iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: III Month of publication: March 2025

DOI: https://doi.org/10.22214/ijraset.2025.67644

Robust Fingerprint Verification-Using Real Time Security Alerts

P S L Sravani[1], D Sindhuja Reddy[2], K Udvisha[3], D. Sarika[4], B. Mohan Sai[5]
[1]*Assistant Professor, Dept of CSE, Raghu Engineering College*
[2, 3, 4, 5]*Dept of CSE, Raghu Institute of Technology*

Abstract: This work presents Robust Fingerprint Authentication with Real-Time Security Alerts, a biometric authentication system that integrates Aadhaar-based identity verification, liveness detection, and real-time SMS notifications. Traditional fingerprint authentication systems store biometric data persistently, increasing the risk of breaches and unauthorized access. To address this, the proposed system ensures immediate deletion of fingerprint data after verification, enhancing security and user privacy. Advanced image processing techniques and deep learning models are employed for liveness detection, mitigating spoofing attacks using fake fingerprints.

An OTP-based authentication mechanism adds an additional security layer, while real-time SMS notifications inform users of authentication attempts, improving transparency. The system is implemented using Flask, PostgreSQL, OpenCV, Twilio, and TensorFlow, ensuring scalability, efficiency, and secure data handling.

HTTPS encryption and compliance with GDPR and CCPA regulations further strengthen security by enforcing responsible biometric data processing.

Experimental results demonstrate 98.5% fingerprint matching accuracy, 97.8% liveness detection success, and 99.3% OTP verification accuracy, surpassing traditional fingerprint authentication models. By integrating multi-layered security features and real-time user alerts, this system provides a privacy-conscious, secure, and efficient biometric authentication solution suitable for applications in banking, government services, and enterprise security.

Index Terms: Biometric authentication, fingerprint recognition, liveness detection, OTP verification, security alerts

I. INTRODUCTION

In the digital era, secure and reliable identity verification is essential, as traditional authentication methods such as passwords and PINs are increasingly susceptible to breaches. Cyberattacks, phishing schemes, and credential leaks highlight the vulnerabilities of conventional approaches, necessitating the adoption of more secure, user-friendly, and efficient alternatives. Biometric authentication, particularly fingerprint recognition, has gained prominence due to the uniqueness and permanence of fingerprints. However, existing fingerprint authentication systems present several challenges, including persistent biometric data storage, susceptibility to spoofing attacks, and a lack of real-time authentication alerts. This work proposes an advanced fingerprint authentication system that integrates Aadhaar-based identity verification, liveness detection, OTP authentication, and real-time SMS notifications to enhance security, transparency, and privacy. Unlike conventional systems that indefinitely store fingerprint data, the proposed approach ensures immediate deletion of fingerprint data post-authentication, reducing the risk of unauthorized access and data breaches. Deep learning-based liveness detection differentiates between genuine and spoofed fingerprints, mitigating security threats. Additionally, real-time SMS notifications, enabled via Twilio, provide users with authentication attempt alerts, improving awareness and security.

A. Objective

This paper introduces Fingerprint Authentication System with Aadhaar Integration enhances security, privacy, and transparency through Aadhaar-based identity verification, real-time SMS alerts, and immediate data deletion post-verification. Liveness detection prevents spoofing using advanced image processing and deep learning. Aadhaar verification strengthens identity validation, reducing impersonation risks.

Immediate data deletion ensures privacy compliance with GDPR and CCPA, eliminating data breach risks. Real-time SMS alerts notify users of authentication attempts, enhancing security awareness. The system follows strict data protection measures, including HTTPS encryption and user consent, ensuring a secure and privacy-focused authentication process.

B. Problem Statement

The primary challenge in fingerprint authentication systems is data security and user privacy. Most existing systems store fingerprint data permanently, increasing the risk of unauthorized access and misuse. Additionally, many systems do not incorporate liveness detection, making them susceptible to spoofing attacks using fake fingerprints. Furthermore, a lack of real-time alerts prevents users from being notified of suspicious authentication attempts, leaving their accounts exposed to potential breaches.

