Facial Authentication System: Bridging IoT and Web Architecture

Gabriel Alexandre Carvalho
Centro de Informática
Universidade Federal da Paraíba
João pessoa, PB - Brazil
gabriel.carvalho@ufpb.br

Laura Francine Araujo Silva
Centro de Informática
Universidade Federal da Paraíba
João pessoa, PB - Brazil
lfas2@academico.ufpb.br

Rosivaldo Lucas Da Silva
Centro de Informática
Universidade Federal da Paraíba
João pessoa, PB - Brazil
rosivaldosilva@eng.ci.ufpb.br

Renato dos Santos Monteiro
Centro de Informática
Universidade Federal da Paraíba
João pessoa, PB - Brazil
renato.santos@academico.ufpb.br

Verônica Maria Lima Silva
Departamento de sistemas de computação
Universidade Federal da Paraíba (UFPB)
João pessoa, PB - Brazil
veronica.lima@ci.ufpb.br

***Abstract*—This study introduces a web-based facial authentication system designed to tackle security challenges in digital access. Leveraging cutting-edge technologies and a modular architecture, the system seamlessly integrates IoT, AI, database management, and frontend components to enhance authentication processes. Evaluation results demonstrate a high success rate of 96%, affirming the system's effectiveness and practical suitability for real-world applications.**

Keywords—Facial authentication, web-based system, IoT, artificial intelligence, modular architecture.

I. INTRODUCTION

In today's digital age, the need for robust security measures to protect sensitive information is ever-present. Traditional authentication methods such as passwords are increasingly vulnerable to breaches and identity theft [2]. As a result, there's a growing demand for more secure and user-friendly authentication solutions. Facial recognition technology has emerged as a promising alternative, offering a unique combination of convenience and security. By utilizing distinct facial features for identity verification, facial authentication systems provide a more reliable and seamless user experience. Against this backdrop, this article introduces a novel web-based facial authentication system. This web system leverages cutting-edge technologies including the ESP32-CAM OV5640 module, artificial intelligence (AI), Internet of Things (IoT), and advanced storage solutions. By integrating these components, the web system aims to deliver an effective and user-friendly authentication experience while ensuring robust security measures are in place. Throughout this article, we'll delve into the architecture of the facial authentication web system, highlighting the roles of key modules such as IoT for communication, backend for data management, AI for facial recognition, and frontend for user interaction. Furthermore, we will discuss the implementation process, describing the integration of key technologies and components. This includes an analysis of the hardware setup with the ESP32-CAM OV5640 module, the configuration of AI models for facial recognition, and the development of backend and frontend modules using modern web technologies. Additionally, we will present a performance evaluation of the system, covering test scenarios such as different genders and different image dimensions. Through these evaluations, we aim to provide a comprehensive assessment of the system's accuracy and reliability in real-world applications. The results will be compared with established benchmarks and discussed in the context of existing research to highlight advancements and potential areas for future improvements. Figure 1 aims to illustrate the architecture of the application developed.

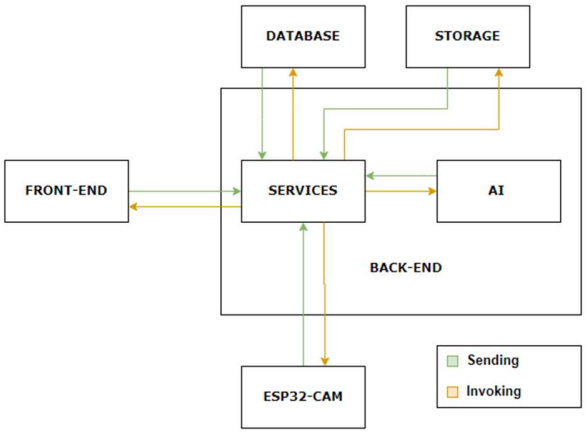

Fig.1 – Diagram Block Architecture

II. RELATED WORK

A. Authentication with Face Recognition and Sign Language using ESP32-CAM

Yalçın, Z., Türkdağlı, O., Dalkılıç, G., Aydın, Ö. (2023). This work explores the use of the ESP32-CAM for facial authentication combined with sign language recognition. The study emphasizes the integration of IoT devices with AI to enhance both security and accessibility. It details the implementation process of the ESP32-CAM for capturing facial images and processing them for recognition while also interpreting sign language gestures. The research demonstrates the effectiveness of such a system in providing a dual-mode authentication process that caters to both hearing and speech-impaired users, thus broadening the usability of facial recognition technologies in diverse environments [1].

