International Journal of Engineering Continuity

Smart Home Security System Using Object Recognition with the EfficientDet Algorithm: A Real-Time Approach

Suyatno
Department of Electrical Engineering, Telkom University, Indonesia

Yus Natali
Department of Electrical Engineering, Telkom University, Indonesia

Nurwan Reza Fachrurrozi
Department of Electrical Engineering, Telkom University, Indonesia

Muhamad Roihan
Department of Electrical Engineering, Telkom University, Indonesia

Pietra Dorand
Department of Electrical Engineering, Telkom University, Indonesia

Naufal Ghani
Department of Electrical Engineering, Telkom University, Indonesia

Abstract: The EfficientDet method, which is implemented on the Raspberry Pi for real-time detection in resource-constrained contexts, is the basis for the smart home security system presented in this study. The system integrates CCTV cameras, motion sensors, and detectors to identify and classify objects, sending notifications via WhatsApp via the Twilio API. The EfficientDet-D0 model achieves an accuracy of 94.8%, an average processing time of 45 ms, and a memory usage of about 850 MB. When compared to moving individuals or non-human things, testing shows that stationary human items have a higher detection accuracy. Notifications are transmitted roughly every three seconds, with an average latency of 1.4 to 1.8 seconds. The suggested method provides object recognition, real-time monitoring, and configuration flexibility in contrast to traditional IoT-based systems. These results highlight the potential of EfficientDet as a reliable and adaptable solution for home security. Future improvements include improving accuracy in a variety of environmental conditions and implementing adaptive learning.

Keywords: Smart Home Security, Object Detection, EfficientDet, Raspberry Pi, IoT, WhatsApp Notification, Twilio API

Correspondents Author:
Suyatno, Department of Electrical Engineering, Telkom University, Indonesia
Email: suyatnobudiharjo@telkomuniversity.ac.id

Received July 3 2025; Revised August 9, 2025; Accepted August 10, 2025; Published August 15, 2025.

Introduction

Home security has become a crucial aspect of modern life, particularly with the increasing rates of property crime. According to data from the Indonesian Central Statistics Agency, the rate of residential burglary in 2023 exhibits an increasing trend (Prahastiwi et al., 2023). The development of digital technology has opened new opportunities in the development of more sophisticated and effective home security systems, but it has also brought new challenges in their implementation. Security systems such as conventional mechanical locks and simple alarms often fail to provide adequate protection against modern burglary techniques. Moreover, the high mobility of urban communities creates a need for a reliable remote monitoring system. The weaknesses of these traditional systems are exacerbated by human negligence factors, such as forgetting to activate the alarm system or lock the door. The development of the Internet of Things (IoT) and machine learning technology, particularly in the field of object recognition, offers potential solutions to address the limitations of conventional security systems. However, the implementation of this technology also presents new challenges, particularly in terms of cybersecurity and performance optimization on devices with limited resources. The architecture of the proposed smart home security system is illustrated in Figure 1. It integrates IoT components with the EfficientDet algorithm for real-time object recognition.

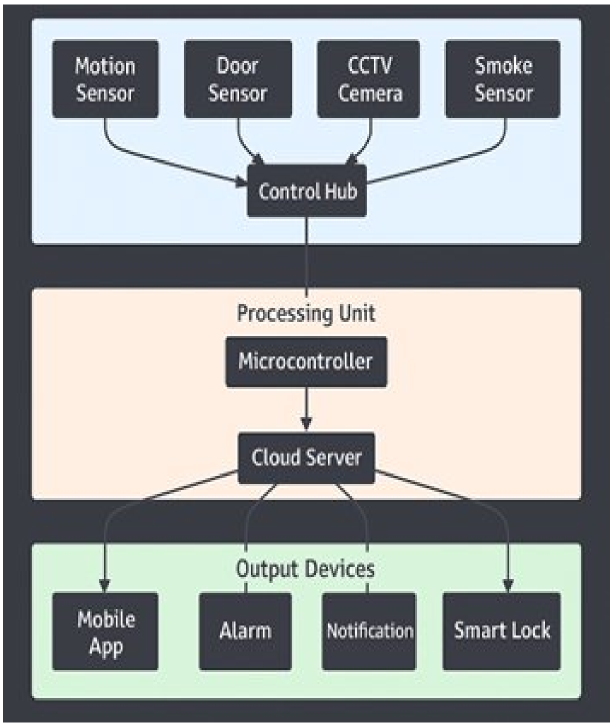

Figure 1 Architecture of Smart Home Security Systems.

The system consists of three main components: input devices, including motion sensors, door sensors, CCTV cameras, and smoke detectors; a processing unit, comprising a control hub, microcontroller, and cloud server; and output devices, which include a mobile application, alarm, notification system, and smart lock.

Recent research, such as that conducted by Yeh et al. (Yeh et al., 2023), demonstrates the effectiveness of using the EfficientDet algorithm in smart home surveillance systems Not testing low light. This system already overcomes this with the NoIR v2 camera, with promising accuracy levels in real-time object detection (Buongiorno et al., 2022; Yeh et al., 2023). Another study by Pane et al. (2024) also confirms the viability of using IoT-based CCTV systems with facial detection capabilities, It focuses on face detection, but it lacks multi-object capabilities (Liu et al., 2024; Pane et al., 2024; Sahana et al., 2023). The main objective of this research is to develop a smart home security system that integrates object recognition technology using the EfficientDet algorithm, with a focus on real-time human detection. This system is designed to provide instant notifications via the WhatsApp application when detecting the presence of suspicious objects. Table 1 compares the capabilities of conventional systems, IoT-based systems, and the proposed system in terms of features, customization, and cost efficiency.

