Technologia Journal: Jurnal Informatika
E-ISSN : 3046-9163
Vol. 3 No. 3, August 2026
DOI: https://doi.org10.62872/5caybe23

An Integrated PIR-Camera IoT Security System with Real-Time Telegram Notification

Imam Tri Suryadin[1]✉, Aang Anwarudin[2], Lazuardi Fatahilah Hamdi[3], Riski Yudhi Prasongko[4]

[1,2,3,4]Universitas Muhammadiyah Gombong, Indonesia
e-mail: * imam.ts@gmail.com

ABSTRACT

The rapid growth of Internet of Things (IoT) technology has motivated numerous studies to develop motion-based security systems that combine Passive Infrared (PIR) sensors with microcontrollers and instant-messaging platforms such as Telegram. A review of four recent studies on PIR-based IoT motion detection, however, shows that each system still carries partial limitations: some provide visual verification through a camera module but lack an offline data-retention mechanism, while others record activity to a cloud platform but do not capture images, leaving alerts without visual evidence. This study aims to synthesize the strengths and shortcomings of the four reviewed systems through a comparative literature analysis and to propose a refined system architecture that consolidates their complementary features while resolving their respective gaps. The research applies a descriptive-qualitative literature review combined with a design-science approach to formulate an enhanced architecture consisting of a PIR sensor, an ESP32-CAM module, a Telegram Bot API for dual text-and-image alerts, cloud-based data logging on ThingSpeak, and a local offline-backup mechanism using SPIFFS, supplemented with an adaptive debounce filter to reduce false triggers. The outcome of this study is a conceptual architecture, block diagram, and operational flowchart that integrate visual verification, persistent logging, and connectivity resilience within a single design, providing a more complete reference for future implementation of IoT-based motion detection security systems.

Keywords: *ESP32; Internet of Things; Motion Detection; PIR Sensor; Telegram Notification.*

INTRODUCTION

Security concerns in both residential and small commercial environments have increased alongside the rise of unauthorized-entry incidents that often occur without direct human supervision (Suherman et al., 2025). Conventional passive surveillance tools such as closed-circuit television record activity but do not actively alert the owner in real time, so a delay often occurs between an intrusion event and the moment it is noticed (Wang & Jiang, 2019). This condition has directed attention toward Internet of Things (IoT) based motion detection, in which physical devices such as motion sensors, microcontrollers, and cameras are interconnected through a network to sense, process, and communicate events automatically (Sethi & Sarangi, 2017).

Among available motion sensors, the Passive Infrared (PIR) sensor is widely adopted because it detects a moving heat source at low cost and low power consumption while avoiding the privacy concerns associated with continuous camera recording (Shokrollahi et al., 2024). When paired with a Wi-Fi-enabled microcontroller such as the ESP32, a PIR sensor can trigger an automatic alert to a mobile messaging application, most commonly Telegram, within seconds of detecting movement (Okokpujie et al., 2023). This combination has been explored in several recent studies with different emphases: automatic room-lighting control (Husna et al., 2019), home-intrusion detection with image capture (Susatyono et al., 2025), early-warning alarms for a small retail jewelry store (Suherman et al., 2025), and motion logging with offline resilience (Z et al., 2026).

A closer comparison of these four studies, however, reveals that no single design combines all of the desirable capabilities identified across the literature. Systems equipped with a camera for visual confirmation, such as those reported by Husna et al. (2019) and Susatyono et al. (2025), do not provide a mechanism to preserve detection data when the network connection is interrupted. Conversely, systems that already include offline data retention and cloud-based logging, such as the one developed by Z et al. (2026), do not capture any image and therefore cannot visually confirm whether a detected trigger corresponds to an actual intrusion or to irrelevant movement such as a pet or a swaying curtain. The system proposed by Suherman et al. (2025) demonstrates a stable response time of about three seconds but likewise offers neither a camera nor a persistent monitoring platform. This uneven distribution of features across the reviewed studies constitutes the research gap addressed in the present study.

The purpose of this study is to synthesize the comparative findings of the four reviewed systems and to formulate an integrated PIR-camera IoT security architecture that unifies visual verification, persistent cloud logging, offline resilience, and false-trigger mitigation within one coherent design. The novelty of this study lies not in re-testing a single hardware prototype, but in producing a consolidated design synthesis, supported by a structured comparative analysis, block diagram, and operational flowchart, that can serve as a refined reference architecture for subsequent implementation and empirical validation.

