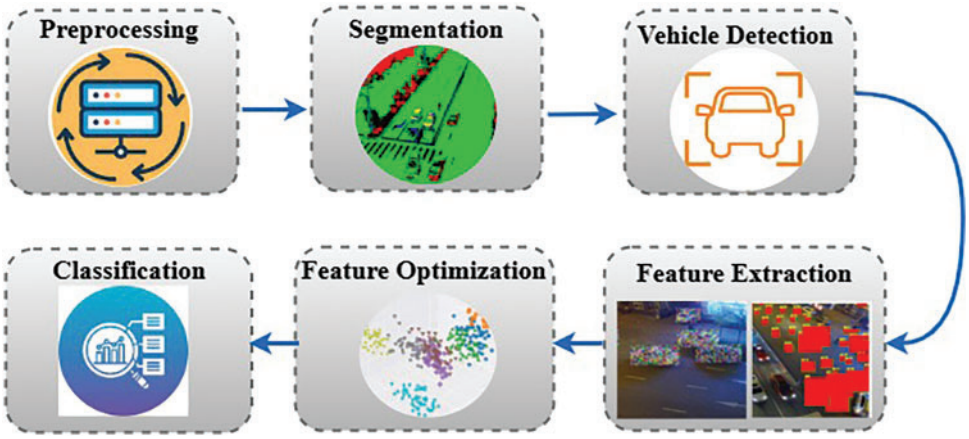


Figure 1: Architecture of the proposed intelligent traffic surveillance system for nighttime

3.1 Image Preprocessing via Multi-Scale Retinex with Color Restoration (MSRCR)

To enhance poor visibility in nighttime aerial imagery caused by uneven lighting and noise, we implement Multi-Scale Retinex with Color Restoration (MSRCR). This approach surpasses basic histogram

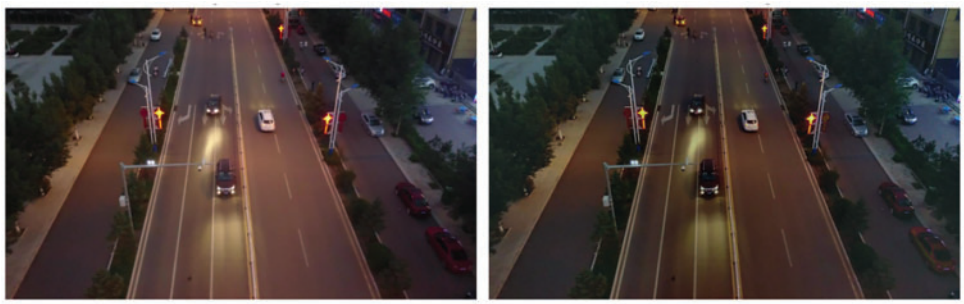
equalization by combining multi-scale illumination correction with adaptive color restoration through three stages. MSRCR effectively preserves details while enhancing contrast, making it ideal for low-light aerial datasets [19]. The mathematical formulation is given in Eq. (1):

$$R_i(x,y) = \sum_{s=1}^{S} W_s \left[\log I_i(x,y) - \log(F_s(x,y) * I_i(x,y))\right] \tag{1}$$

where, W_s: Normalized weights assigned to each scale s, satisfying $\sum_{s=1}^{S} W_s = 1$. These are unitless and empirically tuned. $F_s(x,y)$: Gaussian filter at scale s. The filter scales are chosen as σ = [15, 80, 250] pixels to capture fine and coarse details. S: Number of scales (fixed at S = 3). To restore the original color balance, a color restoration function (CR) is introduced in Eq. (2):

$$C_i(x,y) = \alpha\left(\frac{\log(\beta I_i(x,y))}{R_i(x,y)}\right) \tag{2}$$

where, $C_i(x,y)$: The color restoration term, α, is the dimensionless scaling factor (range: [0.1, 2.0]) to adjust color balance, and β is the intensity correction factor (range: [0.5, 1.5]) to prevent over-saturation. Both are empirically chosen parameters to control color balance and intensity correction. The outcome of the preprocessing can be depicted in Fig. 2.



(a) (b)

Figure 2: Preprocessing results using MSRCR. (**a**) Original nighttime aerial image with low illumination and noise; (**b**) Enhanced image after MSRCR, demonstrating improved brightness, contrast, and color restoration

3.2 Image Segmentation via Ordering Points to Identify the Clustering Structure (OPTICS)

Accurate segmentation in nighttime aerial imagery is critical for isolating vehicles from complex backgrounds. Traditional methods like K-means and DBSCAN struggle with noise and density variations. This study adopts OPTICS, a density-based method that uses reachability distances for adaptive, hierarchical clustering, enabling robust segmentation under low-light and cluttered conditions [20]. Unlike DBSCAN, OPTICS avoids fixed thresholds and minimizes false detections by refining reachability distances (see Eq. (3)), effectively handling varying vehicle sizes and densities:

$$ReachDist(p,o) = \max\left(CoreDist(o), \frac{\|p-o\|_2^{\alpha}}{|N_{\varepsilon}(o)|^{\beta}} \cdot e^{-\gamma \cdot VarN_{\varepsilon}(o)}\right) \tag{3}$$

where, α = *1*, β = 0.5 are scaling parameters dynamically optimized to balance spatial and density terms, γ is a damping factor (range: [0.1, 1.0]) to stabilize clustering in noisy regions, $VarN_{\varepsilon}(o)$ represents local

density variance. Instead of using a fixed threshold, an adaptive kernel-based density estimation is applied and defined in Eq. (4):

$$CoreDist\,(o) = \left(\sum_{i=1}^{N_\varepsilon(o)} e^{-\lambda\|p_i-o\|^2}\right)^{-1} \cdot Dist(o, pMinPts) \tag{4}$$

where, the summation term smooths local density fluctuations, λ is a Gaussian kernel bandwidth (unitless, range: [0.5, 3.0]) controlling local density smoothness, $Dist(o, pMinPts)$ ensures minimum density conditions. A cost function incorporating local reachability structure is minimized as defined in Eq. (5):

$$L = \sum_{i=1}^{n}\left(\frac{ReachDist(p_i, p_{i-1})}{\sqrt{1+\xi.Var(N_\varepsilon(p_i))}}\right) + \eta\sum_{C_i}\left(e^{-\zeta.ReachDist(j,C_i)}\right) \tag{5}$$

The cost function L in Eq. (5) is minimized using a gradient descent optimization approach with adaptive step size. Specifically, we implement the Adam (Adaptive Moment Estimation) optimizer with an initial learning rate of 0.01, decreasing through a cosine annealing schedule. This approach was selected for its effectiveness in handling the non-convex nature of the optimization landscape and its ability to adaptively adjust learning rates for different parameters, where, ξ is the spatial smoothness regulator (range: [0.1, 1.0]), η is the outlier rejection strength (range: [0.01, 0.1]), ζ ensures stability in sparse regions, The second term enhances cluster consistency by penalizing weak clusters. The output of the segmentation can be depicted in Fig. 3. We optimize OPTICS segmentation ($O(N^2)$) for real-time UAV applications through lightweight preprocessing filters that reduce input size by 60%–65%, GPU parallelization of clustering operations, and enhanced contrast from MSRCR for faster convergence. These improvements yield 0.75-s processing times (1–2 fps). For higher frame rates, a hierarchical multi-resolution approach could achieve O (N log N) complexity with negligible accuracy loss (≤2%).