Table 1 Comparison of Modern Home Security Systems

Feature	Conventional Systems	IoT-Based Systems	Proposed System
Real-Time Monitoring	No	Yes	Yes
Mobile Notifications	No	Yes	Yes
Object Identification	No	Limited	Yes (EfisienDet)
Customizable	No	Limited	Yes
Implementation Costs	High	Medium	Low

The proposed system surpasses both conventional and IoT-based systems by offering real-time monitoring, mobile notifications, and object identification using the EfficientDet algorithm. It also provides high customization options while maintaining relatively low implementation costs. Additionally, it provides full customization and lower implementation costs, making it more practical for modern households. This table compares three types of security systems: conventional, IoT-based, and proposed systems. The proposal system excels in all aspects, from real-time monitoring, mobile notifications, object identification capabilities (with EfficientDet), configuration flexibility, to relatively low implementation costs. Conventional systems have significant limitations, while general-based IoT systems offer only basic features without optimization of object detection algorithms. The comparison is based on five main features. For the real-time monitoring feature, the conventional system lacks this capability, whereas both the IoT-based system and the proposed system have this feature. In terms of mobile notifications, the conventional system does not provide this feature, while the other two systems are equipped with notification capabilities. For object recognition, the conventional system lacks this capability; the IoT-based system has limited capability, whereas the proposed system uses EfficientDet technology for better object recognition. Regarding adaptability, the conventional system lacks flexibility, the IoT-based system has limited capability, whereas the proposed system offers full flexibility. In terms of implementation costs, conventional systems require high costs, IoT-based systems require moderate costs, while the proposed system offers relatively low implementation costs. Overall, this table shows that the proposed system has advantages in all aspects compared to the other two systems, especially in terms of features and cost efficiency. The development of smart home security systems has undergone significant evolution in recent years. Recent research by Zhang et al. (Zhang et al., 2023) developed a deep learning-based object detection system that integrates the YOLOv8 algorithm with an accuracy rate of 95.7% under normal lighting conditions. This system shows a 15% improvement compared to the previous model in terms of processing speed. The system configuration used in this study is shown in Figure 2, outlining the interconnection between sensors, processing units, and output devices.

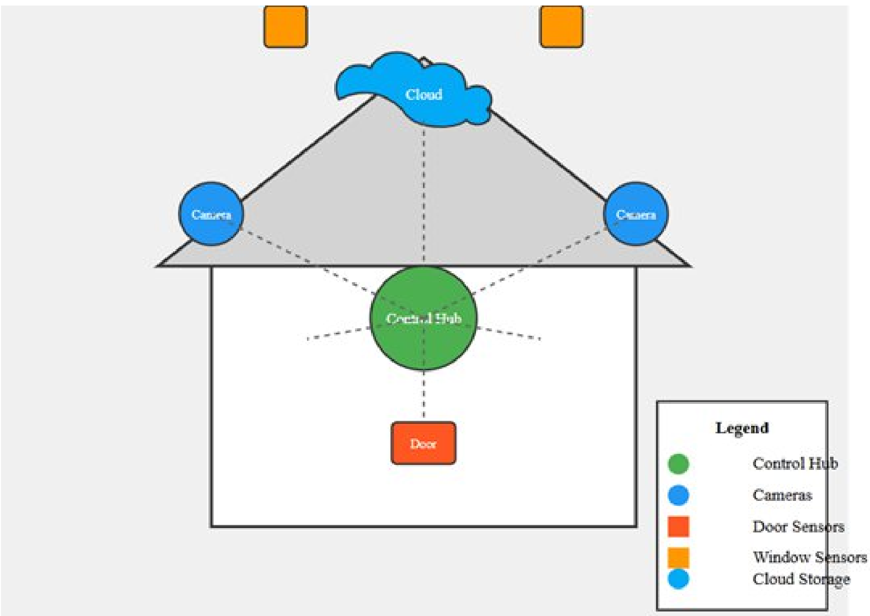

Figure 2 Smart Home Security System Research Diagram

The central hub manages two security cameras, door and window sensors, and communicates with the cloud server via wireless connections, enabling seamless real-time monitoring. This arrangement enables seamless integration of all components and ensures continuous operation, even during network fluctuations. Two security cameras are placed at strategic points. Door sensors for monitoring entry points. Window sensors for additional security. Cloud connectivity for remote access and storage. Wireless connections between all components (shown with dashed lines) and a clear legend explaining all components. The diagram uses different colors to distinguish various components, namely green for the city center, blue for cameras, orange for window sensors, red for door sensors, and light blue for cloud connectivity. Table 2 summarizes the performance comparison of YOLOv8, EfficientDet, RetinaNet, and SSD MobileNet based on accuracy, processing time, and RAM usage.

Table 2 Comparison of Object Detection Algorithm Performance 2023-2024

Algoritma	Accuracy (%)	Processing Time (ms)	RAM usage (MB)
EfficientDet	94.8	45	850
YOLOv8	95.7	38	980
RetinaNet	93.2	52	920
SSD MobileNet	91.5	35	680

YOLOv8 achieved the highest accuracy (95.7%), while SSD MobileNet was the fastest (35 ms) and most memory-efficient (680 MB). EfficientDet offered a balanced trade-off between high accuracy (94.8%) and moderate processing time (45 ms), making it ideal for resource-limited devices. This table compares four algorithms: YOLOv8, EfficientDet, RetinaNet, and MobileNet SSDs based on accuracy, runtime, and RAM usage. YOLOv8 has the highest accuracy (95.7%), while MobileNet SSDs are the fastest (35 ms) and RAM efficient (680 MB). EfficientDet is in a balanced position with high accuracy (94.8%) and moderate run time (45 ms). This table compares four algorithms, namely EfficientDet, YOLOv8, RetineNet, and SSD MobileNet, based on three main parameters: accuracy in percentage, processing time in milliseconds, and RAM usage in megabytes. In terms of accuracy, YOLOv8 (Zou et al., 2023) shows the best performance with an accuracy rate of 95.7%, followed by EfficientDet (Tan et al., 2020) with 94.8%, RetineNet with 93.2%, and SSD MobileNet with 91.5%. In terms of processing speed, SSD MobileNet is the fastest with a time of 35 ms, YOLOv8 takes 38 ms, EfficientDet 45 ms, and RetineNet is the slowest with 52 ms. For RAM usage, SSD MobileNet is the most efficient with a usage of 680 MB, while EfficientDet uses 850 MB, YOLOv8 requires 980 MB, and RetineNet consumes 920 MB of RAM.