B. Home Security System with Face Recognition based on Convolutional Neural Network

Irjanto, N. S., & Surantha, N. (2020). This study presents a comprehensive home security system leveraging convolutional neural networks (CNN) for facial recognition. It provides insights into how CNN models can be trained to accurately identify individuals in a home setting, enhancing the security by preventing unauthorized access. The paper outlines the architecture of the system, including the hardware components, the training process for the CNN, and the methods used to optimize recognition accuracy. The results show significant improvements in security and efficiency, making a strong case for the application of AI in residential security systems [2].

C. Robust Human Face Authentication Leveraging Acoustic Sensing on Smartphones

Zhou, T. B., Xie, Z., Zhang, Y., Lohokare, J., Gao, R., & Ye, F. (2022). This paper investigates a novel approach to facial authentication using acoustic sensing technology integrated into smartphones. The study explores how acoustic signals, in conjunction with traditional image-based facial recognition, can enhance the robustness of authentication processes. The methodology includes capturing facial images while simultaneously using acoustic signals to detect liveness and prevent spoofing attacks. This dual-sensing approach addresses common vulnerabilities in facial recognition systems, providing a more secure and reliable authentication method suitable for mobile devices [4].

D. Face Authentication With Makeup Changes

Guo, G., Wen, L., & Yan, S. (2014). This research addresses the challenge of facial authentication in the presence of makeup, which can significantly alter a person's appearance and affect recognition accuracy. The authors propose advanced algorithms capable of distinguishing between genuine facial features and alterations caused by makeup. The study involves detailed experimentation with various makeup styles and their impact on recognition systems. By enhancing the facial recognition models to account for these changes, the paper provides solutions that improve the robustness of authentication systems, making them more reliable in real-world scenarios where users may frequently change their appearance [5].

III. METHODOLOGY

A. Architecture

The system architecture integrates various components, including an ESP32-CAM OV5640 module, artificial intelligence (AI), frontend, backend, databases, IoT, and storage. Each module plays a crucial role in the functioning of the application, facilitating communication and data flow [1].

1. IoT Module: The IoT module serves as a bridge between the backend and the ESP32-CAM OV5640 board. It receives messages via MQTT from the backend, interprets these messages, and forwards the information back to the backend. Upon receiving a control signal from the backend, the ESP32-CAM captures a photo and preprocesses it, converting the image from a byte array to a base64 string.

2. Backend Module: The backend module is responsible for managing all application data. Utilizing the same MQTT broker as the ESP32-CAM, it listens for messages and processes images received in base64 string format. Upon receiving an image, the backend converts it to a file format and stores it in both the database and storage. Subsequently, it triggers the AI module to perform facial recognition on the captured image.

3. Databases Module: The databases module stores information related to system entities, including users, groups, user-group relationships, and logs. It serves as a repository for managing and retrieving data required by the application.

4. Storage Module: The storage module is tasked with storing images of users and images captured during authentication processes. It provides a scalable and accessible storage solution for the application's image data.

5. AI Module: The AI module is activated by the backend upon receiving an image captured by the ESP32-CAM. It compares the received image with stored images in the database, which were registered by authenticated users. Following the comparison, a control signal is sent back to the backend to validate the authentication of the received image.

6. Frontend Module: The frontend module presents application data and functionalities through a graphical user interface (GUI). It provides users with an intuitive interface for interacting with the system, viewing authentication results, and managing user-related tasks.

B. Database and Storage

The database modeling was structured as follows, comprising four tables. The "users" table stores user information such as name, email, phone number, password, date of birth, and photo. The "groups" table allows administrators to categorize their clients into different groups. It stores group information including name, creation date, and the associated administrator. The "user_group" table establishes a relationship between administrators and groups, representing the clients associated

with an administrator and a group. "Client" refers to a user registered within the web system by an authenticated administrator. Lastly, the "log" table logs information about the success or failure, and type of error encountered during the face verification process for clients. Regarding storage modeling, it consists of two buckets. The "users" bucket is designated for storing user profile images, while the "images" bucket is dedicated to storing client images. These buckets facilitate organized storage and retrieval of image data, enhancing system efficiency and scalability.