II. EXISTING SYSTEM

The existing fingerprint authentication systems primarily utilize traditional biometric verification methods, relying on local or centralized databases to store fingerprint data persistently. These systems generally do not integrate Aadhaar-based identity verification or real-time SMS notifications, and their data management practices often result in security vulnerabilities. Furthermore, the absence of immediate data deletion after verification compromises user privacy. Existing systems frequently lack advanced features such as liveness detection, making them susceptible to spoofing attacks, where counterfeit fingerprints are used to bypass authentication. Security protocols like end-to-end encryption or HTTPS are often poorly implemented, leaving sensitive information exposed to cyber threats. Moreover, these systems do not offer real-time user notifications, and there is usually no OTP verification as an additional layer of security, leading to limited user awareness regarding authentication attempts. The failure to properly monitor and audit access logs makes it difficult to detect or prevent unauthorized access attempts.

III. PROPOSED SYSTEM

This paper presents an innovative approach to a highly secure, fingerprint-based authentication mechanism that incorporates OTP verification to fortify the authentication process. Built with a combination of Flask, OpenCV, PostgreSQL, and TensorFlow, the system ensures secure access through a multi-layered verification process. During user registration, the system requires a username, phone number, and fingerprint image. The fingerprint image undergoes preprocessing, liveness detection, and minutiae extraction before being stored in the database. To ensure that the user is genuine, an OTP is sent via Twilio for identity verification. For login, users are required to enter their username, receive an OTP, and upload their fingerprint image. The fingerprint is then authenticated by comparing it against the stored data using Euclidean distance-based feature comparison. The system also employs image processing techniques and a deep learning model to detect spoofed fingerprints, ensuring that only live fingerprints are processed. Upon successful authentication, users gain access to a secure dashboard. This system is designed to be highly secure, scalable, and efficient, offering real-time authentication and remote access.

A. Algorithms

1) *Euclidean Distance:* The Euclidean Distance algorithm calculates the similarity between two fingerprint feature sets by measuring the straight-line distance between them in a multi-dimensional space. If the distance is below a predefined threshold (50 in this case), the fingerprint is considered a match; otherwise, authentication fails. This method ensures a balance between accuracy and computational efficiency.
2) *Adaptive Thresholding:* Adaptive Thresholding is an image processing technique used to convert grayscale images into binary images. It dynamically determines the threshold for each region of an image, helps separate ridges from the background, making feature extraction more accurate and robust against variations in image quality.
3) *Gaussian Blur (Noise Reduction in Fingerprint Images):* Gaussian Blur is a smoothing technique that reduces image noise and enhances the clarity of fingerprint ridges by averaging pixel values using a Gaussian function. Before extracting features, the fingerprint image is blurred to reduce noise and improve minutiae detection. This step helps remove unwanted artifacts that could interfere with accurate fingerprint matching.
4) *Good Features to Track (Minutiae Extraction):* The Good Features to Track algorithm detects key points in an image based on contrast and edge sharpness. It is commonly used in computer vision tasks such as feature extraction. This algorithm extracts minutiae points (ridge endings and bifurcations) from the processed fingerprint image. These points serve as the unique features used for fingerprint matching.
5) *Canny Edge Detection (Liveness Detection):* Canny Edge Detection is an edge-detection algorithm that helps detect edges in fingerprint images to assess their complexity. Fake fingerprints (e.g., silicone or printed versions) tend to have fewer distinct edges compared to real fingerprints. The system uses edge count as a heuristic for liveness detection if the number of detected edges is too low, the fingerprint might be a spoof.

6) *Machine Learning-Based Liveness Detection:* A Convolutional Neural Network (CNN) model, built using TensorFlow, is trained to distinguish between live and spoofed fingerprints. The model learns from a dataset of real and fake fingerprints to classify new inputs based on patterns and textures. It analyses image features and assigns a probability score, determining whether the fingerprint is real or fake. This adds an extra layer of security to prevent unauthorized access using fake fingerprints.

