Towards an Intelligent Algorithm for Profile Authentication and Identification

Nuno Jerónimo
University of Minho Information Systems, Guimarães, Portugal

Filipe Portela [1]
University of Minho Algoritmi Center, Guimarães, Portugal
IOTECH – Innovation on Technology, Trofa, Portugal

Henrique Santos
University of Minho Algoritmi Center, Guimarães, Portugal

Abstract

In the context digital transformation, the necessity for secure and efficient virtual identity verification has become paramount. Traditional methods often fail to balance security, speed, and usability, leaving gaps in user authentication systems. This paper addresses the critical challenge of creating a virtual ID system that identifies a single profile with improved security, speed, and effectiveness. An innovative face recognition algorithm using dynamic content loading and deep learning techniques is proposed. The utilisation of OpenCV for face recognition and feature extraction, combined with advanced similarity calculation methods, the system achieves superior accuracy in profile authentication tasks. Extensive testing, including identical twin scenarios, demonstrates the robustness of the algorithm and its superiority over existing solutions such as Apple's Face ID. In 150 of the tests conducted with identical twins, the algorithm consistently achieved 100% recognition accuracy. This breakthrough in facial recognition technology promises to create a triple authentication system, which will solve the problem of false positives in terms of identifying and authenticating people. This paper integrates principles from Computer Intelligence and Chatbots, emphasizing the application of deep learning techniques in enhancing virtual identity verification systems. This research contributes to the broader discourse on improving authentication mechanisms in the digital age.

2012 ACM Subject Classification Security and privacy → Web application security

Keywords and phrases Facial recognition, Biometric analysis, Dynamic content loading, User authentication

Digital Object Identifier 10.4230/OASIcs.SLATE.2024.10

Funding This work has been supported by FCT – Fundação para a Ciência e Tecnologia within the R&D Units Project Scope: UIDB/00319/2020.

1 Introduction

In recent years, advancements in computer intelligence and the increase of chatbots have revolutionized how we interact with technology, particularly in the realm of secure online identification. The need for secure online identification methods has grown alongside the increasing use of online platforms for various purposes like social networking, shopping, and banking. This growing necessity has made the topic of "Intelligent Algorithm for Profile Identification" quite important. Such algorithms prevents fraud and protects personal data in the digital age [3].

[1] Corresponding Author

The development of algorithms that can accurately and swiftly identify individuals based on their digital activities poses a significant challenge. These algorithms need to analyze diverse data sources such as text, images, and network connections to make reliable identifications.

An intelligent algorithm for profile identification is a type of machine learning algorithm that learns from data to predict outcomes. These algorithms can identify individuals securely by recognizing patterns in their behavior and characteristics [7].

The use of advanced techniques like deep learning, including recurrent neural networks (RNNs) and convolutional neural networks (CNNs), is vital in this field, especially for image-based profile identification [12].

This project, conducted in collaboration with IOTech company, aims to develop innovative solutions for various applications. Intelligent algorithms are employed to enhance the accuracy and efficiency of systems in different domains. For instance, using profile identification for authentication in voting systems helps prevent fraud by verifying voters' identities through biometric data like fingerprints or iris scans.

The motivation behind this project is to explore and apply artificial intelligence, machine learning, blockchain, investigation, security, and natural language processing to create something original. The project aims to contribute to existing knowledge while addressing real-world challenges.

The document is structured into six sections: Introduction, Literature Review, Materials and Methods, Practical Work, Project Plan, and Conclusion. Each section serves a specific purpose, from outlining the project's goals and context to detailing the methodologies used and presenting progress made.

2 Background

This section is going to be introduce the relevant concepts for the development of this article. In this way, the article theme will be presented in conjunction with the state of art

2.1 Internet of Everything

The Internet of Things (IoT) revolutionizes connectivity by interlinking devices through the internet, enabling data sharing and automation across various sectors. With applications spanning healthcare, agriculture, manufacturing, and transportation, IoT promises efficiency gains and improved decision-making processes. For instance, in healthcare, IoT facilitates remote patient monitoring and emergency alerting [9]. However, alongside its potential benefits, the proliferation of IoT devices introduces significant privacy and security concerns. Cyber-attacks, data breaches, and unauthorized access pose threats to user data integrity and confidentiality. To address these challenges, robust security protocols, data encryption, and user awareness campaigns are imperative [9].

