Mouse Dynamics Behavioral Biometrics: A Survey

SIMON KHAN, Clarkson University, USA
CHARLES DEVLEN, Clarkson University, USA
MICHAEL MANNO, Clarkson University, USA
DAQING HOU, Clarkson University, USA

Utilization of the Internet in our everyday lives has made us vulnerable in terms of privacy and security of our data and systems. Therefore, there is a pressing need to protect our data and systems by improving authentication mechanisms, which are expected to be low cost, unobtrusive, and ideally ubiquitous in nature. Behavioral biometric modalities such as mouse dynamics (mouse behaviors on a graphical user interface (GUI)) and widget interactions (another modality closely related to mouse dynamics that also considers the target (widget) of a GUI interaction, such as links, buttons, and combo-boxes) can bolster the security of existing authentication systems because of their ability to distinguish an individual based on their unique features. As a result, it can be difficult for an imposter to impersonate these behavioral biometrics, making them suitable for authentication. In this paper, we survey the literature on mouse dynamics and widget interactions dated from 1897 to 2023. We begin our survey with an account of the psychological perspectives on behavioral biometrics. We then analyze the literature along the following dimensions: tasks and experimental settings for data collection, taxonomy of raw attributes, feature extractions and mathematical definitions, publicly available datasets, algorithms (statistical, machine learning, and deep learning), data fusion, performance, and limitations. Lastly, we end the paper with presenting challenges and promising research opportunities.

CCS Concepts: • **Security and privacy** → **Intrusion detection systems**; **Biometrics**; *Multi-factor authentication.*

Additional Key Words and Phrases: Behavioral Biometrics, Mouse Dynamics, Widget Interactions, Machine Learning, Multi-modal Authentication, Fusion

1 INTRODUCTION

Use of the Internet has become a part of our daily routine. Due to such intensity in internet usage, the risk of data breaches is higher than ever. An example of this is large-scale data breaches, such as those indicted at Yahoo and Equifax, that essentially exploited the weaknesses of the traditional username/password authentication scheme [26, 27]. Therefore, it has become imperative for us to find additional ways to protect our data, which need to be low-cost, unobtrusive, and widely available. Physiological-based biometrics such as facial recognition, fingerprints, and iris authentication all require additional hardware and tools, which can be cumbersome due to cost and usability. However, mouse dynamics and widget interactions integrated modules are very inexpensive and unobtrusive technologies that can be implemented without interfering with day-to-day computer operations. Unlike existing knowledge-based authentication, such as passwords, that are based on "what you know", these behavioral biometrics verify a user's identity based on "what you are".

Authors' addresses: Simon Khan, shkhan@clarkson.edu, Clarkson University, 8 Clarkson Avenue, Potsdam, NY, USA, 13699; Charles Devlen, devlencm@clarkson.edu, Clarkson University, 8 Clarkson Avenue, Potsdam, NY, USA, 13699; Michael Manno, mannom@clarkson.edu, Clarkson University, 8 Clarkson Avenue, Potsdam, NY, USA, 13699; Daqing Hou, dhou@clarkson.edu, Clarkson University, 8 Clarkson Avenue, Potsdam, NY, USA, 13699.

ACM 0360-0300/2024/1-ART
https://doi.org/10.1145/3640311

Biometrics, using human characteristics for identification and authentication purposes, has been a subject of scientific research for many years [52]. There are two types of biometrics: one is based on physiological characteristics such as fingerprints, iris, facial recognition, and palm/hand geometry [3], and behavioral biometrics, such as mouse dynamics, widget interactions, keystroke dynamics, swipe dynamics, motion, and walking/gait [58, 112]. The earliest example of behavioral authentication may be found in a 19th century paper about telegraph typing, where a dispatcher of telegraph operations could identify operators based on the rhythmic sound of pressing telegraph keys [14, 17, 106]. A biometrics system can be applied in two types of applications: 1) identification and 2) authentication. For identification, the system looks to find a one-to-many match by searching through all of the user templates to establish a person's identity. For authentication, a system verifies a person's identity with a one-to-one comparison between the current and existing template. In a nutshell, it verifies if the person is the same person who they claim to be [81]. Interestingly, Pusara and Broadley [77] categorize behavioral biometrics into further classifications: direct behavioral biometrics (e.g., mouse dynamics, GUI based events, keystrokes) and indirect behavioral biometrics (e.g., command line interface (CLI), call stacks).

In terms of behavioral biometrics, keystroke dynamics have been the center of research for a longer period of time relative to other behavioral biometrics. For example, Spillane from IBM during the 1970s performed a data collection process with respect to keystrokes for individual identification [99]. The following decades produced more survey papers regarding keystroke dynamics [14, 103] compared to mouse dynamics and widget interactions. In contrast, research about mouse dynamics only started to pick up by the beginning of the 21st century and online interaction-based authentication has only been broadly researched since the beginning of the last decade [4–11, 16, 21, 25, 31, 32, 38, 42, 44, 45, 49, 55, 58, 63, 68, 77, 77, 78, 80, 82, 84, 85, 88–94, 100–102, 115, 116]. In our literature review, we have found only one brief mouse dynamics survey by Revett et al. [80] that describes literature on mouse dynamics. In fact, Revett et al. [80] only reviews four papers [4, 31, 38, 78] about users, raw data, feature extraction, algorithms and metrics. Therefore, a comprehensive up-to-date literature survey is justified for the research community. In addition, we also survey widget interactions as a new novel authentication modality.

This paper is organized as follows: Section 2 presents our paper collection methodology. In Section 3, we discuss mouse dynamics along with widget interactions as part of human psychology. In Section 4, we delve into data collection in terms of tasks and experimental settings as well as public datasets. In Section 5, we describe raw data and define different features mathematically. In Section 6, we survey the different statistical and pattern based algorithms on mouse dynamics as well as deep-learning algorithms. In Section 7, we survey widget interactions as a behavioral biometric. In Section 8, we detail some of the challenges and research opportunities in mouse dynamics and widget interactions. Lastly, Section 9 concludes our survey.

2 PAPER COLLECTION METHODOLOGY

We perform our literature survey by gathering all the pertinent papers via *Google Scholar*. We surveyed a total of 123 papers directly for mouse dynamics and some additional papers related to psychology and HCI. We follow a step-by-step process as follows. We search *Google Scholar* with the following keywords and phrases ("mouse authentication" AND "survey" (51 results), "mouse dynamics" AND "signature authentication" (50 results), "mouse authentication" AND "psychology" AND "survey" (6 results), "mouse dynamics" AND "intrusion detection" (581 results), "mouse dynamics" AND "computer security" (573 results), "mouse dynamics" AND "insider threat" (183 results), "mouse dynamics" AND "masquerade attack" (33 results), "mouse dynamics" AND "spoof attack" (7 results)). Among these search results, we first select papers based on two conditions: papers that are of higher quality (with over 50 citations), yielding ~114 papers, and papers that have less than 50 citations and are relatively new (since 2015), yielding ~461 papers. The number of 50 citations was chosen based on our own experience; a paper published 8 years ago should have gathered more than 50 citations to be considered of "good" quality. We further select each paper based on their relevance to mouse dynamics and widget interactions and their overall quality, yielding a total of 123 papers. We also review other relevant papers cited by the ones selected above. Lastly, we summarize this body of literature along the dimensions of data acquisition, feature extraction, datasets, algorithms and training/testing techniques, and performance measures.

3 PSYCHOLOGY BEHIND BEHAVIORAL BIOMETRICS: MOUSE DYNAMICS AND ONLINE BEHAVIOR

In this section, we survey the psychological impacts such as the cognitive, perceptual, and motoric understanding of humans as they generate mouse dynamics and widget interactions.

3.1 Psychology Behind Mouse Dynamics

As early as the 19^{th} century, Bryan and Harter [17] first observed in their psychological experiments that a dispatcher of telegraph operations can identify a person based on the rhythmic sound of pressing keys. Subsequently, this topic based on interaction with a telegraphic device started to be picked up as a subject of experimental psychology, which aims to provide tangible validations of information theory using different measurement and modeling techniques, such as Fitts' and Hick's laws [29, 40]. Since the 1970s, due to the advent of computers, experimental psychology has been used to reduce "Human-Computer Interaction" (HCI) time to save money and improve usability and operational efficiency. For example, Card, English and Burr [18] use mouse, joystick, step key and text key modalities to select a word on a CRT (Cathode Ray Tube) by moving a cursor from a home position to a target position with varying width (W) and amplitude (A). They find that Fitts' law can be modeled after all four modalities; additionally, from their experiments, mouse is determined to be the fastest among all four modalities.

A substantial amount of HCI literature already exists, which tries to make user interactions more efficient. The goal of HCI is to improve efficiency of user interactions by optimizing the interaction speed (i.e., timing, often in terms of milliseconds). Though mouse dynamics is a behavioral biometric which also derives from user interactions with the computer, the goal is to authenticate users by investigating duration, as well as other features that interact with a system distinctly. Therefore, they both share some common metrics (i.e., interaction speed/timing), but for different purposes. This notion of interaction speed/timing has been studied in the field of experimental psychology called interaction ergonomics. For example, how quickly can a user position the mouse and click on a button or a drop-down menu. What would be the best way to minimize error rates during authentication for these events? For computer usage, interaction ergonomics tries to answer these aforementioned questions in the formulation of two "laws" (i.e., Hick's and Fitts' laws) within experimental psychology [80].

Hick's law stipulates that as the number of choices increases, so does the response time in decision making. Hick performed two types of experiments: the first which demonstrates if someone is well-trained at tasks; the response time should be largely proportional to the information extracted based on multiple choices, and the second, which is a ten-choice experiment, where a trained and an untrained user are convinced to reduce their reaction time by willfully making mistakes in selecting the given choices. In terms of a trained user, the response time is relatively constant with a smaller variance. Conversely, the untrained user has a longer response time with a larger variance. To execute the experiments, Hick provided a theoretical background of "quantity of information" based on Shannon's information theory [87], which is written as:

$$H = -\sum_{1}^{i} P_i log P_i$$

where H is the entropy or expected information, and P_i is the probability of possible alternatives. The apparatus used in the experiments was ten lamps as choices in an irregular circle and Morse Keys to provide responses via pressing [40]. Hick's law has been utilized into current experimental psychology when designing HCI regarding multiple choices. Throughout Hick's experiments, he tried to prove with some degree of evidence that the response time is proportional to the extracted information. However, the relationship between response time and the mode of operation is still unclear. For example, for a trained operator, the response time is proportional to the extracted information due to familiarity with the system (i.e., mode of operation); however, for an untrained operator, the response time may not be proportional to the extracted information due to their unfamiliarity with the system [83].

Fitts initiated multiple experiments involving subjects to make successive responses based on a particular movement that encompass the speed/accuracy tradeoff. In the first experiment, the subject had two stylus of different weights touching two plates with varying widths as targets, which were spread out over varying distances. In the second experiment the subject placed a disc over a pin from one position to another. In the third experiment, the subject transferred the pin from one hole to another. To measure the difficulty level of performing a task and performance rate in all three experiments, an index of difficulty and performance rate were used, which are defined below:

$$I_d = log_2 \frac{2A}{W_s} \ bits/response, I_p = \frac{1}{t} log_2 \frac{2A}{W_s} \ bits/sec$$

where I_d is the level of difficulty, W_s width of the target, A distance from the pointing device to the target, I_p performance rate, and t average duration. Fitts observed that as the distances increase and the widths are kept constant for the target or vice versa, I_d of performing such a task increases. Also, $I_p$increases to a certain level in the beginning, but then falls out of range. This observation is the same across all three experiments. Fitts also demonstrated in all the experiments that the length of time in moving an object between targets increases as the distance between them increases. These observations categorize the foundation of the speed-accuracy tradeoff, which describes that as movement times get shorter and as targets size decreases; error rate increases. In all the experiments, the subjects performed in high performance scenarios with a timer. Fitts' law has been applied into current experimental psychology when using a computing pointer device such as a mouse to make HCI more efficient [18, 29, 80]. However, it is unclear at what point the performance rate of Fitts' law will drop and how that will impact the mouse operations in a real-life scenario.

Apart from Fitts' and Hick's laws, which focus on predicting movement time, movement accuracy and uncertainty (commonly associated with pointing/steering tasks) are other important aspects of psychological and behavioral research in HCI. Many studies focus on static target tasks, where the target does not move. These tasks are similar in nature to many of the everyday tasks people perform on their computers (i.e., clicking a desktop icon or button). As aforementioned, Fitts' law encapsulates the idea of the speed-accuracy tradeoff, which is one of the most important aspects of human performance. MacKenzie [64] and Zhai et al. [113] examined the potential of using statistical properties of pointing accuracy to improve the performance of Fitts' law. MacKenzie emphasized a previously underutilized model that employs a normalization of target width, known as effective width W_e, based on the subject's actual behavior (output condition), rather than expected behavior (input condition). Zhai et al. aimed to form a more complete model that encompassed two different layers of the speed-accuracy tradeoff. The first task-specific layer pertained to the accuracy requirements imposed by the task itself, while the second subjective layer is representative of individual biases in precision that are independent of the task requirements. Zhai et al. also introduced the idea of target utilization, which describes how the spread of movement endpoints deviates from target width. Wobbrock et al. [110] focused on modeling error rate based on pointing accuracy (i.e., whether a button was hit or missed) as derived from Fitts' law and with use of effective width. Zhou et al. [117] discussed movement accuracy with tasks other than pointing as well. Zhou et al. focused on temporally constrained (time constrained) trajectory based tasks (tracing a line with a touch stylus on a touchscreen along tunnels of various widths and lengths). Lee et al. [59, 60] studied temporal pointing, a type of pointing that requires minimal or no spatial aiming, imposes a time constraint for the task, and where input is a discrete event such as a button press. In these studies, they focused on modeling movement uncertainty in dynamic target tasks. Zheng found the mean time of a user's actions (a sequence of point-and-clicks over textual URLs) was proportional to the index of difficulty, which followed Fitts' law in a real world environment. Zheng [114] empirically validated Fitts' law by collecting data from users on a web forum in an unrestricted manner using a mouse to point and click as an action on text-based links. Zheng's work is the only one that validated Fitts' law for mouse authentication datasets, but it was not used directly as a feature in mouse authentication.