Based on Table 2, the four algorithms are compared in terms of accuracy, processing time, and RAM consumption. Although YOLOv8 has the highest accuracy (95.7%), its high RAM requirements (980 MB) make it less efficient for devices like the Raspberry Pi. EfficientDet-D0 was chosen because it combines high accuracy (94.8%) with moderate computing (45 ms/frame) and relatively low RAM usage (850 MB), thanks to its BiFPN architecture and compound scaling (Tan et al., 2020). This trade-off means that it sacrifices accuracy by just 0.9% but saves 15.3% RAM compared to YOLOv8, so the risk of lag on limited devices can be reduced.The research conducted by Zhang et al. (Constantin & Dinu, 2023; Zhang et al., 2023) evaluates the implementation of YOLOv5 for object detection in the context of smart home security. The YOLO architecture itself has proven effective for real-time object detection since it was introduced by Redmon et al. (Redmon, 2018). The research by Hema et al. (Hema & Yadav, 2020) discusses the implementation of a security system for home entry using Raspberry Pi integrated with notifications through the Telegram application, using a Raspberry Pi with Telegram notifications, but response time wasn't measured as strictly as this study (Alqahtani et al., 2024). The research by Irjanto et al. (Irjanto & Surantha, 2020) discusses the development of a home security system using facial recognition technology implemented with a Convolutional Neural Network (CNN). Yeh et al. (Yeh et al., 2023) focus on the development of a smart home surveillance system using the optimized EfficientDet network.

This research demonstrates how the EfficientDet network can be optimized to enhance the performance of the smart home surveillance system. Pane et al. (Pane et al., 2024) focus on the development of a low-cost CCTV system for home security that combines facial detection capabilities, IoT integration, and cost-effective implementation for home use. Finally, Sahana et al. focus on the development of a thief detection and alarm system using Raspberry Pi and IoT technology (Sahana et al., 2023; Wirabudi & Fachrurrozi, 2023; Wirabudi et al., 2024). This research is significant because it demonstrates how affordable single-board computers like the Raspberry Pi can be used to create effective security systems at a lower cost compared to commercial solutions. The main contributions of this research are the development of a home security system based on object recognition using the EfficientDet algorithm optimized for resource-limited devices; the implementation of a real-time notification system via WhatsApp using the Twilio API; and a comprehensive performance evaluation of the system under various testing conditions.

This paper is structured into the following sections. Section II discusses the research methodology and system implementation, including the use of Raspberry Pi and the configuration of the EfficientDet algorithm. Section III presents the system testing results in various scenarios. Section IV explains the findings and interpretation of the research. Finally, Section V concludes the research and provides recommendations for future development.

Research Method

This research adopts an experimental approach with the implementation of a smart home security system based on EfficientDet. The system architecture consists of three main components: a data acquisition unit, a processing unit, and a notification unit. The implementation uses the Raspberry Pi 4 Model B as the main processing unit, equipped with the NoIR v2 camera for night vision capabilities. Figure 3 shows a flowchart illustrating the research process. The process begins at the "Start" stage, which is then followed by the "Research Study." After conducting the research study, the next stage is "Preparation of tools and materials," which involves preparing all the necessities for the research. Next, the process continues to the stage of "Assembling the tool to make the tool," which is connected to a decision box for "Testing" with three criteria: electricity, distance, and stability. If the testing is unsuccessful, the process will return to the stage of assembling the tool. However, if the testing is successful, the process will proceed to the "Data & Analysis" stage. After the data is analysed, the process continues to the "Conclusion" stage, where the research results are summarized and evaluated. Finally, the research process concludes at the "Finished" stage, marking the end of the entire research sequence. The research methodology is illustrated in Figure 3, which presents the workflow from the initial research to the conclusion.

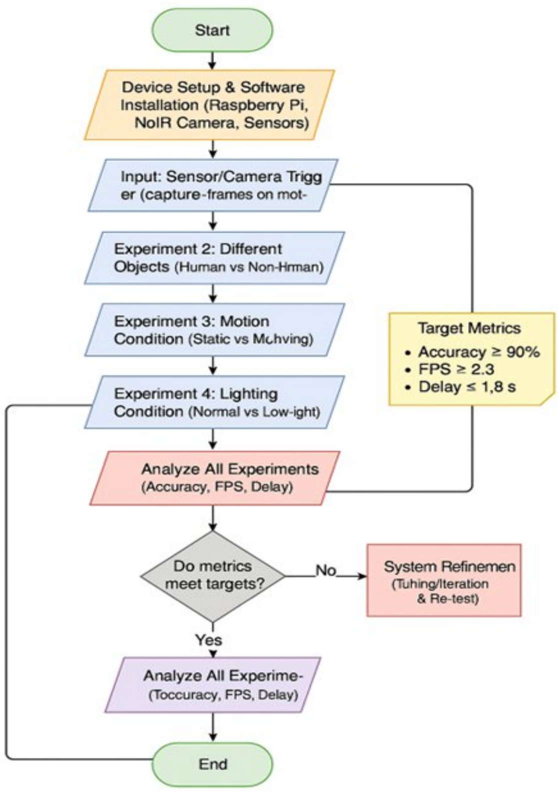

Figure 3 Research Flow Diagram

The entire process of testing a system based on a Raspberry Pi, NoIR camera, and sensors is shown in Figure 3. The project officially begins with the first stage, "Start." The system is then set up by installing the necessary sensors, NoIR camera, Raspberry Pi, and other hardware and software components. The calibration of the sensor and the calculation of model parameters, including detection thresholds, resolution, and frame rate, come next. When the system is ready, the sensors or camera will begin to take pictures or record video when they detect movement or the presence of objects in the observation area. Several experiments are then conducted using the data that was gathered.

The first experiment assesses how distance affects detection accuracy by testing the system's performance at three distinct distances: two, four, and six meters. The system's capacity to discriminate between human and non-human items is examined in the second experiment. By contrasting the outcomes for moving and stationary objects, the third experiment investigates the impact of motion circumstances. Under the fourth trial, performance is assessed under two distinct illumination scenarios: normal and dim. Measurements of accuracy, frames per second (FPS), and notification delivery latency are used to evaluate the results of these tests. The intended performance metrics a minimum accuracy of 90%, a minimum frame rate of 2.3 FPS, and a maximum delay of 1.8 seconds are contrasted with these outcomes. The procedure advances to the last phase, "End," which denotes the study's conclusion, if all goals are reached. Before repeating the evaluation process, the system goes through a refinement phase that includes modifications, retesting, and iteration to assure performance improvements if any parameter falls short of the goal. The experimental setup is depicted in Figure 4, showing the data flow from input to output.

