ESP32-CAM-based Smart Surveillance: Real-time Transmission and Alerts Via Telegram

Vigilancia Inteligente basada en ESP32-CAM: Transmisión en Tiempo Real y Alertas mediante Telegram

Fernando Esparza-Segundo, Shany-Lizbeth Enríquez-Arcadio*
Juan Carlos Gonzalez-Islas, Alberto Suarez-Navarrete, Ma. de Jesus Gutiérrez-Sanchez

Instituto de Ciencias Básicas e Ingeniería, Universidad Autónoma del Estado de Hidalgo
Carretera Pachuca-Tulancingo km. 4.5, Col. Carboneras, 42184, Mineral de la Reforma, Hidalgo, MÉXICO
*en440970@uaeh.edu.mx

KEYWORDS:

Remote video surveillance, Telegram integration, Image compression, ESP32-CAM, App Inventor 2

ABSTRACT

Currently, mobile video surveillance systems represent a fundamental role in security and remote monitoring, allowing continuous and real-time access from any location. This work presents the implementation of a video surveillance system managed via the Telegram messaging application, using the ESP32 WiFi CAM platform. Users can view images and videos through a button integrated into the community group of the application, providing interaction and visualization from a smartphone. An image compression system has been implemented to optimize the sending and receiving times of the images captured by the ESP32-CAM prototype. In addition, integration with other messaging platforms and incorporation of a module developed with App Inventor 2 is planned. The results of the proof-of-concept demonstrate the efficiency and functionality of the video surveillance system, offering users an agile and adaptable solution to diverse technological needs, with an affordable cost and free access.

PALABRAS CLAVE:

Video vigilancia remota, Integración de Telegram, Compresión de datos, ESP32-CAM, App Inventor 2

RESUMEN

Actualmente, los sistemas de videovigilancia móvil representan un papel fundamental en la seguridad y monitorización remota, permitiendo el acceso continuo y en tiempo real desde cualquier ubicación. Este trabajo presenta la implementación de un sistema de videovigilancia gestionado a través de la aplicación de mensajería Telegram, utilizando la plataforma ESP32 WiFi CAM. Los usuarios pueden visualizar imágenes y vídeos a través de un botón integrado en el grupo de comunidad de la aplicación, proporcionando interacción y visualización desde un smartphone. Se ha implementado un sistema de compresión de imágenes para optimizar los tiempos de envío y recepción de las imágenes captadas por el prototipo ESP32-CAM. Además, está prevista la integración con otras plataformas de mensajería y la incorporación de un módulo desarrollado con App Inventor 2. Los resultados de la prueba de concepto demuestran la eficiencia y funcionalidad del sistema de videovigilancia, ofreciendo a los usuarios una solución ágil y adaptable a diversas necesidades tecnológicas, con un coste asequible y de libre acceso.

• **Recibido: 17 de octubre de 2024** • **Aceptado: 10 de abril de 2025** • **Publicado en línea: 17 de junio de 2025**

DOI: 10.30973/progmat/2025.17.2/6

1. INTRODUCTION

In recent years, security has become an important aspect for both residential communities and commercial and industrial environments. Continuous technological evolution has led to the development of increasingly advanced video surveillance systems, capable of providing effective protection adaptable to a variety of needs [1]. The integration of artificial intelligence (AI) has been key in this evolution, allowing video surveillance systems to detect behavioral patterns and anomalies in real time, improving accuracy, and reducing false alarms [2].

However, despite technological advances and the various available in the market surveillance systems with motion detection (video) or person detection (image), the installation of these surveillance systems is extremely costly and traditional surveillance cameras can only record live video [3]. Therefore, accessible and functional solutions have been sought, such as [4], which presents a security camera system consisting of three main parts. ESPCAM-32 as camera, ESP32 microcontroller as controller, and Telegram BOT as interface.

Using the same principles, [5] presents a security device controlled by ESP32, where the PIR sensor functions as a fire sensor and sends similar alerts through Telegram. In the aforementioned cases, in addition to the advantage of being cost-effective, the rapid sending of information is of vital importance. Some related works have been developed in this regard; for example, [6] describes a home surveillance system with real-time monitoring capabilities, intrusion detection, and evidence collection through a mobile application. Unlike the described system, which uses an Android application and Firebase for management and monitoring, the current proposal incorporates a real-time communication platform through Telegram, along with advanced image compression techniques to optimize data transmission. This adaptation improves efficiency in communication and remote visualization, offering a more agile and flexible surveillance system.

The system proposed in this work improves traditional surveillance systems by using Telegram and image compression, providing an affordable and easy-to-implement solution, even for those without advanced technical knowledge. This eliminates the need for specific applications when using App Inventor 2, representing an innovative approach to pre-prototyping. The main code blocks are presented for creating an application to enable image transmission from the ESP32-CAM. In addition, the MIT App Inventor 2 platform allows one to customize the user experience according to different needs, being free and open access. This visual development environment aims to democratize the creation of mobile applications, allowing anyone, regardless of their level of programming skills, to design and build their own applications for Android devices.

Unlike other systems, such as conventional IP surveillance cameras that require specific applications and complicated configurations, our solution is easy to implement. For example, many security cameras require an installation process that can be complicated and costly, as well as monthly subscriptions to cloud storage. In contrast, by using Telegram as a communication platform, anyone with a smartphone can access the surveillance system without hassle and at no additional costs. The combination of ESP32-CAM and Telegram makes it possible to create a practical and adaptable security system without requiring advanced technical knowledge.

