Deep Learning-Based System for Small Trash Detection and Violation Recognition

Chen-Hung Lin, Ci-Yi Hung, Hui-Long Yeo, and Li-Der Chou

***Abstract*— Littering by drivers is a widespread global traffic issue, contributing to urban waste accumulation, degrading scenic environments, and posing significant safety hazards that may lead to traffic accidents. This paper addresses the problem of littering from vehicles in Taiwan by designing and implementing a "Deep Learning-Based Micro-Litter and Traffic Violation Detection System". The system leverages object detection and deep learning technologies to accurately detect littering violations and capture vehicle license plate information, providing a means to alert authorities and the public to deter such behavior. The object detection module identifies illegal littering instances from input video streams and retrieves license plate data to be reported to users, enhancing the enforcement and prevention of littering-related violations.**

***Index Terms*— Deep Learning, Object Detection, Micro-litter Detection, and Object Tracking**

I. INTRODUCTION

According to statistics from the Environmental Protection Bureau, improper disposal of cigarette butts ranked first among environmental violations in 2023, with a total of 91,848 cases [1], followed by littering and spitting betel nut juice or residue. Such actions not only undermine urban cleanliness but also increase the workload of sanitation workers and pose additional risks to traffic safety. Currently, the government's strategy involves public reporting and the use of intersection surveillance cameras for technological law enforcement, as shown in Figure 1. While these measures have been somewhat effective in reducing roadside littering by drivers, there remains room for improvement in these policies.

Currently, the government primarily relies on public reporting and the installation of surveillance cameras at intersections for technological law enforcement. While this approach has been effective to some extent in reducing littering by drivers, limitations in monitoring coverage present notable challenges. For instance, not all road sections are equipped with surveillance cameras, as shown in Figure 1 [2], and remote or less accessible areas often lack adequate monitoring equipment, allowing drivers to dispose of litter with impunity. Additionally, analyzing footage from these systems requires extensive filtering and processing, tasks that are predominantly manual, consuming significant time and limiting enforcement efficiency. Manually searching for violations in video footage is highly inefficient, as it incurs substantial time costs in identifying relevant incidents. Moreover, objects commonly discarded by drivers are often small and difficult to detect in videos, further complicating the efforts of enforcement personnel to effectively capture and penalize such violations.

This system addresses the issue by proposing a deep learning-based micro-litter and traffic violation detection system. It utilizes footage from the dashboard cameras of numerous drivers, leveraging existing devices to capture violations. Through a crowdsourcing approach, the system encourages the public to upload dashcam recordings, enabling the detection of violations even on roads without fixed surveillance cameras. The proposed technology employs object detection techniques to identify litter and track vehicles involved in littering, capturing the license plates of violating cars. This significantly reduces the time enforcement personnel spend manually reviewing footage for violations, greatly enhancing the efficiency of law enforcement.

This paper implements the Deep SORT and YOLOv9E algorithms, optimizing the system architecture and object tracking for specific application scenarios. Particularly in littering detection, the system achieves stable identification of multiple small pieces of litter. When objects are occluded or temporarily disappear, Deep SORT utilizes motion modeling and appearance features to recover tracking, effectively reducing ID loss. The integration of YOLOv9E and Deep SORT, along with prioritized processing of specific object modules, results in a 50.25% speed improvement. Comparative analysis with YOLOv8E and PRB-FPN (Parallel Residual Bi-Fusion Feature Pyramid Network) further validates the superior performance of YOLOv9E in detecting micro-litter, demonstrating the advantages of the proposed system in precision and efficiency.

This paper is organized into five sections as follows. Section I presents the research background, motivation, objectives, and an overview of the paper structure. Section II summarizes and examines relevant studies on micro-litter detection and violation recognition. Section III provides a detailed

Chen-Hung Lin, Ci-Yi Hung, Hui-Long Yeo, and Li-Der Chou with the Department of Computer Science and Information Engineering, National Central University, Taiwan (email;daniel920817@gmail.com;tenwu06@g.ncu.edu.tw; chuilongyeo@gmail.com; cld@csie.ncu.edu.tw).

Corresponding author: Li-Der Chou (email: cld@csie.ncu.edu.tw).

explanation of the training model and the process of applying algorithms within the model. Section IV evaluates the accuracy and performance of the system. Section V summarizes the research findings and discusses future research directions.