The principles of these works have the potential to be applied to mouse dynamics authentication. Considering the width of a target a user is trying to click as well as the amplitude of distance from the start position of the cursor to the target could be useful features following the ideas of W and A in Fitts' law. The principle of effective width W_e as discussed by MacKenzie and Zhai et al. could similarly be used as a feature for classification in mouse dynamics. Effective width shows potential for better characterizing a users behavior since it is based on the measured behavior of a user rather than their expected behavior. Moreover, Zhai et al. showed W_e encompassed both layers of the speed-accuracy tradeoff better than W with different levels of target utilization. The principle of user error prediction as described by Wobbrock et al. could also be adapted to mouse dynamics. Examining the distribution of movement endpoints and determining frequency of error and by what margin could serve as another feature for classification. Another possible feature that could be derived is variability of accuracy, defined as predicted error versus actual. Zhou et al. didn't focus on traditional pointing tasks, but the idea of examining an entire mouse trajectory and again looking at amplitude of distance of the trajectory as well as width of trajectory rather than the width of target could also serve as a feature for classification. Lee et al. didn't focus on tasks where typical features such as velocity and acceleration could be determined, but systems utilizing a fixed cursor like some games do could potentially take advantage of the presented error model in order to better design targets. This area of research is yet to be widely explored, with only Zheng examining Fitts' law as it deals with mouse cursor applications, and as such the greater research community would benefit from more studies in this area.

Another area of research studying the speed-accuracy tradeoff deals with dynamic target based tasks. These tasks involve targets that are not fixed in one position on the screen, and instead are in motion. Jagacinski et al. [51] and Hoffmann [43] both focused on estimating movement time with dynamic target tasks. Jagacinski et al. focused on dynamic tasks with temporal constraints, which are classified by use of a moving target and the need to capture that target within a set time period. Jagacinski et al. also created a new model for index of difficulty I_d incorporating components such as movement time MT, distance amplitude A, width of target W, velocity of target V and three regression coefficients. This model better fit dynamic target tasks than Fitts' original model. Subsequently, Hoffmann mathematically derived two models for predicting movement time with an attempt to take into account the effect of steady state position error on effective target width. These models had shared components of variable of gain K, velocity V, and effective width W_e, with the second model containing additional empirical constants. By applying these models to Jacaginski et al.'s data, Hoffmann found that there was a significantly lower rate of information processing during the accuracy phase compared to the distance-covering phase of the motion. Huang et al. conducted a series of studies to model movement uncertainty and predict error rates in various target selection tasks. In their first study [47], they introduced a Ternary-Gaussian model for 1-D unidirectional moving targets, representing the endpoint distribution as a sum of three Gaussian components: one reflecting user bias, another accounting for uncertainty due to target movement, and the third related to task precision and movement speed. In their subsequent study [48], the researchers extended their modeling to 2-D moving target selection using a 2-D Ternary-Gaussian model, which accounted for endpoints in a new "velocity coordinate system". Finally, in their investigation of crossing-based moving target selection [46], they proposed a Quaternary-Gaussian model, similar to the 1-D Ternary-Gaussian model but adapted for crossing-based selection. Crossing-based selection is selection of a target based on crossing through it, instead of using a discrete means such as pressing a mouse button. Park et al. [75] presented an Intermittent Click Planning (ICP) model that described the process by which users plan and execute click actions and could be directly used for predicting error rate. This model had components of target velocity v_t, target position p_t, target width W, pointer velocity v_p, pointer position p_p, and mean period of click repetition P. This model was also successfully employed to discern cognitive differences among players of a first person shooter game. It was shown that in users playing PlayerUnknown's Battlegrounds that gamers and non-gamers had a similar level of ability to estimate click timing from visual cues, but gamers have better ability to encode the rhythm of clicks with an internal clock.

Dynamic tasks are not yet highly prevalent in the realm of mouse dynamics authentication. However, dynamic authentication systems as a one-time authentication method could be contrived for which the models presented for dynamic targets have potential use. The models for estimating movement time proposed by Jagacinski et al. and Hoffmann et al. have components that could serve as features for classification. Again, the use of distance amplitude A and width W could serve as useful features for classification in the context of the movement trajectory, as well as velocity V of the moving target. The variable of gain K in regards to steady state position could also potentially serve as a useful feature, and effective width W_e also was shown to be of good performance for modeling these tasks. The Gaussian models that Huang et al. proposed could be of use for considering the individual endpoint distributions of dynamic target acquisition tasks in 1D and 2D, which could reveal a user's tendency to hit or miss targets of certain widths and of certain distances from starting points. The models Lee et al. focus on are not of a nature typically found in mouse dynamic tasks, as there is no cursor movement and as such no features to be derived from movement, but the principles of selection uncertainty studied could have potential use for deriving features for mouse dynamics authentication. The ICP model Park et al. proposed also has components that could be used as features for dynamic target tasks, and shows high potential due to already having been used to identify cognitive differences in users. From these studies, it is evident that HCI principles and elements of human psychology have great potential to help classify individuals in a mouse dynamics authentication system, across multiple dimensions, different selection methods, and with both dynamic and static targets.

3.2 Psychological Modeling of Online Behavior

Similarly, widget interactions evolve from HCI. Widget interactions are a special kind of interaction such as hovering over via a cursor on certain widgets (e.g., a button or icon) as a part of online behavior. As mentioned earlier, HCI has been studied for the purpose of improving operational efficiency. On the other hand, the goal of widget interactions is to authenticate users during online activities utilizing time (i.e., milliseconds) as features. Both of them also have common metrics (i.e., timing/duration) like mouse dynamics. There are different kinds of online interactions, such as pageview sessions, monitoring online activities of users (to increase sale), and structured agnostic interaction. Structured agnostic interaction is when a person browses randomly via a mouse on a website for the purpose of extracting mouse movement, speed, curvature, trajectory, etc., instead of widget interactions. In this context, widget interaction behavior is often ignored, and that makes it a novel authentication method [49, 58].

People use different modeling techniques to improve the efficiency of their operations. Efficiency and authentication are inter-related due to the measurement techniques that can be used. One such technique is the GOMS (Goal, Operator, Method and Selection) theory, which is to predict and explain real world experiences of HCI quantitatively through the lens of experimental psychology. By definition, a goal is considered a well-known task within the model, an operator is who performs the task, a method is a way to perform the task, and a selection is to have multiple options to reach the final goal. Due to GOMS' ability to quantify human performance effectively in earlier years, it has become a pioneer to improve the efficiency of HCI by eliminating unnecessary user actions [73]. There are several variants of GOMS, but among the major ones are: CPM-GOMS (Critical

Path Method), KLM-GOMS (Keystroke-Level Model) and NGOMSL (Natural GOMS language). CPM-GOMS extends the GOMS method to model parallel activities among perception, cognition and motor functionalities [34]. KLM-GOMS is considered the simplest variant of GOMS. It uses basic operators like keystrokes, cursor movement time, button presses, and double clicks to estimate task times [53]. Lastly, NGOMSL depends on execution times that are collected based on software by a user performing the tasks with the system [111].

The GOMS theory considers routine perceptual, cognitive and motoric operations of a user for its analysis, such as routine timing information of a keystroke, a single mental operator (i.e., time) to extract the next piece of information from memory, or pointing to a target using a mouse on a display [73]. Gray and Boehm-Davis [33] demonstrates that HCI behavior comprises interaction between artifacts (e.g., a mouse and button) and elementary human cognitive, perceptual, and motoric operations. Their research shows that interactive behavior can be optimized with alternative techniques, such as moving the cursor of a mouse or clicking a button in a certain way, to reduce the cost or time of that interaction and save milliseconds to make HCI more efficient. However, there are shortfalls of using GOMS, such as the lack of parallel modeling based on human perception, cognition and motoric activities; therefore, they use CPM-GOMS to model the parallel activities effectively to reduce time.

Gary, John and Atwood [34] experiments with the CPM-GOMS model on toll and assistance operators (TAO) to find out the possibility of saving $3 million for a company. The major motivation behind the proposal is to optimize interaction with the proposed workstations by reducing average work time per call, which would offset the capital cost compared to older workstations. They use CPM-GOMS to model parallel behaviors displayed by TAOs. Using CPM-GOMS, they are able to determine the proposed workstations would cost $2 million more than older workstations due to the difference in keyboards and screen layouts. As a result, the model is able to show that learning very small differences within the system design can help minimize the cost of operations.

Overall, the purpose of GOMS theories is to improve the efficiency of HCI and the usability of a system. However, as one can see while experimenting with different artifacts/widgets (e.g., different buttons) on a GUI, this provides a conceptual understanding of widget interactions. Hence, we believe that these theories for HCI have provided the very foundation of the widget interactions modality to this day.

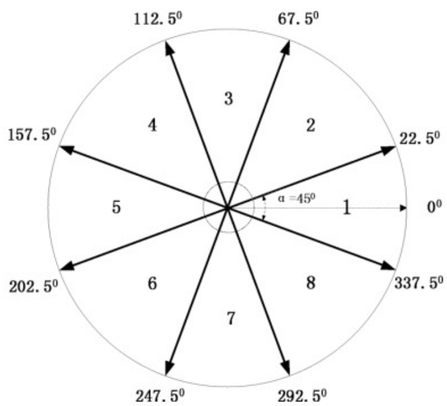
4 DATA COLLECTION FOR MOUSE DYNAMICS: NATURE OF TASKS AND EXPERIMENTAL SETTINGS

Data collection is an inherent part of mouse dynamics research. Many factors can be involved during data acquisition processes. In this section, we categorize the techniques that researchers have applied in collecting raw mouse data, in terms of the nature of the tasks that users are required to perform, as well as the experimental settings [81].

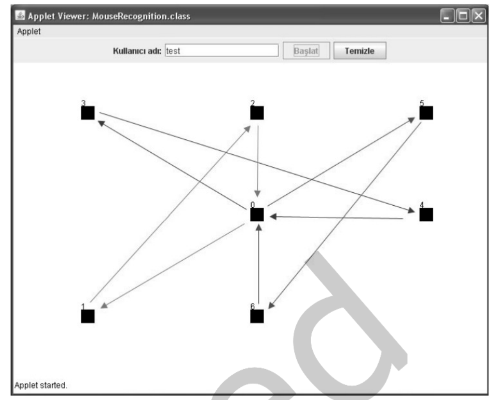
4.1 Nature of Mouse Data Collection Tasks

The nature of mouse data collection tasks may vary (e.g., different tasks, different apps), which can influence user behavior during data collection [4, 7, 8, 28, 78]. Therefore, we divide methods of data collection into four categories based on user actions and the app/data collection software involved as follows: fixed static sequence of actions, app restricted continuous, app agnostic semi-controlled and completely free data collection.

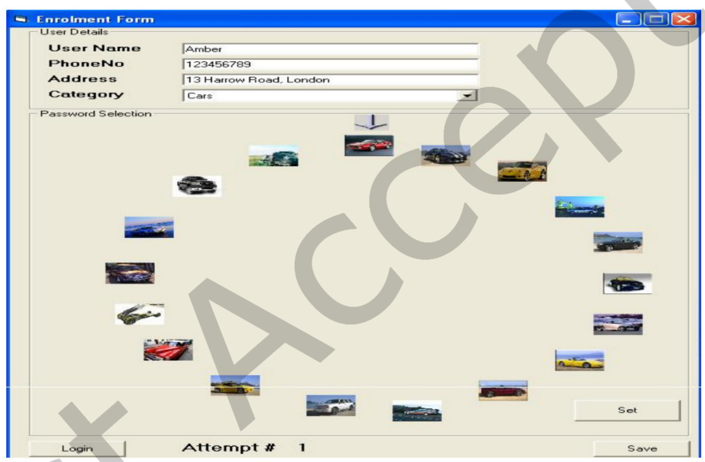
(1) *Fixed static sequence of actions* represents fixed and repeated mouse actions based on the predefined tasks setup by the researchers via apps (i.e., moving a mouse cursor from fixedly situated button A to button B presented by the experiment). For instance, in Shen et al. [90, 91, 93] and Ancien et al. [2], all the participants are asked to perform a fixed set of mouse operating actions to produce patterns. These fixed set of actions are made of 8 consecutive movements isolated by single and double clicks. As a result, the actions generate directions in which each one of them is 45-degrees out of a 360-degree range on a screen (see Figure 1a). Other studies that fall under this category are [6, 11, 19, 38, 56, 80, 84, 85, 100].



(a) Mouse Operation Mode Based on Static Mouse Operating Patterns [90]



(b) Paths Between Squares that a User Must Follow [6]



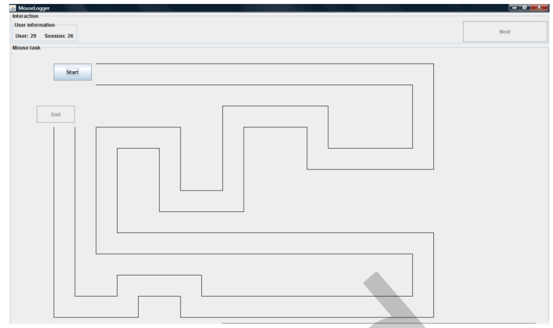
(c) Static Authentication of a Mouse-Lock using Thumbnail Images of Cars [80]

Fig. 1. Examples of Fixed Static Sequence of Actions

(2) *App restricted continuous* data collection represents the scenario where collection limited by the conditions of the specific app. For example, according to Gamboa and Fred [31], they develop a game of grid of 3x6 tiles (i.e., condition of the game), where a user has to match a pair of tiles by clicking on them (see Figure 2a). Other studies that fall under this category are [16, 20, 21, 54, 56, 63, 78, 102].

(3) *App agnostic semi-controlled* data collection represents the type of task that does not depend on the app, per se. However, it depends on many factors other than the app, making the data collection process semi-controlled. For example, Shen et al. [92] describe many factors that contribute to the variability of mouse dynamics data collection aside from apps, such as different brands of mice, GUI settings, emotional state (e.g., anger, despair, happiness, stress, relax), physical condition (e.g., tiredness, illness), distance between

(a) A Memory Game Based on Static Authentication [31]



(b) Navigating the Mouse Pointer following GUI Based Maze [16]

Fig. 2. Examples of App-Restricted Continuous Authentication

the mouse and body, height of the chair and screen size of the computer. Other studies that fall under this category include [8, 22, 55, 88, 115, 116]

(4) *Completely free* data collection represents collection that is free of any set of guidelines for data collection process. For example, Ahmed and Traore [4, 5] collect data using their own software from participants without any restriction. They ask the users to install the software on their computers and perform their routines, but varying activities (e.g., web browsing, word processing). The software runs on the background and monitors user's activity and send to a remote server unobtrusively. Other studies that fall under this category include [7, 9, 10, 10, 20, 21, 25, 32, 42, 44, 45, 68, 70, 82, 86, 89, 94, 101, 115, 116]

From the above classifications, although we can delineate that it may be beneficial in some cases to have a fixed static sequence of actions or app restricted continuous data collection to reduce noise or remove outliers from the experiments, these collection types may not be the best choices due to lessening or restricting the authenticity of a user's behavior. From the empirical perspective, conducting experiments should be geared towards more realistic approaches, such as app agnostic or completely free data collection, to represent the authentic behavior of a user.