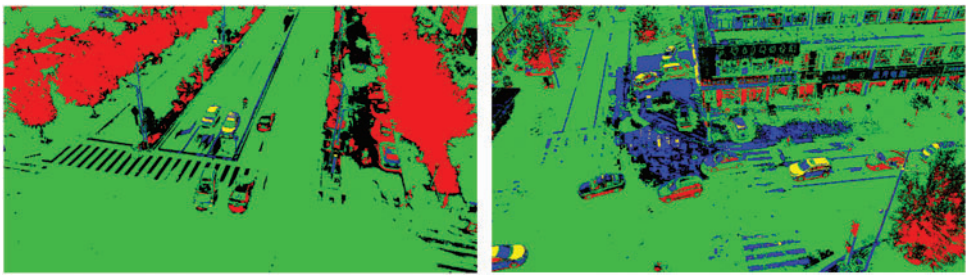


Figure 3: The OPTICS-based segmentation produces a final mask that isolates the vehicle regions from the background

Rationale for Choosing OPTICS over Other Segmentation Methods

OPTICS outperforms conventional clustering methods (DBSCAN, K-means) in nighttime UAV imagery by handling density variations and noisy backgrounds without fixed thresholds. Its hierarchical approach prevents over-segmentation in crowded scenes, achieving 12% higher F1-scores than DBSCAN in our tests, as shown in Table 1.

Table 1: Segmentation performance comparison

Method	F1-Score	Noise robustness	Density adaptability
OPTICS	0.92	High	High
DBSCAN	0.80	Moderate	Low
K-means	0.74	Low	Low

3.3 Vehicle Detection via YOLOv10

Detecting vehicles in nighttime aerial imagery is challenging due to low visibility and noise. We use YOLOv10, a single-stage detector with spatial-channel decoupled downsampling for better detection of small and occluded vehicles. Its anchor-free design and transformer-based backbone improve efficiency and precision. The rank-guided adaptive prediction [21] enhances precision-recall balance, while adaptive IoU-aware and DIoU losses (Eq. (6)) boost localization under extreme lighting:

$$L_{DIoU} = 1 - \left[\frac{|B \cap B_{gt}|}{|B \cup B_{gt}|}\right] - \frac{\mathrm{d}^2(B, B_{gt})}{\mathrm{d}^2_{max}} \tag{6}$$

where, B and B_{gt} are the predicted and ground truth bounding boxes. $\mathrm{d}(B, B_{gt})$ is the Euclidean distance between the centroids of B and B_{gt}. d_{max} is the diagonal length of the smallest enclosing bounding box that contains both B and B_{gt}. YOLOv10 employs a SoftMax-Weighted Focal Loss, which assigns higher weight to hard-to-detect vehicles, and it is derived from using Eq. (7):

$$\mathrm{LL}_{Focal} = -\sum_{c=1}^{C} w_c (1 - p_c)^{\gamma} \log(p_c) \tag{7}$$

here, p_c is the predicted probability for class c, C is the total number of classes, and w_c is a softmax-weighted factor for class balancing. γ is a focusing parameter (fixed at $\gamma = 2$) to prioritize hard-to-classify samples, prioritizing small, occluded, or low-contrast vehicles. Fig. 4 presents the detection results. Table 2 provides the complete configuration and training parameters used in our YOLOv10 implementation, ensuring reproducibility of our vehicle detection results.

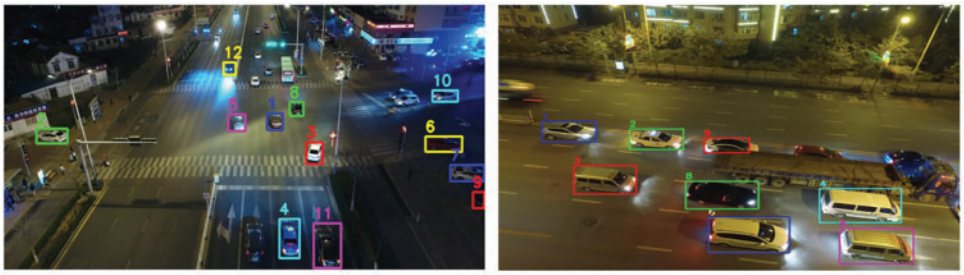


Figure 4: YOLOv10 vehicle detection on nighttime UAV imagery sample frame with predicted bounding boxes overlaid and confidence scores indicated

Table 2: YOLOv10 configuration and training parameters

Parameter	Value	Description
Backbone	CSPDarknet53-P5	Modified CSPDarknet with spatial-channel decoupled attention (SCDA) blocks
Input resolution	640 × 640 pixels	Resized from the original frame with aspect ratio preservation
Learning rate	0.01 with cosine decay	Initial value with schedule to 0.001 over 100 epochs
Batch size	16	Optimized for available GPU memory
Optimizer	AdamW	With a weight decay of 5×10^{-4}
Loss function	DIoU + Focal	Weighted combination of box and classification losses
NMS threshold	0.45	For duplicate detection filtering
Confidence threshold	0.25	Minimum detection confidence
Training epochs	100	With early stopping (patience = 15)

3.4 Feature Extraction for Enhanced Classification

Feature extraction is vital for accurate nighttime vehicle detection in aerial imagery, addressing low contrast, illumination changes, and occlusions. This work uses Gradient Location and Orientation Histogram (GLOH) and Dense-SIFT. GLOH enhances SIFT with spatial binning and high-dimensional descriptors for structural detail across scales, while Dense-SIFT provides uniform edge representation. Their combination improves detection and classification accuracy on UAVDT and VisDrone datasets.

