arXiv:2504.21415v1 [cs.CR] 30 Apr 2025

Optimizing Mouse Dynamics for User Authentication by Machine Learning: Addressing Data Sufficiency, Accuracy-Practicality Trade-off, and Model Performance Challenges

Yi Wang*
The University of Tokyo
Tokyo, Japan
yiwangyww@gmail.com

Chenyv Wu*
Wuhan, China
wcy981021@gmail.com

Yang Liao
Xi'an Jiaotong Univeristy
Xi'an, China
ly905650639@gmail.com

Maowei You
Beijing, China
maowei.you@foxmail.com

***Abstract*—User authentication is essential to ensure secure access to computer systems, yet traditional methods face limitations in usability, cost, and security. Mouse dynamics authentication, based on the analysis of users' natural interaction behaviors with mouse devices, offers a cost-effective, non-intrusive, and adaptable solution. However, challenges remain in determining the optimal data volume, balancing accuracy and practicality, and effectively capturing temporal behavioral patterns. In this study, we propose a statistical method using Gaussian kernel density estimate (KDE) and Kullback-Leibler (KL) divergence to estimate the sufficient data volume for training authentication models. We introduce the Mouse Authentication Unit (MAU), leveraging Approximate Entropy (ApEn) to optimize segment length for efficient and accurate behavioral representation. Furthermore, we design the Local-Time Mouse Authentication (LT-AMouse) framework, integrating 1D-ResNet for local feature extraction and GRU for modeling long-term temporal dependencies. Taking the Balabit and DFL datasets as examples, we significantly reduced the data scale, particularly by a factor of 10 for the DFL dataset, greatly alleviating the training burden. Additionally, we determined the optimal input recognition unit length for the user authentication system on different datasets based on the slope of Approximate Entropy. Training with imbalanced samples, our model achieved a successful defense AUC 98.52% for blind attack on the DFL dataset and 94.65% on the Balabit dataset, surpassing the current sota performance.**

***Index Terms*—User Authentication, Pattern Recognition**

I. Introduction

User authentication is essential to ensure secure access to computer systems and prevent unauthorized usage [1]. Traditional authentication methods, such as passwords, auxiliary devices, and biometric recognition, have several limitations. Passwords are susceptible to being guessed, forgotten, or reused across multiple accounts, leading to security vulnerabilities [2]. Auxiliary devices, such as security tokens, can be costly, prone to loss or theft, and add complexity to the user experience. Biometric recognition, such as facial recognition, while more secure, faces challenges such as being tricked by photos or fake videos, high costs, privacy concerns, and varying accuracy due to environmental factors and user appearance changes. In contrast, mouse dynamics, which involves analyzing user behavior through mouse movement patterns, offers a promising alternative. As a means of secondary auxiliary authentication, mouse dynamics authentication is difficult to replicate, protects user privacy, and does not require additional hardware, making it a cost-effective, non-intrusive, and highly adaptable solution. The collection of mouse dynamics data is imperceptible to users and does not disrupt their normal operations or user experience [3]. Using the natural interactive behaviors of users, this approach eliminates the need for additional user actions. Consequently, it improves the convenience and fluency of the user experience while simultaneously ensuring system security.

Using computers or other devices equipped with touchpads or mouse input systems, we define authorized users of these devices as legitimate users, while all other individuals are considered unauthorized users. To achieve mouse-dynamics-based authentication, researchers worldwide have invested significant effort and resources into collecting data sets related to mouse dynamics [39], in order to identify user behavior patterns and design user authentication systems. Mouse dynamics datasets typically record behavioral data in time series when users interact with mouse devices. The datasets include cursor positions, kinematic features such as speed and acceleration, and event data such as single-click actions, double-click actions, and scroll wheel events. Given these datasets, existing research has proposed various methods for identifying unauthorized users based on mouse dynamics [9]–[12], [19], [21]–[26], [40], [41],. These methods generally involve two steps: first, extracting hand-crafted features from mouse dynamic sequences; second, applying machine learning or deep learning techniques to classify these features for user authentication.

However, existing mouse dynamics-based behavioral authentication systems face the following key challenges:

(i) Determining the appropriate amount of data for effective user authentication remains unresolved. Similar questions have been explored in other fields [43]–[45], but not in mouse dynamics authentication. To address this, we propose a method

for estimating the required dataset size, avoiding issues of insufficient or excessive data, and providing guidance for experiment design.