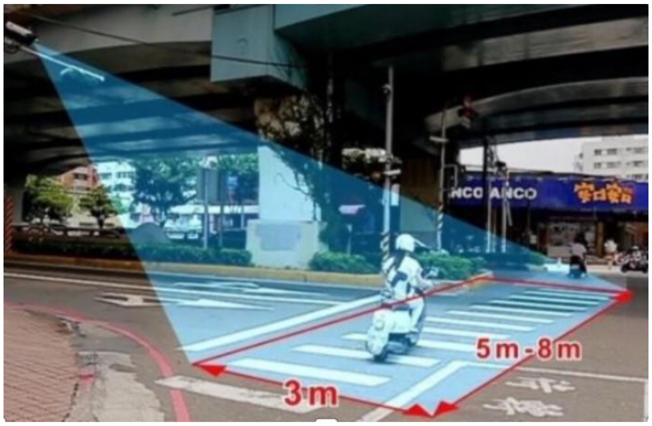


Figure 1. Surveillance Camera Coverage Diagram [2]

II. RELATED WORK

Reference [3] proposes a modular construction method that significantly improves the accuracy of small object detection. It leverages the inherent pyramid structure of deep convolutional networks by utilizing a Feature Pyramid Network (FPN) with lower computational overhead. FPN employs a top-down feature fusion approach, similar to Top-Down Modulation, but includes multiple feature prediction layers rather than just one. FPN is a multi-scale object detection algorithm that uses feature maps of varying resolutions to detect objects of different sizes. As shown in Figure 2, it integrates both low-level and high-level information through successive up sampling and cross-layer fusion mechanisms, enhancing the representational capacity of the output features.

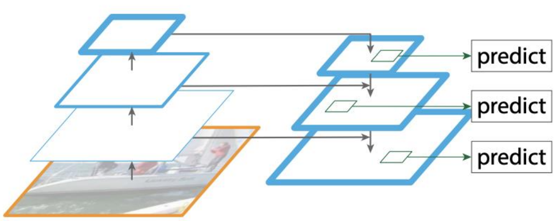


Figure 2. Feature Pyramid Network Pyramid Architecture [3]

FPN and its derived architectures are commonly used techniques to enhance detection accuracy. The structure of the Parallel Residual Bi-Fusion Feature Pyramid Network (PRB-FPN) [4], as shown in Figure 3, integrates both top-down and bottom-up fusion functions for fast and accurate single-shot object detection. Single-shot detection refers to simultaneously performing object localization and classification in a single forward pass. Compared to traditional two-stage detection methods, this approach offers the advantage of faster computation, making it highly suitable for real-time applications. The PRB-FPN architecture improves detection accuracy for small objects by incorporating more precise localization information into each prediction layer. This not only enhances the detection of small objects but also significantly improves the detection performance for medium and large objects.

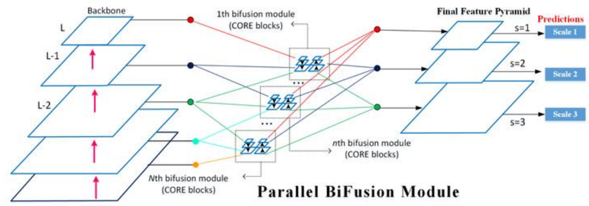


Figure 3. Parallel Residual Bi-Fusion Feature Pyramid Network Architecture [4]

Reference [5] evaluates the performance of two object detection algorithms, YOLOv5 [6] and Faster R-CNN [7], in the context of beach litter detection. The study uses Precision, Recall, and Mean Average Precision (mAP) to comprehensively assess the effectiveness of these algorithms. The results show that YOLOv5 achieves an mAP of 0.542, while Faster R-CNN achieves 0.328. YOLOv5 outperforms Faster R-CNN in all aspects.

Reference [8] explores the use of YOLOv8 [9] for recyclable waste identification. The paper begins with an overview of the YOLOv5 architecture, where image preprocessing includes mosaic data augmentation, Adaptive Image Scaling, and Adaptive Anchor Box Calculation. In the Backbone, the Focus and Cross Stage Partial (CSP) structures are used to extract features and enhance the neural network's learning capacity. The Neck performs multi-scale feature fusion, and the Prediction layer performs regression for object detection. The paper also discusses improvements made in YOLOv8, including replacing the CSP structure with the C2f structure and separately predicting the classification of targets and the regression of bounding boxes, which accelerates training and improves detection capabilities.

Reference [10] primarily focuses on using low-altitude drone images for waste detection. The system utilizes deep learning techniques to detect trash and debris in images captured by drones and marks them on a map, enabling automated waste collection. A detector called "eye in the sky" was developed, leveraging deep learning object detection and GPS to ensure precise image detection. The object recognition technique used is YOLOv4 [11]. Additionally, a dataset named UAVWaste [12] was created, consisting of 772 images and 3,716 manually annotated trash labels. This dataset was publicly released in COCO (Common Objects in Context) [13] format. The COCO database is a large image dataset for image recognition and segmentation, containing over 330,000 images and more than 1.5 million object categories. It provides high-accuracy object annotations, making it an ideal choice for training and evaluating object detection.

