SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V11I12P126

Volume 11 Issue 12, 287-294, December 2024

Original Article

IoT-Based Smart Glasses Assistance System with Facial Recognition

Flavio Cesar Abarca Jahuira[1], David Cana Salas[2], Jesús Talavera Suarez[3]

[1,2,3]*Universidad Nacional de San Agustín de Arequipa, Arequipa, Perú.*

[1]*Corresponding Author : fabarca@unsa.edu.pe*

Received: 14 October 2024 Revised: 15 November 2024 Accepted: 13 December 2024 Published: 31 December 2024

Abstract - *Smart glasses are becoming increasingly available in the population due to their ability to provide real-time environmental information; however, the incorporation of facial recognition through the Internet of Things marks an important advance in the performance and usability of these devices. This study describes the creation of a novel Internet of Things-based human face recognition assistance system for smart glasses. The developed system uses a Raspberry Pi 4 board, an ESP32 board, two camera modules to capture and process the environment through artificial vision, and a sound module responsible for realizing the assistance system. Developing a functional prototype of smart glasses will enable visually impaired people to obtain information from others and be aware of potential environmental dangers. This system is made possible through facial recognition technology and distance estimation algorithms. In addition, this study addresses integrating the proposed system with the Internet of Things, enabling better connectivity and effective communication with other devices and services. Finally, practical tests are used to evaluate the usability and user experience of the system, as well as its accuracy in measuring distances and identifying faces of previously registered users. Studies show that the system is quite effective in assisting people with partial or total blindness and ensures that it is simple to use and intuitive in real-world scenarios.*

Keywords - *Computer vision, Real-time assistance system, Smart glasses, Internet of Things.*

1. Introduction

The numerous applications that smart glasses offer to make people's lives easier have made them a significant technological instrument that provides a unique and fascinating User Experience (UX) [1]. The scientific community and industry have been very interested in it since its inception because it has the potential to significantly change how people use digital information in their daily lives [2].

More work must be done before smart glasses can reach their full potential in terms of usability and convenience. One such issue is to improve their ability to sense and understand their environment. Facial recognition capabilities are especially important for human-computer interaction and personalized digital experiences [3]. In recent decades, the development of deep learning algorithms and the increase of easily accessible training data have led to a significant advancement in computer vision, known as the face recognition branch. This recent technology has applications in various research areas, including biometric authentication, personalization of services, security and surveillance [4].

A potential breakthrough has been achieved in improving the functionality and usability of these smart devices by integrating facial recognition with the Internet of Things (IoT) infrastructure [5]. Smart glasses can provide a more personalized user experience based on individual user requirements, as with integral real-time facial recognition.

This work presents a new method for designing and developing Internet of Things-based assistive solutions incorporating facial recognition into smart glasses. Building specialized hardware and software for these glasses, integrating them smoothly with other IoT devices, and assessing usability and user experience are all essential to ensuring the system functions correctly in various real-world scenarios. This effort is intended to progress the smart glasses field and show how this improved technology may revolutionize several sectors and improve people's daily lives.

The distribution of information in this paper is divided as follows: Section 2 presents work related to this research. Subsequently, Section 3 explains the complete methodology of this proposed system. Section 4 details the design and construction of the hardware and software system and its implementation in a real agricultural environment. Section 5 presents the outcomes and debates that were analyzed. Lastly, Section 6 contains the research's conclusions.

2. Related Work

The popularity of smart glasses has increased significantly in recent years, mainly due to advances in the integration of cutting-edge technologies such as computer vision, Augmented Reality (AR), Internet of Things (IoT) technology and the decreasing quantity and size of electronic components. These advances have led to new uses for smart glasses, which have evolved from essential wearable devices to sophisticated instruments that offer on-site assistance for various applications, from surgery in medicine to their application in the industrial automation sector. Smart glasses have been considered a promising technology platform for human-computer interaction since their earliest versions.

Studies by Niknejad et al. [6] provide a comprehensive analysis of the current state of smart glasses and show how glasses have evolved to incorporate real-time data processing, contextual information display, and augmented reality capabilities. These devices have enhanced the user experience by using Internet of Things connectivity to provide contextualized information based on the user's environment. Mitrasinovic et al. [7], in their paper, took a more specific approach to investigate the use of smart glasses in the healthcare sector for remote patient monitoring, telemedicine and surgery. Their research shows how these smart glasses technology can provide medical staff with hands-free access to data and visual aids, thus improving the quality and efficiency of medical care.