4.2 Experimental Settings

Experimental settings have been one of the key parts for a proper mouse authentication system. Researchers refer to different brands of mouses, different sizes of screen layouts, variations of different movement rhythms of a mouse by the user, temperature and lighting of the room, as well as different types of computers and human emotions as parts of experimental settings [4, 7, 15, 16, 63, 68]. An experimental setting is divided into three categories: uncontrolled [16], semi-controlled [58], and controlled [91]. The uncontrolled setting can be defined as an environment where nothing is controlled in the way a user provides information. For instance, a user is asked to download the software on his/her computer (laptop/desktop) and perform routine tasks freely without any observation from the researcher or any other internal/external influence (e.g., room, temperature, chair, computer, lighting). Uncontrolled setting provides the most realistic scenario for tested and deployed systems [16, 115, 116]. Semi-controlled setting is a little different than controlled and uncontrolled settings. It is where some of the aspects of the experiment are controlled and some are not. For example, participants are remotely logged into a computer to perform the tasks, where the usage of the software is the same, but the surrounding environment for each participant is different [58]. Controlled setting is where the participants are asked to partake in the same environmental setting by the researchers. Same environmental setting means same lighting setup of the room, same computer with software installed to record mouse actions, same height of the chair, more or less same distance from the mouse to the body as well as same controlled temperature of the room for all the participants [80, 90, 92].

5 RAW DATA, FEATURES, AND PUBLIC DATASETS

After data collection, researchers conduct further analysis of the collected raw data to extract features, to create substantial meaning for the classifiers to comprehend and perform binary authentication. In this section, we expound classifications of raw data into different categories and how we perform feature extractions using raw data.

5.1 Raw data: Mouse Events and Actions

Figure 3 illustrate two categories of low-level raw data (i.e., mouse events and mouse actions). The mouse event is the lowest level as component raw data, which can be grouped together into meaningful actions as the next level of raw data, which is mouse actions. Mouse raw data is usually captured by a software that runs in the background within a computer. This software is normally custom-made based on the researcher's specifications, which is used to generate sequences of mouse event data, such as mouse down/up, mouse movement and mouse wheel by the user. These data are analyzed as system messages that define the event type, location and time of the mouse cursor. Practically, these generated system messages or low level raw data are not useful enough for analyzing human behavior for classifications. Therefore, they are usually grouped or combined into the next low-level raw data, called mouse actions, to extract features [88]. These mouse actions can be defined as :

- *Mouse Movement* (MM) actions: The mouse movement can be defined as location changes of a mouse without pressing any mouse up or down button [4, 8, 68]. They can also be defined as a mouse trajectory or data points of a time-series [69]. It is normally up to the researcher to decide how many coordinates

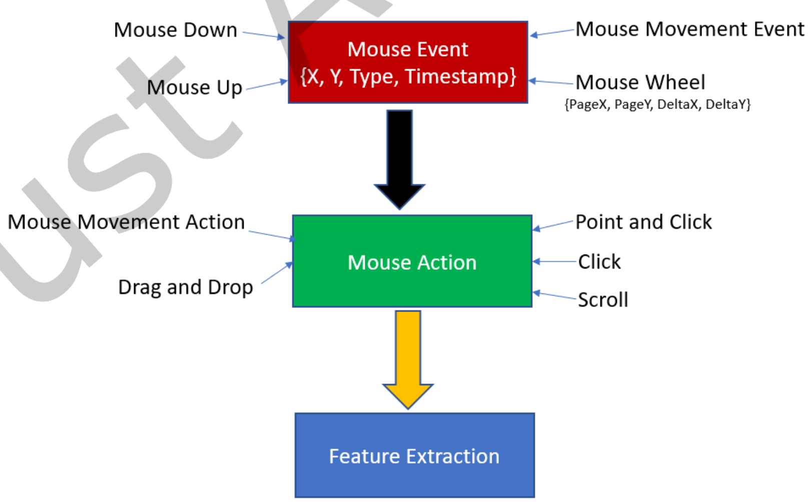
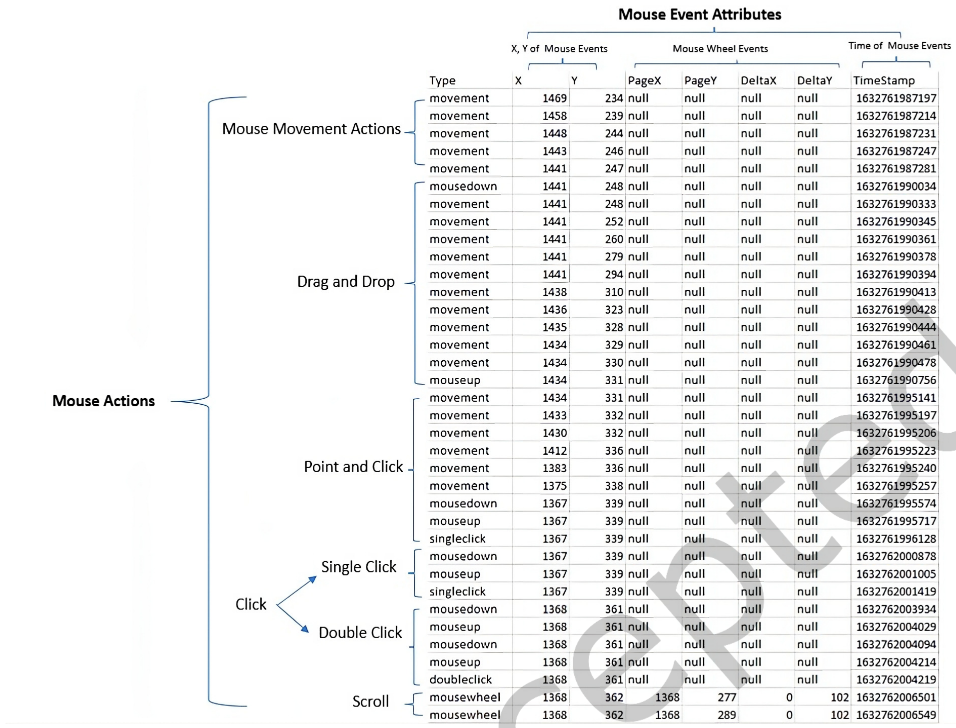


Fig. 3. Raw Data Processing Pipeline: from Mouse Events and Actions to Features



Type	X	Y	PageX	PageY	DeltaX	DeltaY	TimeStamp
movement	1469	234	null	null	null	null	1632761987197
movement	1458	239	null	null	null	null	1632761987214
movement	1448	244	null	null	null	null	1632761987231
movement	1443	246	null	null	null	null	1632761987247
movement	1441	247	null	null	null	null	1632761987281
mousedown	1441	248	null	null	null	null	1632761990034
movement	1441	248	null	null	null	null	1632761990333
movement	1441	252	null	null	null	null	1632761990345
movement	1441	260	null	null	null	null	1632761990361
movement	1441	279	null	null	null	null	1632761990378
movement	1441	294	null	null	null	null	1632761990394
movement	1438	310	null	null	null	null	1632761990413
movement	1436	323	null	null	null	null	1632761990428
movement	1435	328	null	null	null	null	1632761990444
movement	1434	329	null	null	null	null	1632761990461
movement	1434	330	null	null	null	null	1632761990478
mouseup	1434	331	null	null	null	null	1632761990756
movement	1434	331	null	null	null	null	1632761995141
movement	1433	332	null	null	null	null	1632761995197
movement	1430	332	null	null	null	null	1632761995206
movement	1412	336	null	null	null	null	1632761995223
movement	1383	336	null	null	null	null	1632761995240
movement	1375	338	null	null	null	null	1632761995257
mousedown	1367	339	null	null	null	null	1632761995574
mouseup	1367	339	null	null	null	null	1632761995717
singleclick	1367	339	null	null	null	null	1632761996128
mousedown	1367	339	null	null	null	null	1632762000878
mouseup	1367	339	null	null	null	null	1632762001005
singleclick	1367	339	null	null	null	null	1632762001419
mousedown	1368	361	null	null	null	null	1632762003934
mouseup	1368	361	null	null	null	null	1632762004029
mousedown	1368	361	null	null	null	null	1632762004094
mouseup	1368	361	null	null	null	null	1632762004214
doubleclick	1368	361	null	null	null	null	1632762004219
mousewheel	1368	362	1368	277	0	102	1632762006501
mousewheel	1368	362	1368	289	0	102	1632762006549

Fig. 4. Samples of Raw Data From a Real User: Mouse Events (e.g., type, x, y, pageX, pageY, deltaX, deltaY and timestamp) on the top and Actions on the left (e.g., MM, DD, PC, C and S)

would consist of a movement or trajectory (i.e., selecting a certain window size of coordinates or a timing condition).

- *Drag and Drop* (DD): The drag-and-drop is defined as starting the mouse with the button pressed, then dragging it to generate some movement until releasing the button. In other words, it is a sequence of mouse button down, movement and mouse button up [4, 8, 68]; it can also be called a stroke [31] or mouse move DD [28].
- *Point and Click* (PC): When a user points or moves and clicks on an icon or a menu, he/she clicks by pressing and releasing the mouse button, which is called a point-and-click (see Figure 4). PC can be subdivided mainly in two categories: point and single click, and point and double click. They can be further subdivided into point/right click or right double-click and point/left click or left double-click [28].
- *Click* (C): This raw attribute demonstrates only a click without any movement before or within it. It can also be called a pause and click (i.e., where a user pauses for a certain time before using a single/double click or duration between mouse up/down) [44, 115]. Researchers also have been using time to click, paused time, the number of pauses and paused ratio (ratio between the number of pauses and the total duration of the click) before clicking as features [31, 115] for feature extractions.
- *Scroll* (S): Scroll is when a user rolls a wheel on top middle of the mouse to move up or down on the web page. In other words, a scroll consists of a sequence of mouse wheels. For example, Figure 4 shows PageX and PageY to provide coordinates of the mouse wheeling in x and y directions. Additionally, delta values

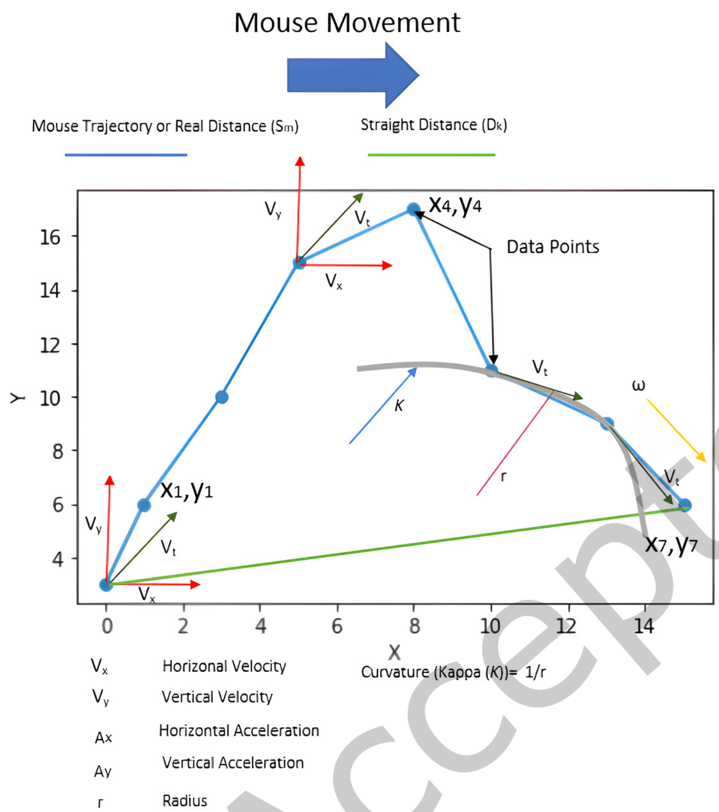


Fig. 5. The Mouse Movement (MM) Consists of Data Points That Define a Trajectory

indicate if the direction of the mouse cursor with respect to the web page is up or down. The delta value can be negative or positive based on a user's mouse wheeling activity [78, 88, 107].

Figure 4 depicts sample mouse data, from which one can observe a sequence of mouse events; and events are grouped into actions. Furthermore, it is evident that every action stems out of mouse movement except clicks, which is measured by duration [8].

5.2 Features

Figure 5 provides a visualization of a typical mouse trajectory, which demonstrates MM actions (Mouse Movement) along with discrete data points for features to be calculated. MM is the most dominant raw attribute to be included in most of the extracted features (although DD, PC, S are different actions, they are all to be extracted for features because of MM). C is the only that can be extracted based on duration without any MM [4, 8, 9, 31, 92]. Table 1 surveys features related to mouse trajectory as a list of feature names, definitions and mathematical formulations. Moreover, it is also common to use standard statistics such as minimum (min), maximum (max), median , mean, max-min, standard deviation (std), variance, skewness and kurtosis as features as well [22, 28, 31]. In the following, as an example we illustrate how to extract one feature for curvature.

