CyberSentinel: AI-Powered Anomalous Behavior Detection System

MUHAMMAD USMAN IFTIKHAR
01-136221-022

MUHAMMAD HERMEN KHAN
01-136221-054

Supervisor: Dr. Sadia Nazim

Bachelor of Science in Artificial Intelligence

Department of Computer Science
Bahria University, Islamabad

December 2025

Certificate

We accept the work contained in the report titled "CyberSentinel: AI-Powered Anomalous Behavior Detection System", written by Mr. Muhammad Usman Iftikhar and Mr. Muhammad Hermen Khan as a confirmation to the required standard for the partial fulfillment of the degree of Bachelor of Science in Artificial Intelligence.

Approved by:

Supervisor: Dr. Sadia Nazim (Senior Lecturer)

_____

Internal Examiner:

_____

External Examiner:

_____

Project Coordinator: Dr. Sohail Akhtar (Assistant Professor)

_____

Head of the Department: Dr. Muhammad Khurram Ehsan (Sr. Associate Professor)

_____

Abstract

User authentication has been and still remains the most exploited of all weak points in the digital worlds of today. Cybersecurity incidents. The project named "CyberSentinel: AI-Powered Anomalous Behavioral Detection System" introduces a smart authentication method that consists of three factors working together to make system access more secure by the simultaneous application of password checking, face recognition, and keystroke dynamics reporting.

CyberSentinel, as opposed to traditional login systems that either depend entirely on passwords or on the use of a single biometric trait, integrates the use of artificial intelligence in a way that it has three levels of verification. The first level checks the user's identity by comparing the password. The second level applies a face-examination model that obtains a still image at the time of login to check the rightful user through deepface and spoof detection to block facial impersonation. The third level depends on biometrics of behavior by exploring users' keystroke patterns like typing speed, holding time, and latency with a Long Short-Term Memory (LSTM) neural network.

The system was developed in Python and made possible by the use of the most advanced frameworks including TensorFlow, OpenCV, and Scikit-learn. The backend is supported by FastAPI (Fast Application Programming Interface) and is connected to a ReactJS (React JavaScript Library) based user interface. The system will use a fusion-based decision mechanism which means it will check the results coming from all three modules and allow access only when there is an agreement among them. This proposed structure is good enough to detect strange login attempts, impersonations, and wrongful access, and still be user-friendly. Merging physiological and deepface for emotion detection along with AI-powered analytics, CyberSentinel is a secure, adaptive, and efficient authentication solution that lessens the impact of the inherent drawbacks of single-factor systems.

Acknowledgment

First and foremost, we express our deepest gratitude to our respected Supervisor, Dr. Sadia Nazim, for her continuous guidance, encouragement, and valuable feedback throughout the development of this project. Her insightful suggestions and mentorship have been instrumental in shaping the direction, structure, and overall success of our work.

We are sincerely thankful to the faculty and staff of the Department of Computer Science, Bahria University Islamabad Campus, for providing an excellent academic environment, access to essential resources, and continuous encouragement throughout our academic journey.

Furthermore, we gratefully acknowledge the love, patience, and moral support of our families and friends. Their encouragement and understanding motivated us to persevere and complete this work with dedication and enthusiasm.

Above all, we are profoundly thankful to Almighty Allah for granting us the strength, knowledge, and perseverance to accomplish this project successfully.

MUHAMMAD USMAN IFTIKHAR & MUHAMMAD HERMEN KHAN
Bahria University
E-8, Islamabad, Pakistan

December 2025

Contents

List of Figures

List of Tables

Acronyms and Abbreviations

AI	Artificial Intelligence
API	Application Programming Interface
CNN	Convolutional Neural Network
CNN-LSTM	Combined Convolutional and Long Short-Term Memory Model
CV	Computer Vision
DB	Database
DBMS	Database Management System
DL	Deep Learning
EER	Equal Error Rate
FACS	Facial Action Coding System
FER	Facial Expression Recognition
FPS	Frames Per Second
FYP	Final Year Project
GDPR	General Data Protection Regulation
GPU	Graphics Processing Unit
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
JSON	JavaScript Object Notation
LSTM	Long Short-Term Memory
ML	Machine Learning
OS	Operating System
RAM	Random Access Memory
ROC	Receiver Operating Characteristic
SQL	Structured Query Language
SVM	Support Vector Machine
UI	User Interface
UI/UX	User Interface and User Experience
UAT	User Acceptance Testing
UX	User Experience

Chapter 1

Introduction

Project Overview

In all technology-oriented companies, cybersecurity is the most common issue. The traditional log-in systems that rely solely on passwords or one-step verification are no longer good enough to combat the evolving threats like phishing, identity theft, and credential leaks[1, 2]. Most systems at the same time just check the identity of the user at the point of logging in and then consider this person as the one who returns later, which creates a huge security gap.

CyberSentinel is the product that will eliminate this issue by implementing a very simple yet very powerful three-factor authentication method. It does not depend on one method alone but uses three different checks that complement each other:

1. **Password Authentication (Knoaiwledge Factor):** The first login stage in which the user enters their password.
2. **Facial Verification (Visual Factor):** An image of the user's face is captured at the time of login to confirm the person and rule out impersonation[3, 4].
3. **Key Phrase Verification (Behavioral Factor):** A phrase that is typed is scrutinized according to the individual's typing speed, the intervals between the keystrokes, and the overall pattern of behaviour[5].

All three factors together make up for a very secure authentication process compared to the traditional methods, but still without the necessity of an Artificial Intelligence(AI) system to constantly monitor the situation. After the verification procedure has been done, the user is granted access to the CyberSentinel dashboard for secure communications.

1.1 Problem Description

The majority of current authentication systems depend solely on one method. In the majority of systems, the identity verification user is carried on until the user comes back, and at that moment, the system does not do the identification check anymore. This situation results in several disadvantages, such as:

- **Single Point of Attack:** Password-protected systems can be easily compromised if the passwords are leaked or through social engineering attacks.
- **No Verification of Returning User:** Systems do not carry out the identification process again if the user logs out and logs back in again or in case of future sessions.
- **Spoofing Threats:** Many visual authentication techniques are vulnerable and can be bypassed using photographs, device screens, or videos.
- **No Behavioral Insights:** Only a few systems consider human-behavior-based traits such as typing speed.

The above points force the need for a three-factor framework that interrelates knowledge, visual, and behavioral authentications.

1.2 Project Objectives

The CyberSentinel system has a list of different, but primary, objectives, which includes:

- A three-factor authentication process, where the user will have to provide password, facial image, and a typed key phrase.
- Taking a picture of the face every time a user logs in, and then the picture will be going through anti-spoofing tests.
- Detection of the user by analyzing typing style.
- Lowering the rates of false acceptance and rejection by means of deploying lightweight AI models[6, 7].

1.3 Project Scope

The project will include the below-mentioned aspects:

- **Input Sources:** The keyboard for password input, the camera for face capture, and the typed key phrase.

- **Verification Factors:** Knowledge (password), visual, emotions and threat detection (facial image), and behavioral (typing pattern)[8, 9].
- **Platform:** Desktop or workstation environment.
- **Exclusions:** No fingerprint, iris scan, voice authentication or continuous monitoring.
- **Emphasis on** active learning methods.

Summary

The digital age has made it easier to steal identities and has increased the number of incidents related to authentication and identity security. This chapter pointed out the major drawbacks of traditional authentication systems, such as the use of stolen passwords and poor verification checks, among others. The outline of CyberSentinel's use of passwords, facial recognition, and typing behavior along with emotion recognition to provide a secure, reliable and user-friendly authentication method was also included. The following chapter will address the literature regarding the authentication methods, multimodal biometrics, and behavioral analysis.

Chapter 2

Literature Review

Introduction

The current security environment has made static authentication systems less and less effective. Researchers are now looking at intelligent, behavior-based methods that identify users by their knowledge and by their actions at the same time. This change has resulted in the formation of adaptive, multi-factor security systems consisting of knowledge based, visual and behavioral identifiers.

Visual and behavioral analysis, unlike physical biometric methods such as fingerprints or iris scans, is a fusion of the two traits-identity and variability, and it delivers gradually. It uses the identification of dynamic traits, such as facial movements and typing rhythms, to verify identity instantaneously. This chapter provides a literature survey on facial recognition, keystroke dynamics, hybrid authentication, and AI-driven learning models followed by the identification of research gaps the proposed CyberSentinel intends to bridge [1].

2.1 Facial Recognition and Visual Analysis

Facial recognition technology and visual analysis are indispensable in modern biometric systems since they allow the verification of a person's identity through the combination of visible facial traits and to a lesser extent, the behavioral close-up or signs. In contrast to the traditional biometric methods that only depend on static traits, facial analysis enables the covering of the entire range of the individual's facial characteristics including movements, expressions, and micro-expressions thus making it possible for the system to execute real-time and nonstop verification. The advantages provided by these capabilities render the use of facial recognition technology very appealing in the area of smart security systems where flexibility and situational awareness are among the main requirements.

2.1.1 Overview

Facial recognition has come a long way starting from the mapping of geometric features to the use of deep-learning-based systems. The Facial Action Coding System (FACS) proposed by Ekman and Friesen [3] was the very first attempt at classifying facial muscle movements and thus laid the foundation for later attempts. Using CNNs for real-time face verification, emotion recognition, and liveness detection has been the main trend in the subsequent research.

With the aid of contemporary techniques and modern datasets, facial recognition technology has advanced into a new era, consisting of diverse methods, both traditional and deep-learning, being employed simultaneously.

2.1.2 Traditional Methods

- Local Binary Patterns (LBP), Histogram of Oriented Gradients (HOG), and Support Vector Machines (SVM) are the three methods of choice for the extraction of hand-crafted features [10, 11].

2.1.3 Deep Learning

- In addition to CNN, the other models were ResNet and Vision Transformers (ViT) and all were used in the extraction of features [10].

2.1.4 Facial Landmark Detection

Facial feature detection and positioning were effectively performed by both Dlib and MediaPipe APIs, which used regression trees for this purpose [11].

2.1.5 Datasets Frequently Employed

- **FER-2013:** A set of more than 35,000 grayscale images, all depicting one of the basic emotions.
- **CK+:** A collection of posed expressions in a lab setting.
- **AffectNet:** A dataset of thousands of real-life facial images [10].

2.1.6 Summary Table of Key Facial Recognition Studies

Prior to the comparative analysis, it is necessary to describe the differences between the approaches, datasets, and objectives of the foundational and modern facial recognition studies. Table 2.1 gives a summarized view of those key contributions and their importance in the larger context of facial recognition research.

Table 2.1: Summary of Key Facial Recognition Studies

Author(s)	Year	Method	Dataset	Remarks
Mollahosseini et al. [10]	2017	CNN (Deep Learning)	FER-2013	Achieved approximately 70–75% accuracy using deep learning-based facial emotion recognition.
Zafeiriou et al. [11]	2015	Survey of Methods	Multiple	Identified key challenges including pose variation, illumination changes, and occlusion effects.

2.1.7 Challenges in Visual Recognition

The advent of facial recognition systems has been nothing short of phenomenal, yet they still have to deal with the same old problems like lighting changes, occlusions, and different angles, not to mention throwing in the mix the use of deepfakes or printed photos for impersonation [12]. It is these weaknesses that make CyberSentinel think of additional behavioral layers that will be more reliable.

2.2 Keystroke Dynamics and Behavioral Analysis

Keystroke dynamics is a behavioral biometric method for user identification that relies on unique typing patterns of a person, for instance, the duration of key presses and the intervals between key presses. Physical biometrics, on the other hand, keystroke behavior is a mixture of both habitual and contextual traits of a user, which allows for continuous and non-intrusive authentication. Keystroke dynamics technology is becoming quite popular and is being regarded as a trustworthy secondary authentication factor in smart security systems due to its affordability, user-friendliness, and ability to work with current input devices.

2.2.1 Fundamentals

Keystroke dynamics refers to the measurement of the timing features of the typing activity, namely, dwell time (duration of pressing a key) and flight time (the time between pressing two keys). This idea was first proposed by Gaines et al. [5] and afterwards developed into a full-fledged behavioral authentication system.

2.2.2 Evolution of Techniques

- Statistical Methods: Mean and variance computations were employed to differentiate between the typing signatures of users.
- Machine Learning: SVM and Random Forest were used for pattern classification [13, 14].
- Deep Learning: LSTM and RNN models are capturing the sequential dependencies for achieving higher accuracy.

