Journal of Telecommunication, Electronic and Computer Engineering
ISSN: 2180 – 1843 e-ISSN: 2289-8131 Vol. 17 No. 3 (2025) 51-58
jtec.utem.edu.my
DOI: https://doi.org/10.54554/jtec.2025.17.03.007

Wireless Surveillance with Human Detection Using Artificial Intelligence and Drone

Muhammad Nasim Sulaiman[1], Muhammad Rusydi Muhammad Razif[2*], Che Aqil Zulhazim Che Hassan[1], Nurul Hasyimah Mohd Mustapha[1], Saiful Wafiy Shahriful Azhar[1], Mohd Norazysyam Azman[3]

[1]*Faculty of Engineering Technology, Universiti Tun Hussein Onn Malaysia, Muar, Johor, 84600, Malaysia.*
[2]*Cybernetics & Power Technology Focus Group, Faculty of Engineering Technology, Universiti Tun Hussein Onn Malaysia, Johor, 84600, Malaysia.*
[3]*Victory Enghoe Plantations Sdn Bhd, Kluang, Johor, 86200, Malaysia.*

Article Info

Article history:
Received Mac 20th, 2025
Revised Jul 21st, 2025
Accepted Sep 30th, 2025
Published Sep 30th, 2025

Index Terms:
Image detection
Internet of Things
Artificial intelligence
Wireless surveillance

Abstract

Conventional human-operated surveillance is vulnerable to errors caused by distractions, fatigue, or biases. The proposed project seeks to improve surveillance effectiveness and precision by utilizing an image detection algorithm and the Internet of Things (IoT). This project addresses the limitations of human detection by deploying a drone equipped with a camera capable of identifying people and suspicious individuals. The system is designed to identify individuals within the drone's visual range, evaluate person detection accuracy, and provide immediate monitoring via IoT connectivity. Person detection is performed using the You Only Look Once (YOLO) algorithm, and the project's scope is limited by camera quality, field of view, and resolution, which limit recognition range. Based on a dataset of 3,500 images, the person detection algorithm achieves a mean average precision (mAP) value of 0.835, a confidence ratio of 0.79, and high accuracy in the confusion matrix. Performance is further improved by integrating OpenVINO with Intel CPU-based computers. The IoT-based monitoring system was implemented through a Streamlit web-based application and Telegram messaging platform. The results show a 2-second delay in streaming from the Raspberry Pi to the host computer, with an average speed of 100 ms per frame.

This is an open access article under the CC BY-NC-ND 4.0 license.

*Corresponding Author: rusydi@uthm.edu.my

I. INTRODUCTION

Unmanned aerial vehicles (UAVs) or drones have emerged as a pivotal technology in surveillance applications due to their versatility and advanced capabilities. Research into UAV applications has attracted significant attention, across various domains such as military operations [1], public health initiatives [2], disaster management [3], and wildlife research [4]. The use of drones for surveillance is well-established, with applications ranging from real-time monitoring of large areas to targeted subject identification and tracking [5]. In public health, drones are used in medical surveillance, disaster site monitoring, and epidemiological studies for tracking the spread of diseases [6]. This broad role demonstrates the importance of drones in modern surveillance systems.

Drone technology has advanced significantly, enabling the development of surveillance systems that provide high-resolution coverage and repeatable monitoring capabilities. These systems increasingly leverage computer vision techniques for comprehensively data analysis [7]. The inherent mobility, flexibility, and adjustable altitude of drones make them well-suited for dynamic observation in remote and inaccessible regions [8]. In wildlife research, drones have revolutionized data collection, enabling researchers to conduct extensive surveillance and apply advanced algorithms data analysis [7]. Additionally, drones have proven invaluable in search and rescue operations during natural disasters, offering immediate airborne surveillance to locate and assist affected individuals [9]. The integration of deep learning and artificial intelligence further enhances drone surveillance by enabling more efficient target identification and recognition [10], [11].

In the image recognition process, automatic supervision plays an important role in improving the security and effectiveness of visual feature detection and identification. Despite progress, challenges remain, especially due to low resolution, variable illumination, and other limiting factors. [12], [13]. To overcome these challenges, pre-processing, segmentation, and target region identification in surveillance videos are typically required for automated visual feature recognition [14]. The quality of images captured by surveillance cameras significantly impacts the effectiveness of image recognition systems, necessitating advanced algorithms to address low-resolution challenges [15]. Research also explores gait analysis as a complement to facial recognition, enhancing identification accuracy in low-resolution settings [16], [17].

To process images captured by cameras, devices such as Raspberry Pi are widely used across domains, including medical applications such as disease detection [18], [19] in agricultural sector [20], [21], such as intelligent farming [22] and security systems for home monitoring and intruder detection [23], [24]. Meanwhile, Rashid *et al.* [25] reported that law enforcement agencies have leveraged Raspberry Pi for real-time facial recognition, demonstrating its capability to enhance security and support individuals with special needs. On Raspberry Pi, image processing typically employs algorithms such as Local Binary Patterns (LBP) for face detection and SQLite for database updates [26] while motion detection using image subtraction algorithms supports real-time alerts and live video streaming capabilities [27]. The adaptability of the Raspberry Pi enables diverse applications, such as social distancing monitoring using the HAAR cascade classifier [28], infant monitoring [29], and vehicle classification in toll systems [30].