Table 1. Mouse Movement (MM) Features Extracted from Raw Data

Feature Name	**Feature Definition**	**Mathematical Formulation**
Traveled Distance (D_i)	Distance between two adjacent mouse positions [8, 93]	$\sqrt{(y_{i+1} - y_i)^2 + (x_{i+1} - x_i)^2}$
Curve length/Real Dist. (S_n)	Length of trajectory traveled [19, 69]	$\sum_{i=0}^{n} D_i$
Elapsed Time/Mouse Digraph	Time elapsed between two events [8, 19, 28, 80, 116]	$\sum_{i=0}^{n-1} (t_{i+1} - t_i)$
Movement Offset	Difference between distance along the curve S_i and straight distance D_i [8, 90]	$S_i - D_i$
Deviation Distance	Maximum distance from mouse trajectory to straight line [63]	$\frac{(y_0 - y_1)x + (x_0 - x_1)y + (x_0 y_1 - x_1 y_0)}{\sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}}$
Straightness, Efficiency	The ratio of straight line distance to trajectory distance [31, 42]	$\frac{D}{S_n}$
Jitter	Tremor of movement measured as the ratio of the interpolated path length I_n and trajectory path S_n [28, 31]	$\frac{I_n}{S_n}$
Velocity (V)	Magnitude and direction of how fast a mouse cursor is moving [66, 67, 93, 105]	$\frac{D_i}{t}$
Horizontal Velocity ($x'_{(2)}$)	Movement speed in x [28, 67, 93, 105]	$\frac{x_2 - x_1}{t_2 - t_1}$
Vertical Velocity ($y'_{(2)}$)	Movement speed in y direction [28, 93]	$\frac{y_2 - y_1}{t_2 - t_1}$
Horizontal Acceleration ($x''_{(3)}$)	Acceleration in x direction [28, 67, 93, 105]	$\frac{x_3 - 2x_2 + x_1}{(t_3 - t_2)^2}$
Vertical Acceleration ($y''_{(3)}$)	Acceleration in y direction [28, 67, 93, 105]	$\frac{y_3 - 2y_2 + y_1}{(t_3 - t_2)^2}$
Speed against distance	Speed as function of traveled distance; obtained through periodic sampling using histogram and represented using discrete form of $Speed_i$ over distance [4, 93]	$(Speed_i, S_i), i = 0, \ldots n$
Average speed against distance	Average speed as function of traveled distance [4, 28, 93]	$(Speed_{(0,i)_{avg}}, S_i), i = 0, \ldots n$
X Acceleration against Distance	Acceleration relative to traveled distance in x direction; can be obtained through periodic sampling and represented in discrete form of acceleration with distance [93]	$(x_{i+2} - 2x_{i+1} + x_i, S_i); i = 0, \ldots n - 2$
Y Acceleration against Distance	Acceleration relative to traveled distance in y [93]	$(y_{i+2} - 2y_{i+1} + y_i, S_i), i = 0, \ldots n$
Avg X Acceleration against Distance	Plot of average movement acceleration with respect to culminating traveled distance by sampling [93]	$(x_{i+2} - 2x_{i+1} + x_i)_{avg}, S_i; i = 0, \ldots n - 2$
Avg Y Acceleration against Distance	Plot of average movement acceleration with respect to culminating traveled distance by sampling [93]	$(y_{i+2} - 2y_{i+1} + y_i)_{avg}, S_i; i = 0, \ldots n - 2$
Tangential Velocity (V_i)	Velocity along a curve [19, 31, 84]	$\sqrt{V_x^2 + V_y^2}$
Tangential Acceleration ($\dot{V}$)	How the tangential velocity of a point on a curve changes relative to time [31]	$\dot{V} = \frac{V_i}{t}\ or A_i = \sqrt{A_x^2 + A_y^2}$
Tangential Jerk ($\ddot{V}$)	Change of mouse acceleration on curve relative to time [28, 31]	$\frac{\dot{V}_2 - \dot{V}_1}{t_2 - t_1}$
Angle of Movement (θ)	Angle of the path tangent to the x axis [28]	$\arctan \frac{y_2 - y_1}{x_2 - x_1}$
Angular Velocity (ω_i)	Rate of change of an angle in a circle [22]	$\frac{V_i}{r}$
Rate of Curvature	Rate in which the mouse curvature is changing direction [28]	$\frac{\delta\kappa}{\delta s}$

Curvature (κ): Geometrically, curvature can be described as one over radius r as if moving along a perfect circle. It can also be described as trajectory curvature [28]. In Figure 5, we can approximate the curvature as a small arc of a circle. The radius of the curvature changes as we move along a trajectory based on different data points. The curvature will be zero if a curve is a straight line and radius will be ∞. In discrete terms, curvature formula is given as [65],

$$\kappa = \frac{x'_{(2)} y''_{(3)} - y'_{(2)} x''_{(3)}}{(x'^2_{(2)} + y'^2_{(2)})^{3/2}}$$

Taking the absolute value of κ will provide the size of the curvature and its radius can be measured as,

$$r = 1/|\kappa|$$

As a result, we can literally calculate the curvature using features $x'_{(2)}, y'_{(2)}, x''_{(3)}, y''_{(3)}$ as formulas from Table 1 by measuring specific data points along the trajectory and plugging them into the curvature equation.

Total Angles, Bending Energy	Summation of angles on trajectory [9, 42]	$\sum_{i=1}^{n-1} \theta_i, \sum_{i=1}^{n-1} \lvert\theta_i\rvert$
Regularity	How regular the distances between points of a curve and its geometric center [42]	$\frac{\mu_d}{\mu_d+\sigma_d} \in [0, 1]$
Trajectory of Center of Mass (TCM)	Mean time during mouse movement where weights are given by traveled distance [28]	$\frac{1}{S_n} \sum_{i=1}^{n-1} t_{i+1} D_i$
Scattering Coefficient (SC)	Deviation of mouse movement from the movement center of mass [28]	$\frac{1}{S_n} \sum_{i=1}^{n-1} {t_{i+1}}^2 D_i - TCM^2$
Curvature Velocity (V_{curve})	Average velocity of curvature [28]	$\frac{\ddot{V}}{(1+\dot{V}^2)^{3/2}}$
Central Moments (μ_n)	Moments around the mean which provides rough idea about shape of curve [42]	$\frac{\sum_{i=1}^{n-2} (\kappa_i-\mu)^n \lvert\theta_i\rvert}{\sum_{i=0}^{n-2} \lvert\theta_i\rvert}$
Self-Intersection	Number of points of a curve intersecting with itself [42]	$\sum_{i=0}^{n-1} x_i, y_i$
Angle Feature (γ) (Law of Cosines)	Angle between the mouse's current location and two click points on squares [6]	$cos^{-1} \left[\frac{a^2+b^2-c^2}{2ab} \right]$
Acceleration Beginning Time	Acceleration Time for the Beginning Segment [39]	$t_1 - t_0$
Skewness	Asymmetry of mouse sequence distributions compared to a normal distribution; also known as the third moment [22, 28]	$\frac{\frac{1}{n} \sum_{i=1}^{n} (X_i-\bar{X})^3}{\sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (X_i-\bar{X})^2}}$
Kurtosis	How much tails of mouse sequence distributions differ from the tails of a normal distribution; also known as the fourth moment	$\frac{\frac{1}{n} \sum_{i=1}^{n} (X_i-X)^4}{\sqrt{\frac{1}{n} \sum_{i=1}^{n} ((X_i-X)^2)^2}}$

5.3 Public Datasets

Although public datasets for mouse dynamics are not as common as keystroke dynamics, there are some publicly available mouse datasets. It is often difficult for researchers to find and compare the works of others because of the lack of standard in data collection. By accepting certain standards during data collection process will help researchers to examine certain algorithms and approaches distinctly. It will also reduce exact replication of the same effort and save time from the perspective of a researcher and the scientific community [71].

The datasets mentioned in the Table 2 of the first row are a supplement to the paper in Shen et al. [91]. This dataset was collected based on a fixed static sequence of actions. Using this dataset, they were able to meet the European standard for commercial biometrics technology [24] if taking a prolonged time for authentication. In the second row, the dataset produced by Shen et al. [88] was collected for continuous authentication in an app agnostic semi-controlled way. In the third row, the dataset developed by Ahmed and Traore [4, 23] under the BioTracker project is completely free, but needs prior approval from the University of Victoria to be used for further research. In the fourth row, the Balabit [13] dataset contains timing and positioning information of mouse pointers. It can be used to authenticate users and also test out performance of different ML algorithms.

Table 2. Publicly Available Mouse Datasets

Authors (dataset)	User Tasks	#Subjects	Amount of Data
Shen et al. [91] (Chaoshen-1)	Fixed Static Sequence of Actions	58	17.4k samples per user
Shen et al. [88] (Chaoshen-2)	App Agnostic Semi-Controlled	28	90k mouse actions over 30 sessions
Ahmed and Traore [23] (ISOT)	Completely Free	48	45 sessions per user for 9 weeks (284 hours of raw data)
Balabit [13]	Completely Free	10	Train set contains avg 937 actions and test set contains avg 50 actions
Antal and Denes-Fazakas [8] (DFL)	Completely Free	21	1k mouse actions
Belman et al. [15] (BB-MAS)	Fixed Static Sequence of Actions	117	Unknown for mouse data
Harilal et al. [37] (TWOS)	App Restricted Continuous	24	320 hours of active participation that included 18 hours of imposter data and at least two instances of insider attack data
Siddiqui, Dave and Saliya [97] (Minecraft Dataset)	Fixed Static Sequence of Actions	10	20 minutes of raw data per user

In the fifth row, Antal and Denes-Fazakas [10, 11] described their own developed dataset called DFL, which is very similar to the Balabit dataset in terms of raw data and also completely free. In sixth row, BB-MAS [15] is a multi-modal large dataset that includes typing, gait, mouse and swipe performed by the same user in a fixed static sequence of actions. In row 7, TWOS (the Wolf of SUTD) [37], dataset was based on a multiplayer game that collects user's mouse behavioral data on simulated interactions with the system making it app restricted continuous data collection. In the last row, Siddiqui, Dave and Saliya [97] developed a fixed static sequence of actions based dataset using Minecraft video game.

6 MACHINE LEARNING ALGORITHMS FOR MOUSE DYNAMICS

In this section, we first classify mouse dynamics research based on statistical and pattern recognition algorithms (Table 3a). Statistical algorithms can be defined as mean, standard deviation, minimum, or maximum, to more complex methods such as T-Test, Euclidean Distance, Manhattan distance, P-Value Test and so on [14]. Conversely, pattern recognition algorithms use various features to find patterns, in order to classify them into different groups, for example, SVM, KNN, GBM, etc. [104]. Based on their contributions, we further group these statistical and pattern recognition algorithms into three categories: 1) Feature Selection (papers that are classified solely based on their feature selection methods as a primary reason to authenticate) 2) Performance Evaluation (papers fall in this category on the sole basis for performance evaluation of different classifiers or models) 3) Spoof Attack (papers that explain mouse authentication to be evaluated against spoof attacks).

6.1 **Feature Selection**

In mouse dynamics research, several feature selection techniques have been explored to improve authentication systems. Zheng, Paloski, and Wang [115, 116] focus on angular-based metrics such as direction, angle of curvature,

Table 3. Mouse Dynamics based on Statistical and Pattern Recognition Algorithms

(a) Gradient Boosting Machines (GBM), Support Vector Machine (SVM), Naive Bayes (NB), K-Nearest Neighbors (KNN), Decision Tree (DT), Random Forest (RF), Distance Metric (DM), Euclidean Distance (ED), Edit Distance (EDD), Error Distance (ERRD), Manhattan Distance (MD), Weibull Distribution (WBD), False Acceptance Rate (FAR), False Rejection Rate (FRR), Equal Error Rate (EER), Area under the Curve (AUC), Average Number of Genuine Actions (ANGA), Average Number of Imposter Actions (ANIA), Not Applicable (NA), Average (Avg), Accuracy (ACC), Cross Validation (CV), MM (Mouse Movement), DD (Drag & Drop), PC (Point & Click)

Authors	User Tasks	# Subjects	Amount of Data	Classifier	Performance Metrics	Training and Testing
Revett et al. [80]	Fixed Static Sequence of Actions	6	Click Duration (100 samples)	DM based on +/- 1.5 standard deviation	FAR (1-4%) and FRR (1-3%)	80/20 split
Bours and Fullu [16]	App Restricted Continuous	28	MM (Avg of 45 sessions per user)	EDD	EER (40.1%)	50/50 split
Zheng, Paloski and Wang [115]	Completely Free and App Agnostic Continuous	30 and 1k	A total of 81,218 PC actions, with average 5,801 actions per user	SVM	FRR (0.86%) FAR (2.96%) in 25 clicks	50/50 split
Gamboa and Fred [31]	App Restricted Continuous	25	MM (5 hours of interaction / 180 strokes per user)	WBD	Mean EER (0.005) with std (0.001) for 100 strokes	50/50 split
Mondal and Bours [68]	Completely Free	49	MM, PC and DD (5000 samples)	SVM	ANGA (NA), ANGA (94)	50/50 split
Shen, Cai and Guan [88]	App Agnostic Semi-Controlled	28	MM, PC, DD, C (90k mouse actions over 30 sessions)	1-class SVM	FAR (0.37%), FRR (1.12%)	1-Class Classification
Shen et al. [93]	Fixed Static Sequence of Actions	26	MM and PC (300 samples per user)	KNN, NN, SVM	EER (2.64% in 110 sec)	1-Class Classification
Shen et al. [90]	Fixed Static Sequence of Actions	37	MM and PC (5550 samples)	1-class SVM	FAR (8.74%) and FRR (7.96%) in 11.8 sec	1-Class Classification
Shen et al. [91]	Fixed Static Sequence of Actions	58	MM and PC (17.4k samples per user)	KNN with MD & ED, 1-class SVM (linear & RBF), K-means, MD	EER (5.68%) with std (4.12)	1-Class Classification
Shen et al. [94]	Completely Free	159	MM, PC, DD, C (1.5m mouse operations for all users)	1 class SVM, KNN, NN	FAR (0.09%), FRR (1%)	1-Class Classification
Shen et al. [89]	Completely Free	20	MM, PC, DD, C, S (600 sessions for all users)	SVM, NN	FAR (1.86%), FRR (3.46%)	50/50 split
Ma et al. [63]	App Restricted Continuous	10	MM and C (500 sessions)	SVM	Accuracy (96.3%), FAR (1.98%), FRR (2.10%)	5-fold CV
Kaixin et al. [55]	App Agnostic Semi-Controlled	12	MM, DD, PC, S (1000 sessions)	SVM	FAR (8.8%), FRR (5.5%) in 30sec	70/30 split
Dominik et al. [22]	App Agnostic Semi-controlled	11	MM, C, S (3283 instance (path) in total)	LibSVM, ANN, DT, RF	(RF) Avg Accuracy Rate (78.1%)	10-fold CV
Almalki, Roy and Chatterjee [7]	Completely Free	10 (Balabit)	MM, DD, PC (Train set contains avg 937 actions (65 sessions) and test set contains avg 50 actions)	DT, KNN, RF	ACC (99.3%), AUC (99.9%)	70/30 split
Tan and Roy [101]	Completely Free	10 (Balabit)	MM (Train set: avg 937 actions (65 sessions) and test set: avg 50 actions)	Linear SVM	Avg EER (0.1829), avg AUC (0.86), avg FAR (0.21), avg FRR (0.0975)	5-fold CV