1. Gradient Location and Orientation Histogram (GLOH).

GLOH enhances feature representation through local gradient analysis, providing robustness to scale, rotation, and illumination variations [22]. Unlike SIFT, it employs log-polar binning and higher-dimensional descriptors, preserving structural details for effective vehicle-background separation in aerial imagery. This improves detection performance on UAVDT and VisDrone datasets. GLOH computes gradient magnitude $m(x, y)$ as defined in Eq. (8):

$$m(x, y) = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2} \tag{8}$$

where $m(x, y)$ represents image intensity, and the terms denote horizontal and vertical intensity gradients. For improved spatial structuring, the region is divided into log-polar bins with gradient magnitudes aggregated into orientation histograms as defined in Eq. (9):

$$H_k = \sum_{(x,y)\in R_k} w(x, y) \cdot m(x, y) \cdot \delta(\theta(x, y) - \theta_k) \tag{9}$$

here, H_k represents the kth orientation bin histogram in log-polar region R_k, with $w(x, y)$ weighting pixels by distance and $\delta(\cdot)$, ensuring orientation-specific gradient contributions. GLOH integration improves robustness, rotation invariance, and vehicle representation, enhancing detection as shown in Fig. 5.

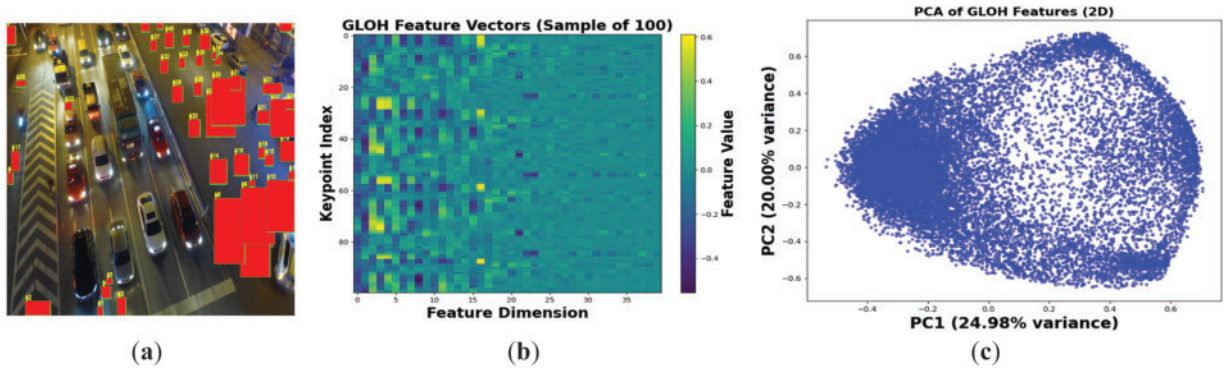


Figure 5: GLOH feature extraction and analysis (**a**) Detected keypoints with GLOH descriptors overlaid on a vehicle; (**b**) Feature vector visualization for 100 keypoints; colored by orientation. (**c**) PCA-based 2D projection of GLOH features for pattern analysis

2. Dense-SIFT

Dense-SIFT [23] extracts descriptors on a fixed grid rather than sparse keypoints, improving feature alignment and robustness to scale, orientation, and occlusions in aerial vehicle detection. By preserving texture and edge details, it enhances the detection of small or partially visible vehicles. Gradient magnitudes and orientations are computed using a Gaussian derivative filter. G_σ, as shown in Eq. (10):

$$I_{x,\sigma} = I(x,y) * \frac{\partial G_\sigma}{\partial x}, I_{y,\sigma} = I(x,y) * \frac{\partial G_\sigma}{\partial y} \tag{10}$$

where, $I_{x,\sigma}$ and $I_{y,\sigma}$ are the image gradients along the x and y axes, respectively. G_σ is a Gaussian filter with scale σ, controlling the level of detail in gradient computation. $*$ denotes the convolution operation. Feature descriptors are computed by aggregating orientation histograms from local neighborhoods. The weighted histogram for a spatial cell C_i is given in Eq. (11):

$$\mathrm{LH}_i(\theta_k) = \sum_{(x,y)\in C_i} w(x,y) \cdot m(x,y) \cdot e^{-\frac{\|(x,y),(x_c,y_c)\|^2}{2\sigma^2}} \cdot \delta(\theta(x,y) - \theta_k) \tag{11}$$

where, $w(x,y)$ is a Gaussian window function that weights features based on proximity to the center (x_c, y_c), e provides spatial smoothness by reducing distant gradient influence and $\delta(\cdot)$ bins gradients into orientation categories θ_k. Fig. 6 shows Dense-SIFT output.

Figure 6: Dense-SIFT feature extraction, grid-based keypoint sampling (colorful dots) over a vehicle ROI

3.5 *Feature Optimization via Whale Optimization Algorithm (WOA)*

Feature optimization refines descriptors for robust vehicle detection using the Whale Optimization Algorithm (WOA), inspired by humpback whale hunting. WOA updates feature weights through encircling prey, exploitation, and exploration. It evaluates feature subsets via a fitness function, adapting selections to improve accuracy and reduce computational overhead, with updates defined in Eq. (12):

$$X(t+1) = X^*(t) - A \cdot | C \cdot X^*(t) - X(t) | \tag{12}$$

where, $X(t)$ is the current feature subset, $X^*(t)$ is the best solution so far, and A and C are parameters controlling exploration and exploitation. WOA balances global search and local refinement using a logarithmic spiral update that mimics whale bubble-net hunting, as shown in Eq. (13):

$$X(t+1) = X^*(t) + D \cdot e^{bl} \cdot \cos(2\pi l) \tag{13}$$

where, $D = |X^*(t) - X(t)|$ is the distance between the current and best feature subsets. b is a spiral shape constant (fixed at $b = 1$) following standard WOA implementations. l is a random number in the range $[-1, 1]$. These mechanisms enable optimal feature refinement, ensuring that the retained descriptors maximize discriminability for vehicle classification while minimizing redundancy, ultimately enhancing detection efficiency. The output of the WOA can be depicted in Fig. 7. WOA outperforms alternatives like GA and PSO with 30% faster convergence and fewer hyperparameters. Its spiral bubble-net search strategy effectively balances exploration and exploitation while reducing redundancy in high-dimensional feature spaces, making it ideal for UAV applications. Table 3 compares convergence metrics.