AI-based person detection for surveillance has yielded highly efficient monitoring with minimal computational power. Wang et al. [31] introduced a system leveraging a YOLOv8n-based model optimized for low-resource environments to achieve real-time person detection in surveillance footage with high accuracy and minimal computational overhead. Unlike standard person detection systems, which solely identify human presence, this project enhances functionality by incorporating mask detection, capturing screenshots of masked persons, and automatically sending captures via Telegram notifications. This real-time alerting and documentation pipeline represents a meaningful advancement in surveillance applications, enabling faster response and improved traceability compared to basic detection systems.

II. METHODOLOGY

This section outlines the experimental setup and process flow of the proposed system, which integrates a Raspberry Pi, Pi camera, and host computer with a drone for real-time video streaming and image processing. The methodology comprises two main components: hardware setup and software workflow. The hardware setup details the physical arrangement and connections between the components, while the software workflow explains the data transmission, image processing, and detection mechanisms implemented in the system.

Figure 1 illustrates the experimental setup, in which a camera, Raspberry Pi, and host computer are integrated with a drone to enable live video streaming and image processing. The Pi camera, securely mounted on the drone, captures real-time footage and transmits it to the Raspberry Pi for initial processing. The Raspberry Pi, acting as an intermediary device, streams the video feed to the host computer over a Wi-Fi connection using a pre-configured IP address. The host computer, positioned remotely, receives the feed for further image processing and data visualization. This setup ensures seamless communication between the hardware components, enabling efficient real-time monitoring and analysis.

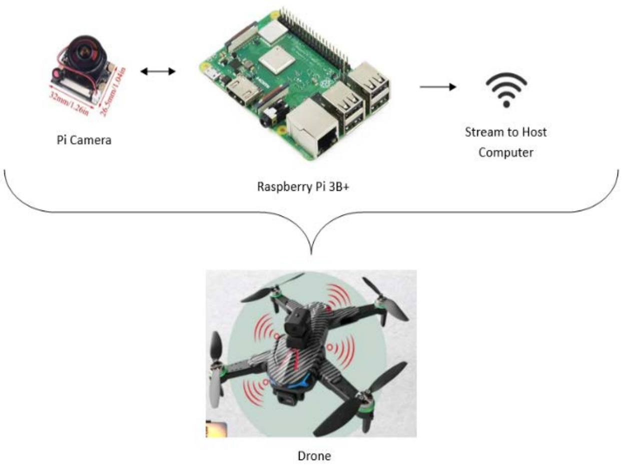


Figure 1. Experiment setup of the project

Figure 2 shows the process flow of the system, which is structured around the interaction between the Raspberry Pi and the host computer. The Raspberry Pi first captures live feed footage from the drone-mounted Pi camera and transmitting it to the host computer via a stable Wi-Fi connection. Proper IP configuration between the Raspberry Pi and the host computer is crucial for ensuring uninterrupted data transmission. Once the host computer receives the video stream, it initializes a virtual camera and executes the main program, which is designed to perform image processing using the You Only Look Once (YOLO) algorithm. The processed data is then visualized through Streamlit, providing real-time insights. If the system detects a person wearing a mask within a video frame, it captures a snapshot and automatically stores the image in a designated folder on the host computer. Additionally, the detection results, including the timestamp, the number of detected individuals, and the file location of the saved images, are compiled for record-keeping. The system remains active until manually stopped using the 'Stop' button or by closing the command prompt.