The face recognition system, a key feature of smart glasses technology, has advanced significantly in recent years, especially with deep learning methods. In evaluating advances in face recognition algorithms, Wang et al. [3] note that Convolutional Neural Networks (CNNs) have made possible more accurate face detection in a variety of environments, including variations in lighting, facial expressions, and locations, thereby reducing the error rate that was common when this CNN technology was not used. This has been essential for security applications, biometric authentication, and personalization services, which are now widely used. In addition, the paper by Guo and Zhang [4] discussed the impact of advances in deep learning-based face recognition, noting that deep neural networks have improved accuracy and processing speed, which is crucial for real-time systems such as the one proposed in this research. However, optimising these algorithms to work efficiently on resource-constrained devices such as smart glasses based on microcontrollers or embedded microprocessors remains challenging.

Many research papers have focused on integrating wearable devices with the Internet of Things. In their study, Dian et al. [8] investigated how the Internet of Things (IoT) might improve wearable technology's capabilities, such as smart glasses, by facilitating constant connectivity and communication with other devices and networks. In this case, IoT connectivity expanded the potential uses of wearables while allowing them to communicate in real-time with more extensive infrastructures, including industrial systems or dispersed sensor networks in cities. In addition, Rahmani et al. [5] discussed the advantages and difficulties of combining IoT technologies. In particular, it was noted that the main challenge is achieving stable and secure connectivity, minimising power consumption, and ensuring data privacy, which is critical for deploying smart glasses with facial recognition and obstacle detection capabilities.

Another work that specifically examined the feasibility of integrating real-time face recognition systems into smart glasses was that of Casado et al. [9], where the proposed algorithm's hardware performance and efficiency are emphasised. Their results demonstrate that, by combining advanced cameras and computers, these systems are technically feasible despite the current problems of processing speed and power consumption. This indicates a promising future for integrating facial recognition and machine vision algorithms into tiny devices such as smart glasses. A Recent research examines a facial recognition smart glasses system made specifically for blind and mildly visually impaired people. This research combines a Raspberry Pi board with low-power cameras connected through an Internet of Things network using a natural language processing technique. The results highlight the importance of efficient image processing and a user-friendly interface for improving user comfort.

Real-time distance estimation is a critical functionality for smart glasses-based assistive systems, especially for visually impaired people, as it allows for detecting obstacles and estimating the proximity of objects in the environment. Computer vision, combined with triangulation and stereo vision, has proven to be an effective technique for this purpose. According to Mellouk and Kortli [10, 11], the accuracy of distance estimates has improved significantly due to the use of neural networks and deep learning algorithms. This is important for real-time applications where accurate and timely responses are required. These advances are possible because mobile devices now have more processing power, but there is still a challenge for algorithms to be tuned and optimized to run on resource-constrained hardware, such as smart glasses. Besides stereo vision, complementary technologies have been explored to enhance distance estimation. For example, several studies have combined machine vision with LIDAR or ultrasonic sensors to recognize objects in more challenging-to-reach locations, improving accuracy and redundancy. Yeong et al. [12] claim that using various sensing methods increases the system's durability in challenging situations, including limited visibility or reflective surfaces, and improves accuracy. This is particularly crucial in urban settings, where moving items and quickly shifting surroundings present obstacles to navigation and aid systems. The difficulty, though, is attempting to fit more electronics into a smaller package like smart glasses.

3. Methodology

The article presents the development of an assistance system for smart glasses based on facial recognition from the Internet of Things. The system uses a Raspberry Pi 4 board, an ESP32 board, two camera modules to capture the environment and process them through computer vision and a sound module to realize the assistance system. The complete system design can be seen in Figure 1.

