IJEMD-CSAI, Special Issue (2026) https://doi.org/10.54938/ijemdcsai.2026.04.2.664

IJEMD

International Journal of Emerging Multidisciplinaries: Computer Science and Artificial Intelligence

Research Paper

Journal Homepage: www.ojs.ijemd.com
ISSN (print): 2791-0164 ISSN (online): 2957-5036

Behavioural Biometric Authentication via Mouse Dynamics

Madelyn Soon Zhi Cian [1], Ashley Bernandette Nolidin [1], Daphne Chua En Xi [1], Dominic Lee Ming Li [1], Ng Jien Tze [1], Toh En Suen [1], Yap Hui Ee [1], Siva Raja Sindiramutty [1]

1. School of Computer Science, Taylor's University, Subang Jaya 47500, Selangor Malaysia

1. Abstract

Authentication is an essential component to a secure system by verifying the credentials of a user before the access right was granted to access system resources legitimately. However, previous research proved that as the computer intrusion techniques continue to advance, traditional authentication methods such as passwords and one time password (OTP) gradually reveal their security weakness, and it is no longer enough to preserve the security of the system. Therefore, a more secure authentication method, behavioural biometrics, was introduced to thwart intruders and enhance the security by verifying users' unique behavioural pattern that was difficult to mimic. This project proposes an authentication system that integrates mouse dynamics behavioural biometrics and traditional password authentication to detect suspicious login attempts. The user needs to predefine three mouse click positions as part of their user profile. The same mouse click position and sequence must be performed during authentication before proceeding to username and password entry. Successful login requires both the mouse click coordinates and user's credentials to match the stored profile. The proposed system was experimentally evaluated by comparing the accuracy, security and performance with traditional password authentication. The mouse dynamic biometric method has an overall accuracy of 92%. Although it performs higher latency and lower throughput than password authentication, it provides stronger security and greater resistance to brute-force attack. These results show that the proposed authentication system provides a more secure and effective approach to user authentication.

Keywords:
Behavioural biometrics, Mouse dynamics authentication, Password authentication, Multifactor authentication, Security, Privacy

Email Addresses : 0380967@sd.taylors.edu.my, 0366190@sd.taylors.edu.my, 0368861@sd.taylors.edu.my, 0366027@sd.taylors.edu.my, 0367302@sd.taylors.edu.my, 0380692@sd.taylors.edu.my, 0380744@sd.taylors.edu.my, siva.sindiramutty@taylors.edu.my

2. Introduction and Problem Identification

Authentication is a process in cybersecurity that can ensure encryption or security by verifying the identity of someone or something is what they claimed they are (Barney, 2021). It is a necessary process during accessing a system or application, it can check whether the personal credentials someone entered are equivalent to the credentials stored in the database of the specific system. For example, during a login process in an application, a person will enter the personal credential that he or her has entered previously during the sign-up process where they need to enter their personal credential when creating an account for the application, if the credentials match, the person will gain access to the application. By having authentication, it can ensure encryption, security and prevent different types of cyber-attacks.

To prevent intrusion, behavioral biometrics can provide protection and efficiently prevent hackers from attempting unauthorized access. Behavioral biometrics is an authentication method that authenticates users based on their patterns or interactions, it can be a personal unique pattern that could be difficult to mimic. For example, authenticate the users by their typing rhythm, mouse path patterns, screen-touching patterns or movements and their walking patterns while holding their devices. This authentication method checks whether the behavior that the user performs matches with the pattern saved by the user in the database, if it was the same, access will be granted to the specific application, if not, access will be denied by the application (Holdsworth & Kosinski, 2025; Ghosh et al., 2022).

Currently, with only common authentication methods like password, fingerprint, face recognition and OTP (One-Time-Password) to authenticate a user does not provide efficient protection to the user. Based on the article stated, the password authentication method provides a weak security protection to the users as it could be simple, easy to guess and easily leaked. Hackers can use cyber-attack techniques like brute force attack, dictionary attacks and password spraying method to guess the password of the target user and perform cyber-attack (Swisher, 2024; Humayun et al., 2021). Therefore, the fingerprint method is not safe as well, as fingerprints can be stolen or faked using mold as the fingerprint of the target user can be obtained in fingerprint templates if it was leaked. Moreover, storing the fingerprint templates in the database of the system raises privacy concerns as it may happen with the leakage of templates or misuse issues (Bhansali, 2024; Linqiang et al., 2024).

Moreover, OTP (One-Time-Password) authentication method is a method that can verify a user based on the temporary passcode sent to the user while the user needs to enter the temporary passcode during the login process to access the specific application. OTP still does not provide the best protection for user access, as there were many types of OTPs like SMS-based and email-based, for SMS-based OTP, the possible risk could be SIM swap attacks or man-in-the-middle attack which attackers may transfer the victim's mobile number to a new SIM card that is under the attacker's control, intercepting OTP. For email-based OTP, it can be more secure than SMS-based, but the OTP can still be stolen by the attacker if the email account was being hacked by the attacker. Therefore, OTP can be inconvenient, as there would be a delay occurring when sending the OTP message to the user or unsuccessful sending the message to the user (Wilson, 2024). Hence, by applying behavioral biometric authentication, it can ensure that the

access can be granted for the real user only, preventing hackers from gaining access with leaked credentials.

In the real-world, there were security incidents that highlighted the security issues of traditional authentication methods. In the year 2022 to 2023, the LastPass Breach happened as hackers attacked into LastPass, they gained access and stole the important internal information from the LastPass developer's account, then they used the account to access the LastPass cloud storage. Then, the hackers stole both encrypted and unencrypted data like credentials, notes, URLs, IP addresses and customer details. Although the credentials were secured and encrypted, they were exposed hence leading to increased risk (Kost, 2023; Humayun, Jhanjhi, Almufareh, et al., 2022). Based on this breach, it shows that when the password manager provider is attacked by hackers and stolen their information from the cloud storage, it can deal a huge impact on not only the company itself, but the users will be affected too. Having a stronger authentication method or a combination of two or more authentication methods would have a better protection to prevent this particular incident from happening again, implementing strong password policy and dynamic biometrics for the internal administrator could be better for security purposes.

Other than that, in an incident reported by an article, a transport company in United Kingdom named KNP Logistics Group where has ran 158 years in business was hit by a ransomware attack, where hackers gained access by weak or easily guessed password, leading to around 700 employees losing job from the company and the company unable to recover from the attack (Bilton, 2025). Based on this incident, it is proved that the common authentication methods can be easily stolen by hackers or by cyber-attacks. This could be due to the lack of continuous monitoring and relying on only one authentication method without additional verification methods. Thus, behavioral biometrics can be used as an additional verification method to authenticate users continuously to prevent imposters after entering only personal credentials like password after login process.

These real-world incidents showed that the insecurity of only using traditional authentication methods and the reason for these incidents may be caused by the underuse of behavioral biometrics or did not combine behavioral biometrics and traditional authentication methods effectively. With the usage of the combination of traditional authentication methods and behavioral biometrics, for example the user can only access the specific application with username, password, and behavioral biometrics, it can effectively address the issue of security vulnerabilities. This leads to the importance of applying behavioral biometrics with other authentication methods to achieve complete encryption and protection to the user, and prevent being hacked or stolen credentials.

3. Literature Review and Gap Analysis

3.1 Literature Review

3.1.1 Overview of Behavioural Biometrics

Authentication plays a critical role in safeguarding the confidentiality, integrity, and availability of system resources. Contemporary user authentication mainly relies on passwords or a combination of factors in order to authenticate and identify users, mainly used in log-in events (Subash A. , Song, Lee, & Lee, 2025). In recent years, behavioural biometrics, which capture distinctive patterns of human behaviour, have emerged as a compelling approach for continuous and passive authentication, demonstrating promising security benefits.

Behavioural biometric systems operate by capturing traits such as typing rhythm, mouse movement characteristics, gait patterns, touchscreen interactions, and behavioural responses. These characteristics are inherently more difficult to replicate which offers enhanced usability and security by reducing reliance on static credentials. The evolution of dynamic biometrics highlights a shift toward capturing real-time behavioural signals to support continuous authentication, enabling systems to reassess identity persistently rather than only at login events. Behavioural biometrics solutions employ artificial intelligence (AI) and machine learning (ML) algorithms to examine user behaviour patterns and construct models of a user's customary behaviour (Holdsworth & Kosinski, 2025). Across keystroke, mouse, and gait-based systems, recent research emphasises non-invasiveness, scalability, and compatibility with continuous authentication. These modalities allow authentication without requiring users to provide highly personal or intrusive data, strengthening user acceptance and operational feasibility.

3.1.2 Advancements made in Keystroke Dynamics, Mouse Dynamics and Gait Biometrics

Significant research advancements have been made across behavioural biometric modalities.