Antal and Egyed-Zsigmond [9]	Completely Free	10 (Balabit)	MM, DD and PC (Train set contains avg 937 actions (65 sessions) and test set contains avg 50 actions)	RF	Avg EER (18.80%), avg AUC (89.94%)	10-fold CV
Salman and Hameed [82]	Completely Free	48	MM, DD and PC (998 sessions from 48 users)	Gaussian NB	Avg ACC (93.56%), avg FRR (0.822), avg FAR (0.009), avg EER (0.08), avg AUC (0.98)	3-fold CV
Hu et al. [44]	Completely Free	24	PC (5000 samples per user)	RF, GBM, MLP, SVM and CNN	FRR (less than 1%), FAR (less than 1%) with 20 move-to-and-left click except CNN	70/30 split
Aksari and Artuner [6]	Fixed Static Sequence of Actions	10	MM (111 sessions)	DM based on +/- 1.5 SD	FAR (5.9%), FRR (5.9%), EER (5.9%)	90/10 split
Gao et al. [32]	Completely Free	10 (Balabit)	MM (No mention of sample size)	SVM, KNN	FAR (0.075), FRR (0.0664)	Unknown
Zheng, Paloski and Wang [116]	Completely Free	30	MM and PC (3160 mouse actions per user (150 hours of raw data)	SVM	FAR (0.86%) and FRR (2.96%) after 25 clicks	50/50 split
Antal, Fejer and Buza [11]	Fixed Static Sequence of Actions	120	MM (52 blocks (MM into fixed size) per user)	CNN and 1-class SVM	AUC (0.94)	75/25 split
Pusara and Broadley [78]	App Restricted Continuous	18	7.6k unique cursor locations	DT	Avg FAR (0.43%), avg FRR (1.75%)	75/25 split
Antal and Denes-Fazakas [8]	Completely Free (Balabit, DFL)/App Agnostic Semi-controlled (ChaoShen)	21	MM, DD and PC (1k mouse actions)	RF	Avg AUC (0.9922) with std (0.0061)	70/30 split
Jorgensen and Yu [54]	App Restricted Continuous	17	MM, DD, PC for [31] and MM for [4] (325 actions per user)	NN, Logistic Regression	Avg FAR (21% with std 14.3%) and avg FRR (21.5% with std 13.4%) [31]. Avg FAR (30.3% with std 9.8%) and avg FRR (37.1% with std 17.7%) [4]	60/40 split

and curvature distance, making them environment-independent features. Gamboa and Fred [31] authenticate users based on mouse movement strokes using sequential forward selection (SFS) for feature isolation. Shen et al.[89] proposed two feature selection techniques using sequential forward selection (SFS) and Plus-M-Minus-R (PMMR). They found that PMMR with SVM provided superior results in authentication. They later used the *PrefixSpan* algorithm for feature selection from frequent mouse activities, achieving stable mouse patterns on a

Carniero et al. [19]	Fixed Static Sequence of Actions	53	MM, DD and S (two datasets with 2.4k and 162 instances)	NB	ACC (86.4%)	10-fold CV
Ancien et al. [2]	Fixed Static Sequence of Actions	58	MM and PC (15K samples)	RF	Avg ACC (93%) on one mouse trajectory	70/30 split
Kaminsky, Enev and Andersen [56]	Fixed Static Sequence of Actions	15	MM (Unknown samples)	SVM, KNN	ACC (93%)	10-fold CV
Nakkabi, Traore and Ahmed [70]	Completely Free	48	MM, DD and PC (2184 sessions)	Fuzzy Clustering Method	FAR (0%), FRR (0.36%)	1-hold out CV
Schulz [86]	Completely Free	72	MM (3.6k mouse curves)	ED	EER (11.1%)	1 to 1 comparison
Hashia, Pollett and Stamp [38]	Fixed Static Sequence of Actions	15	MM (15-minute block per user)	DM based on +/- 1.5 SD	EER (15%)	15 minutes/last ten states
Syukri, Okamoto and Mambo [100]	Fixed Static Sequence of Actions	21	MM (100 signatures)	ERRD points within threshold (50 pixel)	ACC (93%)	75/25 split
Tan et al. [102]	Completely Free and App Restricted Continuous	10 (Balabit) and 20 (TWOS)	MM (Balabit: Train and test: avg 937 actions (65 sessions)/avg 50 actions) (TWOS: 320 hours of active participation that included 18 hours of imposter data and at least two instances of insider attack data)	SVM, 1D-CNN, 2D-CNN	2D-CNN (Balabit) Baseline AUC 0.96 and EER 0.10; 2D-CNN (TWOS) Baseline AUC 0.93 and EER 0.13	80/20 split

fixed static sequence of actions dataset[88]. In a later study by Shen et al. [94], a similar pattern-growth based mining method was used to extract features from consistent behavioral segments with a completely free dataset, demonstrating the stability of feature selection methods. Hamdy et al. [36] incorporated principles of visual search capability and short-term memory effect into a static biometric authentication system using mouse dynamics. They demonstrated that visual search capability and short term memory were very important features that significantly bolstered performance of their mouse dynamics system. These methods all examined different features that can be used to bolster performance in different datasets. Some features may have better performance with a specific type of dataset, but some in the case of Shen et al. [88, 94] are shown to work on both fixed static sequence of actions and completely free datasets.

6.2 Performance Evaluation

In mouse dynamics, there have been several studies pertaining to the performance evaluation of different classifiers. These studies utilized many different types of datasets, and demonstrated the performance of many

Table 4. Deep Learning-based Mouse Dynamics

(a) Neural Network (NN), Convolutional Neural Network (CNN), Long Short Term Memory (LSTM), MM (Mouse Movement), DD (Drag & Drop), PC (Point & Click)

Authors	**User Tasks**	**# Subjects**	**Amount of Data**	**Classifier**	**Performance Metrics**	**Training and Testing**
Ahmed and Traore [4]	Completely Free	22	MM, DD and PC (45 sessions per user for 9 weeks (284 hours of raw data))	NN	FAR (2.4649%) and FRR (2.4614%)	1-hold out CV
Sayed and Traore [85]	Fixed Static Sequence of Actions	30	MM (4350 samples of gesture templates over 174 sessions)	Learning Vector Quantization (LVQ) NN	FRR (4.59%) FAR (5.26%)	80/20 split
Shen et al. [92]	App Agnostic Semi-Controlled	10	MM, C (300 sessions in total)	NN	FAR (0.55%), FRR (3.0%)	1-hold out CV
Hu et al. [45]	Completely Free	10 (Balabit)	MM, DD, C and S (36k images)	CNN (7 Layers)	FAR (2.94% and FRR (2.28%)	85/15 split
Chong et al. [20]	Completely Free (Balabit) and App Restricted Continuous (TWOS)	10 (Balabit) and 20 (TWOS)	MM ((Balabit) Train and test sets (avg 937 actions (65 sessions)/avg 50 actions) (TWOS) 320 hours of active participation that included 18 hours of imposter data and at least two instances of insider attack data)	SVM, LSTM, CNN-LSTM 1D-CNN, 2D-CNN	2D-CNN (Balabit) Baseline Avg AUC 0.96 and Avg EER 0.10; 2D-CNN (TWOS) Baseline AUC 0.93 and EER 0.13	5-fold CV
Antal and Fejer [10]	Completely Free	10 (Balabit) and 21 (DFL)	MM (370 blocks per user)	1D CNN	AUC (0.98)	80/20 split
Everitt and McOwan [25]	Completely Free	41	MM (unknown samples)	NN	FAR (4.4%), ACC (99%)	4/96 split
Hinbarji, Albatal and Gurrin [42]	Completely Free	10	MM, DD and PC (16.5k actions)	NN	EER (5.3%), Authentication time (18.7 minute)	50/50 split
Li et al. [61]	App Restricted Continuous	6	Keystroke, MM (25 minutes per subject)	SVM	CCR Leave One Subject Out 75.5%; CCR All Subjects 78.9%;	10-fold CV
Hamdy et al. [36]	App Restricted Static	274	MM (2740 sessions)	Weighted-Sum	EER 2.11%	70/30 split
Fu et al. [30]	App Restricted Continuous	18	MM (11 trials per subject)	CNN-RNN, PADTW	CNN RNN EER 2.69%; PADTW EER 8.53%;	5-fold CV
Siddiqui et al. [98]	App Restricted Continuous	40	MM	1D-CNN, LSTM-RNN, ANN, KNN, SVM, RF	1D-CNN Mean ACC 0.8573; Mean FPR 0.1546; Mean F1 0.9099;	50/50 split

Authors	User Tasks	# of Subjects	Amount of Data	Classifier	Performance Metrics	Training and Testing
Kang et al. [57]	App Restricted Continuous	60	MM, Keystroke (5,000–10,000 KLM operators per experiment)	Decision Tree	AUC Photoshop 0.72-0.92; AUC 3DS Max 0.84-0.87;	NA
Lopez et al. [62]	App Restricted Continuous	58	Balabit, TWOs, Typing-BD	SVM, RF, DNN	Mean AUC SVM 0.84; Mean AUC RF 0.86; Mean AUC DNN 0.86;	70/30 split
Hassan et al. [39]	Fixed Static Sequence of Actions	50	MM (3859 total actions)	LightGBM, XGBoost, Logistic Regression, Random Forest, SVC, KNN	ACC LightGBM 0.92; ACC XGBoost 0.88; ACC Logistic Regression 0.77; ACC Random Forest 0.87; ACC SVC 0.73; KNN 0.68;	70/30 split
Shi et al. [95]	Completely Free	41	MM, Keystroke (4 hours per user)	AAN, NN, LR, SVM, DT KNN, NB	ACC AAN 89.22; ACC SVM 71.6496; ACC LR 74.5226; ACC DT 68.4263; ACC NN 76.5779; ACC NB 56.4139; ACC KNN 65.0020	5-fold CV
Abin et al. [1]	Completely Free	12	MM, Keystroke	OCSVM-PSO, SVM, HNS, RF, NB, K-Means Clustering	ACC SVM 34.68%; ACC HNS 88.29%; ACC RF 96.48%; ACC NB 88.29%; ACC K-Means Clustering 52.01%; ACC OCSVM-PSO 97.83%;	80/20 split

different classifiers in regards to them. It is important to note that the best performing classifier varied by data collection type as well as the features extracted from the raw data. Unfortunately, benchmarking studies based on common datasets do not exist.

Jorgensen and Yu [54] investigated two work (Ahmed and Traore [4] and Gamboa and Fred [31]) by mimicking their experiments to determine limitations of their mouse authentication system and evaluate performances. At first, they tried to determine if the environment was tightly controlled how effective a mouse was. Secondly, if the enrollment and verification data were collected in different computing environments how effective it was to authenticate a user. Gamboa and Fred's method performed better in the first experiment, and in the second experiment, average error rates rose for both cases (Ahmed and Traore, and Gamboa and Fred) when training and testing data were collected on two different devices. They recommended that one way to improve error rates is to reduce the noise in the raw data and apply fusion techniques. Conversely, Mondal and Bour [68] used six classifiers for their experiments to evaluate performances. Among them, Support Vector Machine (SVM) performed the best. In a similar manner but with 17 classifiers, Shen et al. [91] established a public dataset for the sake of enhancing research in this area. The Nearest Neighbor with Manhattan using 200 samples gives the best result for authentication. They also explore scalability of the system by ranging users from 2 to 58. Furthermore, they found that at around 22 users, EER becomes stable as a result of adequate genuine users in the authentication system. Almalki, Roy and Chatterjee [7] investigated a practical evaluation of different ML algorithms, also conducted on an open source dataset (ISOT) [5]. Among all the classifiers tested, they found random forest (RF) provided the best AUC and ACC. Tan and Roy [101] investigated the performance of different curve smoothing techniques on the mouse movement sequences for an authentication model. They used three time-series forecasting models: cubic spline, AR (Auto Regressive) Model and lastly, AR with Moving Average (ARMA). They found AR curve fitting technique with the RF classifier showed the superior result. Siddiqui et al. [98] expanded on their earlier public Minecraft dataset [97] by increasing the number of users from 10 to 40 in an attempt to provide the research community with more naturalistic mouse dynamics data. They also sought to give baselines of performance of several classifiers (1D-CNN, LSTM-RNN, ANN, KNN, SVM, RF) for the wider research community and attempted to discover a best performing classifier for mouse dynamics. After analysis of each classifier, it was found that the deep learning models outperformed the machine learning models in accuracy, with ANN performing the best out of all of them. Hassan et al. [39] presented a mouse dynamics model that they used for detecting different mouse users for anti-cheat. 39 different features were extracted from raw mouse movement data across 2 trials. The performance of several different classifiers was tested for this purpose (KNN, SVC, RF, Logistic Regression, XGBoost, and LightGBM). LightGBM was shown to have the best performance, followed by XGBoost, RF, Logistic Regression, SVC, and KNN. Abin et al. [1] presented an anomaly detection method for authenticating users using OCSVM (One Class SVM) with PSO (Particle Swarm Optimization) on a completely free non application constrained dataset. This proposed method was compared against multiple other classifiers, including one unsupervised model (OCSVM-PSO, SVM, HNS, RF, NB, K-Means Clustering). The best performing method was OCSVM-PSO, followed by RF, NB and HNS, K-Means Clustering, and SVM.