(ii) The length of data segments significantly affects recognition accuracy and real-time performance. Short segments (1–2 seconds) improve responsiveness but lack sufficient behavioral information, reducing accuracy. Longer segments (30 seconds or more) capture richer features but are impractical in scenarios requiring real-time performance.

(iii) Mouse dynamics data includes dimensions like time, speed, acceleration, and direction, often noisy and redundant. Traditional models, such as SVMs and Decision Trees, rely on manually extracted features, which are limited to basic statistics and fail to capture complex behavioral patterns.

As shown in Figure 1, in this study, we employ statistical methods to address the data volume required for mouse user behavior authentication, aiming to achieve a balance between accuracy and practicality. We propose Local-Time Mouse Authentication (LT-MAuthen), a mouse user verification framework that integrates both local and long-term temporal information.

To construct a user authentication model based on mouse dynamics, we discuss and analyze the raw data content, collection environments, and dataset sizes of different types of mouse dynamics datasets. To ensure adaptability across various devices, reduce model parameter complexity, and enhance inference speed, we compute mouse movement velocity as an input variable for the user authentication system. Furthermore, we propose a general statistical method to determine the appropriate total data volume for modeling user mouse behavior. This method utilizes Gaussian Kernel Density Estimation (KDE) and evaluates the similarity between two density functions with different data volumes using the Kullback–Leibler (KL) divergence. If adding more data results in only minimal changes to the density function—indicated by a KL divergence below a predefined threshold, it suggests that the additional data contributes no new information, and the current data volume is deemed sufficient.

In practical model training and system deployment, the collected mouse dynamics data must be segmented into multiple short sequences, which serve as inputs for user authentication. We define a Mouse Authentication Unit (MAU) as a smaller, independent temporal sequence segment extracted from continuous mouse trajectory data. Each MAU represents an analyzable behavioral fragment containing sufficient dynamic information to support feature extraction and pattern recognition for user identity verification. To ensure that each MAU encapsulates adequate information without being excessively long—thus compromising system efficiency—we introduce the concept of Approximate Entropy (ApEn) to measure the information content of authentication units. As the length of the MAU increases, approximate entropy decreases while the information content rises, enhancing the discriminability of user behavior. This approach enables a flexible trade-off between authentication speed and recognition accuracy, dynamically determining the optimal MAU length to meet system performance requirements across various application scenarios.

Additionally, to effectively integrate both local and global features of mouse movement sequences, we design a two-step scheme called Local-Time Mouse Authentication. In the first step, a 1D-ResNet is employed to analyze the local features of mouse velocity sequences. In the second step, the trained blocks of the 1D-ResNet are transferred and combined with a GRU (Gated Recurrent Unit) to capture the temporal characteristics within mouse velocity sequences.

Our main contributions are shown in below:

1) We propose a statistical method using KDE and KL divergence to determine the optimal data volume for mouse dynamics authentication model training.
2) We introduce the Mouse Authentication Unit (MAU) with Approximate Entropy (ApEn) to balance authentication accuracy and efficiency.
3) We design the LTMouseAuthen framework, combining 1D-ResNet for local feature extraction and GRU for temporal pattern modeling.

The remainder of this paper is organized as follows: Section 2 reviews related work in biometric authentication and mouse dynamics. Section 3 presents our analysis of mouse dynamic data. Sections 4 and 5 detail our methods for determining appropriate data volume and MAU length, respectively. Section 6 describes the proposed LTMouseAuthen framework. Section 7 presents comprehensive experimental results and comparisons. Section 8 discusses limitations and future directions, followed by conclusions in Section 9.

II. Related Work

Biometric-based User Authentication Biometric features like keystroke dynamics, mouse movements, and user-system interactions [51] have been widely used for user authentication. Early work, such as typing patterns [52], dates back to the 1990s. With the growth of large datasets, machine learning methods have enhanced biometric identification. Bailey [15] used a GUI to collect keystroke and mouse dynamics data, applying deep learning for identity verification. Meng [16] proposed touch gesture-based authentication on mobile devices, using dynamic training to adapt to data variations. Buriro [17] combined micro-movements, touch strokes, and facial features with random forests and MLPs for classification.