Keystroke dynamics is widely regarded as a highly reliable behavioural biometric-based authentication technique. Originally designed for basic user verification, keystroke dynamics today serves a pivotal function across various sectors where its non-intrusive, continuous authentication feature provides significant benefits compared to conventional security methods (Subash A. , Song, Lee, & Lee, 2025; Badjoua, 2025; Badjoua, 2025). Traditional keystroke dynamics research used statistical models or classical machine learning such as SVM, Random Forest. But recently, deep learning methods have been increasingly used to capture temporal and non-linear behavioural patterns. The involvement of AI in keystroke dynamics authentication has been revolutionary by enabling systems to capture complex temporal patterns and adapt to free-text scenarios with unparalleled accuracy. The evolution from statistical methods to AI-driven approaches has yielded evident improvements (Badjoua, 2025). Building on these supervised methods, deep learning architectures have transformed keystroke dynamics by modelling intricate temporal and spatial dependencies, significantly improving performance albeit with increased computational demands. For example, Chen and Liu demonstrated that convolutional neural

networks (CNNs) achieved 96.2% accuracy on touchscreen-based keystroke data, reducing equal error rates by 22% compared to traditional approaches, particularly in free-text authentication scenarios (Chen & Liu, 2021). Similarly, LSTM networks have demonstrated 98.5% accuracy with a false positive rate of 0.8%, while hybrid CNN–SVM models achieve approximately 97.9% accuracy, albeit requiring more than twice the computational resources of standard SVMs (Patel & Kim, 2022). Despite these advances, supervised machine-learning methods tend to plateau at around 95.2% accuracy, whereas deep learning approaches reduce error rates by 40–60%, though often at the cost of architectures exceeding 10 million parameters (Lee & Kim, 2020). The integration of AI in keystroke dynamics has therefore been instrumental in advancing continuous authentication, allowing systems to adapt dynamically to user behaviour and achieve unprecedented accuracy.

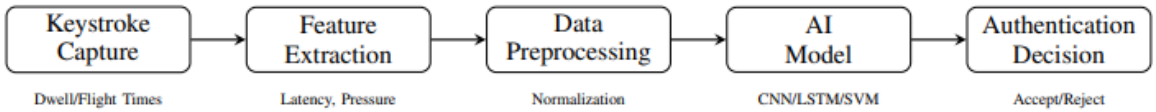

Figure 1 Pipeline of keystroke dynamics authentication.

The working theory of mouse dynamics biometric involves extracting the behavioural features from mouse movements and analysing them to derive a unique signature for each individual, enabling effective discrimination between users. Mouse dynamics has been proven to be a continuous, lightweight, and non-intrusive method for user authentication (Siddiqui, Dave, Vanamala, & Seliya, 2022). Advancements in mouse dynamics have expanded the capabilities of dynamic biometric authentication by moving beyond simple movement metrics toward more sophisticated behavioural interaction patterns. Traditionally, machine learning models have been trained to extract spatial characteristics from mouse dynamics data, and the capacity of CNNs to analyze intricate patterns has been utilized for user authentication. Yet, recent methodologies regard mouse dynamics as time-series signals, employing 1D-CNNs to convolve along the temporal axis of the data, hence enhancing authentication efficacy (Antal & Fejér, 2020). Combining mouse dynamics with a lightweight EfficientNet-based deep learning model has produced excellent authentication results. This approach demonstrates that such architectures can effectively capture discriminative behavioural features from mouse trajectories while remaining computationally efficient (Aljoubory & Mahdi, 2025; Sanjaya et al., 2025). The proposed EfficientNet-based model achieved a classification accuracy of 99.24% and an F1-score of 0.991, highlighting the potential of modern deep learning techniques to deliver highly accurate and reliable mouse dynamics authentication (Aljoubory & Mahdi, 2025).

Gait-based recognition is a contact-free biometric technology that identifies individuals based on their unique walking patterns which may vary due to variations in muscle strengths, bone length, and much more factors (Shu Shen, 2023; Kiyani et al., 2024). The two main approaches to obtain gait data includes vision-based which involves video and pose estimation; and sensor-based that contains accelerometers which are embedded in smartphones and also wearable IoT devices with the latter approach gaining more attention (Shu Shen, 2023). Recent studies have demonstrated that multi-modal fusion, combining gait

with other behavioural modalities can further enhance authentication performance and accuracy (Ayeswarya S., 2025). The multi-modal fusion approach has shown a significant increase in authentication performance by achieving 98.25% of combined accuracy and Equal Error Rate (EER) of 2.35%. The system offers smooth and non-intrusive authentication, maintaining strong security and supporting user convenience in varying scenarios (Ayeswarya S., 2025). Sensor-based gait recognition approaches can be broadly categorized into machine learning and deep learning methods. Traditional machine learning techniques, including SVM, KNN, Bayesian classifiers, and template matching, have been used to extract features such as acceleration and angular velocity from wearable sensors. These methods, however, often require gait cycle detection and labeled datasets, making them time-consuming and resource-intensive. To address these challenges, hybrid deep learning approaches have been developed. For example, CNN-LSTM architectures have been used to capture both spatial and temporal features from sensor data, achieving accuracies above 93%–95% (Lam Tran, 2021; Keon et al., 2025). Without the need for a manual feature design, the technique, which is based on 1D-CNN and a bidirectional LSTM network, can automatically extract features from the raw acceleration and gyroscope information. More advanced models combining residual blocks with attention mechanisms have further improved classification accuracy and efficiency, particularly when using multi-position sensor datasets (Ling-Feng Shi, 2023).

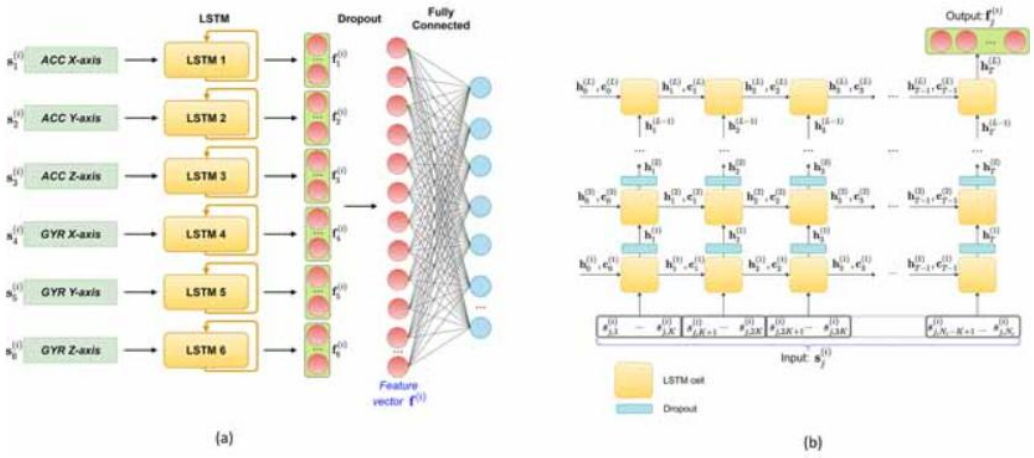

Figure 2 multi-model LSTM gait recognition network

3.2 Gap Analysis

Despite significant progress in behavioural biometrics, several limitations and research gaps persist. These gaps reduce the reliability, scalability, and practicality of current authentication systems. The following subsections outline the key issues identified in existing literature.

3.2.1 Limitations of Single-Modality Approaches

Unimodal biometric authentication systems which rely on only one single trait often suffer from noise, intra-class variation and also inter-class similarity (Alsaedi & Jaha, 2022; Weiqi et al., 2024). Most existing behavioural authentication systems rely on single-modality inputs, such as keystroke dynamics, mouse movement patterns, or gait signals to authenticate an user. While it is simpler to implement, unimodal systems often lack the robustness and distinctiveness necessary for a high level of security. On the other hand, multimodal fusion works by combining two or more behavioural traits in order to provide more accuracy and tend to be more reliable than unimodal dynamic biometric systems. Fusion strategies enhance the discriminative power of biometric systems and reduce false acceptance and rejection rates. Despite the evident advantages, multimodal biometric systems they're not widely implemented due to limitations such as cost, complexity, and infrastructure integration (Mehta, 2025; Ioannis Stylios, 2020).

3.2.2 Real-World Variability Challenges

Dynamic biometrics are said to suffer from reduced accuracy due to variation in samples and poor repeatability (Khan S. , Parkinson, Grant, Liu, & McGuire, 2020). Behavioural signals change with context and are inherently non stationary. Given an example in keystroke dynamics, a user's typing pattern can change significantly depending on their physical or emotional state. In a similar pattern, mouse dynamics are affected by factors such as device sensitivity, surface texture, and user posture. Gait patterns can vary with footwear, walking speed or injury despite coming from the same individual. Wearable sensors for certain data collection or detection use are prone to physical damage which relies on external factors such as incorrect extraction of data (Khan S. , Parkinson, Grant, Liu, & McGuire, 2020). Some behavioural biometric modalities generate data only when users perform specific actions such as typing for keystroke dynamics or moving a cursor for mouse dynamics. When users remain inactive for extended periods or switch to mobile or touch-based interfaces, these behavioural signals are absent, reducing the effectiveness and reliability of continuous authentication (Baig, Eskeland, & Yang, 2023).In motion-based authentication, continuous data collection is often inconsistent. Users may not always carry the sensing device, may remain stationary for extended periods, or may engage in activities that alter their normal movement. These irregularities can degrade the consistency and reliability of the resulting biometric signal. Prior studies from sources such as Google Patents and ScienceDirect further highlight that both inter-user variability and substantial intra-user variability like changes due to time of day, mood, health, or context challenge the assumption of stable and consistent behavioural signatures (Baig A. E., 2021). These variabilities cause elevated false rejection and false acceptance rates when models are trained in controlled settings are deployed in real world scenarios. A recent survey of keystroke and mouse dynamics reports that "current approaches cannot adapt to user behavior evolution and suffer from limited efficacy."

(Subash A. , Song, Lee, & Lee, 2025). In order to overcome such issues, models need to be adaptive and carefully designed.