Reference [14] introduces the PRB-FPN+ module, which modifies the original PRB-FPN by incorporating the P6 module into the P5 module to accommodate larger image inputs. As shown in Figure 4, the model uses both Aux head and Lead head to detect and locate objects of various sizes without compromising efficiency, improving the precision of small object detection, such as trash. The study also found that its accuracy in harsh environments outperforms YOLOv7 [15]. Furthermore, the SMILEtrack (Similarity Learning for Occlusion-Aware Multiple Object Tracking) [16] tracker was introduced to handle long-term occlusion issues or fast-moving objects.

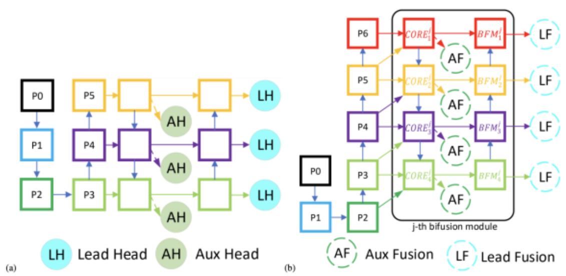


Figure 4. PRB-FPN+ module [14]

Reference [16] introduces SMILE track, which combines a detector with a Siamese network-like similarity learning module [17]. Its goal is to predict the trajectory of each target and link them across every frame in a video. In a Multiple Object Tracking (MOT) performance comparison, SMILE track outperforms other trackers in terms of tracking accuracy and real-time performance, achieving 80.7% Multiple Object Tracking Accuracy and 65.0% Higher Order Tracking Accuracy at 37.5 FPS. MOT relies on a Trace-Before-Detect approach, where tracking is based on detection results. The architecture of SMILE track, as shown in Figure 5, can be divided into the following steps:

1. Target Localization: Accurately determines the position of the target object.

2. Data Association: Analyzes the relationships between objects across adjacent frames to address the multi-object tracking problem.

After detection, the motion similarity matrix and appearance similarity matrix for each frame are computed. The Hungarian Algorithm [18], a combinatorial optimization method used to solve assignment problems in polynomial time, is then applied to solve the linear assignment problem. In object tracking, the Hungarian Algorithm matches objects in each frame based on the results of the motion and appearance similarity matrices, determining the corresponding relationships of objects in the time sequence.

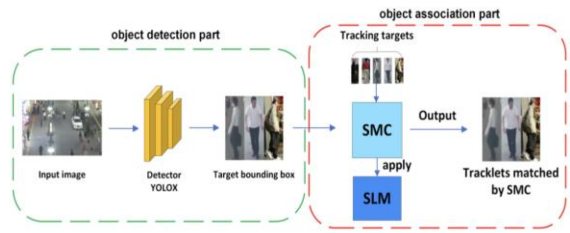


Figure 5. SMILE track Architecture [16]

The reference introduces the concept of "Programmable Gradient Information (PGI)," which is designed to address the various changes required in deep learning networks to achieve multiple objectives. PGI can provide complete input information to compute the objective function, thereby obtaining reliable gradient information to update network weights.

Additionally, the paper designs a novel lightweight network architecture called the "Generalized Efficient Layer Aggregation Network (GELAN)" based on gradient path planning. The architecture of GELAN demonstrates the superior performance of PGI in lightweight models, as shown in Figure 6. The authors validated GELAN and PGI in object detection using the COCO dataset. The results show that GELAN, by using traditional convolution operations, can surpass deep convolution-based methods in terms of parameter utilization. PGI is applicable to a wide range of models, from lightweight to large-scale, allowing models trained from scratch to outperform state-of-the-art models pre-trained on large-scale datasets.

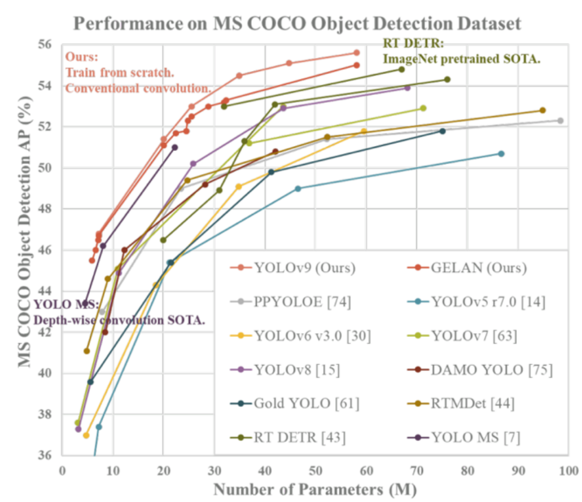


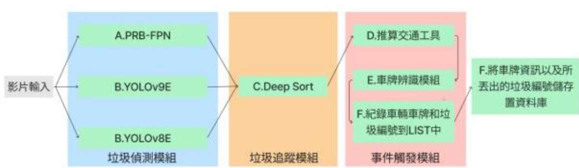
Figure 6. The performance comparison chart of YOLOv9. [19]