Other biometrics, such as fingerprint recognition [59] and voice biometrics [60], also offer reliable authentication. Recent studies combine these with behavioral features, like keystroke dynamics and facial recognition, to improve robustness [61].

User Authentication on Mouse Dynamics Ahmed et al. [24] first applied machine learning to mouse dynamics, achieving a FAR of 2.46% and FRR of 2.46%. Shen et al. [41], [49] used PCA and stream learning to address behavior interference, while Xu et al. [50] applied random forests to reduce overfitting. More recent approaches, such as [25], categorize users into groups to reduce authentication time, and [26] used CNNs for deeper feature extraction. Penny et al. [10] combined CNNs and RNNs for enhanced feature capture.

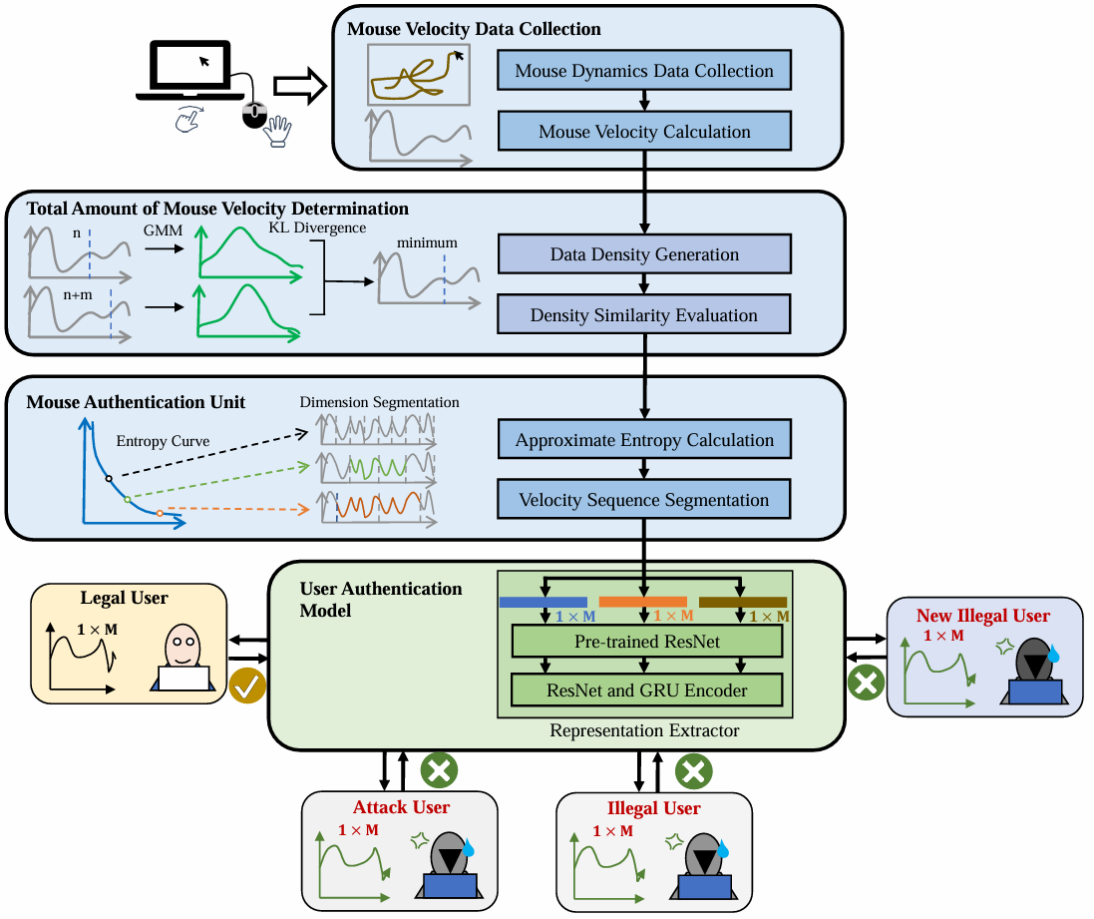


Fig. 1: Structure of the Paper

TABLE I: MAJOR MOUSE DYNAMICS DATASETS

Dataset Name	Time	Content	User Amount	Environment
Mouse-Behavior Data for Continuous Authentication [49]	2012	Timestamp, Click, State, ID, X and Y	28	Experiment Collection Software
Balabit [39]	2016	Timestamp, Click, State, X and Y	10	Daily Usage
DFL [40]	2018	Timestamp, Click, State, X and Y	21	Experiment Collection Software
Minecraft-Mouse-Dynamic-Dataset [22]	2022	Timestamp, Click, Scroll, State, ID, X and Y	10	Daily Usage