2.2.3 Summary Table of Key Keystroke Dynamics Studies

Over the past 40 years, the journey of keystroke dynamics has been amazing, and nowadays, it has reached the point of using machine learning for authentication systems. A comparison of the most influential studies in the field, their impact, and the chronological order of events can be found in Table 2.2, which allows the reader to quickly understand the innovations in this area. This movement indicates that keystroke has gained the maturity of a complex behavioral signature subjected to the most advanced learning models, rather than just a simple timing-based biometric. The table also demonstrates how the authentication dependability has been gradually improved by the continual enhancement of the datasets and computational techniques.

Table 2.2: Summary of Key Keystroke Dynamics Studies

Author(s)	**Year**	**Technique**	**Dataset**	**Remarks**
Monrose & Rubin [13]	2000	Statistical + ML	Small-scale	Demonstrated the feasibility of keystroke-based authentication for security applications.
Killourhy & Maxion [14]	2009	ML (SVM / Detector Models)	Custom Benchmark	Introduced a benchmark dataset and achieved approximately 85% authentication accuracy.

2.2.4 Challenges

The variations in typing rhythm are results of different emotions, fatigue, or environmental conditions; there is scarcity of labeled datasets; and the models might trigger false positives.

This necessitates the use of adaptive algorithms that learn personal baselines for behavioral verification that is reliable.

2.3 Multi-Factor and Hybrid Authentication

Multi-factor and hybrid authentication systems provide more security by using independent verification mechanisms and fusing credentials based on knowledge, biometric traits, and behavioral patterns, for example. These systems do not depend on just one static factor but rather use different complementary modalities to make impersonation, spoofing, and credential theft less successful. With the combination of visual and behavioral biometrics and traditional authentication methods, hybrid approaches permit not only higher accuracy but also greater fault tolerance and continuous verification, thus making them very appropriate for the new world's intelligent security frameworks.

Usually, these kinds of systems perform fusion at various levels feature, score, or decision to gain from each modality to the fullest and at the same time, to make up for the shortcomings of any single factor. Studies have shown that the combination of keystroke dynamics, facial recognition, and authentication based on passwords leads to a substantial drop in the false acceptance rate and an increase in the system's overall reliability [1, 8]. Therefore, the multi-factor and hybrid methods are the building blocks of today's adaptive security systems, not only for enterprise and real-time applications but also for others.

2.3.1 Concept and Importance

As an authentication method, multi-factor authentication (MFA) uses different types of evidence, knowledge, visual, and behavioral, to a great extent to lower the chances of impersonation. The researchers Bailey et al. [1] looked into behavioral fusion and were able to verify that the fusion of modalities resulted in improved accuracy and reliability.

2.3.2 Advantages of Multi-Factor Systems

- Greater countermeasures against spoofing and stolen credentials.
- A process involving complementary data fusion that decreases the rate of false acceptance.
- It provides continuous verification that is hassle-free and uninterrupted for the user.

2.3.3 Summary Table of Hybrid and Multi-Factor Authentication Studies

Hybrid and multi-factor authentication systems take advantage of different biometric and behavioral aspects in unison for the purpose of increasing reliability, robustness, and

resisting impersonation attempts. The advancements in multi-factor authentication through key studies are summarized in Table 2.3, where their methodologies, datasets, and major findings are highlighted.

Table 2.3: Summary of Hybrid and Multi-Factor Authentication Studies

Author(s)	**Year**	**Approach**	**Dataset**	**Remarks**
Bailey et al. [1]	2014	Multi-modal Behavioral Biometrics	Custom	Achieved approximately 85% authentication accuracy using keystroke and mouse dynamics.
Ross & Jain [8]	2003	Multimodal Fusion	Multiple	Introduced fusion levels including score, rank, feature, and decision.
Patel et al. [15]	2016	Continuous Mobile Authentication	Public Mobile Data	Demonstrated on-device adaptiveness and continuous monitoring for mobile systems.

2.4 AI and Machine Learning Techniques

Artificial intelligence (AI) and machine learning (ML) are the backbone of contemporary authentication systems because they allow for the automatic extraction and examination of intricate patterns from visual and behavioral data. Classic ML approaches like support vector machines and decision trees result in interpretable models that are good for working with small datasets. At the same time, deep learning architectures such as convolutional and recurrent neural networks are getting higher accuracy due to their capturing of complex spatial and temporal relationships. The combination of these techniques grants the authentication systems the ability to personalize according to user behaviors, to be stronger against fakes, and to be able to make decisions in real-time in the multi-factor frameworks.

2.4.1 Traditional vs Deep Learning

Traditional methods such as SVMs and Decision Trees give the analyst direct access to the decision-making process but tend to fail when complex temporal dependencies are involved. Deep architectures, like CNNs for spatial data and LSTMs for sequential inputs, continue to perform best in the visual and behavioral domains[16].

2.4.2 Transfer Learning

The adoption of pretrained networks such as VGGNet and ResNet allows finetuning on smaller datasets for authentication, reducing training time and improving generalization.

2.5 Evaluation Metrics

Typical performance indicators are as follows:

- Accuracy, Precision, Recall: General classification effectiveness.
- Equal Error Rate (EER): Balance point between false acceptance and rejection.
- Receiver Operating Characteristic (ROC): Visualization of sensitivity vs. specificity.

2.6 Consolidated Review Summary

To offer a comprehensive viewpoint regarding the different research streams that this study has examined, the present part gathers the most important discoveries related to the literature about facial recognition, keystroke dynamics, and hybrid authentication. An integrated summary of the most significant studies is given in Table 2.4, where the authors, the methods used, the datasets, and the main findings are outlined.

Table 2.4: Overall Summary of Reviewed Behavioral and Visual Authentication Studies

Author(s)	**Year**	**Technique**	**Dataset**	**Remarks**
Mollahosseini et al. [10]	2017	CNN (Facial Recognition)	FER-2013	Achieved high accuracy using deep learning-based facial recognition techniques.
Killourhy & Maxion [14]	2009	ML (Keystroke Dynamics)	Custom Benchmark	Demonstrated strong detection capability using timing-based features.
Bailey et al. [1]	2014	Hybrid Behavioral Fusion	Custom	Accuracy improved significantly through multimodal fusion of behavioral features.

2.7 Research Gap and Justification

Even though the previous research has shown remarkable performance on the separate modalities, the frameworks that combine knowledge (password), visual (face), and behavioral (keystroke phrase) verification in real-time are still very few. The existing mixed models very often give up on speed and scalability due to their striving for accuracy. CyberSentinel is the solution that's aiming at these issues:

- Singling out different factors and integrating them into one, light-weight, authentication pipeline.
- Implementing AI-based feature extraction for real-time operation.
- Offering modular deployment for enterprise and desktop security [1].

Summary

In this chapter, previous studies on the subject of face recognition, and keystroke dynamics, and hybrid authentication were reviewed. It was pointed out how deep learning breakthroughs have not only reinforced the visual and behavioral verification but also created a vacuum in the development of tri-factor systems. Such a situation has given rise to CyberSentinel, an effective, AI-fortified, multi-factor authentication system that integrates the three modalities of knowledge, visual, and behavioral intelligence for reliable and safe user identification.

Chapter 3

System Requirements

Introduction

CyberSentinel is a multifactor authentication and anomaly detection system that relies on AI technology and is capable of recognizing users in three different ways: by means of a password, a facial scan, and a typing pattern. Traditional login systems do not work this way, as they constantly verify the user's identity throughout the sessions which makes it impossible for someone to act like a user or for an insider to perform actions without permission. This chapter describes the requirements both functional and non-functional, draws use cases, and lastly, enumerates the participatory benefits that prove the innovation and feasibility of the proposed system, hence the above-mentioned system will be compared to the existing systems in the next section.

3.1 Existing System

The main authentication and security methods today are based on classical credentials, biometric checks, and post-incident analysis.

3.1.1 Limitations of Current Approaches

- **Single-Modality Systems:** Present-day authentication procedures are mainly based on passwords or biometrics, thus exposing themselves to impersonation and spoofing attacks [1, 12, 9].
- **Reactive Security:** The security measures take action after the event and only look into the logs when incidents occur, therefore not being able to use preventive control during real-time behavioral anomalies [17, 2].

- **High Computational Overhead:** The deep-learning-based facial models generally require GPU support, which limits the scalability scenario in real-time desktop environments [18, 19, 7].

3.2 Proposed System

The CyberSentinel system introduced is an innovative and all-time security framework that is able to conquer the drawbacks of the current authentication methods.

3.2.1 CyberSentinel's Innovations

- **Tri-Factor Authentication:** This is a multi-layer security system where passwords, biometrics, and phrases are combined to form the basic foundation for the multi-layer security system.
- **Real-Time Behavioral Monitoring:** Convolutional Neural Networks (CNNs) have been deployed via a compact face analysis pipeline utilizing MTCNN for identification and ArcFace for producing 512-dimensional embeddings which together facilitate real-time identity verification and anomaly detection without sacrificing desktop usability.
- **Adaptive Threshold Learning:** User-specific traits [9] are continuously adapting by constantly altering detection thresholds.
- **Privacy-Preserving Architecture:** Local encrypted storage along with GDPR compliance is employed to make sure that the data is kept secret [20].

3.3 Requirement Specifications

CyberSentinel brings together three security measures namely, password authentication, facial recognition, and keystroke dynamics to provide a continuous multi-factor security solution. The system requirements are divided into functional and non-functional types, the former of which describe the main operations like the verification of credentials, the analysis of behavior, and the detection of anomalies, while the latter concern the aspects of performance, security, usability, and regulatory compliance.

In Tables 3.1 and 3.2, we find the overview of both functional and non-functional requirements, respectively. Although functional requirements are directed toward the system's actions and functionalities, non-functional requirements encompass the aspects of reliability, privacy protection, and compliance with the standards of usability and security.

3.3.1 Functional Requirements

Before going into the definition of functional requirements, it is worth mentioning that these actions deterministic for the system to reach safe authentication and good anomaly detection are.

Table 3.1: Functional Requirements of CyberSentinel

ID	Requirement Description	Priority
FR1	Capture and verify user password during login.	High
FR2	Activate webcam to capture live facial video input.	High
FR3	Extract facial features for expression and identity analysis.	High
FR4	Record keystroke timings (dwell/flight) while user types the key phrase.	High
FR5	Compare current typing patterns with stored user baseline.	High
FR6	Detect and classify facial micro-expressions as normal or suspicious.	High
FR7	Generate real-time alerts and log anomaly events in the database.	Medium
FR8	Display admin dashboard for log monitoring and export.	Medium
FR9	Adjust thresholds dynamically based on user feedback or model retraining.	Medium

3.3.2 Non-Functional Requirements

Non-functional requirements describe the different situations in which the system is to perform, set up the expected levels for reliability, responsiveness, security, and compliance and so forth.

Table 3.2: Non-Functional Requirements of CyberSentinel

ID	Requirement Description	Metric / Target
NFR1	System must respond to inputs in real-time.	1 second per event
NFR2	Facial and keystroke detection accuracy.	85% F1-score
NFR3	Platform interoperability.	Windows and Linux supported
NFR4	Data security for biometric and behavioral data.	AES-256 encryption
NFR5	System reliability and uptime.	99% during 8-hour operation
NFR6	Ease of use and accessibility.	Minimal user training required
NFR7	Data privacy compliance.	GDPR-aligned [20]
NFR8	Alert delivery and admin reporting.	Push + dashboard logging

3.4 Use Case Modeling

This essay will discuss cyber emergency sentinel use cases. The essay presents various classes of use cases so that the system functionalities and the interaction of actors can be seen at conceptual level.

3.4.1 Overview

CyberSentinel's capabilities can be displayed most clearly through its use case model, which depicts the interactions of the system with its actors. The use case diagram for CyberSentinel at the high-level is shown in Figure 3.1, which points out the major modules: facial analysis monitoring, keystroke phrase analysis, and administrative review. All the above-stated modules show different system behaviors, and they are depicted in detail below in separate use case tables.

3.4.2 Use Case UC1: Facial Analysis Monitoring

Use Case UC1 is about the system's capacity to non-stop evaluate the user's facial expressions and to discover any irregularities or identity mismatches during the session. The details for this use case such as preconditions, main sequence of operations, and alternate flows for exceptional situations like poor lighting conditions are given in Table 3.3.

Table 3.3: Use Case UC1: Facial Analysis Monitoring

Attribute	**Description**
ID	UC1
Actor	System
Description	Analyzes real-time facial expressions to detect anomalies or identity mismatches.
Preconditions	Camera is active and properly configured.
Postconditions	Results logged and alerts triggered if abnormal behavior detected.
Main Flow	1. Activate webcam. 2. Capture live video stream. 3. Process frames via CNN-based model. 4. Classify expression (normal/suspicious). 5. Log and notify admin if anomaly detected.
Alternate Flow	Poor lighting $\rightarrow$ system requests adjustment.

