***DOI: 10.22413/ijastems/2026/v12/i4/1***

# AFRS: An Agentic AI-Based Facial Recognition Continuous Post-Authentication Security on Mobile System for Devices

***Dr.M. Roshini*, G. Balaji, K. Dinesh Reddy, V. A. Lokesh, K. S. Huzaifa, A. Durgesh***
***Aditya College of Engineering, Madanapalle – 517325, Andhra Pradesh, India***
****Corresponding author: caihod@acem.ac.in***

***Abstract*—This paper presents the Agentic Facial Recognition System (AFRS), a continuous and non-intrusive mobile security framework designed to verify user identity beyond the initial device unlock. Conventional mobile authentication methods such as PINs, passwords, and one-time biometric scans protect only the entry point of a session, leaving the device exposed to unauthorized use after unlock. AFRS addresses this post-authentication gap by leveraging the front-facing camera to capture real-time facial data, which is processed by deep learning models to generate unique high-dimensional vector embeddings for ongoing user verification. Rather than blocking access upon detecting anomalies, the system maintains a seamless user experience while executing an intelligent security response: it logs the suspicious event, tracks repeated occurrences, and initiates an automated workflow through n8n to alert designated security agents. The proposed architecture is built on three cooperating AI agents, namely an Identity Agent, a Behaviour Agent, and a Context Agent, coordinated through an agentic orchestration layer. Experimental results demonstrate that the system achieves reliable facial identification and real-time anomaly detection with low computational overhead suitable for mobile deployment. AFRS establishes a robust, auditable layer of behavioural security that ensures persistent surveillance with minimal disruption to the legitimate user.***

***Index Terms*—Agentic AI, Facial Recognition, Vector Embeddings, Continuous Authentication, Mobile Security, Anomaly Detection, n8n Automation, Behavioural Biometrics.***

## I. INTRODUCTION

The rapid proliferation of smartphones and mobile applications has made these devices central repositories of sensitive personal, financial, and professional information. Modern smartphones routinely store banking credentials, private communications, corporate documents, and health records. Despite identify applicable funding source here. If none, delete this line. the value of this data, the security model governing most mobile devices remains fundamentally static. Authentication is treated as a one-time event performed at the moment of device unlock, after which the system extends unconditional trust to whoever operates the device for the remainder of the session [1].

This one-time verification model creates what is commonly referred to as the *post-authentication gap*. Once a device is unlocked by a legitimate owner, any subsequent user—whether an opportunistic bystander, a thief, or a malicious insider— gains unrestricted access to all applications and data. Scenarios such as snatch-and-run theft, shoulder-surfing attacks, and temporary device borrowing expose this weakness clearly. Traditional security mechanisms, including PINs, pattern unlock, fingerprint scanning, and static face recognition, are entirely blind to such incidents because they do not monitor what happens after the initial access decision is made [2].

The limitations of one-time authentication have led researchers to explore *continuous authentication*, a paradigm in which the system monitors the current device user throughout the entire session and re-evaluates legitimacy at frequent intervals [3]. Early continuous authentication approaches relied on passive behavioral signals such as keystroke dynamics [4], touch interaction patterns [5], and inertial sensor data [6]. More recent work has extended these approaches by combining multiple behavioral modalities with biometric identity verification to build more robust and context-aware security systems [7].

A parallel development in this space is the rise of agentic AI, where autonomous reasoning agents are used to coordinate multi-dimensional data analysis, make real-time decisions, and trigger automated responses [8]. Unlike traditional rule-based monitoring, agentic systems can adapt to changing conditions, handle uncertainty, and orchestrate complex workflows without explicit human instruction. These properties make agentic architectures well suited for building persistent authentication systems that must operate reliably across varying user states, environmental conditions, and threat scenarios [9].

This paper presents AFRS, an Agentic Facial Recognition System designed to provide continuous, non-intrusive post-authentication security on mobile devices. The system integrates deep learning-based facial verification, behavioral monitoring, contextual risk assessment, and automated alert generation through a coordinated three-agent architecture. The primary contribution of AFRS is the combination of ongoing facial identity verification with intelligent workflow automation using n8n, enabling the system to detect and respond to suspicious activity while preserving the experience of the legitimate user.

The remainder of this paper is organized as follows. Section II reviews related work in continuous authentication and behavioral biometrics. Section III presents the system architecture of AFRS. Section IV describes the implementation methodology. Section V reports experimental evaluation results. Section VI discusses limitations and future directions, and Section VII concludes the paper.

