A Secure and Scalable user Authentication Model for Smart City Access

E. O. Bennet; Obelley Happiness; O. E. Taylor
Department of Computer Science, Rivers State University, Port Harcourt, Nigeria

Abstract
Security threats and inefficiency remain significant concerns in today's smart city infrastructure. As the adoption of smart technologies increases, so does the risk of advanced cyber-attacks, especially in systems where authentication delays and security gaps exist. These vulnerabilities and inefficiencies compromise not only the safety of digital systems but also the trust of users. This research focuses on developing an advanced, secure, and scalable authentication model tailored to address these growing concerns. The primary objectives of this study are to achieve low latency in authentication processes, ensure faster response times, and integrate a strong and intelligent security framework suitable for real-time applications. The research employs Object-Oriented Analysis and Design (OOAD) alongside a constructive research methodology, which involves the iterative implementation and modular development of the proposed solution. The system was built using the Python programming language, chosen for its flexibility and powerful libraries in AI and biometric integration. A Face ID authentication system was developed, utilizing eye iris recognition and enhanced with machine learning and biometric security models to ensure precision and reliability. The experimental phase of the research demonstrated the effectiveness of this approach. The Face ID system achieved an accuracy score of 95%, surpassing the One-Time Password (OTP) method, which recorded an 88% accuracy rate. Furthermore, the success rate during real-time use was 98% for Face ID compared to 85% for OTP. This indicates that Face ID authentication is significantly more sturdy, with a lower rate of failure and reduced risk of false positives or negatives. These findings underscore the significance of integrating biometric-based authentication systems in smart cities. The proposed model not only enhances security but also improves user experience by offering faster and more reliable access. This positions Face ID as a superior alternative for safeguarding a smart city infrastructures in a rapidly evolving digital landscape.

Keywords: Authentication delays, Face ID authentication system, Iris recognition, Machine learning and Biometric-based authentication

I. Introduction

Urbanization has become a defining feature of the 21st century, with more than half of the world's population now residing in urban areas (United Nations, 2018). The rapid rise of technology has enabled non-rural residents to benefit from smart city innovations ranging from traffic management to energy utilization and public safety. Smart cities are designed to efficiently manage resources, improve living conditions for residents, and provide advanced infrastructure by leveraging digital technology. Smart cities rely heavily on applications such as water supply networks, waste management systems, e-learning platforms, smart libraries, and online payment services. These systems collect vast amounts of data through cameras, sensors, buildings, and devices, which can be used to foster innovation, minimize energy consumption, and improve service delivery. Citizens benefit through simplified governance, faster access to healthcare and education, and quick responses to everyday needs.

A key enabler of these services is user authentication systems, which ensure that only authorized individuals gain access to smart city services. Authentication not only supports access restriction but also enhances security by tethering users to service platforms, especially for fee-based utilities. However, the interconnected nature of smart city environments exposes them to significant security challenges. The proliferation of networks, gateways, and Internet of Things

DOI: https://doi.org/10.5281/zenodo.18692503

(IoT) devices increases vulnerabilities, creating opportunities for cyber attackers to compromise system integrity and access sensitive citizen data.

A recent cybersecurity breach in the London transportation system (September 2024) highlighted this vulnerability, where the bank details and personal information of over 5,000 passengers were exposed. Although solutions such as system updates and improved threat detection mechanisms have been deployed, scalability and efficiency remain major gaps. The growing reliance on digital technologies emphasizes the need for scalable and secure authentication systems that can withstand evolving cyber threats while ensuring reliability.

II. Related Works

A major challenge of inefficiency and security during access were identified in smart city. This review explores existing literature on these issues to identify gaps and potential.

Smart cities often consist of various interconnected systems, ranging from access control to e-learning platforms and administrative systems [1]. The seamless integration of these diverse systems is a fundamental aspect of a smart city. However, this integration often leads to inefficiency challenges in the authentication process. As each system may have its own authentication mechanisms, users are required to authenticate across different platforms.