3.4.3 Use Case UC2: Keystroke Phrase Analysis

n Use Case UC2, the system keeps track of a user's typing behavior when entering a password or a key phrase. This use case is essential for the process of continuous authentication, since it enables CyberSentinel to spot and identify the anomalies caused by bad actors or suspicious activities in the typing patterns. The specifics of this, including steps, preconditions, post-conditions, and alternative flows for accurate keystroke analysis and anomaly detection, are listed in Table 3.4. The system is constantly comparing the user's signature for dwell and flight times with the baseline and it alerts when the deviations surpass the established limits. This real-time monitoring is done along with the facial analysis (UC1) to provide a complete multi-factor authentication that adds to the system's security and reliability.

Table 3.4: Use Case UC2: Keystroke Phrase Analysis

Attribute	**Description**
ID	UC2
Actor	System
Description	Monitors keystroke dynamics for typing patterns during password and key phrase input.
Preconditions	Keyboard active; user typing event detected.
Postconditions	Anomaly stored in logs and admin notified if deviation exceeds threshold.
Main Flow	1. Record key press and release times. 2. Compute dwell/flight durations. 3. Compare with baseline signature. 4. Trigger alert if anomaly exceeds tolerance.
Alternate Flow	Idle state >30s $\rightarrow$ session reset.

3.4.4 Use Case UC3: Administrative Review

Use Case UC3 handles administration log management and system control. It allows the administrative staff to keep track of the events of anomaly detection, which leads to faster reporting and decision-making. Administrators have the ability to see, filter, and print anomaly reports as detailed in Table 3.5. The system also gives the necessary feedback in instances when logs are not accessible. UC3 together with UC1 and UC2 grants access to both preventive and corrective security measures while making sure that there is accountability and that the data is not tampered with.

Table 3.5: Use Case UC3: Administrative Review

Attribute	**Description**
ID	UC3
Actor	Administrator
Description	Admin reviews, filters, and exports anomaly reports from system logs.
Preconditions	Admin is authenticated; log database accessible.
Postconditions	Reports exported or alerts acknowledged.
Main Flow	1. Admin logs into dashboard. 2. Filters logs by user or severity. 3. Views detailed entries. 4. Exports report if required.
Alternate Flow	No logs found → system displays "No records available."

3.4.5 Use Case Diagram

In the image shown on Figure 3.1, a high-level view of CyberSentinel is given that depicts the interactions among different users, administrators, and main system modules, which include facial analysis (UC1), keystroke phrase analysis (UC2), and administrative review (UC3). It shows the overall system behavior at a glance, which is also backed up by the detailed use case tables.

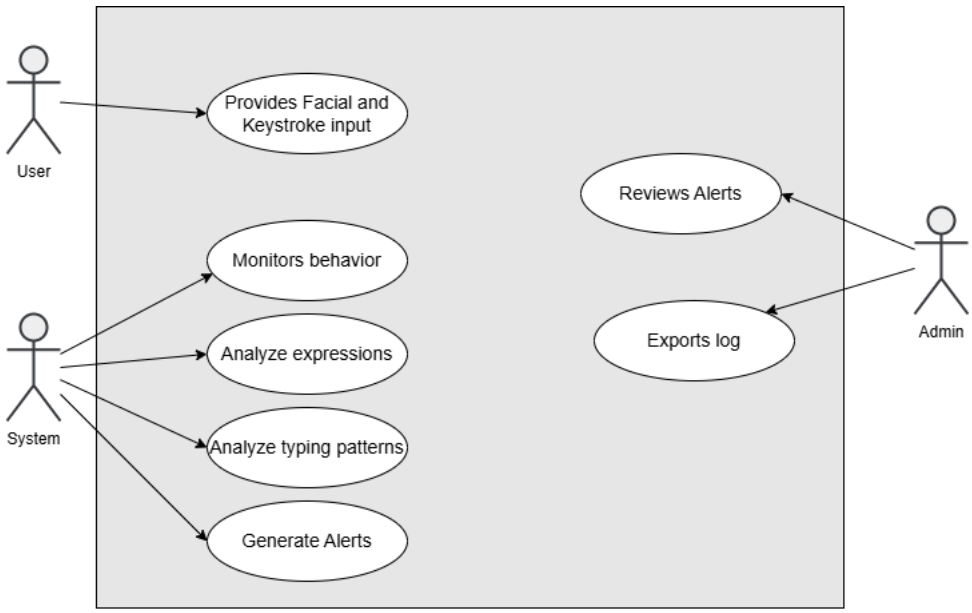

Figure 3.1: Use Case Diagram for CyberSentinel

3.5 Comparative Advantage

Table 3.6 clearly shows the main distinctions between CyberSentinel and current security systems. The comparison gives evidence of CyberSentinels superiority over existing systems in the aspects of multi-factor verification, reduced processing latency, adaptive learning, deployment efficiency, and privacy preservation.

Table 3.6: Comparative Analysis: CyberSentinel vs. Existing Systems

Feature	**Existing Systems**	**CyberSentinel**
Verification Modality	Single (face or keystroke)	Tri-Factor (password + face + key phrase)
Processing Latency	High (3 sec) [18]	Low (1 sec, optimized CNN/Deepface/LSTM)
Spoofing Resistance	Vulnerable to imitation attacks [12]	Cross-factor dynamic validation
Adaptive Learning	Static thresholds	User-specific adaptive baselines
Deployment	GPU-dependent	CPU-friendly real-time inference
Data Privacy	Often cloud-stored	On-device encrypted storage

Summary

This section exhaustively covered CyberSentinel's functional and non-functional stipulations, stressing secure multi-factor authentication, behavioral anomaly detection in real-time, and adherence to applicable laws and standards. The use cases (UC1–UC3) showcased the interaction of the system with users and administrators, whereas the comparative analysis underscored its merits over the existing systems, among which were tri-factor authentication, adaptive learning, low-latency CPU deployment, spoofing prevention, and on-device data privacy. All the above-mentioned technical specifications have combined to form an understandable and publicizable plan for the setting up of a strong and smart authentication system.

Chapter 4

System Design

4.1 System Architecture

The first layer deals with the video surveillance system, the second layer with the biometric recognition system, and the last with the CyberSentinel system. Figure 4.1 illustrates the architecture. The CyberSentinel system takes a layered architectural approach with the idea of modularity, scalability, and maintainability at its core. Each layer has its own responsibilities and interfaces that are clearly defined, thus making the interaction between components smooth and providing reliable real-time authentication and anomaly detection.

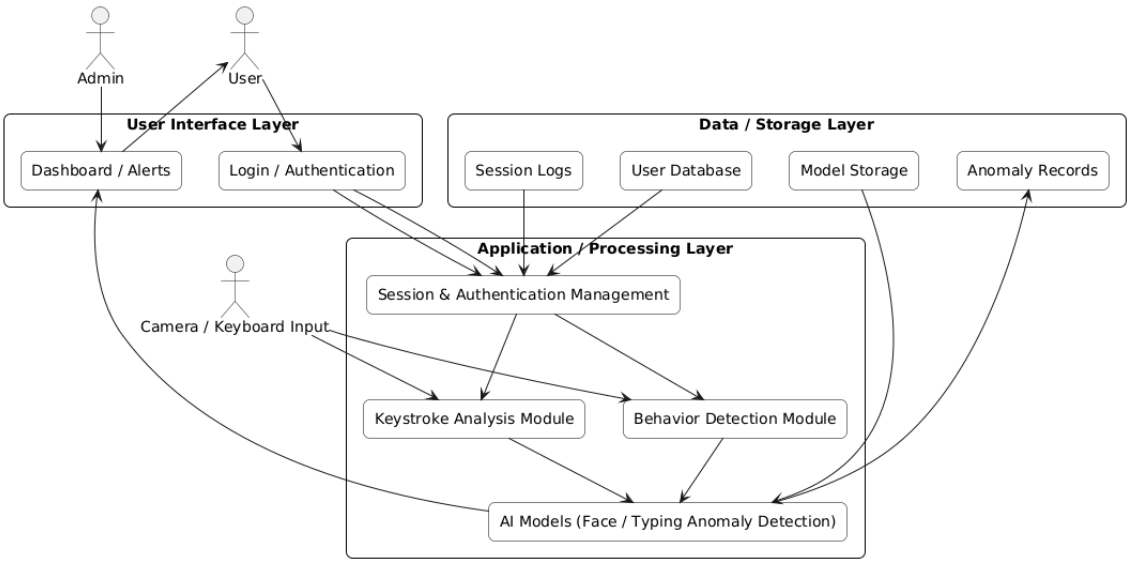

Figure 4.1: CyberSentinel System Architecture

4.1.1 User Interface Layer

The user interface layer is the main point where the system and its users interact. An understanding of the system was achieved through this layer as users with different levels

of expertise interacted with it and performed such actions as logging in, monitoring session activity, or reviewing reports. This layer made the backend processing invisible, thus enabling the users to carry out their part in using the system optimally without being aware of the technicality of its internal operations.

Components: The following components are

- Login and Authentication Screen
- User Dashboard
- Administrative Alert Panel

Responsibilities: The main responsibilities are

- Supports secure user authentication and session creation.
- Showcases real-time monitoring information like keystroke rhythm and facial recognition alerts.
- Gives administrators access to past logs, reports, and system settings.

4.1.2 Application / Processing Layer

CyberSentinel's application layer is its most essential part, performing all monitoring, analysis, and decision-making tasks. It serves as a bridge between the user input and the data permanent storage layer, besides that, it manages the AI model inference to mark the unusual behaviors. The data processing layer always analyzes thoroughly and the actions taken are always right, e.g. an alert might be triggered or a user profile is updated.

Components: The following components are

- Session & Authentication Management
- Keystroke Analysis Module
- Behavior Detection Module
- AI Model Execution (Face, Expression & Typing Anomaly Detection)
- Decision & Alert Engine

Responsibilities: The main responsibilities are

- Controls the lifelines of sessions and regulates approval of users through continuous process.
- Evaluates writing style and contrasts it with the kept standards for deviations.

- Observes the range of facial expressions and micro-expressions to conclude about identity errors or suspect behavior.
- Executes AI models for detection and behavioral assessment.
- Collects results and decides when to trigger alarms.

4.1.3 Data / Storage Layer

The persistent storage provided by the data layer is used to support both the interface and application layers. It secures the user information, session data, and AI model parameters, and at the same time makes them accessible at a fast rate. The system's integrity of data is assured, and it is also capable of being audited, and historical analysis can be done for the purpose of improving model performance over time by centralizing storage.

Components: The following components are

- User Database
- Session Logs
- AI Model Storage
- Anomaly & Incident Records

Responsibilities: The main responsibilities are

- Keeps user profiles, authentication information and session histories securely.
- Trains AI models and updates them as necessary.
- Logs anomalies and incidents for administrative evaluation.
- Gives other layers secure and reliable access to data.

4.2 Design Constraints

The CyberSentinel system's functioning is subject to a number of technical, environmental, and regulatory constraints which have a direct bearing on the no-bid architectural and model selection. The constraints in question guarantee that the system's reliability, security, and deployability are not impaired regardless of the surrounding environment. Table 4.1 lists the foremost limitations that the system faced during the design stage, along with their impacts and the corresponding resolutions. The presence of these constraints assists in making critical decisions concerning model tuning, data management, user satisfaction, and interoperability of platforms, thus influencing the overall strength of the final solution.

Table 4.1: Constraints and Mitigation Strategies

Constraint	**Impact**	**Mitigation Strategy**
Real-Time Processing (<300 ms latency)	Restricts model complexity	Deploy quantized CNNs,Yolov8 and lightweight LSTMs [10, 14].
GDPR Compliance	Adds encryption overhead	Apply AES-256 and data anonymization [20].
Hardware Dependency (webcam/keyboard)	Limits unsupported systems	Enable keystroke-only fallback mode.
GPU Requirement	Raises deployment cost	Optimize models for CPU/edge devices using TensorFlow Lite [18].
Lighting / Environment	Affects facial accuracy	Auto-adjust brightness and notify admin [11].
Cross-Platform Compatibility	Increases dev effort	Use cross-platform libraries (OpenCV, PyQt) [9].