6.3 Spoof Attack

In mouse dynamics, many studies have been done to determine how the modality could be spoofed, and what methods performed best for doing so, as well as defenses against these methods. Dominik et al. [22] proposed a mouse behavioral dynamics visualization tool that could be used for forensic purposes to gather and store digital information for any cyber security violations. Their tool worked on several news agency websites to collect mouse data from participants and provided a defense mechanism against spoof attacks. Also in a forensic manner, Hu et al. [44] proposed a mouse movement simulation method to inspect the vulnerability of the existing authentication methods. They created synthetic and simulation data on par with real data, which were moving tracks consisting of points along with timestamps, after identifying different objects (e.g., icons of certain applications) as coordinate representations. Finally, they showed that synthetic data works almost the same as real data and better than simulation data to inspect the vulnerability of existing authentication method. Conversely, Antal and Egyed-zsigmond [9] explored mouse dynamics in light of intrusion detection (i.e. a form of spoof attack). They measured how many mouse actions (MM, PC, DD) are needed to get an optimum result to detect an intrusion and which types of mouse actions are the most important ones. From that dataset, they determine that MM and PC performed almost equally and AUC became 1 after a set of actions. Tan et al. [102] developed a threat model based on prior knowledge of how attackers can bypass an authentication system. They explore three different attack strategies: 1) statistics-based (i.e., assuming that the attacker has access to the recorded target user's data) 2) imitation-based (i.e., trained model produce mouse trajectories to mimic a user's mouse movement sequence also known as teacher-forcing approach, which is based on recurrent neural network [109]), and 3) surrogate-based (i.e., train a substitute classifier assuming that it learns the same functionalities as the target classifier from a substitute dataset and performs a white-box attack starting with random mouse movement sequences and alter them with some constant repetitions [74]). The authors showed that imitation or surrogate based attacks performed better than a statistical approach. Lopez et al. [62] focused on an attack technique for behavioral biometric systems based on reusing genuine user inputs and reapplying them in order to impersonate a user on a mouse and keystroke dynamics protected system. Two approaches were used to perform attacks, 1) SCRAP (Synthetically Composed Replay Attack Procedure) and 2) Adversarial Black-Box Attacks. They tested against 3 different machine learning classifiers, SVM, RF, and DNN. The performance of SCRAP against these three classifiers was higher than that of adversarial machine learning. In order to limit the effectiveness of SCRAP attacks, they also proposed employing adversarial training by training with attack samples, which they found to be a well performing countermeasure. Instead of focusing on just features to improve classification, Fu et al. [30] developed a continuous mouse dynamics authentication system that operated around the principle of "induced expertise". Induced expertise is the notion that if a user interacts with a modified version of a system in their daily tasks, they will collect a certain level of expertise on that system over an untrained user. Two classifiers were used, CNN-RNN and PADTW (an improved version of the Dynamic Time Warping algorithm), with CNN-RNN yielding better performance in every scenario. They found that an inexperienced attacker is more distinguishable than an experienced attacker, but even in the case of an experienced attacker, performance is still excellent with their system.

6.4 Fusion of Mouse Dynamics with Other Modalities

Li et al. [61] developed a user independent model to recognize human attention level. They used three different modalities in order to classify a participants attention level in their data: 1) Facial Expression 2) Eye Gaze 3) Mouse Dynamics. They extracted 7 total mouse features from their raw data, 9 eye gaze features, and 80 facial expression features. They used the wrapper method with a best-first searching approach based on a linear SVM for classification in order to reduce the number of features and to use only important features. They found that mouse dynamics was the worst performing modality for their data, but still offered improvement over their baseline, and that facial expression was their best performing modality. They concluded that human attention level could successfully be classified best by the fusion of the three modalities studied. The poor performance of mouse dynamics compared to other modalities is likely linked to the low feature count, and what features were extracted. Mouse dynamics has been shown to perform at a very high level in other studies we surveyed. However, it is worth noting that the best performance achieved in this paper was through the fusion of the three modalities.

6.5 Deep Learning Based Mouse Dynamics

In this section, we review the literature on mouse dynamics based on deep learning algorithms (i.e., artificial neural network (ANN), which is based on the concept of how biological neurons function together in the human brain to be implemented as a non-linear data modeling tool) [72]. An early example of this research is a study by Everitt and McOwan [25], who proposed a Java based online behavioral biometric authentication system using keystrokes and mouse signatures. The system mainly authenticated using NN in three phases: registration, training and testing. The participants were asked to register and train with their own information, and also provided a test set of forged samples for a random selection of other users via a Web-based applet remotely. A later study by Ahmed and Traore [4] developed a different novel technique that modeled human behavior from captured data and classified it using a three-layer feed forward perceptron Neural Network (NN). They created a concatenation of 39 features from the set of factors and used them as inputs to the NN. The NN gives the same importance to all the features simultaneously as a deciding factor for authentication. For the training phase, they used an NN to train each user and keep the trained network stored in a database. In the testing phase, they loaded the genuine user's stored NN network and compared it with the confidence ratio (CR) with all the users to determine the similarity between two behaviors. Shen et al. [92] addressed the issue of mouse variability by pre-processsing features using principal component analysis (PCA) and feature space analysis, using manifold learning called ISOMAP (isometric mapping)) to reduce dimensionality. They then utilized NN for classification to get the best authentication results [35, 50]). Sayed et al. [85] demonstrated a new mouse dynamics framework using mouse gestures for static authentication. Unlike Ahmed and Traore and Shen et al., Sayed et al. used a data smoothing technique (weighted least-squares regression (WLSR) method and Pierce's criterion) on their data. This was then used with an NN in order to authenticate users. Hinbarji, Albatal and Gurrin [42] proposed a method of authenticating users based on mouse movements using NN. To improve performance, they combined multiple curves (sessions), because a single curve did not contain enough information. Furthermore, they tested their system by increasing the size of signatures by 100, 200 and 300 curves. Chong et al. [21] investigated two open source datasets (Balabit and TWOS) using 2D-CNN in comparison with 1D-CNN and hand crafted features. They also presented a multi-label joint training classifier, where it predicts a set of target labels, in which each label represents a different classification problem. They used transfer learning using GoogleNet architecture, modified to a multi-label architecture. They also used their own mapping methods to convert time series data into 2D image data, where they found fused curve (i.e., mouse curves are meshed together to get a longer curve in 2D image) to provide the best result for their experiments. Hu et al. [45] took a different approach by mapping all basic mouse actions to images and then classify them using CNN. Similarly to Chong et al., they also used the Balabit dataset, but developed their own mapping technique to map time-series data to 2D image data for the CNN input. Their results showed that they can complete user authentication in 1.78s. Similarly to Chong et al. and Hu et al., Antal and Fejer [10] used the Balabit and DFL datasets, but used NN model in order to learn features automatically from the raw data without any feature extraction.

7 WIDGET INTERACTIONS AS A BEHAVIORAL BIOMETRIC AUTHENTICATION SYSTEM

Widget interactions refer to what (e.g., clicking on a button, hover over an icon, moving from a window to window) and how a user interacts (e.g., duration) with the composites of a GUI system [58]. Other authors also call widget interactions a GUI based authentication mechanism [12, 49, 77]. We believe that widget interactions, a new kind of modality, is closely related to mouse dynamics, due to the user's interaction with widgets (e.g., buttons, icons) via a mouse on a GUI. Retrospectively, it is also different from mouse dynamics, because widget interactions only include widgets at the time of interaction for authentication, which is not considered in mouse dynamics. Generic mouse dynamics only analyzes the mouse behavior, such as mouse movement and mouse wheeling, without providing any relevance to the widgets. The main goal of mouse dynamics is to authenticate users based on mouse movement/trajectory patterns. As a result, it can be hypothesized that widget interactions will provide many other discriminating factors that mouse dynamics does not consider.

Earlier papers that discuss a GUI based or GUI related authentication system include Pusara [77], Imsand [49] and Bailey, Okolica and Peterson [12]. Pusara took the next step of proposing a multi-modal re-authentication (continuous authentication) system in a close setting environment (i.e., a strictly controlled environment) with 61 users to apply supervised learning algorithms. She then examined the supervised classifier's ability to detect the unseen users as a part of her experiments. She characterized the GUI events into two categories: spatio-temporal and temporal events. The Spatio-temporal category involves both spatial and temporal activities (e.g., window (i.e., scroll bar, min/max, restore/move), control (i.e., application and process control, open/close), menu (e.g., open, select, navigate, close) and item (e.g., list, button etc.)). She calculated mean, standard deviation and skewness of distance, speed, angle, X, Y coordinates and n-graph related to GUI events. Respectively, the temporal category involves only time features (e.g., icons, dialog, query, combo box and miscellaneous activities). She also calculates

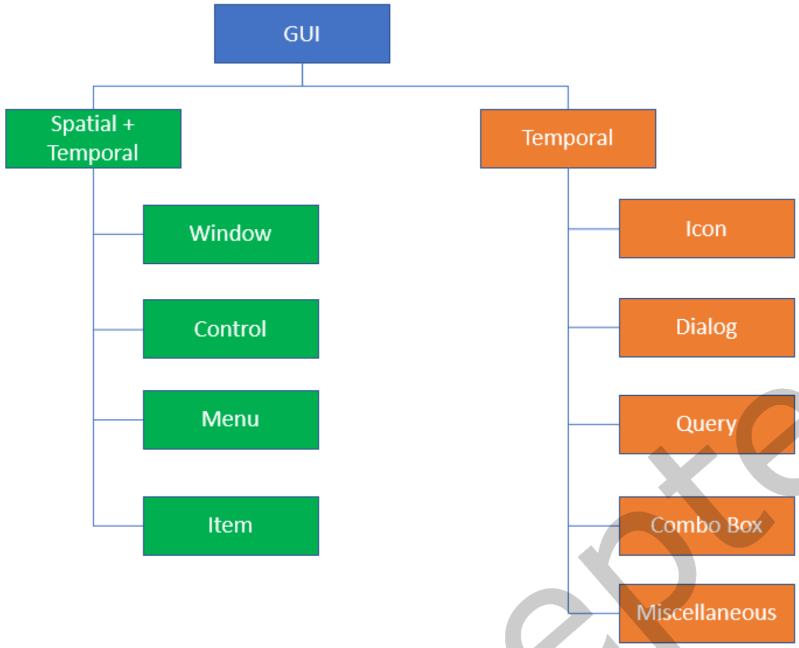


Fig. 6. Pusara's Capture of GUI Events [77]

mean, standard deviation and skewness of *n*-graph duration between temporal events (see Figure 6). Furthermore, she used a decision tree as the final classifier for authentication. The exploratory results are encouraging with FAR (23.37%) and FRR (1.5%) in a detection time of 50s, and with unseen data, the FRR is 2.25% with a detection time of 49.3s. Imsand proposed only a GUI based authentication system on 31 users. His main goal was to authenticate users as well as protect normal users from any masquerade attacks (e.g., disgruntled employee of a company). He collected raw data using the Windows hook procedure [41] based on events (i.e., occurrence of an action such as left button click), object class such as edit, message recipients (e.g., message transmitted from operating system to a running process (e.g., Internet Explorer (IE)), handle (e.g., event specific information such as an ID to keep track of control). He utilized similarity scores based on count to calculate the differences between reference and unknown templates using TF-IDF (term frequency-inverse document frequency) and Jaccard Similarity index. He also experimented with static and variable cutoffs to determine an attack. Variable cutoff was used to label a session being attacked to be customized for each user. The author found that Jaccard coefficient is the most effective way to authenticate users and detect attacks with varying cutoffs of FAR (0%) and FRR (8.66%). Bailey, Okolica and Peterson proposed a multi-modal authentication system (keystrokes, mouse and GUI) with 31 participants. Their raw data collection for GUI system was identical to Imsand [49]. Next, they used two different fusion techniques: feature and decision fusions. Decision fusion provided the best FAR (2.24%) and FRR (2.10%) as an ensemble of all decisions.

Pusara [77], Imsand [49] and Bailey, Okolica and Peterson [12] all proposed a GUI based authentication system in a different way. They all collected data from users performing different tasks (navigation on different websites as a fixed static sequence of actions). Additionally, Pusara, and Bailey, Okolica and Peterson developed a multi-modal authentication system, whereas Imsand developed a system to authenticate users based on a GUI only. Moreover, their authentication methods of using different feature extraction techniques and algorithms were completely different from one another. Pusara experimented with raw data to calculate mean, standard deviation and skewness of distance, angle, speed and duration from the GUI events. On the other hand, Imsand used only a counting based method to calculate the frequency of an event (i.e., frequency of certain words to extract features using TF-IDF to determine the importance of an event to the overall corpus) and Jaccard index (i.e., similarity computation between two sets of data). Bailey, Okolica and Peterson performed two different fusion techniques to improve FAR and FRR for a multi-modal system. They used a window sliding technique in feature level fusion to utilize a chunk of feature samples from keystrokes, mouse activities and GUI at a time for authentication. Hence, these research papers provide a new outlook within the behavioral biometrics community for further explorations.

Our prior publication [58] is closely related to Pusara, Imsand, and Bailey, Okolica and Peterson in terms of data collection process, primarily with respect to the GUI based authentication system. However, we only focused on Facebook to collect raw data (e.g., types of icons, hours-of-day, days-of-week) instead of different websites and word processing activities to authenticate users focusing at a granular level. At first, we invited 8 voluntary users to provide data remotely using our own developed software (a Firefox extension). After collecting raw data, we extracted features using one-hot encoding method. One-hot encoding method is a count based method similar to TF-IDF, where categorical features are assigned with Boolean value of 0 or 1. We also used trigonometric approaches to model our features continuously, such as hour-of-day and day-of-week, as an alternative representation due to the nature of them being ordinal cyclic variables. For authentication, we used two classifiers, SVM and GBM, to measure EER and ACC. Our GBM provided the best results with EER (2.75%) and ACC (97.85%). We also performed an ablation study to investigate the importance of features and measure how overlapping data impact the metrics.

A more recent study by Kang et al. [57] created a model based on the Keystroke Level Model (KLM) for authenticating users based on UI interaction sequences. Mouse movement and keystroke data are both captured in this study, and KLM rules were used to encode the data, which are based off of HCI principles. This essentially directly fuses the two modalities so they can be considered as one for authentication. The use of KLM also makes this authentication system non application constrained. Autodesk 3DS Max and Adobe Photoshop were used to validate the model. Overall they found model performance to be reasonably good, with performance being higher in 3DS Max than in Photoshop. This gives optimism for adopting other HCI models in mouse dynamics authentication research. This is an interesting approach because it deals with fusion of keystroke and mouse for widget dynamics interactions.

8 CHALLENGES AND RESEARCH OPPORTUNITIES IN MOUSE DYNAMICS AND WIDGET INTERACTIONS

Although promising results have been attained through mouse dynamics and GUI based authentication systems, in the following we discuss several remaining challenges before such systems can be deployed.

Data quality can be severely impacted by data interoperability issues when the users use different computer/laptop/mouse during data collection. According to Jain, Ross and Pankanti [81], interoperability issues may arise from different computers, or computer related equipment, having different screen resolutions, speed, memory, different brands of mice, etc. Phillips et al. [76] demonstrated that the effectiveness of facial recognition algorithms on different images is drastically impacted by different camera brands, but there has no investigation of this issue in mouse dynamics.