Margit et al. [12] released the SapiMouse dataset, and Siddiqui et al. [22] introduced a dataset from Minecraft to avoid dataset homogeneity. However, challenges remain, including the complexity of multidimensional data, non-standardized sequence lengths, and the need for better feature extraction methods.

III. Analysis of Mouse Dynamic Data

Mouse dynamics datasets shown in Table I are widely utilized in behavioral biometric authentication as well as in cognitive and psychological research, which are primarily collected from two distinct environments:

1) Daily usage environments
2) Controlled laboratory environments, with standardized mouse task software

Compared to laboratory environments, collecting mouse or touchpad usage data from daily usage environments can more accurately reflect the natural characteristics of user behavior. However, this approach is more costly and time-consuming. As shown in Figure2 and Figure 3, datasets collected from users' daily usage environments, such as the Balabit dataset, are significantly smaller than those obtained from laboratory environments, such as the DFL dataset. To avoid issues of insufficient datasets failing to adequately represent data characteristics, or excessive data leading to wasted time and resources, determining a reasonable data volume is essential.

Due to differences in application scenarios and data collection methods, the mouse dynamics datasets presented in the Table I have varying feature contents, particularly in the State

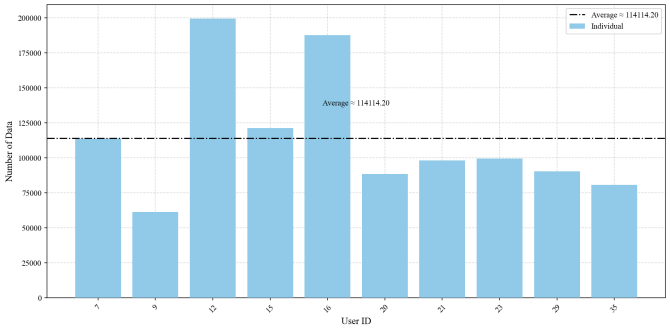


Fig. 2: Amount of Individual User Mouse Behavior in the Balabit Dataset

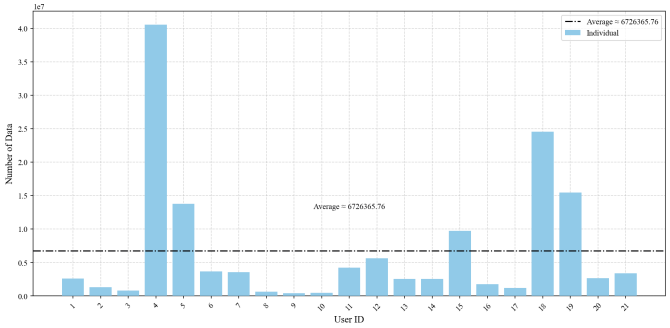


Fig. 3: Amount of Individual User Mouse Behavior in the DFL Dataset

attribute. For instance, Balabit [39] and DFL [40] datasets' states include dragging, double-click and so on, while Mouse-Behavior Data for Continuous Authentication [49] contains left/right button clicks and movement with left or right button held down, etc. However, all datasets share the basic fundamental movement state. Therefore, before determine a reasonable data volume, we need to choose the variables to describe the mouse dynamic behaviors for our authentication task. To make our method more generalized and reduce the complexity of data processing, we utilize the moving velocity information of user mouse movements as the input data for the authentication system. Furthermore, relying solely on velocity data without directly storing mouse position information enhances user privacy protection and minimizes the risk of sensitive information leakage.

