Ad Hoc Networks 196 (2027) 104459

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

# A geo-aware UAV system for detecting non-compliant parking in smart cities☆

Chung Yi Lai [a], Hudyjaya Siswoyo Jo [a], Jonathan Kua [b,*], Chean Hung Lai [a], Michelle Dunn [c]

[a] School of Engineering and Science, Swinburne University of Technology Sarawak Campus, Kuching, Sarawak, 93350, Malaysia
[b] School of Information Technology, Deakin University, Geelong, VIC, 3220, Australia
[c] School of Engineering, Swinburne University of Technology, Melbourne, VIC, 3122, Australia

ARTICLE INFO

*Keywords:*
Unmanned Aerial Vehicle (UAV)
YOLOv8
Object detection
Geolocation
Non-compliant parking

ABSTRACT

Non-compliant parking in open parking facilities remains a persistent challenge in urban environments, reducing parking efficiency, obstructing access, and undermining the reliability of smart parking systems. Existing vision-based monitoring approaches predominantly rely on fixed camera infrastructure, which is limited in spatial coverage and often lacks location-referencing capabilities for parking violations. This paper proposes a geo-aware system based on an Unmanned Aerial Vehicle (UAV) for detecting non-compliant parking using geo-tagged imagery. The proposed system integrates autonomous UAV patrol, frame-level UAV geolocation extraction, and deep learning-based object detection to identify vehicles parked outside designated spaces and associate each detection event with the corresponding UAV telemetry data. A processing pipeline synchronizes video frames with telemetry data, enabling the generation of geo-referenced outputs for enforcement and navigation purposes. The system is evaluated through real-world experiments conducted in an open parking facility, using a YOLOv8-based detection model trained on a custom annotated dataset. Experimental results show that the system achieves an accuracy of 99.56% with zero observed false positives under the tested conditions (evaluated dataset and post-processing pipelines), while providing reliable location-referencing information for detected parking violations. The proposed approach enables mobile, geo-aware sensing of parking facilities, addresses key limitations of static monitoring systems, and provides a practical solution for improving parking compliance, reducing operational inefficiencies, and supporting sustainable urban mobility management.

## 1. Introduction

Non-compliant parking in urban environments remains a persistent challenge for efficient parking management in cities that increasingly adopt paradigms towards autonomous Industrial Cyber-Physical Systems (ICPS) [1,2]. Recent advances in UAV-enabled ICPS have demonstrated the effectiveness of autonomous aerial platforms in intelligent logistics and other industrial applications, highlighting their potential for real-time monitoring and decision-making in complex environments [3,4].

Vehicles parked outside designated spaces obstruct access, reduce parking availability, and disrupt traffic flow. These issues degrade user experience and introduce inefficiencies in parking operations. As cities adopt smart parking systems to improve utilization and accessibility, non-compliant parking continues to limit the effectiveness of these solutions, particularly in vision-based monitoring systems [5–10].

Vision-based parking monitoring systems are widely used to detect parking occupancy and support vehicle identification through license plate recognition. These systems rely on visual data to determine space availability and track vehicle presence. However, existing studies [8] highlight a critical limitation. Non-compliant parking introduces visual obstructions that reduce detection accuracy and compromise system reliability. Obstructed views can prevent correct identification of parking spaces and license plates. This limitation affects both operational performance and enforcement capability. Despite this challenge, current approaches have not provided effective solutions to detect and manage non-compliant parking within parking facilities.

☆ This article is part of a Special issue entitled: 'Intelligent and Resilient Networking' published in Ad Hoc Networks.

* Corresponding author.
*E-mail addresses:* cylai@swinburne.edu.my (C.Y. Lai), hsiswoyo@swinburne.edu.my (H.S. Jo), jonathan.kua@deakin.edu.au (J. Kua), clai@swinburne.edu.my (C.H. Lai), jdunn@swin.edu.au (M. Dunn).

https://doi.org/10.1016/j.adhoc.2026.104459
Received 14 April 2026; Received in revised form 18 September 2026; Accepted 17 October 2026
Available online 27 October 2026
1570-8705/© 2026 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Most existing methods [11–13] rely on fixed camera infrastructure. These systems can detect parking violations within predefined regions, but they lack flexibility in coverage and often provide limited location-referencing information for detected violations. Fixed viewpoints also limit the ability to capture detailed visual information such as front-facing vehicle features. As a result, enforcement actions may be delayed or require manual verification. In large parking facilities, these limitations become more pronounced due to spatial scale and occlusions.

Mobile sensing offers a promising alternative. A moving camera platform can adapt its viewpoint, extend coverage, and capture spatial context across the parking facility [14]. This capability enables the detection of non-compliant vehicles while providing location-referencing information associated with the detection event. Such location information can assist parking enforcement personnel in identifying the area of a violation and supports integration with navigation systems.

This paper proposes a geo-aware system based on an Unmanned Aerial Vehicle (UAV) for detecting non-compliant parking in open parking facilities. The system integrates autonomous UAV patrol, geo-tagged image acquisition, and deep learning-based object detection to identify vehicles parked outside designated spaces and associate each detection event with the corresponding UAV telemetry data. The proposed approach enables mobile, high-coverage sensing and provides location-referencing information for parking violations. The novelty of this work is in the development and experimental validation of a practical geo-aware UAV framework specifically designed for detecting non-compliant parking in open parking facilities. This work aims to improve the reliability of parking monitoring systems and enhance enforcement efficiency in urban environments.

This paper makes the following contributions:

- A geo-aware UAV-based system for detecting non-compliant parking in open parking facilities, integrating autonomous aerial sensing with vision-based detection and event geo-tagging capabilities.
- A processing pipeline that synchronizes geo-tagged video frames with telemetry data, enabling association of detected parking violations with the corresponding UAV telemetry record and supporting navigation for enforcement actions.
- A real-world experimental evaluation demonstrating the effectiveness of the proposed system in detecting double-parking violations in a practical parking environment, with high accuracy and reliable performance for parking enforcement applications.

The remainder of this paper is organized as follows. Section 2 reviews related work on vision-based parking monitoring and non-compliant parking detection. Section 3 presents the proposed UAV-based system and its processing pipeline. Section 4 discusses the experimental setup and evaluation results. Section 5 concludes the paper and outlines directions for future work.

## 2. Background and related work

This section reviews vision-based parking monitoring systems and examines their limitations in the presence of non-compliant parking. It then discusses existing vision-based approaches for detecting non-compliant vehicles, with a focus on their ability to mitigate obstruction-related accuracy degradation.

### 2.1. Vision-based parking monitoring systems

Vision-based parking monitoring systems have gained significant attention due to their ability to detect parking occupancy and extract additional information such as license plate details [15,16]. These systems typically rely on either stationary cameras deployed across parking facilities or mobile platforms that capture images from varying viewpoints.

A drone-based approach for detecting vacant parking spaces was proposed in Yuske and Mbaitiga [5], where low-altitude images were processed using color feature extraction and template matching. The method achieved a detection rate of 90%, but the evaluation was limited to simple scenarios and focused only on vacancy detection. A UAV-based monitoring system using geo-tagged imagery and YOLOv8 was introduced in Lai et al. [6]. The system records spatial and temporal attributes for each detected parking space and supports an integrated parking management framework.

Several studies have combined occupancy detection with vehicle identification. A drone-based system with convolutional neural networks for occupancy detection and license plate recognition was presented in Sarkar et al. [7]. High detection accuracy was reported under controlled conditions, but the implications for real-world deployment were not discussed. A UAV-based automatic license plate recognition system was investigated in Ye et al. [9], demonstrating strong performance under favorable imaging conditions.