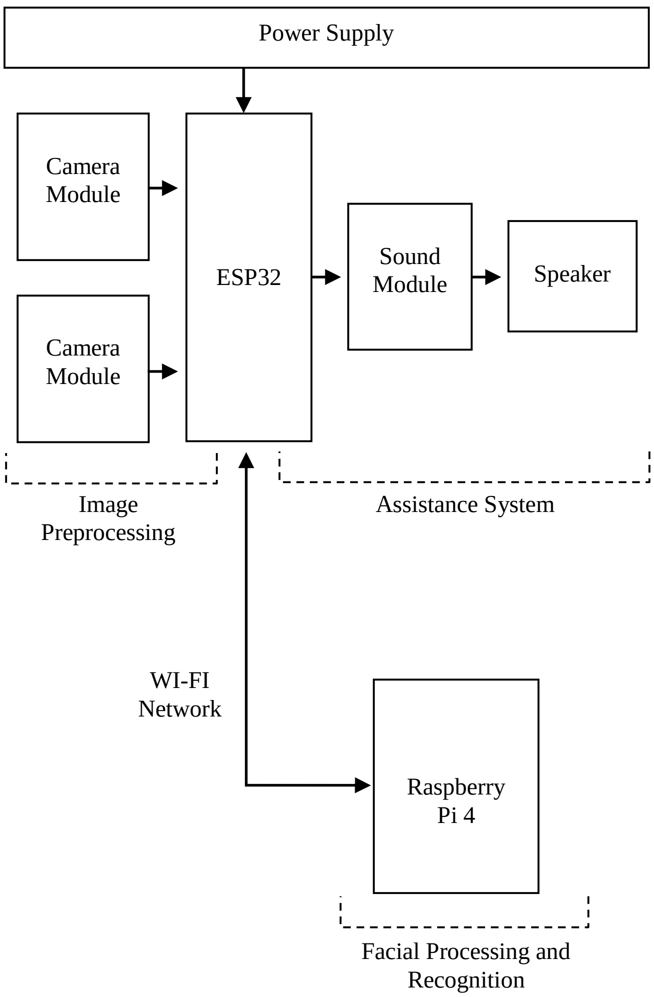

Fig. 1 Block diagram of the complete proposed system

The proposed system is divided into three main stages based on the tasks assigned to the different software and hardware modules:

- Image Preprocessing Stage: This first stage is controlled by the ESP32, coupled to two cameras. Stereo vision can estimate distance since the ESP32 captures images from multiple angles. The ESP32 preprocesses the images and sends them to the Raspberry Pi after they are taken.
- Facial Processing and Recognition Stage: The Raspberry Pi 4 receives the preprocessed images from the ESP32 via a Wi-Fi network. The Raspberry Pi is used to run facial recognition algorithms, which compare the faces in the images with a pre-established database; a Convolutional Neural Network (CNN) algorithm previously trained for real-time face identification is used for this facial recognition. The system sends a message with their name if the person is recognized.
- Assistance System Stage: Using the images taken by both cameras, the Raspberry Pi calculates the distance of the observed objects in addition to facial recognition. Triangulation and stereo-vision methods form the basis of this calculation. After processing the data, Raspberry Pi instructs the ESP32 to play audio warnings via the sound module, alerting the user to faces it has identified or obstacles that are nearby.

In addition, to validate the developed system, the following tests were proposed:

- Tests for distance estimation: These assess how well a user can estimate the distance to items in their immediate vicinity. This is accomplished by comparing the system's output with the actual measurements using established distance references.
- Facial recognition results: An assessment of the system's capacity to recognize faces at different illumination levels and capture angles. Accuracy is determined by comparing the outcomes with a database of recognized faces.
- Real-Time Latency and Performance Testing: The system's response time is assessed from the ESP32's image capture to the sound module's alert playback. The goal is to ensure the system runs in real-time with the least latency in data processing and transmission.

4. Experimental Development

The proposed system's implementation began with the configuration of the hardware components. There were two boards, a Raspberry Pi 4 and an ESP32. Both boards worked with the client-server model, with the Raspberry Pi as the server, the one that would do the image processing, and the ESP32 as the client that sent all the data to the server through a Wi-Fi network. In addition, two OV7670 model camera modules were connected to the ESP32 to capture the images of the environment, and a YX5300 MP3 sound module was also connected to the ESP32 to reproduce the sound of the assistance system. Figure 2 shows the prototype developed.