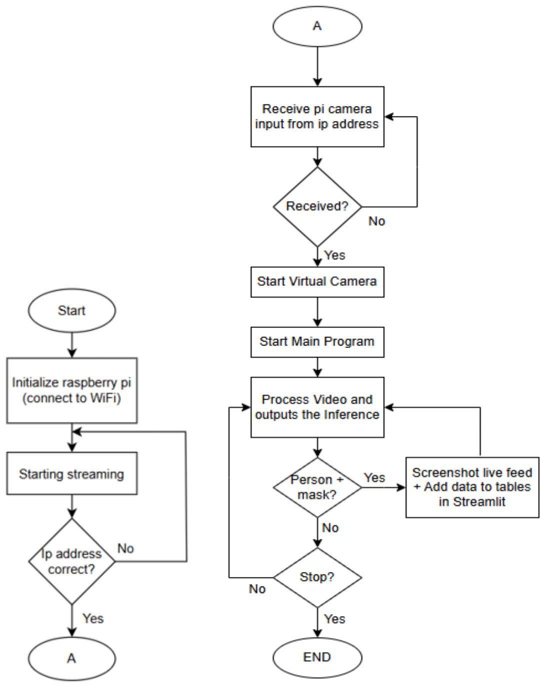


Figure 2. System Flowchart

Figure 3 presents the block diagram of the system, showing the interaction between hardware and software components. The Raspberry Pi 3B+ serves as the core streaming unit, receiving power from a 5V supply to ensure continuous operation. A Pi camera, mounted on the drone, captures real-time footage and transmits it to the Raspberry Pi, which then streams the live video feed to the host computer via Wi-Fi. Then, the host computer processes the received footage using the YOLO algorithm for person detection and visualizes the results through a web-based interface developed with Streamlit. This structured integration enables real-time monitoring and analysis, ensuring efficient detection and data representation. Overall, the synergy between the drone, Raspberry Pi, and host computer provides a responsive platform for real-time data processing and visualization.

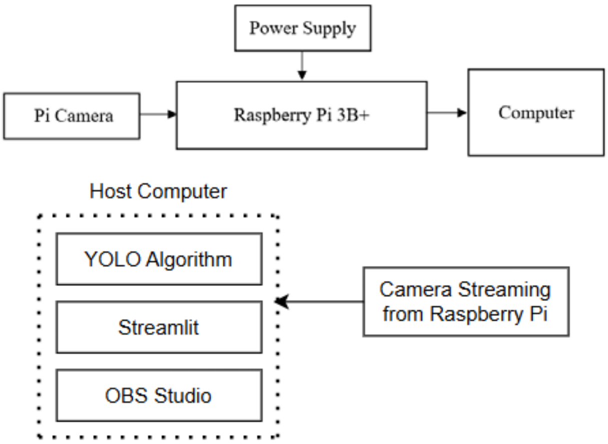


Figure 3. Block diagram of the project

In addition, a custom dataset was built to develop a person and mask detection algorithm based on the YOLO framework. Two versions of the YOLO models—YOLOv8 and YOLOv11—were executed to evaluate their efficiency. Their effectiveness in identifying individuals with and without masks across various environments was systematically tested and recorded. The dataset comprises 3,400 pre-annotated images of individuals wearing masks, sourced from the Roboflow platform [32], complemented by an additional 100 images captured within the drone's field of vision. Figure 4 displays the pre-annotated images obtained from the Roboflow platform, while Figure 5 illustrates sample images featuring both masked and unmasked individuals within a single frame.

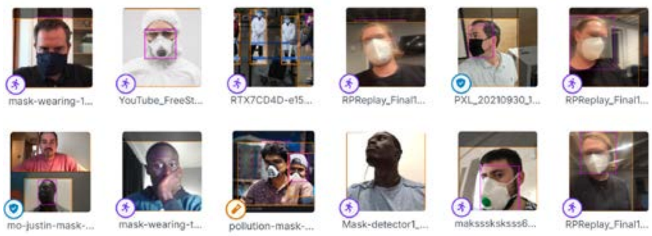


Figure 4. pre-annotated images from Roboflow

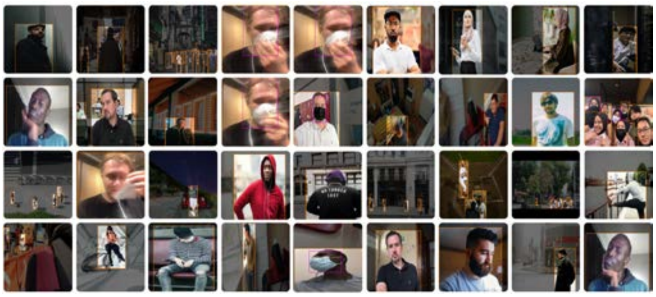


Figure 5. Sample of labeled datasets

III. RESULTS AND DISCUSSION

This section presents experimental results and discussions on the performance of YOLOv8 and YOLOv11 models using a custom dataset for person and mask detection. The evaluation considers key criteria, including completion time, model complexity, mAP50, and inference time. In addition, the system was tested under varied environmental conditions to assess its effectiveness in detecting individuals and masks in real-world scenarios. The results are presented using the Streamlit web application and Telegram app, which enable real-time detection and notification, demonstrating the system's potential as a comprehensive IoT solution for real-time monitoring.

A. Dataset Validation

Once the custom dataset for person and mask detection has been labeled, YOLOv8 and YOLOv11 models were trained. This step was essential to validate the dataset and assess the performance of both models. The training took place in an Anaconda environment using YOLO's command-line interface. The models were trained for 75 epochs, as prior experiments indicated that optimal performance is typically achieved at or around 50 epochs. Training was performed on a high-performance workstation equipped with an NVIDIA GeForce RTX 4070 GPU, enabling faster processing times and efficient utilization of computational resources. The image size for training was set to 640 pixels, balancing detection accuracy and computational efficiency.