3.2.3 Dataset and Evaluation Limitations

One major limitation in developing behavioural biometric systems is the need for vast amounts of training data to ensure accuracy and minimize false acceptance rate (FAR). Although recent research highlights the promise of hybrid deep learning models, their increasing complexity introduces practical implementation challenges, particularly in resource-constrained environments (Wang & Zhang, 2023). This challenge is intensified by the limited size of current available gait datasets has stunted the growth in developing more effective recognition systems (Hill, Pamplona Segundo, & Sarkar, 2023). While deep learning is able to reduce errors and fault rates by 40–60%, it demands over 10 million parameters, which is computationally expensive and extremely hard to deploy in real-world settings (Lee & Kim, 2020; Riskhan et al., 2025). Although their approach surpassed traditional methods based on handcrafted features, it remains limited in complex scenarios owing to insufficient temporal information (Hussain, Khan, Khan, Shabaz, & Baik, 2024). There is a lack of large datasets with multiple activities, such as typing, gait and swipe performed by the same person. Furthermore, large datasets with multiple activities performed on multiple devices by the same person are non-existent (Belman, et al., 2019; Brohi et al., 2020). The lack of accessible public behavioural biometrics datasets further restricts research progress as benchmarking remains a significant problem with the lack of data (Ioannis Stylios, 2020; Ying et al., 2024). Recent work in gait recognition and multi-modal data collection highlights efforts to address training-data shortages, but significant gaps remain in large-scale, longitudinal multimodal datasets and standardised CA benchmarks (Cole Hill, 2023; Chaudhary et al., 2022). Another significant concern in biometric data collection is the risk of demographic bias. When systems are trained mainly on data from one dominant group, such as white male subjects, they tend to perform less accurately for women and ethnic minorities. This imbalance can cause incorrect classifications, increased false positives, or unnecessary login rejections for users who are underrepresented in the training data (Gerner, 2020).

4. Propose Secure System Overview

4.1 System Overview

Based on the topic Secure Authentication using Behavioural Biometrics, behavioural biometrics is a form of authentication that examines the pattern of user's activity. There are types of behavioural biometrics that can be used to secure authentication such as digital gesture and mouse dynamic, typing pattern, smartphone use and habits and others.

Our proposed system is a secure system that authenticates users according to their mouse dynamic. We name it Hybrid Behaviour Authentication Framework (HBAF). The main objective of the system is to develop an innovative method for login authentication to ensure the system is secure and reduce the risk of data theft from keyloggers or any form of passive attacks.

The major feature of our system is that instead of only requiring the user to login with their username and password, we also include mouse clicks to differentiate if a user is legitimate or impersonation. To give an overview of what our system can do to secure authentication is that our system will allow user to click anywhere in the screen three times, and it will record and save it to the user profile, so when user want to access their profile, they need to click the three same place and enter their credential to login to their account.

System Architecture

The HBAF system is fully developed in Python and is built using 5 main components: GUI Component, Behavioural Registration Component, Behavioural Authentication Component, Login Component and Data Storage Component. Each component interacts with each other to perform a complete login system. The interactions between these modules form the entire HBAF architecture, where user inputs such as mouse clicks and credentials input are collected and sent to the behavioural components for processing, authentication, and storage. Figure 14 shows the system architecture diagram of the proposed HBAF system.

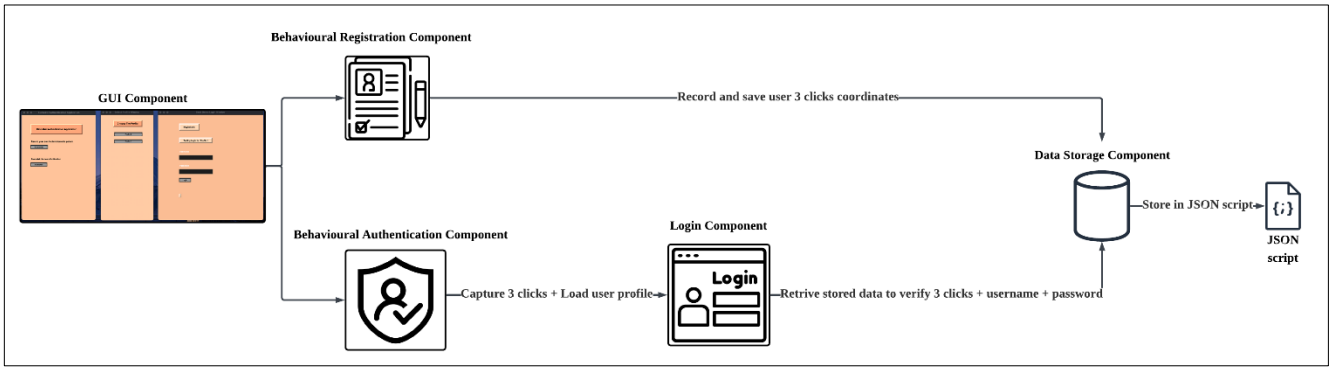

Figure 3: Proposed HBAF system

Security Advantages

The HBAF system provides a strong security advantage because of the behavioural biometric layer that is hidden to attackers and extremely difficult to replicate. By requiring the users to perform the 3 unique mouse-click patterns before entering their credentials, HBAF is able to protect against common attack techniques such as keylogging, credentials theft, brute-force attack, shoulder surfing, and replay attacks. Because the click pattern is invisible and not prompted on the interface, attackers will not even know that an additional authentication step exists. This makes HBAF significantly more resistant to impersonation attempts compared to the traditional password only systems for authentication, while still being privacy-friendly, easy to use, and does require any extra hardware.

GUI Component

GUI Component is the graphical interface of the system which the user will interact with. The GUI Component is also used to beautify the interface to make it into a user-friendly visual interface that allows users to perform different actions instead of running the system in the console. The GUI is developed using the Tkinter module which is a Python's built-in library for creating GUI interfaces and allows programmers to develop a simple and intuitive desktop application (Geeksforgeeks, 2025). We use Tkinter because it provides various widgets that we need like buttons, text boxes, labels, and images. Figure 2 shows the widgets we used and how we designed them.

```
window = Tk()  # instantiate an instance of a window for us
window.geometry("420x420") #width x height
window.title("Biometric Authentication Application")

icon = PhotoImage(file=r'C:\Daphne\Taylors\Degree Daphne\Year 2 Sem 1\Computer Intrusion Detection\Group Assignment\CID Project\CID Project\email-icon-121.png')
window.iconphoto(True, icon)

window.config(background="#A7C7E7")

# Label Section
top_title = Label(window,
                  text="Biometric Authentication Application",
                  font=('Arial',10,'bold'),
                  fg='black', # fg =foreground to change text color
                  bg="#A7C7E7",
                  relief=RAISED,
                  bd=5,
                  padx=15,
                  pady=15
                  )
# top_title.pack() # To instantly put it in but it be at the top middle
top_title.place(x=80,y=50) # To "place" the title at certain position
```

Figure 4: GUI Components Widgets

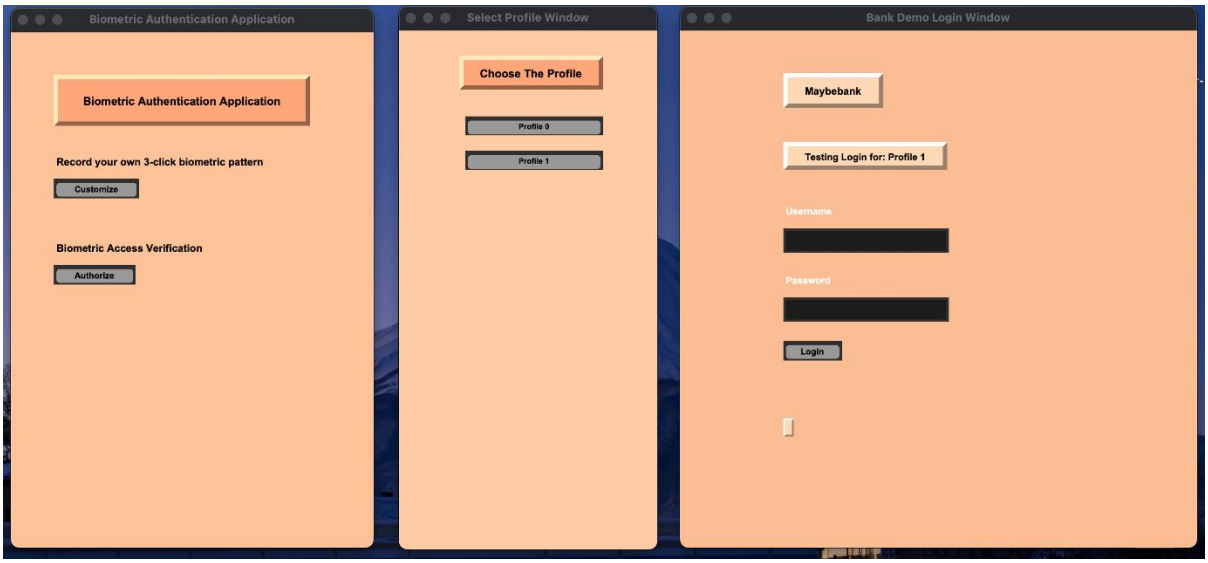

Figure 3: GUI Interfaces

Biometric Registration Components

Biometric Registration Components is a component that allows users to register and save their coordinates into the JSON file and assign a profile to them. This component uses Pynput which is a python library that allows us to control and monitor input devices. In developing the system, we specifically set the input devices to mouse, then we add listeners to capture the mouse click coordinate. In the registration components, we use sequential data recording algorithms. This algorithm listens for mouse clicks and records clicks in the order they occur, then save each click coordinate x and y into JSON file and stops when user clicks up to 3 times.

```python
# On click, it will record down the x & y position the mouse is at when the click happen
# This will be the function that saves the coordinates for password
def on_click(x, y, button, pressed):
    global click_counter # To access the click_counter that is outside of this function
    if click_counter < 5 and pressed == True and button == mouse.Button.left:
        print(f"Left click at ({x}, {y})")
        save_coordinates(x,y)
        click_counter += 1
    else:
        if click_counter >= 3:
            print ("Max Clicks")
            # Save the json file just in case, when its done
            with open(main_file_path, "w") as file:
                json.dump(data, file, indent=4)
            return False # Close the entire thing

# Saves the x & y coordinates of each click into the CHBA text file
def save_coordinates(x, y):
    data["Row " + str(row_number)].append({"x":x, "y":y})
    print("Saved: x=" + str(x) + ", y=" + str(y))
    # Saves after each click, just in case.
    with open(main_file_path, "w") as file:
        json.dump(data, file, indent=4)

# Program instantly starts listening for mouse clicks
with mouse.Listener(on_click=on_click) as listener:
    listener.join()
```