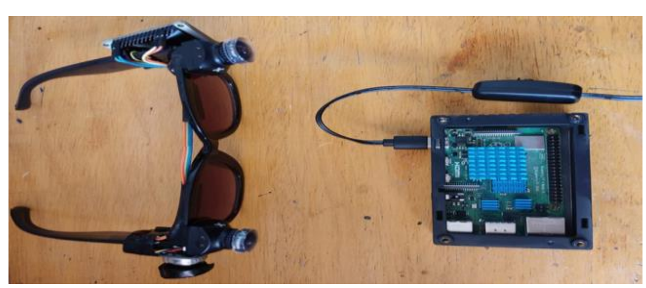

Fig. 2 Image taken from the prototype of the system

4.1. Hardware Configuration

This section describes the connections and configurations of the electronic modules required to develop this assistance system. First, two OV7670 camera modules are connected to an ESP32 microcontroller. However, specific procedures must be followed to ensure proper communication and avoid data conflicts. The pins for each component must be located and prepared. The 3.3 V power supply (VCC) and ground (GND) of the Raspberry Pi 4 board are shared by both camera modules, and this is essential for their functionality, see Figure 3. In addition, the GPIO22 and GPIO21 pins of the ESP32 are used for the SDA and SCL pins on the I2C bus shared by both cameras, respectively. This way, the same serial communication channel can send configurations to both cameras. The input clock (XCLK), which distributes the clock signal to both cameras for synchronization, is implemented by the GPIO32 pin of the ESP32. Each camera requires individual pins for control and data signals to avoid interference. For the first camera, the GPIO36 and GPIO26 pins are assigned for data transmission (D0-D7), the GPIO25 pin for Pixel Clock (PCLK), GPIO23 for Vertical Synchronization (VSYNC) and GPIO26 for Horizontal Reference (HREF). In the second camera, pins GPIO14 and GPIO16 transmit data (D0-D7), pin GPIO19 for PCLK clock signal, GPIO18 for VSYNC, and pin GPIO5 for HREF.

On the other hand, the YX5300 sound player module can be connected to the ESP32 microcontroller more easily because it only needs two wires for serial communication to work. The RX pin of the YX5300 sound module must be directly linked to the TX pin (2) of the ESP32 microcontroller, and the TX pin (1) of the ESP32 must be directly connected to the RX pin of the YX5300 module for the serial communication to be successful. Next, the power supply is connected to pins (3) and (4) of the YX5300 module, representing GND and VCC, respectively, see Figure 4.

The selected arrangement allows the two cameras in this system to function independently because each one needs its own set of data and control pins. The ESP32 board implements code that offers alternative picture capturing by initializing and deactivating each camera as needed. The produced code uses the esp_camera library to disable the current camera, configure the other camera, collect an image frame, and repeat these steps until the process is finished. The ESP32 microcontroller can control the two cameras with the limited resources available by employing a different technique. This action avoids conflicts on the data bus and ensures that a single camera is operational and transmitting data without any problems. This methodology is necessary for machine vision systems, specifically in monitoring and automation applications, where it is required to capture and process images from multiple sources in a single microcontroller with limited performance, as in the case of ESP32. The connection and configuration described above allow the ESP32 board to efficiently manage two OV7670 camera modules, facilitating its use in Internet of Things (IoT) projects and other embedded systems. The connections of the electronic components can be seen in Figures 3 and 4.