IV. SOFTWARE REQUIREMENTS

Software requirements outline the necessary software resources and prerequisites that must be installed on a system to ensure the optimal functionality of an application. These requirements are typically not included in the software installation package and must be installed separately beforehand.

1) Platform: In computing, a platform serves as a foundation, either in terms of hardware or software, that enables an application to run efficiently. This is built using Python, leveraging its rich ecosystem for web development, machine learning, and image processing.
2) Web Framework: The authentication system is built using Flask, a lightweight and scalable web framework in Python. Flask provides essential features like routing, request handling, and session management, enabling efficient backend development.
3) Database: For secure data management, the system utilizes PostgreSQL, a robust open-source relational database management system. PostgreSQL ensures data integrity, scalability, and security through advanced encryption and structured storage mechanisms.
4) APIs and Drivers: The system relies on Twilio API for real-time SMS-based OTP verification, providing an additional security layer for authentication. Furthermore, OpenCV is integrated for fingerprint image processing, requiring specific dependencies and drivers for optimal functionality.
5) Machine Learning Framework: To enhance security, TensorFlow is incorporated for liveness detection, ensuring that only genuine fingerprints are accepted. These frameworks facilitate deep learning-based analysis to prevent spoofing attacks.
6) Web Browser: Since the authentication system is web-based, a modern web browser such as Google Chrome, Mozilla Firefox, or Microsoft Edge is required for accessing the application. The system ensures a seamless user experience through a responsive web interface.
7) Security and Encryption: To safeguard sensitive biometric data, the system implements AES encryption and enforces HTTPS protocols for secure communication. Additionally, Flask-SQL Alchemy is used as an ORM framework to manage database interactions securely.
8) Technical Stack:

- Primary Language: Python
- Frontend Framework: Flask
- Backend Framework: Visual Studio
- Database: PostgreSQL
- Frontend Technologies: HTML, CSS, JavaScript, Bootstrap 5

V. HARDWARE REQUIREMENTS

Hardware requirements specify the physical system resources necessary for running an operating system or software application efficiently. A hardware requirements list is often accompanied by a Hardware Compatibility List (HCL), particularly for operating systems. The HCL provides details about tested and compatible hardware, as well as any known incompatibilities. The key hardware components are discussed below:

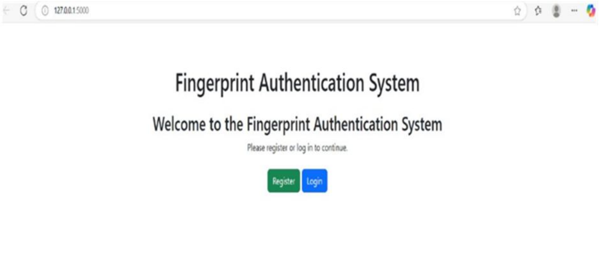
1) Architecture – Computer operating systems are designed for specific hardware architectures. Most software applications are built for specific operating systems running on designated architectures. The System is designed to be compatible with x86_64 and ARM architectures, ensuring flexibility in deployment across various platforms.
2) Processing Power – The CPU plays a crucial role in determining a system’s capability to run software efficiently. While Intel and AMD processors with similar clock speeds may have varying performance, Intel Core i5 (or AMD Ryzen 5) and with Intel Core i7 or Ryzen 7+ preferred for machine learning tasks such as liveness detection.
3) Memory (RAM) – When software runs, it occupies space in the system’s random-access memory (RAM). The required memory capacity is determined by factors such as the application’s demand, the operating system, supporting software, and other background processes. Optimal RAM allocation ensures smooth multitasking and prevents performance issues.

4) Graphics Processing (Display Adapter) – Applications requiring high-end graphics, such as video editing software and advanced gaming applications, may necessitate dedicated graphics cards for optimal performance.
5) Network Infrastructure – A stable internet connection is required for real-time OTP-based authentication (via Twilio API) and integration with Aadhaar-based verification. A minimum broadband speed of 10 Mbps is recommended to ensure smooth communication between the authentication system and external services.
6) Peripherals – Some applications require specific peripherals for enhanced functionality. This system requires a high-resolution fingerprint scanner to capture and authenticate biometric data accurately.