*DOI: 10.22413/ijastems/2026/v12/i4/1*

## II. LITERATURE SURVEY

Research in continuous authentication has grown significantly over the past decade as the inadequacy of one-time verification became increasingly apparent. This section reviews the most relevant prior work across three domains: behavioral biometrics, facial recognition, and automated security systems.

### A. Behavioral Biometrics for Mobile Security

Frank et al. [4] were among the first to demonstrate that touch dynamics on mobile devices could serve as a reliable continuous authentication mechanism. Their work showed that touch gesture patterns including velocity, pressure, and trajectory could distinguish users with high accuracy. This foundational result motivated subsequent research in multimodal behavioral authentication.

Shen et al. [10] extended this work by combining touch dynamics with accelerometer data, demonstrating that multimodal fusion produces significantly better authentication performance than single-signal approaches. Similarly, Sagbas¸ et˘ al. [11] proposed a continuous authentication method using soft-keyboard typing behavior, grip style, and motion sensor data, reporting low equal error rates on a smartphone dataset. These studies confirm that behavioral biometrics captured passively during normal device usage can support robust continuous verification.

Bo et al. [12] demonstrated that motion sensor data alone, specifically accelerometer and gyroscope readings captured during device handling, could uniquely identify users based on how they hold and interact with a phone. This finding supports the integration of inertial sensor signals as a complementary authentication channel alongside facial and touch-based verification.

### B. Facial Recognition and Deep Embeddings

The development of deep learning-based facial recognition significantly improved the accuracy and scalability of identity verification systems. Schroff et al. [13] introduced FaceNet, which uses a deep convolutional neural network trained with triplet loss to map face images into a compact 128-dimensional embedding space. Euclidean distances between embeddings are used to determine face similarity, achieving 99.63% accuracy on the Labeled Faces in the Wild benchmark. FaceNet provides the embedding-based verification foundation upon which AFRS builds its identity verification layer.

Taigman et al. [14] developed DeepFace, which uses 3D alignment and deep neural networks to achieve near-human performance in face verification. King [15] introduced dlib's face recognition toolkit, which has been widely used in practical applications due to its accessibility and reliable performance. These tools, along with MediaPipe-based face alignment [16], form the technical basis of the facial processing pipeline in AFRS.

### C. Continuous Authentication Frameworks

Albuquerque et al. [17] proposed a continuous authentication framework for mobile banking using touch dynamics, specifically targeting the post-unlock security gap in financial applications. Their results confirmed that behavioral patterns collected during a banking session could detect unauthorized users with high precision. Progonov et al. [18] presented a behavior-based mobile authentication system suitable for IoT environments, arguing that continuous behavioral analysis provides stronger protection than event-triggered verification alone.

Baig et al. [19] addressed privacy concerns in continuous authentication by proposing a framework that processes biometric signals locally without transmitting raw data to remote servers. This privacy-by-design approach aligns closely with the AFRS architecture, which stores vector embeddings rather than raw facial images.

### D. Agentic AI and Automation in Security

The application of autonomous agents to cybersecurity tasks has attracted growing attention. Kindervag [20] introduced the Zero Trust security model, which advocates continuous verification rather than perimeter-based trust assumptions. This model directly motivates the persistent authentication paradigm implemented in AFRS.

Recent work on agentic AI architectures demonstrates that multi-agent systems can coordinate complex analytical tasks, adapt to dynamic inputs, and execute automated responses faster than rule-based systems [8]. Satyanarayanan [21] showed that edge-cloud hybrid architectures can support real-time intelligent processing for resource-constrained devices, which informs the distributed deployment strategy of AFRS.

Akinsola et al.[22] proposed a human-centered security approach that emphasizes integrating user behavior, awareness, and usability into cybersecurity strategies. Charanarur *et al*. [23] proposed a machine learning-based spam email detection system that employs classification algorithms to accurately distinguish spam from legitimate emails.

Goodfellow et al. [24] presented a comprehensive foundation of deep learning, covering neural network architectures, optimization techniques, and representation learning. Hochreiter et al.[25] introduced the Long Short-Term Memory (LSTM) network to overcome the vanishing gradient problem in recurrent neural networks.

## III. SYSTEM ARCHITECTURE

AFRS is designed as a distributed multi-layer architecture that separates lightweight data capture on the mobile device from deeper analysis and decision-making on backend and automation services. The system consists of four primary layers: the Mobile Client Layer, the Backend Server Layer, the Agentic Automation Layer, and the Reporting and Notification Layer.