Given mouse movement trajectory of one specific user as $x = \{(x^i, y^i)\}_{i=1}^N$, where x^i and y^i represent the mouse coordinates at the $i-$th time point. First, to convert $\{(x^i, y^i)\}_{i=1}^N$ to velocity sequence $v = \{v^i\}_{i=1}^N$, we calculate the Euclidean distance d^i between adjacent points:

$$d^i = \sqrt{(x^i - x^{i-1})^2 + (y^i - y^{i-1})^2} \tag{1}$$

Ignoring device latency, the time interval at each time point is fixed and denoted as Δt. Therefore, the mouse velocity sequence v^i is given by:

$$v^i = \frac{d^i}{\Delta t} \tag{2}$$

IV. Appropriate Volume of Mouse Dynamic Data Determination

Overly small volume of mouse dynamic data may fail to capture the user's unique operational patterns, while excessively big amount of data may introduce redundant information and high cost. Therefore, determining the optimal volume of mouse dynamics data is crucial for real user authentication task. To establish a unified determination paradigm, we can assume that the user's mouse behavior is largely influenced by the uncertainty caused by the surrounding environment (such as desktop space and current tasks) and the user themselves (such as emotions), but over a long period of use, the mouse dynamics data of individual users will tend to converge, i.e., the user's mouse dynamics data will contain unique and stable identity characteristics.

Based on the this assumption, we estimate the appropriate data quantity by calculating the convergence point of the density function derived from the collected data [**?**]. We use $F(v; n)$ to express the distribution of user$j$ mouse velocity data sequence $v = \{v^i\}_{i=1}^N$, and its density is $f(v; n) = \frac{d}{dx} f(v; n)$ under n time. The density of distribution $f(v; n)$ will vary with different data quantities n. If a sufficient amount of data is provided, the observed density will no longer exhibit significant changes with the addition of extra data. For example, if the quantity $\hat{n}$ represents a sufficient amount of mouse dynamics data for user authentication, then the density $p(x, \hat{n}$ should exhibit only minor changes when supplemented with an additional m length of data, i.e.,

$$f(x; \hat{n}) \approx f(x; \hat{n} + m) \tag{3}$$

If adding more data does not alter the original data distribution too much, the additional data is considered redundant. Therefore, the observed data quantity may be appropriate because: i) This data volume can capture almost all the potential features of user mouse behavior; ii) Adding more data cannot provide additional useful information but will increase data collection difficulty and computational overhead for the model. As shown in Fig4, We generate random data following a normal distribution $\mathcal{N}(0, 1)$ for specified sample sizes n, m, r, s, and t, where the intervals are identical, but the sample sizes r, s and t are significantly larger than n and m. In the figure4, the distributions for sample sizes n and m are similar, whereas the distributions for sample sizes r, s and t exhibit minimal differences, almost overlapping. It indicates that when the dataset is sufficiently large, adding more data has a negligible impact on altering the overall distribution of the dataset.

We use Gaussian mixture model (GMM) [46] to define $f(x; n)$, regarding the complex probability distribution of mouse velocity data as a weighted combination of multiple Gaussian distributions to fit the complex data distribution flexibly. Given a mouse velocity dataset $\{v^i\}_{i=1}^N$ with the

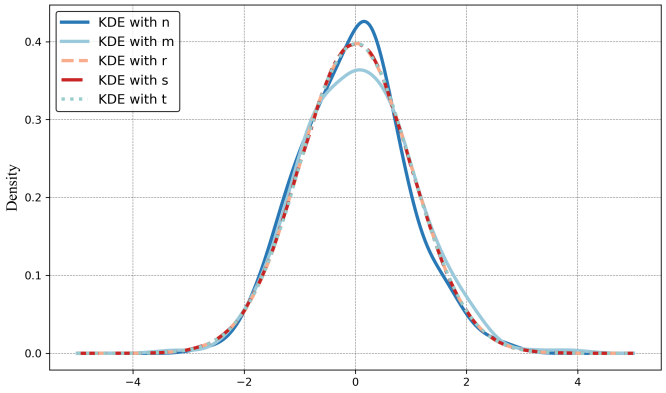


Fig. 4: Illustrationsof the different distribution density with different amount of data for normal distribution $N(0,1)$ example

density function $f(x;n)$, the density function from data sample v can be estimated by [47]

$$f(x;n) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{(2\pi)^{\frac{D}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(v-v^i)^\top \Sigma^{-1}(v-v^i)\right) \tag{4}$$

where D is the dimension of the mouse velocity data; Σ is the covariance matrix, i.e., bandwidth in 1D data, $|\Sigma|$ is the determinant of bandwidth. In this work, we estimated the bandwidth by [42]

$$h = 1.06 \cdot \hat{\sigma} \cdot n^{-\frac{1}{5}} \tag{5}$$

where $\hat{\sigma}$ is the standard deviation of the dataset $\{v^i\}_{i=1}^N$.