In conclusion, while IoT holds immense promise for industry transformation and efficiency enhancement, its adoption necessitates a concerted effort towards mitigating security risks. By implementing effective security measures, stakeholders can harness the full potential of IoT while safeguarding user privacy and data integrity.

2.2 Security

In today's interconnected digital landscape, safeguarding sensitive information has become paramount. Both the National Institute of Standards and Technology (NIST) and the International Organization for Standardization (ISO) stress the significance of robust security

measures in protecting assets from various threats. According to NIST [2], security information serves as the bedrock of an effective cybersecurity program, guiding the development, implementation, and maintenance of policies, procedures, and controls. Moreover, ISO defines security as the protection of assets against threats, underscoring the importance of identifying, classifying, and managing information to uphold its confidentiality, integrity, and availability. With the proliferation of intelligent algorithms for profile identification, which leverage personal data, ensuring the security of such information is paramount to prevent unauthorized access, use, or disclosure. By adhering to established frameworks and standards, organizations can proactively mitigate risks to their systems and data, fortifying their defenses in an ever-evolving digital landscape

2.3 Artificial Intelligence (AI)

Artificial Intelligence (AI) stands at the forefront of computer science, revolutionizing tasks traditionally reserved for human intelligence. From speech recognition to medical diagnosis, its applications are vast and evolving. At the core of AI lies machine learning, a paradigm enabling systems to learn and adapt through data analysis, thereby driving progress in various domains [11].

In the domain of profile identification algorithms, AI techniques offer profound advantages. By leveraging machine learning, algorithms can sift through vast datasets, discerning patterns and making predictions with increasing accuracy over time. This capability finds resonance in combating fraudulent profiles, where AI-driven algorithms excel in identifying aberrant behaviors or suspicious patterns.

Natural Language Processing (NLP), another component of AI, further empowers profile identification algorithms. By parsing and comprehending textual content within user profiles, NLP enables algorithms to discern common interests, personality traits, and linguistic nuances. This proficiency not only aids in connecting like-minded users but also serves as a bulwark against automated accounts, detecting linguistic cues indicative of non-human interactions [8]. Recent advancements in AI, particularly deep learning architectures and enhanced data processing capabilities, have propelled profile identification algorithms to new heights. These algorithms can now discern intricate patterns and subtle anomalies with unprecedented precision, bolstering their efficacy in serving diverse platforms and user bases [6].

In conclusion, the integration of AI techniques into profile identification algorithms holds immense promise for enhancing accuracy and effectiveness. By continuously learning and adapting, these algorithms offer invaluable insights into user behavior, fostering safer and more engaging online environments.

2.4 Related Works

Before attempting to answer these questions there is no logic other than investing what has already been Intended by others and how it approaches a solution for the entire problem. In recent years, advancements in computer vision and machine learning have propelled the development of various applications related to facial recognition, augmented reality, and remote collaboration.

2.4.1 Facial Recognition and Authentication Systems

Facial recognition systems are widely used in security, surveillance, and user authentication, with recent advancements aiming to enhance accuracy, robustness, and privacy. A novel deep learning-based facial recognition algorithm achieved state-of-the-art accuracy on benchmark

datasets while mitigating concerns about bias and fairness [13]. Furthermore, proposed a privacy-preserving facial authentication framework using federated learning techniques. This framework enables multiple devices to collaboratively train a facial recognition model without sharing sensitive user data, thereby bolstering privacy protection in authentication systems [1].

2.4.2 Secure and Efficient Communication Protocols

Recent advancements in Augmented Reality (AR) have revolutionized remote collaboration, enabling users to seamlessly interact with virtual elements within their real-world environments. Research conducted by Gabriel Rosenberg has investigated the incorporation of AR functionalities into video conferencing platforms. This integration allows participants to overlay digital annotations, 3D models, and spatial cues during remote meetings, creating an immersive collaboration environment. Such enhancements greatly facilitate communication and foster complex discussions among geographically dispersed teams [4]

3 Materials and Methods

The purpose of this section is to disclose the project's research, identification, creation, development, and documentation methods.

3.1 Design Science Research (DSR)

Design Science Research (DSR) is a dynamic research methodology that tackles real-world challenges across diverse sectors including information systems, engineering, healthcare, and business. It revolves around an iterative process of crafting and assessing innovative solutions to generate novel insights and tangible outcomes.