### A. Mobile Client Layer

The mobile application is developed using Flutter and serves as the primary sensing and user interaction component. It continuously captures facial data through the front-facing camera, collects touch interaction signals, monitors typing patterns, and reads motion data from the device accelerometer and gyroscope. Raw camera frames are processed locally using MediaPipe for face detection and alignment before being converted into 128-dimensional

*DOI: 10.22413/ijastems/2026/v12/i4/1*

embeddings by the FaceNet model. These compact embeddings, rather than raw images, are transmitted to the backend, ensuring that identifiable visual data never leaves the device in its original form.

The mobile application also displays the current security status to the user and delivers real-time notifications when suspicious activity is detected. To minimize battery consumption and processor load, the application applies a thin-client approach in which heavy analysis is offloaded to backend services.

### B. Backend Server Layer

The backend is implemented using Node.js and Express.js, forming the coordination and storage hub of the system. It exposes RESTful APIs through which the mobile application submits processed embeddings, behavioral features, and contextual signals. The backend stores registered user profiles, authorized device records, behavioral baselines, facial embedding references, and security event logs in a persistent database.

Upon receiving new session data, the backend correlates the input with the stored owner profile and forwards the combined context to the agentic automation engine for trust evaluation. The backend also manages session records and maintains a complete audit trail of monitoring events.

### C. Agentic Automation Layer

The agentic reasoning engine is implemented using n8n, an open-source workflow automation platform. Three specialized workflows operate as autonomous agents within this layer.

The *Identity Agent* compares the current facial embedding against the enrolled reference embedding using Euclidean distance. If the computed distance exceeds a defined similarity threshold, the agent raises an identity mismatch signal.

The *Behaviour Agent* analyzes incoming touch dynamics, typing rhythm, and motion patterns against the stored behavioral baseline. Statistical deviation measures are used to quantify how much the current session differs from the owner's normal interaction profile.

The *Context Agent* evaluates the sensitivity of the active application, the device network conditions, and usage timing to determine whether the current operational context presents elevated risk.

The outputs of all three agents are aggregated into a unified suspicion score. If the score remains below a defined threshold, the session continues without interruption. If the threshold is crossed, the system initiates a protective response sequence including event logging, report generation, and owner notification.

### D. Reporting and Notification Layer

When suspicious activity is confirmed, AFRS generates a structured security report that describes the specific anomalies detected, their severity, and the contributing signals from each agent. This report is stored in the backend database and delivered to the device owner through the mobile application. The architecture ensures that every alert is accompanied by an explanation, improving transparency and enabling the user to understand why a protective action was triggered [22].

## IV. Implementation

### A. Facial Embedding Pipeline

The facial verification pipeline begins when the front-facing camera captures a frame during an active session. MediaPipe Face Detection is applied to locate the face region within the frame and perform landmark-based alignment, ensuring that the face is normalized for consistent embedding generation. The aligned face image is passed to a FaceNet-based model that produces a 128-dimensional embedding vector representing the user's facial identity.

During enrollment, the legitimate owner's facial embedding is computed and stored securely in the backend as the reference template. During monitoring, the current session embedding is compared to the stored template using Euclidean distance. A match is confirmed if the distance falls below the configured similarity threshold.

### B. Behavioral Signal Processing

Touch events captured by the mobile application are characterized by a feature set including start and end coordinates, stroke duration, average velocity, pressure, and contact area. Typing patterns are encoded as digraph latencies representing the timing relationship between consecutive key interactions. Motion sensor readings from the accelerometer and gyroscope are summarized using statistical features such as mean, standard deviation, and the Signal Vector Magnitude (SVM), defined as:

$$SVM = \sqrt{x^2 + y^2 + z^2} \quad (1)$$

These features are compared against the behavioral baseline stored during the initial enrollment period to detect deviations that may indicate a different user.

### C. Trust Score Computation

The suspicion score $S$ is computed as a weighted combination of the outputs from the three agent workflows:

$$S = w_I \cdot I + w_B \cdot B + w_C \cdot C \quad (2)$$

where $I$ is the identity mismatch signal (0–1), $B$ is the behavioral deviation score (0–1), $C$ is the contextual risk level (0–1), and $w_I$, $w_B$, $w_C$ are empirically assigned weights satisfying $w_I + w_B + w_C = 1$. When $S$ exceeds the defined threshold $\tau$, the system triggers a protective response.

### D. Automation Workflow with n8n

The n8n automation platform hosts the three agent workflows described in Section III. Each workflow is triggered by a webhook call from the backend server. Agent outputs are posted to a shared evaluation context, and a final orchestration workflow reads all agent outputs, computes $S$, and determines the appropriate response action. This modular structure allows individual agent workflows to be updated independently without modifying the broader system architecture.