Table 1 summarizes the training results, highlighting distinct differences between YOLOv8 and YOLOv11. In terms of completion time, YOLOv11 finished training in 0.711 hours compared to YOLOv8's 0.806 hours, indicating a faster convergence. Regarding model complexity, YOLOv8 operates at 67.7 GFLOPs, which is lower than YOLOv11's 78.7 GFLOPs, suggesting that YOLOv8 is more computationally efficient. In terms of accuracy (mAP50), both models achieved nearly identical performance, with YOLOv8 (0.578) slightly outperforming YOLOv11 (0.577). Finally, for inference time, YOLOv11 exhibited a significant advantage by processing images in just 3.2 ms compared to YOLOv8's 6.5 ms, making it more suitable for real-time applications.

Table 1
Summary of training results

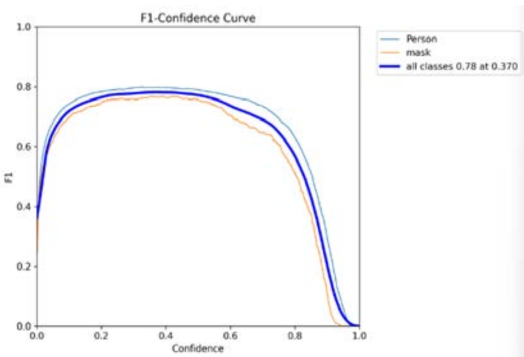
Performance	YOLOv8	YOLOv11
Completion time	0.806 hours	0.711 hours
Model complexity	67.7 GFLOPs	78.7 GFLOPs
mAP50	0.578	0.577
Inference time	6.5 ms	3.2 ms

B. Fi-Confidence Curve

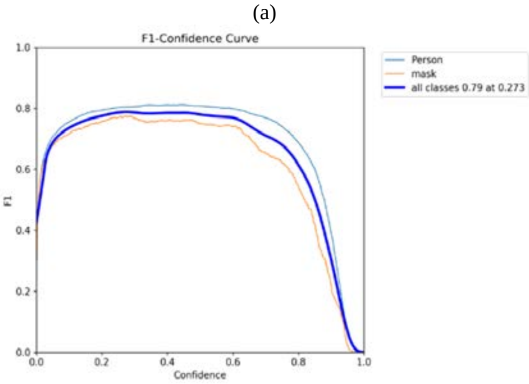
The F1-Confidence curve measures variations in the F1 score as the confidence threshold changes. The F1 score is a composite metric that evaluates a model's accuracy by considering both precision and recall. A higher F1 score indicates better overall performance.

Figure 6 shows the F1-Confidence curves for both YOLOv8 and YOLOv11 across the two classes which are person and mask. From these curves, YOLOv8 achieves its maximum F1 score of 0.78 at a confidence threshold of 0.370, whereas YOLOv11 reaches a slightly higher maximum F1 score of 0.79 at a lower threshold of 0.273. This suggests that YOLOv11 requires a lower confidence threshold to optimize its overall performance compared to YOLOv8.

For the person detection class, both models achieve their highest F1 scores, with YOLOv11 slightly outperforming YOLOv8 at higher confidence levels, as evidenced by a higher peak in its curve. In contrast, for the mask detection class, both models exhibit lower F1 scores compared to the person detection task. The F1 score for mask detection declines more sharply as the confidence threshold increases, indicating that both models struggle to maintain precision for mask detection when the threshold increases.



(a)



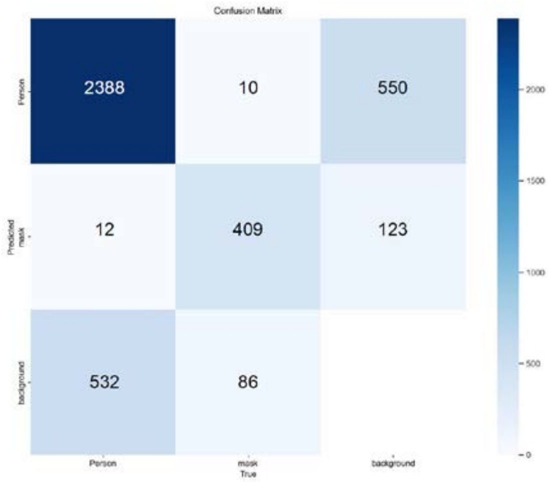
(b)

Figure 6. F1-Confidence curve: (a) YOLOv8; (b) YOLOv11

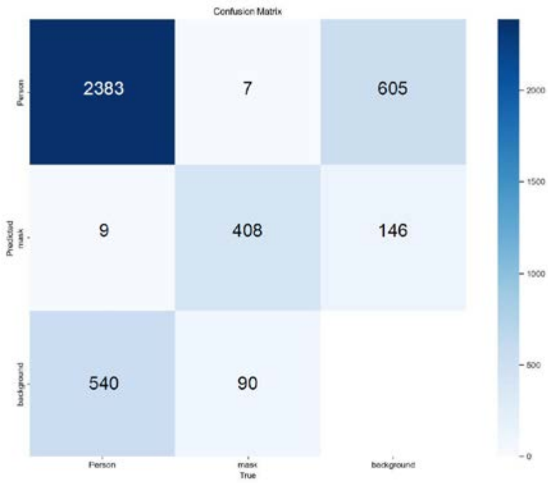
In summary, YOLOv11 is better at maintaining overall balance (precision and recall) for all classes, especially at lower confidence thresholds. YOLOv8 demonstrates competitive performance but appears better suited to higher confidence thresholds, making it slightly less adaptable in scenarios with varying detection challenges. Mask detection remains a challenge for both models, with significant room for improvement.