Siddineni et al. [7], present the arrangement and the implementation of a smart security system for homes wherein a person will be able to monitoring on Blynk through their smartphone. The ESP32-CAM controller unit is equipped at remote end device end. The use of ESP32-CAM allows a real time and low-cost application. Also, a PIR sensor is used and if some motion is found, it will send a notification through the IFTTT and telegram platform. A key point that compared to Arduino-integrated solutions for robotic surveillance, the use of the ESP32-CAM has significant advantages [8]. The ESP32-CAM microcontroller combines a camera module and Wi-Fi connectivity in a single compact device, eliminating the need for an additional microcontroller and simplifying system design.

These developments not only improve the flexibility and adaptability of the surveillance system, but also address the need for more robust solutions in various operational contexts. This approach not only reduces costs and the physical space required, but also improves data transmission efficiency by directly integrating image processing and connectivity on a single chip, offering a more robust and higher-performance solution for surveillance applications. A crucial problem in related work is real-time communication within video surveillance systems, which is still quite prominent, mainly due to the slowness in the transmission of images and videos and the poor integration of additional functionalities, such as the handling of image compression [9]. The main technical challenges include the need to accelerate data transmission via Telegram to ensure seamless communication, as well as the development of an application in App Inventor 2 that facilitates the effective viewing and management of multimedia content. The proposed solution addresses these problems by adopting advanced techniques for image compression and an optimized integration between Telegram and App Inventor 2, with the aim of improving the efficiency and flexibility of the video surveillance system.

To address these challenges, this work focuses on the development of a video surveillance system that operates through the Telegram messaging application. The choice of Telegram is based on its wide compatibility with a variety of devices and its ability to manage real-time communications efficiently. The system includes an image compression module to improve the speed of data transmission and reception, thus optimizing operational efficiency. In addition, strategies are contemplated for the future expansion of the system, such as integration with other messaging platforms and the incorporation of additional modules through tools such as App Inventor 2. This expansion will allow greater flexibility and customization of the system, adapting to different environments and needs. specific, and offering an advanced and robust solution for security requirements.

2. PROPOSED FRAMEWORK

The monitoring and automation system illustrated in Figure 1 employs an ESP32-CAM to capture and optimize images and videos, ensuring efficient transmission. The system components include the ESP32-CAM, file compression techniques, a motion detection sensor, and a notification module that sends Telegram alerts to devices such as smartphones, computers, and Tablets.

Figure 1. General framework of the surveillance IoT ESP32-CAM.

Real-time interaction is facilitated through an application in App Inventor 2, compatible with Android devices. Internet connectivity, either through Wi-Fi or SIM, allows the transmission of data and alerts. Upon detecting motion, the ESP32-CAM captures and optimizes the image, sending a notification to the user via Telegram or the application in App Inventor 2, providing real-time monitoring and security.

The implementation of a video surveillance system controlled through the Telegram messaging application uses various technologies to maximize its effectiveness. The integration of IP cameras offers significant advantages, such as the ability to capture high-resolution images and videos, allowing precise and detailed surveillance. In addition, these cameras can be remotely operated and monitored, eliminating the need for physical contact with the equipment and allowing centralized control [10].

The use of the Telegram API facilitates fast and efficient communication between the surveillance system and users, allowing the transmission of images and videos in real time. Incorporating image compression algorithms optimizes sending and receiving times, ensuring that captured images are transmitted quickly without significant loss of quality.

3. IMAGE ACQUISITION SYSTEM

The data acquisition stage is one of the most relevant in the proposed framework, if all well, the common domain in various applications; in this case it represents an essential component for the optimization of the system. The workflow of the image acquisition stage is illustrated in Figure 2.

To avoid equipment corrosion or any adverse environmental effects, the PIR sensor has been placed inside a protective casing that protects it from inclement weather and offers space for future adaptations. The detection distance of the PIR sensor has been set between 3 and 7 meters, calibrating the potentiometers to adjust the voltage and sensitivity (Figure 3).

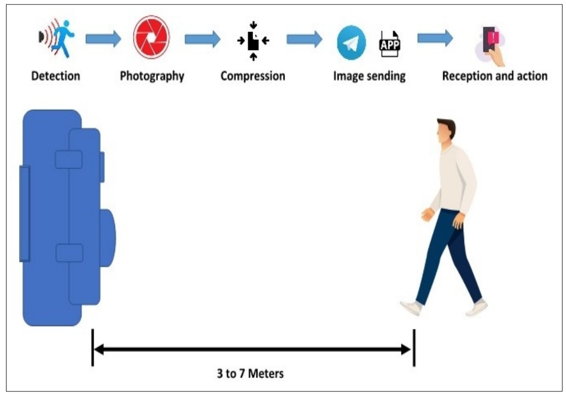


Figure 2. Workflow of the acquisition and processing stage.