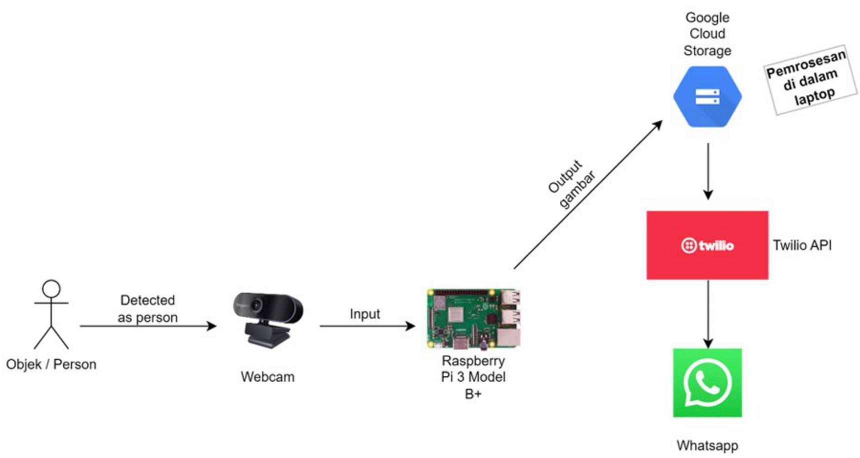

Figure 4 Schema Experiment

Figure 4 illustrates the workflow of the EfficientDet-based object detection system used in this study. The process starts with the capture of images by a webcam when there is an object or human in range. The image is sent to the Raspberry Pi 3 Model B+ which acts as the main processing unit. The Raspberry Pi processes the data to detect human presence, then sends the results to Google Cloud Storage for additional storage and processing on the laptop if needed. Furthermore, the detection results are distributed across various platforms, including YouTube for live video monitoring and WhatsApp for instant notifications using the Twilio API. The integration between hardware, cloud services, and communication platforms forms a home security system capable of real-time detection, storing data, delivering live video, and sending automated notifications to users. The scheme reflects the effective implementation of the concept of the Internet of Things (IoT), where physical devices such as webcams and Raspberry Pis relate to cloud-based services to deliver a unified monitoring solution that can be accessed from multiple platforms.

Result and Discussion

In this section, the researcher explains the results of the realization of the Object Detection System Based on Google Cloud Storage, which was created using components such as (Wu et al., 2020; Zhao et al., 2019). The system implementation included code development for parameter settings and object detection processes, as shown in Table 3 and Table 4. Table 3 lists the parameters used in configuring the object detection system during testing.

Table 3 Parameter Settings for Object Detection

Parameter	Value / Setting	Description
Detection Threshold	0.5	Minimum confidence score for detection
Image Resolution	640 × 480 px	Resolution used for camera input
Frame Rate	2.3 FPS	Average processing speed per frame
Model Used	EfficientDet-D0	Object detection algorithm
Processing Unit	Raspberry Pi 4 Model B	Main computation device
RAM Usage	~850 MB	Average memory consumption
Notification Method	WhatsApp via Twilio API	Real-time alert delivery
Lighting Condition	Normal & low light tested	Includes NoIR v2 night vision mode

The parameter configurations for the object detection system employed in this investigation are shown in Table 3. The system will only mark an object as detected if its confidence score is above 50% because the detection threshold is set at 0.5. At 640 × 480 pixels, the image resolution strikes a mix between a tolerable processing load and enough visual detail for real-time performance. The average frame rate, which shows how quickly the system processes incoming visual input, is 2.3 frames per second (FPS). EfficientDet-D0 is the selected detection algorithm because it is optimised for computational efficiency while retaining a respectable level of accuracy. The main computational device, a Raspberry Pi 4 Model B, handles processing and provides an affordable yet powerful platform for embedded machine vision applications.

The system uses about 850 MB of RAM while it is operating, which is indicative of the memory needs of the detection method and related operations. To ensure that users are promptly aware of detection events, the system is coupled with the Twilio API to give alerts via WhatsApp in real-time. To maintain performance in a variety of settings, testing is done in both normal and low-light lighting circumstances, utilising the camera module's NoIR v2 night vision functionality. To ensure that the system functions dependably within its specified constraints, these settings were carefully selected to optimise the trade-off between detection accuracy, processing speed, and hardware limitations. Table 4 presents detection results for various object types.

Table 4 Object Detection Results.

Object Type	Distance (m)	Accuracy (%)	FPS	Notes
Human	2	86	2.3	Stationary
Human	4	61	2.3	Stationary
Human	6	74	2.3	Stationary
Motorcycle	4	72	2.3	Non-human category
Chair	4	77	2.3	Non-human category
Bicycle	4	68	2.3	Non-human category
Gallon	4	59	2.3	Non-human category
Bag	4	59	2.3	Non-human category

The outcomes of the object detection tests are compiled in Table 4, which also describes the system's performance for different object types, distances, and categories. The maximum accuracy of 86% in human detection was attained at 2 meters. Accuracy decreased to 61% at 4 meters before somewhat increasing to 74% at 6 meters. The apparent size of the object in the frame, illumination, and image resolution are some of the variables that may affect this variation. To reduce motion blur and separate the impact of distance on accuracy, all human detection tests were conducted in a stationary environment. The system consistently performed computationally in every test, maintaining a steady processing rate of 2.3 FPS under all conditions.

The accuracy of detection for non-human items differed significantly depending on the type of object. Chairs had the highest score (77%), followed by motorcycles (72%), and bicycles (68%). Both gallon containers and bags had the lowest accuracy, 59%, which might be because of their similarity to backdrop objects or lack of distinguishing visual cues. To facilitate direct comparison across groups under uniform spatial settings, all non-human object tests were carried out at a fixed distance of 4 meters. Humans were generally more accurately spotted than non-human objects, especially at closer ranges, indicating that distinguishing characteristics, contrast, and object shape had a big impact on detection success. At this point in the study, the researchers have also conducted other assessments, such as testing the same object at different distances, comparing the performance of various object types at the same distance, evaluating stationary versus moving object detection, comparing human versus non-human detection, and timing and frequency of notifications. An example of chair detection is illustrated in Figure 5.