Alternative sensing platforms have also been explored. A smartphone-based parking monitoring system using geo-tagged images and Detectron2 was proposed in Mots'oehli and Yang [8]. The system identifies parking spaces and supports navigation using GPS coordinates. A parking space detection method using fish-eye cameras and YOLOv3 was introduced in Xu and Hu [10], achieving high accuracy on a custom dataset.

Despite these advances, several limitations remain. Many studies evaluate performance under controlled conditions and do not account for real-world complexities such as occlusions, dynamic objects, and irregular parking behavior. Visual obstructions reduce detection accuracy and affect system reliability. Non-compliant parking further exacerbates this issue by blocking parking spaces and obscuring vehicle features required for identification. These limitations highlight the need for more robust sensing approaches that can operate effectively in realistic and secure (through trusted edge computing) parking environments [17–19].

### 2.2. Vision-based non-compliant parking detection

The previous subsection highlighted the limitations of vision-based parking monitoring systems in the presence of visual obstructions. Non-compliant parking represents a major source of such obstructions, yet it has received limited attention in the context of parking facilities. This subsection reviews vision-based approaches for detecting non-compliant parking and evaluates their applicability to parking environments.

A convolutional neural network-based detection system for outdoor parking violations was proposed in Ng et al. [11]. The method defines restricted regions within a parking area and identifies vehicles located within these regions as non-compliant. The system achieved an accuracy of 94%, but the evaluation was conducted in a simplified setup. The approach relies on fixed high-angle cameras and cannot capture detailed vehicle features such as license plates.

A double-parking detection algorithm based on background subtraction techniques was presented in Fadilah et al. [12]. Vehicles that remain stationary within restricted zones are classified as non-compliant. The method achieved an accuracy of 91%, but its performance degraded under challenging lighting conditions. Similar limitations were observed in a detection approach using Gaussian Mixture Models and Kalman filtering, as introduced in Akhawaji et al. [13]. This method also depends on fixed camera views and is sensitive to environmental variations.

Deep learning-based detection approaches have also been explored. A vehicle detection system using the Single Shot MultiBox Detector was proposed in Xie et al. [20] for highway scenarios. Vehicles that remain stationary over a defined duration are classified as non-compliant. Although high detection accuracy was reported, the method does not

distinguish between compliant and non-compliant parking in structured parking environments. A YOLOv3-based detection and tracking system was introduced in Liu et al. [21], where stationary vehicles are identified and license plate recognition is applied. However, the method does not clearly define criteria for differentiating compliant and non-compliant parking.

More recent work has explored improved labeling and classification strategies. An approach based on in-vehicle cameras and YOLOv3 was proposed in Peng et al. [22]. The method introduces a labeling scheme that captures multiple parking conditions within a single bounding box and enables classification of different types of non-compliant parking behavior. The system achieved an F-score of 90.66% and demonstrated the ability to distinguish between compliant and non-compliant parking.

Despite these advances, several limitations remain. Most approaches rely on fixed camera infrastructure and predefined restricted regions, which limits flexibility and scalability. These systems often lack location-referencing information for detected vehicles, which restricts their usefulness for enforcement applications [23,24]. In addition, many methods are evaluated under controlled conditions and do not consider the impact of occlusions and complex parking scenarios. Although some approaches can distinguish between compliant and non-compliant parking, they do not provide sufficient spatial information for navigation or response.

These limitations highlight the need for a mobile and geo-aware sensing approach that can detect non-compliant parking and provide practical location-referencing information in real-world parking environments.

### 2.3. Research gaps and problem formulation

The preceding review highlights several limitations in existing vision-based parking monitoring and detection systems. These limitations restrict their effectiveness in real-world parking environments and motivate the need for more robust solutions, as follows:

(i) Most vision-based parking monitoring systems are designed to detect occupancy under controlled conditions. They do not account for visual obstructions caused by non-compliant parking, which can significantly degrade detection accuracy and system reliability. This limitation reduces the effectiveness of automated parking management systems in practical deployments.

(ii) Many existing approaches rely on fixed camera infrastructure with predefined monitoring regions. These systems lack flexibility in coverage and cannot adapt to dynamic parking environments. Fixed viewpoints also limit the ability to capture detailed visual information, particularly in scenarios where vehicles obstruct each other.

(iii) Current methods for detecting non-compliant parking often depend on manually defined restricted areas or temporal rules such as vehicle immobility. While these approaches can identify violations, they do not generalize well to complex parking layouts and may fail in the presence of occlusions or irregular parking behavior.

(iv) Most existing systems do not provide location referencing information for detected parking violations. The absence of actionable location information limits their usefulness for enforcement, as parking personnel require location context to respond efficiently. Although detection methods with strong classification performance have been reported in Peng et al. [22], they do not support association of detected violations with location information. In contrast, geo-tagging techniques for object localization have been explored in Lai et al. [6],Mots'oehli and Yang [8], but these approaches focus on parking occupancy rather than non-compliant parking detection. Integrating non-compliant parking detection with practical location-referencing capabilities remains relatively unexplored.

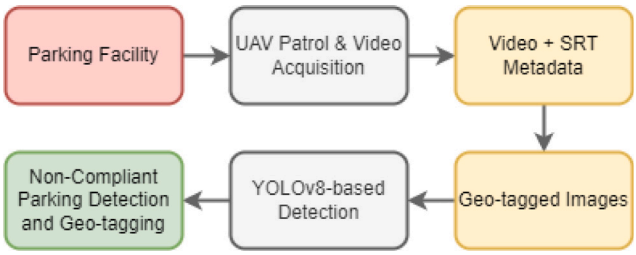



**Fig. 1.** System overview of the proposed solution.

To address these gaps, this paper proposes a geo-aware system based on an Unmanned Aerial Vehicle (UAV) for detecting non-compliant parking in open parking facilities. The proposed system leverages mobile sensing to extend spatial coverage, integrates geo-tagged imagery to provide location-referencing information for detected violations, and employs deep learning-based detection to identify non-compliant parking under realistic conditions. This approach aims to improve the robustness, scalability, and practical utility of parking monitoring systems in urban environments.

## 3. Methodology

This section presents the proposed geo-aware UAV-based system for detecting non-compliant parking in open parking facilities. The system integrates UAV-based video acquisition, frame-level geo-tagging and vision-based detection to identify non-compliant vehicles and associate detected violation events with the corresponding UAV telemetry data. The overall processing workflow of the proposed system is illustrated in Fig. 1.

### 3.1. Parking lot experimental venue

The proposed system was evaluated at the parking facilities of Swinburne University of Technology Sarawak Campus, Malaysia. The parking layout consists of perpendicular parking spaces with vehicles typically reverse parked.

The selected venue exhibits frequent non-compliant parking behavior commonly referred to as double parking, whereby a vehicle is parked adjacent to and directly in front of a legitimately parked vehicle, obstructing its access and egress. This behavior is often observed during peak demand periods when parking capacity is constrained [25]. The resulting obstruction not only degrades the usability of the parking facility but also presents challenges for automated parking monitoring systems. Specifically, the double-parked vehicle may partially or fully occlude the parked vehicle, reducing the visibility of key features required for accurate occupancy detection.

The characteristics of this environment provide a realistic testbed for evaluating the robustness of the proposed system under practical conditions.

### 3.2. UAV deployment

A UAV-based sensing approach similar to that in Lai et al. [6] is adopted for aerial data acquisition. A DJI Mavic 2 Enterprise [26], equipped with a 12-megapixel camera and an onboard geolocation system, is used to patrol the parking facility.