This can result in slow authentication responses and delays during peak usage periods, affecting the usability of the system [2]. During busy hours, multiple requests from users can lead to congestion, which significantly impacts the performance of the authentication systems.

A study by Sharma and Soni highlights that many smart campus authentication systems fail to scale effectively to accommodate a growing user base, resulting in latency and slower response times [3].

Recent advancements focus on: Biometric Authentication leveraging biometrics for faster and more secure access, though scalability and privacy concerns persist [4]; Blockchain Technology, decentralized systems that improve authentication efficiency via secure and transparent identity verification [5]; and AI-Driven Authentication, machine-learning models for detecting anomalous login attempts in real time to enhance both speed and security [6]. Another major issue contributing to slow authentication response times in smart cities is the presence of data silos… this fragmentation results in significant delays as users may need to undergo multiple rounds of authentication to access various services [7].

Research has shown that IoT devices like IP cameras and smart speakers can be attacked by malware such as the Mirai virus, which can damage the internet's infrastructure, including IoT networks [8]. Cloud servers are also at risk when attackers take advantage of security gaps such as fraud, spying, or data falsification, whether from outside hackers or harmful insiders [9]. Because of these threats, it's important to confirm the identity of every user before giving access to applications or connected devices, often by checking several pieces of information about them [10]. As far back as 2010, the European Commission highlighted the importance of security and privacy in IoT, noting that these factors are essential for IoT to be widely adopted. To protect IoT systems fully, strategies must cover device security, message checks, user permissions, data integrity, privacy, and reliable availability [11]. All of these are integrated in the development of the proposed secure and efficient authentication process.

The balance between security and efficiency is a major concern. As smart cities become increasingly reliant on digital technologies, the risk of cyberattacks, including malware threats, also rises. These attacks not only compromise the confidentiality and integrity of user data but can also lead to slow authentication responses as security protocols attempt to mitigate threats. Many smart cities lack sophisticated malware detection systems that can identify and respond to emerging threats in real time; as a result, security mechanisms can inadvertently introduce delays in the authentication process, further compounding the issue of slow responses [12].

III. System Design

This system adopts constructive research methodology and an object oriented data analysis methodology.

The system design is represented by a high-level view as shown in figure 1.

DOI: https://doi.org/10.5281/zenodo.18692503

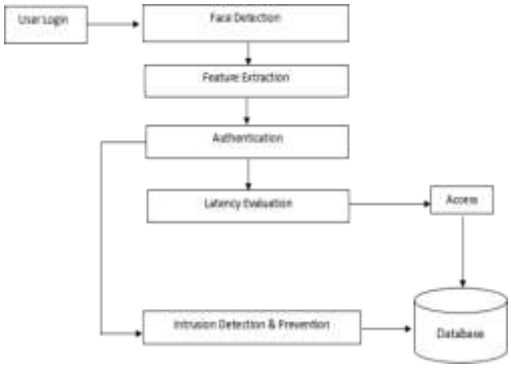

Figure 1: Architecture of the System

The architectural design of the smart city authentication model, as illustrated in Figure 1, defines the structured interaction between key components to ensure secure, efficient, and real-time authentication. The system architecture integrates multiple systems; each component plays a crucial role in enhancing security, minimizing latency, and ensuring reliable user authentication in smart city applications. It also highlights the flow of data and interactions between these components.

The design of an efficient model follows a modular, layered architecture, integrating biometrics, AI, MFA, and IoT for interoperability and scalability. A microservices approach separates key components like identity verification, encryption, and incident detection, allowing them to function independently and scale as needed.

Security is reinforced through multi-layered defenses, AI-driven monitoring, and biometric authentication, ensuring real-time processing with minimal latency. The system is designed to handle large-scale data flow securely; the system integrates seamlessly with smart city infrastructures and adapts to evolving security needs.

Functional Design

In designing a model for a secure authentication system for smart cities, functional design defines how the different sub-system would interact to achieve reliable and efficient performance. It outlines the key operational processes, algorithms, and techniques that ensure seamless authentication, security, and user experience.