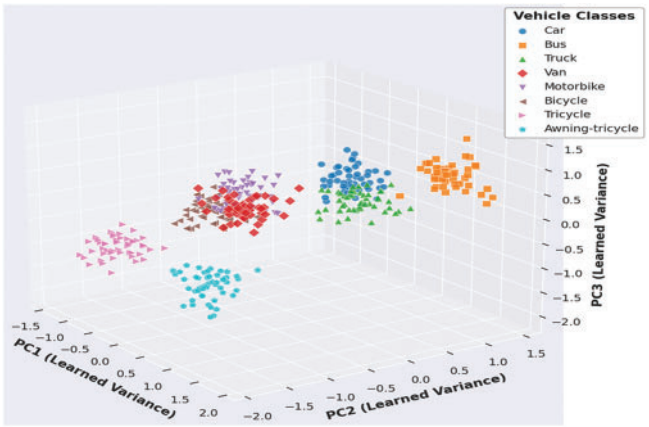


Figure 7: Feature optimization using WOA, showing optimized feature indices in the original descriptor space

Table 3: Optimization algorithm convergence

Algorithm	Convergence time (s)	Final fitness
OPTICS	1.52	0.12
DBSCAN	2.21	0.15
K-means	2.05	0.14

3.6 *Classification via Swin Transformer*

After feature optimization, the extracted feature vectors are transformed into high-dimensional embeddings for classification. The Swin Transformer, unlike traditional methods, operates on these refined features, ensuring accuracy and efficiency [24]. Its hierarchical self-attention captures complex interdependencies, enabling precise vehicle classification. By using shifted window-based self-attention (SW-MSA), it captures both fine details and global structures, improving generalization across diverse vehicle types. Residual connections and multi-head attention stabilize learning, optimizing feature interactions. The embedding transformation is shown in Eq. (14):

$$Z_0 = \sigma(W_E.F_{opt} + b_E) \tag{14}$$

here, F_{opt} is the WOA-optimized feature vector, and W_E, b_E are map features to the Swin Transformer's latent space. $\sigma(\cdot)$ is a non-linear activation, and Z_0 is the projected representation for classification. The Swin Transformer uses SW-MSA to capture feature dependencies (Eq. (15)).

$$Z^{l+1} = LN\left(Z^l + \sum_{h=1}^{H} \alpha_h.softmax\left(\frac{Q_h K_h^T}{\sqrt{d_k}}\right) V_h\right) \tag{15}$$

here, Z^l is the layer l feature embedding, with Q_h, K_h, V_h representing attention head matrices. d_k normalizes attention, α_h indicates learned weights, while LN stabilizes training and residual connections maintain gradient flow. Fig. 8 shows the Swin Transformer architecture, and Algorithm 1 presents our system workflow. The detailed configuration and training parameters of our Swin Transformer implementation are presented in Table 4, specifying the exact architecture and optimization settings used for vehicle classification.

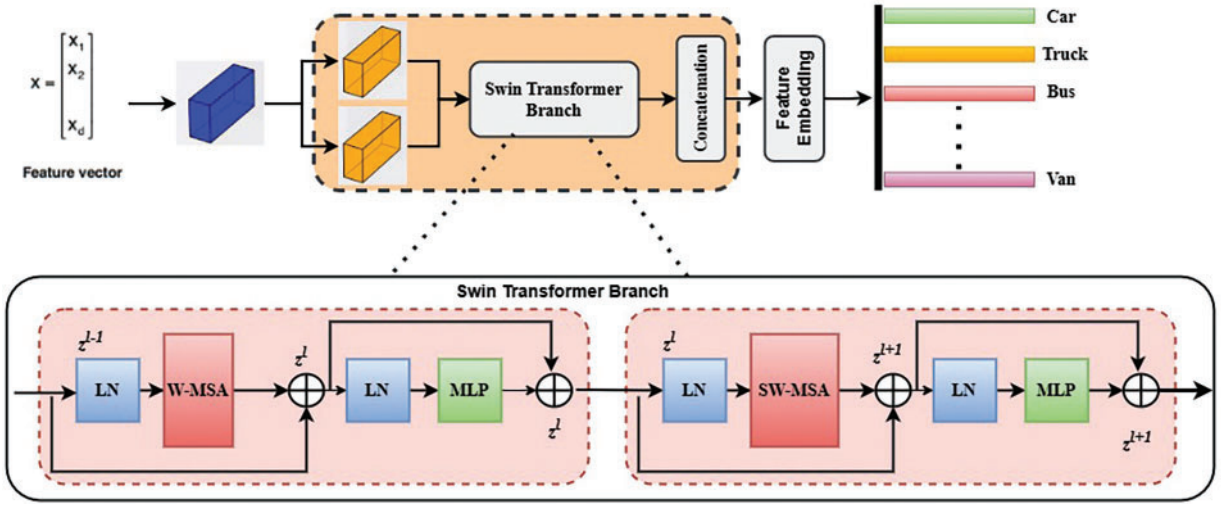


Figure 8: Swin Transformer architecture for vehicle classification depicts hierarchical shifted-window self-attention blocks, patch-merging layers, and the final classification head

Algorithm 1: Proposed approach

Input: Video frames $I = \{I_t\}_{t=1}^{T}$

Output: Classified vehicle labels $\mathcal{L}_t = \{c_{t,i}\}$ for each frame t

1. Preprocessing (MSRCR)

For each t:

$$R_t = MSRCR(I_t) = \sum_{s=1}^{S} w_s \left[\log I_t - \log(F_s * I_t)\right] + \alpha \left[\log(\beta I_t) - \log\left(\sum_{j=1}^{3} I_{t,j}\right)\right]$$

where F_s is a Gaussian of scale σ_s, $w_s \in [0,1]$, $\sum w_s = 1$, $\alpha \in [0,1]$, $\beta > 0$.

Output: enhanced frame R_t.

2. Segmentation (OPTICS)

Convert R_t to grayscale G_t.

Compute reachability distances

$$RD(o,p) = \max\{\alpha\, Var(o),\ \| o - p \|_2\} \ \&\ \hat{\rho}(x) = \sum_{i=1}^{N} \exp\left(- \| x - x_i \|^2 / \left(2\lambda^2\right)\right).$$

Run OPTICS to obtain mask $M_t = \{0,1\}^{H \times W}$.

Output: segmentation mask M_t.

3. Detection (YOLOv10)

$B_t = \{(b_i, p_i)\} = \text{YOLOv10}(R_t)$

where $p_i = \Pr(\text{vehicle} \mid b_i)$.

Filter $\{b_i \mid p_i \geq \tau\}$.

Output: retained bounding boxes B_t.

4. Feature Extraction (GLOH + Dense-SIFT)

For each $b_i \in B_t$, crop region $r_{t,i}$.

Compute: $x_{t,i}^{GLOH} = GLOH(r_{t,i})$, $\quad x_{t,i}^{DSIFT} = DSIFT(r_{t,i})$.