Thus, we can drive the density function $f(v;n)$ from the mouse velocity data $\{v^i\}_{i=1}^N$. We will assess the similarity between two adjacent kernel functions estimated from n and $n+m$ data observations. Utilizing Kullback-Liebler (KL) divergence [48], we can evaluate the difference from different density:

$$KL(f(v;n+m)||f(v;n)) = \int \left[f(v;n+m)\times \log \frac{f(v;n+m)}{f(v;n)}\right] \tag{6}$$

Therefore, the KL can qualify the level of similarity between two density given by mouse velocity data with different length. When $KL(f(v;n+m)||f(v;n))$ approaches 0, it indicates that $f(v;n)$ is extremely close to $f(v;n+m)$, where the additional data would not supply more useful information to density function. Furthermore, it is essential to calculate the KL divergence between $f(v;n+m)$ and $f(v;n+2m)$, as we aim to ensure that the decrease in KL divergence is not abrupt but instead shows a smooth and convergent behavior to ensure our data is enough.

We thus determine the proper amount n of mouse velocity data so that $KL(f(v;n+m)||f(v;n))$ itself is small and change very slightly, even more data added:

$$\begin{cases} |KL\big(f(v;n+m)\|f(v;n)\big)| \le \epsilon_1, \\ |KL\big(f(v;n+2m)\|f(v;n+m)\big) - KL\big(f(v;n+m)\|f(v;n)\big)| \le \epsilon_2 \end{cases} \tag{7}$$

where ϵ is a small positive value. It is obvious that a larger value of ϵ_1 and ϵ_2 can lead a small amount of required mouse velocity data.

V. Mouse Authentication Unit Length Determination

Having established the amount of data required for the user authentication system to learn mouse movement patterns, we now determine the size of the Mouse Authentication Unit (MAU). Each MAU represents a time-bounded segment of mouse movement data, serving as the foundational element for user verification models. Adjusting the length of the MAU based on each user's mouse behavior pattern preserves as much valuable information as possible while minimizing the introduction of redundant data, thereby improving the efficiency of the authentication system.

Because information and data predictability are closely linked to data complexity, i.e., entropy, we employ Approximate Entropy (ApEn) [5] in this study to estimate the information complexity and determine an appropriate MAU length.

Given one individual user's mouse velocity data sequence $\{v^i\}_{i=1}^n$ with certain n-dimension, we form a length-m MAU $v(i)$:

$$v(i) = (v^i, v^{i+1}, ..., v^{i+m-1}), i = 1, 2, ..., n-m+1 \tag{8}$$

Therefore, we create a set $\{v(i)\}_{i=1}^{n-m+1} = \{v(1), v(2), ..., v(n-m+1)\}$ containing all length-m MAU.

Next, for each pair of length-m MAU $v(p)$ and $v(q)$, we use Chebyshev distance to measure how "close" they are:

$$d[v(p), v(q)] = \max_{s=1,2,...,m} |v^{p+s-1} - v^{q+s-1}| \tag{9}$$

In ApEn analysis, each length-m MAU $v_j(p)$ is compared against all others $v(q)$ in time series to determine how many of the distances between them lie within a specified tolerance r. Formally,

$$C_p^m(r) = \frac{\#\{p \ne q \,|\, d[v(p), v(q)] \le r\}}{N-m+1} \tag{10}$$

where $d[,]$ denotes the Chebyshev distance metric, and the numerator $\#\{p \ne q \,|\, d[v(p), v(q)] \le r\}$ counts how many mouse velocity windows remain sufficiently close to $v(p)$ under the threshold r. Consequently, $C_p^m(r)$ serves as a measure of the local similarity or "cohesion" for each mouse velocity windows $v(p)$ and forms the basis for evaluating the overall regularity or predictability of the mouse velocity sequence when computing ApEn.