As mentioned in Section 3, we hypothesize that Fitts' and Hick's laws can be used directly towards mouse authentication and widget interactions [58]. In our own prior study, a user hovered over different widgets on a Facebook platform [58]. Different widgets have different text based lengths, which can be construed as widths

(W) and trajectory distances can be considered as amplitudes (A). According to Fitts' law, this W and A along with I_d and I_p can be used as features for mouse authentication. Other models derived from Fitts' law could also have potential as features, either for predicted movement time versus actual, or predicted error time versus actual. Considering the various applications and dimensions of these models, there is much work needed to be done in performing performance evaluations for each model for mouse dynamics authentication. Similarly, we can also apply Hick's law to use as a feature based on our study. For example, when a user hovers over a like button, multiple choices appear for selection. As a result, we can measure the response time it takes for a user to select a widget as a feature.

In general psychology as a science has been largely ignored in behavioral biometric research. We believe that human psychology can play an important role in both explaining and improving how users are authenticated. For instance, Fitts' and Hick's laws from experimental psychology have been applied in HCI research to make it more efficient, as have multiple models discussing the speed-accuracy tradeoff. However, these laws and models have not been explored by the research community for authentication. Moreover, there are many other unknown factors due to a lack of understanding based on social and behavioral sciences, such as how human emotions and societal situations can affect the system output during authentication [96].

Benchmarking on public datasets is important to increase research generalizability. The mouse dynamics scientific community lacks benchmarking with public datasets to further improve their authentication techniques. Researchers need to objectively measure how well they are doing on a particular problem based on established benchmarks of a particular dataset. Like other behavioral biometric modalities, more research is needed to improve our understanding of both the intra-subject and inter-subject variabilities of mouse dynamics. Mouse dynamics and widget interactions based on a GUI system can be susceptible to other vulnerabilities, such as real data being replaced with synthetic data or different feature sets, hijacking the classifier to falsely allow an imposter to circumvent the system. Furthermore, data can be intercepted and replayed by an attacker with their own data (replay attack) [79].

While the focus of this survey was on mouse dynamics, it would be desirable to include additional consideration to include trackpads for laptop computers. Modern laptop computers offer the speed and power of desktop computers, with the advantage of size, mobility, and convenience. Laptop computers may use trackpads opposed to the traditional mouse. It would be interesting to apply this research from data collected from a trackpad and compare the results to the physical movements of a mouse. Consideration would need to account for the continued point of origin. Meaning, when using a mouse, the device typically is not repositioned after small movements, whereas when using a trackpad, the finger moves the cursor, and is repositioned for comfort and limited range, before continuing to its destination. The length of this range movement of the finger opposed to the mouse is an interesting topic, and one that would require future investigation. Another focus would include touchscreen and handheld devices such as tablets, and cell phones. The evolution of computing devices has introduced devices beyond traditional computers with a mouse. The popularity of small devices that contain sensitive information is increasing with the adoption of cell phones and tablets. In some cases, these have become the primary devices.

Lastly, there remains the question of how the mouse dynamics can be adopted in real-life scenarios. Given the authentication performance reported in the literature, it is unlikely for mouse dynamics to act as a primary authentication modality alone. Therefore, future work is needed to investigate how mouse dynamics may comlement other identification and authentication methods. These would require formal experiments such as Wahab, Hou, and Schuckers [108].

9 CONCLUSION

In this paper, we provided a survey of state-of-the-art behavioral biometric systems that utilize mouse dynamics and widget interactions. Although these modalities have shown promising performance in user authentication, they are also relatively new. Therefore, there exist plenty of research opportunities for further exploration. We began by reviewing the literature that investigates the impact of human psychology on mouse dynamics and widget interactions. We then surveyed the literature on mouse dynamics and widget interactions, along the dimensions of tasks and experimental settings for data collection, public datasets, features, algorithms (statistical, machine learning, and deep learning), and fusion and performance. Lastly, we discussed the challenges that exist when dealing with these modalities, which provided directions for future explorations.

ACKNOWLEDGMENTS

Simon Khan and Michael Manno are also affiliated with and funded by the US Air Force Research Lab (AFRL), Rome, NY. Daqing Hou and Charles Devlen were partially supported by United States NSF award TI-2122746.

REFERENCES

[1] Ahmad Ali Abin, Parisima Hosseini, and Alireza Torabian Raj. 2023. Continuous User Authentication Using a Combination of Operation and Application-related Features. *Journal of Innovations in Computer Science and Engineering (JICSE)* (2023), 11–27.

[2] Alejandro Acien, Aythami Morales, Julian Fierrez, and Ruben Vera-Rodriguez. 2020. BeCAPTCHA-Mouse: Synthetic mouse trajectories and improved bot detection. *arXiv preprint arXiv:2005.00890* (2020).

[3] Olufemi Sunday Adeoye. 2010. A survey of emerging biometric technologies. *International Journal of Computer Applications* 9, 10 (2010), 1–5.

[4] Ahmed Awad E Ahmed and Issa Traore. 2007. A new biometric technology based on mouse dynamics. *IEEE Transactions on dependable and secure computing* 4, 3 (2007), 165–179.

[5] Ahmed Awad E Ahmed and Issa Traore. 2010. Mouse dynamics biometric technology. In *Behavioral Biometrics for Human Identification: Intelligent Applications*. IGI Global, 207–223.

[6] Yigitcan Aksari and Harun Artuner. 2009. Active authentication by mouse movements. In *2009 24th International Symposium on Computer and Information Sciences*. IEEE, 571–574.

[7] Sultan Almalki, Prosenjit Chatterjee, and Kaushik Roy. 2019. Continuous authentication using mouse clickstream data analysis. In *International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage*. Springer, 76–85.

[8] Margit Antal and Lehel Denes-Fazakas. 2019. User Verification Based on Mouse Dynamics: a Comparison of Public Data Sets. In *2019 IEEE 13th International Symposium on Applied Computational Intelligence and Informatics (SACI)*. IEEE, 143–148.

[9] Margit Antal and Elöd Egyed-Zsigmond. 2019. Intrusion detection using mouse dynamics. *IET Biometrics* 8, 5 (2019), 285–294.

[10] Margit Antal and Norbert Fejér. 2020. Mouse dynamics based user recognition using deep learning. *Acta Universitatis Sapientiae, Informatica* 12, 1 (2020), 39–50.

[11] Margit Antal, Norbert Fejér, and Krisztian Buza. 2021. SapiMouse: Mouse Dynamics-based User Authentication Using Deep Feature Learning. In *2021 IEEE 15th International Symposium on Applied Computational Intelligence and Informatics (SACI)*. IEEE, 61–66.

[12] Kyle O Bailey, James S Okolica, and Gilbert L Peterson. 2014. User identification and authentication using multi-modal behavioral biometrics. *Computers & Security* 43 (2014), 77–89.

[13] Balabit. 2016. balabit/mouse dynamics challenge. https://github.com/balabit/Mouse-Dynamics-Challenge

[14] Salil P Banerjee and Damon L Woodard. 2012. Biometric authentication and identification using keystroke dynamics: A survey. *Journal of Pattern Recognition Research* 7, 1 (2012), 116–139.

[15] Amith K Belman, Li Wang, SS Iyengar, Pawel Sniatala, Robert Wright, Robert Dora, Jacob Baldwin, Zhanpeng Jin, and Vir V Phoha. 2019. Insights from BB-MAS–A Large Dataset for Typing, Gait and Swipes of the Same Person on Desktop, Tablet and Phone. *arXiv preprint arXiv:1912.02736* (2019).

[16] Patrick Bours and Christopher Johnsrud Fullu. 2009. A login system using mouse dynamics. In *2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*. IEEE, 1072–1077.

[17] William Lowe Bryan and Noble Harter. 1897. Studies in the physiology and psychology of the telegraphic language. *Psychological Review* 4, 1 (1897), 27.

[18] Stuart K Card, William K English, and Betty J Burr. 1978. Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. *Ergonomics* 21, 8 (1978), 601–613.

[19] Davide Carneiro, Paulo Novais, José Miguel Pêgo, Nuno Sousa, and José Neves. 2015. Using mouse dynamics to assess stress during online exams. In *International Conference on Hybrid Artificial Intelligence Systems*. Springer, 345–356.

[20] Penny Chong, Yuval Elovici, and Alexander Binder. 2019. User authentication based on mouse dynamics using deep neural networks: A comprehensive study. *IEEE Transactions on Information Forensics and Security* 15 (2019), 1086–1101.

[21] Penny Chong, Yi Xiang Marcus Tan, Juan Guarnizo, Yuval Elovici, and Alexander Binder. 2018. Mouse authentication without the temporal aspect–what does a 2d-cnn learn?. In *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE, 15–21.

[22] Dominik Ernsberger, R Adeyemi Ikuesan, S Hein Venter, and Alf Zugenmaier. 2017. A web-based mouse dynamics visualization tool for user attribution in digital forensic readiness. In *International Conference on Digital Forensics and Cyber Crime*. Springer, 64–79.

[23] Ahmed et al. 2007. ISOT Mouse Dynamics Dataset, University of Victoria. https://www.uvic.ca/engineering/ece/isot/datasets/behavioral-biometric/index.php

[24] Standard Number EN 50J33-J : J996IAJ: 2002 Technical Body CLClTC 79 European Committee for Electrotechnical Standardization (CENELEC) 2002 European Standard EN 50133-1 Access control systems for use in security applications, Part 1: System requirements. [n. d.]. https://standards.iteh.ai/catalog/standards/clc/3f3cd487-5dcd-45be-a3ff-ae9ab5d69eac/en-50133-1-1996

[25] Ross AJ Everitt and Peter W McOwan. 2003. Java-based internet biometric authentication system. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25, 9 (2003), 1166–1172.

[26] FBI. 2017. U.S. Charges Russian FSB Officers and Their Criminal Conspirators for Hacking Yahoo and Millions of Email Accounts. https://www.justice.gov/opa/pr/us-charges-russian-fsb-officers-and-their-criminal-conspirators-hacking-yahoo-and-millions

[27] FBI. 2020. Chinese Military Personnel Charged with Computer Fraud, Economic Espionage and Wire Fraud for Hacking into Credit Reporting Agency Equifax. https://www.justice.gov/opa/pr/chinese-military-personnel-charged-computer-fraud-economic-espionage-and-wire-fraud-hacking

[28] Clint Feher, Yuval Elovici, Robert Moskovitch, Lior Rokach, and Alon Schclar. 2012. User identity verification via mouse dynamics. *Information Sciences* 201 (2012), 19–36.

[29] Paul M Fitts. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology* 47, 6 (1954), 381.

[30] Shen Fu, Dong Qin, George Amariucai, Daji Qiao, Yong Guan, and Ann Smiley. 2022. Artificial Intelligence Meets Kinesthetic Intelligence: Mouse-based User Authentication based on Hybrid Human-Machine Learning. In *Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security*. 1034–1048.

[31] Hugo Gamboa and Ana LN Fred. 2003. An Identity Authentication System Based On Human Computer Interaction Behaviour.. In *PRIS*. 46–55.

[32] Lifang Gao, Yangyang Lian, Huifeng Yang, Rui Xin, Zhuozhi Yu, Wenwei Chen, Wei Liu, Yefeng Zhang, Yukun Zhu, Siya Xu, et al. 2020. Continuous authentication of mouse dynamics based on decision level fusion. In *2020 International Wireless Communications and Mobile Computing (IWCMC)*. IEEE, 210–214.

[33] Wayne D Gray and Deborah A Boehm-Davis. 2000. Milliseconds matter: An introduction to microstrategies and to their use in describing and predicting interactive behavior. *Journal of experimental psychology: applied* 6, 4 (2000), 322.

[34] Wayne D Gray, Bonnie E John, and Michael E Atwood. 1993. Project Ernestine: Validating a GOMS analysis for predicting and explaining real-world task performance. *Human-computer interaction* 8, 3 (1993), 237–309.

[35] Mehul Gupta. 2020. Dimension Reduction using Isomap. https://medium.com/data-science-in-your-pocket/dimension-reduction-using-isomap-72ead0411dec

[36] Omar Hamdy and Issa Traoré. 2011. Homogeneous physio-behavioral visual and mouse-based biometric. *ACM Transactions on Computer-Human Interaction (TOCHI)* 18, 3 (2011), 1–30.

[37] Athul Harilal, Flavio Toffalini, John Castellanos, Juan Guarnizo, Ivan Homoliak, and Martín Ochoa. 2017. Twos: A dataset of malicious insider threat behavior based on a gamified competition. In *Proceedings of the 2017 International Workshop on Managing Insider Security Threats*. 45–56.

[38] Shivani Hashia, Chris Pollett, and Mark Stamp. 2005. On using mouse movements as a biometric. In *Proceeding in the International Conference on Computer Science and its Applications*, Vol. 1. The International Conference on Computer Science and its Applications (ICCSA …, 5.

[39] Hadeer A Hassan Hosny, Abdulrahman A Ibrahim, Mahmoud M Elmesalawy, and Ahmed M Abd El-Haleem. 2022. An Intelligent Approach for Fair Assessment of Online Laboratory Examinations in Laboratory Learning Systems Based on Student's Mouse Interaction Behavior. *Applied Sciences* 12, 22 (2022), 11416.

[40] William E Hick. 1952. On the rate of gain of information. *Quarterly Journal of experimental psychology* 4, 1 (1952), 11–26.

[41] Hickeys. 2012. Hooks Overview - Win32 apps. https://docs.microsoft.com/en-us/windows/win32/winmsg/about-hooks

[42] Zaher Hinbarji, Rami Albatal, and Cathal Gurrin. 2015. Dynamic user authentication based on mouse movements curves. In *International Conference on Multimedia Modeling*. Springer, 111–122.

[43] Errol R Hoffmann. 1991. Capture of moving targets: a modification of Fitts' Law. *Ergonomics* 34, 2 (1991), 211–220.

[44] Shujie Hu, Jun Bai, Hongri Liu, Chao Wang, and Bailing Wang. 2017. Deceive mouse-dynamics-based authentication model via movement simulation. In *2017 10th International Symposium on Computational Intelligence and Design (ISCID)*, Vol. 1. IEEE, 482–485.

[45] Teng Hu, Weina Niu, Xiaosong Zhang, Xiaolei Liu, Jiazhong Lu, and Yuan Liu. 2019. An insider threat detection approach based on mouse dynamics and deep learning. *Security and Communication Networks* 2019 (2019).