From concatenated descriptor

$\mathbf{x}_{t,i} = [x_{t,i}^{GLOH},\ x_{t,i}^{DSIFT}]$.

Output: feature vectors $\mathbf{x}_{t,i}$.

5. Feature Optimization (WOA)

Initialize population $\{\mathbf{x}^{(j)}\}_{j=1}^{P}$.

Repeat for $t = 1 \ldots T_{\max}$:

$$A = 2a\, r - a, \quad C = 2r, \quad r \sim U(0,1), \quad a = 2\left(1 - \frac{t}{T_{\max}}\right),$$

$$\mathbf{x}^{(j)} \leftarrow \begin{cases} \mathbf{x}^* - A \mid C\,\mathbf{x}^* - \mathbf{x}^{(j)} \mid, & \text{(encircle)} \\ \mid \mathbf{x}^* - \mathbf{x}^{(j)} \mid e^{b\,l} \cos(2\pi l) + \mathbf{x}^*, & \text{(spiral)} \end{cases}$$

Evaluate fitness $f(\mathbf{x})$ via classification error; update $\mathbf{x}^* = \text{argmin} f$.

Output: optimized descriptors $\mathbf{x}_{t,i}^*$.

6. Classification (Swin Transformer)

Embed: $\mathbf{z}_{t,i}^0 = W_e\, \mathbf{x}_{t,i}^* + b_e$.

For each layer ℓ:

$$\hat{\mathbf{z}}^{\updownarrow} = \mathbf{z}^{\updownarrow - 1} + SW\text{-}MSA\left(LN\left(\mathbf{z}^{\updownarrow - 1}\right)\right), \quad \mathbf{z}^{\updownarrow} = \hat{\mathbf{z}}^{\updownarrow} + MLP\left(LN\left(\hat{\mathbf{z}}^{\updownarrow}\right)\right).$$

$$c_{t,i} = \text{argmax} Softmax\left(W_c\, \mathbf{z}_{t,i}^L + b_c\right).$$

Output: labels $c_{t,i}$.

7. Return $\mathcal{L}_t = \{c_{t,i} \mid b_i \in B_t\}$.

Table 4: Swin Transformer configuration and training parameters

Parameter	Value	Description
Feature input dimension	1024	Concatenated and optimized GLOH (512) + Dense-SIFT (512)
Embedding dimension	128	Projection dimension for feature inputs
Depth	4	Number of Transformer blocks
Window size	7×7	For shifted window-based attention
MLP ratio	4.0	Expansion ratio for feed-forward network
Drop path	0.1	Stochastic depth for regularization
Learning rate	5×10^{-4}	With linear warmup (5 epochs) and cosine decay
Batch size	32	Optimized for available GPU memory
Loss function	Cross-Entropy with label smoothing ($\varepsilon = 0.1$)	For multi-class vehicle classification
Training epochs	50	With early stopping (patience = 10)

4 Experimentation and Results

The methodology was implemented in Python 3.8 using advanced deep learning and image processing libraries, including PyTorch 1.10 (YOLOv10-based vehicle detection), OpenCV 4.5 (preprocessing and feature extraction with GLOH and Dense-SIFT), scikit-learn 0.24 (WOA-based feature optimization), and pydensecrf 1.0 (segmentation with OPTICS). Experiments were conducted on an Intel Core i5-12500H (2.50 GHz) processor, 24 GB RAM, and an NVIDIA RTX 3050 GPU (4 GB VRAM). The model demonstrated superior performance in vehicle detection, feature extraction, optimization, and classification across multiple datasets, including VisDrone and UAVDT, with dataset details provided.

4.1 Dataset Description

4.1.1 UAVDT Dataset

The UAVID Dataset [25] contains 4K aerial images (3840 × 2160 px) from UAVs over urban areas, with 42,874 annotated instances of vehicles under diverse conditions. It offers pixel-wise annotations, supporting vehicle detection, classification, and trajectory analysis for aerial surveillance, traffic monitoring, and intelligent transportation research.

4.1.2 VisDrone Dataset

The VisDrone Dataset [26] contains 10,209 images and 8599 video frames from UAVs across urban, suburban, and highway environments, with annotations for 10 object classes. Its comprehensive nature makes it ideal for vehicle detection and classification in autonomous surveillance applications.

4.2 Model Evaluation and Experimental Results

We evaluated our model using 5-fold cross-validation on the VisDrone and UAVDT datasets. Data was partitioned into five equal subsets with preserved class distribution, using 80% for training and 20% for testing in each fold. The process was repeated five times, with each subset serving once as the test set. Results represent average performance across all folds, ensuring reliable generalization estimates across diverse

scenarios. Our model Achieved 95.50% accuracy on UAVDT (Fig. 9) and 92.67% on VisDrone (Fig. 10), with detailed precision, recall, and F1-scores in Tables 5 and 6. State-of-the-art comparisons and computational complexity are presented in Tables 7 and 8.

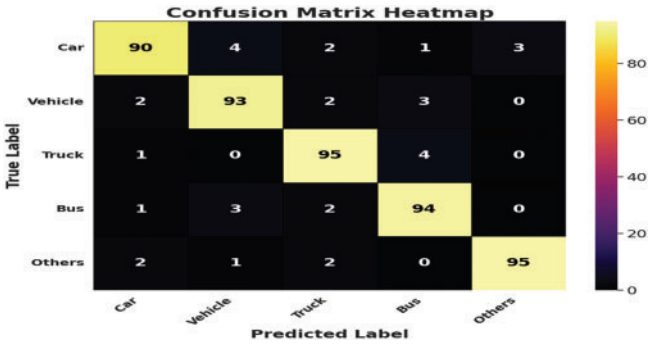


Figure 9: Confusion matrix for UAVDT dataset. Rows represent ground-truth classes; columns show predicted labels. Diagonal values indicate class-wise accuracy, while off-diagonal values highlight misclassifications

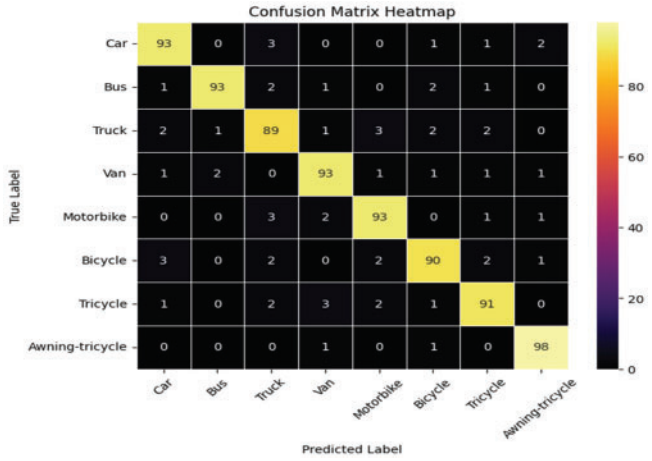


Figure 10: Confusion matrix for VisDrone dataset, class-wise precision (diagonal) and common misclassifications