The DSR process comprises six steps. First, researchers identify problems or possibilities within a certain field by conducting in-depth literature studies, stakeholder consultations, and empirical analysis. This initial stage aims to pinpoint and outline the most significant issues in the landscape [10]. This step can be found on section 1 (Introduction)

Next, the objectives of a solution are defined, ensuring they are measurable, achievable, and relevant [10]. With a clear understanding of the problem, researchers proceed to design and develop novel solutions or artifacts, drawing upon concepts from various fields to create treatments that effectively address identified difficulties [5]. The objectives are also defined on section 1 (Introduction)

The designed solutions then enter the demonstration stage, where prototypes or demonstrations showcasing the proposed solutions are created. Through iterative refinement and feedback from stakeholders and subject matter experts, researchers enhance their designs [5]. Creation of the Solution Architecture. This information can be found in section 4.1 (Architecture)

Demonstration artifact serves as a tangible representation of the proposed solution, allowing researchers to illustrate its functionality and potential impact. Demonstration artifacts can take various forms, such as software prototypes, physical models, or detailed process diagrams, depending on the nature of the solution. By engaging with these artifacts, stakeholders and experts can provide valuable feedback, which drives further refinement and validation of the research findings [10]. The demonstration artifact is presented on section 5.2 (Proof of Concept)

Rigorous evaluation forms the core of DSR, where scientists methodically judge the viability and effectiveness of their designed objects. This step often involves user testing, simulations, and case studies, along with quantitative and qualitative assessments [10]. This artifact can be found on section 5.3 (Discussion)

Finally, effective communication of research findings is crucial for maximizing impact and fostering knowledge dissemination. Researchers employ various channels such as academic publications, conferences, and industry forums to share their insights with the broader community [5]. On this paper the communication is accomplished through the dissertation document and articles.

3.2 Materials

This section focus on technologies and algorithms used to develop this project.

(a) **Machine Learning Algorithms:** Algorithms for machine learning can be used to examine user data and spot trends that point to a user's profile. For this, machine learning algorithms like decision trees, random forests, and neural networks can be employed. The Face Cascade object uses Haar Cascade classifiers and the Euclidean Distance function computes distance between two images' pixel values. Both algorithms are a type of machine learning-based object detection method.

(b) **Natural Language Processing (NLP):** NLP approaches can be applied to the analysis of user-generated content, including text, photos, and videos, in order to find patterns that may be used to determine the profile of the user. Among the NLP techniques are named entity recognition, topic modeling, and sentiment analysis. Email content generation, indirectly related to NLP, involves text processing tasks typically associated with natural language processing.

(c) **Deep Learning:** Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) are two deep learning approaches, that can be used to evaluate user data and find patterns that may be used to determine a user's profile. Deep learning methods come in handy especially when working with big datasets and intricate data structures. The augment image function uses deep learning techniques for image augmentation, enhancing model generalization in computer vision tasks.

In the section 4 it is presented with more detail the technologies and the algorithms used in the application.

4 Solution Architecture and Technologies

The objective of this section is to offer an understanding of the advancements achieved and the approaches used. There will be a comprehensive overview of the suggested system before going into the specifics of the work that has been done. The architecture of the project will be described in this overview, along with how its various parts work together to solve the problem as a whole.

4.1 Architecture

The system architecture is built on the Flask web framework and makes use of a number of tools and packages, including Flask-Mail for email communication, Google Cloud services for storage and authentication, and OpenCV for image processing. The architecture is based on a client-server paradigm, in which the server manages data and processes requests, and the

client uses a user interface to interact with the web application. The solution architecture is shown in detail in the figure that is presented in Figure 1. Five levels are displayed: Front-end, Back-end, Communication, Additional Components, and Development Workflow.