C. Confusion Matrix

The confusion matrix is a tool used to evaluate a classification model by comparing predicted labels against actual labels. It provides a detailed breakdown of a model's performance across different classes, making it an essential approach to analyze classification accuracy and identifying errors. Figure 7 shows the confusion matrices resulting from training YOLOv8 and YOLOv11.



(a)



(b)

Figure 7. Confusion matrix: (a) YOLOv8; (b) YOLOv11

Based on Figure 7, the matrices summarize the number of correct and incorrect predictions for each of the three categories which are "Person," "Mask," and "Background." For YOLOv8, the person detection class shows 2383 true positives (correct detections), 7 false negatives (missed detections), and 605 false positives (misclassifications as person). This indicates that YOLOv8 is highly accurate in detecting persons, though it occasionally misclassifies objects from other categories as persons. For the mask detection class, YOLOv8 records 408 true positives, 9 false negatives, and 146 false positives, suggesting that the model struggles to accurately detecting masks compared to persons.

Additionally, the model misclassifies 540 background objects as either persons or masks.

In the case of YOLOv11, the person detection results show a slight improvement, with 2388 true positives, 10 false negatives, and 550 false positives, indicating a marginal enhancement in detecting persons with fewer misclassifications. For mask detection, YOLOv11 achieves 409 true positives, 12 false negatives, and 123 false positives, which is comparable to YOLOv8 but with a slight increase in false negatives. Moreover, YOLOv11 misclassifies 532 background objects as either persons or masks, a small improvement over YOLOv8's results. In summary, both models demonstrate high accuracy, with YOLOv11 marginally outperforming YOLOv8 in terms of slightly higher true positives, fewer false positives, and reduced background misclassifications.

D. Detection Performance in Various Conditions

The detection test evaluated the performance of YOLO models in identifying individuals under various conditions, including those wearing face masks. The primary objective was to assess the models' accuracy, detection capabilities, and limitations across different environments. The test was conducted in three settings which are a public area during peak hours to simulate a high-density crowd, a home environment representing a high dynamic range scene, and a drone field of view. Additionally, the test aimed to identify instances of false negatives, that is cases where the model failed to correctly detect persons or masks, and to measure the detection range limit. For detecting individuals wearing masks, the system was tested using masks of different colors, specifically black and white, to evaluate its sensitivity to color variations.

Figure 8 illustrates a comparison of detection performance in different lighting environments. The left image, captured in a low-brightness setting, shows that the system successfully detects a person with a confidence score of 0.79 but fails to detect the mask worn by the individual. In contrast, the right image, taken in a well-lit laboratory, demonstrates that the system effectively detects both the person (0.88) and the mask (0.37) as the person walks. These results indicate that improved lighting conditions significantly enhance detection accuracy, emphasizing the critical role of adequate illumination in achieving reliable detection performance.

Figure 8. Detection performance under varied lighting conditions: (Left) Low-brightness environment; (Right) Bright environment

Figure 9 presents detection tests conducted in an open space during the evening under adequate lighting conditions. In the left image, a person moving and facing directly toward the camera, is detected at a distance of 5 meters, demonstrating the system's ability to accurately track subjects even in dynamic conditions. The middle image shows a person wearing a bright mask being reliably detected at a greater distance, further validating the system's performance in identifying masked individuals. In the right image, while the system successfully detects a person wearing a dark mask at 5 meters, it also registers a false detection by mistakenly identifying a red car as a person, albeit with a low confidence score of 0.60. Overall, these results highlight the system's effectiveness in various detection scenarios while also indicating areas for further improvement to reduce misclassifications.

Figure 9. Detection performance in an open space under adequate lighting conditions: (Left) A moving person at less than 5 meters; (Middle) A person wearing a bright mask at 5 meters; (Right) A person wearing a dark mask at 5 meters

E. Real-Time Detection and IoT Notification

The system records a suspicious person when two conditions are met which are both a person and a mask are detected simultaneously in a single frame. Once detection occurs, the system logs the data in the Streamlit web app, including the timeframe, person count, and confidence ratio, while saving a corresponding screenshot in a designated directory. Figure 10 shows a sample table in the Streamlit web app displaying these suspicious person records. Additionally, a screenshot of the detection frame is saved as an image file in a dedicated local computer folder, as illustrated in Figure 11.