Figure 6: Sequential Data Recording Algorithm

Behavioural Biometric Authentication Component

A Behavioural Biometric Authentication Component is to authenticate which user is legitimate by comparing the coordinates clicked during logging with the stored coordinates. If it matches, then access is granted, if not access is denied. The Authentication Component uses the same python library as the Registration Component which is Pynput because registration is also required to capture mouse clicks. In the authentication component, we use the coordinate-matching algorithm where we first load the save coordinates for selected user profiles, then we capture new coordinates clicked from the user, compare saved coordinates and new coordinates, ensuring both coordinates match to grant access to the user.

```python
def compare_coordinates():
    global access_successful
    pixel_tolerance = 15 # The amount of difference in pixels allowed (Basically in case the user clicks abit off from the save point)
    success = True

    # Checks if both x and y are close enough, if yes = access granted, if no = access denied
    for i in range(len(get_coordinates)):
        coordinateX = get_coordinates[i]["x"]
        coordinateY = get_coordinates[i]["y"]
        authenticateX = authenticate_clicks[i]["x"]
        authenticateY = authenticate_clicks[i]["y"]

        # This is to calculate how far apart the points of the saved coordinates and the clicked coordinates are.
        difference_x = abs(coordinateX - authenticateX)
        difference_y = abs(coordinateY - authenticateY)

        if difference_x <= pixel_tolerance and difference_y <= pixel_tolerance:
            print("Comparison Matches")
        else:
            print("Mismatch at point"+ str(i+1))
            success = False

    if success == True:
        access_successful = True
    elif success == False:
        access_successful = False
```

Figure 7: Coordinates-Compare Algorithm

Login Component

Login Component is a login page that allows users to enter their credentials and login into their account. If the coordinates are correct and the user enters the correct credentials to their account, it will prompt that your login is successful.

```python
def login_action():
    username = login_entry.get()
    password = password_entry.get()

    global auth_result

    # Biometric Authentication
    if auth_result == "Access_Granted":
        if username == "admin123" and password == "abc123":
            result_label.config(text="Login successful, Welcome User")
        else:
            result_label.config(text="Login is unsuccessful, please check the username and password.")
    else:
        result_label.config(text="Login IS unsuccessful, please check the username and password.")
```

Figure 8: Login Algorithm

Data Storage Component

Data Storage Component is responsible to save user click coordinates and retrieve it during the authentication phase. In our system, we utilized the JSON (JavaScript Object Notation) module which is also a built-in module in Python. It is used to store and transfer data between the browser and the server. HBAF uses the JSON module because it has been integrated into Python and it is simple to use.

JSON modules help to transfer clicked coordinates to the system then store those coordinates into a specific user profile for the ease of retrieving in the future. Each user profile is represented as a JSON

object containing a list of x and y coordinates that is captured during registration/customization. When a user login to their profile, the system will read the JSON file to retrieve stored coordinates for authentication.

JSON files were used because they are readable by humans, easily updated, stored, viewed and managed. Also, it doesn't require a full database to store the coordinates, so it is suitable for our system as we were just developing a prototype.

According to figure 7, we show the code of the system where we utilized JSON. Based on the code, we show how the system stores data in a JSON file where the file acts as the coordinates/biometric database. Then we load the JSON file and if it's empty we print an error code. For figure 8, we show the format of storing user coordinates.

```
# Change the file path depending on your device. NOTE: USE ".json" or it wont work
main_file_path = "C:\Daphne\Taylors\Degree Daphne\Year 2 Sem 1\Computer Intrusion Detection\Group Assignment\CID Project\CID Project\CID_Prac1.json"

if os.path.exists(main_file_path):
    try:
        # Loads the file/database
        with open(main_file_path,"r") as file:
            data = json.load(file)
    except json.JSONDecodeError:
        print("File is empty or not valid JSON")
        data = {}
else:
    data={}
```

Figure 9: Usage of JSON in System

```
{
    "Row 0": [
        {
            "x": 1875,
            "y": 456
        },
        {
            "x": 1875,
            "y": 456
        },
        {
            "x": 1875,
            "y": 456
        }
    ],
```

Figure 10: Example of JSON file Storing Format

Flow Chart

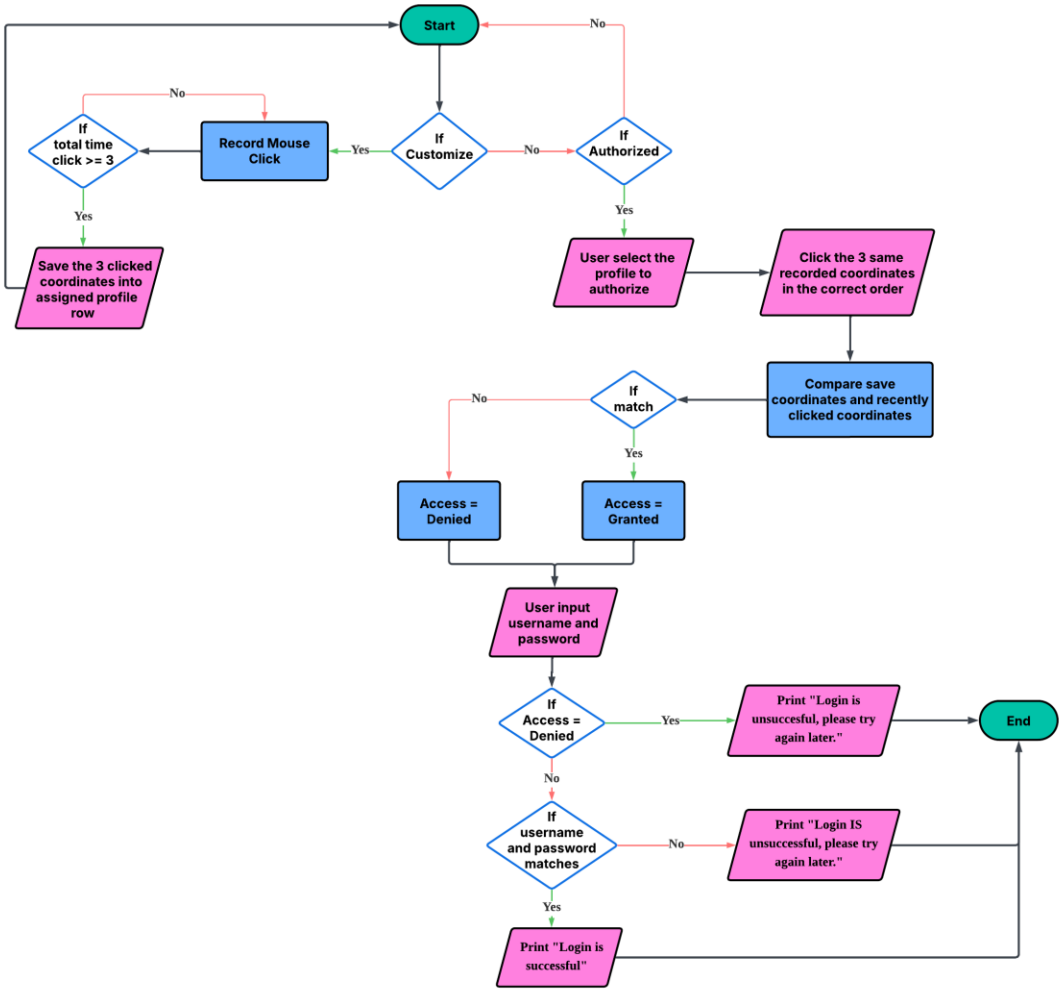

Figure 11: Flow chart

4.2 Compare Other System with Our system

We chose BioCatch to compare it to our HBAF system. The main reason for choosing BioCatch for the comparison is because it has some similar features as HBAF does. BioCatch is a behavioural biometrics security platform that is used in the financial industry. It analyses thousands of behavioural parameters such as mouse movements, typing speed, scrolling rhythm, the way of holding devices and other micro movements into a unique behavioural profile for users. BioCatch doesn't just rely on password and username, instead, they authenticate by user's behaviour. Even after logging into the system, it keeps monitoring to ensure the correct person is still operating the system or device.

In terms of privacy, the HBAF system is superior because BioCatch collects up to thousands of behavioural parameters continuously, which requires users to grant multiple permissions to allow

BioCatch to collect all the parameters data, it leads to higher privacy concerns for users who value their personal behavioural data. In contrast, HBAF solved these concerns by using only three mouse-click coordinates for authentication which resulted in lower privacy concern, and it is much more privacy friendly.