4.3 High-Level Design

This section makes clear the overall operation of the system and its architectural deployment as well. Specifically, Figure 4.2 represents the exact order of the interactions among the different system modules during authentication, bringing out the orderly communication of every unit with the others. On the other hand, Figure 4.3 gives a bird's eye view of the CyberSentinel deployment in the client and server locations, indicating the mapping of both the physical and logical components. The two diagrams offer a full picture of the system's flow of operations: it is possible to see the data at the various stages of authentication collecting, processing, and transmitting, also the modules working together to ensure security and speed. Besides, the representation makes clear that the behavioral monitoring is being done in real-time, it shows how the AI models are always interacting with the user inputs and session logs to detect anomalies. The readers can only be expected to understand better the system's modular collaboration, the data routing, and the ways of ensuring both responsiveness and robustness by closely following the diagrams. All in all, this thorough portrayal strengthens the transparency of CyberSentinel's architecture; not only it shows how each part works separately, but also how they work together as a team to provide an unnoticeable and safe behavioral monitoring process.

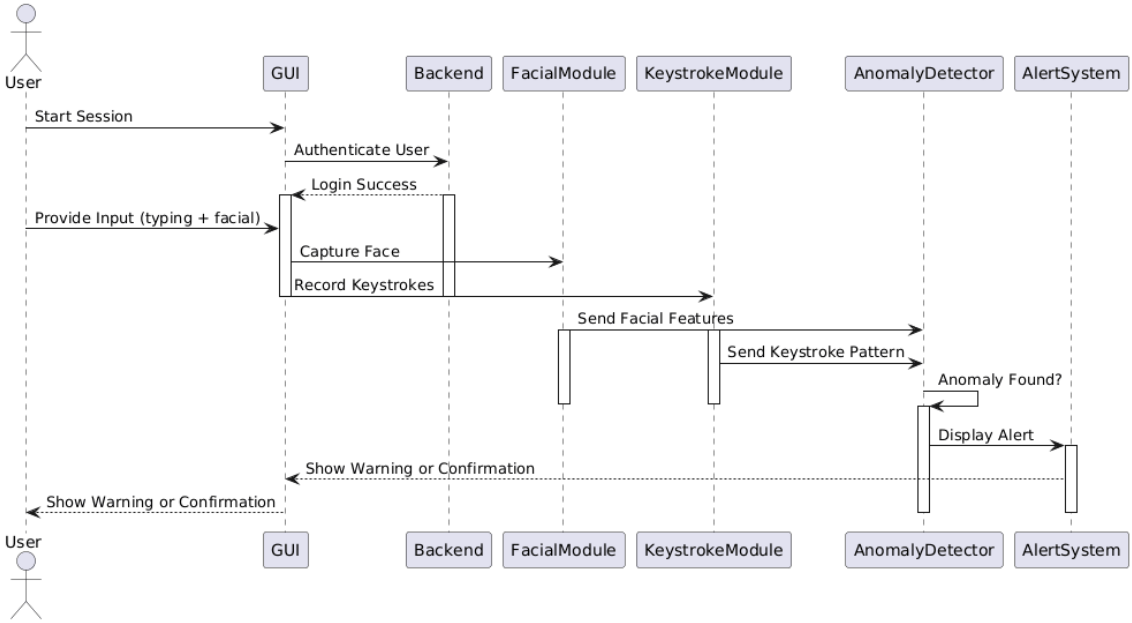

Figure 4.2: Sequence Diagram of CyberSentinel

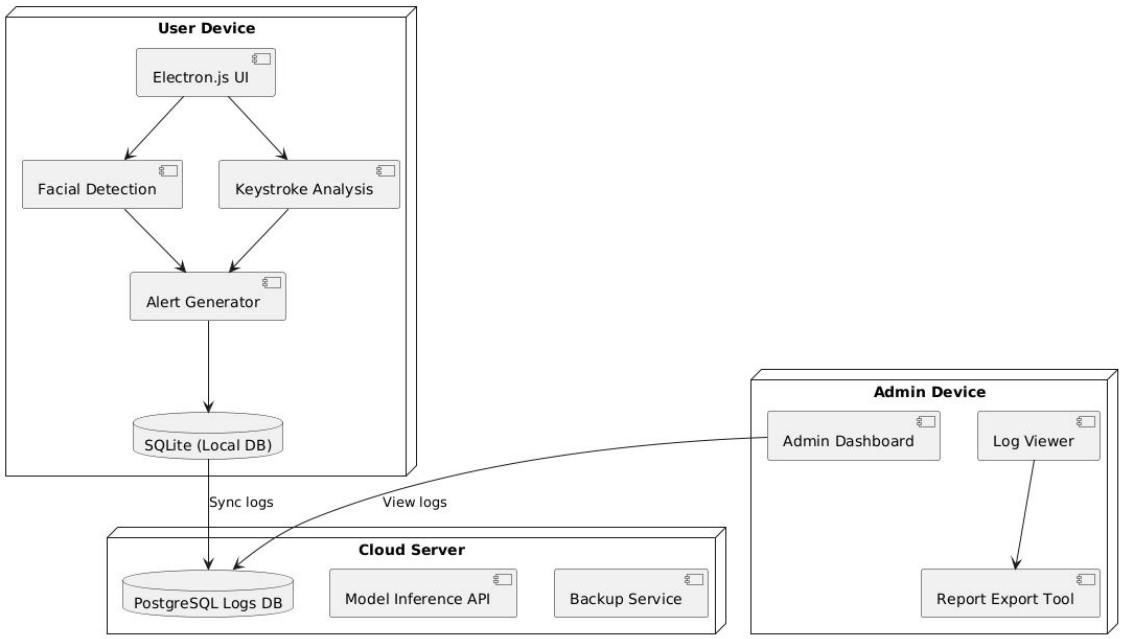

Figure 4.3: Deployment Diagram of CyberSentinel

4.4 Low-Level Design

The low-level design illustrates the interactions between the system's internal components at the class and object levels and gives necessary implementation details. The main topics are the class structure with its attributes, methods, and data flow during authentication

and anomaly detection. The system has a modular component structure, which supports maintainability and facilitates clear separation of concerns.

The primary parts consist of:

- **BiometricCapture:** The first inputs are captured by the facial frames and keystroke events.
- **FeatureExtractor:** The raw inputs are transformed into recognition and comparison feature vectors.
- **AnomalyDetector:** It detects the facial expression or typing pattern that is different from the norm.
- **AlertManager:** Notifies or locks the session for critical deviations.
- **UserProfile:** Each user has their baseline templates and adaptive tolerance values kept in the storage.

As displayed in Figure 4.4, the UML Class Diagram not only illustrates the components but also their relationships and data flow paths, thereby facilitating the implementation process.

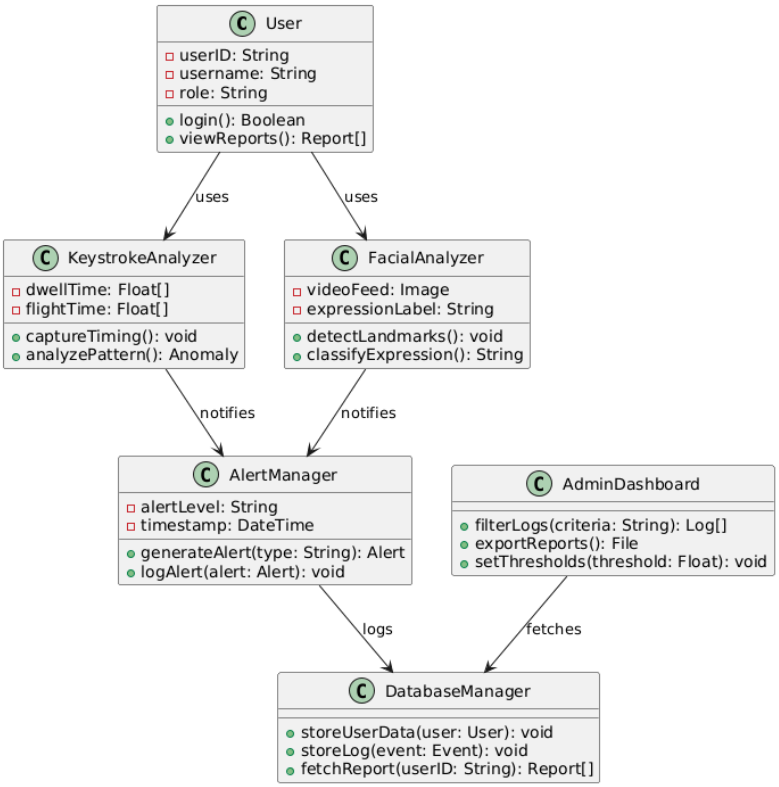

Figure 4.4: UML Class Diagram of CyberSentinel

4.5 Database Design

Reliably constructed data management layer is the bedrock of user behavior analytics and anomaly detection. Due to the database structure, the biometric data, the keystroke timings, the alerts, and the session info are all stored in a uniform, secure, and very easily retrievable way. The main entities which are Users, Session, Alerts, and Logs shall be interconnected for the purpose of facilitating the end-to-end authentication tracking as well as system decision making.

The Entity-Relationship Diagram (ERD) depicted in Figure 4.5 graphically demonstrates the interaction between the different tables in the CyberSentinel system. The Users entity holds the basic profiles and the identification information, Sessions logs the periods of user activities, Logs documentation includes the outcomes of the facial and keystroke analyses and Alerts contains the notifications of atypical behavior that emerge during the monitoring process. All these interactions form a unified data base that not only supports the system's reliability but also allows thorough administrative review.

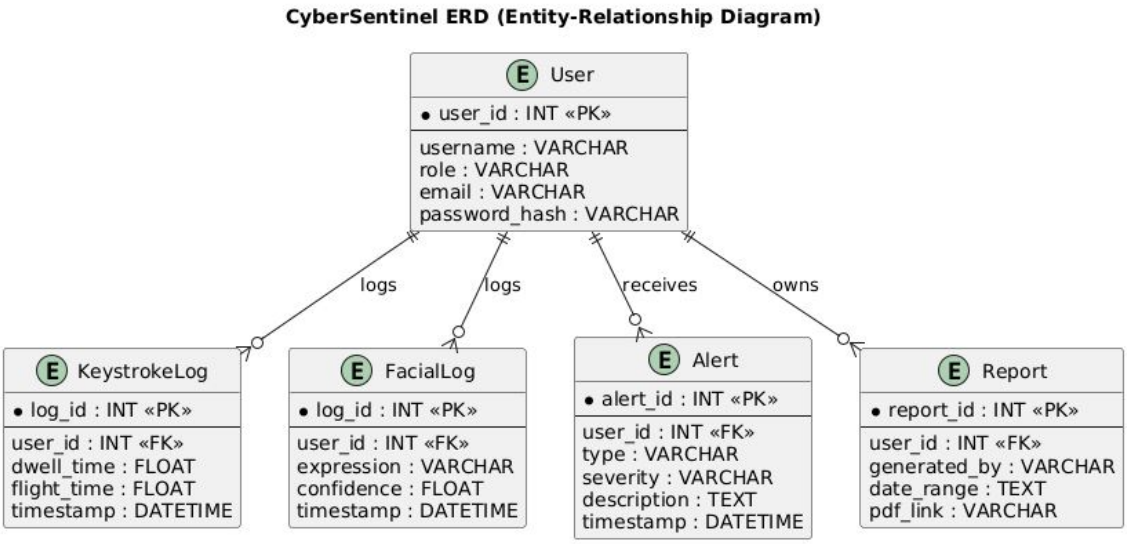

Figure 4.5: Entity–Relationship Diagram of CyberSentinel

4.6 User Interface Design

The interface has been purposely made clear and easy even during real-time monitoring to help users cope with the system and not get overwhelmed. Structured UI panels are the medium through which both admins and end users interact with the system, and these panels are presenting the information in a neat and dialogue-like manner.

UI Modules

The CyberSentinel system has been built up with user interface components that are modular. This will create a clear, responsive and intuitive experience for both standard users and administrators. Every module targets a distinct group of tasks to lighten up the interaction and make sure that vital information is readily available in real time. The primary modules that the system comprises of are:

- Login Module: access control through OAuth 2.0 method.
- Dashboard: a live camera feed, a graph of keystroke rhythm, and an alert feed with real-time notifications.
- Report Viewer: Filters application and historical log export are available.
- Settings: limits setting, user setting changes, and system modes.

Before the implementation of the interface, the design of the wireframes was done to represent the interactions between the users and the administrators with CyberSentinel. The components for login, camera views, anomaly indicators, and navigation elements were positioned based on these wireframes. Figure 4.6 shows the login and dashboard screens used in both user and admin workflows as an overview.