It is recommended to set the voltage potentiometer at an initial value of 3.3V, with an adjustment range between 3.0V and 3.6V, to balance power consumption and sensor stability. For sensitivity, an initial value of 50% is suggested, adjusTable between 30% and 70%, allowing reliable detection within the desired range and minimizing false alarms. Calibration is performed by installing the PIR sensor in its housing, measuring and adjusting the voltage potentiometer starting with 3.3V, and then adjusting the sensitivity to 50%, testing and adjusting as necessary.

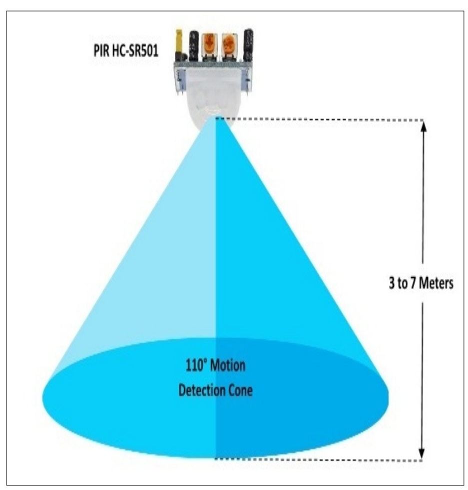


Figure 3. Detection range of the PIR sensor.

4. API CHATBOT Y APP INVENTOR INTELLIGENT COMMUNICATION APP

Integration with the Bot Father API for Telegram establishes bidirectional communication between the ESP32-CAM and users through the messaging platform. It is a key tool for creating and managing bots on Telegram, facilitating a wide range of applications and uses [11]. Bots created with Bot Father can be used to automate responses, send notifications, collect information, and perform complex interactions with users through text messages, images, and multimedia files. This makes them ideal for customer service applications, event management, information dissemination, and remote control of devices, as in the case of the ESP32-CAM.

For security issues, Bot Father controls access and interaction with the bot by setting specific permissions and commands. Commands can be customized to activate specific bot functions, such as starting and stopping image capture, adjusting camera settings, or sending alerts if motion is detected. This ensures that the bot responds only to authorized commands and protects against misuse. In Figure 4, the interaction and function of each of the bot components is visualized.

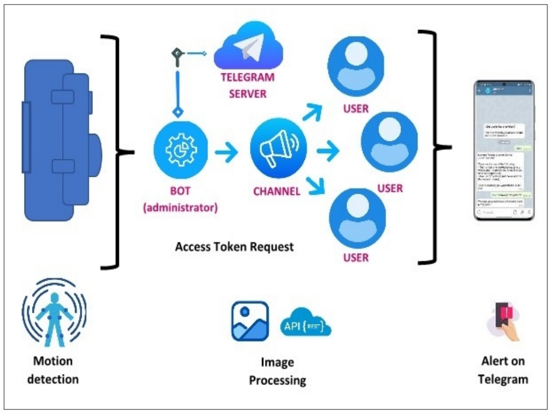


Figure 4. Functional diagram of the bot.

The application developed with App Inventor allows users to monitor in real time the camera connected to the ESP32-CAM. With an intuitive and easy-to-use interface, users can view captured images, receive notifications of detected events (such as motion) and control specific camera functions through built-in commands. This will provide an interactive and accessible experience for remote surveillance and automated control, enhancing the versatility and utility of the implemented system. Figure 5, illustrates the conceptual design of the proposed App.

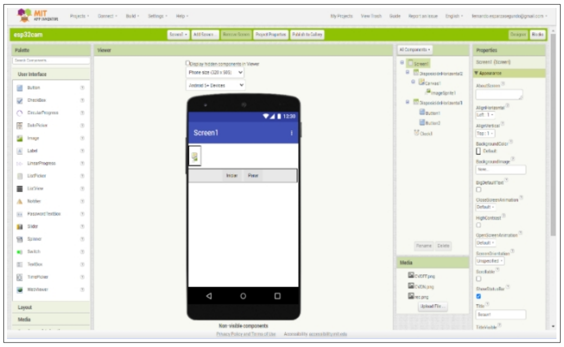


Figure 5. App Inventor 2 modelling interface.

5. IMAGE PROCESSING STAGE

The optimization process for sending images captured by the ESP32-CAM via Telegram is based on the direct implementation in the ESP32 firmware. The camera captures images, which are compressed using the JPEG standard to reduce their size while maintaining adequate quality. This was done with the JPEGDecoder library, which efficiently decodes and manipulates JPEG images on the microcontroller. Before transmission, images are resized to adjust their resolution based on viewing needs on Telegram, which optimizes bandwidth usage and improves transfer speed. This approach ensures that images are transmitted quickly and efficiently, adheres to the resource constraints typical of IoT environments, and provides a seamless experience to the end user [12]. Table 1, summarizes a comparison of the size and sending time of compressed and uncompressed images.

Table 1. Image sending parameters.

Image type	Size (KB)	Sent time (s)
Uncompressed	2048	5 - 10
Compressed JPEG	320	2

*Considering that 1 megabyte (MB) is equal to 1024 kilobytes (KB).

Using JPEG compression for images captured by the ESP32-CAM provides a significant improvement in data transmission efficiency, regardless of image resolution. This method effectively reduces file sizes, thus optimizing bandwidth usage and allowing fast transfers over conventional networks. Furthermore, JPEG being a universally compatible format, it ensures that images can be viewed without problems on various platforms and applications such as Telegram, as long as adequate visual quality is maintained by precisely adjusting the compression parameters.