The UAV captures video data of parking rows to support detection of non-compliant parking and facilitate downstream license plate identification. The system is designed to complement parking enforcement workflows that require front-facing vehicle information.

The flight path is predefined using the waypoint mission function in the DJI Pilot application installed on the remote controller. The trajectory follows a row-by-row scanning pattern, where each row of




**Fig. 2.** An image taken for frame 430 in a video.

```
430
00:00:14,312 --> 00:00:14,346
<font size="36">FrameCnt: 430, DiffTime: 34ms
2024-04-23 16:07:16,597,973
[iso : 100] [shutter : 1/200.0] [fnum : 280] [ev : 0] [ct : 5983] [color_md : default]
[focal_len : 240] [dzoom_ratio: 10000, delta:0],[latitude: 1.531252] [longtitude:
110.357109] [rel_alt: 3.464 abs_alt: 77.144] [Drone: Yaw:0.0, Pitch:0.2, Roll:0.3] </font>
```

**Fig. 3.** Subtitle file showing details of frame 430 in a video.

parking spaces is traversed before moving to the next. This approach is conceptually similar to path optimization strategies such as the Traveling Salesman problem [27], ensuring efficient coverage of the parking area.

The UAV speed is fixed at 1 m/s to maintain image clarity during motion, which is critical for reliable object detection. The flight altitude is set between 3 m and 4 m. Empirical observations indicate that an altitude of 3.5 m provides the best trade-off between image quality and operational safety. Higher altitudes reduce image resolution, while lower altitudes increase the risk of collision with obstacles. The gimbal is adjusted to a 45-degree angle to capture both parking spaces and potential obstructions within the field of view.

The UAV is manually launched before autonomous operation begins. After acquiring a stable GPS signal, the predefined flight path is uploaded and executed. The UAV then follows the designated route and returns to the home position upon completion of the patrol.

All UAV flights were conducted only after a stable GPS lock had been established and in accordance with applicable operational and safety requirements. To protect privacy, vehicle license plates appearing in manuscript figures have been anonymized through blurring.

### 3.3. Video footage

The selected parking facility consists of multiple rows of parking spaces. The UAV records video data for each row during the patrol, resulting in multiple video segments upon completion of the flight path.

Fig. 2 illustrates a representative frame captured by the UAV. Each video is accompanied by a subtitle file in SubRip Text (SRT) format, which stores frame-level metadata. This metadata includes information such as timestamp, latitude, longitude, altitude, and camera orientation. The SRT file is automatically generated by the UAV and recorded concurrently with the video stream. In addition to telemetry information, each SRT entry contains a FrameCnt identifier that corresponds to the recorded video frame, enabling direct frame-to-metadata association without requiring additional synchronization or timestamp interpolation. Fig. 3 shows an example of the recorded subtitle data corresponding to a video frame.

All recorded videos are stored on a Wi-Fi Secure Digital (SD) card within the UAV. After the UAV completes its flight and returns to the base location, an edge device connects to the Wi-Fi network created by the SD card. The recorded video and corresponding subtitle files are then transferred to the edge device for further processing.

### 3.4. Acquiring geo-tagged images

The recorded video files and their corresponding subtitle files are processed on an edge device to generate geo-tagged image data. A Python-based processing pipeline is developed to associate each video frame with its corresponding geolocation and metadata.

The subtitle file is stored in SubRip Text (SRT) format and contains frame-level information including frame index, timestamp, latitude, longitude, relative altitude, and camera orientation. This information is first parsed and structured into a searchable format to enable efficient retrieval during frame processing.

The video file is then processed sequentially. Each frame is extracted and matched with the corresponding metadata entry using the frame identifier recorded in the SRT file. This process establishes a direct mapping between visual content and geospatial information.

The metadata corresponding to a video frame is retrieved using the frame identifier recorded in the SRT file. This relationship can be expressed as

$$M_t = F(FrameCnt_t) \tag{1}$$

where $FrameCnt_t$ denotes the frame identifier for frame $t$, $F(\cdot)$ represents the metadata lookup function, and $M_t$ is the corresponding metadata record containing information such as latitude, longitude, altitude, timestamp and camera orientation.

The geo-tagged image is then represented as

$$G_t = (I_t, M_t) \tag{2}$$

where $I_t$ denotes the image frame and $G_t$ represents the resulting geo-tagged image consisting of the visual content and its associated metadata.

As a result, each processed frame is associated with geographic coordinates and contextual metadata, forming a set of geo-tagged images. These geo-tagged images serve as the input to the object detection module and enable subsequent association of detected parking violations with the UAV location recorded at the time of image capture.

### 3.5. Development of object detection algorithm

Geo-tagged images generated from the previous stage are processed on the edge device using a deep learning-based object detection model. The YOLOv8 framework developed by Ultralytics[1] is employed due to its high detection accuracy and real-time inference capability.

The model is designed to detect two annotated classes: *Occupied* and *Illegal*, where the *Illegal* class represents non-compliant parking. Although both classes correspond to vehicles, they differ in spatial context. The *Occupied* class refers to vehicles parked within designated boundaries, while the *Illegal* class refers to vehicles positioned outside these boundaries and potentially obstructing access. The model is trained to distinguish between these spatial configurations.

#### 3.5.1. Annotated dataset

The annotated dataset used for model training is obtained from Roboflow Universe.[2] Roboflow provides tools for image annotation, preprocessing, and dataset management. Images are labeled using bounding boxes to indicate object locations and class categories.

The dataset contains two annotated classes, *Occupied* and *Illegal*, where the *Illegal* class represents non-compliant parking as defined in this study. For the *Occupied* class, bounding boxes are drawn to include both the parked vehicle and the corresponding parking space markings. For the *Illegal* class, bounding boxes are drawn around vehicles positioned outside designated parking spaces. The labeling strategy follows guidelines for outdoor surveillance data annotation [28].

[1] https://docs.ultralytics.com/

[2] https://universe.roboflow.com/chung-yi-lai/illegal-parking-swinburne

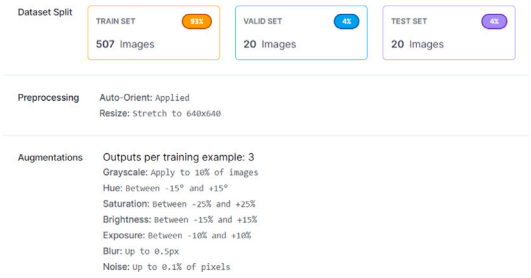



**Fig. 4.** Details of the dataset.

The dataset is partitioned into training, validation, and testing sets prior to data augmentation, following standard practice [29]. All images are resized to 640 × 640 pixels to reduce computational cost and improve training efficiency. To enhance model robustness, data augmentation is applied to the training set using Roboflow. Up to three augmented variants are generated for each training image, increasing the diversity of training samples. Fig. 4 summarizes the dataset composition, preprocessing steps, and augmentation strategy.

### 3.5.2. Model training

The YOLOv8 model is trained using Google Collaboratory, following implementation guidelines provided in Skalski [30]. The training environment supports execution of Python-based deep learning workflows with access to GPU resources.

The annotated dataset [31] used in this study corresponds to version 2, containing 209 images prior to augmentation. After augmentation, the dataset comprises a total of 547 images, partitioned into 507 training images, 20 validation images, and 20 testing images. The training set consists of 492 annotated images and 15 unannotated background images. Among the annotated training images, 396 images contain both Illegal and Occupied parking instances, 51 images contain only Illegal parking instances, and 45 images contain only Occupied parking instances. This diverse composition exposes the model to mixed parking scenarios, single-class scenes, and background-only scenes, enabling it to distinguish non-compliant parking from legally occupied parking while reducing false detections in images without target objects.