Another reason why HBAF is better is because when using HBAF to login to an account, the system does not display any instructions such as “Start clicking three coordinates”. Instead, the process was hidden and only the legitimate user who has customized their coordinates knows when to start clicking. This reduces the chance of attackers noticing that there is a hidden step before entering their password and username. However, BioCatch creates a unique behavioural profile for each user, but if an attacker uses a spyware attack, they can secretly monitor and record the user's behaviour. The stolen data can be used to impersonate the user and login to his/her account successfully.

Lastly, HBAF is superior in terms of login speed and user profile creation. When a user wanted to login using HBAF, they were only required to click three points and proceed to login using username and password. The entire login process does not take more than 1 minute making it very convenient for everyday use. Additionally, HBAF can create a user profile instantly after users customize their three coordinates. On the other hand, BioCatch needs to run multiple real-time behaviour analysis checks during login, which slow down the process. Also, if a user has some behaviour changes, the system may flag it as suspicious and log the user out. Furthermore, creating a user profile in BioCatch requires monitoring the user for some time before a customized user profile can be created.

5. Experimental Setup and Comparative Analysis

5.1 Comparing the approach with existing security solutions

When comparing our approach to behavioral biometric authentication and biometric authentication such as fingerprint authentication, our approach is much harder to replicate as it relies on mouse pointer movement as well as click patterns. As for fingerprint authentication, it is static which means that it relies on a single, permanent and reusable credential, being the fingerprint. Because of this, it is difficult to change the authentication credential and makes it vulnerable to fake fingerprints. Thus, our approach is more resilient to spoofing as it is more difficult for the attacker to mimic behavior of the user that can change over time, instead of just bypassing a single scan. (Plurilock, n.d.)

Another aspect of comparison is the user capabilities, in terms of hardware and physically. When using behavioral biometric authentication, no extra hardware is needed as authentication is integrated into normal mouse interactions. On the other hand, fingerprint authentication requires a fingerprint scanner which is not included in a lot of devices. Not only that, but if the finger were to be in bad condition such as worn-down fingerprint due to manual labor, the authentication process can fail due to not being able to scan the fingerprint correctly. This proves that integration of behavioral biometric authentication is much more seamless when compared to integrating fingerprint authentication. (Scott, What is Behavioral Biometrics and How Is It Used?, 2024)

Moving on, One-Time-Passcode (OTP) is also another type of security solution that can be compared with biometric behavioral authentication. Firstly, when using OTP, the passcode can be intercepted via phishing or through malware while it is difficult to replicate user movement and mouse clicks when using behavioral authentication. This makes behavioral authentication much more secure compared to OTP as attackers must go through the difficulty of either acquiring the pattern of mouse clicks or brute-forcing the patterns when behavioral biometric authentication is used instead of just obtaining the code when OTP is used. (CSA, 2023)

Next, behavioral biometric authentication is less sensitive to data breaches compared to OTP. This is because behavioral data is non-physical abstract and non-identifying which makes it extremely difficult for the attacker to reconstruct the user's identity from the data alone even if they can access the raw mouse-click patterns. As for OTP, linked accounts to the OTP may reveal identity when the delivery channels are intercepted, allowing the attacker to take over the account to obtain personal information. (GEETEST, 2025)

Benchmarking experiment for comparative analysis

This experiment aims to benchmark the behavioral biometric mouse-click–based authentication method against traditional password authentication. The goal is to evaluate the performance, security, and behavior of the mouse-click authentication method. Both systems were implemented in Python and use the same storage format (JSON), making them directly comparable.

The focuses of this benchmarking include:

1. System metrics
2. System performance metrics
3. Error and robustness metrics
4. Scalability and storage considerations

Technical Metrics used in Experiment

The table 1.1 shows the technical metrics that were selected along with their respective description based on their categories.

Category	Technical Metric	Description
Accuracy	True Acceptance Rate (TAR)	% of genuine logins accepted
	False Rejection Rate (FRR)	% of genuine logins incorrectly rejected
	False Acceptance Rate (FAR)	% of logins correctly rejected
Security	Credential Space Size	Highest possible number of existing secrets
	Brute-force Capability	Number of attempts to guess the credential
	Collision Probability	Likelihood of 2 users having the same credentials
Performance	Latency	Process time for one login attempt
	Throughput	Highest possible number of login attempts

Table 1: Selected technical metrics.

Benchmarking Overview

This benchmarking experiment was structured to evaluate and compare the mouse-click authentication against the traditional password authentication, using a controlled and repeatable dataset of simulated login attempts. Each authentication system was tested with an equal number of genuine and imposter attempts to ensure accurate evaluation of the accuracy, security, and performance metrics. A total of 50 authentication attempts were executed for each system, consisting of 25 genuine attempts and another 25 imposter attempts.

All attempts were done using predefined inputs to ensure that no human error can influence the results of this experiment. With this, the experiment delivers a fair and unbiased comparison between the mouse-click authentication method and the traditional password authentication method.

Genuine attempts

The genuine attempts represent authentication attempts executed by using the correct credentials for each authentication method. For the mouse-click method, this involved providing the 3 correct mouse-click coordinates stored in the JSON file. As for the traditional password authentication method, genuine attempts used the exact same saved password.

These genuine attempts were important for evaluating the True Acceptance Rate (TAR) and the False Rejection Rate (FRR). These metrics measure how reliable each system is at accepting legitimate users.

Imposter attempts

The imposter attempts measure how effectively both the systems reject incorrect or unauthorized login attempts. For the mouse-click authentication method, imposter attempts were done by deliberately clicking on the wrong coordinates that did not match the stored values. As for the traditional password authentication method, predefined incorrect passwords were used.

These imposter attempts were used for the calculation of the False Acceptance Rate (FAR). This metric indicates how resistant these models are towards unauthorized access.

5.2 Results and Comparative Analysis

This section shows the results of the benchmarking experiments using the previously defined technical metrics, such as accuracy, security, performance, and scalability.

Accuracy and Error Rates

In table 2, it shows accuracy rates of the mouse-click authentication and the password authentication.

Metric	Mouse-click	Password
True Acceptance Rate (TAR) (%)	88	100
False Rejection Rate (FRR) (%)	12	0
False Acceptance Rate (FAR) (%)	4	0
Overall Accuracy (%)	92	100

Table 2: shows the Accuracy Metrics of Mouse-click Authentication and Password Authentication

Although the overall accuracy of the password authentication method is perfect due to its deterministic string matching, the mouse-click authentication approach still has a strong overall accuracy of 92% while providing various security benefits that password authentication method does not. The given False Rejection Rate is given due to small behavioral variations in simulated coordinate-based input, which can be greatly reduced through an improved tolerance radius.

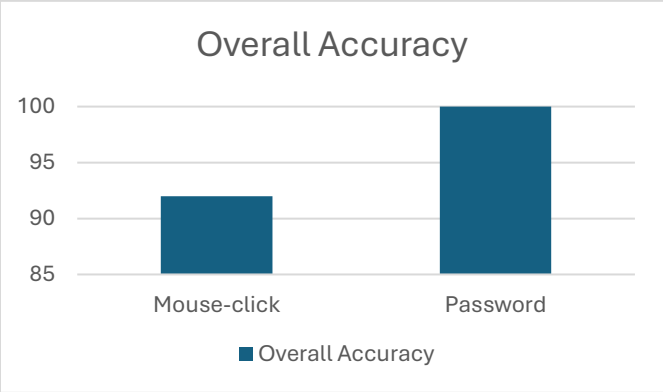

Figure 12: Shows the Overall Accuracy of Mouse-click Authentication and Password Authentication

Security Metrics

Three different metrics have been used to benchmark and compare the security metrics of the mouse-click authentication method and the password authentication method.

Credential space size

Mouse-click authentication: ~8.9×10^{18} combinations
Password authentication: ~2.8×10^{12} combinations

The credential space size when using the mouse-click authentication method is much larger than that of the password authentication method. This makes the mouse-click method much more resistant to brute-force attacks.

Brute-force resistance

Mouse-click authentication: ~1.4×10^{8} years
Password authentication: ~45 years

Both methods have a guessing rate of 1000 guesses a second. The difference between the two authentication methods is very drastic as the mouse-click method proves to be much more brute-force resistant, making guessing attacks impractical.

Collision Probability

Mouse-click authentication: Near zero, even with millions of users
Password authentication: Moderate risk without enforcing complexity rules

Mouse-click authentication method having a near zero collision probability makes it far more scalable and unique, especially in systems that have large user populations.

Latency

In table 3, it shows the average latency, in milliseconds (ms), of the mouse-click authentication and the password authentication.

Authentication Method	Average Latency (ms)
Mouse-click	320ms
Password	180ms

Table 3: Average latency for both authentication method in milliseconds

The mouse-click authentication method has a lot more latency when compared to the password authentication method due to the extra processing of the sequence of the 2 spatial inputs rather than just a single string. This extra processing contributes greatly to the strength of security and makes automatic attacks extremely difficult.