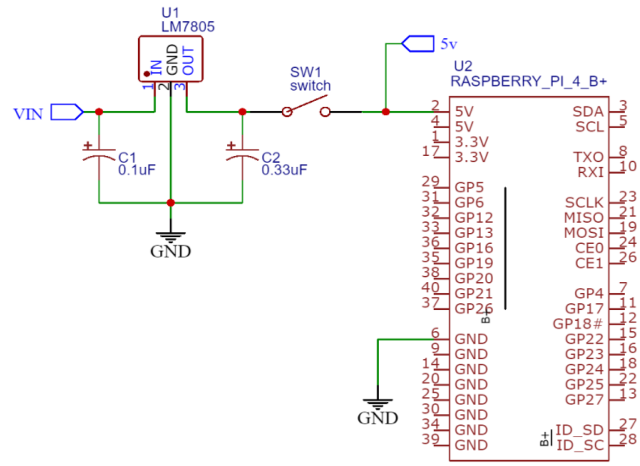

Fig. 3 Connections of the Raspberry Pi board to the power supply

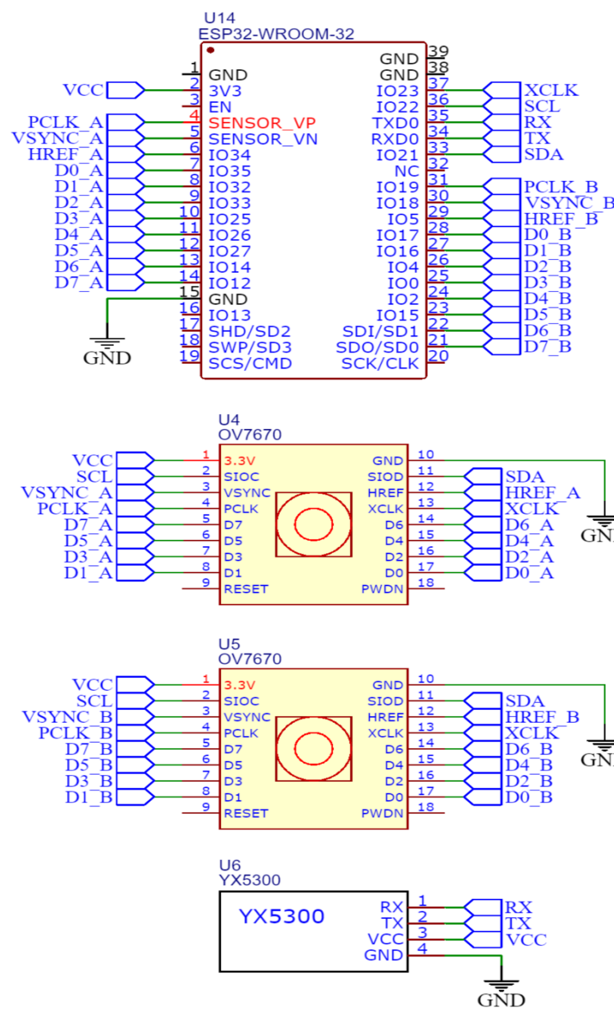

Fig. 4 Connection of the ESP32 with the two cameras and the sound player module

4.2. Algorithm Developed

The proposed system consists of an ESP32 connected to two OV7670 cameras and a YX5300 sound player module, which communicates via Wi-Fi with a Raspberry Pi 4 to execute the image processing algorithms and control the assistance system. Next, the method designed for distance estimation and face recognition is reviewed, as well as how device communication is handled. The created algorithm's complete flowchart is displayed in Figure 5.

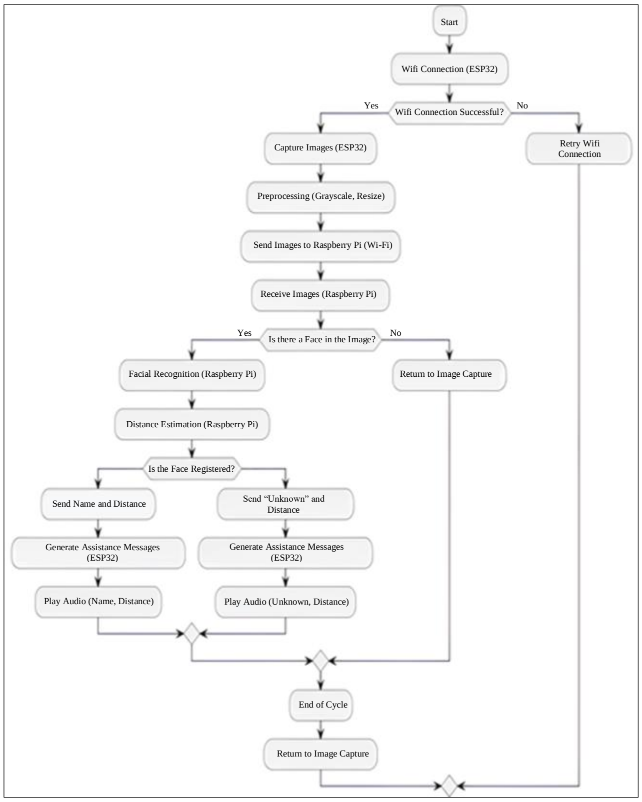

Fig. 5 Flowchart of the developed algorithm

4.2.1. Image Capture and Preprocessing on the ESP32

The ESP32 board is connected to two model OV7670 cameras to capture image frames. The limited resources of the ESP32 board prevent the cameras from operating at the same time. Instead, the algorithm alternates between the two cameras to capture images from each side of the lenses, which was considered a correct triangulation. The ESP32 initializes the system by connecting the two cameras with a common I2C bus. The control signals (GPIO) are set up to alternate image capture such that only one camera is active during each capture cycle, preventing data bus conflicts. To do this, the capture is switched in this way:

- Camera 1: It captures an image of the environment and stores it temporarily in the ESP32's memory for later processing.
- Camera 2: The first camera takes a picture, and then the second camera repeats the procedure by turning on and off. In order to increase the accuracy of distance calculation, the system can gather photos from two somewhat different perspectives thanks to this continuous cycle.