This section provides a structured breakdown of how each component functions, ensuring that the authentication system meets the requirements of accuracy, speed, security, and scalability within the smart city framework. The system processes is expressed by 3 key algorithms:

Algorithm 1: Face Detection

```
BEGIN FaceDetection
// Image Acquisition
image = CAPTURE FRAME
// Preprocessing
gray_image = CONVERT_TO_GRAYSCALE
resized_image = RESIZE_IF_LARGE
processed_image = ENHANCE_IMAGE

// Face Detection
candidate_regions =
SCAN_FOR_FACE_LIKE_PATTERNS
detected_faces

FOR EACH region IN candidate_regions DO
 IF
DETECT_WITH_HAAR_CASCADE(region)
OR DETECT_WITH_CNN(region) THEN
 detected_faces.ADD(region)
  END IF
END FOR

Recognition and Liveness
FOR EACH face IN detected_faces DO
encoding = EXTRACT_FACE_ENCODING
 match = COMPARE_WITH_DATABASE

 IF IS_LIVE_FACE(face) THEN
 OUTPUT face.RegionOfInterest, match
  ELSE
  FLAG_AS_SPOOF
  END IF
  END FOR
 END FaceDetection
```

Algorithm 2: Face Extraction

```
BEGIN FaceFeatureExtraction
// Input Preparation
input_frame=
GET_FRAME_FROM_CAMERA_OR_VIDE
O()
detected_faces=
GET_PREVIOUSLY_DETECTED_FACE_R
EGIONS
// Face Isolation and Preprocessing
FOR EACH face_region IN detected_faces
DO
 cropped_face = CROP_REGION
```


DOI: https://doi.org/10.5281/zenodo.18692503

```
 resized_face=
RESIZE_TO_STANDARD_DIMENSION
 normalized_face=
NORMALIZE_PIXEL_VALUES
  aligned_face=
ALIGN_FACIAL_FEATURES
 // Feature Extraction
 feature_vector=
EXTRACT_FEATURE_VECTOR
STORE feature_vector AS DigitalIdentity
END FOR
END FaceFeatureExtraction
```

Algorithm 3: Feature Matching

```
BEGIN FaceRecognitionSystem
// Input
input_feature_vector=
RECEIVE_VALIDATED_FEATURE_VECT
OR()
// Retrieve Stored Identities
stored_vectors= FETCH_FROM_DATABASE
// Compare Input Vector with Stored Vectors
min_distance = INFINITY
matched_face_id = NULL
FOR EACH stored_vector IN stored_vectors
DO
distance=
CALCULATE_EUCLIDEAN_DISTANCE(in
put_feature_vector, stored_vector.vector)
 IF distance < min_distance THEN
 min_distance = distance
 matched_face_id = stored_vector.face_id
 END IF
END FOR
// Decision Making
IF min_distance < THRESHOLD THEN
 RETURN matched_face_id
ELSE
RETURN "Unknown"
END IF
END FaceRecognition
```

IV. Results & Discussion

System Setup

Edge Devices were installed in smart city access; public buildings, transportation hubs, and surveillance areas. Each device was configured with a local lightweight operating system; Embedded Linux and ran a C++ program for capturing biometric inputs; fingerprint or facial scan) and environmental data (etimestamp, location). The devices used Message Queuing Telemetry Transport (MQTT) protocol for lightweight messaging to the central server.

A Central Authentication Server hosted on an Ubuntu 20.04 LTS server, this system was set up to handle user data requests, perform authentication, and issue authorization tokens using JWT (JSON Web Tokens). The backend was developed in Python using Flask, integrated with MySQL for structured data storage, and equipped with AES-256 encryption to protect sensitive information during storage and transmission.

A Real-Time Monitoring Dashboard built using JavaScript (React.js), the dashboard was hosted via an Nginx server. It provided real-time visualization of authentication attempts, alerts for suspicious access patterns, and performance metrics. The frontend consumed REST APIs exposed by the Python backend.