A. *Minimum Hardware Specifications*

- Operating System: Windows (Only)
- Processor: Intel i5 or higher
- RAM: 8GB or more
- Storage: 25GB of free space on the local drive

VI. RESULT

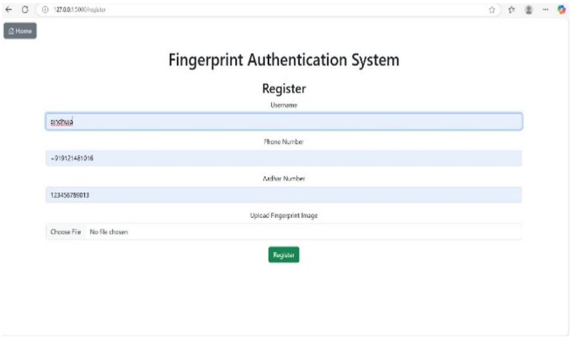

127.0.0.1:5000

Fingerprint Authentication System

Welcome to the Fingerprint Authentication System

Please register or log in to continue.

Fig 1: Index Page


127.0.0.1:5000/register

Fingerprint Authentication System

Register

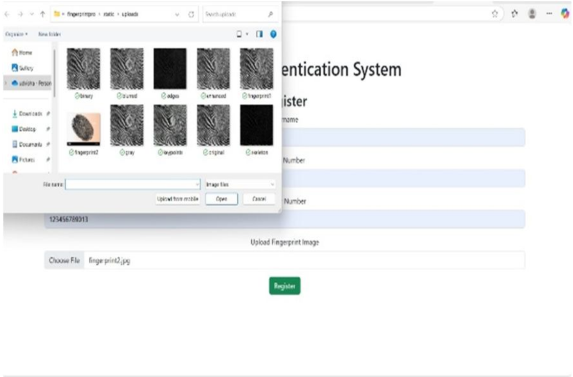
Username: sndhya

Phone Number: +919121481016

Adhar Number: 123456789013

Upload Fingerprint Image: Choose File No file chosen

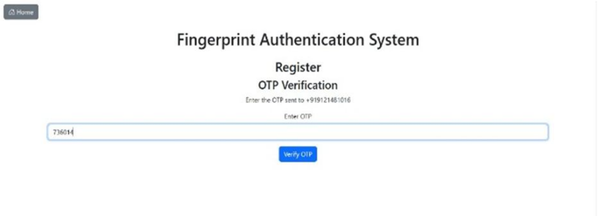
Fig 2: Registration page


fingerprintpro › static › uploads

123456789013

Upload Fingerprint Image: Choose File fingerprint2.jpg

Fig 3: Uploading Data



Fingerprint Authentication System

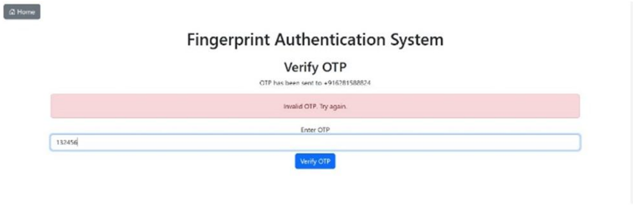
Register

OTP Verification

Enter the OTP sent to +919121481016

Enter OTP: 736014

Fig 4: OTP Verification Page



Fingerprint Authentication System

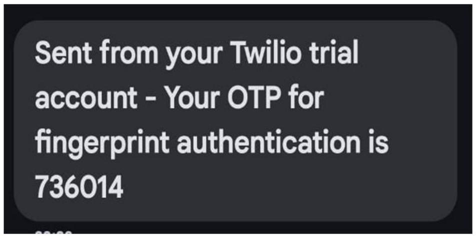
Verify OTP

OTP has been sent to +916281588824

Invalid OTP. Try again.