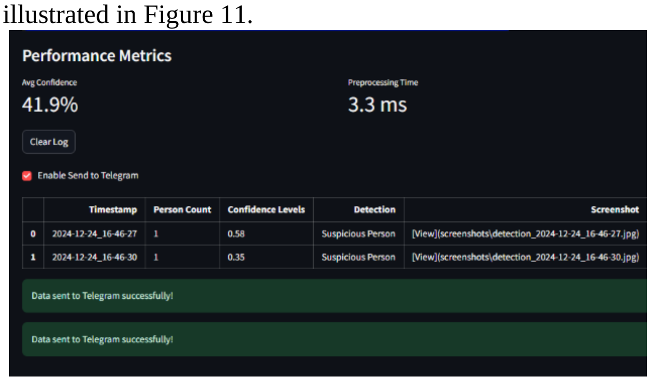


Figure 10. Streamlit web apps interface and records

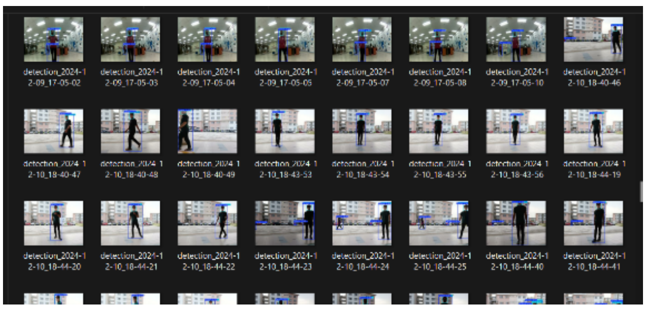


Figure 11. Saved images in local computer folder

Beyond local storage, a function for transmitting detection data to Telegram has been implemented, enabling remote access via the internet. This IoT feature can be enabled or disabled through a checkbox in the Streamlit app. When activated, the system forwards the detection data, both the table records and the saved images, to Telegram via a chatbot. Figure 12 illustrates the Telegram interface, displaying a message with the detection data and an accompanying image. In summary, the integration of the Streamlit web app with Telegram notifications enhances security by providing both local and remote access to detection records, ensuring that critical information is readily available to security personnel.

Figure 12. Telegram chats with detection records

F. Detection Test with Drone

A drone detection test was performed to evaluate the system's performance in detecting persons and masks, with both the detection range and false positives recorded. The experimental setup shown in Figure 13, consists of a Raspberry Pi mounted beneath the drone and a Raspberry Pi camera positioned on top. Due to the limitation of providing wired power to the Raspberry Pi, a power bank was used to supply power and the power bank was held separately during flight.


Figure 13. Drone setup

Based on the drone testing, the detection system accurately recognized a person at an altitude of approximately 2 meters, with effective person detection observed up to a range of 7 meters. However, mask detection was only effective within approximately 2 meters in front of the drone. Figure 14 shows detection results, highlighting a person wearing a mask detected at 2 meters, while at 7 meters the system detected the person but was unable to identify the mask. In summary, while the detection system delivers accurate results, it is constrained by practical limitations. The use of a handheld power bank to power the Raspberry Pi restricts the drone's operational autonomy, and the camera's quality limits the effective range for mask detection.

(a)

(b)

Figure 14. Drone-based detection: (a) Close-range detection of both person and mask; (b) Long-range detection of person only.

G. Security and Privacy

The use of drone-based surveillance with real-time person and mask detection using the YOLO model inevitably raises privacy and ethical concerns. Filming individuals in public or semi-public areas without prior notice can be perceived as intrusive. To address this, several responsible measures are proposed: data anonymization, transparent deployment, and strict adherence to local laws.

Personally identifiable information (PII) is not stored unless strictly necessary. Drone footage is retained only if a suspicious individual is detected. The public is informed through appropriate signage or announcements about areas under drone surveillance. All operations will comply with relevant privacy regulations, such as the Personal Data Protection Act (PDPA) 2010 in Malaysia, ensuring proper permissions and responsible data handling.

IV. CONCLUSION

In conclusion, this project successfully developed a wireless surveillance system using a Raspberry Pi to detect individuals wearing masks. The YOLO algorithm was applied for human detection, with performance compared between two versions, namely YOLOv8 and YOLOv11. The system with YOLOv11 achieved optimized detection accuracy, achieving a mAP of 0.578, an F1-confidence score of 0.79, and an inference time of 3.2 ms. It delivered excellent results with minimal false positives, although some misclassifications occurred, primarily due to object detection at long distances that yielded low confidence scores.

The system effectively detected individuals at distances up to 50 meters and recognized masked persons within a 5-meter range. A Streamlit web application was developed for real-time monitoring and recording of detections, incorporating IoT, allowing users to view results via Telegram. Additionally, the system was mounted on a drone for wireless surveillance, demonstrating promising practical applications despite some limitations.

For future improvements, it is recommended to use a higher-quality camera to enhance detection accuracy, expand the dataset to 6,000 images for better model training, and upgrade the hardware, for example replacing the power bank and cable with a Li-Po battery to power the Raspberry Pi, and upgrading the drone for improved flying efficiency. Furthermore, developing a more user-friendly setup would facilitate seamless Raspberry Pi streaming and overall system operation.