III. Research Methodology

These system aims to achieve an illegal littering detection system by comparing three different object detection modules: YOLOv8E, YOLOv9E, and PRB-FPN. Both YOLOv8E and YOLOv9E are part of the YOLO series object detection models, with YOLOv9E providing significant improvements over YOLOv8E, especially in terms of accuracy, computational efficiency, and deployment flexibility. PRB-FPN, with its bidirectional fusion capability, is a commonly used technique to enhance detection accuracy, designed for fast and precise single-shot object detection.

The overall system workflow is illustrated in Figure 7 (a). The system receives video streams from surveillance or other sources. YOLOv9E is employed to detect objects, and the Deep Sort algorithm is used to track the detected objects' trajectories in the video. The system examines whether the variation in the Y-axis value of the object's trajectory is below a specified threshold. If so, it captures the object's trajectory, computes the behavior of the vehicle, and invokes the license plate recognition module to identify the license plate of the violating vehicle, recording it in a list. Otherwise, no further action is taken, and the process returns to its initial state. Finally, the system outputs the recognized license plate information along with the littering incident identifier. The overall system architecture is depicted in Figure 7 (b).

(a)



(b)

Fig. 7. (a) System Flowchart (b) System Architecture Diagram

The following steps outline the implementation process of this system:

A. Construction of the PRB-FPN Module

This module integrates the PRB-FPN module to perform feature extraction on objects such as litter and license plates, extracting higher-level features from input images. This process transforms input images into abstract representations called Feature Maps, which capture important patterns and structures in the images. The design of this module follows the method described in [14], incorporating the module presented in this paper by adding lower-level feature maps, P0 and P1, into the Backbone network.

The P0 and P1 feature maps focus on capturing detailed and localized information, offering more precise image detail descriptions and making them particularly suitable for detecting smaller objects. The Backbone network, serving as the foundational structure, extracts multi-level features from input images and provides them to subsequent modules for object detection. The lower-level feature maps, P0 and P1, have higher resolutions, preserving fine details such as edges and textures—essential for detecting extremely small objects like litter. In contrast, higher-level feature maps emphasize global semantic information. By combining the upper and lower feature maps, the module effectively captures multi-scale object features, further enhancing detection accuracy.

B. Construction and Training of the YOLO Module

The litter detection module is built using YOLOv8E and YOLOv9E. Compared to earlier versions, these newer versions offer improved overall performance. The YOLO series employs a single-stage detection algorithm, providing faster processing speeds. The "prediction mode" is used to input data into the module, adjusting parameters such as the confidence threshold and model weights to achieve optimal detection results.

Figure 8 presents the results of testing the YOLOv8E module using a vehicle-related dataset for training. The dataset was obtained from the Roboflow vehicle dataset. In Figure 8, detected vehicles are enclosed in bounding boxes, with labels showing the type of vehicle and the corresponding confidence score. The closer the score is to 1, the higher the detection accuracy.

Figure 9 illustrates an experimental scenario where vehicle recognition was performed using images, demonstrating YOLOv8E's excellent performance in relatively complex environments, achieving a mean Average Precision (mAP) of approximately 90%.

The mAP formulae used in this paper are shown below as (1) and (2).

$$AP = \int_0^1 P(r)\,dr \qquad \textbf{\textit{(1)}}$$

$$\mathrm{mAP} = \frac{1}{n}\sum_{i=1}^{n} \mathrm{AP}_i \qquad \textbf{\textit{(2)}}$$

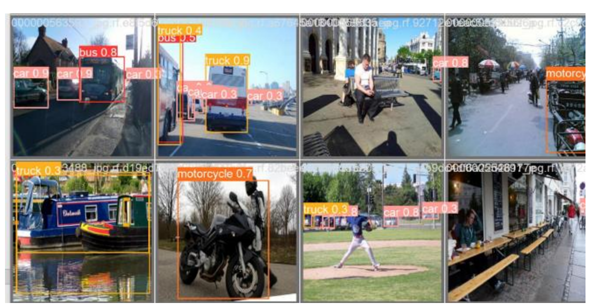


Figure 8. The module's labeling results on the training dataset.