METHODOLOGY

This study employs a descriptive-qualitative literature review combined with a design-science research approach. The design of the research was chosen because the objective is not to build and test a new physical prototype, but to critically compare existing validated systems and derive an improved conceptual design from the pattern of strengths and weaknesses found among them.

The primary data sources consist of four peer-reviewed articles published between 2019 and 2026 that specifically implement a PIR sensor integrated with

a Wi-Fi microcontroller and Telegram-based notification for either lighting automation, home security, or retail security purposes: Husna et al. (2019), Susatyono et al. (2025), Suherman et al. (2025), and Z et al. (2026). These four studies were selected because they share the same core sensing principle (PIR-based motion detection) and the same notification channel (Telegram), which allows a direct feature-by-feature comparison while still exhibiting meaningful differences in supporting components such as camera modules, cloud logging platforms, and offline-handling mechanisms.

Data collection was carried out through a structured content analysis of each article, extracting information on the hardware and software components used, the notification and logging mechanisms implemented, the reported performance metrics (such as response time, detection accuracy, and network quality), and the limitations acknowledged by the respective authors. The extracted information was organized into a comparative matrix (Table 1) to visualize which functional capabilities are present or absent in each system.

The analysis technique applied is a gap analysis, in which the comparative matrix is used to identify functional capabilities that appear in some systems but are missing in others. The identified gaps then become the basis for the design-science stage, in which a refined system architecture is proposed by combining the complementary strengths of the reviewed systems and adding a new functional element, the adaptive debounce filter, to address the false-trigger issue observed in two of the reviewed studies. The proposed architecture is presented through a block diagram and an operational flowchart to describe both the physical component relationships and the logical decision flow of the enhanced system.

RESULTS AND DISCUSSION

A. Comparative Analysis of Existing PIR-Based IoT Motion Detection Systems

Table 1 summarizes the comparison of the four reviewed studies across eight aspects: the sensor and controller configuration, the presence of a camera for visual verification, the notification channel, the availability of cloud-based data logging, the presence of an offline-backup mechanism, the reported performance, and the main limitation acknowledged in each study.

Table 1
Comparative Matrix of Four PIR-Based IoT Motion Detection Studies

Study	Sensor / Controller	Camera	Notification	Cloud Logging	Offline Backup	Reported Performance	Main Limitation
Husna et al. (2019)	PIR, LDR; Arduino Uno + NodeMCU	Yes (VC0706)	Telegram	None	None	Throughput 1,162.45 bytes/s; delay 0.941 s; 0% packet loss (QoS)	QoS-only focus; no persistent log; two controllers add complexity

Study	Sensor / Controller	Camera	Notification	Cloud Logging	Offline Backup	Reported Performance	Main Limitation
Susatyono et al. (2025)	PIR + ESP32-CAM	Yes (ESP32-CAM)	Telegram + mobile app (FCM)	Firebase Storage	None	93% accuracy; notification delay 2–4 s	No data retention during Wi-Fi outage; occasional false trigger
Suherman et al. (2025)	PIR HC-SR501 + ESP32	No	Telegram	None	None	Response time 3.08–3.64 s over 1–3 m	No visual evidence; no logging platform; single-case study
Z, Purwadi & Maesaroh (2026)	PIR HC-SR501 + ESP32	No	Telegram	ThingSpeak	SPIFFS	90–100% accuracy, 50 cm–3 m; misses fast objects at 2 m	No image capture; no filter for non-human triggers

Source: synthesized by the author from Husna et al. (2019), Susatyono et al. (2025), Suherman et al. (2025), and Z et al. (2026).

B. Identified Gaps and Limitations

The comparative matrix shows a clear complementary pattern rather than a single dominant weakness. The studies by Husna et al. (2019) and Susatyono et al. (2025) both integrate a camera module, allowing visual confirmation of a detected event; however, neither study provides a mechanism to retain detection data locally when the wireless connection is temporarily lost, which means an intrusion event occurring during a network outage may go entirely unrecorded. Susatyono et al. (2025) further reported that a static object occasionally triggered a false detection, indicating that the PIR sensor alone, without an additional filtering stage, cannot fully distinguish between relevant human movement and irrelevant environmental change.