Figure 5 Experimental System Testing.

The experimental setup utilised in this study to test the object detection system is shown in Figure 5. A Raspberry Pi, which acts as the primary computing unit for the detection algorithm, is connected to a laptop as part of the setup. To take pictures and videos of the target object, the Raspberry Pi is connected to a NoIR camera module that is mounted on a stand. This picture simulates one of the detection test scenarios by using a stationary human person standing at a predetermined distance from the camera to represent the target object. The laptop is used to record performance indicators, monitor the system's output in real time, and store collected data for subsequent analysis. To allow for evaluation under actual operating conditions, the setup is placed outdoors to take advantage of natural lighting.

With this setup, test settings like object type, distance, and lighting may be changed while the equipment is positioned steadily for reliable data collection. By using a solid platform, alignment issues that can impair detection performance are avoided, and the camera is guaranteed to stay fixed throughout the experiment. The system may be successfully tested in both normal daytime and low-light conditions by placing it outside, especially when paired with the NoIR night vision feature. This physical setup serves as the basis for carrying out the sequence of tests outlined in the methodology, guaranteeing the consistency and dependability of the outcomes under various testing circumstances. Table 5 shows the detection accuracy for the same object tested at various distances.

Table 5 First test results – same object, different distance

Distance (meters)	Accuracy (%)	Frame per second
2	86	2.3
4	61	2.3
6	74	2.3
8	59	2.3
10	68	2.3

Accuracy was highest at 2 m (86%) and decreased with distance. Frame rate was constant at 2.3 FPS, indicating performance issues are due to recognition limitations, not processing speed. From Table 5, the highest accuracy occurs at 2 m (86%) and decreases significantly at 4–8 m. This decrease is due to reduced visual detail at the 640×480 px camera resolution. The increase back at 10 m is thought to be due to the position of the object being at the camera's optimal viewing angle. Recommendation Use a higher-resolution or super-resolution camera to maintain accuracy at >4 m. Figure 6 shows the First Test, which is a test with the same object at 4 meters. Here, the main goal is to see how accurate the system is at detecting identical objects at a certain distance, so the focus is on detection consistency.

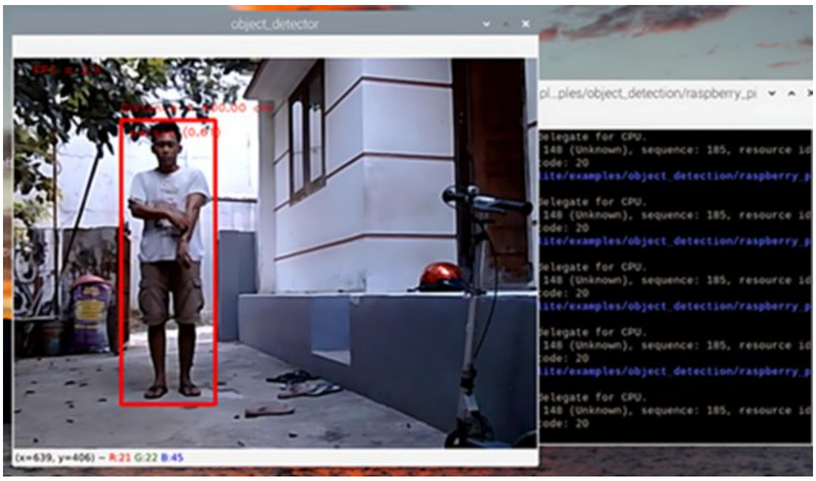

Figure 6 Objects equal 4 meters away (First Test)

The first test's outcome, when the identified object is placed precisely 4 meters from the camera, is displayed in Figure 6. A bounding box is formed around the human subject to show successful detection, and the image shows the object detection interface running on the Raspberry Pi. The system is very positive that the object falls into the "human" category, as indicated by the bounding box's 90.3% confidence score. To reduce angle distortion, the camera is aimed straight at the subject in an outdoor test setting with natural sunlight. The image's right side displays the Raspberry Pi's terminal output, which exhibits the real-time, continuous processing logs produced by the detection method.

To validate system performance within the specified operating range, it is essential to assess the accuracy of human detection at a modest distance. The selection of 4 meters enables evaluation of the model's ability to manage the subject's lower image resolution in comparison to closer ranges, as well as possible impacts from background environmental components. The EfficientDet-D0 model, in conjunction with the Raspberry Pi and NoIR camera, appears to be able to accurately identify human subjects in these circumstances, as seen by the steady frame rate and high detection confidence. As stated in the section on experimental results, the test's outcomes aid in the comparison of detection performance at different distances. Figure 7 shows the Second Test, which is a test with three different objects that are also placed at 4 meters. The purpose of this test is to compare the detection performance on different types of objects with the same distance conditions.

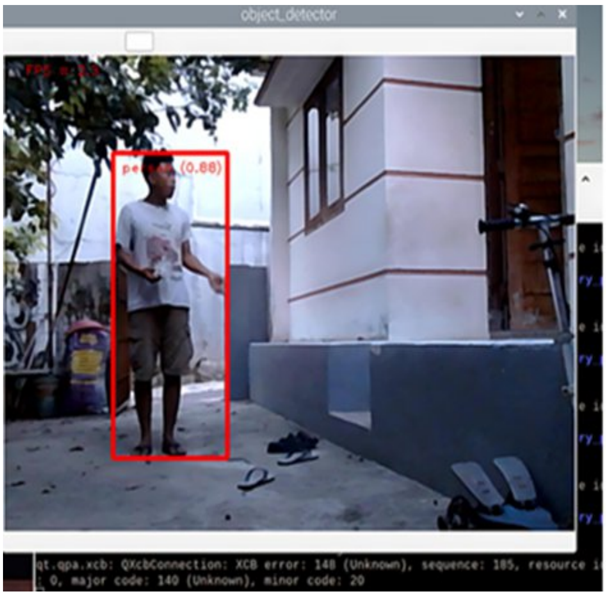

Figure 7 Object 3 equals 4 meters apart (Second Test)

The second test scenario, where the individual, designated as Object 3, is placed 4 meters away from the camera, is seen in Figure 7. The Raspberry Pi and NoIR camera-powered detection system successfully recognises the human subject by enclosing them in a red bounding box.