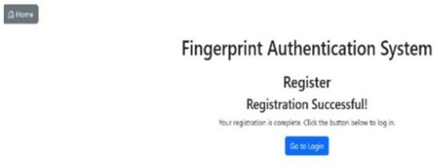
Enter OTP: 132456

Fig 5: Mismatch OTP



Sent from your Twilio trial account - Your OTP for fingerprint authentication is 736014

Fig 6: SMS from Twilio



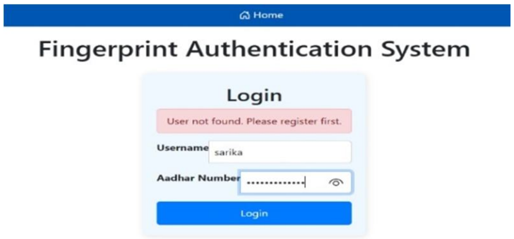
Fingerprint Authentication System

Register

Registration Successful!

Your registration is complete. Click the button below to log in.

Fig 7: Successful Verification



Fingerprint Authentication System

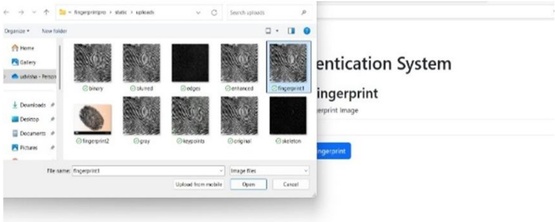
Login

User not found. Please register first.

Username: sarika

Aadhar Number: ••••••••••••

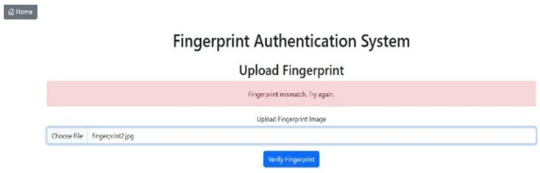
Fig 8: Login Form with Invalid details



File name: fingerprint1

Image files

Fig 9: Upload Fingerprint to Authenticate user



Fingerprint Authentication System

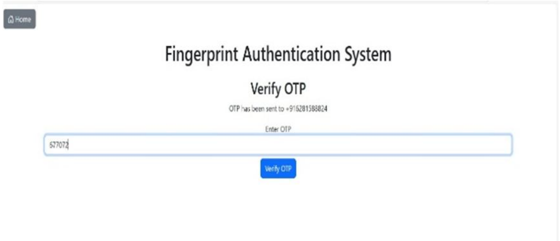
Upload Fingerprint

Fingerprint mismatch. Try again.

Upload Fingerprint Image

Choose File fingerprint2.jpg

Fig 10: Mismatch Fingerprint



Fingerprint Authentication System

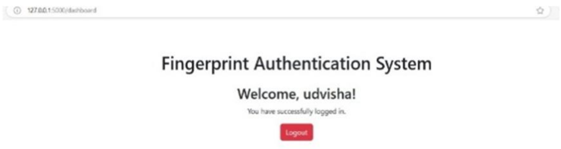
Verify OTP

OTP has been sent to +916281588824

Enter OTP

577072

Fig 11: Verification OTP to registered Phone Number


127.0.0.1:5000/dashboard

Fingerprint Authentication System

Welcome, udvisha!

You have successfully logged in.