In contrast, the system developed by Z et al. (2026) already resolves the connectivity problem through a SPIFFS-based offline backup and provides continuous cloud logging on ThingSpeak, yet it does not include any camera, so every alert delivered to the user consists of text only, without visual evidence of what caused the trigger. The same study also reported a reduced detection accuracy of 90% at a two-meter distance when the moving object traveled quickly, suggesting that response latency, rather than sensor range, becomes the limiting factor for fast-moving subjects. The system proposed by Suherman et al. (2025) achieved a stable response time of 3.08 to 3.64 seconds across a one-to-three-meter range, which is well suited for an early-warning application in a retail environment, but the study explicitly identifies the absence of a camera module and of any persistent data-logging platform as directions for future development.

Taken together, these findings indicate that no individual system among the four reviewed studies simultaneously provides visual verification, persistent cloud logging, offline resilience, and false-trigger mitigation. This distributed pattern of strengths forms the basis for the integrated architecture proposed in the following subsection.

C. Proposed Integrated System Architecture

Based on the gap analysis above, this study proposes an integrated architecture that combines an ESP32-CAM module for image capture (adopted from Husna et al., 2019 and Susatyono et al., 2025), a dual notification and logging mechanism using Telegram Bot API and ThingSpeak (adopted from Z et al., 2026), a SPIFFS-based local offline backup for connectivity resilience (also adopted from Z et al., 2026), and a newly introduced debounce/interval filtering stage placed between motion detection and image capture to reduce the false-trigger issue reported by Susatyono et al. (2025). Figure 1 illustrates the block diagram of the proposed architecture.

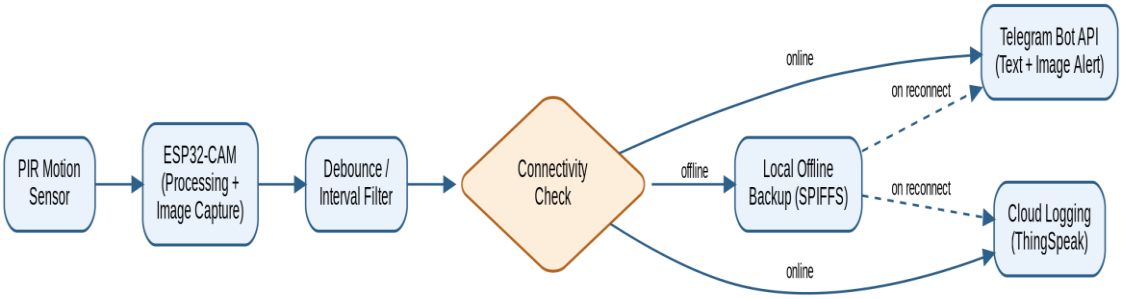

Figure 1. Block Diagram of the Proposed Integrated PIR-Camera IoT Security System

In this architecture, the PIR sensor continuously monitors the surrounding area, and every detected motion event is first processed by the ESP32-CAM module before being passed through the debounce/interval filter. A detection that passes the filter proceeds to a connectivity check: when the device is online, the system simultaneously sends a Telegram alert containing both a text message and a captured image and writes the event data to ThingSpeak for long-term monitoring; when the device is offline, the event is instead written to local SPIFFS storage and automatically synchronized to Telegram and ThingSpeak once the connection is restored.

D. Proposed Operational Flow

Figure 2 presents the operational flowchart derived from the block diagram, describing the sequence of decisions executed by the microcontroller from system initialization to notification delivery.