ACKNOWLEDGMENT

This research was supported by the Ministry of Higher Education (MOHE) through the Fundamental Research Grant Scheme (FRGS) (FRGS/1/2022/TK07/UTHM/02/3) and Universiti Tun Hussein Onn Malaysia (UTHM) through GPPS (vot Q811). The authors would like to thank the Faculty of Engineering Technology, University Tun Hussein Onn Malaysia for providing the necessary research facility for this study.

CONFLICT OF INTEREST

Authors declare that there is no conflict of interest regarding the publication of the paper.

AUTHOR CONTRIBUTION

The authors confirm contribution to the paper as follows: study conception and design: Muhammad Nasim Sulaiman and Che Aqil Zulhazim Che Hassan; data collection: Muhammad Nasim Sulaiman and; analysis and interpretation of findings: Muhammad Rusydi Muhammad Razif and Mohd Norazysyam Azman; draft manuscript preparation: Muhammad Nasim Sulaiman and Nurul Hasyimah Mohd Mustapha. All authors had reviewed the findings and approved the final manuscript.

REFERENCES

[1] Salam, A., & Mohamed, M., “Selection of Military UAV using LMAW and TOPKOR Methods in Neutrosophic Environment”, Multicriteria Algorithms with Applications, 6, 34-56, 2025. doi: 10.61356/j.mawa.2025.6452.

[2] Zhou, Yanan, et al. "Monitoring soil arsenic content in densely vegetated agricultural areas using UAV hyperspectral, satellite multispectral and SAR data." Journal of Hazardous Materials 484, pp.136689, 2024., doi: 10.1016/j.jhazmat.2024.136689.

[3] Bartlett, B., Santos, M., Dorian, T., Moreno, M., Trslic, P., & Dooly, G., “Real-Time UAV Surveys with the Modular Detection and Targeting System: Balancing Wide-Area Coverage and High-Resolution Precision in Wildlife Monitoring”, Remote Sensing, 17(5), 879, 2025. doi: 10.3390/rs17050879.

[4] M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, and M. Debbah, “A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems,” IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2334–2360, 2019, doi: 10.1109/COMST.2019.2902862.

[5] S. Park, H. T. Kim, S. Lee, H. Joo, and H. Kim, “Survey on Anti-Drone Systems: Components, Designs, and Challenges,” IEEE Access, vol. 9, pp. 42635–42659, 2021, doi: 10.1109/ACCESS.2021.3065926.

[6] J. C. Rosser, V. Vignesh, B. A. Terwilliger, and B. Parker, “Surgical and Medical Applications of Drones: A Comprehensive Review,” Journal of the Society of Laparoendoscopic Surgeons, vol. 22, no. 3, p. e2018.00018, May 2018, doi: 10.4293/jsls.2018.00018.

[7] G. D. Larsen and D. W. Johnston, “Growth and opportunities for drone surveillance in pinniped research,” Mamm Rev, vol. 54, no. 1, pp. 1–12, May 2023, doi: 10.1111/mam.12325.

[8] C. Singhal and S. Barick, “ECMS: Energy-Efficient Collaborative Multi-UAV Surveillance System for Inaccessible Regions,” IEEE Access, vol. 10, pp. 95876–95891, 2022, doi: 10.1109/ACCESS.2022.3206375.

[9] B. Mishra, D. Garg, P. Narang, and V. K. Mishra, “Drone-surveillance for search and rescue in natural disaster,” Comput Commun, vol. 156, pp. 1–10, May 2020, doi: 10.1016/j.comcom.2020.03.012.

[10] F. Samadzadegan, F. D. Javan, F. A. Mahini, and M. Gholamshahi, “Detection and Recognition of Drones Based on a Deep Convolutional Neural Network Using Visible Imagery,” Aerospace, vol. 9, no. 1, p. 31, May 2022, doi: 10.3390/aerospace9010031.

[11] Wang, Shaohua, et al. "Advances in Deep Learning Applications for Plant Disease and Pest Detection: A Review." Remote Sensing 17.4, pp.136689, 2025, doi: 10.3390/rs17040698.

[12] Z. Cheng, X. Zhu, and S. Gong, “Surveillance Face Recognition Challenge,” arXiv Cornell University, May 2018, doi: 10.48550/arxiv.1804.09691.

[13] B. Li, T. Li, and D. Huang, “Research on Automatic Fire Monitoring and Recognition Technology Based on Digital Image Processing Technology,” Proceedings of the 2018 8th International Conference on Management, Education and Information (MEICI 2018), vol. 163, pp. 1–6, May 2018, doi: 10.2991/meici-18.2018.16.

[14] B. K. Güntürk, A. U. Batur, Y. Altunbasak, M. H. Hayes, and R. M. Mersereau, “Eigenface-domain super-resolution for face recognition,” IEEE transactions on image processing, vol. 12, no. 5, pp. 597–606, May 2003, doi: 10.1109/tip.2003.811513.