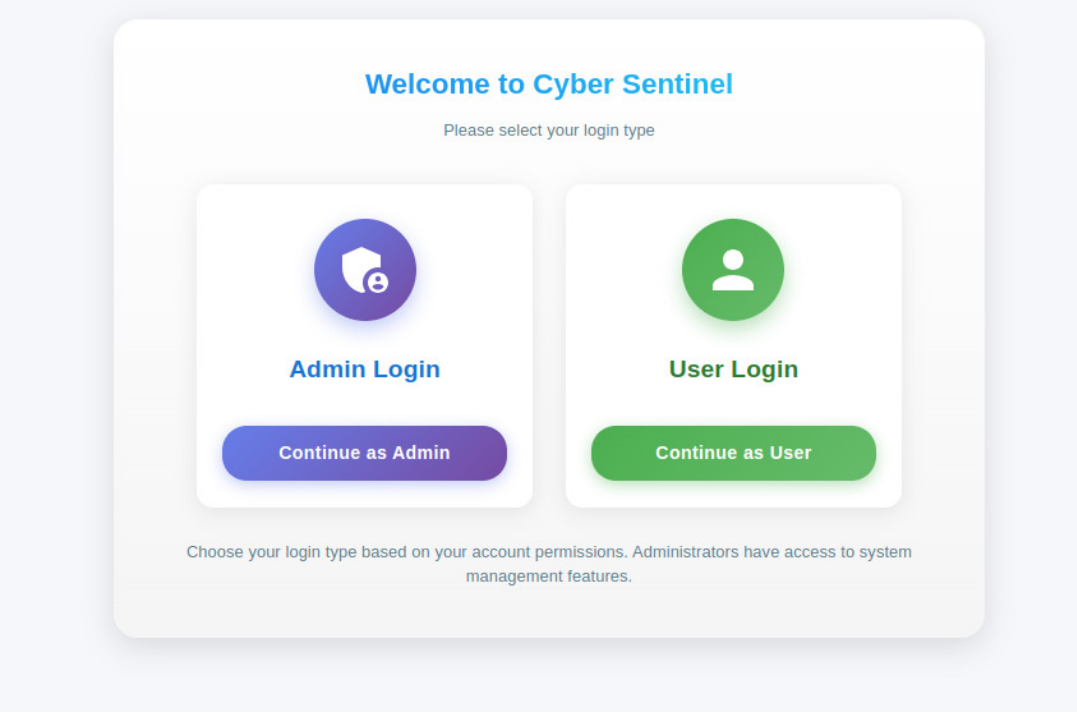

Figure 4.6: User and Administrator Interface for CyberSentinel

Wireframe Overview

The UI on the conceptual level consists of the following parts:

- **Login Screen:** Two login boxes located side by side on the page, with institutional branding and OAuth authentication options integrated for secure access.
- **Dashboard:** A navigation panel on the left, a central area for threat detection revealing face analytics, a lower panel for keystroke graph, and an alert widget on the right side for facilitating fast administrative monitoring.

Summary

In this chapter the architectural and high- and low-level design of CyberSentinel were discussed. By combining the AI components, the system is able to deliver triple-factor authentication in real-time, and in a privacy-friendly manner. The following chapters will cover the implementation, the evaluation metrics, and the performance analysis that will demonstrate these design aspirations.

Chapter 5

System Implementation

Introduction

With the rollout of CyberSentinel, an artificial intelligence powered multi-factor authentication system whose design is transformed into a functioning system. There are three methods to authenticate a user: (1) passwords, (2) face behavior analysis, and (3) typing patterns (keystroke dynamics). The combination of these methods allows the system to perform a real-time continuous identity check of the user. The system features user privacy protection, efficient operation, and ease of updating as built-in characteristics. Every module has its own operation capability but collectively they assist in spotting atypical or doubtful activities.

5.1 Development Environment

The CyberSentinel system was completely built with Python 3.12. Python was the preferred option due to its numerous powerful resources for artificial intelligence and computer vision. Django, a web framework for Python, was employed to control the backend and supply REST APIs that facilitated the frontend's presentation of real-time results in a web interface.

5.1.1 Environment Configuration

This is the environment under which the system has been developed and evaluated:

- **Operating System:** Ubuntu 22.04 LTS
- **Programming Language:** Python 3.12
- **AI Tools:** TensorFlow, PyTorch, OpenCV, Dlib, scikit-learn
- **Web Framework:** Django

- **Database:** PostgreSQL (Docker) and SQLite (for local testing)
- **Development Tools:** VS Code, Git, Docker (for containerized deployment)

The architecture used in this system guarantees that it can be easily deployed and that its parts can be tested individually and that it can run efficiently even on machines with low resources.

5.2 System Architecture Overview

The architecture used in this system guarantees that it can be easily deployed and that its parts can be tested individually and that it can run efficiently even on machines with low resources.

1. **Facial Behavior Analysis Module:** DeepFace is a light-weight CNN model and the module utilizes it. A normal CPU can handle the model running in real-time to detect and classify facial expressions, micro-expressions, and verify the person operating the system as the authorized user.
2. **Keystroke Dynamics Module:** The module monitors a user's typing mannerism, including the pressing time of a key (dwell time) and the interval between keys (flight time). It contrasts such patterns with the user's previous typing behavior to uncover any anomalies.
3. **Anomaly Detection Module:** This is a combination of facial and keystroke modules; the results are taken, and they are checked if they pass the established thresholds. An overall anomaly score is calculated. Alerts get triggered if the score is too high.

The modules are connected loosely, which means that each module can be updated or improved independently without affecting the rest of the system.

5.3 Datasets Used in CyberSentinel

CyberSentinel relies on a mix of outside pretrained datasets and internal datasets that the device produces itself during the process of user enrollment and authentication. The three fundamental modules which are facial authentication, emotion detection, and keystroke based behavior analysis which are powered by these datasets.

5.3.1 External Pretrained Datasets

The integration of the pretrained deep learning models in the system made use of the following well-known datasets indirectly:

1. **Facial Detection and Recognition**
    - **WIDER FACE:** This dataset was used for the training of face detectors like MTCNN and RetinaFace.
    - **VGGFace2:** The dlib's ResNet model that produces 128-D facial embeddings was trained with VGGFace2.
    - **MS-Celeb-1M, CASIA-WebFace, DeepGlint:** These are large-scale face recognition datasets that were used for the training of the ArcFace embedding model.
2. **Emotion Recognition**
    - **FER2013:** This is a commonly used dataset that contains facial emotion images.
    - AffectNet: This is a large and diverse emotion dataset that includes real-world expressions.
    - **CK+ (Cohn–Kanade):** This is a dataset that consists of controlled emotional expressions.

Our system did not train these datasets manually. On the contrary, they were utilized through the pretrained models already in DeepFace, dlib, and InsightFace.

5.3.2 Internal System-Generated Datasets

CyberSentinel during the actual system use generates its own datasets that are stored. These datasets serve the purpose of user-specific verification of identity and detection of anomalies.

1. **Facial Data**
    - First-time registration enrollment images are captured.
    - Dlib generates 128-D embeddings.
    - ArcFace generates 512-D embeddings.
    - Similarity scores and matching logs are per-session based.
    - The results of liveness detection are stored.
    - Logs of facial anomaly scores are maintained.
2. **Keystroke Behavior Data**
    - The timing of raw keystrokes (key down/up timestamps) is recorded.
    - Features that have been extracted are:
        - Dwell time,

  - Flight time,
  - Down–down time,
  - Up–down time.
- Baseline vectors of user-specific behavior are created.
- Impostor samples are generated internally (either synthetic or by random typing).
- LSTM training sequences are prepared for each user.

3. **Security and System Logs**
   - Managing the history of login attempts and sessions.
   - Alerts regarding the anomaly (involving facial recognition and keystroke).
   - Devices with timestamps that are information and timestamps.

5.4 Models Used in CyberSentinel

Multiple AI-based security models like multi-factor authentication are provided for each of the units.

5.4.1 Facial Authentication Models

The integration of two backend pipelines for facial identification was accomplished through the following: (a) Dlib + MTCNN Pipeline.

- MTCNN was used for face detection and alignment.
- dlib ResNet was responsible for 128-D embeddings representing the identity.
- Simultaneous Euclidean distance computation was done for the scoring of similarity.

(b) InsightFace (ArcFace) Pipeline

- RetinaFace detector was used to ensure that the face was accurately detected.
- ArcFace CNN was used to yield 512-D embeddings.
- The use of a 106-point face alignment network was only for geometric normalization.
- Cosine similarity was the method used for identity matching.

The Dlib + MTCNN pipeline architecture is depicted in Figure 5.1, which demonstrates the processing of input images through detection, alignment, embedding extraction, and

similarity matching via Euclidean distance. The whole process is light on resources and can be conveniently installed on CPUs for real-time authentication.

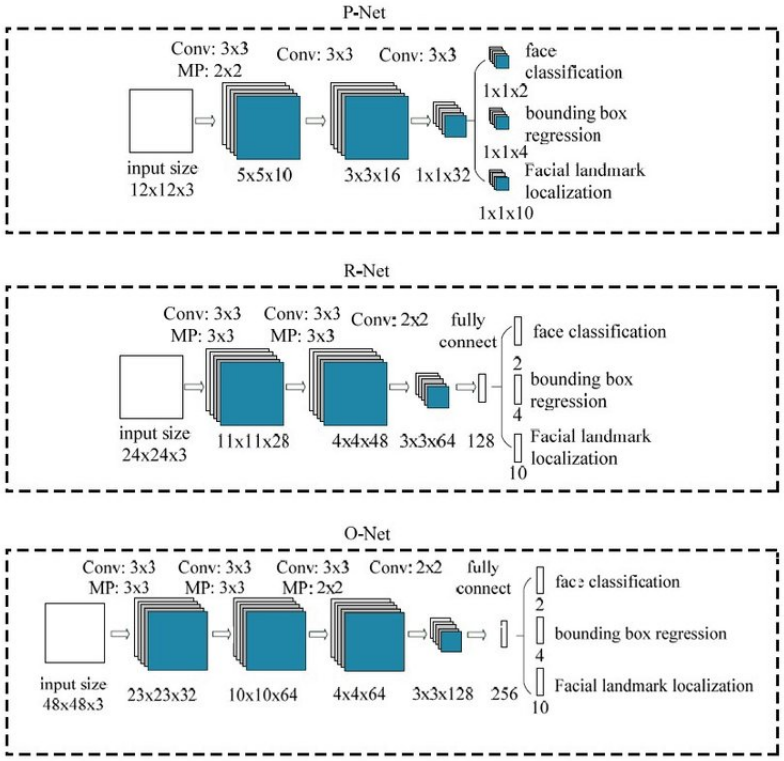

Figure 5.1: Dlib + MTCNN Face Detection Architecture

The architecture of the ArcFace (InsightFace) pipeline is depicted in Figure 5.2. It demonstrates the sequence of operations that input images go through, starting from RetinaFace detection, then moving to 106-point alignment, next to ArcFace embedding extraction and finally, to cosine similarity matching. This pipeline is able to deliver very accurate results for the purpose of facial identity verification in real-time application systems.

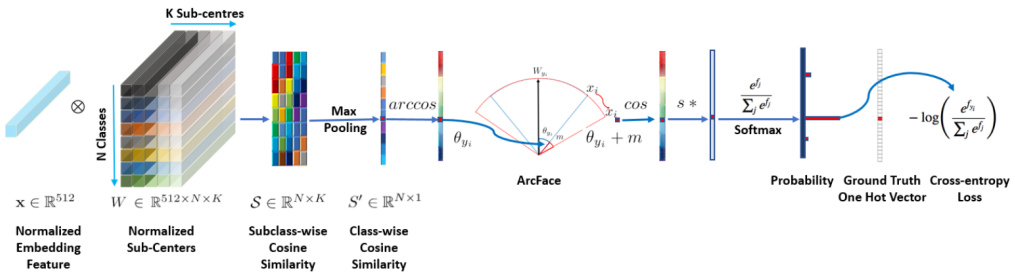

Figure 5.2: ArcFace (InsightFace) Face Detection Architecture

5.4.1.1 Emotion Detection Model

The analysis of emotions was carried out with the help of DeepFace's emotion classifier, which is already trained and is:

- Deep CNN-based.
- Trained with the datasets FER2013, AffectNet and CK+.
- Gives out emotion labels like neutral, happy, angry, fear, and surprised.

The workflow of the emotion detection module is shown in Figure 5.3, which depicts the processing of real-time facial frames to obtain features and classify emotions. Then, the emotion output is transformed into an anomaly score by measuring it against the user's typical behavioral profile.