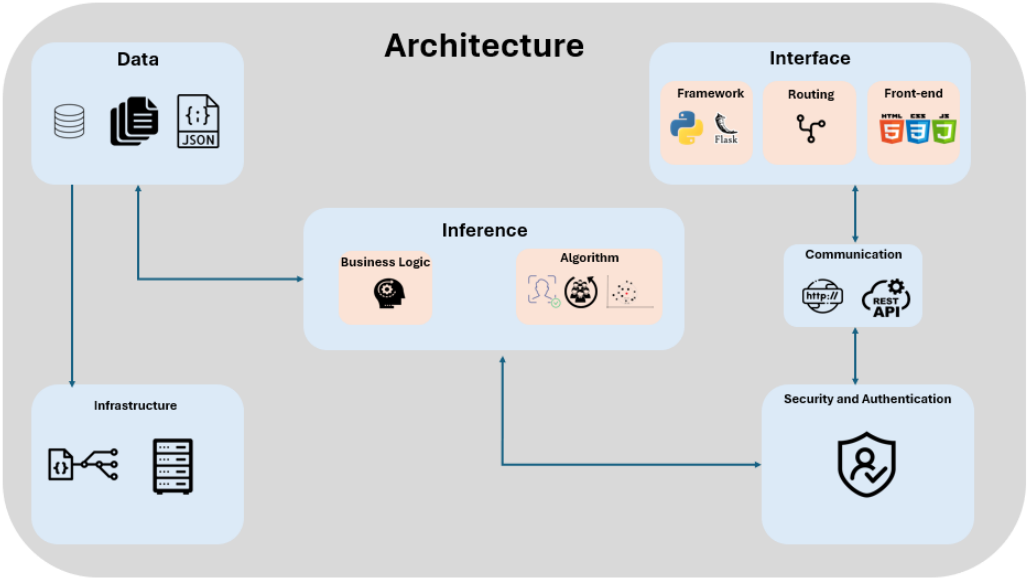

Figure 1 Solution Architecture.

4.1.1 Data

The Data Layer encompasses the management and storage of structured information within the application. It includes a relational database for structured data storage and JSON format for efficient data interchange between components. The database ensures persistent storage and retrieval of application data, while JSON facilitates lightweight and standardized communication between the server-side components and the Inference layer.

4.1.2 Inference

The Inference component encapsulates the core business logic of the application. This layer includes algorithms and procedures that drive decision-making and data manipulation based on specific requirements. For instance, it employs advanced algorithms like face recognition and data augmentation to enhance image processing capabilities.

4.1.3 Security and Authentication

Authentication mechanisms are crucial for securing the application and its resources. They encompass various security measures implemented to protect user data and ensure authorized access. Techniques like HTTPS encryption, CORS policies, and input validation mechanisms safeguard against malicious attacks and unauthorized access attempts. Authentication mechanisms are integrated seamlessly into the application to maintain data integrity and user confidentiality.

4.1.4 Communication

Communication protocols, primarily HTTP and APIs, facilitate seamless interaction between different layers of the application. HTTP protocols govern the exchange of data between the client-side user interface and the server-side back-end operations, adhering to RESTful principles for structured and efficient communication. JSON format plays a pivotal role in data serialization and deserialization, ensuring compatibility and coherence across distributed systems.

4.1.5 Interface

The Interface Layer serves as the bridge between the application's back-end functionality and the user-facing front-end. Implemented with Python Flask, this layer handles incoming HTTP requests from clients and coordinates data processing, database interactions, and response generation. It defines endpoints that specify the available operations (e.g., user authentication, data retrieval) accessible via HTTP APIs. This layer also incorporates business logic modules responsible for executing application-specific algorithms, such as face recognition using OpenCV and Euclidean distance calculations.

4.1.6 Infrastructure

Infrastructure components encompass the foundational elements that support the application's development, deployment, and operational processes. Git, a robust version control system, enables collaborative code management and facilitates efficient branching strategies. Server infrastructure provides a reliable platform for hosting the application, ensuring scalability, availability, and cost-efficiency. The integration of these components forms a resilient infrastructure backbone essential for the application's seamless operation and evolution.

4.2 Technologies

Listed here are the technologies previously examined within the project's scope, all slated for integration throughout the solution's developmental phase

4.2.1 Tools and Libraries

This section redirects its focus towards the foundational software technologies that have been carefully selected to drive the advancement and functionality of the project. Each technology plays a pivotal role in realizing the vision of the solution, encompassing tasks such as enabling robust back-end operations and promoting smooth integration with external services.