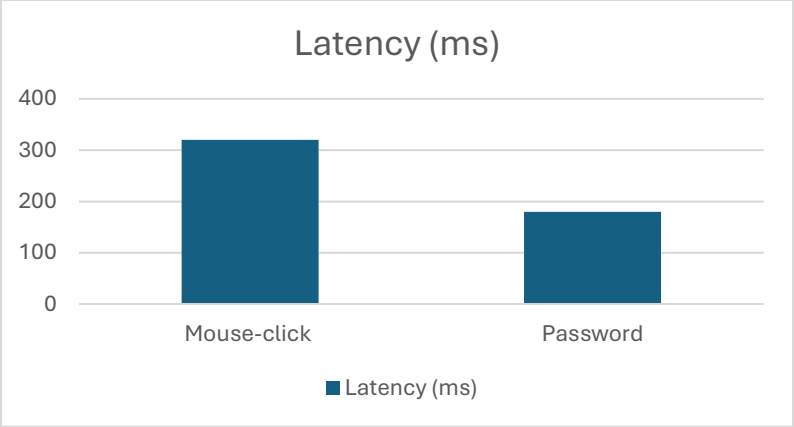

Figure 13: Shows the Total Latency in ms for Mouse-click Authentication and Password Authentication

Throughput

The table 4 shows the highest number of logins attempts that can be done per second of the mouse-click method and the password method.

Authentication Method	Attempts per second
Mouse-click	3.1
Password	5.4

Table 4: Highest number of logins attempt per second.

While the password authentication method is able to process more attempts per second, this also means that the password authentication method is more vulnerable to automated brute-force attacks, especially when offline.

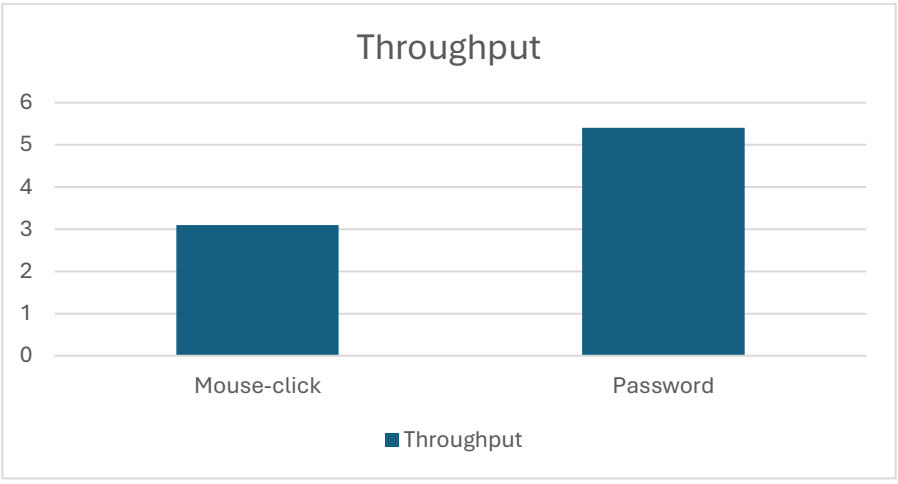

Figure 14: shows the Throughput in ms for Mouse-click Authentication and Password Authentication

6. Implementation Challenges & Feasibility

Although HBAF can mitigate issues that traditional login methods face—such as keylogging, brute-force attacks, and password theft through shoulder surfing—it still presents its own set of challenges when it comes to implementation and evaluating how it will hold up in a real-world environment (Traitware, 2020).

6.1. Deployment Issues

6.1.1. Device Dependency

One trait of the system that may hinder its overall success is the strong dependency on a graphical user interface (GUI), since the system is based on click-pattern behavioural authentication. A GUI-based system can behave differently across operating systems, which makes cross-platform compatibility a major concern during implementation.

Simple factors such as screen resolution and display scaling could cause discrepancies even for the authorised users when attempting to access their account from a new device (Scott, What Is Behavioral Biometrics and How Is It Used?, 2024).

6.1.2. Data Storage and Security

In the system prototype provided, the 3-click coordinates are stored inside a JSON file. Prototype-wise, that method is suitable due to its readability and ease of use; However, the same could not be applied when it comes to the actual production of the system. Storing credentials locally makes the system prone to sensitive data exposures—such as confidentiality and integrity breaches (Reiter, 2024).

In terms of real-world deployment, an encrypted database with proper access control and backup mechanism will be more suitable as the system would need a properly secured and centralized storage space. This is easier said than done, as integrating a secure infrastructure, handling existing accesses, and ensuring compliance with data protection laws are all difficult processes to manage.

6.2. Scalability

As the system grows, there will be an increase of the variation that needs to be considered across different types of users in different kinds of environments, such as the different OS settings, screen resolution, monitor variations, and different click behaviours. Subsequently, getting the system to be consistently accurate proves challenging as it is harder to tune threshold values for a large number of users. With more users adopting the system, the issue of device dependency could worsen, leading to an increase of false acceptance rate (FAR) and false rejection rate (FRR) (Shuwandy, et al., 2025).

However, this is not to say that the system is inherently unscalable, but rather, a few measures must be taken to ensure accuracy and reliability for larger user bases. Measures such as fuzzy coordinate matching, device calibration, and maintaining per-device profiles, all allow the system to tolerate natural variations while keeping FAR and FRR low.

6.3. Cost

HBAF uses open-source software such as Python, Tkinter, and Pynput, with the only hardware requirement being devices that users already own, such as their desktop and laptop computers, along with a mouse. While that can lower the price of development, it is important to note that there are still other expenses to be made. For instance, server hosting, maintenance, auditing, and user training (Shuwandy, et al., 2025).

In terms of cost structure, expenses for development and training would be a one-time upfront payment, while infrastructure, maintenance and audits are the recurring costs. With the system being software-based, the recurring costs are relatively moderate, though they could increase with a larger user base. That said, HBAF is financially feasible, greatly due to the fact that it leverages existing user devices.

6.4. Ethical Concerns

As mentioned before, the system utilizes click-pattern behavioural authentication, which means it relies heavily on the position of the cursor on screen during the three mouse clicks, to authenticate the user. There would only be a small window where the system would be able to track and monitor mouse events—which is before the login process. The system will not track or record any other activity outside of that.

That said, concerns may be raised by users regarding the risk of continuous monitoring of their online activity by the system and the misuse of data without consent (Oloid Desk, 2025). Hence why there should be a strong emphasis on transparency, clearly stating what is being collected and why, in order to address ethical concerns and raise trust among users.

6.5. User Adoption

HBAF is relatively easy and quick to use, with the only change in the normal login routine being the three specific click points before entering the correct credentials. Due to this simplicity factor, it is more likely to encourage adoption among regular users. Although, some people might be unfamiliar with the concept of behavioural biometrics, so it is very important that clear communication and instructions are given prior to the set-up, to prevent users from getting locked out of their own accounts.

Another thing that might hinder user adoption is how the system carries out error handling. False negatives are bound to happen in behavioural biometrics. Therefore, it is very important to implement mitigation strategies—such as fallback authentication and retry mechanisms—to address these issues, as false negatives can greatly frustrate users and reduce trust in the system.

7. Evaluation and Discussion

When designing any system, it is crucial to critically analyse and evaluate the system in terms of its strengths and limitations, while also assessing how it compares to current industry standards. Such evaluation ensures that the system carries out its intended goals, highlights any potential improvements needed, and guarantees that it remains aligned with the expectations of the existing industry.

7.1. Strengths

7.1.1. Enhanced Authentication System

The use of mouse dynamics in addition to the traditional username and password authentication—complements each other quite well and can greatly enhance the authentication security (Terranova Security, 2023). By requiring attackers to not only enter the right credentials but also imitate the 3 specific clicks set by the user, it adds an extra layer of protection that could mitigate vulnerabilities such as keylogging and shoulder surfing.

In addition to that, each click can occur at any coordinate within the screen area; therefore, the total number of possible sequences grows exponentially with the device's screen resolution. Consequently, it takes longer and more effort for the attacker to bypass authentication.

7.1.2. Stability and Sustainability

Another advantage comes from the fact that behavioural biometrics like click-based mouse dynamic, is not tied to a user's physical condition. Physical biometric identifiers, notably fingerprints and facial recognition, are susceptible to environmental factors, including lighting conditions or unexpected alteration due to injuries (Plurilock, 2024). Compared to that, mouse dynamics is considerably more stable for daily and long-term usage because of its ability to remain consistent.

The motion of clicking is a straightforward and low-effort motor behaviour that is performed repeatedly throughout daily computer interaction. Which means the physical execution will likely remain stable even across different circumstances that may affect the user's motor skills, such as major fatigue. The stability of the system ensures limited variances between login attempts, reducing false rejections while being sustainable.

7.2. Weaknesses

7.2.1. Vulnerable to Observation Attack

The system's security relies heavily on the secrecy of the click-coordinate sequence and the attacker's lack of knowledge regarding the existence—making it a discreet authentication method. However, maintaining that discretion is also a challenge in itself. There is always a risk that an attacker may eventually discover the sequence or the existence of the authentication mechanism.

In such a scenario, the system loses its biggest advantage, lowering its degree of security and reducing it to just another form of traditional password-based login. Which is why it is so important to have complementary security measures in addition to the system.

7.2.2. Vulnerability to Pattern-Guessing Attack

Although the random 3-click combination on an average desktop display of 1920×1080 would be extremely secure in theory, the system will still be vulnerable to a pattern-guessing attack. Mathematically speaking, it would be very time-consuming and impractical to launch a brute force attack on the system; however, the underlying issue here is human predictability.

Users rarely produce genuine random click points for fear of forgetting the sequence later and subsequently having to either deal with getting locked out of their account or sit through the whole password reset process. Hence, users tend to choose predictable areas even on larger screens, such as the corners, centre regions, and near UI elements. This weakness highlights the importance of user training and guidance.

7.3. Comparison Against Industry Standards

7.3.1. NIST SP 800-63B

The National Institute of Standards and Technology (NIST), USA has published guidelines and technical requirements for federal agencies that are implementing digital identity services. The standard focused on here will be the NIST SP 800-63B – Digital Identity Guidelines: Authentication and Lifecycle Management (NIST, n.d.).