Training is performed using the YOLOv8 command-line interface, with a maximum of 300 epochs. Early stopping is applied if no improvement is observed over 50 consecutive epochs. The trained model and associated outputs are exported for deployment on the edge device.

Multiple YOLOv8 model variants, including YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x [32], are evaluated. These variants differ in model size and computational complexity. Smaller models provide faster inference, while larger models offer higher detection accuracy.

### 3.5.3. Model deployment and inference

The trained model is integrated into the video processing pipeline on the edge device. Each geo-tagged frame is processed to detect objects and assign class labels.

The object detection process can be represented as

$$D_t = f(I_t) \tag{3}$$

where $f(\cdot)$ denotes the trained YOLOv8 model and $D_t$ is the set of detected objects in frame $I_t$.

Each detected object $d_i \in D_t$ is represented by

$$d_i = (x_i, y_i, w_i, h_i, c_i, p_i) \tag{4}$$

where $(x_i, y_i)$ is the bounding box centre, $w_i$ and $h_i$ are the bounding box width and height, $c_i$ is the predicted class label, and $p_i$ is the confidence score.



**Fig. 5.** Bounding boxes of detected objects using YOLOv8.

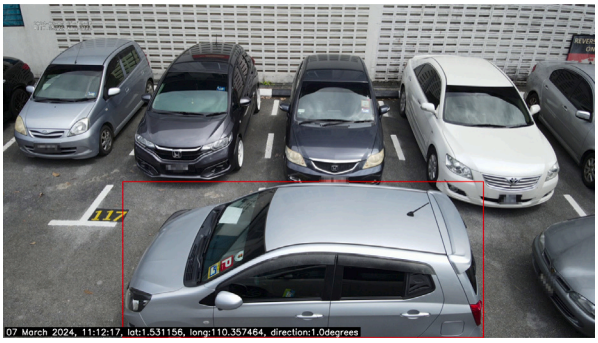



**Fig. 6.** Saved image of a detected non-compliant vehicle.

For the final implementation, a detection confidence threshold of 70% is applied, as determined from the experimental evaluation described in Section 4.4, to reduce false positive detections while maintaining a high detection rate. Detected objects are tracked across consecutive frames using the built-in tracking functionality of YOLOv8. To improve detection reliability, an object is considered valid only if it is detected in more than 15 frames.

For each valid detection, the associated UAV geolocation is recorded when the detected object is closest to the center of the frame, providing a consistent location reference for the detection event.

Fig. 5 illustrates detected objects with bounding boxes, class labels, confidence scores, and tracking identifiers.

### 3.6. System outputs and data integration

The proposed system generates multiple output formats to support validation, analysis, and integration with parking enforcement systems.

First, a processed video file in MPEG-4 (MP4) format is generated. This video overlays bounding boxes, class labels, confidence scores, and tracking identifiers on detected objects. The processed video provides a visual representation of the detection results and supports qualitative validation. A sample output video has been made available publicly.[3]

Second, images of detected non-compliant vehicles are extracted and stored in Joint Photographic Experts Group (JPEG) format. Each image includes metadata such as timestamp, latitude, longitude, and camera orientation. These images serve as visual references for parking enforcement personnel. Fig. 6 shows an example of a saved image.

Third, structured detection results are stored in JavaScript Object Notation (JSON) format. Each entry includes the object tracking identifier, timestamp, geographic coordinates, UAV altitude, and frame index. The JSON format enables efficient data exchange and supports integration with external applications.

[3] https://youtu.be/hRw5rxCFo58

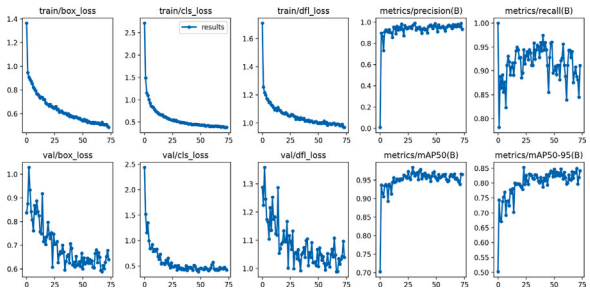



**Fig. 7.** YOLOv8n training and validation results.

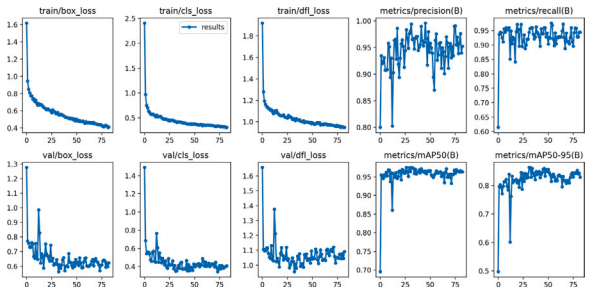



**Fig. 8.** YOLOv8s training and validation results.

**Table 1**
Time taken to train each YOLOv8 model.

| Model | Epochs | Time taken | Average time taken per epoch |
|---|---|---|---|
| YOLOv8n | 74 | 0.445 h | 21.65 s |
| YOLOv8s | 83 | 0.526 h | 22.81 s |
| YOLOv8m | 125 | 0.978 h | 28.17 s |
| YOLOv8l | 103 | 0.861 h | 30.09 s |
| YOLOv8x | 149 | 2.093 h | 50.57 s |

To facilitate practical deployment, a navigation link based on the UAV coordinates associated with each detection event is generated. This allows enforcement personnel to access the approximate location of the detected violation using mapping services.

All processing operations are implemented in a Python-based pipeline and packaged as an executable application for deployment on a Windows-based edge device. This design simplifies system execution and supports integration with existing parking management workflows.

## 4. Results and discussion

This section presents the experimental results of the proposed system and evaluates its performance under practical conditions.

### 4.1. YOLOv8 model training results

The performance of the object detection model is influenced by both dataset quality and model complexity. Following established guidelines for dataset preparation and training [28–30], multiple YOLOv8 model variants were trained and evaluated using the annotated dataset.

Table 1 summarizes the training time for each YOLOv8 variant. Models with larger parameter sizes require longer training time due to increased computational complexity. The YOLOv8n model exhibits the fastest training time, while YOLOv8x requires the longest time per epoch.

The training and validation results for each model are illustrated in Figs. 7–11. The loss functions, including box loss, classification loss, and distribution focal loss, show consistent convergence across all models. This indicates that the models successfully learn both object localization within the image and classification tasks.

Model performance is evaluated using the mean average precision metric across multiple intersection-over-union thresholds ($mAP_{50-95}$). This metric provides a comprehensive assessment of overall object detection performance.

Among the evaluated models, YOLOv8s and YOLOv8m demonstrate the strongest overall performance. Both models achieve high $mAP_{50-95}$ values and exhibit stable convergence during training. In particular, YOLOv8m shows slightly more stable precision characteristics and consistently strong validation performance, while YOLOv8s provides comparable detection accuracy with lower computational requirements.

Based on the training result, YOLOv8s and YOLOv8m are identified as the two strongest candidate models. Both models demonstrate strong detection capability and stable training behavior. Their performance is further evaluated using the test dataset in the following section, where additional detection metrics are considered to determine the most suitable model for deployment.