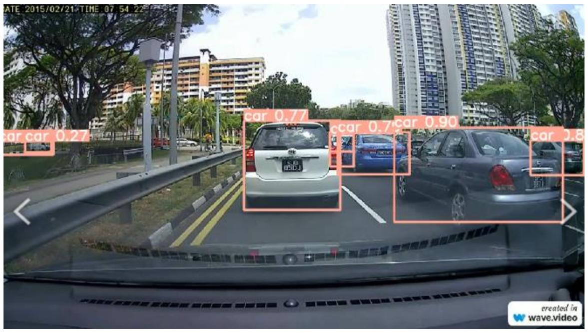


Figure 9. The module's labeling results on real-road video footage.

C. Deep Sort Tracking

YOLOv8E and YOLOv9E can only process information from the current frame and are unable to capture the relationships between the previous, current, and even future frames. Therefore, object tracking is introduced to correlate each frame and achieve more accurate recognition results.

In the study described in [21], the researchers employed a Kalman filter combined with the Hungarian algorithm for frame-by-frame data association, using the overlap ratio of the bounding boxes as the metric for association. This approach achieves stable and efficient object tracking at high frame rates. Additionally, to address the issue of identity recovery for objects after long-term occlusion, the method introduces the Mahala Nobis distance, as shown in equation (3), and a cosine distance-based association strategy that combines appearance features. This design effectively enhances the continuity and stability of object tracking, ensuring accurate tracking even in occlusion scenarios.

$$d^{(1)}(i,j) = (d_j - y_i)^T S_i^{-1}(d_j - y_i) , \qquad (3)$$

In this study, this technique will be used for object tracking. Based on the specific requirements of this research, its functionality will be further extended by adding multiple attributes during the detection process to improve the event determination of the objects being tracked.

After YOLOv8E performs object detection, the bounding boxes of the objects are identified, and each bounding box is assigned a unique ID, recording the presence of each ID in every frame, as shown in Figure 10(a). Before correction by Deep Sort, if the vehicle with ID 2 is occluded by the vehicle with ID 1 and reappears, it might be mistakenly labeled as ID 3. However, this result is not expected, as ID 2 and ID 3 should be considered the same vehicle.

By calculating the Intersection over Union (IoU) of the centers of the bounding boxes, it is possible to determine if the object was occluded by a nearby object, preventing it from being detected, as shown in Figure 10(b). This allows for the correction of ID 3 back to ID 2. With this feature, the module can accurately recognize the license plate numbers of moving vehicles, even in different scenarios or when occlusion occurs.

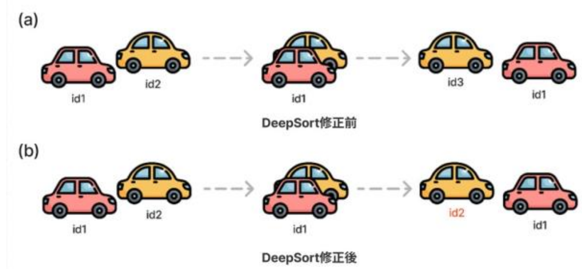


Figure 10. The Deep Sort correction diagram.

Deep Sort possesses superior multi-object tracking capabilities, which will enable this paper to simultaneously track multiple moving vehicles engaged in littering behavior, thus improving accuracy.

D. Vehicle and Litter Bounding Box Capture

For litter detection, the litter detection module uses a custom dataset from Roboflow [22], while vehicle detection is trained using the vehicle dataset from Roboflow. Since YOLOv9E only supports images with a width and height of 640px, the TensorFlow resize_with_pad function will be used to resize all images to 640x640 pixels, ensuring they meet the module's input requirements.

E. License Plate Detection

To train the license plate detection module, this system will use the car plate images from the Kaggle platform's car-plate-license dataset [22]. The dataset is in XML format, and it will be converted into the TXT file format required by YOLO. Each line in the TXT file will contain information such as the object class, center coordinates, width, and height.

A subdirectory named "datasets" will be created under the project directory, where 80% of the TXT files will be used for the training set and 20% for the validation set. By inputting the YOLOv8E training command in this directory and setting the mode to "train," the module can be trained.

F. Vehicle Detection and Database Storage

After the training, the module is now capable of performing license plate recognition and litter detection. In this phase, the module will use the distance between the center of the vehicle's bounding box and the center of the bounding box where the litter appears as the basis for determining vehicle violations.

Additionally, by setting a distance threshold, the module will only detect violations when the distance between the litter and the vehicle's bounding box center is within this threshold. This ensures the accuracy of the detection and prevents misidentifying litter that is already on the ground as a violation.