Table 5: Vehicle detection having, precision, recall, and F1-score evaluation of UAVDT dataset

Classes	Precision	Recall	F1-Score
Car	0.9265	0.9000	0.9183
Vehicle	0.9188	0.9300	0.9254
Truck	0.9346	0.9500	0.9360
Bus	0.9216	0.9300	0.9307
Others	0.9500	0.9504	0.9500

Table 6: Vehicle detection having, precision, recall, and F1-Score evaluation of VisDrone dataset

Classes	Precision	Recall	F1-Score
Car	0.9208	0.9300	0.9254
Bus	0.9688	0.9556	0.9490
Truck	0.8812	0.8900	0.8856
Van	0.9208	0.9300	0.9254
Motorbike	0.9205	0.9301	0.9289
Bicycle	0.9184	0.9000	0.9091
Tricycle	0.9192	0.9100	0.9146
Awning-tricycle	0.9515	0.9611	0.9655
Mean	0.9251	0.9258	0.9254

Table 7: Comparison of proposed model with other state-of-the-art methods

Authors	Mode of detection	Dataset name/No. of samples	Classification	Accuracy
Zhang et al. [27]	Night	Video frames/12,000	SVM classifier	Accuracy = 86.14%
Dong et al. [28]	Night	Bit Vehicle/9850	Half Supervised CNN	Accuracy = 89.4%
Zou et al. [29]	Night	5 different videos of road traffic. Positive frames = 2000 Negative = 6000	AdaBoost classifier	Accuracy = 86.4%
Tu & Du [30]	Different conditions	Camera recording/3425	Neural network	Accuracy = 90.8%
Kuang et al. [31]	Night	Hong Kong nighttime dataset/8794	SVM	Accuracy = 91.85%
Zhang et al. [32]	Nighttime	Custom Night Urban/600 images	SVM	Accuracy = 92.51%
Zhang & Zuo [33]	Nighttime	VisDrone/~10,000 images	YOLOv8	Accuracy = 89.2%
Namana et al. [34]	Nighttime	ExDark/7363 nighttime images	YOLOv8 + BiFPN	Accuracy = 90.8%
Proposed method	Night	UAVDT/10,000 images, VisDrone/11,500 images	Swin Transformer	95.50% (UAVDT)92.67% (VisDrone)

Table 8: Computational complexity analysis

Methods	Computational complexity	Execution time (s)	Memory usage (MB)
Pre-processing (MSRCR)	O(N·M·K)	0.23	150
Segmentation (OPTICS)	$O(N^2)$	0.75	320

(Continued)

Table 8 (continued)

Methods	Computational complexity	Execution time (s)	Memory usage (MB)
Vehicle detection	O(N·C)	0.79	450
Feature extraction (GLOH)	O(N·D)	0.55	120
Feature extraction (DSIFT)	$O(N \cdot S^2)$	0.49	180
Feature optimization (WOA)	O(G·P·D)	1.52	600
Classification	$O(N \cdot d^2)$	0.87	385

Table 9 presents the detection-specific metrics for YOLOv10 on both datasets. These metrics evaluate the model's ability to correctly localize vehicles in nighttime imagery before classification occurs.

Table 9: YOLOv10 detection performance on UAVDT and VisDrone datasets

Dataset	Precision	Recall	F1-Score	mAP@0.5	mAP@0.75	mAP@0.5:0.95
UAVDT	0.941	0.923	0.932	0.915	0.831	0.762
VisDrone	0.927	0.906	0.916	0.897	0.803	0.735

The detection metrics demonstrate that YOLOv10 achieves strong localization performance even in challenging nighttime conditions, with mAP@0.5 values of 0.915 and 0.897 for UAVDT and VisDrone datasets, respectively. Table 10 presents an ablation study quantifying how each component contributes to overall classification accuracy on both datasets.

Table 10: Ablation study of the proposed nighttime UAV vehicle classification pipeline

Experiment	MSRCR	OPTICS	YOLOv10	GLOH	Dense-SIFT	WOA	Classifier	UAVDT Acc (%)	VisDrone Acc (%)
Full pipeline (Baseline)	✓	✓	✓	✓	✓	✓	✓	95.50	92.67
Without MSRCR	✗	✓	✓	✓	✓	✓	✓	92.80	90.40
Without OPTICS	✓	✗	✓	✓	✓	✓	✓	92.20	89.80
Without YOLOv10	✓	✓	✗	✓	✓	✓	✓	89.00	85.50
Without GLOH	✓	✓	✓	✗	✓	✓	✓	92.10	89.30

(Continued)

Table 10 (continued)

Experiment	MSRCR	OPTICS	YOLOv10	GLOH	Dense-SIFT	WOA	Classifier	UAVDT Acc (%)	VisDrone Acc (%)
Without Dense-SIFT	✓	✓	✓	✓	✗	✓	✓	91.90	89.10
Without WOA	✓	✓	✓	✓	✓	✗	✓	91.30	88.50
Simple CNN (instead of Swin)	✓	✓	✓	✓	✓	✓	ResNet-based CNN	90.00	87.00

Ablation study shows each component's importance. Removing MSRCR causes accuracy drops of 2.7%/2.3% (UAVDT/VisDrone), while disabling OPTICS segmentation reduces performance by 3.3%/2.9%. YOLOv10 removal produces the largest decline (6.5%/7.2%), highlighting its detection importance. Eliminating GLOH or Dense-SIFT individually decreases accuracy by ~3.5%, while skipping WOA causes a 4.2% drop on both datasets. Replacing the Swin Transformer with a CNN significantly reduces performance (5.5%/5.7%), demonstrating the value of hierarchical self-attention for complex pattern modeling.

Table 7 demonstrates our Swin Transformer-based model outperforms existing approaches including SVM, AdaBoost, and CNN-based methods, highlighting its superior robustness and precision for nighttime vehicle classification across diverse aerial datasets.

The computational complexity of the approach is analyzed across key stages, balancing accuracy and efficiency. Preprocessing (MSRCR) runs at O(N·M·K), ensuring fast enhancement. OPTICS segmentation has quadratic complexity $O(N^2)$, increasing memory usage. YOLOv10 detection runs at O(N·C), influenced by class count. Feature extraction methods (Dense-SIFT: $O(N \cdot S^2)$, GLOH: O(N·D)) depend on keypoint density. WOA optimization runs at O(G·P·D). Finally, Swin Transformer classification operates at $O(N \cdot d^2)$, ensuring computational feasibility.