Preprocessing also includes reducing the resolution of the images to a more manageable format eliminating unnecessary details that do not affect the performance of the facial recognition algorithm. In addition, basic filtering is applied to reduce visual noise in the captures, especially in low-light conditions or complex environments. These optimizations allow the system to transmit images efficiently, maintaining low latency without compromising the processing accuracy of the Raspberry Pi.

4.2.2. Raspberry Pi Processing and Configuration

The Raspberry Pi is the primary server in a client-server communication arrangement, while the ESP32 is set up as a client. The two devices communicate with one another over a local Wi-Fi network. Upon initialization, the ESP32 autonomously establishes a connection with the Wi-Fi network created by the Raspberry Pi, transferring the preprocessed pictures to the Pi using Hypertext Transfer Protocol (HTTP). After capturing and preprocessing an image, the ESP32 packages the data into blocks and transmits them to the Raspberry Pi. The ESP32 also sends additional data, such as capture metadata (time, camera origin, etc.) required for processing on the Raspberry Pi.

The Raspberry Pi is responsible for heavy processing tasks like facial recognition and distance estimation. After receiving images from the ESP32, it analyses the data and sends help commands back to the ESP32. After receiving them, the Raspberry Pi decodes the images so facial recognition software can process them. A Convolutional Neural Network (CNN) algorithm previously trained for real-time face identification is used for this facial recognition. First, the Raspberry Pi runs a CNN algorithm to compare the detected faces with a database stored locally on the Raspberry Pi. If the face matches one in the database, an ID is generated. The system is flexible enough to accommodate various users and situations since it allows recognized faces to be added and changed. Following that processing, the system uses the two images the cameras took to create a stereo-vision image. By triangulating the coincident spots in the two photos, the algorithm determines the relative distance of the spotted object. This procedure is necessary to provide safe navigation and to notify the user of any close obstructions.

4.2.3. Voice Assistance System

After analyzing the images and generating the results, the Raspberry Pi microprocessor sends this data to the ESP32 board, which uses a sound playback module to inform the user of possible obstacle warnings or the identification of any faces. Depending on the face recognition result or adjacent hazard detection, the Raspberry Pi sends a text message to the

ESP32 board. This message can warn about a discovered impediment or identify a known individual. The ESP32 converts the text it receives into audio messages using an integrated voice synthesizer. The sound module plays back these messages, warning the user when potentially dangerous things are nearby or when it recognizes a face.

5. Results

Various settings and conditions were used for the experiments, with the main goals being power consumption, system usability, facial recognition accuracy, and system latency and integration with IoT devices. Furthermore, the system's behavior in real-world usage scenarios and its capacity for distance estimation was assessed. Although areas for development were found to maximize the system's performance in practical applications, the results collected show that the system is functional and effective.

5.1. Tests for Distance Estimation

The objective of this test was to evaluate the system's accuracy in estimating distances between the user and detected objects. This is especially relevant when face recognition and detection of nearby hazards are critical for providing real-time assistance to visually impaired people. Several tests were conducted indoors and outdoors to evaluate the feasibility of distance estimation in different environments. Using triangulation and stereo-vision techniques, the system used two cameras to measure the user's distance concerning the elements identified on his route. Each test involved measuring the distance to a reference object every one to five meters and contrasting it with the distance predicted by the system. Percent error was calculated to assess the accuracy of the estimate. Table 1 presents an overview of the data acquired, including the actual distances, the distances estimated by the system, and the percent inaccuracy of the estimated distances.

Table 1. Distance estimation

Real Distance (m)	Estimated Distance (m)	Absolute Error (m)	Relative Error (%)
0.5	0.52	0.02	4.0
1.0	0.97	0.03	3.0
1.5	1.46	0.04	2.7
2.0	1.94	0.06	3.0
2.5	2.42	0.08	3.2
3.0	2.89	0.11	3.7
3.5	3.41	0.09	2.6
4.0	3.80	0.20	5.0
4.5	4.26	0.24	5.6
5.0	4.65	0.35	7.0

The results of the distance estimation tests indicate that the system is suitable for estimating distances with a relatively low margin of error. Absolute and relative errors were also calculated for each measurement to have objective values to analyze. The proposed system showed high accuracy for short and medium distances, with a maximum distance of 3.5 m and an average relative error of 3.17 %. At distances above 4 meters, the error tends to increase slightly, reaching 7 % at 5 meters. These findings suggest that, although the system is reliable at short and medium distances, the triangulation algorithm may need further fine-tuning to increase the system's accuracy at longer distances. Scientific evidence suggests that adding additional sensors to the stereo vision system, such as LIDAR or ultrasonic sensors, is desirable for applications requiring high accuracy at long distances where cameras are insufficient.