A Mobile Authentication App developed in Kotlin for Android devices, the app allowed users to register and authenticate using biometrics and QR code scanning. It connected to the server via HTTPS using OAuth 2.0 for secure access token management.

Security Setup using Transport layer Security - TLS 1.3 encryption was enabled across all communication channels. Firewalls, token expiration mechanisms, and anomaly detection models (implemented in Python) were deployed to prevent attacks such as spoofing, brute force, and unauthorized access.

Figure 2 shows the sample dataset used

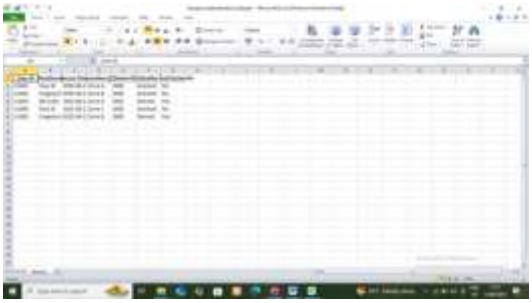

Figure 2: Sample Dataset

The sample dataset displays the data-set designed to simulate real-world smart city authentication events across multiple entry points (e.g., public buildings, smart transportation stations, and security checkpoints). It includes essential attributes required to process, authenticate, and detect anomalies in user access attempts.

DOI: https://doi.org/10.5281/zenodo.18692503

Figure 3 shows the user access attempts

Figure 3: OTP for User Access Using Login

Figure 3 serves as the initial point of interaction for users attempting to access the system. Upon entering their username and password, a one-time password (OTP) was generated and sent to the user's registered mobile number or email address.

Figure 4 shows the sample face recognition output.

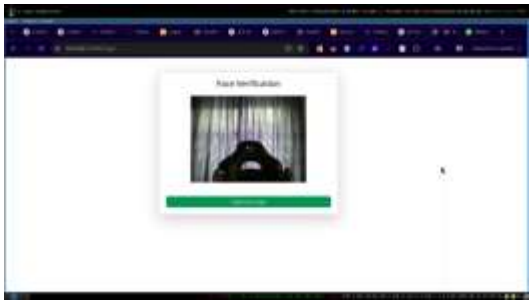

Figure 4: **Face Recognition Page Using Eye Iris**

Following successful OTP validation, shows how users were redirected to the biometric authentication interface, where facial recognition based on eye iris scanning was employed.

A sample of user telephone number page is shown in figure 5.

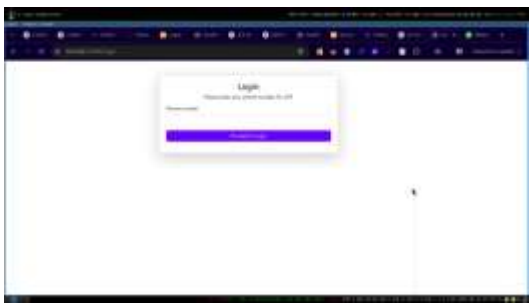

Figure 5: User Phone Number Page to Receive the OTP Code

Presented a form where users either verified their existing number or updated it for OTP delivery.

A user access page after login is shown in figure 6.

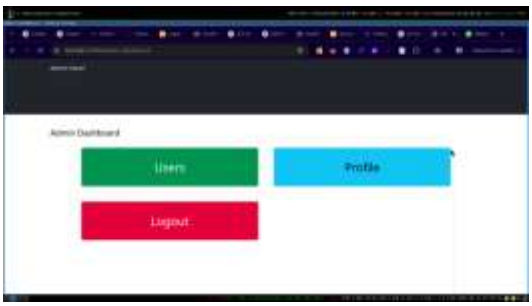

Figure 6: **User Access Page After Login**

Figure 6 shows the user access page after a successful OTP authentication.