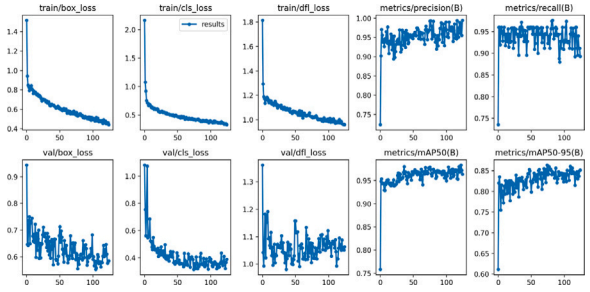



**Fig. 9.** YOLOv8m training and validation results.

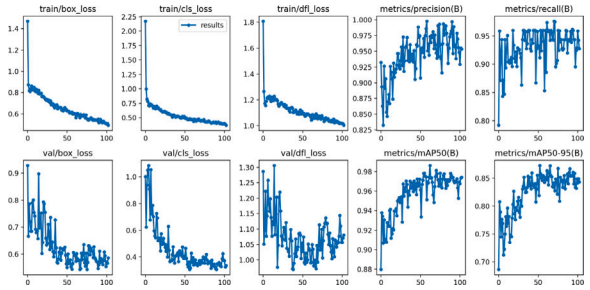



**Fig. 10.** YOLOv8l training and validation results.

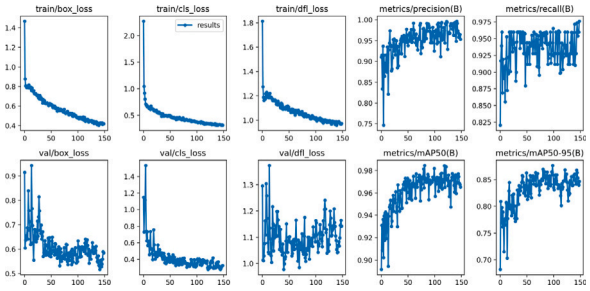



**Fig. 11.** YOLOv8x training and validation results.

### 4.2. YOLOv8 model validation results

The trained models are evaluated using the validation dataset described in Section 3.5.1. Confusion matrices are generated to assess classification performance for each YOLOv8 variant. The results are shown in Figs. 12–16.

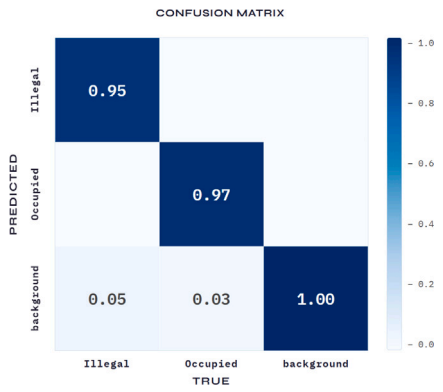



**Fig. 12.** YOLOv8n validation confusion matrix.

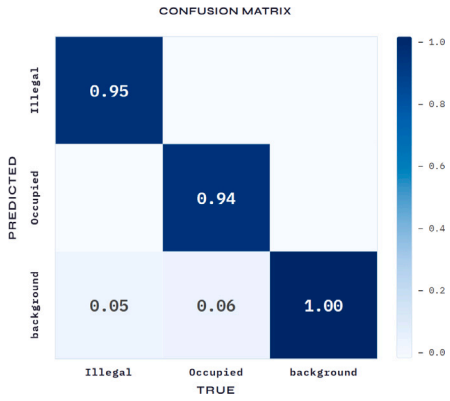



**Fig. 13.** YOLOv8s validation confusion matrix.

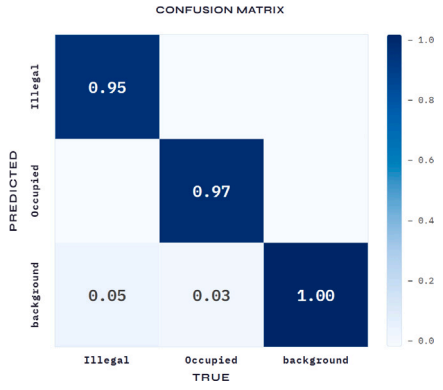



**Fig. 14.** YOLOv8m validation confusion matrix.

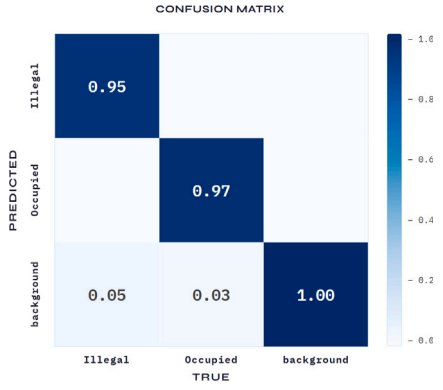



**Fig. 15.** YOLOv8l validation confusion matrix.

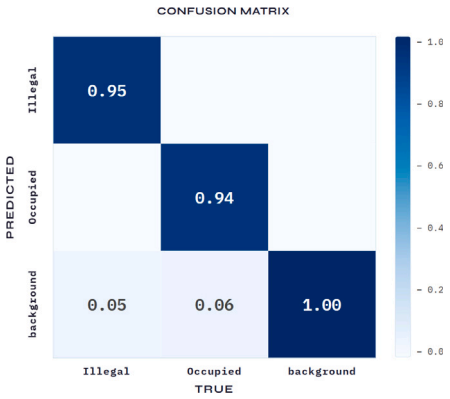



**Fig. 16.** YOLOv8x validation confusion matrix.

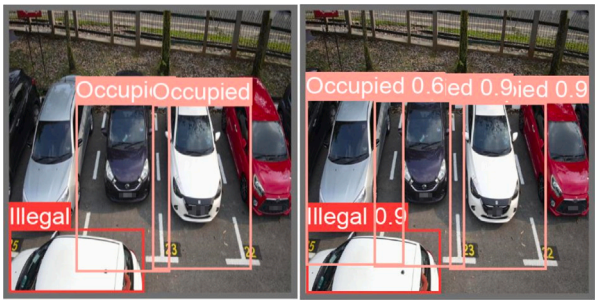



**Fig. 17.** Annotated image (left) vs validation results (right).




**Fig. 18.** Downloaded annotated image (left) vs Roboflow annotated image (right).

The confusion matrices indicate that all models achieve comparable true positive rates, demonstrating effective detection capability for both classes. However, differences are observed in the rate of incorrect classifications. Among the evaluated models, YOLOv8m exhibits the lowest false positive rate for non-compliant parking detection, indicating more reliable classification performance.

Further analysis of the validation results reveals that some false positives correspond to valid detections that were not annotated in the ground truth dataset. This results in apparent misclassification despite correct model predictions. An example is shown in Fig. 17, where the model identifies objects that are missing from the annotation.

False negatives are also observed in a small number of cases. Inspection of these samples indicates inconsistencies in the annotation data. In particular, discrepancies between the original annotations and the exported dataset lead to incorrect evaluation outcomes. Fig. 18 illustrates an example where bounding box definitions differ between the annotation source and the downloaded dataset.



**Fig. 19.** The same non-compliant vehicle is detected multiple times.

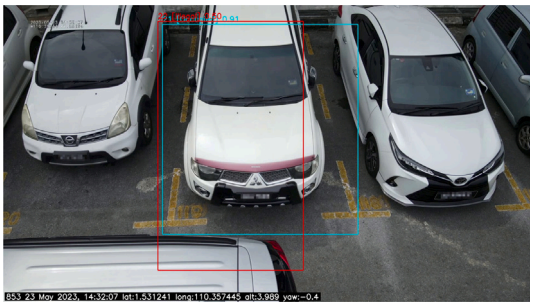


**Fig. 20.** A properly parked vehicle misclassified as non-compliant.

**Table 2**
Confusion matrix summary across YOLOv8 models.

| Model | TP | TN | FP | FN |
|---|---|---|---|---|
| YOLOv8n | 55 | 172 | 10 | 0 |
| YOLOv8s | 55 | 172 | 10 | 0 |
| YOLOv8m | 55 | 172 | 4 | 0 |
| YOLOv8l | 55 | 172 | 14 | 0 |
| YOLOv8x | 55 | 172 | 11 | 0 |

**Table 3**
Time taken to complete testing.

| Model | Time taken (seconds) |
|---|---|
| YOLOv8n | 2653.23 |
| YOLOv8s | 2857.23 |
| YOLOv8m | 3278.48 |
| YOLOv8l | 4147.02 |
| YOLOv8x | 4912.85 |

These observations suggest that part of the validation error is attributable to annotation quality rather than model performance. Despite these limitations, YOLOv8m demonstrates consistent and reliable behavior, supporting its selection for deployment in the proposed system.