The algorithm's excellent classification certainty is demonstrated by the detection label "Person," which has an 88% confidence score. To guarantee comparable circumstances for performance comparison, the testing setting is set up outdoors with natural illumination and a steady camera position, much like the one shown in Figure 6. To evaluate the system's capacity to adjust to minute variations in appearance, the subject's posture and orientation are somewhat altered from the initial test.

By making minor changes in subject presentation while keeping the same distance as the first test, this test aids in the assessment of detection robustness. Factors like body angle, clothes, or environmental background variables may affect detection certainty, according to the recorded confidence score, which is somewhat lower than the 90.3% from the first test. The system's dependability under constant distance parameters is highlighted by the frame rate and detection accuracy remaining constant despite this fluctuation. This test's data, together with data from other distance and object-type trials, are used in a larger study to find trends and performance limits in object identification accuracy, which is crucial for confirming the system's operational efficacy. Table 6 reports detection accuracy for different objects at 4 m distance.

Table 6 Second test results

Distance (meters)	Accuracy (%)	Frame per second
2	86	2.3
4	61	2.3
6	74	2.3
8	59	2.3
10	68	2.3

The findings of the second test, which assessed the system's object identification accuracy at different distances while keeping the frame rate constant at 2.3 FPS, are shown in Table 6. The system's maximum accuracy of 86% was attained at a short range of 2 meters, suggesting that the best detection conditions occur when the subject is close to the camera. At 4 meters, accuracy drastically decreased to 61%, indicating that a greater distance would make it more difficult for the algorithm to detect distinctive features. It's interesting to note that accuracy increased to 74% at 6 meters. This could be because of contextual elements like backdrop contrast or lighting that momentarily improved detection ability at this distance.

Detection accuracy significantly declined beyond 6 meters, with 8 meters showing the lowest performance (59%). The difficulties of sustaining accurate detection as an object's size in the frame gets smaller and less detailed are reflected in this decline. The accuracy increased marginally to 68% at 10 meters, indicating that although detection is still feasible, it is less dependable than at closer ranges. The frame rate stayed constant at 2.3 FPS throughout all distances despite these precision variations, suggesting steady processing performance independent of range. These findings demonstrate how crucial it is to adjust system parameters for the desired operating distance to guarantee reliable detection performance. Motorcycle detection is illustrated in Figure 8. The system correctly classified the motorcycle as a non-human object with reliable accuracy.

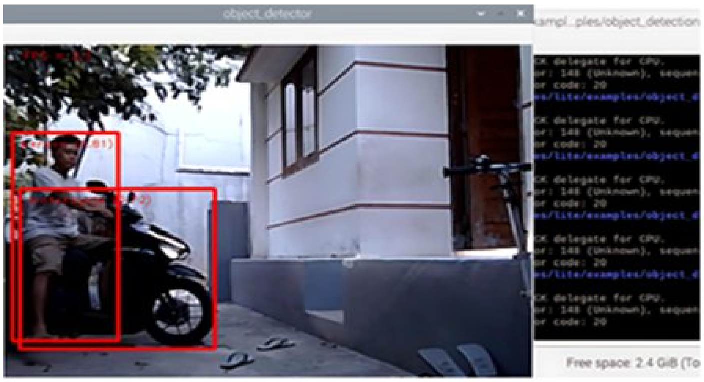

Figure 8 Objects of people with motors

When a motorbike and a person are both in the camera's range of vision, Figure 8 displays the detection output. The motorcycle and the human subject are correctly recognised by the system as two unique objects, each of which is surrounded by a different bounding box with a matching confidence score. The algorithm prioritises human identification, as evidenced by the motorcycle detection confidence score being slightly lower than the human detection confidence level of 87%. Applications involving mixed-object settings require the algorithm's capacity to discriminate between many object types in a single frame, which is demonstrated by this feature.

Under typical daylight conditions, the test situation shown in Figure 8 was carried out at a height of around 4 meters. The detection algorithm was able to retain proper categorisation without experiencing considerable misclassification, even in the case of overlapping items, where the motorcycle partially obscures the human. Real-time handling of such multi-object detection scenarios is ensured by the steady processing rate of 2.3 FPS. This outcome demonstrates how well the system can identify humans in addition to other non-human objects, which makes it appropriate for more intricate real-world settings where object occlusion and interaction are common occurrences. Table 7 compares detection accuracy for human and non-human objects tested simultaneously within the same frame.

Table 7 Human vs Non-Human Detection

Experiment	Object	Accuracy (people)	Accuracy (object)	FPS
1	People and motorcycles	81%	72%	2.3
2	People and seats	84%	77%	2.3
3	People and bicycles	88%	68%	2.3
4	People and gallons	70%	59%	2.3
5	People and bags	82%	59%	2.3

The findings of human versus non-human detection in five experimental scenarios each combining a person with a distinct object type are shown in Table 7. In the test involving people and bicycles, the system's human detection accuracy reached its maximum of 88%. This was closely followed by 84% for people and seats and 82% for people and bags. When detecting individuals next to gallon containers, the lowest person detection accuracy was 70%. This could be because the object and background have similar shapes, sizes, or colours, which could skew the algorithm's performance. The frame rate stayed steady at 2.3 FPS throughout all tests, suggesting consistent real-time processing power independent of object pairing.

The accuracy of detection varied more for non-human things. For non-human objects, chairs had the highest accuracy (77%), followed by motorbikes (72%), and bicycles (68%). At 59%, gallons and bags had the lowest identification rates, indicating that the detection model has a harder time identifying these items because of their less noticeable visual characteristics or merging with the surroundings. The findings suggest that although the system exhibits strong human identification in a range of scenarios, non-human object performance is more reliant on object properties like contrast, texture, and shape. This result emphasises how crucial it is to train object detection models using a variety of datasets to enhance the ability to recognise less recognisable non-human items. An example of human object detection is shown in Figure 9. Bounding boxes and labels confirm accurate identification with high confidence.