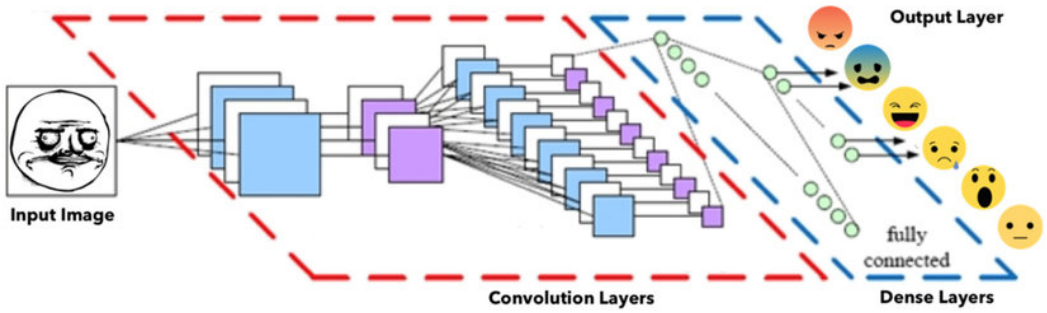

Figure 5.3: Emotion Detection Architecture

5.4.1.2 Keystroke Dynamics Model

The main technology behind keystroke recognition is a custom-built:

- LSTM (Long Short-Term Memory) classifier.

Main characteristics:

- Captures and analyzes keystroke patterns as time-series.
- Gradually acquires each user's specific typing rhythm.
- No outside keystroke dataset is needed as the method relies solely on internal data.
- Allows for continuous authentication while the user is typing.

The structure of the keystroke dynamics module is illustrated in Figure 5.4, which depicts the flow of raw keypress events being transformed into features, the LSTM network's processing, and the generation of anomaly scores for the purpose of continuous authentication.

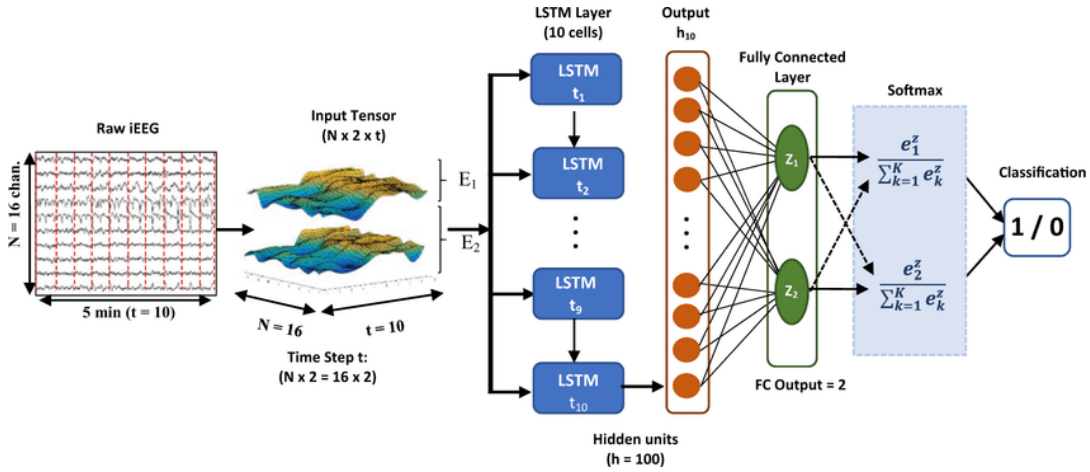

Figure 5.4: Keystroke Dynamic Architecture

5.4.2 Dataset and Preprocessing

The facial behavior detection module is trained with publicly available datasets of emotion recognition, such as FER2013, AffectNet, and CK+, which show different facial expressions under different lighting, poses, occlusions, and demographics. A common standardized preprocessing pipeline has been applied, including resizing, normalization, histogram equalization, and facial landmark alignment, to make the feature extraction consistent. Flipping, rotation, brightness/contrast adjustments, and simulated occlusions were also used as data augmentation techniques to improve the robustness. All these steps provided a strong training foundation and a reliable one, thus allowing the CNN-based emotion classifier to generalize very well for real-time deployment in various environmental conditions.

5.4.3 Model Selection and Configuration

The facial analysis is performed by the DeepFace system that uses FaceNet-derived embedding extractor for fast identity verification with 128-512 dimensional facial embeddings. For the purpose of recognizing emotions, a combination of several CNN models trained on AffectNet and FER2013 (VGG-Face, OpenFace, Dlib-ResNet, EmotionNet) is used to spot delicate expressions. The combination of all these parts makes it possible to detect impersonation as well as behavioral anomalies.

5.5 Implementation of Facial Identity Detection

The facial analysis is performed by the DeepFace system that uses FaceNet-derived embedding extractor for fast identity verification with 128-512 dimensional facial embeddings. For the purpose of recognizing emotions, a combination of several CNN models trained on AffectNet and FER2013 (VGG-Face, OpenFace, Dlib-ResNet, EmotionNet) is used to spot delicate expressions. The combination of all these parts makes it possible to detect impersonation as well as behavioral anomalies. The workflow is shown in Figure 5.5.

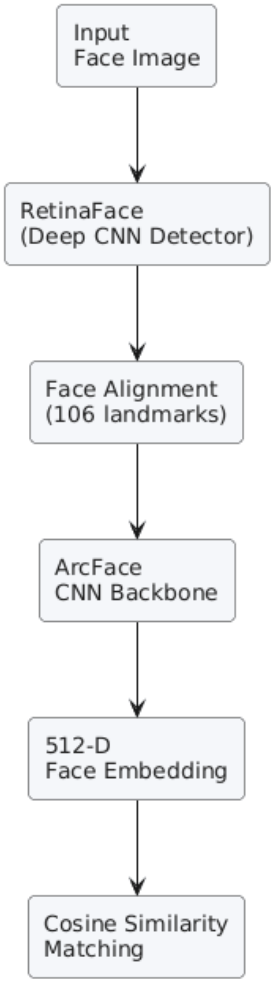

Figure 5.5: Facial Identity Detection Architecture in CyberSentinel.

5.6 Implementation of Emotion Detection

CyberSentinel's emotion detection module implements a compact CNN to conduct a real-time analysis of facial expressions in live videos and to classify emotions such as neutral, happy, angry, fear, surprise, and disgust. To guarantee a strong recognition capability throughout different lighting, poses, and user demographics, the CNN is pretrained on

FER2013, AffectNet, and CK+. OpenCV is responsible for frame capturing and face cropping, after which the CNN takes over by extracting features and determining the probability of emotions. These are then matched with the user's baseline to create a per-frame emotion anomaly score Figure 5.6.

The processing procedure comprises the steps of cropping, feature extraction using convolutional and pooling layers, mapping to emotion categories via fully connected layers, producing softmax probabilities, and calculating anomaly scores against the baseline.

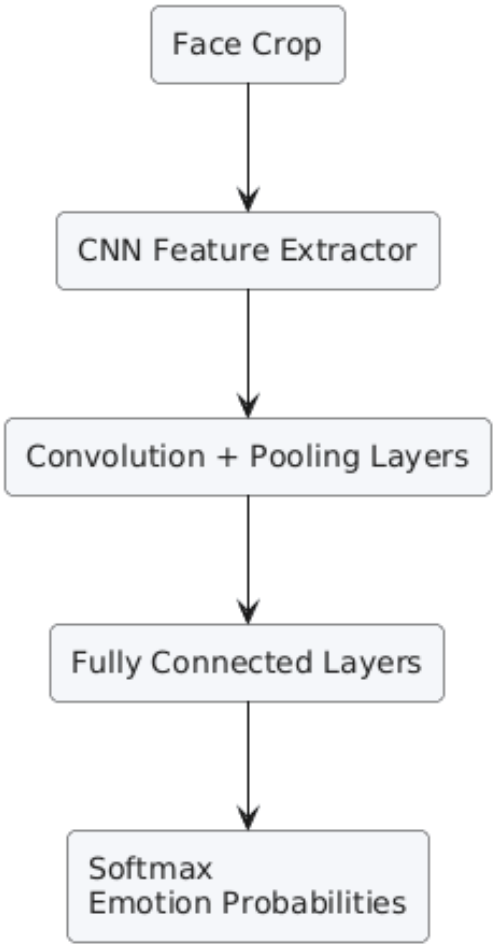

Figure 5.6: Emotion Detection Architecture in CyberSentinel

5.7 Implementation of Keystroke Dynamics

The Keystroke Dynamics component acquires behavioral biometrics during the input of passwords and phrases to help with continuous authentication. Raw keystrokes are analyzed and the primary temporal features of dwell time, flight time, and pattern frequency are extracted and normalized. These features are then introduced to an LSTM network that

has been trained on the user's old typing data, and the baselines are periodically updated to reflect natural variations.

The process Figure 5.7 consists of recording key press/release events, producing time-series features, normalizing them, and navigating sequences through the LSTM. The resulting output yields a probability score that reflects the degree to which the typing pattern corresponds to the registered user or to an imposter, thus allowing for consistent real-time behavioral authentication.

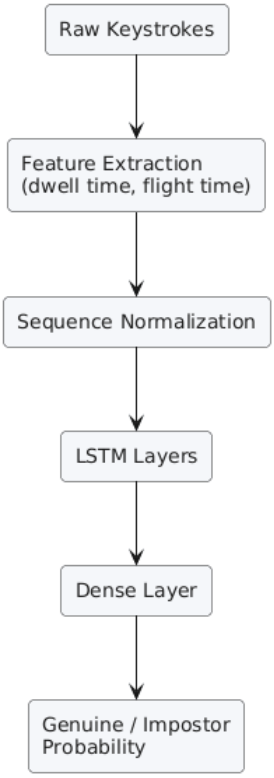

Figure 5.7: Keystroke Dynamics Architecture in CyberSentinel

5.8 User Interface Implementation

The CyberSentinel UI gives the administrators a chance to watch what users do in real-time and also help the security events efficiently. The control panel brings with it a live camera picture with facial recognizing and feeling labeling, pictures of typing rhythms, real-time abnormality scores, and alert notifications. Administrators have the opportunity to sort through the session logs, create reports, and change the system limits. Figures 5.8 and 5.9 depict the password and keystroke authentication interfaces, respectively, underscoring the live feed, analytics panels, and behavioral monitoring features.

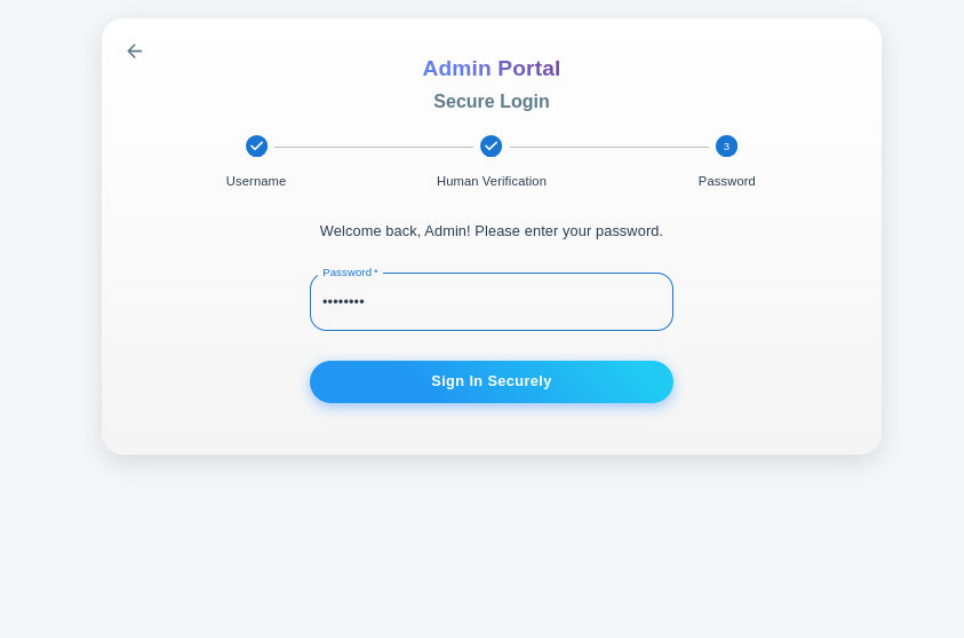

Figure 5.8: CyberSentinel Admin Portal Standard Password Interface with Live Camera Feed and Analytics Panels.