Figure 7 shows the Matric Comparison gragh

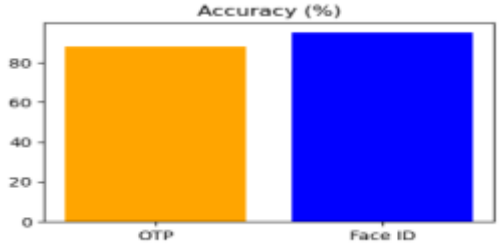

Figure 7: **Matrix Comparison Graph: OTP vs Face ID**

Figure 7 analyzes the accuracy comparison graph. It is focused specifically on measuring the authentication correctness of the two methods: OTP and Face ID. The bar representing Face ID stood taller, reflecting a 95% accuracy, while OTP lagged behind with 88%.

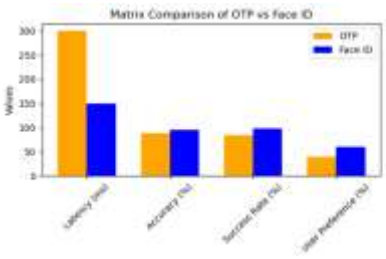

Figure 8: Matrix OTP Latency

It analyzes the graph focused exclusively on the performance of the OTP authentication method across the four-evaluation metrics. Visual representation allowed a deeper

DOI: https://doi.org/10.5281/zenodo.18692503

understanding of OTP's strengths and limitations.

The latency bar was the tallest, reaching 300 milliseconds, which highlighted OTP's biggest drawback it typically takes longer due to the processes of code generation, transmission (via SMS or email), and user input. While accuracy (88%) and success rate (85%) were relatively close, they revealed that OTP still had notable chances of failure or incorrect verification.

The user preference bar was the shortest (40%), indicating that less than half of users favored OTP. This could stem from the cumbersome nature of entering codes manually, the possibility of delay or message delivery failure, or concerns about phishing and SIM-swap attacks.

Authentication Method Metrics:

Metric	OTP	Face ID
Latency (ms)	300	150
Accuracy (%)	88	95
Success Rate (%)	85	98
User Preference (%)	40	60

Figure 9: Classification Graph

The Classification Graph Indicated that 60% of users preferred Face ID while only 40% preferred OTP, suggesting a growing acceptance of biometric systems due to their ease and speed. Overall, the matrix comparison graph effectively highlighted the superiority of Face ID across key authentication parameters, reinforcing its suitability for secure, efficient, and user-friendly smart city applications.

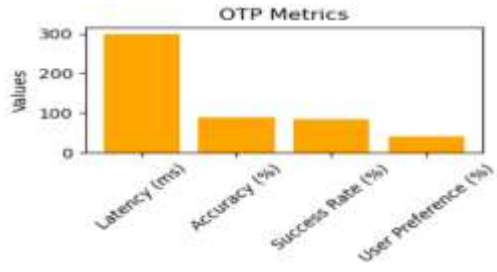

Figure 10: OTP Metrics Graph

The graph shows the overall performance of the Face ID authentication system across the same four metrics. It showcased a balanced and consistent superiority, with low latency (150ms), high accuracy (95%), excellent success rate (98%), and high user preference (60%).

The relatively low latency bar reflected the system's capability to rapidly authenticate users using facial or eye recognition, aided by optimized machine learning models and fast sensors. The tall accuracy and success rate bars further reinforced the dependability of Face ID not only was it precise, but it also delivered consistent results under various lighting or facial conditions.

A high user preference of 60% suggested that users favored the convenience of simply looking into a device for authentication over manually entering OTPs. The consistency of the metric heights in this graph highlighted Face ID as a well-rounded solution, ideal for real-time applications in smart city environments such as autonomous transportation access, facial ticketing systems, or smart building entry.

Table 1: Metrics Evaluation

Metric	OTP	Face ID (Eye Iris)
Latency (ms)	300	150
Accuracy (%)	88	95
Success Rate (%)	85	98
User Preference (%)	40	60

From the table 1, it is evident that Face ID consistently outperformed OTP across all evaluation criteria. Its low latency ensured a smooth user experience, while its high accuracy and success rate confirmed its effectiveness as a reliable authentication tool.