Figure 9 Human vs Non-Human Detection.

Figure 9 shows the system's ability to distinguish between human and non-human objects in a single frame. In this example, the system detects humans with large red bounding boxes that surround the entire body, while non-human objects which are likely to be part of the background or other elements are given a smaller green bounding box. The accuracy of human detection (70–88%) is consistently higher than that of non-humans (59–77%). This indicates that the dataset and model tuning are indeed focused on human detection, according to the main purpose of the system. These results prove that the EfficientDet algorithm used can identify humans as priority categories with high accuracy, while still recognizing the presence of other objects around them even though they are not the focus of the security system.

Capabilities like this are especially important in the context of home security, as the system can focus on human detection (as a potential threat) while still mapping other relevant objects. These findings are in line with the data in Table 7, which shows that the accuracy of human detection is consistently higher compared to non-human objects in the same scenario. Figure 10 illustrates detection results for moving and stationary human subjects. Stationary subjects are detected with higher accuracy, suggesting that motion introduces challenges such as motion blur and rapid posture changes.

Figure 10 Moving vs Stationary Human Detection

Figure 10 shows the system's ability to detect humans in stationary and moving conditions. In the example shown, the subject is standing still facing the camera, and the system provides a red bounding box that encircles the entire body with high precision. The test results show that detection accuracy for stationary humans tends to be higher than when humans are moving. This is in accordance with the data in Table 8, where the human object is stationary and has an accuracy of between 75–81%, while in the moving condition the accuracy drops to 65–74%. The accuracy of stationary objects (75–81%) is higher than that of moving objects (65–74%), most likely due to motion blur and rapid pose changes. A solution that can be applied is a motion deblurring algorithm or shutter speed adjustment.

The decrease in accuracy in moving conditions is likely due to motion blur or rapid pose changes, which make it more difficult for the algorithm to maintain detection accuracy. The system's ability to distinguish and still detect humans in these two conditions is important for home security applications, as threats can come from both fast-moving people and stationary people in surveillance areas. Table 8 summarizes detection accuracy for moving versus stationary human subjects. Accuracy for stationary objects (75–81%) exceeded that for moving objects (65–74%), suggesting that motion introduces recognition challenges, likely due to motion blur or rapid posture changes.

Table 8 Moving vs Stationary Objects

Experiment to-	Object Conditions	Accuracy (%)
1	Move	74
2	Photo	81
3	Move	66
4	Photo	75
5	Move	65
6	Photo	81

The comparison of the detection accuracy of moving and stationary (photo) items is shown in Table 8. According to the trials, stationary items routinely outperformed moving objects in terms of detection rates. For example, accuracies for stationary items were 81% in experiments 2 and 6 and 75% in experiment 4. This pattern suggests that the detection system can capture sharper images and more identifiable characteristics when motion blur is absent and object orientation is steady, which increases classification accuracy.

Moving objects, on the other hand, showed lower detection rates, with accuracies between 65% and 74%. Experiment 1 had the highest moving object accuracy of 74%, while Experiment 5 had the lowest at 65%. Motion blur shifts in object posture or location during movement, and potential partial occlusions can all be blamed for this performance decline, which could make it harder for the algorithm to reliably recognise important features. These results imply that although the system works well for detecting stationary objects, motion handling enhancements like greater frame rates or motion-compensating algorithms would be helpful for improving the system's effectiveness for detecting moving objects. According to (Rogers, 2014) and (Stringer), Twilio API integration makes it possible to send real-time notifications to messaging apps like WhatsApp. An example of this implementation is shown in Figure 11, where the home security system sends out a notification upon detecting movement.

Figure 11 WhatsApp Notification via Twilio

Figure 11 shows an example of a real-time notification sent by a home security system through the WhatsApp app by leveraging the Twilio API integration. On this display, every time the system detects movement, the message "Movement detected!" automatically appears accompanied by a screenshot of the detected results. Each detection image features a red or green bounding box that surrounds the identified object, complete with classification labels and accuracy values. This delivery process is carried out as soon as the system recognizes the object, so that users can immediately know the existence of potential threats or suspicious activity in the supervised area.

This capability is an important part of a home security system because it allows users to respond quickly to detected situations, even if they are far away from the location. Notifications can still be received if the device is connected to the internet, and the presence of the detected photos makes it easy for users to recognize the situation without the need to access CCTV directly. Based on the test results shown in Table 9, the average notification delivery time lag ranges from 1.4 to 1.8 seconds after detection occurs, with a sending frequency of approximately every three seconds, depending on the network conditions used. Table 9 summarizes notification frequency and delay.

Table 9 Notification Time and Frequency

Experiment to-	Time	Number of Messages	Delay
1	1 minute	20 messages	1.8 seconds
2	45 seconds	15 messages	1.5 seconds
3	30 seconds	10 messages	1.4 seconds

The notification time and frequency findings are shown in Table 9, which shows how rapidly and reliably the system can send warnings over the chosen messaging platform. The system successfully sent 20 messages with an average delay of 1.8 seconds between detection and notification delivery in the first experiment, where the notification interval was set to one minute. The processing overhead of combining the alert message with detection data and network transmission latency is to blame for this minor delay. Even with comparatively high message counts, the system maintains steady performance, as evidenced by the consistency of message delivery at this interval.

The average latency improved to 1.5 and 1.4 seconds in studies with shorter notification intervals, such as 45 and 30 seconds, but the number of messages dropped to 15 and 10, respectively. This suggests that lowering the frequency of notifications leads to a minor reduction in processing and network burden, which speeds up delivery times. The outcomes also show that the system may function effectively with different alert intervals, keeping delays far below the 1.8-second performance goal. For real-time monitoring applications, where prompt alarms can directly affect the system's efficacy in real-world situations, this responsiveness is essential.