### 4.3. YOLOv8 model video test results

Evaluation using static images does not fully capture real-world performance. The models are therefore tested on 12 unseen parking lot videos to assess their ability to detect non-compliant parking under dynamic conditions.

All models are evaluated using the proposed processing pipeline with a confidence threshold of 50%. Detection results are summarized using confusion matrices, as shown in Table 2. The total inference time for each model is reported in Table 3.

The results indicate that all models successfully detect non-compliant vehicles, with a total of 55 instances identified across the test videos. However, differences are observed in classification accuracy and computational efficiency. Among the evaluated models, YOLOv8m achieves the lowest number of false positives, with four incorrect detections and no false negatives. In contrast, larger models such as YOLOv8l and YOLOv8x introduce more false positives despite higher complexity.

Inference time increases significantly with model size, as shown in Table 3. Larger models require more computational resources, which limits their suitability for real-time or edge deployment.

Detailed inspection of the output videos reveals two primary sources of error. First, inconsistent object tracking can result in multiple detections of the same vehicle, as illustrated in Fig. 19. Second, the model may misclassify vehicles based on learned visual features rather than spatial context. An example is shown in Fig. 20, where a properly parked vehicle is incorrectly classified as non-compliant.

These errors are primarily attributed to limited training data and class imbalance. In particular, the model may associate specific vehicle appearances with non-compliant behavior, leading to incorrect generalization.

Considering both detection accuracy and computational efficiency, YOLOv8m provides a balanced performance across image-based and video-based evaluations. Although YOLOv8l achieves higher accuracy on static images, its higher inference cost and increased false positives in video scenarios reduce its suitability for deployment. YOLOv8m is therefore selected as the preferred model for the proposed system.

### 4.4. Summary of test results

All YOLOv8 model variants were successfully trained and evaluated. Model performance is influenced by both parameter size and dataset scale. Larger models such as YOLOv8x require longer training and inference time. These models also exhibit reduced robustness when trained on a limited dataset, which affects their ability to generalize to unseen scenarios.

Experimental results indicate that smaller and mid-sized models provide more stable performance under the given dataset constraints. In particular, YOLOv8m achieves a suitable balance between detection accuracy, robustness, and computational efficiency across both image-based and video-based evaluations.

The confidence threshold plays a critical role in controlling detection quality. A threshold of 70% is selected based on empirical observations. Lower thresholds increase false positives, including duplicate detections of the same vehicle and misclassification of properly parked vehicles. In contrast, correct detections are typically associated with confidence scores above 80%. The selected threshold therefore improves reliability while maintaining high detection accuracy.

To further reduce misclassification between non-compliant and properly parked vehicles, an overlap-based filtering rule is introduced. If the bounding box of a detected non-compliant vehicle overlaps with that of an occupied parking space by more than 50% of the area, the detection is discarded. This rule leverages spatial context to improve classification consistency and reduce false positives caused by ambiguous visual features. The 50% threshold was selected based on the characteristics of the target parking violation. In the studied parking environment, a double-parked vehicle typically obstructs access to adjacent parking spaces and occupies a substantial portion of the parking area associated with another vehicle. Consequently, detections with more than 50% overlap are more likely to correspond to vehicles legitimately occupying designated parking spaces rather than obstructive parking behavior.

### 4.5. Final implementation

Based on the experimental analysis, the system is refined to improve detection performance and reliability. The training dataset is first cleaned by manually inspecting and removing incorrectly annotated

**Table 4**
Confusion matrix for final YOLOv8m model.

| | Actual Positive | Negative | Total |
|---|---|---|---|
| Predicted Positive | 54 | 0 | 54 |
| Predicted Negative | 1 | 172 | 173 |
| Total | 55 | 172 | 227 |

**Table 5**
TPR, FPR, Precision and Accuracy of YOLOv8m Model.

| Performance | Percentage (%) |
|---|---|
| TPR | 98.18 |
| FPR | 0.00 |
| Precision | 100.00 |
| Recall | 98.18 |
| Accuracy | 99.56 |

samples generated by the auto-labeling process. The refined dataset is then used to retrain the model.

The YOLOv8m model is selected for the final implementation due to its strong performance in video-based evaluation and moderate inference cost. The model completes training in 1.179 h, converging after 159 epochs. A confidence threshold of 70% is applied to reduce false positives. In addition, the overlap-based filtering rule is incorporated to suppress incorrect detections in occupied parking spaces.

The refined system is evaluated using the same video test setup. The resulting confusion matrix is shown in Table 4.

Performance metrics are computed based on the confusion matrix [33]. The true positive rate (TPR), false positive rate (FPR), precision, recall, and accuracy are defined as follows:

$$TPR = \frac{TP}{TP + FN} \tag{5}$$

$$FPR = \frac{FP}{FP + TN} \tag{6}$$

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

where $TP$, $TN$, $FP$, and $FN$ denote true positive, true negative, false positive, and false negative, respectively.

The performance results are summarized in Table 5. The system achieves a false positive rate of 0% and a precision of 100%, indicating that no properly parked vehicles are misclassified as non-compliant. A single false negative is observed, resulting in an overall accuracy of 99.56%. The corresponding recall is 98.18%, indicating that 54 out of 55 non-compliant parking instances are successfully detected. These results demonstrate that the elimination of false positives is achieved while maintaining a high detection rate for non-compliant parking violations.

The remaining error is primarily attributed to limited training data. Insufficient variation in training samples reduces the model's ability to generalize across diverse parking scenarios. In particular, vehicles with similar visual characteristics may be incorrectly classified due to insufficient representation of contextual differences.

Increasing the size and diversity of the training dataset can improve model performance. Updated dataset versions with improved annotations have been shown to enhance detection accuracy.[4] However, collecting sufficient samples of non-compliant parking remains challenging, as such events occur infrequently. This limitation highlights the need for long-term data collection and continuous dataset refinement.

[4] Dataset available at: https://universe.roboflow.com/chung-yi-lai/illegal-parking-swinburne

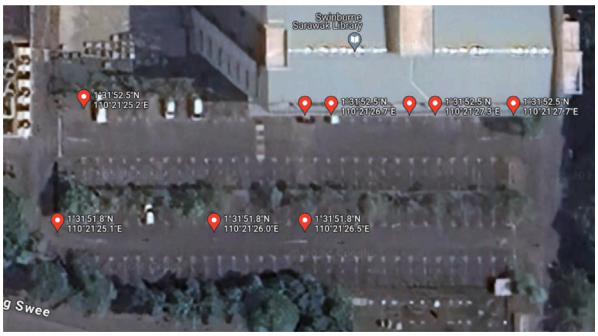



**Fig. 21.** UAV GPS coordinates associated with non-compliant parking events.

### 4.6. Location-referencing data of non-compliant parking events

Following video processing, a JavaScript Object Notation (JSON) file is generated to store violation event records associated with detected non-compliant vehicles. Each entry includes the UAV latitude and longitude coordinates at the time the detection frame was captured, together with metadata such as timestamp and detection identifier.