## V. Result and Discussion

*DOI: 10.22413/ijastems/2026/v12/i4/1*

### A. Experimental Setup

The system was prototyped and tested in a controlled environment simulating normal mobile device usage. The backend server was deployed on a workstation with an Intel Core i7 processor, 16 GB RAM, and an NVIDIA GTX 1650 GPU. The mobile client was tested on Android devices with standard front-facing cameras. A dataset of ten registered users was constructed, with each user contributing facial enrollment data and a behavioral baseline collected over a three-day period. Impostor sessions were simulated by having different users attempt to use an enrolled device.

### B. Performance Results

The system's detection performance was evaluated across four primary metrics: facial verification accuracy, behavioral anomaly detection rate, false positive rate, and average response latency. The facial embedding pipeline achieved a verification accuracy of 97.4% under normal indoor lighting conditions. The behavioral deviation module detected impostor sessions with a true positive rate of 93.2% across all tested users. The overall false positive rate, representing legitimate users incorrectly flagged as suspicious, was measured at 4.1%. The average end-to-end response time from session data capture to alert delivery was 340 milliseconds, which is suitable for real-time mobile deployment.

### C. Comparison with Related Systems

When compared with touch-only continuous authentication approaches, AFRS demonstrated higher detection accuracy by incorporating facial identity verification as an additional modality. Systems relying solely on behavioral signals are more susceptible to imitation attacks where an intruder consciously mimics the owner's interaction style. The addition of facial embedding comparison provides a complementary

TABLE 1:AFRS performance summary

| Metric | Result |
|---|---|
| Facial Verification Accuracy | 97.4% |
| Impostor Detection Rate (True Positive) | 93.2% |
| False Positive Rate | 4.1% |
| Average End-to-End Latency | 340 ms |
| Memory Footprint (Mobile Client) | <45 MB |
| Battery Overhead (16 Hz sampling) | ≈ 8% |

verification channel that is difficult to spoof without physical resemblance to the enrolled owner. The automated n8n workflow layer also provides more flexible and auditable response logic compared to hardcoded threshold-based systems used in earlier work.

### D. Privacy Considerations

A significant concern in any system involving facial data collection is user privacy. AFRS addresses this through a privacy-by-design approach. The mobile application processes camera frames locally and converts them to numerical embeddings before any data leaves the device. Raw facial images are never transmitted to or stored on the backend server. Since embeddings are abstract mathematical vectors, they cannot be reversed to reconstruct the original face image, reducing the risk associated with a potential data breach. The system also complies with the data minimization principle by storing only the minimum information required for verification.

### E. Limitations

The system's facial verification performance is sensitive to environmental factors such as poor lighting, extreme viewing angles, and partial occlusions caused by accessories. Under such conditions, the identity agent may produce higher mismatch distances for the legitimate user, increasing the false positive rate. Future work will explore adaptive thresholds that adjust to ambient lighting conditions and improved embedding models robust to pose variation.

The behavioral baseline established during enrollment may also drift over time as the user's interaction patterns naturally evolve. A periodic baseline update mechanism using verified sessions will be investigated to maintain long-term authentication accuracy.

### F. Future Directions

Several enhancements are planned for future versions of AFRS. Integration of an LSTM-based sequence model for predicting anomalous application navigation patterns represents one direction. Incorporating thermal or depth sensors for liveness detection would strengthen resistance to photographbased spoofing attacks. The deployment of the full agentic stack on-device using lightweight model optimization techniques would eliminate backend latency and improve privacy further. Additionally, extending the system to wearable devices such as smartwatches would enable cross-device behavioral verification.

## VI. CONCLUSION

This paper presented AFRS, an agentic AI-based facial recognition system for continuous post-authentication security on mobile devices. The system addresses the post-unlock security gap left by conventional one-time authentication methods by monitoring user identity and behavior throughout an active device session. Through a three-agent architecture coordinated by n8n automation workflows, AFRS combines deep learningbased facial verification, behavioral biometric analysis, and contextual risk assessment to generate a real-time suspicion score. When suspicious activity is detected, the system logs the event, generates a structured security report, and delivers an alert to the device owner without disrupting the legitimate user.

Experimental evaluation demonstrated that AFRS achieves a facial verification accuracy of 97.4% and an impostor detection rate of 93.2% with an end-to-end response latency of 340 milliseconds and a battery overhead of approximately 8%. These results confirm that the proposed framework is both effective and practical for real-world mobile deployment. By treating post-authentication security as a continuous and intelligent process rather than a fixed checkpoint, AFRS provides a more adaptive and

***DOI: 10.22413/ijastems/2026/v12/i4/1***

reliable approach to protecting sensitive mobile data in modern usage environments.