5.2. Facial Recognition Results

The smart glasses face recognition algorithm proposed in this paper was subjected to a rigorous testing process under various environmental circumstances, most notably considering changes in environmental lighting. The facial data sets comprised variations in lighting, locations and facial accessories, e.g., glasses or hats. The system was evaluated using criteria of accuracy, sensitivity (true positive rate) and specificity (true negative rate). According to the results in Table 2, the average accuracy was 94.5 % in normal illumination environments and 85.3 % in very low-light environments. The false negative and false positive rates were 3.4 % and 2.1 %, respectively. These findings demonstrated that the proposed system maintains high accuracy despite being affected by adverse lighting conditions.

Table 2. Facial recognition results

Test Condition	Accuracy (%)	False Positive Rate (%)	False Negative Rate (%)
Normal Illumination	94.5	2.1	3.4
Low Illumination	85.3	4.6	10.1
Angle Change (±15°)	92.0	3.0	5.0
Use of Accessories (glasses, caps)	88.7	5.2	6.1
Outdoor Environment with Visual Noise	83.0	6.5	10.5

5.3. Real-Time Latency and Performance Testing

The study also evaluated the latency of the system. The duration between capturing an image by both cameras and generating the audio assistance for the users was measured in various environmental conditions.

Table 3. Latency results of the developed system

Test Condition	Average Latency (ms)	Maximum Latency (ms)	Minimum Latency (ms)	Standard Deviation (ms)
Stable Wi-Fi Network	250	270	230	12
Wi-Fi Network with Fluctuations	380	450	320	45
Image Processing in Normal Illumination	260	280	240	15
Image Processing in Low Illumination	310	340	290	25
With Distance Variation (>3m)	280	310	260	20

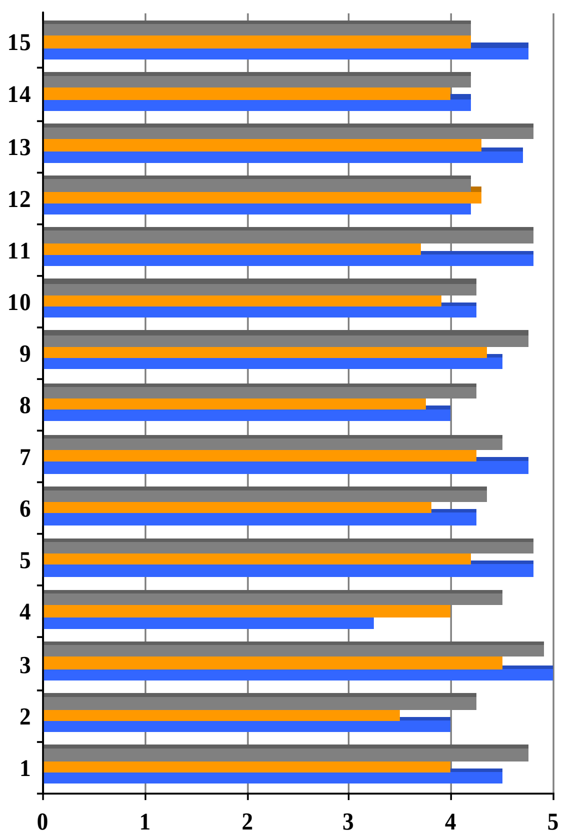

Fig. 6 Usability results of the proposed system

In stable network configurations, the tests showed a mean response time of 250 ms, which meets the processing requirements to be considered real-time (< 300 ms). However, in scenarios with fluctuations in network connectivity, increases in latency of up to 380 ms were observed, which could affect the user experience in critical and real test situations. Table 3 shows more details of the times employed by the algorithm as a function of the environment illumination and Wi-Fi network fluctuations.