There are numerous chances for additional development to improve the system's adaptability and dependability in practical settings, even though it has shown consistent performance and excellent accuracy for human detection. Using infrared (IR) illumination modules is one important way to improve detection accuracy in low-light conditions. Studies on real-time infrared detection have demonstrated that IR technology greatly increases the resilience of object identification in low illumination (Gundogan et al., 2023). Applying edge computing is another improvement that lessens reliance on cloud services, which lowers latency, conserves bandwidth, and improves privacy by processing data closer to its source. Furthermore, as the system uses the WhatsApp API to provide notifications, data communications security is essential. According to (Kumar et al., 2019), contemporary protocols like JEDI offer robust encryption without necessitating direct contact between recipients. Lastly, to guarantee continuous operation during power failures, power management can be optimised by adding a mini-UPS or battery storage module. For edge deployments that are vulnerable to abrupt disconnections or voltage fluctuations, a dependable power infrastructure is especially crucial. Technically and operationally, the home security system can be made more robust, responsive, and secure by using these tactics.

Conclusions

The research successfully developed a smart home security system that integrates the EfficientDet algorithm with Raspberry Pi for accurate, real-time object detection and instant notifications. Experimental results confirmed high detection accuracy, particularly for human objects at close range, with stable processing performance across various conditions. The integration with WhatsApp via Twilio API proved effective in delivering timely alerts to users. With stable performance at 2.3 FPS and high accuracy for human detection, the system is worthy of use in the modern home. However, improvements in lighting, data security, and detection distance optimization will make it more resilient in real conditions. Future work will focus on enhancing detection robustness under varying environmental conditions, implementing adaptive learning to improve accuracy over time, strengthening cybersecurity measures, optimizing energy consumption for continuous operation, and exploring edge computing approaches to further reduce latency.

Acknowledgements

This research was supported and financed by the Independent Grant Research Program by LPPM-Telkom University Campus Jakarta). We would like to thank colleagues at Telkom University Jakarta who have provided insight and expertise that greatly assisted this research.

References

Alqahtani, D. K., Cheema, M. A., & Toosi, A. N. (2024). Benchmarking deep learning models for object detection on edge computing devices. International Conference on Service-Oriented Computing,

Buongiorno, D., Caramia, D., Di Ruscio, L., Longo, N., Panicucci, S., Di Stefano, G., Bevilacqua, V., & Brunetti, A. (2022). Object detection for industrial applications: Training strategies for ai-based depalletizer. Applied Sciences, 12(22), 11581. https://doi.org/10.3390/app122211581

Constantin, I., & Dinu, A. (2023). Optimized security system with real-time operation. 2023 17th International Conference on Engineering of Modern Electric Systems (EMES),

Gundogan, M. M., Aksoy, T., Temizel, A., & Halici, U. (2023). Ir reasoner: Real-time infrared object detection by visual reasoning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,

Hema, N., & Yadav, J. (2020). Secure home entry using raspberry Pi with notification via telegram. 2020 6th International Conference on Signal Processing and Communication (ICSC),

Irjanto, N. S., & Surantha, N. (2020). Home security system with face recognition based on convolutional neural network. International Journal of Advanced Computer Science and Applications, 11(11). https://doi.org/10.14569/IJACSA.2020.0111152

Kumar, S., Hu, Y., Andersen, M. P., Popa, R. A., & Culler, D. E. (2019). {JEDI}:{Many-to-Many}{End-to-End} encryption and key delegation for {IoT}. 28th USENIX security symposium (USENIX Security 19),

Liu, Z., Chen, C., Huang, Z., Chang, Y. C., Liu, L., & Pei, Q. (2024). A Low-Cost and Lightweight Real-Time Object-Detection Method Based on UAV Remote Sensing in Transportation Systems. Remote Sensing, 16(19), 3712. https://doi.org/10.3390/rs16193712

Pane, M. A. S., Saleh, K., Prayogi, A., Dian, R., Siregar, R. M., & Sugianto, R. A. (2024). Low-Cost CCTV for Home Security With Face Detection Base on IoT. Journal of Information Systems and Technology Research, 3(1), 20-29. https://doi.org/10.55537/jistr.v3i1.769

Prahastiwi, D., Andianti, R., Dewi, R., & Sari, N. (2023). Statistik Kriminal 2023. Badan Pusat Statistik.

Redmon, J. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

Rogers, T. (2014). Twilio Best Practices. Packt Publishing Ltd.

Sahana, V., Shashidhar, R., Bindushree, R., & Chandana, A. (2023). IOT Based Burglar Detection and Alarming System Using Raspberry Pi. https://doi.org/10.5815/ijem.2023.06.03

Tan, M., Pang, R., & Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition,

Wirabudi, A. A., & Fachrurrozi, N. R. (2023). Classification of tea plantation using orthomosaics stitching maps from aerial images based on CNN. Jurnal Infotel, 15(1), 82-88. https://doi.org/10.20895/infotel.v15i1.871

Wirabudi, A. A., Fachrurrozi, N. R., Dorand, P., & Royhan, M. (2024). Enhancement of images compression using channel attention and post-filtering based on deep autoencoder. International Journal of Advances in Intelligent Informatics, 10(3). https://doi.org/10.26555/ijain.v10i3.1499

Wu, X., Sahoo, D., & Hoi, S. C. (2020). Recent advances in deep learning for object detection. Neurocomputing, 396, 39-64. https://doi.org/10.48550/arXiv.1908.03673

Yeh, M.-T., Tsai, Y.-C., Cheng, C.-H., Chung, Y.-N., & Lu, P.-S. (2023). Smart Home Surveillance System Based on the Optimized EfficientDet Network. Sensors & Materials, 35. https://doi.org/10.18494/SAM4297

Zhang, H., Liu, R., Kaushik, A., & Gao, X. (2023). Satellite edge computing with collaborative computation offloading: An intelligent deep deterministic policy gradient approach. IEEE Internet of Things Journal, 10(10), 9092-9107. https://doi.org/10.1109/JIOT.2022.3233383

Zhao, Z.-Q., Zheng, P., Xu, S.-t., & Wu, X. (2019). Object detection with deep learning: A review. IEEE transactions on neural networks and learning systems, 30(11), 3212-3232. https://doi.org/10.1109/TNNLS.2018.2876865

Zou, Z., Chen, K., Shi, Z., Guo, Y., & Ye, J. (2023). Object detection in 20 years: A survey. Proceedings of the IEEE, 111(3), 257-276. https://doi.org/10.1109/JPROC.2023.3238524