(a) **Python:** Programming language used for back-end development.
(b) **OpenCV:** OpenCV (Open Source Computer Vision Library) is a popular open-source computer vision and machine learning software library. It's used for various image and video processing tasks such as face detection, image augmentation, and video capturing.
(c) **Flask:** Flask is a micro web framework for Python used to develop web applications. In this code, Flask is utilized to create web endpoints for serving video frames, handling HTTP requests, and rendering HTML templates.
(d) **Google-Auth:** Library for authenticating with Google services.
(e) **Google Cloud Storage:** The code interacts with Google Cloud Storage, which is a cloud storage service provided by Google Cloud Platform, to store and retrieve data. 10

(f) **NumPy:** Library used for numerical computing, particularly for handling arrays in image processing.
(g) **Base64:** This module provides functions for encoding binary data to ASCII characters and decoding from ASCII characters back to binary data.
(h) **Requests:** Library for making HTTP requests, used for accessing external APIs (e.g., ipinfo.io).
(i) **Json:** Standard library for JSON manipulation.
(j) **Traceback:** This module provides functions for printing stack traces of Python programs.
(k) **Scipy:** Library for scientific computing, used for calculating Euclidean distance.
(l) **Uuid:** This module provides functions for generating universally unique identifiers (UUIDs).

4.2.2 APIs/Services

This section explores the pivotal role of APIs (Application Programming Interfaces) and services within the project's ecosystem. Each API and service has been carefully selected to enhance functionality and improve the user experience, serving as integral components in achieving the project's overarching objectives. From enabling smooth data storage and retrieval to leveraging external sources for valuable insights, these APIs and services are essential in shaping the project's capabilities and expanding its scope.

(a) **Google Cloud Storage:** Used for storing data, possibly including images and meeting information.
(b) **Ipinfo.io:** API used for getting location information based on IP address.
(c) **SMTP (Simple Mail Transfer Protocol):** Protocol used for sending emails, likely through Gmail's SMTP server.

4.2.3 Algorithms

Innovative technological solutions rely heavily on algorithms, which intricately intertwine functionality and intelligence. This section delves into the algorithmic landscape, highlighting the role of designed algorithms in defining the capabilities and intelligence of a project. Each algorithm plays a vital role in enabling the project to perceive, analyze, and intelligently respond to various simulations.

(a) **Face Detection (Haar Cascade):** The code uses the Haar Cascade classifier provided by OpenCV for detecting faces in images and video frames.
(b) **Image Augmentation:** An image augmentation function is implemented to apply various transformations to images, such as rotation, noise addition, brightness adjustment, contrast adjustment, translation, scaling, flipping, and shearing. These augmentations are applied randomly to generate diverse image data for training machine learning models.
(c) **Euclidean Distance Calculation:** The Euclidean distance algorithm is utilized to measure the similarity between two images by comparing their pixel intensities

5 Application Walkthrough

This section presents a detailed exposition of the application's functionality, elucidating the utilization of image processing and augmentation methodologies to bolster the system's efficacy. The endeavor to construct a web-centric application geared towards attendance

tracking employing facial recognition technology. The system facilitates user authentication and facial registration.

(a) **User Authentication:** Users can log in using their credentials, and their identity is verified through facial recognition.
(b) **Facial Registration:** Users can register their faces by capturing images through the device's camera. These images are stored securely for future authentication.
(c) **Dynamic Page Loading:** The application dynamically loads content.
(d) **Error Handling:** The system includes error handling mechanisms to manage cases where no face is detected, or errors occur during the authentication process.
(e) **Device Registration:** Users' devices are registered securely to ensure access control and prevent unauthorized access.

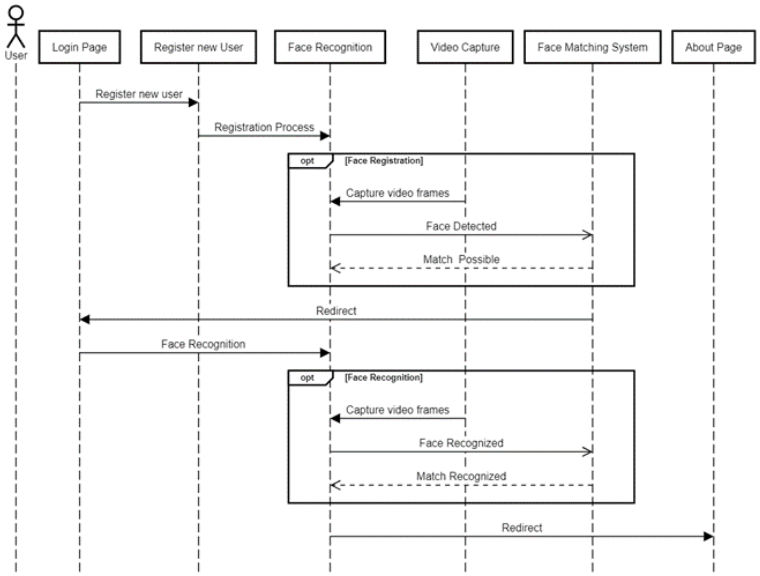

Figure 2 Sequence Diagram.