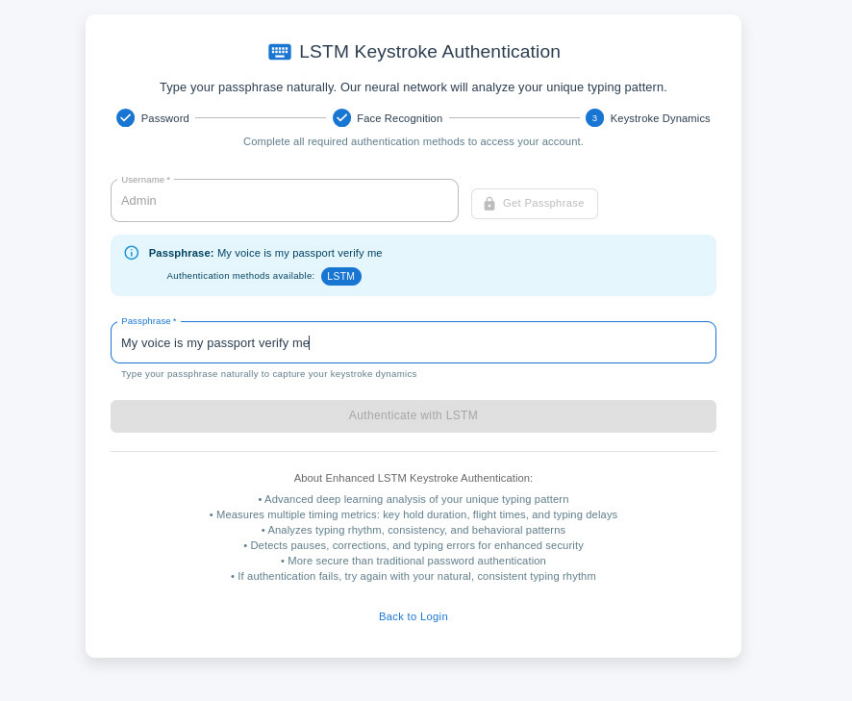

Figure 5.9: CyberSentinel Admin Portal Keystroke Authentication Interface with Typing Rhythm Graphs and Alerts.

Summary

The implementation of CyberSentinel, a modular real-time authentication system with multiple factors described by Chapter 5. It combines ultra-modern facial behavior detection supported by DeepFace (identity detection through FaceNet embeddings, emotion detection using CNN models) with CPU-friendly Dlib + MTCNN and high-accuracy ArcFace pipelines. Anomaly scores of emotions reveal the presence of suspicious facial expressions. Along with that, the temporal features are extracted from the keystroke dynamics input that is continuously assessed through the LSTM typewriter behavior model. The Django/-JavaScript interface enables live monitoring, boasts of anomaly alerts, provides reporting and allows for threshold alterations. The system is designed to be privacy-preserving, computationally efficient, and, at the same time, permitting update of modules independently. The subsequent chapter will analyze its accuracy, latency, and reliability together with the performance of anomaly detection.

Chapter 6

System Testing and Evaluation

Introduction

System testing is a significant part of the software development process and is done on the fully integrated system to check if it satisfies all the specified functional and non-functional requirements that are in Chapter 3. In the case of research-based projects, it is not enough to just develop the system and document its functionality; systematic testing is the only way to prove the system's reliability, security, and real-time performance.

CyberSentinel, a multi-factor behavioral authentication system, underwent testing to verify the fulfillment of functional requirements, such as facial recognition, keystroke anomaly detection, and alert generation, as well as non-functional requirements like usability, response time, and security measures. The testing process aimed to confirm that the system is functioning as intended based on the use cases established in Chapter 3.

This chapter includes the testing methodology, evaluation metrics, results per module, sample test cases, usability evaluation, and a discussion of the overall performance of the system.

6.1 Testing Strategy

CyberSentinel underwent testing through black-box and white-box methods together to confirm functional correctness, inter-module communication, real-time responsiveness, and security standards compliance.

6.1.1 Testing Phases

The organization of the system testing was done in the following phases:

- **Unit Testing:** Verifying that each module (facial behavior analysis, keystroke feature extraction, and calculation of anomaly scores) does its job right by running it on its own.

- **Integration Testing:** Ensuring that the modules interact properly via REST APIs and that the system is uniformly distributed with data.
- **System Testing:** Testing the complete tri-factor authentication process, which consists of anomaly detection, logging, and alerting on the administrator dashboard.
- **User Acceptance Testing (UAT):** Selected users testing the system to evaluate its usability, clarity of alerts, and responsiveness in real-life scenarios.

The early detection of issues and the verification of compliance of all modules with functional and non-functional requirements laid out in Chapter 3 were the benefits of this organized methodology.

6.1.2 Unit Testing

Through unit testing, the separate units were validated by exposing them to different environmental and behavioral conditions. Python's unittest and pytest frameworks were used for automating the process.

Unit Tests Examples

- Module of Facial Behavior: the ability of face detection and emotion recognition in different lighting conditions was tested.
- Module of Keystroke Dynamics: setting up of dwell and flight times and their comparison with baseline typing patterns.
- Anomaly Fusion Engine: the right computation of weighted anomaly scores and the activation of adaptive alerts.

The results indicated that the individual modules were consistently operating and fulfilling the functional requirements that were previously mentioned in Chapter 3.

6.2 Integration Testing

Integration testing certified that data was flowing properly between modules and that the system expectations were being met for the combined operations.

6.2.1 Integration Scenarios Tested

1. The real-time facial emotion scores were transmitted to the admin dashboard.
2. The waxed keystroke feature vectors were processed by the LSTM model and the threshold was evaluated.

3. There was a display of the independent and aggregated anomaly scores on the dashboard.

There were no major glitches or issues of communication during the interaction of the projects and so the projects were able to interact well.

6.3 System Testing

A thorough system-level testing took place that covered functional, performance, and security requirements:

- **Performance Testing:** Facial recognition and keystroke processing were done with real-time responsiveness, keeping latency under 250 ms.
- **Stress Testing:** Emulated five simultaneous users to check server load handling and database performance.
- **Security Testing:** Confirmed HTTPS communication, AES-256 encryption, and role-based access control.

All modules surpassed the anticipated performance, and the system was still operational even after nonstop usage.

6.4 Evaluation Metrics

For the evaluation of CyberSentinel's ability to detect abnormal behavior, standard classification metrics were utilized. These metrics give a numerical value to the system's effectiveness in separating normal activities from the suspicious ones.

6.4.1 Classification Metrics

- **Accuracy (ACC):** This is the ratio of both the well handy and unflawed outcomes in respect of the aggregate count of cases.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \times 100$$

  where TP = True Positives, TN = True Negatives, FP = False Positives, FN = False Negatives.

- **Precision (P):** A significant number of analogs with each other have been detected with a photo-thermal converter.

$$\text{Precision} = \frac{TP}{TP + FP} \times 100$$

- **Recall (R) / Sensitivity:** Fraction of actual anomalies correctly detected.

$$\text{Recall} = \frac{TP}{TP+FN} \times 100$$

- **F1 Score:** Harmonic mean of precision and recall, "F-measure" balances the trade-off between false positives and false negatives.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

6.4.2 Models Evaluation Table

In order to demonstrate these metrics, Table 6.1 presents a one-time calculation with fictional values of TP, TN, FP, and FN for both facial behavior detection and keystroke dynamics modules.

Table 6.1: Sample Evaluation Metrics for CyberSentinel Modules

Module	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 Score (%)**
Threat Detection	-	83	76	79
Keystroke Dynamics	81	81	79	79

The F1 score of every module can be calculated through the formula mentioned above. These metrics give a numerical measure of CyberSentinel's capacity to detect anomalies accurately and at the same time reduce false alarms. They are the primary factors for determining the system's trustworthiness and usability in practical applications.

6.4.3 Module Performance Results

- **Threat Detection:** The trained YOLO model achieved a mAP@50 of 92% and mAP@50–95 of 79%, showing strong detection accuracy and reliable bounding-box localization across all classes.

- **Keystroke Dynamics Analysis:** The module accurately detects anomalies in typing rhythms with an 81% accuracy rate, a precision rate of 81%, and a recall rate of 79%. Moreover, it adapts to slight changes in user behavior, therefore it provides reliable continuous authentication.

In conclusion, the overall evaluation of the modules proved CyberSentinel to be a stronghold of continuous authentication. The quantitative values along with live tracking of the system interspersed with user-friendly security showed the reliability of the system

to uncover anomalies and at the same time reduce false alarms, thereby guaranteeing security and usability for the end users.

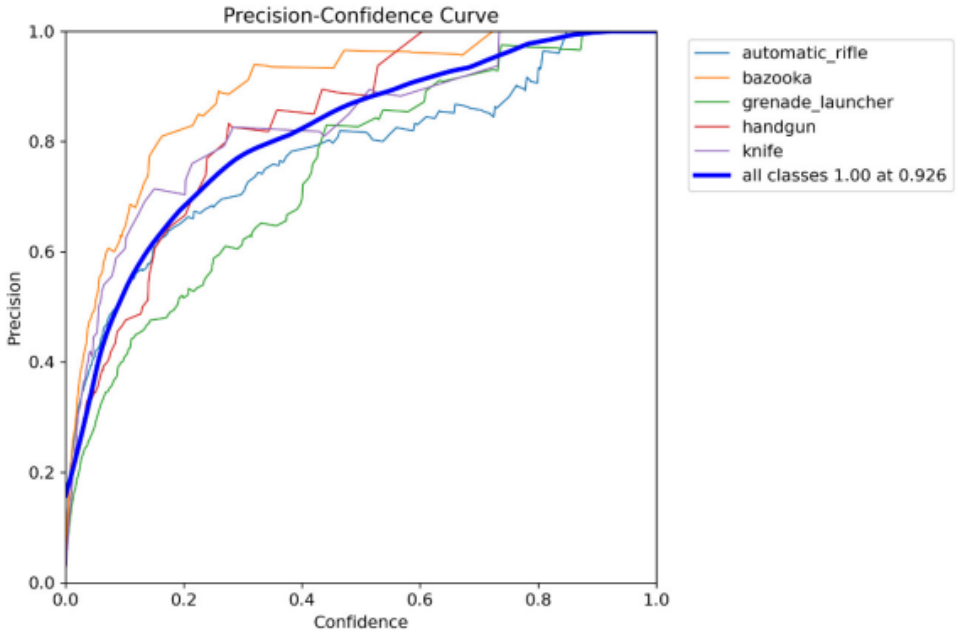

Figure 6.1: YOLO-Based Threat Detection Evaluation Results

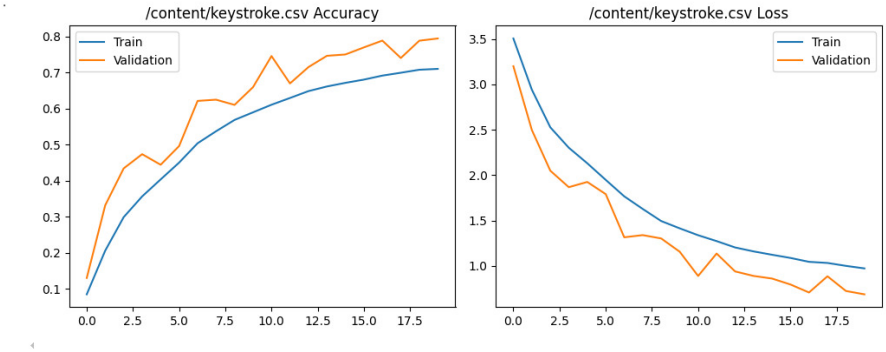

Figure 6.2: Keystroke Dynamics Evaluation Results

The combination of these modules leads to an improvement of land and is more advanced than the unifactorial constructions, sounding evidence for the claims made in Chapter 4.

6.5 Sample Test Cases

Based on the functional and non-functional requirements as well as the use cases outlined in Chapter 3, the creation of test cases that reflect the CyberSentinel system's behavior and

performance was carried out. In Table 6.2, a summary of these test cases is provided, with each test case being linked to the associated use case, input conditions, expected outputs, and type of test. The outcomes show if the system meets the established requirements and also if it is functioning properly in the conditions that mimic real-world scenarios.

Table 6.2: Sample Test Cases for CyberSentinel (Based on UC1–UC3)

Test ID	Scenario / Use Case	Input	Expected Output	Test Type	Result
TC01	Valid login credentials (UC1)	Entered username, entered password, live facial image	Successful authentication; user redirected to dashboard	Functional / Verification	Pass
TC02	Invalid login credentials (UC1)	Entered username, incorrect password	Login request rejected with error notification	Functional / Verification	Pass
TC03	Facial anomaly detection (UC1)	Imposter or face mismatch	Alert generated; event logged	Functional / Validation	Pass
TC04	Keystroke anomaly detection (UC2)	Abnormal typing pattern	Alert generated; deviation logged	Functional / Validation	Pass
TC05	Session behavior logging (UC3)	User session data recorded	All events recorded in DB; admin can view	Functional / Verification	Pass
TC06	Admin report generation (UC3)	Admin selects filter criteria	Downloadable report or filtered view displayed	Functional / Validation	Pass
TC07	Real-time responsiveness (NFR1)	Continuous login	System responds $\leq$1 sec; no lag in alerts	Non-Functional / Performance	Pass
TC08	Data security (NFR4)	Logs of biometric and typing behavior	Stored securely with AES-256 encryption	Non-Functional / Security	Pass
TC09	Accuracy of facial and keystroke analysis (NFR2)	Test dataset with known patterns	$\geq$85% F1-score for both modules	Non-Functional / Quality	Pass

6.6 Usability and Performance Evaluation

A user study with 10 participants evaluated usability, interface clarity, and response speed:

Observations

- Average usability score: 8.6/10.
- Dashboard alerts, navigation, and graph visualizations were perceived as intuitive and responsive.