6. COMPRESSION EVALUATION

During the tests performed with the ESP32-CAM module, image compression in SXGA resolution (1024x768) was evaluated using the JPEG format and the native “esp_camera” library. This compression system was chosen for its wide compatibility, its ability to reduce file size while maintaining accepTable visual quality, and its efficiency in environments with bandwidth limitations. JPEG compression optimizes image transmission over networks, facilitating its use in video surveillance and remote monitoring applications.

The selected SXGA (1024x768) resolution is based on several key considerations. First, this resolution provides an appropriate balance between image quality and file size. With enough detail to identify important features, such as faces and vehicle license plates, it is ideal for surveillance needs. In addition, SXGA resolution is a well-established standard that ensures compatibility with a wide variety of visualization and analysis software, making it easy to integrate into existing systems.

Another advantage of the SXGA resolution is its resource efficiency. Compared to higher resolutions, such as UXGA (1600x1200), SXGA consumes less processing and memory resources, which is crucial for a microcontroller such as the ESP32. This enables more efficient operation and extends the battery life in power-limited applications. It also enables real-time transmission over bandwidth-limited networks, making it ideal for IoT applications. A bandwidth of 1 to 3 Mbps is recommended to capture and transmit images in SXGA (1024x768) JPEG quality efficiently.

The original size of an image in color format (24-bit) with SXGA resolution is approximately 2.36 MB without compression. The following are the results obtained with different quality settings, which allow for a balance between the visual quality of the images and the file size.

Table 2. Quality adjustment parameters and size resolution from an image of 2.36 MB.

Quality	Description	Estimated compression (%)	Estimated size (KB)
0	Very high quality	0%	800 - 1200
10	High quality	40%	400 - 800
20	Good quality	50%	300 - 600
30	Medium quality	60%	200 - 400
40	Fair quality	70%	150 - 300
50	Low quality	80%	100 - 200
63	Very low quality	90%	50 - 100

** Quality = (jpeg_quality) image quality parameter for the “esp_camera” library.

Using a quality setting of 20, a final file size of approximately 300 to 600 KB was achieved, representing a 50% reduction compared to the original size without compression. With a quality setting of 40, the size was reduced to about 150 to 300 KB, indicating a reduction 70%. You can fine-tune the compression quality value for “jpeg_quality” according to your needs or make use of the libwebp library available under the project name WebM in the GitHub repository with the license type COPYING; note that to improve the results, you can rewrite the code using both libraries and adapt them to improve the compression of the image and have higher visual quality without loss by using the esp_camera library, which was applied in this project.

7. EXPERIMENTAL SETUP

This section presents the experimental configuration of the interconnection of the components of the proposed system (Figure 6).

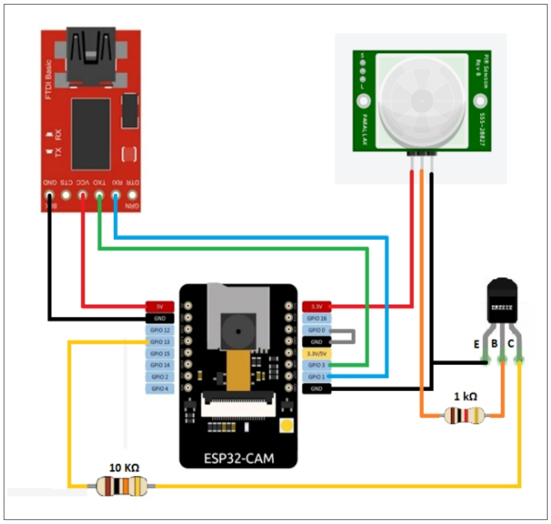


Figure 6. Component interconnexion diagram.

This monitoring system uses an ESP32-CAM along with a PIR sensor and a 2N2222 transistor. The 2N2222 transistor was selected for its optimal characteristics in applications, ability to handle up to 800 mA of collector-emitter current, and fast switching speed. This component acts as a switch controlled by the PIR sensor, amplifying the motion detection signal to initiate image capture by the ESP32-CAM. In addition, an FTDI adapter was integrated for serial programming and communication with the ESP32-CAM, facilitating firmware loading and code debugging during system development. This configuration not only optimizes event detection and visual data capture, but also provides capability to handle currents higher than what the ESP32-CAM can directly handle, ensuring an adequate level of isolation and protection for the microcontroller and allowing an efficient control of the monitoring system in real time [13].

The design of the prototype has been customized, specifically adapting to the dimensions of the ESP32 card and the PIR sensor module. This adaptation guarantees a precise and safe assembly, ensuring that both components are optimally integrated for the operation of the video surveillance system. A meticulous approach has been taken to design the enclosure that houses these components, considering not only physical dimensions but also protection against adverse environmental conditions that may affect system performance. In addition, necessary interfaces have been integrated to facilitate the connection and operation of the prototype with other devices and systems, ensuring robust functionality and versatile installation in different environments [14].

8. CASE

The cover is designed with precise dimensions to house the camera lens and PIR sensor dome, ensuring an optimal fit that protects these key components. It incorporates enough space for two 18650 batteries, allowing the system to maintain its operation during power interruptions through the main cable. In addition, it has four attachment points to firmly secure the front and back cover, using screws, and has two strategically located notches to facilitate mounting on vertical surfaces. This design not only optimizes the safety and stability of the prototype, but also provides effective protection against various adverse environmental conditions, ensuring the robustness and reliability of the system in demanding applications. Figure 7 and Figure 8 show the front view and rear view of the 3D model, respectively.