C. *Artificial Intelligence*

The face-api.js library is a JavaScript tool designed to streamline facial recognition integration into web applications. It offers a range of functionalities, including face detection, recognition, and expression analysis, directly accessible within web applications. This library finds utility across various applications, from photo editing to facial recognition-based security systems, simplifying the incorporation of advanced facial recognition. In our implementation, we followed a structured approach. Initially, the backend triggers the AI module by passing the image from ESP32-CAM as a parameter, which is referred to as the reference image, for processing the reference image is appropriately formatted. Subsequently, we utilized specific face detection models available within the face-api.js library. For face detection, our project implemented an SSD (Single Shot Multibox Detector) based on MobileNetV1. This model aims for high accuracy in detecting face bounding boxes, prioritizing accuracy over low inference time. Trained on the WIDERFACE dataset, it accurately computes face locations in images, returning bounding boxes along with probabilities for each face. Additionally, we employed 68 Point Face Landmark Detection Models, offering lightweight and fast detection of 68 key facial landmarks. These models, available in default and tiny sizes, utilize depthwise separable convolutions and densely connected blocks [5]. Trained on a dataset of face images labeled with 68 face landmark points, they facilitate precise facial landmark detection. Within these models, we utilized specific functions, including key facial point detection and canvas resizing for precise feature extraction. Comparison of key points extracted from the reference image with those from client images associated with authenticated users enabled authentication validation. Control signals were then generated to inform the backend of authentication success or failure. Upon validation, indicating that the face in the reference image matches the face stored in the database, the backend logs the authentication result in the database. This logging mechanism ensures the recording of authentication outcomes, facilitating traceability and accountability in the authentication process.

D. *IoT System*

The purpose of the IoT system is to capture photos, send and receive control signals to and from the web system.

The IoT system was implemented as follows: Initially, a code was developed using the Arduino IDE, utilizing C++ language, to establish MQTT connection with the predetermined broker. The AI-Thinker code example was employed to capture images using the ESP32-CAM module. Additional functions were implemented to send and receive control signals to and from the web system's backend. A function was created to convert and send the image as a base64 string, segmented into packets of 128 bytes, with a transmission rate of 10 milliseconds per packet to the web system.

This code is compiled and deployed onto the ESP32-CAM - OV5640 board, which is equipped with the ESP32S3-WROOM-1 chip. This board serves as a robust tool for developers, offering advanced image processing and wireless connectivity capabilities for integration into various projects. Using this camera module, images are captured, and control signals are transmitted via Wi-Fi to the web system.

E. *Web System*

For the implementation of the web system, the following languages and tools were utilized: Next.js, React.js, Tailwind CSS, Supabase, PostgreSQL, Docker, MQTT.js and face-api.js. The landing page of the web system provides users with the option to either log in or register on the web platform. If the user already has an account, they can simply log in; otherwise, they need to register. Upon registration, the user's information is saved in the "users" table of the database. After logging in or registering, the user is swiftly redirected to the dashboard page. The dashboard page consists of a client management table, where the administrator user can remove, enable, or ban a client from the group. Additionally, it features a timeline chart displaying the quantity of successful and error logs over time, a card summarizing log information for the current month, and a pie chart illustrating the percentage of error types occurring in the current month. The dashboard also includes a sidebar with topics redirecting to the following pages:

- Add Clients: This page enables the administrator user to add clients by filling in their details such as name, phone number, email, date of birth, and photo. The photo can be uploaded or taken using the ESP32-CAM. Upon submitting the form and saving the client information in the "users," "groups," "user-groups" tables, and the image in storage, the administrator user receives a toast notification informing them of the successful submission.
- Verify Authenticator: This page aims to verify the functioning of the authentication system by visually indicating whether facial recognition was successful. Initially, communication is established via MQTT between the ESP32-CAM and the backend broker. The administrator user can send a control signal to the ESP32-CAM to request facial recognition. Upon receiving the control signal, the ESP32-CAM captures an image and sends it to the backend in base64 string format, divided into packets of 128 bytes. These packets are received by the backend, aggregated, and converted into a file type. The image is then saved in the "images" bucket and compared with all client face images associated with the administrator user. The AI service is invoked for facial authentication via the face-api.js. The result is returned to the frontend,

displaying a toast notification indicating success or failure. In the case of success, a square is drawn around the identified client's face, also including the client's name.