7.3.2. Comparison and Alignment

The NIST SP 800-63B highlights the best practices for secure digital authentication, putting a big emphasis on the use of strong passwords, multi-factor authentication (MFA) and limited failed login attempts (NIST, n.d.). HBAF aligns well with these guidelines because it is an MFA system, since it combines traditional password logins with behavioural biometrics—the three specific mouse click coordinates.

Security-wise, the NIST highlights the importance of having secure storage of authenticators and preventing the occurrence of compromised credentials (NIST, n.d.). As of now, the HBAF prototype stores the click-coordinate data in a JSON file; however, in the actual production, the click-coordinates will likely be stored inside an encrypted relational database, to increase security and fully comply with the standard.

HBAF also aligns with NIST in terms of usability, in which the guideline states the need for balancing security with convenience (NIST, n.d.). HBAF does not intrude too much on the login process. It is fast and user-friendly, requiring no more than one minute to place the three click coordinates. Not to mention that HBAF does not require any additional hardware, making it easier and more convenient to adopt.

Overall, HBAF does largely align with the NIST's guidelines. While it exceeds more in certain aspects—such as the enhanced behavioural protection area—future adjustments could be made in terms of secure storage, rate-limiting, and user training in order to ensure complete compliance.

8. Conclusion & Future Work

A prototype that supports multifactor authentication by integrating mouse dynamic biometric method and password authentication without requiring additional hardware was successfully developed. The result of comparative analysis shows that the mouse dynamic biometric method provided high accuracy (92%), greater security and adequate performance compared to traditional password authentication. Furthermore, it provides a more secure and stable behavioural biometrics authentication system that can reduce impersonation risk and aligns well with the National Institute of Standards and Technology (NIST). Behavioural biometrics are promising for continuous and privacy-friendly authentication, making them an effective component in the multifactor authentication systems.

Despite the advantages, the prototype still faces several challenges in real world deployment, including device dependency, data storage security, scalability and other practical issues. These issues need to be handled to improve the system and reduce false acceptance and rejection rates. Since the developed prototype was solely covering the basic critical part to show a secure authentication using mouse dynamic biometrics and password authentication, there are several future works that can be done to improve its overall performance, efficiency, usability and security.

The prototype is currently using the JSON file for data storage. However, the JSON file does not provide secure data security and integrity, its capacity may also be insufficient as the number of registered users increases. Therefore, the JSON file can be converted to the SQL database, which provides more mechanisms to ensure secure access and data storage. It supports efficient query features, helping the system to reduce performance latency (Rai, 2024).

Moreover, the future work can also include adding additional features to enhance the usability and security of the prototype. For example time limit starting from the first click, detection of click pressure, key hold time and typing rhythm. These additional functions can be supported by AI and machine learning, which can be trained to detect suspicious attempts efficiently. Additionally, AI can provide a smarter fault tolerance for mouse click position to maintain the accuracy while improving user satisfaction at the same time (Kapron, 2025).

Finally, since the prototype is currently a simple python program, extending platform compatibility is also an important part of future work. The prototype should be deployed with web integration and expanded to a mobile version that allows for touchscreen gestures instead of mouse click to improve system's flexibility and accessibility. The mobile gait analysis can also be included as an additional measurement of authentication to enhance the mobile version function (Choi, Choi, & Kang, 2023).

9. References

[1] Aljoubory, S. H., & Mahdi, M. S. (2025). User Authentication Based on Mouse Dynamics Using an Efficient-Net Model. Iraqi Journal for Computers and Informatics (IJCI), 224–242. doi:10.25195/ijci.v51i2.612

[2] Alsaedi, N. H., & Jaha, E. S. (2022). Dynamic audio-visual biometric fusion for person recognition. Computers, Materials & Continua, 71(1), 1283–1311. doi:10.32604/cmc.2022.021608

[3] Antal, M., & Fejér, N. (2020). Mouse dynamics based user recognition using deep learning. Acta Universitatis Sapientiae, Informatica, 39-50. doi:10.2478/ausi-2020-0003

[4] Ayeswarya S., J. S. (2025, August 20). Enhancing security and usability with context aware multi-biometric fusion for continuous user authentication. Scientific Reports. doi:https://doi.org/10.1038/s41598-025-14833-z

[5] Badjoua, B. D. (2025, July). Keystroke Dynamics Authentication: A Comprehensive AI-Driven Review. International Conference onCyberSecurity and Information Engineering (ICCSIE).

[6] Baig, A. E. (2021). Security, Privacy, and Usability in Continuous Authentication: A Survey. Sensors, 21(17). doi:10.3390/s21175967

[7] Baig, A. F., Eskeland, S., & Yang, B. (2023). Privacy-preserving continuous authentication using behavioral biometrics. International Journal of Information Security, 1833–1847. doi:10.1007/s10207-023-00721-y

[8] Barney, N. (2021, September). What is Authentication? Retrieved from techtarget.com: www.techtarget.com/searchsecurity/definition/authentication

[9] Belman, A. K., Wang, L., Iyengar, S. S., Sniatala, P., Wright, R., Dora, R., . . . Phoha, V. V. (2019). Insights from BB-MAS – A Large Dataset for Typing, Gait and Swipes of the Same Person on Desktop, Tablet and Phone. arXiv. Retrieved from https://arxiv.org/abs/1912.02736

[10] Bhansali, A. (2024, December 26). Understanding The Security Implications Of Fingerprint Authentication. Retrieved from OmniDefend: https://www.omnidefend.com/understanding-the-security-implications-of-fingerprint-authentication/

[11] Bilton, R. (2025, July 21). Weak passwords allowed hackers to sink a 158-year-old company. Retrieved from bbc.co.uk: https://www.bbc.com/news/articles/cx2gx28815wo

[12] Brohi, S. N., Jhanjhi, N., Brohi, N. N., & Brohi, M. N. (2020). Key Applications of State-of-the-Art Technologies to Mitigate and Eliminate COVID-19.pdf. TechRxiv. https://doi.org/10.36227/techrxiv.12115596.v1

[13] Chaudhary, M., Gaur, L., Jhanjhi, N., Masud, M., & Aljahdali, S. (2022). Envisaging employee churn using MCDM and machine learning. Intelligent Automation & Soft Computing, 33(2), 1009–1024. https://doi.org/10.32604/iasc.2022.023417

[14] Chen, Y., & Liu, Z. (2021). Deep convolutional neural networks for keystroke dynamics authentication. IEEE Access, 45678–45689.

[15] Choi, J., Choi, S., & Kang, T. (2023, July 14). Smartphone Authentication System Using Personal Gaits and a Deep Learning Model. Sensors, 12. Retrieved from MDPI: https://www.mdpi.com/1424-8220/23/14/6395

[16] Cole Hill, M. P. (2023). Reducing Training Demands for 3D Gait Recognition with Deep Koopman Operator Constraints. Retrieved from https://arxiv.org/abs/2308.07468

[17] CSA. (2023, March 23). Importance of Using Secure Multi-Factor Authentication Methods. Retrieved from CSA: https://www.csa.gov.sg/alerts-and-advisories/advisories/ad-2023-006/?utm_source=chatgpt.com

[18] Geeksforgeeks. (2025, 8 4). Python Tkinter. Retrieved from GeeksforGeeks: https://www.geeksforgeeks.org/python/python-gui-tkinter/

[19] GEETEST. (2025, September 1). What is SMS OTP, and how to ensure SMS OTP security in 2025. Retrieved from GEETEST: https://www.geetest.com/en/article/sms-otp-security-2025?utm_source=chatgpt.com

[20] Gerner, M. (2020, May). Weighing the pros and cons of behavioural biometrics. Retrieved from Raconteur: https://www.raconteur.net/technology/behavioural-biometrics-pros-cons

[21] Ghosh, S., Singh, A., Kavita, N., Jhanjhi, N. Z., Masud, M., & Aljahdali, S. (2022). SVM and KNN Based CNN Architectures for Plant Classification. Computers, Materials & Continua/Computers, Materials & Continua (Print), 71(3), 4257–4274. https://doi.org/10.32604/cmc.2022.023414

[22] Hill, C., Pamplona Segundo, M., & Sarkar, S. (2023). Reducing Training Demands for 3D Gait Recognition with Deep Koopman Operator Constraints. arXiv. doi:10.48550/arXiv.2308.07468

[23] Holdsworth & Kosinski. (2025, June 11). What is behavioral biometrics? Retrieved from ibm.com: https://www.ibm.com/think/topics/behavioral-biometrics

[24] Holdsworth, J., & Kosinski, M. (2025). What is Behavioral Biometrics? Retrieved from IBM: https://www.ibm.com/think/topics/behavioral-biometrics?