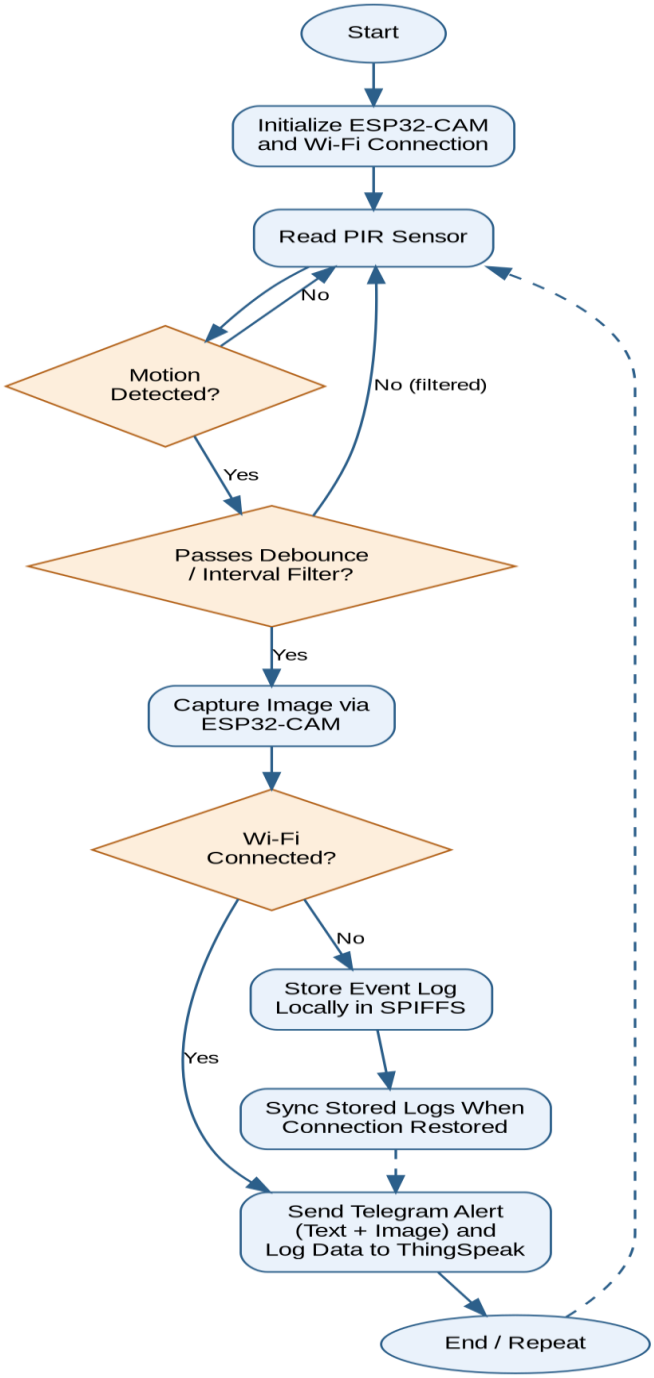

Figure 2. Operational Flowchart of the Proposed System

The flowchart begins with the initialization of the ESP32-CAM module and the Wi-Fi connection, followed by continuous reading of the PIR sensor. When motion is detected, the debounce/interval filter evaluates whether the trigger falls within a minimum time window since the previous detection or coincides with a known static-object pattern; a trigger that does not pass this filter is discarded, and the system returns to sensor reading without generating an alert. A trigger that passes the filter proceeds to image capture, after which the system checks Wi-Fi availability: if the connection is active, the system sends the

Telegram alert and logs the event to ThingSpeak; if the connection is unavailable, the event is stored in SPIFFS and queued for synchronization once connectivity is restored.

E. Discussion of Expected Contribution

The proposed architecture is expected to address the limitations identified in Subsection B in three specific ways. First, the addition of the ESP32-CAM module to a system that previously relied only on text-based Telegram alerts, as in Suherman et al. (2025) and Z et al. (2026), provides the visual verification capability that both studies identified as a direction for future work. Second, the incorporation of SPIFFS offline backup and ThingSpeak logging into a camera-equipped design, as in Husna et al. (2019) and Susatyono et al. (2025), closes the connectivity gap that could otherwise cause detection data to be permanently lost during a network outage. Third, the debounce/interval filtering stage is proposed specifically in response to the false-trigger behavior observed by Susatyono et al. (2025), and its logic is informed by the fast-motion detection gap reported by Z et al. (2026), so that both static-object interference and rapid movement can be handled through an adjustable time-window parameter rather than through hardware modification alone.