IV. Results

To assess the performance and robustness of the facial authentication system, a series of structured experiments were conducted. The goal was to evaluate the system's accuracy under varying conditions of distance and lighting. The experiments were designed to encompass authentication at distances of 30cm, 40cm, 60cm, and 1m, as well as under good, medium, and low lighting conditions.

A total of 500 authentication tests were carried out. Each test involved the system capturing an image of a registered client using the ESP32-CAM, processing the image with the AI module (utilizing face-api.js), and attempting to match the image against stored profiles in the database. After being reviewed from previous journals, namely in research [2], a target accuracy rate of approximately 95% was set. The system achieved a notable precision rate of 96%, translating to 480 successful authentications out of the 500 tests conducted. This suggests that despite its cost-effectiveness, the designed facial authentication system maintains an accuracy rate meeting the initial goal. Thus, the intended objective of the endeavor has been accomplished.

Performance was less optimal at the close range of 30cm, primarily due to distortion and inadequate framing. The best results were achieved at 40cm and 60cm, while accuracy slightly dropped at 1m but remained acceptable. The system performed excellently under good lighting, remained robust in medium lighting with minor drops in precision, and showed a decline in accuracy under poor lighting, especially at 30cm.

In summary, the system demonstrated high accuracy under various operational conditions, with some limitations in close-range and low-light scenarios. This evaluation underscores the system's reliability and effectiveness for real-world applications, meeting stringent security and usability standards while identifying areas for potential improvement in handling extreme conditions.

V. Conclusion

In conclusion, the development of our web-based facial authentication system, integrated with the ESP32-CAM module, represents a significant milestone. The system's architecture integrates essential components including frontend, backend, databases, IoT, and AI modules, each playing a crucial role in its operational framework. Achieving an impressive accuracy rate of 96% through comprehensive testing surpasses our initial target of 95%, affirming the reliability and efficacy of our low-cost facial authentication solution across diverse use cases.

Looking forward, future efforts will focus on enhancing security measures to fortify the system against emerging threats. This includes further refinement of AI algorithms to improve facial recognition accuracy and robustness, implementation of advanced encryption techniques for secure data transmission and storage, and rigorous vulnerability testing to ensure resilience against potential attacks.

Additionally, to facilitate widespread deployment, our roadmap includes optimizing system scalability and performance. This involves streamlining database architectures for faster query processing, enhancing frontend interfaces for intuitive user interactions, and exploring cloud-based solutions for flexible deployment options. These advancements aim to bolster system reliability, scalability, and usability, paving the way for broader adoption in various real-world applications.

In summary, this project has successfully delivered an efficient and user-friendly facial authentication solution, leveraging accessible yet sophisticated technologies to address contemporary security challenges.

References

[1] Martin Clinton Tosima Manullang et al 2020 IOP Conf. Ser.: Earth Environ. Sci. 537 012021 YALÇIN, Z., TÜRKDAĞLI, O., DALKILIÇ, G., AYDIN, Ö. (2023). Authentication with face recognition and sign language using ESP32-CAM. DEUFMD, 25(74), 481-489.

[2] Irjanto, N. S., & Surantha, N. (2020). Home Security System with Face Recognition based on Convolutional Neural Network. International Journal of Advanced Computer Science and Applications (IJACSA), 11(11).

[3] Martin Clinton Tosima Manullang et al 2020 IOP Conf. Ser.: Earth Environ. Sci. 537 012021.

[4] Ti B. Zhou, Z. Xie, Y. Zhang, J. Lohokare, R. Gao and F. Ye, "Robust Human Face Authentication Leveraging Acoustic Sensing on Smartphones," in IEEE Transactions on Mobile Computing, vol. 21, no. 8, pp. 3009-3023, 1 Aug. 2022, doi: 10.1109/TMC.2020.3048659.

[5] G. Guo, L. Wen and S. Yan, "Face Authentication With Makeup Changes," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 24, no. 5, pp. 814-825, May 2014, doi: 10.1109/TCSVT.2013.2280076.