5.4. Integrated System Results

Lastly, to verify the system's dependability, its IoT connectivity was examined. It was assessed whether the system could automatically rejoin while mimicking a Wi-Fi network outage. 95% of the time, the system rejoined rapidly with an average time of less than 5 seconds, demonstrating its strong resilience to network interruptions. Furthermore, the system's ability to integrate into a heterogeneous network of devices was confirmed by compatibility testing with other Internet of Things devices. Furthermore, 15 volunteers who agreed to be evaluated for these tests with visual impairment and aged 40 to 60 years (50 ± 7 years) used the device in various urban environments to provide the usability tests. On average, participants rated the utility and intelligibility of the aural aid 4.6 out of 5. Most participants praised the usefulness of facial recognition technology and warning alerts. Making the device smaller for convenience was one of the many aspects that may be improved. A depiction of the usability results is shown in Figure 6.

6. Conclusion

This study developed and implemented an assistive system solution based on smart glasses implemented with the Internet of Things (IoT) for wholly or partially blind people. The system quickly alerts users to known faces and obstacles in their environment by merging face recognition and distance estimation technologies using triangulation algorithms. After extensive testing, the system demonstrated high face recognition accuracy, reaching 94.5 %, and low latency, less than 300 ms, both essential for a better user experience and to achieve a real-time system. Data collected showed that the system maintains good accuracy in regulated environments and that low light levels do not affect performance. In addition, the successful integration of IoT connectivity allowed the glasses to efficiently connect to other platforms and devices to provide continuous and dynamic assistance. Finally, the usability of the developed device was analyzed, and the average satisfaction rating was 4.6 out of 5 points, indicating that the developed system is comfortable and easy to use. These elements that were the subject of this study are necessary to ensure a more reliable use in practical and everyday situations.

Future studies will focus on refining the facial recognition algorithm to better handle the challenges posed by the adverse lighting conditions that this detection system may encounter. This could involve integrating new technologies, including advanced image pre-processing techniques or using infrared cameras to improve detection in low light conditions. Additionally, this study found that to enhance system performance on embedded devices without sacrificing battery life, more effective deep learning algorithms that can be optimized on low-power devices still need to be implemented. These findings will need to be reviewed in the ongoing work.

Acknowledgments
The authors wish to express their sincere gratitude to the Universidad Nacional de San Agustín de Arequipa.

References
[1] Niek Zuidhof et al., "Defining Smart Glasses: A Rapid Review of State-of-the-Art Perspectives and Future Challenges from a Social Sciences' Perspective," *Augmented Human Research*, vol. 6, pp. 1-18, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] Dawon Kim, and Yosoon Choi, "Applications of Smart Glasses in Applied Sciences: A Systematic Review," *Applied Sciences*, vol. 11, no. 11, pp. 1-21, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[3] Xinyi Wang et al., "A Survey of Face Recognition," *arXiv preprint*, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] Guodong Guo, and Na Zhang, "A Survey on Deep Learning Based Face Recognition," *Computer Vision and Image Understanding*, vol. 189, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[5] Amir Masoud Rahmani et al., "The Internet of Things for Applications in Wearable Technology," *IEEE Access*, vol. 10, pp. 123579-123594, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[6] Naghmeh Niknejad et al., "A Comprehensive Overview of Smart Wearables: The State of the Art Literature, Recent Advances, and Future Challenges," *Engineering Applications of Artificial Intelligence*, vol. 90, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[7] Stefan Mitrasinovic., "Clinical and Surgical Applications of Smart Glasses," *Technology and Health Care*, vol. 23, no. 4, pp. 381-401, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[8] F. John Dian, Reza Vahidnia, and Alireza Rahmati, "Wearables and the Internet of Things (IoT), Applications, Opportunities, and Challenges: A Survey," *IEEE Access*, vol. 8, pp. 69200-69211, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[9] Constantino Álvarez Casado et al., "Face Detection and Recognition for Smart Glasses," *International Symposium on Consumer Electronics*, pp. 1-2, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[10] Wafa Mellouk, and Wahida Handouzi, "Facial Emotion Recognition Using Deep Learning: Review and Insights," *Procedia Computer Science*, vol. 175, pp. 689-694, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[11] Yassin Kortli et al., "Face Recognition Systems: A Survey," *Sensors*, vol. 20, no. 2, pp. 1-36, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] De Jong Yeong et al., "Sensor and Sensor Fusion Technology in Autonomous Vehicles: A Review," vol. 21, pp. 1-37, no. 6, 2021. [CrossRef] [Google Scholar] [Publisher Link]