[46] Jin Huang, Feng Tian, Xiangmin Fan, Huawei Tu, Hao Zhang, Xiaolan Peng, and Hongan Wang. 2020. Modeling the endpoint uncertainty in crossing-based moving target selection. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–12.

[47] Jin Huang, Feng Tian, Xiangmin Fan, Xiaolong Zhang, and Shumin Zhai. 2018. Understanding the uncertainty in 1D unidirectional moving target selection. In *Proceedings of the 2018 CHI conference on human factors in computing systems*. 1–12.

[48] Jin Huang, Feng Tian, Nianlong Li, and Xiangmin Fan. 2019. Modeling the uncertainty in 2D moving target selection. In *Proceedings of the 32nd annual ACM symposium on user interface software and technology*. 1031–1043.

[49] Eric Shaun Imsand. 2008. *Applications of GUI usage analysis*. Auburn University.

[50] Zakaria Jaadi. [n. d.]. A Step-by-Step Explanation of Principal Component Analysis (PCA). https://builtin.com/data-science/step-step-explanation-principal-component-analysis

[51] Richard J Jagacinski, Daniel W Repperger, Sharon L Ward, and Martin S Moran. 1980. A test of Fitts' law with moving targets. *Human Factors* 22, 2 (1980), 225–233.

[52] Anil K Jain, Arun Ross, and Sharath Pankanti. 2006. Biometrics: a tool for information security. *IEEE transactions on information forensics and security* 1, 2 (2006), 125–143.

[53] Bonnie E John and David E Kieras. 1996. The GOMS family of user interface analysis techniques: Comparison and contrast. *ACM Transactions on Computer-Human Interaction (TOCHI)* 3, 4 (1996), 320–351.

[54] Zach Jorgensen and Ting Yu. 2011. On mouse dynamics as a behavioral biometric for authentication. In *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*. 476–482.

[55] Wang Kaixin, Liu Hongri, Wang Bailing, Hu Shujie, and Song jia. 2017. A User Authentication and Identification Model Based on Mouse Dynamics. In *Proceedings of the 6th International Conference on Information Engineering*. 1–6.

[56] Ryan Kaminsky, Miro Enev, and Erik Andersen. 2008. Identifying game players with mouse biometrics. *University of Washington. Technical Report* (2008).

[57] Shin Jin Kang and Soo Kyun Kim. 2023. User Interface-Based Repeated Sequence Detection Method for Authentication. *Intelligent Automation and Soft Computing* 35, 3 (2023), 2573–2588.

[58] Simon Khan, Cooper Fraser, Daqing Hou, Mahesh Banavar, and Stephanie Schuckers. 2021. Authenticating Facebook Users Based on Widget Interaction Behavior. In *2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 1–8.

[59] Byungjoo Lee, Sunjun Kim, Antti Oulasvirta, Jong-In Lee, and Eunji Park. 2018. Moving target selection: A cue integration model. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–12.

[60] Byungjoo Lee and Antti Oulasvirta. 2016. Modelling error rates in temporal pointing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 1857–1868.

[61] Jiajia Li, Grace Ngai, Hong Va Leong, and Stephen CF Chan. 2016. Multimodal human attention detection for reading from facial expression, eye gaze, and mouse dynamics. *ACM SIGAPP Applied Computing Review* 16, 3 (2016), 37–49.

[62] Christian López, Jesús Solano, Esteban Rivera, Lizzy Tengana, Johana Florez-Lozano, Alejandra Castelblanco, and Martín Ochoa. 2023. Adversarial attacks against mouse-and keyboard-based biometric authentication: black-box versus domain-specific techniques. *International Journal of Information Security* (2023), 1–21.

[63] Lei Ma, Chungang Yan, Peihai Zhao, and Mimi Wang. 2016. A kind of mouse behavior authentication method on dynamic soft keyboard. In *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 000211–000216.

[64] I Scott MacKenzie. 1992. Fitts' law as a research and design tool in human-computer interaction. *Human-computer interaction* 7, 1 (1992), 91–139.

[65] Math24. 2021. Curvature and Radius of Curvature. https://www.math24.net/curvature-radius

[66] Mathworks. 2019. How to calculate the second and third numerical derivative of one variable f(x). https://www.mathworks.com/matlabcentral/answers/496527-how-calculate-the-second-and-third-numerical-derivative-of-one-variable-f-x

[67] MIT. 2021. Lecture Notes. http://web.mit.edu/16.unified/www/FALL/systems/Lab_Notes/traj.pdf

[68] Soumik Mondal and Patrick Bours. 2013. Continuous authentication using mouse dynamics. In *2013 International Conference of the BIOSIG Special Interest Group (BIOSIG)*. IEEE, 1–12.

[69] Soumik Mondal and Patrick Bours. 2017. A study on continuous authentication using a combination of keystroke and mouse biometrics. *Neurocomputing* 230 (2017), 1–22.

[70] Youssef Nakkabi, Issa Traoré, and Ahmed Awad E Ahmed. 2010. Improving mouse dynamics biometric performance using variance reduction via extractors with separate features. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 40, 6 (2010), 1345–1353.

[71] MS Obaidat and DT Macchairolo. 1994. A multilayer neural network system for computer access security. *IEEE Transactions on Systems, Man, and Cybernetics* 24, 5 (1994), 806–813.

[72] Mohammad S Obaidat and David T Macchiarolo. 1993. An online neural network system for computer access security. *IEEE Transactions on Industrial electronics* 40, 2 (1993), 235–242.

[73] Judith Reitman Olson and Gary M Olson. 1995. The growth of cognitive modeling in human-computer interaction since GOMS. In *Readings in Human–Computer Interaction*. Elsevier, 603–625.

[74] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. 2017. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*. 506–519.

[75] Eunji Park and Byungjoo Lee. 2020. An intermittent click planning model. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13.

[76] P Jonathon Phillips, Alvin Martin, Charles L Wilson, and Mark Przybocki. 2000. An introduction evaluating biometric systems. *Computer* 33, 2 (2000), 56–63.

[77] Maja Pusara. 2007. *An examination of user behavior for user re-authentication*. Ph. D. Dissertation. Purdue University.

[78] Maja Pusara and Carla E Brodley. 2004. User re-authentication via mouse movements. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*. 1–8.

[79] Nalini K Ratha, Jonathan H Connell, and Ruud M Bolle. 2001. An analysis of minutiae matching strength. In *International Conference on Audio-and Video-Based Biometric Person Authentication*. Springer, 223–228.

[80] Kenneth Revett, Hamid Jahankhani, Sergio Tenreiro De Magalhaes, and Henrique MD Santos. 2008. A survey of user authentication based on mouse dynamics. In *International Conference on Global e-Security*. Springer, 210–219.

[81] Arun A Ross, Karthik Nandakumar, and Anil K Jain. 2006. *Handbook of multibiometrics*. Vol. 6. Springer Science & Business Media.

[82] Osama A Salman and Sarab M Hameed. 2018. Using mouse dynamics for continuous user authentication. In *Proceedings of the Future Technologies Conference*. Springer, 776–787.

[83] Timothy A Salthouse. 1986. Perceptual, cognitive, and motoric aspects of transcription typing. *Psychological bulletin* 99, 3 (1986), 303.

[84] Bassam Sayed. 2009. *A static authentication framework based on mouse gesture dynamics*. Ph. D. Dissertation.

[85] Bassam Sayed, Issa Traoré, Isaac Woungang, and Mohammad S Obaidat. 2013. Biometric authentication using mouse gesture dynamics. *IEEE systems journal* 7, 2 (2013), 262–274.

[86] Douglas A Schulz. 2006. Mouse curve biometrics. In *2006 Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference*. IEEE, 1–6.

[87] Claude E Shannon and Warren Weaver. 1949. The mathematical theory of information. *Urbana: University of Illinois Press* 97 (1949).

[88] Chao Shen, Zhongmin Cai, and Xiaohong Guan. 2012. Continuous authentication for mouse dynamics: A pattern-growth approach. In *IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2012)*. IEEE, 1–12.

[89] Chao Shen, Zhongmin Cai, Xiaohong Guan, and Jinpei Cai. 2010. A hypo-optimum feature selection strategy for mouse dynamics in continuous identity authentication and monitoring. In *2010 IEEE International Conference on Information Theory and Information Security*. IEEE, 349–353.

[90] Chao Shen, Zhongmin Cai, Xiaohong Guan, Youtian Du, and Roy A Maxion. 2012. User authentication through mouse dynamics. *IEEE Transactions on Information Forensics and Security* 8, 1 (2012), 16–30.

[91] Chao Shen, Zhongmin Cai, Xiaohong Guan, and Roy Maxion. 2014. Performance evaluation of anomaly-detection algorithms for mouse dynamics. *computers & security* 45 (2014), 156–171.

[92] Chao Shen, Zhongmin Cai, Xiaohong Guan, Huilan Sha, and Jingzi Du. 2009. Feature analysis of mouse dynamics in identity authentication and monitoring. In *2009 IEEE International Conference on Communications*. IEEE, 1–5.

[93] Chao Shen, Zhongmin Cai, Xiaohong Guan, and Jialin Wang. 2012. On the effectiveness and applicability of mouse dynamics biometric for static authentication: A benchmark study. In *2012 5th IAPR International Conference on Biometrics (ICB)*. IEEE, 378–383.

[94] Chao Shen, Yufei Chen, Xiaohong Guan, and Roy A Maxion. 2017. Pattern-growth based mining mouse-interaction behavior for an active user authentication system. *IEEE transactions on dependable and secure computing* 17, 2 (2017), 335–349.

[95] Yutong Shi, Xiujuan Wang, Kangfeng Zheng, and Siwei Cao. 2023. User authentication method based on keystroke dynamics and mouse dynamics using HDA. *Multimedia Systems* 29, 2 (2023), 653–668.

[96] Ben Shneiderman. 1980. *Software psychology: Human factors in computer and information systems (Winthrop computer systems series)*. Winthrop Publishers.

[97] Nyle Siddiqui, Rushit Dave, and Naeem Seliya. 2021. Continuous Authentication Using Mouse Movements, Machine Learning, and Minecraft. *arXiv preprint arXiv:2110.11080* (2021).

[98] Nyle Siddiqui, Rushit Dave, Mounika Vanamala, and Naeem Seliya. 2022. Machine and deep learning applications to mouse dynamics for continuous user authentication. *Machine Learning and Knowledge Extraction* 4, 2 (2022), 502–518.

[99] R Spillane. 1975. Keyboard apparatus for personal identification. *IBM Technical Disclosure Bulletin* 17 (1975), 3346.

[100] Agus Fanar Syukri, Eiji Okamoto, and Masahiro Mambo. 1998. A user identification system using signature written with mouse. In *Australasian conference on information security and privacy*. Springer, 403–414.

[101] Yi Xiang Marcus Tan, Alexander Binder, and Arunava Roy. 2017. Insights from curve fitting models in mouse dynamics authentication systems. In *2017 IEEE Conference on Application, Information and Network Security (AINS)*. IEEE, 42–47.
[102] Yi Xiang Marcus Tan, Alfonso Iacovazzi, Ivan Homoliak, Yuval Elovici, and Alexander Binder. 2019. Adversarial attacks on remote user authentication using behavioural mouse dynamics. In *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–10.
[103] Pin Shen Teh, Andrew Beng Jin Teoh, and Shigang Yue. 2013. A survey of keystroke dynamics biometrics. *The Scientific World Journal* 2013 (2013).
[104] Sergios Theodoridis, Aggelos Pikrakis, Konstantinos Koutroumbas, and Dionisis Cavouras. 2010. *Introduction to pattern recognition: a matlab approach.* Academic Press.
[105] Unknown. 2021. Applications of the Derivative. https://understandingcalculus.com/chapters/06/6-2.php
[106] John R Vacca. 2007. *Biometric technologies and verification systems.* Elsevier.
[107] W3Schools. [n. d.]. MouseWheel-Deltay. https://www.w3schools.com/jsref/event_wheel_deltay.asp
[108] Ahmed Anu Wahab, Daqing Hou, and Stephanie Schuckers. 2023. A User Study of Keystroke Dynamics as Second Factor in Web MFA. In *Proceedings of the Thirteenth ACM Conference on Data and Application Security and Privacy* (Charlotte, NC, USA) *(CODASPY '23)*. Association for Computing Machinery, New York, NY, USA, 61–72. https://doi.org/10.1145/3577923.3583642
[109] Ronald J Williams and David Zipser. 1989. A learning algorithm for continually running fully recurrent neural networks. *Neural computation* 1, 2 (1989), 270–280.
[110] Jacob O Wobbrock, Edward Cutrell, Susumu Harada, and I Scott MacKenzie. 2008. An error model for pointing based on Fitts' law. In *Proceedings of the SIGCHI conference on human factors in computing systems.* 1613–1622.
[111] Changxu Wu and Yili Liu. 2008. Queuing network modeling of transcription typing. *ACM Transactions on Computer-Human Interaction (TOCHI)* 15, 1 (2008), 1–45.
[112] Roman V Yampolskiy and Venu Govindaraju. 2008. Behavioural biometrics: a survey and classification. *International Journal of Biometrics* 1, 1 (2008), 81–113.
[113] Shumin Zhai, Jing Kong, and Xiangshi Ren. 2004. Speed–accuracy tradeoff in Fitts' law tasks—on the equivalency of actual and nominal pointing precision. *International journal of human-computer studies* 61, 6 (2004), 823–856.
[114] Nan Zheng. 2014. Exploiting behavioral biometrics for user security enhancements. (2014).
[115] Nan Zheng, Aaron Paloski, and Haining Wang. 2011. An efficient user verification system via mouse movements. In *Proceedings of the 18th ACM conference on Computer and communications security.* 139–150.
[116] Nan Zheng, Aaron Paloski, and Haining Wang. 2016. An efficient user verification system using angle-based mouse movement biometrics. *ACM Transactions on Information and System Security (TISSEC)* 18, 3 (2016), 1–27.
[117] Xiaolei Zhou, Xiang Cao, and Xiangshi Ren. 2009. Speed-accuracy tradeoff in trajectory-based tasks with temporal constraint. In *Human-Computer Interaction–INTERACT 2009: 12th IFIP TC 13 International Conference, Uppsala, Sweden, August 24-28, 2009, Proceedings, Part I 12.* Springer, 906–919.