4.3 Real-Time Feasibility and Computational Constraints

Though highly accurate offline, our framework's real-time UAV deployment requires strategic optimization. YOLOv10 detection and feature extraction run efficiently onboard (18–22 FPS on embedded GPUs with quantization), while compute-intensive processes operate as post-processing on ground stations. Future work will focus on edge computing optimizations for complete real-time operation.

Computational Efficiency and Practical Deployment Considerations

Our framework's computational requirements, detailed in Table 8, indicate that the complete pipeline processes frame at approximately 0.21 FPS on our test hardware (Intel Core i5-12500H, NVIDIA RTX 3050 GPU). This processing rate presents challenges for real-time UAV applications, necessitating careful consideration of optimization strategies. Based on our theoretical analysis and computational complexity assessment, we propose several approaches to improve real-time performance. The estimated performance gains achieved through various optimization strategies are summarized in Table 11.

Table 11: Estimated performance improvements with optimization strategies

Optimization strategy	Approach	Estimated performance	Theoretical impact on accuracy
Algorithm simplification	Replace OPTICS ($O(N^2)$) with region-based segmentation ($O(N)$)	~0.8 FPS	3%–5% reduction
Resolution reduction	Down sampling to 640 × 480 with preprocessing	~0.95 FPS	4%–6% reduction
Feature selection	Pre-computed feature maps with reduced dimensionality	~0.75 FPS	2%–3% reduction
Combined approach	Integration of all above optimizations	~1.2–1.5 FPS	7%–10% reduction

With proper optimization, our framework could achieve 1–2 FPS performance—suitable for applications like periodic traffic monitoring and surveillance where accuracy outweighs speed. For UAV deployment, we estimate requirements of 4 GB GPU memory, 15–20 W power budget, and adequate thermal management. Although we haven't conducted field tests, modern embedded platforms should handle optimized versions of our framework for specific applications.

4.4 Limitations and Trade-Offs

Although the model demonstrated improved performance, our approach has significant drawbacks. The computational needs of OPTICS segmentation ($O(N^2)$) and WOA optimization result in a trade-off between accuracy and speed, with processing speeds of ~0.2 fps. MSRCR performance decreases in extremely low-light circumstances (<2 lux), especially for little vehicles. Heavy occlusions in congested traffic areas offer difficulties, whereas minor occlusions are more manageable. Rare vehicle types may be misclassified into similar groups. Despite the durability of MSRCR, adverse weather lowers contrast and distorts looks. Models trained in urban areas must be recalibrated for rural or highway deployment due to differences in illumination patterns and vehicle distribution. Future research will concentrate on further low-light enhancement, occlusion-aware identification, and optimized implementations for better real-time performance.

5 Conclusion

This research introduces a multi-stage vehicle detection and classification framework optimized for nighttime aerial imagery. The proposed six-stage pipeline effectively addresses the challenges of low illumination, noise, and occlusions in UAV-based surveillance by integrating MSRCR preprocessing, OPTICS segmentation, YOLOv10 detection, and GLOH/Dense-SIFT feature extraction with WOA optimization and Swin Transformer classification. Experimental validation on the UAVDT and VisDrone datasets demonstrates the framework's effectiveness, achieving classification accuracies of 95.50% and 92.67% respectively, outperforming state-of-the-art approaches in precision, recall, and F1-score metrics particularly for challenging nighttime scenarios. Future work will focus on enhancing computational efficiency for real-time deployment, improving performance in extreme low-light conditions, and integrating advanced tracking

models for enhanced vehicle trajectory analysis in aerial surveillance applications. Additional research on domain adaptation techniques would improve generalization across varied environments and lighting conditions, further advancing the practical application of UAV-based nighttime traffic monitoring systems.

Acknowledgement: This work was supported through Princess Nourah bint Abdulrahman University Researchers Supporting, Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Funding Statement: This work was supported through Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2025R508), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Author Contributions: Study conception and design: Abdulwahab Alazeb and Dina Abdulaziz AlHammadi; data collection: Muhammad Hanzla and Naif Al Mudawi; analysis and interpretation of results: Mohammed Alshehri and Haifa F. Alhasson; funding: Dina Abdulaziz AlHammadi and Haifa F. Alhasson; draft manuscript preparation: Ahmad Jalal. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: All publicly available datasets are used in the study.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Nguyen Anh DD, Thai BN, Hoang AN, Nguyen KD. Aerial vehicle detection at night: a synthesized dataset and performance evaluation of state-of-the-art object detection models. In: Proceedings of the 2024 13th International Conference on Control, Automation and Information Sciences (ICCAIS); 2024 Nov 26–28; Ho Chi Minh City, Vietnam. p. 1–6.
2. Bilal M, Rehmat S, Akhtar N, Gulzar H. Cost-effective drone detection with deep neural networks for day and night surveillance. In: Proceedings of the 2024 International Conference on Frontiers of Information Technology (FIT); 2024 Dec 9–10; Islamabad, Pakistan. p. 1–6.
3. Pan M, Xia W, Yu H, Hu X, Cai W, Shi J. Vehicle detection in UAV images via background suppression pyramid network and multi-scale task adaptive decoupled head. Remote Sens. 2023;15(24):5698. doi:10.3390/rs15245698.
4. Alahvirdi D, Tuci E. Autonomous traffic monitoring and management by a simulated swarm of UAVs. In: Proceedings of the 2023 11th RSI International Conference on Robotics and Mechatronics (ICRoM); 2023 Dec 19–21; Tehran, Iran. p. 91–6.
5. Liu C, Ding Y, Zhu M, Xiu J, Li M, Li Q. Vehicle detection in aerial images using a fast oriented region search and the vector of locally aggregated descriptors. Sensors. 2019;19(15):3294. doi:10.3390/s19153294.
6. Hamadi R, Ghazzai H, Massoud Y. Image-based automated framework for detecting and classifying unmanned aerial vehicles. In: Proceedings of the 2023 IEEE International Conference on Smart Mobility (SM); 2023 Mar 19–21; Thuwal, Saudi Arabia. p. 149–53.
7. Abro GEM, Zulkifli SABM, Masood RJ, Asirvadam VS. Comprehensive review of UAV detection, security, and communication advancements to prevent threats. Drones. 2022;6(10):284. doi:10.3390/drones6100284.
8. Seidaliyeva U, Ilipbayeva L, Taissariyeva K, Smailov N. Advances and challenges in drone detection and classification techniques: a state-of-the-art review. Sensors. 2023;24(1):125. doi:10.3390/s24010125.
9. Singhal N, Prasad L. Sensor-based vehicle detection and classification: a systematic review. Int J Eng Syst Model Simul. 2022;14(2):87–103.
10. Teixeira K, Miguel G, Silva HS, Madeiro F. A survey on applications of unmanned aerial vehicles using machine learning. IEEE Access. 2023;11:12245–60. doi:10.1109/access.2023.3326101.
11. Ahmed M, Sumon MRA, Sutradhar U. System design for ML-based detection of unauthorized UAVs and integration within the UTM framework. In: Proceedings of the 2024 IEEE 29th Asia Pacific Conference on Communications (APCC); 2024 Nov 5–7; Bali, Indonesia. p. 1–5.