Facial recognition is essential in identifying and correlating faces within images, with the codebase relying on the OpenCV library, particularly the cv2 module, for image processing operations. The Haar Cascade classifier, referred to as facecascade, is pivotal for detecting faces within images. Upon receiving an image, preprocessing involves decoding from base64 format and conversion into a NumPy array for further processing. Validation checks ensure the image's integrity and non-emptiness. The detectMultiScale() function from OpenCV, utilizing the Haar Cascade classifier, detects faces and generates bounding boxes around them. After successful detection, the code matches faces against known images stored in the database. This involves traversing each user's folder, loading known images, and computing the Euclidean distance between facial regions of the input and known images. Normalizing the distance by pixel count enables meaningful comparison.

The image with the smallest distance below a predefined threshold is designated as the best match.The code extends comparison to augmented versions of known images, housed in the augmented images folder within each user's directory. Augmentation involves transformations like rotation, scaling, and noise addition, broadening the comparison spectrum. The same distance computation methodology is applied to augmented images to discern the optimal match.

Figure 2 is a sequence diagram that explains this process in more detail.

5.1 Algorithm Pseudocode

The pseudocode outlines a program's main function aimed at implementing a face recognition system with a parameter of trust, denoted as TRUST, which determines the precision level of the recognition process. The system also prompts for a username, denoted as USERNAME, to register a new user.

Listing 1 Main Function.

```
function main()
    input TRUST
    input USERNAME
    registerNewUSer(TRUST, USERNAME)
    about(TRUST, USERNAME)
end function
```

The main() function begins by accepting inputs for TRUST and USERNAME. Then, it calls two functions: registerNewUser() and about().

(a) **registerNewUser(TRUST, USERNAME):** This function registers a new user with the specified trust level and username. The trust level parameter is crucial as it influences the system's precision in recognizing the user's face.

(b) **about(TRUST, USERNAME):** This function provides information or details about the registered user, potentially including their trust level and username.

Listing 2 Augmentation Image Tuple.

```
augment_image = {
    "rotation":{"angle_range":(-15,15)},
    "noise":{"std_dev_range":(10,30)},
    "brightness":{"alpha_range":(0.7,1.3),"beta_range":(-30,30)},
    "contrast":{"alpha_range":(0.7,1.3),"beta_range":(-30,30)},
    "translation":{"x_range":(-20,20),"y_range":(-20,20)},
    "scaling":{"scale_range":(0.8,1.2)},
    "flipping":{"flip_horizontal":True,"flip_vertical":False},
    "shearing":{"shear_range":(-0.2, 0.2)},
    }
```

The augmentation tuple contains various augmentation techniques along with their control parameters. Each key represents an augmentation technique, and its corresponding value is another dictionary containing control parameters for that technique.

Listing 3 Face Recognition Function.

```
function faceRecognition(TRUST, USERNAME)
    if TRUST < 10 then
    else
        main()
    end If
end function
```

The provided pseudocode outlines a basic function for face recognition, with parameters for trust level (TRUST) and username (USERNAME). The function evaluates the trust level, and if it's less than 10, it doesn't proceed further. However, if the trust level is 10 or higher, it calls the main function for face recognition.

Listing 4 About Function.

```
function about(TRUST, USERNAME)
    output "About Page"
    input STATE
```

```
    if STATE = True then
        meeting(TRUST, USERNAME)
    else
        output "Votation page"
        votation(TRUST, USERNAME)
    end If
end function
```

The function starts by outputting the "About Page" and then takes input for the STATE parameter. If the STATE is true, indicating a positive condition, the function calls the meeting() function with the TRUST and USERNAME parameters. This suggests that users with a high level of trust are directed to a meeting page, presumably for more detailed information or interaction.

5.2 Proof of Concept

The primary objective of the Proof of Concept is to verify the practicality and operational efficacy of the system within real-world contexts. Identical twins present a unique challenge for facial recognition technology due to the highly similar facial features exhibited by these individuals. The objective was to ascertain whether the system could accurately distinguish between twins under a range of conditions.