The cover design makes it easy to access the interior for changing batteries or making technical adjustments, ensuring convenient and efficient installation and maintenance. Figure 9 presents the proposed version of the casing.

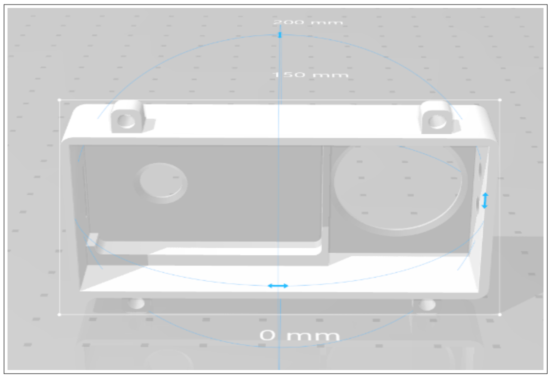


Figure 7. Front view of the 3D model of the case.

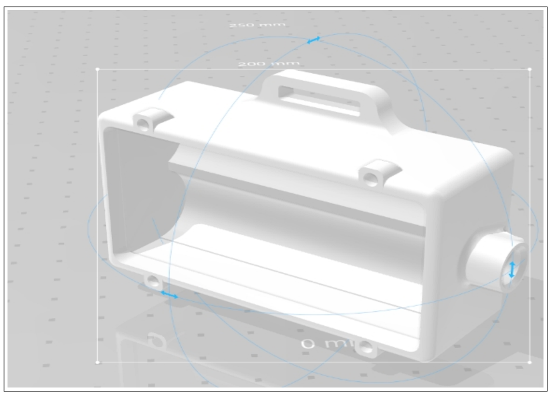


Figure 8. Rear view of the 3D model of the case.

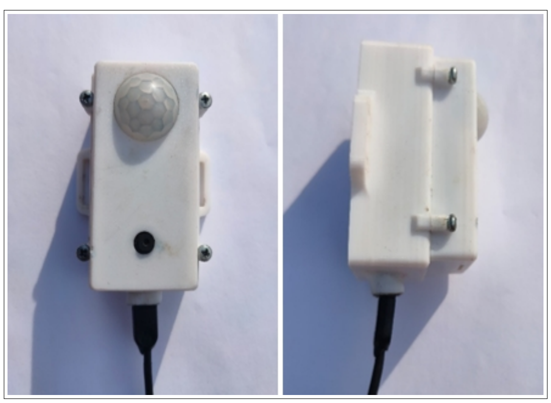


Figure 9. Final prototype case version.

9. TESTING AND EVALUATION SCENARIO

The prototype was evaluated over a period of 7 continuous days outdoors and indoors in homes, where temperatures ranged from 10°C to 30°C, with a semi-dry temperate climate. During the heat tests, the system was exposed to temperatures approaching 30°C and to direct sunlight at times when it was shining on the prototype. These conditions allowed the evaluation of its heat resistance, and the performance was positive, as the prototype operated stably without overheating, demonstrating its ability to operate in hot climates typical of the region. To optimize thermal management in the future, it is proposed to integrate a temperature sensor that detects overheating conditions inside the chamber and sends alerts to the user.

The prototype integrated with the ESP32-CAM dissipates approximately 1 W of heat during operation, calculated by the equation $P=VI$ where V is the voltage (5 V) and I is the current (approximately 200 mA). This device is suitable for operation in a temperature range of 10°C to 30°C, ensuring stable and reliable performance without significant risk of overheating. Although it can tolerate temperatures up to 85°C [15], adequate ventilation should be ensured and prolonged exposure to direct sunlight should be avoided. For humidity testing, the system was exposed to moderate rainfall under a covered surface, which prevented direct contact with water, but kept it in constant contact with ambient humidity. Although the housing has an airtight design, there was a need to improve the internal protection of the electronic components, especially by reinforcing the dome part of the PIR sensor, to ensure its durability in prolonged high humidity conditions, such as those typical of the rainy season in the region. Additionally, it is recommended to implement a thermal dissipation system, which can be active (fans) or passive (materials such as thermal pads), to maintain operational stability during prolonged exposure to high temperatures, thus ensuring greater durability in more demanding conditions.

10. COMMUNICATION PROTOCOL AND COMMANDS SEQUENCE

The integration of the Bot Father API for Telegram provides a robust two-way communication channel between the ESP32-CAM module and users, using the Telegram messaging platform as the main interface. This API is founded on the REST (Representational State Transfer) API approach, which provides an efficient and flexible framework for interaction between devices and the Telegram server. It relies on the HTTP protocol to carry out requests and receive responses in a structured format, thus facilitating effective communication between the bot and end users.

In operational terms, the Telegram API uses the HTTP protocol, allowing communication via methods such as GET and POST. For example, to send a message to a user, the endpoint https://api.telegram.org/bot<token>/sendMessage is used, where <token> represents the unique identifier of the bot, which is obtained during its creation through Bot Father. Similarly, to transmit images captured by the ESP32-CAM, the endpoint https://api.telegram.org/bot<token>/sendPhoto is used, enabling fast and efficient transmission of visual content to users.