[25] Humayun, M., Hamid, B., Jhanjhi, N., Suseendran, G., & Talib, M. N. (2021). 5G Network Security Issues, Challenges, Opportunities and Future Directions: a survey. Journal of Physics Conference Series, 1979(1), 012037. https://doi.org/10.1088/1742-6596/1979/1/012037

[26] Humayun, M., Jhanjhi, N., Almufareh, M. F., & Khalil, M. I. (2022). Security threat and vulnerability assessment and measurement in secure software development. Computers, Materials & Continua/Computers, Materials & Continua (Print), 71(3), 5039–5059. https://doi.org/10.32604/cmc.2022.019289

[27] Hussain, A., Khan, S. U., Khan, N., Shabaz, M., & Baik, S. W. (2024, January). AI-driven behavior biometrics framework for robust human activity recognition in surveillance systems. Engineering Applications of Artificial Intelligence. doi:10.1016/j.engappai.2023.107218

[28] Ioannis Stylios, S. K. (2020). Biometric systems utilising health data from wearable devices: Applications and future challenges in computer security. Information Fusion. doi:10.1016/j.inffus.2020.08.021

[29] Kapron, Z. (2025, January 27). Beyond The Swipe: How Artificial Intelligence Is Redefining Biometrics. Retrieved from Forbes: https://www.forbes.com/sites/zennonkapron/2025/01/27/beyond-the-swipe-how-artificial-intelligence-is-redefining-biometrics/

[30] Keon, B. T. W., Zhe, L. C., Cahyono, R. C., Balakrishnan, S., Sindiramutty, S. R., Wei, G. W., & Hendrawati, T. D. (2025). Enhancing Trust in EEG-based Depression Classification: A LIME-powered XAI Approach to Dissecting Deep Learning Black Box Results. 2025 International Conference on Metaverse and Current Trends in Computing (ICMCTC), 1–6. https://doi.org/10.1109/icmctc62214.2025.11196423

[31] Khan, S., Parkinson, S., Grant, L., Liu, N., & McGuire, S. (2020). Biometric systems utilising health data from wearable devices: Applications and future challenges in computer security. ACM Computing Surveys, 1–29. Retrieved from https://pure.hud.ac.uk/ws/files/20330016/ACM_Computing_Survey_Review_of_using_health_data_as_a_biometric_in_security_accepted_version_.pdf

[32] Khan, S., Parkinson, S., Grant, L., Liu, N., & McGuire, S. (2020). Biometric Systems Utilising Health Data From Wearable Devices: Applications and Future Challenges in Computer Security. ACM Computing Surveys, 1(1), 1-29. doi:10.1145/3400030

[33] Kiyani, F. F., Hamid, B., Humayun, M., Sindiramutty, S. R. a. L., & Chowdhury, S. (2024). Discovery of Influential Publications Using Research Article's Usage Context. 2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC, 1–7. https://doi.org/10.1109/etncc63262.2024.10767576

[34] Kost, E. (2023, August 3). The LastPass Data Breach (Event Timeline And Key Lessons). Retrieved from UpGuard: https://www.upguard.com/blog/lastpass-vulnerability-and-future-of-password-security

[35] Lam Tran, T. H. (2021). Multi-Model Long Short-Term Memory Network for Gait Recognition Using Window-Based Data Segment. IEEE Access, 9. doi:10.1109/ACCESS.2021.3056880

[36] Lee, H., & Kim, S. (2020). Neural network optimization for keystroke dynamics on mobile devices. Computers & Security, 97.

[37] Ling-Feng Shi, Z.-Y. L.-J. (2023, January 11). Novel Deep Learning Network for Gait Recognition Using Multimodal Inertial Sensors. Sensors, 23(2). doi:10.3390/s23020849

[38] Linqiang, Y., Sindiramutty, S. R. a. L., Ashraf, H., Muzammal, S. M., Balakrishnan, S. a. P., Gupta, S., & Kavita, N. (2024). Intelligent Household Waste Classification System Based on Machine Learning. 2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), 760–768. https://doi.org/10.1109/etncc63262.2024.10767563

[39] Mehta, G. B. (2025, August 22). Beyond fingerprints: Power of multimodal biometric authentication. Retrieved from OLOID: https://www.oloid.com/blog/beyond-fingerprints-the-power-of-multimodal-biometric-authentication

[40] NIST. (n.d.). Digital Identity Guidelines: Authentication and Lifecycle Management. Retrieved from NIST: https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-63b.pdf

[41] Oloid Desk. (2025, October 17). Things You Must Know About Behavioral Biometrics. Retrieved from OLOID: https://www.oloid.com/blog/things-you-must-know-about-behavioral-biometrics

[42] Patel, K., & Kim, J. (2022). Hybrid AI models for robust keystroke dynamics authentication. IEEE Transactions on Neural Networks, 30(5), 789–801.

[43] Plurilock. (2024, February 2). Mouse Dynamics. Retrieved from Plurilock: https://plurilock.com/deep-dive/mouse-dynamics/#:~:text=One%20of%20the%20primary%20applications,Continuous%20Authentication

[44] Plurilock. (n.d.). Why is behavioral biometrics better than traditional biometric solutions like fingerprint scans? Retrieved from Plurilock: https://plurilock.com/answers/why-is-behavioral-biometrics-better-than-traditional-biometric-solutions-like-fingerprint-scans/?utm_source=chatgpt.com

[45] Rai, P. (2024, March 26). JSON vs. SQL: What's the Difference? Retrieved from Olibr: https://olibr.com/blog/json-vs-sql-whats-the-difference/

[46] Reiter, R. (2024, January 1). What is Sensitive Data Exposure and How to Prevent It. Retrieved from Sentra: https://www.sentra.io/learn/sensitive-data-exposure

[47] Riskhan, B., Roua, A., Mahaba, B. S., Uswa, D., Gheisari, M., & Sindiramutty, S. R. (2025). Enhancing Diagnostic Accuracy for Breast Cancer Using Classical-Quantum Hybrid and Transfer Learning Technique. Journal of Soft Computing and Data Mining, 6(2), 41–56. https://penerbit.uthm.edu.my/ojs/index.php/jscdm/article/view/22061

[48] Sanjaya, F. N., Balakrishnan, S., Sindiramutty, S. R., & Narputro, P. (2025). XAI Technology in EEG Signal Based Disease Detection Models. 2025 International Conference on Metaverse and Current Trends in Computing (ICMCTC), 1–7. https://doi.org/10.1109/icmctc62214.2025.11196271

[49] Scott, M. O. (2024, May 22). What is Behavioral Biometrics and How Is It Used? Retrieved from PingIdentity: https://www.pingidentity.com/en/resources/blog/post/behavioral-biometrics.html?utm_source=chatgpt.com

[50] Scott, M. O. (2024, May 22). What Is Behavioral Biometrics and How Is It Used? Retrieved from PingIdentity: https://www.pingidentity.com/en/resources/blog/post/behavioral-biometrics.html#Pros-and-Cons-of-Using-Behavioral-Biometrics

[51] Shu Shen, S.-S. S.-J.-C.-Y. (2023, May). A classifier based on multiple feature extraction blocks for gait authentication using smartphone sensors. Computers and Electrical Engineering, 108. doi:10.1016/j.compeleceng.2023.108663

[52] Shuwandy, M. L., Alasad, Q., Hammood, M. M., Yass, A. A., Abdulateef, S. K., Alsharida, R. A., . . . Abd, N. S. (2025). A Robust Behavioral Biometrics Framework for Smartphone Authentication via Hybrid Machine Learning and TOPSIS. Journal of Cybersecurity and Privacy, 20.

[53] Siddiqui, N., Dave, R., Vanamala, M., & Seliya, N. (2022). Machine and Deep Learning Applications to Mouse Dynamics for Continuous User Authentication. Machine Learning and Knowledge Extraction, 4(2). doi:10.3390/make4020023

[54] Subash, A., Song, I., Lee, I., & Lee, J. (2025). Adaptability of current keystroke and mouse behavioral biometric systems: A survey. Computers & Security, 160. doi:10.1016/j.cose.2025.104731

[55] Subash, A., Song, I., Lee, I., & Lee, K. (2025). Adaptability of Current Keystroke and Mouse Behavioral Biometric Systems: A Survey. Computers & Security. doi:10.1016/j.cose.2025.104731

[56] Suleyman, M. &. (2019, July 31). Using AI to give doctors a 48-hour head start on life-threatening illness. Retrieved from DeepMind Blog: https://deepmind.google/discover/blog/using-ai-to-give-doctors-a-48-hour-head-start-on-life-threatening-illness/

[57] Swisher, J. (2024, December 13). How Weak Passwords Expose You to Serious Security Risks. Retrieved from Jetpack: https://jetpack.com/resources/weak-passwords/

[58] Terranova Security. (2023, January 31). Everything You Need to Know About Biometrics Hacking. Retrieved from Terranova Security: https://www.terranovasecurity.com/blog/hacking-biometrics

[59] Traitware. (2020, January 17). Common Issues with Passwords as a means of Authentication. Retrieved from TraitWare: https://traitware.com/common-issues-with-passwords-as-a-means-of-authentication/#:~:text=Password%20authentication%20is%20hard%20to,passwords%20more%20vulnerable%20to%20attackers.

[60] Wang, Q., & Zhang, X. (2023). Hybrid CNN-SVM models for robust keystroke dynamics authentication. IEEE Transactions on Biometrics, Behavior, and Identity Science, 5(3), 412–425.

[61] Weiqi, X., Hooi, S. T. C., Sindiramutty, S. R. a. L., Asirvatham, D. a. L., Kumar, D., & Verma, S. (2024). Surface Anomaly Detection Using Machine Learning Technique. 2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), 1–7. https://doi.org/10.1109/etncc63262.2024.10767562

[62] Wilson, M. (2024, July 12). Under The Microscope: The Risks Of One-Time Passwords' Vulnerabilities. Retrieved from Forbes: https://www.forbes.com/councils/forbestechcouncil/2024/07/12/under-the-microscope-the-risks-of-one-time-passwords-vulnerabilities/

[63] Ying, X., Murugesan, R. K., Sindiramutty, S. R., Wei, G. W., Balakrishnan, S., Kumar, D., & Verma, S. (2024). Scene Text Recognition using Deep Learning Techniques. 2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), 1–9. https://doi.org/10.1109/etncc63262.2024.10767484