12. Rangkuti AH, Athala VH. Development of vehicle detection and counting systems with UAV cameras: deep learning and Darknet algorithms. J Image Graph. 2023;11(3):248–59. doi:10.18178/joig.11.3.248-262.
13. Pavel MI, Tan SY, Abdullah A. Vision-based autonomous vehicle systems based on deep learning: a systematic literature review. Appl Sci. 2022;12(14):6831. doi:10.3390/app12146831.
14. Ragab M, Abdushkour HA, Khadidos AO. Improved deep learning-based vehicle detection for urban applications using remote sensing imagery. Remote Sens. 2023;15(19):4747. doi:10.3390/rs15194747.
15. Misbah M, Khan MU, Yang Z, Kaleem Z. TF-NET: deep learning empowered tiny feature network for nighttime UAV detection. In: Proceedings of the International Conference on Wireless and Satellite Systems; 2023 Mar 12–13; Cham, Switzerland: Springer Nature; 2023. p. 1–10.
16. Carion N. End-to-end object detection with transformers. In: Proceedings of the European Conference on Computer Vision; 2020 Aug 23–28; Glasgow, UK. Berlin/Heidelberg, Germany: Springer. p. 213–29.
17. Chen R, Li D, Gao Z, Kuai Y, Wang C. Drone-based visible-thermal object detection with transformers and prompt tuning. Drones. 2024;8(9):451. doi:10.3390/drones8090451.
18. Almujally NA, Qureshi AM, Alazeb A, Rahman H, Sadiq T, Alonazi M, et al. A novel framework for vehicle detection and tracking in night ware surveillance systems. IEEE Access. 2024;12:88075–85. doi:10.1109/access.2024.3417267.
19. Li Y, Chen Z, Sun W. An image enhancement algorithm based on multi-scale Retinex theory to improve the images quality of sensors. In: Proceedings of the MIPPR 2023: Pattern Recognition and Computer Vision; 2023 Nov 10–12; Wuhan, China. Bellingham, WA, USA: SPIE; 2024. p. 92–103.
20. Wang Y, Lv H, Deng R, Zhuang S. A comprehensive survey of optical remote sensing image segmentation methods. Can J Remote Sens. 2020;46(5):501–31. doi:10.1080/07038992.2020.1805729.
21. Sundaresan Geetha A, Alif MAR, Hussain M, Allen P. Comparative analysis of YOLOv8 and YOLOv10 in vehicle detection: performance metrics and model efficacy. Vehicles. 2024;6(3):1364–82. doi:10.3390/vehicles6030065.
22. Zhao Z. Robust region feature extraction with salient MSER and segment distance-weighted GLOH for remote sensing image registration. IEEE J Sel Top Appl Earth Obs Remote Sens. 2024;17(5):2475–88. doi:10.1109/jstars.2023.3344474.
23. Sabry ES, Elagooz SS, El-Samie FE, El-Bahnasawy NA, El-Banby GM, Ramadan RA. Evaluation of feature extraction methods for different types of images. J Opt. 2023;52(2):716–41. doi:10.1007/s12596-022-01024-6.
24. Sun Z, Liu C, Qu H, Xie G. A novel effective vehicle detection method based on Swin Transformer in hazy scenes. Mathematics. 2022;10(13):2199. doi:10.3390/math10132199.
25. Li X, Li X, Li Z, Xiong X, Khyam MO, Sun C. Robust vehicle detection in high-resolution aerial images with imbalanced data. IEEE Trans Artif Intell. 2021;2(3):238–50. doi:10.1109/tai.2021.3081057.
26. Cao Y, He Z, Wang L, Wang W, Yuan Y, Zhang D, et al. VisDrone-DET2021: the vision meets drone object detection challenge results. In: Proceedings of the IEEE/CVF International Conference on Computer Vision; 2021 Oct 11–17; Montreal, BC, Canada. p. 2847–54.
27. Zhang RH, You F, Chen F, He WQ. Vehicle detection method for intelligent vehicle at nighttime based on video and laser information. Int J Pattern Recognit Artif Intell. 2018;32(4):1850009. doi:10.1142/s021800141850009x.
28. Dong Z, Pei M, He Y, Liu T, Dong Y, Jia Y. Vehicle type classification using unsupervised convolutional neural network. In: Proceedings of the 22nd International Conference on Pattern Recognition; 2015 Jan 10–12; Lisbon, Portugal. p. 313–16.
29. Zou Q, Ling H, Luo S, Huang Y, Tian M. Robust nighttime vehicle detection by tracking and grouping headlights. IEEE Trans Intell Transp Syst. 2015;16(5):2838–49. doi:10.1109/tits.2015.2425229.
30. Tu C, Du S. A Hough space feature for vehicle detection. In: Advances in Visual Computing: 13th International Symposium, ISVC 2018; 2018 Nov 19–21; Las Vegas, NV, USA; 2018. p. 147–56.
31. Kuang H. Feature selection based on tensor decomposition and object proposal for night-time multiclass vehicle detection. IEEE Trans Syst Man Cybern Syst. 2019;49(1):71–80. doi:10.1109/tsmc.2018.2872891.
32. Zhang L, Xu W, Shen Huang CY. Vision-based on-road nighttime vehicle detection and tracking using improved HOG features. Sensors. 2024;24:1590. doi:10.3390/s24051590.

33. Zhang X, Zuo G. Small target detection in UAV view based on improved YOLOv8 algorithm. Sci Rep. 2025;15(1):421. doi:10.1038/s41598-024-84747-9.
34. Namana MSK, Kumar BU. An efficient and robust night-time surveillance object detection system using YOLOv8 and high-performance computing. Int J Saf Secur Eng. 2024;14(6):1763–73. doi:10.18280/ijsse.140611.