Furthermore, the biometric system aligned well with user expectations in a smart city, where quick and secure access is crucial. Despite OTP still being a viable secondary method, its reliance on mobile networks and susceptibility to human error made it less desirable in dynamic urban environments.

The evaluation of the results involved both quantitative and qualitative assessments. All performance data collected were systematically reviewed and tabulated to understand the strengths and weaknesses of each authentication method.

DOI: https://doi.org/10.5281/zenodo.18692503

V. Conclusion

This paper focused on the development of a low-latency, efficient authentication and advanced security system specifically designed for smart cities, where real-time responsiveness and high security are critical. Two primary authentication methods were implemented and evaluated: One-Time Password (OTP) and Face ID using eye iris recognition. The goal was to determine which method provides better performance in terms of speed, accuracy, success rate, and user satisfaction.

The data clearly showed that Face ID significantly outperformed OTP. It demonstrated lower latency (150 ms) compared to OTP's 300 ms, higher accuracy (95%), a greater success rate (98%), and a higher user preference rate (60%). These results were visualized through four individual bar graphs and a matrix comparison graph, supported by a structured dataset.

References

[1] Khan, A., Aslam, S., Aurangzeb, K., Alhussein, M., & Javaid, N. (2022). Multiscale modeling in smart cities: A survey on applications, current trends, and challenges. *Sustainable Cities and Society, 78*, 103517.

[2] Sharma, R., & Arya, R. (2022). A secure authentication technique for connecting different IoT devices in the smart city infrastructure. *Cluster Computing, 25*(4), 2333–2349.

[3] Rathore, M. M., Paul, A., Ahmad, A., Chilamkurti, N., Hong, W.-H., & Seo, H. C. (2018). Real-time secure communication for smart city in high-speed big data environment. *Future Generation Computer Systems, 83*, 638–652.

[4] LiZhao, Y., Chen, M., Xu, Z., Zheng, X., Hu, H., Yao, J., Qian, L. (2019). Inkjet-printed unclonable quantum dot fluorescent anti-counterfeiting labels with artificial intelligence authentication. *Nature Communications, 10*(1), 1–9.

[5]Chen, X., Wang, Y., Li, Z., & Zhao, T. (2020). FedCluster: Boosting the convergence of federated learning via cluster-cycling. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 6207–6216). IEEE.

[6]Dhillon, P. K., & Kalra, S. (2019). A secure multi-factor ECC based authentication scheme for cloud-IoT based healthcare services. Journal of Ambient Intelligence and Smart Environments, 11(2), 149–164.

[7]Kumari, A., Gupta, R., & Tanwar, S. (2021). Amalgamation of blockchain and IoT for smart cities underlying 6G communication: A comprehensive review. Computer Communications, 171, 139–158.

[8]Antonakakis et al. (2017) Antonakakis, M., April, T., Bailey, M., Bernhard, M., Bursztein, E., Cochran, J., ... Kallitsis, M. (*2017). Understanding the Mirai botnet. In Proceedings of the 26th USENIX Security Symposium (USENIX Security 17) (pp. 1093–1110). USENIX Association.*

[9] Yang, Y., Wu, L., Yin, G., Li, L., & Zhao, H. (2017). A survey on security and privacy issues in Internet-of-Things. *IEEE Internet of Things Journal, 4*(5), 1250–1258.

[10] National Institute of Standards and Technology. (2017). Digital identity guidelines: Authentication and lifecycle management *(NIST Special Publication 800-63B).*

[11] European Commission. (2010). Internet of Things: An action plan for Europe. *COM(2009) 278*.

[12] Neshenko, N., Bou-Harb, E., Crichigno, J., Kaddoura, G., & Ghani, N. (2019). Demystifying IoT security: An exhaustive survey on IoT vulnerabilities and a first empirical look on Internet-scale IoT exploitations. *IEEE Communications Surveys & Tutorials, 21(3), 2702–2723*.

DOI: https://doi.org/10.5281/zenodo.18692503