To achieve this, several scenarios were tested. Here's a detailed breakdown of each scenario and the testing process:

(a) **Scenario:** Twin 1 has an account, Twin 2 does not.
   (a) **Objective:** To verify that the system can correctly recognise Twin 1 and reject Twin 2.
   (b) **Procedure:** Twin 1 attempted to access the system while Twin 2 also attempted to access the system without an account.
   (c) **Testing:** This scenario was tested 50 times with different sun exposures, locations, and face proximity's, to ensure consistent results.

(b) **Scenario:** Twin 2 has an account, Twin 1 does not.
   (a) **Objective:** To check that the system can correctly recognise twin 2 and reject twin 1.
   (b) **Procedure:** Twin 2 attempted access while Twin 1 also attempted to access the system without an account.
   (c) **Testing:** This scenario was tested 50 times with different sun exposures, locations, and face proximity's, to ensure consistent results.

(c) **Scenario:** Both twins have accounts.
   (a) **Objective:** To evaluate the system's ability to correctly identify each twin when presented individually.
   (b) **Procedure:** Both Twin 1 and Twin 2 had accounts and attempted to access them individually.
   (c) **Testing:** This scenario was tested 50 times with different sun exposures, locations, and face proximity's, to ensure consistent results.

Each scenario was designed and tested to assess the accuracy and reliability of the system in distinguishing between the twins under different conditions. Table 1 provides a comprehensive he results of tests conducted to evaluate the recognition accuracy of an unspecified algorithm and Apple's FaceID in distinguishing between identical twins. The tests involve three different scenarios where one or both twins have accounts set up.

Table 1 Use Case 1.

ID	Case Description	Results			
		Algorithm		Apple (Face ID)	
1	Rule: Twin 1 have account Tested 50 times	Recognized	Not Recognized	Recognized	Not Recognized
	Twin 1	X		X	
	Twin 2		X	X	
	Accuracy	100%		0%	
2	Rule: Twin 2 have account Tested 50 times	Recognized	Not Recognized	Recognized	Not Recognized
	Twin 1		X	X	
	Twin 2	X		X	
	Accuracy	100%		0%	
3	Rule: Twin 1 and twin 2 have account Tested 50 times	Recognized	Not Recognized	Recognized	Not Recognized
	Twin 1	X		X	
	Twin 2	X		X	
	Accuracy	100%		0%	

5.3 Discussion

This proof of concept assesses the efficacy of a facial recognition system in discerning between twins for authentication purposes. Although tested across a number of scenarios, the most notable aspect of the system is its ability to differentiate between twins. Three scenarios were tested: Individual accounts for each twin and a shared account. Both systems received the same images. Results demonstrate the system's high accuracy in differentiating between twins, even outperforming Apple's Face ID in certain scenarios. These findings underscore the potential of custom facial recognition systems in addressing the unique challenges posed by identical twins.

Methodology. The proof of concept involved testing the facial recognition system under three distinct scenarios:

(a) **Scenario 1 - Single Account for Twin 1:** Twin 1 possessed an account in the system, while Twin 2 did not. The system underwent 50 trials to assess its capability in correctly recognizing Twin 1 while rejecting Twin 2.

(b) **Scenario 2 - Single Account for Twin 2:** Similar to Scenario 1, but with roles reversed. Twin 2 had the account, while Twin 1 did not. The system was tested 50 times.

(c) **Scenario 3 - Accounts for Both Twins:** Both Twin 1 and Twin 2 had accounts in the system. The system underwent 50 trials to evaluate its ability to correctly identify each twin when presented individually.

Results.

(a) **Scenario 1:** The system successfully recognized Twin 1 while rejecting Twin 2 in all 50 trials, indicating a high accuracy rate.

(b) **Scenario 2:** Similarly, the system correctly identified Twin 2 while rejecting Twin 1 in all 50 trials, reaffirming its effectiveness in twin authentication.

(c) **Scenario 3:** In all 50 tests, the system accurately recognized both Twin 1 and Twin 2 when presented individually, suggesting robust differentiation capabilities even in scenarios where both twins have accounts.

Comparison with Apple Face ID. Apple's Face ID feature was tested alongside the custom facial recognition system. In all scenarios, Face ID successfully recognized both twins, indicating its limitations in distinguishing between them compared to the custom system.

The proof of concept demonstrates the viability of using this algorithm for accurately distinguishing between identical twins. The system exhibited high accuracy rates, even surpassing commercial solutions like Apple's Face ID in certain scenarios. Further research and development in this area could lead to the implementation of such systems in real-world applications, enhancing security and personalization while addressing the unique challenge posed by identical twins.