When a violation is detected, the relevant vehicle information, including the license plate and detection time, will be recorded in the database for further processing.

This feature contributes to the establishment of a sound waste management system by providing real-time information storage, allowing for quick responses to littering incidents. With such a system in place, cities will be able to address littering issues more efficiently and improve the overall quality of the environment.

IV. EXPERIMENTS AND DISCUSSION

This experiment evaluates the performance of three modules—PRB-FPN, YOLOv8E, and YOLOv9E—in litter detection and training. Figure 11 shows the training performance analysis of the YOLOv8E module, with categories such as Plastic Bag, Plastic Bottle, Aluminum Can, Cigarette, and Paper Cup achieving high Average Precision (AP) values close to 1, indicating the module's accuracy in classifying these categories. These categories are consistently detected with minimal errors. However, the category Cigarette, with a lower AP value of 0.917, shows a noticeable drop in accuracy, likely due to the smaller size of the object. The mAP value is 0.979, and the curve's proximity to the upper right corner demonstrates the overall excellent performance of the module across all categories.

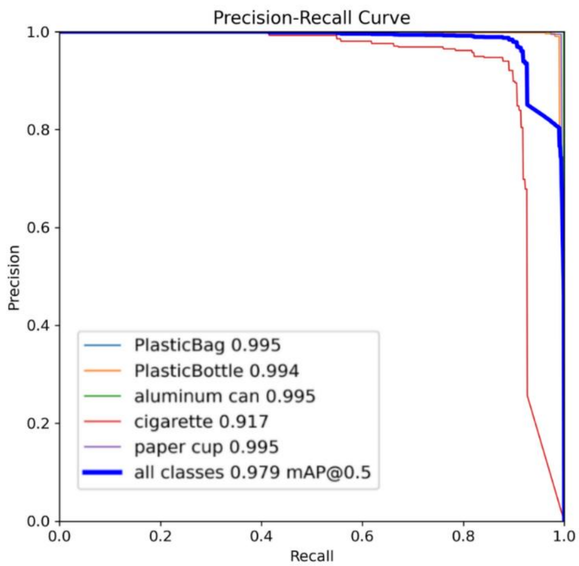


Figure 11. The training performance chart of the YOLOv8E module.

Figure 12 shows the training results of the YOLOv9E module. The AP values for Plastic Bag and Aluminum Can remain at 0.995, indicating strong detection accuracy for these categories. However, the AP for the Cigarette category drops to 0.811, with a noticeable decline in the curve, suggesting a drop in performance compared to Figure 11. The overall mAP value decreases to 0.956, slightly lower than the mAP value of 0.979 in YOLOv8E.

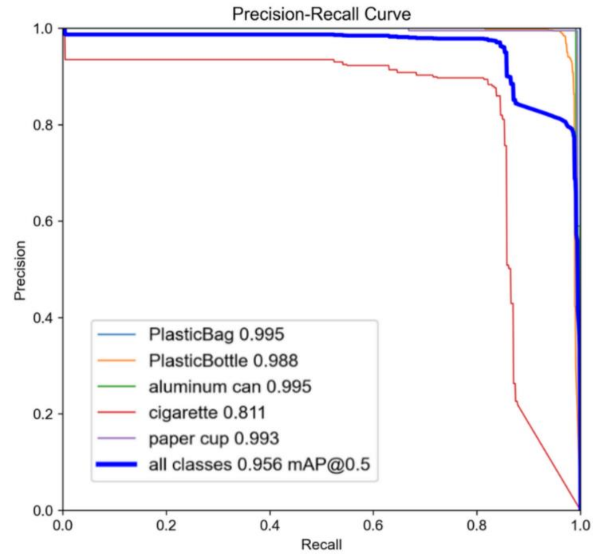


Figure 12. The training performance chart of the YOLOv9E module.

Figure 13 displays the training results for the PRB-FPN module. The average precision is 0.972, demonstrating stable and accurate performance. Categories such as Plastic Bag (accuracy of 0.996) and Aluminum Can (accuracy of 0.995) show curves very close to the upper-right corner, indicating near-perfect detection. The Cigarette category, however, has a lower accuracy of 0.885, and the curve shows a noticeable drop in the high Recall region, reflecting a decrease in precision.