[15] Liu, N., Lim, M. H., Yuen, P. C., & Lai, J. H., “Image Set-Based Face Recognition: A Local Multi-keypoint Descriptor-Based Approach”, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 160-165, 2013.

[16] X. Xu, W. Liu, and L. Li, “Low Resolution Face Recognition in Surveillance Systems,” Journal of computer and communications, vol. 02, no. 02, pp. 70–77, May 2014, doi: 10.4236/jcc.2014.22013.

[17] O. A. Aghdam, B. Bozorgtabar, H. K. Ekenel, and J.-P. Thiran, “Exploring Factors for Improving Low Resolution Face Recognition”, 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2019, pp. 2363–2370. doi: 10.1109/CVPRW.2019.00290.

[18] Alaidi, A. H. M., Ramadhan, Z. A., Alrubaye, J. S., ALRikabi, H. T. S., Mutar, H. A., & Svyd, I., “AI-based monkeypox detection model using Raspberry Pi 5 AI Kit”, Sustainable Engineering and Innovation, 7(1), pp. 1-14, 2025, doi: 10.37868/sei.v7i1.id393.

[19] Mehta, Samaira. "OVision A raspberry Pi powered portable low cost medical device framework for cancer diagnosis." Scientific Reports 15.1, pp. 7124, 2025, doi: 10.1038/s41598-025-91914-z.

[20] H. Rahmat, S. Wahjuni, and H. Rahmawan, “Performance Analysis of Deep Learning-based Object Detectors on Raspberry Pi for Detecting Melon Leaf Abnormality,” International Journal on Advanced Science, Engineering and Information Technology, vol. 12, no. 2, pp. 572, May 2022, doi: 10.18517/ijaseit.12.2.13801.

[21] V. González-Huitrón, J. A. León-Borges, A. E. Rodríguez-Mata, L. E. Amábilis-Sosa, B. Ramírez-Pereda, and H. Rodríguez, “Disease detection in tomato leaves via CNN with lightweight architectures implemented in Raspberry Pi 4,” Comput Electron Agric, vol. 181, pp. 105951, May 2021, doi: 10.1016/j.compag.2020.105951.

[22] Y. Zhong, J. Gao, Q. Lei, and Y. Zhou, “A Vision-Based Counting and Recognition System for Flying Insects in Intelligent Agriculture,” Sensors, vol. 18, no. 5, pp. 1489, May 2018, doi: 10.3390/s18051489.

[23] I. G. M. N. Desnanjaya and I. N. A. Arsana, “Home security monitoring system with IoT-based Raspberry Pi,” Indonesian journal of electrical engineering and computer science, vol. 22, no. 3, pp. 1295, May 2021, doi: 10.11591/ijeecs.v22.i3.pp1295-1302.

[24] N. Surantha and W. R. Wicaksono, “An IoT based House Intruder Detection and Alert System using Histogram of Oriented Gradients,” Journal of computer sciences/Journal of computer science, vol. 15, no. 8, pp. 1108–1122, May 2019, doi: 10.3844/jcssp.2019.1108.1122.

[25] S. T. Rashid, N. S. Jumaah, and R. Talib, “Real-time Face Recognition for Enhanced Law-Enforcement Services in Cities,” Wasit journal for pure sciences, vol. 1, no. 1, pp. 1–7, May 2022, doi: 10.31185/wjps.10.
[26] A. S. Nadhan et al., “Smart Attendance Monitoring Technology for Industry 4.0,” J Nanomater, vol. 2022, no. 1, pp. 1–9, May 2022, doi: 10.1155/2022/4899768.
[27] S. F. AG, K. H. Fen, and others, “Development of a portable community video surveillance system,” International Journal of Electrical and Computer Engineering, vol. 9, no. 3, pp. 1814, 2019.
[28] M. I. Maulana, I. G. P. W. Wedashwara, and A. Zubaidi, “Deteksi Pelanggaran Social Distancing Menggunakan Raspberry Pi Kamera Berbasis IOT,” J-Cosine (Journal of Computer Science and Informatics Engineering), vol. 6, no. 1, pp. 39–46, May 2022, doi: 10.29303/jcosine.v6i1.417.
[29] Y. K. Dubey and S. Damke, “Baby monitoring system using image processing and IoT,” Int J Eng Adv Technol, vol. 8, no. 6, pp. 4961–4964, 2019.
[30] A. Suryatali and V. B. Dharmadhikari, “Computer vision based vehicle detection for toll collection system using embedded Linux,” in 2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015], pp. 1–7, 2015, doi: 10.1109/ICCPCT.2015.7159412.
[31] Q. Wang, G. Feng, and Z. Li, “A Lightweight Person Detector for Surveillance Footage Based on YOLOv8n,” Sensors, vol. 25, no. 2, 2025, doi: 10.3390/s25020436.
[32] Joseph Nelson, “Mask and Person Detection Computer Vision Dataset by Joseph Nelson,” Roboflow, Accessed: Jan. 2024, [Online]. Available: https:// https://universe.roboflow.com/joseph-nelson/mask-wearing.