Interaction with the API is done in JSON format, a widely used standard that simplifies data manipulation. This format is lightweight and easily understood by both humans and machines, increasing the efficiency in processing the information exchanged between the ESP32-CAM and the users. Figure 10, shows the general interaction process.

The bot has the ability to receive updates, such as messages from users, through two main methods. The first is Long Polling, in which the bot makes periodic requests to the API to check for the availability of new messages. The second method is Webhook, a technique that allows Telegram to send updates to the bot in real time to a specific URL. The latter strategy enables faster and more efficient interaction, ensuring that users receive notifications almost instantaneously.

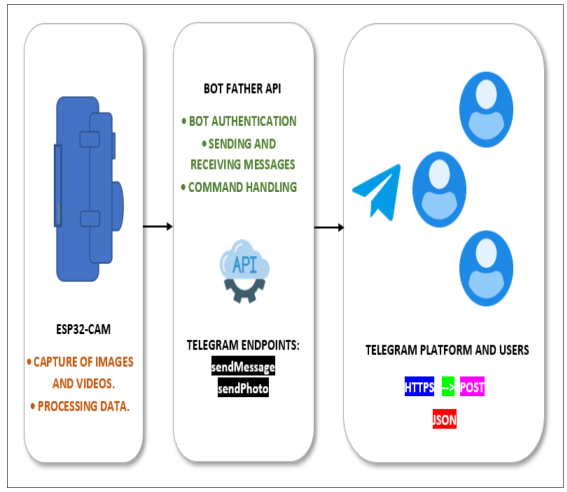


Figure 10. Communication between the ESP32 and the API.

In addition, the Bot Father API implements several security mechanisms that safeguard bot communication and interactions. Each bot is authenticated by a unique token, generated during its creation, which must be kept secret to prevent unauthorized access. The API also allows custom commands with specific permissions, ensuring that only authorized users can execute certain actions. Communication is carried out through the HTTPS protocol, which guarantees that the transmitted data is encrypted and protected against possible interceptions, thus strengthening the overall security of the system.

Communication between the ESP32-CAM and the application developed in App Inventor 2 is established using HTTP (Hypertext Transfer Protocol), where the ESP32-CAM acts as a web server, exposing an endpoint accessible through a specific IP address (e.g. http://192.168.0.122/capture) to receive image capture requests. To avoid saturating the ESP32-CAM during video delivery, frame forwarding is implemented, allowing for more efficient transmission. The client application sends HTTP POST requests to the server at the push of a button, triggering the capture of images in formats such as JPEG or PNG. In addition, a block has been programmed into the application that allows periodic requests to be made, updating the image in the interface, and simulating a continuous display. This approach optimizes the interaction between the camera and the ESP32-CAM, improving image capture and transmission in real time.

11. RESULTS AND DISCUSSION

The proposed system is functional through the Telegram application using the Bot API created specifically for this purpose. In addition, it is compatible with an Android application developed with App Inventor, providing an intuitive interface for remote monitoring and control of the system.

11.1. TELEGRAM INTERFACE

Control via Telegram is done efficiently by associating the user account with the device. Once linked, a new conversation will appear in the chat list from where you can remotely interact with the camera prototype. Connected to a WiFi network, the system allows you to take full control of the equipment and receive alerts remotely.

Among the available functions are instant photo capture, enabling or disabling the PIR sensor to avoid false alarms, and enabling or disabling the flash for photos. It is important to note that the prototype performs regular checks to ensure the proper functioning of its components and guarantees a period of lens stabilization before each capture. Figure 11, presents a screenshot of the running application.

11.2. APP INVENTOR 2

The application developed in App Inventor (Figure 12) was designed to provide an interactive real-time surveillance experience by continuously capturing and viewing images. By activating the image capture function, which is done at a regular interval of approximately one second, the application automatically updates the canvas ('Canvas1') with each new image captured. This feature allows users to instantly and constantly monitor what is happening in the environment where the camera is located, thus facilitating remote monitoring and event detection in real time.

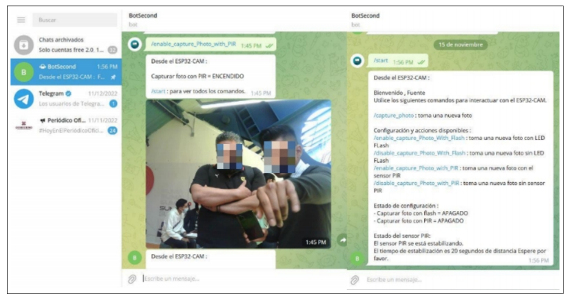


Figure 11. Telegram operating interface.

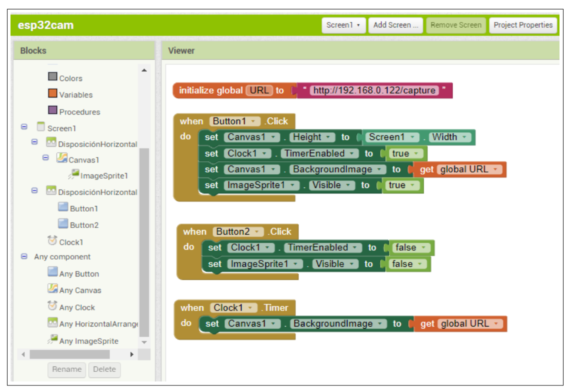


Figure 12. Main code block for real-time visualization.