6 Conclusion

In conclusion, the developed facial recognition algorithm, featuring a triple authentication mechanism, exhibits robust performance in differentiating between identical twins. This conclusion is based on comprehensive testing across multiple scenarios. However, the generalizability of these results is somewhat limited due to the relatively small number of different profiles tested.

In Scenario 1 and Scenario 2, where only one twin had an account in the system while the other did not, the algorithm achieved a perfect recognition rate with 100% accuracy. In all 50 tests, it accurately identified the enrolled twin while rejecting the unenrolled one. This illustrates the algorithm's high degree of accuracy and reliability in the authentication of individual twins.

Furthermore, in Scenario 3, where both twins had accounts, the algorithm demonstrated its capacity to correctly identify each twin independently in all 50 tests with 100% accuracy. This further emphasizes the algorithm's effectiveness in distinguishing between twins even when they both have registered accounts, thereby demonstrating its robustness in real-world scenarios.

In comparison, the algorithm demonstrated superior performance in twin authentication tasks, whereas Apple's Face ID feature exhibited limitations in differentiating between the twins. The custom facial recognition system, on the other hand, exhibited consistent accuracy across all scenarios.

In conclusion, the results validate the efficacy of the developed facial recognition algorithm with its triple authentication mechanism in accurately identifying individuals in complex scenarios. This establishes its potential for various real-world applications, including security systems, access control, and personalized user experiences. Further optimization and refinement of the algorithm could enhance its performance, extend its utility across diverse domains and extend the tests to more people and profile types.

References

1 Youssif Abuzied, Mohamed Ghanem, Fadi Dawoud, Habiba Gamal, Eslam Soliman, Hossam Sharara, and Tamer Elbatt. A privacy-preserving federated learning framework for blockchain networks. *Cluster Computing*, pages 1–18, March 2024. doi:10.1007/s10586-024-04273-1.

2 Matthew Barrett. Framework for improving critical infrastructure cybersecurity version 1.1, 2018-04-16 2018. doi:10.6028/NIST.CSWP.04162018.

3 Kaikai Deng, Ling Xing, Longshui Zheng, Honghai Wu, Ping Xie, and Feifei Gao. A user identification algorithm based on user behavior analysis in social networks. *IEEE Access*, 7:47114–47123, January 2019. doi:10.1109/ACCESS.2019.2909089.

4 Sanjay Dhurandher, Jagdeep Singh, Petros Nicopolitidis, Raghav Kumar, and Geetanshu Gupta. A blockchain-based secure routing protocol for opportunistic networks. *Journal*

of Ambient Intelligence and Humanized Computing, 13:1–13, April 2022. doi:10.1007/s12652-021-02981-9.

5 Shirley Gregor and Alan Hevner. Positioning and presenting design science research for maximum impact. *MIS Quarterly*, 37:337–356, June 2013. doi:10.25300/MISQ/2013/37.2.01.

6 Muskan Khan. Advancements in artificial intelligence: Deep learning and meta-analysis, July 2023. doi:10.31219/osf.io/twyfh.

7 Hongyu Liu and Bo Lang. Machine learning and deep learning methods for intrusion detection systems: A survey. *Applied Sciences*, 9(20), 2019. doi:10.3390/app9204396.

8 Bill Manaris. Natural language processing: A human-computer interaction perspective. *Advances in Computers*, 47:1–66, December 1998. doi:10.1016/S0065-2458(08)60665-8.

9 Mark Ogonji, George Okeyo, and Joseph Wafula. A survey on privacy and security of internet of things. *Computer Science Review*, 38:100312, November 2020. doi:10.1016/j.cosrev.2020.100312.

10 Ken Peffers, Tuure Tuunanen, Marcus Rothenberger, and S. Chatterjee. A design science research methodology for information systems research. *Journal of Management Information Systems*, 24:45–77, January 2007.

11 Soorya Ramdas and Neenu Agnes. Leveraging machine learning for fraudulent social media profile detection. *Cybernetics and Information Technologies*, 24:118–136, March 2024. doi:10.2478/cait-2024-0007.

12 Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification, 2014. arXiv:1406.4773.

13 Zitong Yu, Yunxiao Qin, Xiaobai Li, Chenxu Zhao, Zhen Lei, and Guoying Zhao. Deep learning for face anti-spoofing: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5609–5631, 2023. doi:10.1109/TPAMI.2022.3215850.