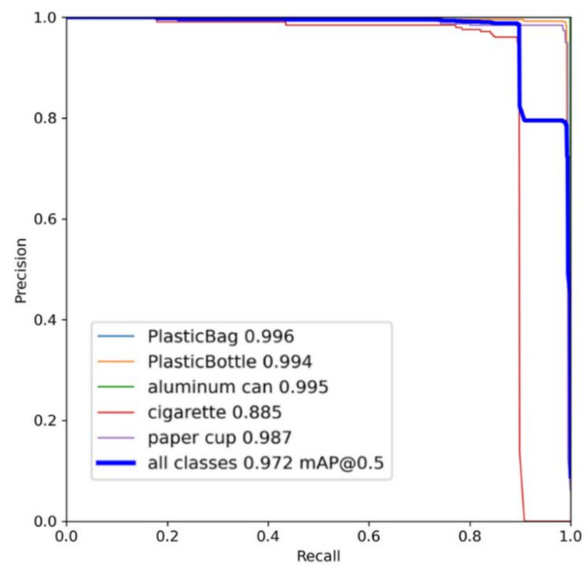


Figure 13. The training performance chart of the PRB-FPN module.

Although YOLOv9E shows instability in its loss function graph during training, with an mAP of 0.956, which seems lower than expected, its detection performance in practice outperforms the other modules. Particularly in cases where litter is briefly occluded, YOLOv9E maintains good recognition abilities. YOLOv8E seems to be overfitted to the training dataset, resulting in slightly weaker performance on real-world video data compared to YOLOv9E. Therefore, YOLOv9E was ultimately selected as the primary detection module for both modules in this system to ensure stable detection of littering behavior by vehicle drivers in complex scenarios.

TABLE I summarizes the performance of the modules, categorizing them into three groups: Plastic Bottle representing large objects, Cigarette representing very small objects, and mAP representing the overall performance of the modules.

TABLE I

Accuracy of different modules for the same object.

	Plastic Bottle	Cigarette	mAP
YOLOv8E	0.994	**0.917**	**0.979**
YOLOv9E	0.994	0.885	0.972
PRB-FPN	**0.998**	0.811	0.956

After completing object and vehicle detection with the YOLOv9E module, this system further employs the Deep Sort technique to perform multi-object tracking on the detected targets. Deep Sort combines the Kalman filter's position prediction with appearance features extracted by deep learning to stably track the same vehicle and litter objects across different frames.

In the event determination process, the system continuously tracks the trajectory and related information of the litter objects, calculating the ratio of their Y-axis change to the frame difference. Based on the initial distance and area ratio between the litter and the nearest vehicle, the system dynamically adjusts the decision threshold: the greater the distance and the smaller the area, the lower the initial threshold. As the litter object's survival frame count increases during the tracking process, the threshold gradually increases, ensuring high accuracy under various scenarios.

When this ratio exceeds the dynamic threshold, the event is triggered. At this point, the system will trace back the trajectory to deduce which vehicle discarded the litter and confirm the responsible vehicle by calculating the Intersection over Union (IOU) value between the litter and vehicle. The system will then automatically capture an image of the vehicle, and through the license plate recognition module, identify the license plate number, returning the result to the system for further processing and recording.

The entire process flow is as follows Figures14 (a), (b), (c), (d), (e):

A. Create a litter object dataset and use YOLOv9E for module training and detection.

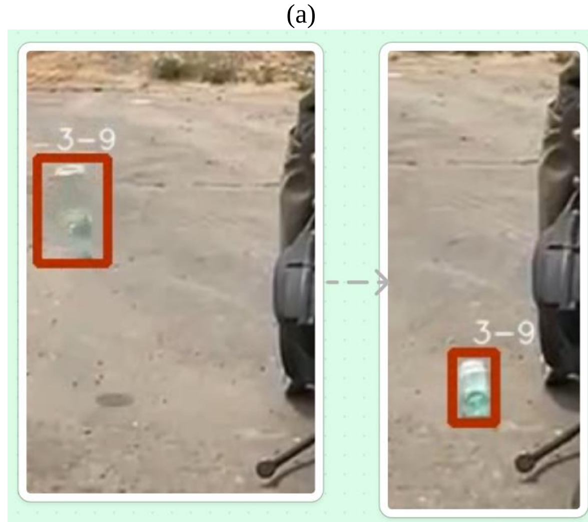
B. If the detected litter in the video moves too much, the system will use the tracked trajectory to deduce which vehicle discarded it and call the vehicle module for vehicle segmentation.

C. Call the license plate recognition module to identify the license plate information from the segmented vehicle image and return the plate number to the list.

D. Call the license plate recognition module again for any further vehicle image segmentation, identifying the license plate and updating the list.

E. After the video is processed, all detected events in the video will be compiled into an Excel file and sent to the user.