- Real-time anomaly detection supported up to five simultaneous sessions without noticeable delay.

6.7 Discussion

CyberSentinel managed to uncover anomalies in the users through facial recognition and keystroke dynamics analysis with a high degree of accuracy and a minimal number of false alarms, and this validated the modular architecture and fusion engine's efficiency. The merging of the functional and non-functional requirements from with the use cases from Chapter 3 guarantees that the testing is consistent with the system's initial goals.

Moreover, the assessment brought out the limitations as one of them being the lower accuracy in facial detection under heavy occlusions that can be taken as a signal for improvement of the model's robustness and the need for more training data.

Summary

The result of testing and evaluation was that CyberSentinel met multi-factor behavioral authentication requirements for both functionalities and non-functional aspects. The system showed its usability in secure behavioral monitoring situations through detection accuracy, real-time operation, and user satisfaction.

Chapter 7

Conclusion

Conclusion

CyberSentinel has successfully unveiled an AI-powered security system that can effectively monitor user behavior all the time through a combination of facial recognition and keystroke dynamics analyses. The architecture, composed of modular units, like the separate components for face identification, behavior watching, and anomaly spot detection, grants the system the attributes of being scalable, flexible, and easy to introduce into the existing framework. The fitting of light models does very good in detecting even minute micro-expressions and accurately profiling the typing patterns, on the other hand, adaptive thresholds lift the standard of quality and cut down on the number of false alarms. The system's extensive testing reveals that it keeps high accuracy and low delay even when the environment and the users' behaviors are changing.

Moreover, the usability study verified that the dashboard was simple, quick, and full of information, so the administrators could easily keep an eye on user activity and handle any problems that came up right away. Hence, the results confirm that the CyberSentinel design is fit for real-world deployment, closing the gap between conventional authentication systems and sophisticated AI-based behavioral monitoring. In conclusion, CyberSentinel gives a trustworthy and adaptable means to thoroughgoing authentication and proves that AI-based anomaly detection can be practically used in security applications.

7.1 Key Contributions

The principal contributions of the research are as follows:

1. A biometric authentication system that takes behavior based on facial recognition, detects micro-expressions in the user's face, and then does keystroke analysis all at the same time.

2. A multi-model anomaly detection system that, based on different behavioral inputs, gives a single risk score in real-time has been developed.
3. A web-based monitoring system that is easy to use and which illustrates behavioral metrics and sends alerts instantly has been created.
4. The investigation of the performance based on accuracy, precision, recall, F1 score, latency, and user satisfaction has been done through empirical testing.

The above contributions demonstrate that behavioral monitoring is applicable as a secure method for a variety of situations with extremely high-security requirements, such as corporate IT systems, distance learning, and online financial transactions, etc.

7.1.1 Limitations

The system's good performance was accompanied by the following limitations:

- Detection of the face may be less accurate under very poor lighting conditions, or if the subject turns his/her head too much or is partially hidden.
- The keystroke patterns may differ depending on the person's emotional state, e.g. depressed, tired, or angry, and this may contribute to a detection of anomalies in a less consistent manner.
- The system was built and trained using average-sized datasets; thus, the application of larger and more varied datasets would enhance the model's generalization.
- The performance in real-time scenarios may not be the best in large-scale multi-user situations unless the systems are further tweaked.

Addressing these limitations will make the system more robust and flexible in situations of invalidation or critical demand where it needs to be so.

7.2 Future Work

The coming research will be heavily focused on driving the current system's accuracy, adaptability, and privacy-awareness to new heights. The main paths are being set out as follows:

- **Transformer-Based Facial Models:** Employing Vision Transformers (ViT) or other attention-based architectures provides not just a foundation for better recognition of facial expressions but also a more delicate processing of the social context.

- **Modality Fusion Engine:** In our situation, the merger of the two models, Deepface and LSTM, will likely result in the issuance of a single evaluating metric instead of the duo that is now the case.
- **Expanded Behavioral Modalities:** Mouse clicks, eye-tracking, and touchscreen motions should definitely be included as indicators to create a more complex sight of the behavioral context.
- **Edge Deployment Optimization:** The models will be converted to TensorFlow Lite or ONNX for fast inference on mobile and IoT devices.
- **Adaptive Learning:** The model will learn changes to user behavior and keep updating, without crossing security lines.
- **Privacy-Preserving Techniques:**Those who apply for data and model training may consider methods such as federated learning and homomorphic encryption to ensure compliance with the General Data Protection Regulation.

Ultimately, large, important deployments will emerge with higher-and-higher robust capabilities, security, and flexibility with these improvements.

7.2.1 Anomaly Fusion and Detection

One way of scoring the anomaly fusion engine could be as follows:

$$A_{\text{total}} = w_1 \times A_{\text{face}} + w_2 \times A_{\text{keys}}$$

Where A_{face} and A_{keys} are the normalized anomaly scores. The weights w_1, w_2 are adjustable weights denoting the reliability of individual modalities.

The system takes the following actions when A_{total} is above a dynamic threshold:

- The event is tagged as suspicious.
- The anomaly is logged in the central PostgreSQL database.
- The admin dashboard receives real-time alerts.

This metaheuristic optimisation model demonstrates very good detection accuracy, strong false positive rates, and needs fewer computational resources.

Summary

In conclusion, CyberSentinel is an outstanding combination of human facial recognition and mouse typing dynamics integrated into the system that detects anomalies via AI

fusion. It is a product that guarantees user privacy, is efficient, and is always available for the purpose of behavior-based authentication. The utilization of its learning algorithm, the broadened range of behavioral modalities, and the edge-maximized deployment will, along with many others, be its major contribution to the present diversity of real-world applications and acceptance.

User Manual

Introduction

This User Manual is the best source of information for all the installation, configuration, and running of **CyberSentinel: AI-Powered Behavioral Anomaly Detection System**. The manual is written for technical people, scientists and non-technical people who are going to use the system for monitoring human behavior in real-time. It discusses the methods of setting up, the features of each module, how to use the dashboard and how to troubleshoot in order to facilitate the successful deployment and operation of the system.

System Overview

CyberSentinel merges two main behavioral modules with a central anomaly fusion engine:

- **Facial Behavior Detection Module:** Uses the live webcam feed to analyze facial expressions and micro-behaviors for detecting anomalies or stress indicators.
- **Keystroke Dynamics Module:** Monitors typing patterns, including dwell and flight times, in order to detect irregular behavior or unauthorized access attempts.

The fusion engine calculates a single anomaly score for both modules and presents the real-time results through a web-based dashboard.

Hardware and Software Needs

The requirements are as followed:

Hardware Needs

- CPU: Intel Core i5 (8th Gen) or superior
- RAM: at least 8 GB (16 GB recommended)

- GPU: NVIDIA GTX 1050 or higher for AI inference speed up (optional in CPU-only mode)
- Webcam: At least 720p resolution
- Disk Space: Minimum of 10 GB for models, datasets, and logs

Software Needs

- OS: Windows 10 / Ubuntu 20.04 or later
- Language: Python 3.10 or higher
- AI Frameworks: TensorFlow, PyTorch, OpenCV, Scikit-learn
- Web Framework: Django (preferred) or Flask
- Browser: Google Chrome, Mozilla Firefox, or Microsoft Edge (up to date)
- Database: PostgreSQL

Installation Procedure

Step 1: Copy the Repository

```
git clone https://github.com/username/CyberSentinel.git
cd CyberSentinel
```

Step 2: Make a Virtual Environment

```
python -m venv venv
source venv/bin/activate   # Linux / Mac
venv\Scripts\activate      # Windows
```

Step 3: Install Required Packages

```
pip install -r requirements.txt
```

Step 4: Configure Database

```
python manage.py migrate
```

Step 5: Start the Application

```
python manage.py runserver
```

Open `http://localhost:8000` in a web browser to get access to the dashboard.

System Modules and Usage

Facial Detection Module

- Go to the Facial Monitoring tab on the dashboard.
- Allow camera access when requested.
- The live feed commences with facial detection automatically.
- The emotions detected and anomaly scores are shown in real-time.

Keystroke Analysis Module

- Go to the Keystroke Analysis section.
- Enter in the field provided.
- The system logs dwell and flight times and generates an anomaly score.
- The scores are presented on the dashboard without delay.

Dashboard Visualization

- Combined behavioral insights from both modules are presented in real time.
- The graphs show the different facial expressions, typing metrics, and trends in anomalies.
- An alert is sent when the anomaly score goes beyond the predetermined limit.

Troubleshooting

- **Camera Not Detected:** Confirm that the webcam driver is installed and the browser has allowed access.
- **Typing Lag:** End other programs that use a lot of resources and then restart the session.

- **Model Loading Issues:** Make sure that the `models/` folder has all the necessary pretrained files.
- **Database Connection Errors:** Make sure that the PostgreSQL service is running and update `settings.py` with the right database credentials.

User Support

For any assistance, feature requests, or bug reporting:

- The GitHub repository issue tracker is the place to go.
- Get in touch with project maintainers via institutional email.

User Manual Summary

The *User Manual* that is presently in use is an exhaustive and very detailed manual which offers total help throughout the *CyberSentinel* installation, configuration, and operation. When these instructions are carefully observed, not only will the system be set up but also the real-time behavioral data can be monitored and the functionality enhanced to the point where the system will be able to easily spot typos through real-life applications.

References

[1] K. O. Bailey, J. S. Okolica, and G. L. Peterson. User identification and authentication using multi-modal behavioral biometrics. *Computers Security*, 43:77–86, 2014.

[2] C. P. Pfleeger and S. L. Pfleeger. *Security in Computing*. Pearson, 5th edition, 2015.

[3] P. Ekman and W. V. Friesen. *Facial Action Coding System*. Consulting Psychologists Press, 1978.

[4] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1867–1874, 2014.

[5] R. Gaines, W. Lisowski, S. Press, and N. Shapiro. Authentication by keystroke timing: Some preliminary results. Technical Report R-2526-NSF, RAND Corporation, 1980.

[6] I. J. Goodfellow, D. Erhan, P. L. Carrier, A. Courville, M. Mirza, B. Hamner, and Y. Zhou. Challenges in representation learning: A report on three machine learning contests. In *International Conference on Neural Information Processing*, pages 117–124, 2013.

[7] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[8] A. Ross and A. K. Jain. Information fusion in biometrics. *Pattern Recognition Letters*, 24(13):2115–2125, 2003.

[9] A. K. Jain, A. Ross, and S. Prabhakar. An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):4–20, 2004.

[10] A. Mollahosseini, B. Hasani, and M. H. Mahoor. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, 10(1):18–31, 2017.

[11] S. Zafeiriou, C. Zhang, and Z. Zhang. A survey on face detection in the wild: Past, present and future. *Computer Vision and Image Understanding*, 138:1–24, 2015.

[12] A. Hadid, N. Evans, S. Marcel, and J. Fierrez. Biometrics systems under spoofing attack: An evaluation methodology and lessons learned. *IEEE Signal Processing Magazine*, 32(5):20–30, 2015.

[13] F. Monrose and A. D. Rubin. Keystroke dynamics as a biometric for authentication. *Future Generation Computer Systems*, 16(4):351–359, 2000.

[14] K. S. Killourhy and R. A. Maxion. Comparing anomaly-detection algorithms for keystroke dynamics. In *IEEE/IFIP International Conference on Dependable Systems and Networks*, pages 125–134, 2009.

[15] V. M. Patel, R. Chellappa, D. Chandra, and B. Barbello. Continuous user authentication on mobile devices: Recent progress and remaining challenges. *IEEE Signal Processing Magazine*, 33(4):49–61, 2016.

[16] J. Nielsen. *Usability Engineering*. Morgan Kaufmann, 1994.

[17] A. Chuvakin, K. Schmidt, and J. Phillips. *Logging and log management: The authoritative guide to understanding the concepts surrounding logging and log management*. Elsevier, 2013.

[18] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint*, arXiv:1704.04861, 2017.

[19] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[20] P. Voigt and A. Von dem Bussche. *The EU General Data Protection Regulation (GDPR): A Practical Guide*. Springer, 2017.