It should be noted that the outcome of this study is a validated conceptual synthesis rather than a newly measured hardware prototype; the performance figures cited in Table 1 are those reported in the respective original studies. Consequently, empirical testing of the proposed integrated architecture, particularly regarding the added latency introduced by simultaneous image capture, cloud logging, and offline synchronization, remains an important step for future research before the design can be considered fully validated for field deployment.

CONCLUSION

This study compared four PIR-based IoT motion detection systems and found that their capabilities are complementary rather than overlapping: two systems provide visual verification through a camera but lack offline data resilience, while the other two provide offline backup and persistent cloud logging but lack any camera, and one system exhibits an unresolved false-trigger issue. Based on this gap analysis, the study proposes an integrated architecture that combines a PIR sensor, an ESP32-CAM module, a debounce/interval filter, dual Telegram text-and-image notification, ThingSpeak cloud logging, and SPIFFS-based offline backup within a single coherent design. The proposed block diagram and flowchart demonstrate how these previously separate features can be unified to overcome the individual shortcomings identified across the reviewed literature, providing a more complete reference architecture for the future development and empirical testing of IoT-based motion detection security systems.

REFERENCES

Al-Turjman, F., & Alturjman, S. (2018). Context-sensitive access control for IoT-cloud convergence: A survey. IEEE Internet of Things Journal, 5(5), 3520–3535.

Akhund, T. M. N. U., & Teramoto, K. (2025). Privacy-concerned averaged human activeness monitoring and normal pattern recognizing with single passive infrared sensor using one-dimensional modeling. Sensors International, 6, 100303.

Firdaus, M. A., & Budhisantosa, N. (2025). Rancang bangun sistem pemantauan real-time ruang server dengan integrasi Telegram Bot dan ESP32. Jurnal Pendidikan Tambusai, 9(1), 1–9.

Hercog, D., Lerher, T., Truntič, M., & Težak, O. (2023). Design and implementation of ESP32-based IoT devices. Sensors, 23(15), 6739.

Husna, A., Hidayat, H. T., & Mursyidah. (2019). Penerapan IoT pada sistem otomatisasi lampu penerangan ruangan dengan sensor gerak dan sensor cahaya menggunakan Android. Jurnal Teknologi Rekayasa Informasi dan Komputer, 3(1), 10–16.

Kavitha, G., & Sundaram, S. (2021). IoT-based intrusion detection and alert system using PIR sensor and cloud computing. Journal of Ambient Intelligence and Humanized Computing, 12(5), 5413–5423.

Khan, R., Khan, S. U., Zaheer, R., & Khan, S. (2012). Future internet: The Internet of Things architecture, possible applications and key challenges. In Proceedings of the 10th International Conference on Frontiers of Information Technology (pp. 257–260).

Okokpujie, K., Okokpujie, I. P., Young, F. T., & Subair, R. E. (2023). Development of an affordable real-time IoT-based surveillance system using ESP32 and TWILIO API. International Journal of Safety and Security Engineering, 13(6), 1069–1075.

Serepas, F., Papias, I., Christakis, K., Dimitropoulos, N., & Marinakis, V. (2025). Lightweight embedded IoT gateway for smart homes based on an ESP32 microcontroller. Computers, 14(9), 391.

Sethi, P., & Sarangi, S. R. (2017). Internet of Things: Architectures, protocols, and applications. Journal of Electrical and Computer Engineering, 2017, Article 9324035.

Shokrollahi, A., Persson, J. A., Malekian, R., Sarkheyli-Hägele, A., & Karlsson, F. (2024). Passive infrared sensor-based occupancy monitoring in smart buildings: A review of methodologies and machine learning approaches. Sensors, 24(5), 1533.

Suherman, Hermansyah, & Syahpita, J. (2025). Sistem alarm deteksi gerak berbasis IoT menggunakan sensor PIR dan ESP32 dengan notifikasi Telegram real-time. Jurnal Komputer Teknologi Informasi Sistem Komputer (JUKTISI), 4(2), 1469–1476.

Susatyono, J. D., Febryantahanuji, & Kuncoro, A. A. (2025). Pengembangan sistem keamanan rumah berbasis IoT dengan deteksi intrusi real-time menggunakan sensor PIR dan kamera, serta notifikasi otomatis melalui aplikasi mobile. Jurnal Publikasi Ilmu Komputer dan Multimedia, 4(2), 30–41.