The recorded coordinates are represented in decimal degrees with six decimal places and serve as a geotag for the detection event. It should be noted that these coordinates correspond to the UAV position rather than the exact geographic location of the detected vehicle. Therefore, the recorded coordinates provide an approximate spatial reference for the violation event, while the spatial relationship between the recorded UAV position and the actual vehicle position depends on factors such as UAV altitude, camera orientation, viewing angle, and GPS accuracy.

Fig. 21 illustrates the recorded detection events visualized on Google Maps [34]. Each point is represented in the format (latitude, longitude), indicating the UAV position when a non-compliant vehicle was detected. For example, the coordinate (1.531062, 110.357368) corresponds to the UAV location associated with one detected violation event.

The recorded coordinates can be integrated with parking-space referencing information obtained from the parking occupancy monitoring framework to assist enforcement personnel in locating the general area where the violation occurred. A Google Maps URL can also be generated from the recorded coordinates to provide navigation to the vicinity of the detected event. For example, navigation to the recorded detection location can be accessed using the following URL: https://www.google.com/maps/dir/my+location/1.531062,110.357368

This capability supports practical deployment by providing a convenient method to review, visualize, and access the approximate location of detected parking violations. However, the proposed method does not perform precise vehicle geolocation and should not be interpreted as providing the exact ground coordinates of the offending vehicle.

## 5. Conclusions and future work

This paper presents a geo-aware UAV-based system for detecting non-compliant parking in open parking facilities. The proposed approach integrates aerial image acquisition, deep learning-based object detection, and GPS-based location referencing to enable accurate and scalable parking monitoring. Experimental results demonstrate that the system achieves an accuracy of 99.56%, while maintaining a false positive rate of 0% and a precision of 100%. The system also records location information associated with each detected violation

event, enabling visualization and navigation through digital mapping platforms.

The results indicate that UAV-based monitoring can effectively address limitations of fixed camera systems, particularly in handling visual obstructions and covering large outdoor areas. The integration of location-referencing information further enhances the practical utility of the system by supporting timely enforcement and decision-making. These capabilities highlight the potential of the proposed approach to improve the reliability and efficiency of vision-based parking management systems.

This work contributes to the advancement of intelligent urban infrastructure by addressing the impact of non-compliant parking on automated monitoring systems. The proposed framework demonstrates how aerial sensing and edge-based intelligence can be combined to improve detection accuracy and system responsiveness in real-world environments [35].

Future work will focus on improving model robustness through larger and more diverse datasets, particularly for non-compliant parking scenarios. Additional enhancements include extending the detection framework to other forms of visual obstruction, such as pedestrians, motorcycles, and temporary structures. Integration with real-time communication systems and cloud-based platforms will also be explored to enable scalable deployment across smart city environments.

## CRediT authorship contribution statement

**Chung Yi Lai:** Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation. **Hudyjaya Siswoyo Jo:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Jonathan Kua:** Writing – review & editing, Writing – original draft, Visualization, Validation. **Chean Hung Lai:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration. **Michelle Dunn:** Validation, Supervision, Software, Resources, Project administration.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgment

This work was supported by Swinburne University of Technology Sarawak Campus and Sarawak Digital Economy Corporation (SDEC) through the SDEC Translational Research Grant for “Development of Automated UAV-based Parking Enforcement and Monitoring System”.

## Data availability

Data will be made available on request.

## References

[1] J. Jin, K. Yu, J. Kua, N. Zhang, Z. Pang, Q.-L. Han, Cloud-fog automation: Vision, enabling technologies, and future research directions, IEEE Trans. Ind. Inform. 20 (2) (2024) 1039–1054, http://dx.doi.org/10.1109/TII.2023.3272696.

[2] J. Jin, Z. Pang, J. Kua, Q. Zhu, K.H. Johansson, N. Marchenko, D. Cavalcanti, Cloud-fog automation: The new paradigm towards autonomous industrial cyber-physical systems, IEEE J. Sel. Areas Commun. (2025).

[3] S. Wang, J. Kua, J. Jin, Y.W. Wong, P.P. Jayaraman, Z. Pang, Assisting mission-critical traffic flows with active queue management in industrial internet of things, J. Ind. Inf. Integr. (2026) 101170.

[4] L. Zhou, S. Leng, T.Q.S. Quek, VerDT: A versatile digital twins framework for UAVs-based industrial cyber-physical systems, IEEE Trans. Mob. Comput. 24 (10) (2025) 10099–10117, http://dx.doi.org/10.1109/TMC.2025.3567284.

[5] T. Yuske, Z. Mbaitiga, Development of drone detecting free parking space for car parking guidance, in: 2019 International Conference on Intelligent Informatics and Biomedical Sciences, ICIIBMS, 2019, pp. 385–387, http://dx.doi.org/10.1109/ICIIBMS46890.2019.8991452.

[6] C.Y. Lai, H.S. Jo, C.H. Lai, M. Dunn, Vision-based automated parking monitoring using geo-tagged UAV images, in: 2024 International Conference on Green Energy, Computing and Sustainable Technology, GECOST, 2024, pp. 136–140, http://dx.doi.org/10.1109/GECOST60902.2024.10474933.

[7] S. Sarkar, M. Totaro, K. Elgazzar, Intelligent drone-based surveillance: application to parking lot monitoring and detection, 2019, p. 3, http://dx.doi.org/10.1117/12.2518320.

[8] M. Mots’oehli, Y.C. Yang, Public parking spot detection and geo-localization using transfer learning, 2022, arXiv:2209.00213. URL: https://arxiv.org/abs/2209.00213.

[9] C.L.K. Ye, H.S. Jo, R.S. Jo, Development of UAV-based automated vehicle recognition system for parking enforcement, in: 2019 4th International Conference on Robotics and Automation Engineering, ICRAE, 2019, pp. 111–115, http://dx.doi.org/10.1109/ICRAE48301.2019.9043803.

[10] C. Xu, X. Hu, Real time detection algorithm of parking slot based on deep learning and fisheye image, J. Phys.: Conf. Ser. 1518 (2020) 012037, http://dx.doi.org/10.1088/1742-6596/1518/1/012037.

[11] C.K. Ng, S. Cheong, W. Yap, Y.-L. Foo, Outdoor illegal parking detection system using convolutional neural network on raspberry pi, Int. J. Eng. Technol. 7 (2018) 17, http://dx.doi.org/10.14419/ijet.v7i3.7.16197.

[12] N. Fadilah, S. Soon, H. Radi, Embedded automated vision for double parking identification system, Indones. J. Electr. Eng. Comput. Sci. 10 (2018) 1221–1226, http://dx.doi.org/10.11591/ijeecs.v10.i3.pp1221-1226.

[13] R. Akhawaji, M. Sedky, A.-H. Soliman, Illegal parking detection using Gaussian mixture model and Kalman filter, in: 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications, AICCSA, 2017, pp. 840–847, http://dx.doi.org/10.1109/AICCSA.2017.212.

[14] J. Kua, G. Armitage, P. Branch, A survey of rate adaptation techniques for dynamic adaptive streaming over HTTP, IEEE Commun. Surv. Tutor. 19 (3) (2017) 1842–1866.

[15] A. Ditta, M.M. Ahmed, T. Mazhar, T. Shahzad, Y. Alahmed, H. Hamam, Number plate recognition smart parking management system using IoT, Meas.: Sensors 37 (2025) 101409, http://dx.doi.org/10.1016/j.measen.2024.101409, URL: https://www.sciencedirect.com/science/article/pii/S2665917424003854.