## REFERENCE


[1] J. Kindervag, "Build Security Into Your Network's DNA: The Zero Trust Network Architecture," Forrester Research, Cambridge, MA, 2010.

[2] V. M. Patel, R. Chellappa, D. Chandra, and B. Barbello, "Continuous user authentication on mobile devices: Recent progress and remaining challenges," *IEEE Signal Process. Mag.*, vol. 33, no. 4, pp. 49–61, 2016.

[3] L. O'Gorman, "Comparing passwords, tokens, and biometrics for user authentication," *Proc. IEEE*, vol. 91, no. 12, pp. 2021–2040, 2003.

[4] M. Frank, R. Biedert, E. Ma, I. Martinovic, and D. Song, "Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 1, pp. 136–148, 2013.

[5] N. Zheng, A. Paloski, and H. Wang, "An efficient user verification system via mouse movements," in *Proc. ACM Conf. Comput. Commun. Security (CCS)*, Chicago, IL, 2011, pp. 139–150.

[6] Singh, Laishram Hemanta, et al. "Advancements in Detecting Deepfakes: AI Algorithms and Future Prospects − a Review." Discover Internet of Things, vol. 5, no. 1, May 2025, p. 53 https://doi.org/10.1007/s43926-025-00154-0..

[7] C. Shen, Y. Zhang, X. Guan, and R. A. Maxion, "Performance analysis of multi-motion sensor behavior for active smartphone authentication," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 1, pp. 45–57, 2016.

[8] A. Chan, A. Salvi, and M. Zeng, "The landscape of emerging AI agent frameworks," *arXiv preprint arXiv:2404.11584*, 2024.

[9] Y. Shoham and K. Leyton-Brown, *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge, UK: Cambridge University Press, 2009.

[10] C. Shen, Z. Cai, X. Guan, Y. Du, and R. A. Maxion, "User authentication through mouse dynamics," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 1, pp. 16–30, 2013.

[11] E. A. Sagbas¸, S. Oral, and A. Korkmaz, "Machine learning-based novel˘ continuous authentication method using typing behaviour and grip style in mobile devices," *Expert Syst. Appl.*, vol. 238, p. 122036, 2024.

[12] C. Bo, L. Zhang, X. Li, Q. Huang, and Y. Wang, "Who is using my phone? Transparent authentication based on motion sensor," in *Proc.ACM Int. Conf. Mobile Comput. Netw.*, 2013.

[13] Saxena, Shruti, et al. "Blockchain Enhanced Smart Healthcare Management for Chronic Diseases." Discover Computing, vol. 28, no. 1, June 2025, p. 112. https://doi.org/10.1007/s10791-025-09574-6.

[14] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Columbus, OH, 2014, pp. 1701–1708.

[15] D. E. King, "Dlib-ml: A machine learning toolkit," *J. Mach. Learn. Res.*, vol. 10, pp. 1755–1758, 2009.

[16] C. Lugaresi et al., "MediaPipe: A framework for building perception pipelines," *arXiv preprint arXiv:1906.08172*, 2019.

[17] R. O. Albuquerque, D. M. Amaral, W. F. Gomes, and R. T. de Sousa Junior, "A framework for continuous authentication based on touch´ dynamics biometrics for mobile banking applications," *Sensors*, vol. 21, no. 12, p. 4212, 2021.

[18] Charanarur, Panem, et al. "Design Optimization-Based Software-Defined Networking Scheme for Detecting and Preventing Attacks." Multimedia Tools and Applications, vol. 83, no. 28, Feb. 2024, pp. 71151–69.https://doi.org/10.1007/s11042-024-18466-8.

[19] A. F. Baig, S. Eskeland, and K. Groven, "Privacy-preserving continuous authentication using biometrics," *Computers & Security*, vol. 125, p. 103053, 2023.

[20] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Enhancing security and privacy in biometrics-based authentication systems," *IBM Syst. J.*, vol. 40, no. 3, pp. 614–634, 2001.

[21] M. Satyanarayanan, "The emergence of edge computing," *Computer*, vol. 50, no. 1, pp. 30–39, 2017.

[22] A. Akinsola, "Human-centered security: Bridging the gap between people and technology," *J. Cybersecurity Priv.*, vol. 3, no. 2, pp. 178– 191, 2023.

[23] Charanarur, Panem, et al. "Machine-Learning-Based Spam Mail Detector." SN Computer Science, vol. 4, no. 6, Nov. 2023, p. 858. https://doi.org/10.1007/s42979-023-02330-x.

[24] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA: MIT Press, 2016.

[25] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.