(a)



(b)

(c)

(d)

(e)

Figure 14. (a) Object Detection, (b) Object Tracking, (c) Object Detection, (d) License Plate Recognition, (e) Returning Information

V. Conclusions

This study addresses the issue of littering by drivers in Taiwan and implements a "Deep Learning-based Small Object and Violation Detection System" aimed at effectively solving the environmental pollution and traffic safety risks caused by littering. The combination of YOLOv9E and Deep Sort significantly improves the detection and tracking accuracy of violations. By prioritizing specific objects for processing, the system reduces detection errors for small objects, such as cigarette butts or small pieces of trash, and further optimizes recognition speed.

The contribution of this paper lies in its ability to conduct crowd-sourced violation detection using driving recorder footage in areas without fixed cameras, expanding the monitoring range and improving enforcement efficiency. In the future, this system is expected to be applied in other environmental monitoring or urban management fields, contributing to enhancing urban quality of life and promoting sustainable environmental development.

References

[1]Ministry of Environment (https://data.moenv.gov.tw/dataset/detail/STAT_P_136)

[2] The launch of the multi-directional violation monitoring system (https://news.ebc.net.tw/news/living/184062)

[3] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hari haran, B., &Belongie, S. (2017). Feature Pyramid Networks for Object Detection. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 936–944. doi:10.1109/CVPR.2017.106

[4] Chen, P.-Y., Chang, M.-C., Hsieh, J.-W., & Chen, Y.-S. (2021). Parallel Residual Bi-Fusion Feature Pyramid Network for Accurate Single-Shot Object Detection. IEEE Transac-tions on Image Processing, 30, 9099–9111. doi:10.1109/tip.2021.3118953

[5] Pfeiffer, R., Valentino, G., Farrugia, R. A., Colica, E., D'Amico, S., & Calleja, S. (2022). Detecting beach litter in drone images using deep learning. 2022 IEEE International Workshop on Metrology for the Sea; Learning to Measure Sea Health Parameters (MetroSea), 28–32. doi:10.1109/MetroSea55331.2022.9950804

[6] YOLOv5(https://github.com/ultralyti cs/yolov5)

[7] Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/1506.01497

[8] Jhang, C., Chen, Y., &Jhao, S. (2023). Re-search on Recyclable Waste Identification Method Based on YOLOv8. 13(5), 1019–1025. https: // doi.org/10.12677/CSA.2023.135099

[9] YOLOv8(https://github.com/ultralyt ics/ultralytics)

[10] Kraft, M., Piechocki, M., Ptak, B., &Walas, K. (2021). Autonomous, Onboard Vision-Based Trash and Litter Detection in Low Altitude Aerial Images Collected by an Unmanned Aerial Vehicle. Remote Sensing, 13(5). doi:10.3390/rs13050965

[11] Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal Speed and Ac-curacy of Object Detection. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/2004.10934

[12] UAVVaste dataset (https://www.google.com/search?q=UAVVaste&rlz=1C5CHFA_enTW1083TW1084&oq=UAV-Vaste&gs_lcrp=EgZjaHJvbWUyCwgAEEUYExg5GIAEqAIAsAIA&sourceid=chrome&ie=UTF-8)

[13] COCO dataset(https://cocodataset.org)

[14] Wang, B.-S., Chen, P.-Y., Hsieh, Y.-K., Hsieh, J.-W., Chang, M.-C., He, J., Tseng, Y.-C. (2023). PRB-FPN+: Video Analytics for En-forcing Motorcycle Helmet Laws. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 5477–5485. doi:10.1109/CVPRW59228.2023.00579

[15] Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. M. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/2207.02696

[16] Wang, Y.-H., Hsieh, J.-W., Chen, P.-Y., Chang, M.-C., So, H. H., & Li, X. (2024). SMILEtrack: SiMIlarity LEarning for Occlusion-Aware Multiple Object Tracking. Proceedings of the AAAI Conference on Artificial Intelligence, 38(6), 5740-5748. https://doi.org/10.1609/aaai.v38i6.28386

[17] Chen, X., & He, K. (2020). Exploring Simple Siamese Representation Learning. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/2011.10566

[18] Mills-Tettey, G. A., Stentz, A., & Dias, M. B. (2007). The dynamic Hungarian algorithm for the assignment problem with changing costs. CMU-RI-TR-07-27. Robotics Institute, Carne-gie Mellon University.

[19] https://github.com/WongKinYiu/yolov9

[20] car-plate-li-cense(https://www.kaggle.com/datasets/andrewmvd/car-plate-detection/data)

[21] Wojke, N., Bewley, A., & Paulus, D. (2017). Simple Online and Realtime Tracking with a Deep Association Metric. Proceedings of the IEEE International Conference on Image Processing (ICIP), 3645–3649.

[22] Roboflowdataset(https://universe.rob oflow.com/browse/transportation/cars)