To ensure reliability and constant connection, the app assigns a fixed IP address to the device, which is crucial to maintain control and receive continuous updates even in case of momentary disconnections. In Figure 13, a screenshot of the image display interface on the smartphone is presented.

Through the sequential capture of images with a reduced interval between each capture, it is possible to generate a continuous sequence of frames that, when joined together, allow a video to be viewed in real time.

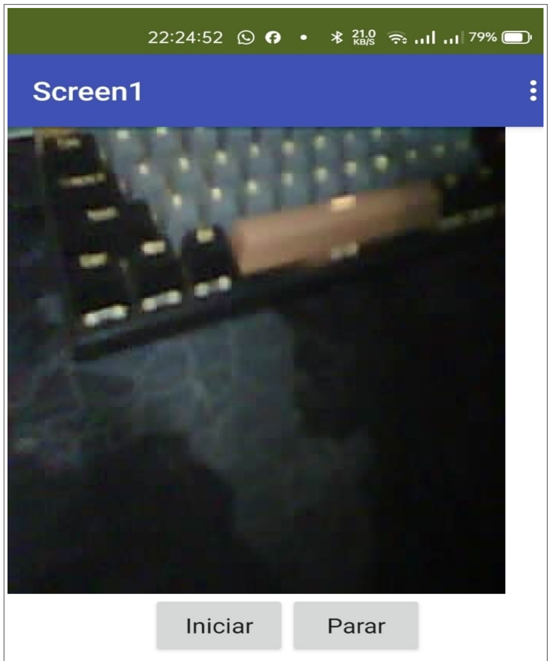


Figure 13. Image display from the APP.

12. CONCLUSIONS

In this work, an integrated IoT-based system for security and surveillance was presented, which combines the ESP32-CAM with a PIR sensor. During a four-week evaluation, the operational reliability of the system has been demonstrated, ensuring sTable connectivity to the local WiFi network and efficient data transmission through Telegram and through the mobile application developed in App Inventor. Implementing JPEG compression for captured images significantly optimized data transmission efficiency, reducing file sizes without compromising visual quality. Communication with the Telegram platform was robust, facilitating effective control and timely receipt of alerts and notifications.

For future improvements, it is proposed to adjust the PIR sensor parameters to reduce false alarms and improve detection in various environmental conditions. Additionally, the integration of artificial intelligence algorithms will be explored to improve the accuracy in identifying critical events. Expanding connectivity to include additional protocols will enable broader integration with other security devices and systems, thereby improving remote monitoring and management capabilities in dynamic and complex environments.

The integration of platforms such as WhatsApp and Discord through APIs would allow direct communication and the sending of notifications in real time, facilitating faster responses to critical situations. It is also proposed to implement artificial intelligence algorithms for data analysis, which would optimize the detection of relevant events and reduce false alarms through continuous learning based on behavioral patterns. These improvements would not only enhance the functionality of the system, but also enrich the user experience by providing more accurate and personalized alerts. Together, these innovations represent a significant advance in the efficiency and usability of the proposed system.

REFERENCES

[1] Rahimi Ardabili B, Danesh Pazho A, Alinezhad Noghre G, Katariya V, Hull G, Reid S, et al. Exploring public's perception of safety and video surveillance technology: a survey approach. *Technology in Society*. 2024; 78: 102641. doi: 10.1016/j.techsoc.2024.102641.

[2] Maheswari P, Karishma VR, Vigneswaran T. Artificial intelligence in video surveillance. In: Swarnalatha P, Prabu S, editors. *Handbook of research on deep learning techniques for cloud-based industrial IoT*. Hershey, PA: IGI Global; 2023. p. 1-17. doi: 10.4018/978-1-6684-8098-4.ch001.

[3] Samuda P, Sivachandar K, Praveena NG, Nithiya C, Kamalesh D, Lokesh. Low-cost prototype for IoT-based smart monitoring through Telegram. In: 2023 *5th Int. Conf. on Smart Systems and Inventive Technology* (ICSSIT); 2023 Jan 23-25; Tirunelveli, India. IEEE; 2023. 1-5. doi: 10.1109/ICSSIT55814.2023.10061155.

[4] Sudaryanto A, Sasongko D, Kridoyono A, Budiarti RPN, Mahadewi ST, Arvianto FA. ESPCam control using Telegram on ESP32 microcontroller-based security camera systems. *Applied Technology and Computing Science Journal*. 2022; 5(2): 21-32. doi: 10.33086/atcsj.v5i2.3760.