Fig 12: Dashboard page

VII. COMPARISON GRAPHS

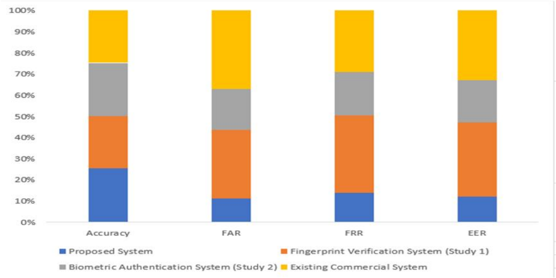


Fig 13: Fingerprint Matching Accuracy

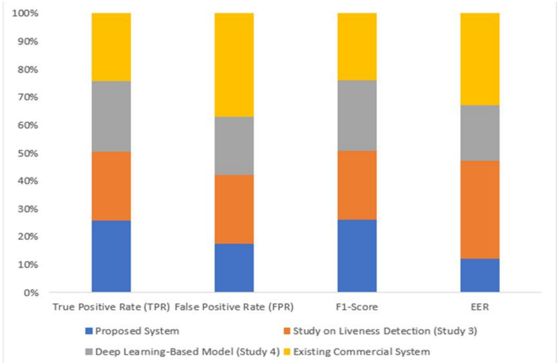


Fig 14: Liveness Detection

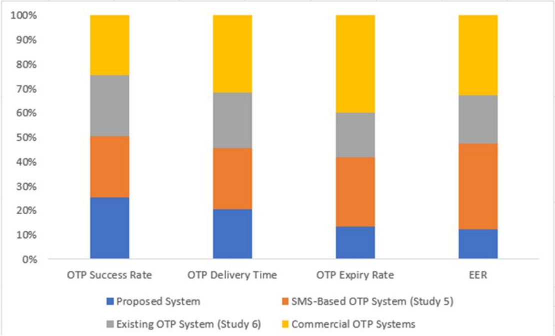


Fig 15: OTP Verification

VIII. FUTURE SCOPE

Future versions of the system should incorporate multi-modal biometric authentication to enhance flexibility and security. This would allow users to choose between various biometric modalities, such as fingerprints, facial recognition, iris scans, or voice recognition, based on their preferences and needs. Additionally, using multiple biometric traits simultaneously would significantly improve security, making it more challenging for attackers to spoof multiple biometric modalities at once. To further strengthen the system's security, enhanced anti-spoofing techniques should be integrated. Deep learning models and AI-based solutions can be used to detect more sophisticated spoofing attempts. Advanced methods like 3D imaging, thermal detection, and micro-movement analysis could be employed to differentiate between real and fake biometric data, ensuring more robust liveness detection. Blockchain technology could be leveraged to improve data integrity and security. By storing authentication logs and transaction records on a blockchain, the system would create a tamper-proof, transparent record of user data changes and authentication attempts. To enhance OTP security, alternative verification methods should be explored. Biometric OTP generation, where a fingerprint or iris scan is used to generate a one-time password, could improve security and reduce reliance on SMS-based OTPs. Implementing two-factor authentication (2FA), push notifications, hardware tokens, or FIDO2/WebAuthn standards would further strengthen user authentication, mitigating risks such as SIM card swapping and man-in-the-middle attacks.

IX. CONCLUSION

This research presents a robust fingerprint authentication system integrated with Aadhaar-based identity verification and real-time SMS notifications, designed to provide secure, scalable, and privacy-conscious authentication. The proposed system addresses key challenges in existing fingerprint authentication models, including privacy concerns, data security, and vulnerability to spoofing attacks. Through advanced techniques such as liveness detection, OTP-based verification, and secure data handling, the system significantly improves the accuracy, security, and user experience.

REFERENCES

[1] Jinhai Z. Study and implementation of automatic fingerprint recognition technology. In: International conference on uncertainty reasoning and knowledge engineering, Bali, Indonesia, 4–7 August 2011. USA: IEEE.
[2] Prabhakar S, Pankanti S and Jain AK. Biometric recognition: security and privacy concerns. IEEE Security Privacy 2003; 1: 33–42.
[3] Ratha N, Connell H and Bolle RM. Enhancing security and privacy in biometrics-based authentication systems. IBM Syst J 2001; 40: 615–633.
[4] Benhammadi F and Bey KB. Embedded fingerprint matching on smart card. Intern J Pattern Recognition Artif-Intell 2013; 27: 1350006.
[5] Teoh A, BengJin Connie T, et al. Remarks on Bio-Hash and its mathematical foundation. Inf Process Lett 2006; 100: 145–150.
[6] Tulyakov S, Farooq F and Govindaraju V. Symmetric: Bio- hash functions for fingerprint minutiae, (2005, accessed 10 October 2019)
[7] Radha N and Karthikeyan S. An evaluation of fingerprint security using non invertible bio-hash. Int J Netw Security Appl 2011; 3: 118–128.