[16] S.S. Channamallu, S. Kermanshachi, J.M. Rosenberger, A. Pamidimukkala, G. Hladik, Determinants of user satisfaction in smart parking applications, Transp. Econ. Manag. 3 (2025) 214–221, http://dx.doi.org/10.1016/j.team.2025.05.001, URL: https://www.sciencedirect.com/science/article/pii/S2949899625000140.

[17] R. Mahmud, J. Jin, J. Kua, M. Afrin, S. Mistry, A. Krishna, Trusted microservice orchestration for secure edge computing in industrial cyber-physical systems, IEEE Netw. (2025).

[18] T. Tran, Q. Nguyen, T. Luu, M. Tran, J. Kua, T. Hoang, M. Dien, Empowering robotic training with kinesthetic learning and digital twins in human–centric industrial systems, J. Ind. Inf. Integr. 43 (2025) 100743.

[19] T. Luu, Q. Nguyen, T. Tran, M. Tran, S. Ding, J. Kua, T. Hoang, Enhancing real-time robot teleoperation with immersive virtual reality in industrial IoT networks, Int. J. Adv. Manuf. Technol. 139 (11) (2025) 6233–6257.

[20] X. Xie, C. Wang, S. Chen, G. Shi, Z. Zhao, Real-time illegal parking detection system based on deep learning, 2017, pp. 23–27, http://dx.doi.org/10.1145/3094243.3094261.

[21] Z. Liu, W. Chen, C.K. Yeo, Automatic detection of parking violation and capture of license plate, in: 2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference, IEMCON, 2019, pp. 0495–0500, http://dx.doi.org/10.1109/IEMCON.2019.8936164.

[22] X. Peng, R. Song, Q. Cao, Y. Li, D. Cui, X. Jia, Z. Lin, G.-B. Huang, Real-time illegal parking detection algorithm in urban environments, IEEE Trans. Intell. Transp. Syst. 23 (11) (2022) 20572–20587, http://dx.doi.org/10.1109/TITS.2022.3180225.

[23] G. P C P da Luz, G. Massuyoshi Sato, L. Fernando Gomez Gonzalez, J. Freitag Borin, Smart parking with pixel-wise ROI selection for vehicle detection using YOLOv8, YOLOv9, YOLOv10, and YOLOv11, Internet Things 36 (2026) 101858, http://dx.doi.org/10.1016/j.iot.2025.101858.

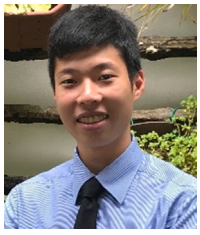
[24] J.Á. Morell, G. Luque, E. Alba, On-street parking space localization with deep learning using low-quality images from public cameras, Internet Things 32 (2025) 101619, http://dx.doi.org/10.1016/j.iot.2025.101619.

[25] N.M. Shariff, Private vehicle ownership and transportation planning in Malaysia, in: International Conference on Traffic and Transportation Engineering, vol. 64, 2012, p. 68.

[26] DJI, Mavic 2 enterprise series - DJI, 2023, URL: https://www.dji.com/mavic-2-enterprise.

[27] V. Verroios, V. Efstathiou, A. Delis, Reaching available public parking spaces in urban environments using ad hoc networking, in: 2011 IEEE 12th International Conference on Mobile Data Management, vol. 1, 2011, pp. 141–151, http://dx.doi.org/10.1109/MDM.2011.49.

[28] J. Gallo, How to label outdoor surveillance data for computer vision models, 2024, URL: https://blog.roboflow.com/label-outdoor-surveillance-data/.
[29] J. Solawetz, Train, validation, test split for machine learning, 2020, URL: https://blog.roboflow.com/train-test-split/.
[30] P. Skalski, How to train YOLOv8 object detection on a custom dataset, 2023, URL: https://blog.roboflow.com/how-to-train-yolov8-on-a-custom-dataset/.
[31] C.Y. Lai, Illegal parking swinburne dataset, 2024, URL: https://universe.roboflow.com/chung-yi-lai/illegal-parking-swinburne. visited on 2024-05-20.
[32] G. Jocher, B. Qaddoumi, J. Qiu, Object detection, 2023, URL: https://docs.ultralytics.com/tasks/detect/.
[33] P. Dangeti, Statistics for Machine Learning, Packt Publishing Ltd, 2017.
[34] C.Y. Lai, Non-compliant vehicles, 2024, URL: https://maps.app.goo.gl/XWenfRSwhe4NJEfdA.
[35] M. Ladegourdie, J. Kua, Performance analysis of OPC UA for industrial interoperability towards industry 4.0, IoT 3 (4) (2022) 507–525.



**Chung Yi Lai** received his Master of Engineering degree from Swinburne University of Technology. He currently serves as a Staff Engineer in Applications Engineering at Western Digital, Sarawak, Malaysia, where he focuses on bridging practical industrial applications with automation, intelligent alert systems, and smart factory implementations. His research interests center on computer vision and object detection, particularly YOLO-based architectures. Prior to pursuing his master's degree, he worked as an R&D Engineer at ViTrox Corporation, where he gained experience in machine programming and automation systems.



**Hudyjaya Siswoyo Jo** is an Associate Professor in Robotics and Mechatronics Engineering at Swinburne University of Technology Sarawak Campus, Malaysia. He received his B.Eng. (Hons.) in Robotics and Mechatronics Engineering in 2008 and Ph.D. in 2013 from Swinburne University of Technology. His research interests include robotics and mechatronics, industrial automation, Industrial Internet of Things (IIoT), embedded systems, artificial intelligence and smart manufacturing, with a particular focus on the development and application of intelligent systems for industrial digital transformation.



**Jonathan Kua** received the B.Eng. (First Class Hons.) degree in telecommunications and network engineering and the Ph.D. degree in telecommunications engineering from Swinburne University of Technology, Australia, in 2014 and 2019, respectively. He is currently a Senior Lecturer and Academic Director (Cloud Computing) within the School of Information Technology, Deakin University, Australia. His research interests include low-latency data transport protocols, adaptive streaming, content delivery, Internet of Things and industrial cyber-physical systems, with an overarching focus on improving their network performance and quality of service. He was the recipient of the Netflix Ph.D. scholarship award (2015–2019) and the second runner-up of the international DASH-IF Best Ph.D. Dissertation Award on "Algorithms and Protocols for Adaptive Content Delivery over the Internet" (2019). He previously served as a Guest Editor for the IEEE Journal on Selected Areas in Communications (JSAC) Special Issue on "Co-Design of Communication, Computing, and Control in Industrial Cyber-Physical Systems".



**Chean Hung Lai** eceived the B.Eng. (Hons.) degree in Robotics and Mechatronics Engineering in 2006 and the Ph.D. degree in Engineering in 2012, both from Swinburne University of Technology. He is currently an Associate Professor with the Faculty of Engineering, Computing and Science, Swinburne University of Technology Sarawak Campus, Malaysia. His research interests include robotics and intelligent automation, autonomous robotic systems, machine vision, Internet of Things (IoT), renewable energy systems, smart microgrids, and energy management and control. His current research focuses on integrating robotic technologies, intelligent sensing, and IoT solutions to address sustainability challenges, particularly in precision agriculture and rural development.



**Michelle Dunn** is an academic in the School of Engineering at Swinburne University of Technology, Australia. Her research interests span robotics and autonomous systems, with applications in space robotics, advanced manufacturing, assistive technologies, and intelligent sensing. Her work focuses on the development and application of engineering technologies to address practical, real-world challenges.