[5] Cahyono FYA, Suharto N, Mustafa LD. Design and build a home security system based on an ESP32 Cam microcontroller with Telegram notification. *Journal of Telecommunication Network*. 2022; 12(2): 58-62.
[6] Ahmed T, Nuruddin ATB, Latif AB, Arnob SS, Rahman R. A real-time controlled closed loop IoT based home surveillance system for Android using Firebase. In: *2020 6th Int. Conf. on Control, Automation and Robotics* (ICCAR); 2020 Apr 20-23; Singapore. IEEE; 2020. 601-606. doi: 10.1109/ICCAR49639.2020.9108016.
[7] Siddineni B, Nanditha R, Satyanarayana TJ, Rama Krishn VS. Design of an IoT based surveillance system using Blynk, IFTTT, and Telegram. In: *2021 12th Int. Conf. on Computing Communication and Networking Technologies* (ICCCNT); 2021 Jul 6-8; Kharagpur, India. IEEE; 2021. doi: 10.1109/ICCCNT51525.2021.9579790.
[8] Kurniawan FA, Rochimawati I, Saddiq A, Saragih Y, Saragih CI. Implementation of an ESP-32 Wi-Fi CAM & Arduino-based robot for IoT surveillance. *Seminar Nasional Teknik Elektro*. 2023; 2(1): 1-4.
[9] Ashok MS, Sneha G, Pushpa NG, Keerthika TT. ESP32: the ultimate IoT solution for surveillance. *International Journal for Advancced Research & Technology*. 2023; 13(10): 137-145.
[10] Rodríguez Silva DA, Adkinson Orellana L, González Castano FJ, Armino Franco I, González Martinez D. Video surveillance based on cloud storage. In: 2012 *IEEE 5th Int. Conf. on Cloud Computing*; 2012 Jun 24-29; Honolulu, HI, USA. IEEE; 2012. 991-992. doi: 10.1109/CLOUD.2012.44.
[11] Martín Turrión R. *Diseño e implementación de un sistema embebido de telecontrol basado en Arduino Cloud* [master's thesis]. Santander: Escuela Técnica Superior de Ingenieros Industriales y de Telecomunicación, Universidad de Cantabria; 2024.
[12] Cameron N. ESP32-CAM camera. In: *ESP32 formats and communication: application of communication protocols with ESP32 microcontroller*. Berkeley, CA: Apress; 2023. p. 447-88. doi: 10.1007/978-1-4842-9376-8_11.
[13] Gutiérrez Bravo JM. *Desarrollo de un sistema de detección de movimiento usando un sensor infrarrojo pasivo para implementación en robótica* [undergraduate thesis]. Medellín, Colombia: Instituto Tecnológico Metropolitano; 2021.
[14] Vedaei SS, Fotovvat A, Mohebblan MR, Rahman GME, Babyn P, Marateb HR, et al. COVID-SAFE: an IoT-based system for automated health monitoring and surveillance in post-pandemic life. *IEEE Access*. 2020; 8: 188538-188551. doi: 10.1109/ACCESS.2020.3030194.
[15] Espressif Systems. ESP32 datasheet [Internet]. 2024 [cited 2024 Oct 2]. Available from: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf

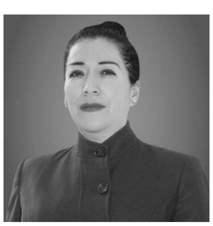
ABOUT THE AUTHORS

Fernando Esparza Segundo is a Technician in Computer Equipment Support and Maintenance with experience in the public, private, and educational sectors. He has developed technological solutions that optimize digital infrastructure, including web platforms to improve administrative processes, internal networks, servers, and video surveillance. His focus is on creating efficient tools that modernize systems and enhance user experience, always ensuring security and technological functionality in every project.

Shany Lizbeth Enriquez Arcadio is a Human Resources Administration Technician and a student of Telecommunications Engineering at the Autonomous University of the State of Hidalgo, currently in her 9th semester. Throughout her university career, she has earned recognitions such as First Place in the Intermediate Category of the "Expo Computación, Electrónica y Telecomunicaciones" at UAEH and Third Place in the AMRob Prototype Competition at the XXIV Mexican Robotics Congress of the Mexican Robotics Association at UAEH, both in 2022. She is currently completing her professional internship at the University System of Autonomous Media, the university's radio and television system at UAEH, where she carries out telecommunications-related activities, from networking to operational tasks.

Juan Carlos González Islas graduated from the Electronics and Telecommunications Engineering and the Computer Science PhD at the Hidalgo Estate Autonomous University (UAEH), in 2006 and 2023, respectively. He obtained a Master of Science degree in Electronics and Telecommunications from the Center for Scientific Research and Higher Education of Ensenada Baja California (CICESE) in 2009. Since 2023, he has been a member of the National System of Researchers (SNII) Level I. He is currently a researcher attached to the Academic Area of Computing and Electronics of the UAEH. He has three patents granted by the Mexican Institute of Industrial Property (IMPI). Twice winner of the Hidalgo Award for Science, Technology and Innovation (CITNOVA) in Mexico.

Alberto Suárez Navarrete graduated from the Autonomous University of the State of Hidalgo with a degree in Computer Systems. He also completed his master's studies at Rey Juan Carlos University in Madrid, Spain, in the Master's in Decision Engineering. He is currently a member of the Advanced Graphics Technology for Educational Environments Research Group, focusing on Educational Computational Technology.



Ma. de Jesus Gutierrez Sanchez is a graduate of the Autonomous University of the State of Hidalgo with a degree in Computer Systems. She completed her master's studies at Rey Juan Carlos University in Madrid, Spain, in Decision Engineering. She also completed her PhD in Educational Technology. Currently, she is a member of the Advanced Graphics Technology for Educational Environments Research Group, focusing on Educational Computational Technology.