Then, we repeat the same procedure for another length-$m+1$ MAUs $\{v'(i)\}_{i=1}^{n-m} = \{v(1), v(2), ..., v(n-m)\}$ again. ApEn takes the ratio of these similarity measures at length m and $m+1$ as below:

$$\text{ApEn}(m) = \frac{1}{n-m+1} \sum_{i=1}^{n-m+1} \log C_i^m(r) - \frac{1}{N-m} \sum_{i=1}^{n-m} \log C_i^{m+1}(r) \quad (11)$$

where m is length for MAU. The MAU of different lengths possess varying levels of information complexity, which can be quantified by approximate entropy $\text{ApEn}(m)$.

Approximate entropy decreases as the length of MAU increases, which means it contains more information. But as the length of MAU increase, the race of the increasing of approximate entropy will be slower. It indicates that the data's predictability does not increase much and is insufficient to compensate for the increased collection time required for mouse dynamic sequences. In the subsequent experiments, we aimed to balance the trade-off between the sequence collection time and accuracy. Generally, we selected the length of the sequence with a slow rate of decrease in approximate entropy as the segmentation length for the mouse dynamic sequence.

VI. Local-Time Mouse Authentication (LTMouseAuthen)

In order to effectively extract deep features from mouse velocity sequences that can distinguish between different users, we propose a user authentication framework that integrates ResNet residual blocks and a GRU to fully exploit both local and global temporal information. Specifically, the input to the proposed LT-AMouse model is of fixed length. First, a 1D-CNN plus ResNet block module is employed to progressively extract and refine local features. Next, a GRU is utilized to capture contextual correlations among these features. Finally, a fully connected network performs binary classification on the extracted deep features to determine whether the input MAU belongs to the corresponding legitimate user (e.g., user j) or not.

A. ResNet Block for Local Features

Compared to commonly used multi-modal mouse data (e.g., data with timestamps, (x,y) coordinates, and interaction types), the mouse movement velocity sequence provides only single-channel velocity information. This results in a lower input dimensionality, which accelerates inference and reduces the burden of model deployment. Nevertheless, the reduced dimensionality makes it difficult for conventional manual feature engineering to adequately capture the potential temporal and local detail features. Therefore, we employ a one-dimensional convolutional neural network (1D-CNN) to extract local features from the mouse velocity sequence.

Specifically, as illustrated in Figure 1, we encode the input N-dimensional mouse velocity sequence using one-dimensional convolution while preserving the sequence length as much as possible to retain its temporal encoding. To further capture deeper local representations of the mouse data, we adopt a ResNet architecture, thereby deepening the convolutional layers and introducing residual connections. This design enables the model to refine key velocity variation patterns while preserving the complete temporal context, laying a solid foundation for subsequent classification or identity verification tasks.

B. GRU for Time Series Context Information

After completing the local convolution and residual encoding, the resulting feature maps primarily emphasize patterns of local velocity changes. To capture global trends and longer-range temporal dependencies, we use a Gated Recurrent Unit (GRU) as the temporal feature extractor. The GRU module can remember past states in the sequence and selectively retain or update crucial information through its gating mechanism, which is particularly important for modeling the temporal evolution of mouse velocity.

In the overall network architecture, the dimensionality of the features fed into the GRU remains similar with the original sequence length, owing to the ResNet design that preserves sequence length. This allows the GRU to fully utilize the complete temporal information of the original sequence, leading to more effective modeling of velocity patterns.

After extracting both local and global features of the MAU, we feed the resulting hidden states into a fully connected network (FN) for final classification or identity verification. Similar to typical binary classification tasks, this fully connected layer can employ the Softmax function to output a probability distribution, thereby determining whether the input sample belongs to the legitimate user.

C. Transferring and Training

During the training process, we formulate user identity verification as a binary classification problem and use cross-entropy loss [53] as the objective function:

$$\mathcal{L}_C = -\frac{1}{N} \sum_{c=1}^{C} y_c log \hat{y_c} \quad (12)$$

where N is the number of mouse velocity segmentation sequences, y_c is the true value for user authentication, and $\hat{y_c}$ is the probability of each velocity segmentation sample after softmax operation.

For the optimization algorithm, we select the Adam optimizer [7] to balance convergence speed and training stability. The parameter β_1 is 0.9 and β_2 is 0.999.

VII. Experiment

A. Experimental Setting

Dataset In this study, the Balabit and DFL datasets were selected as representatives of data collected from daily usage environments and laboratory environments, respectively, to investigate the trade-offs between data sufficiency, efficiency,

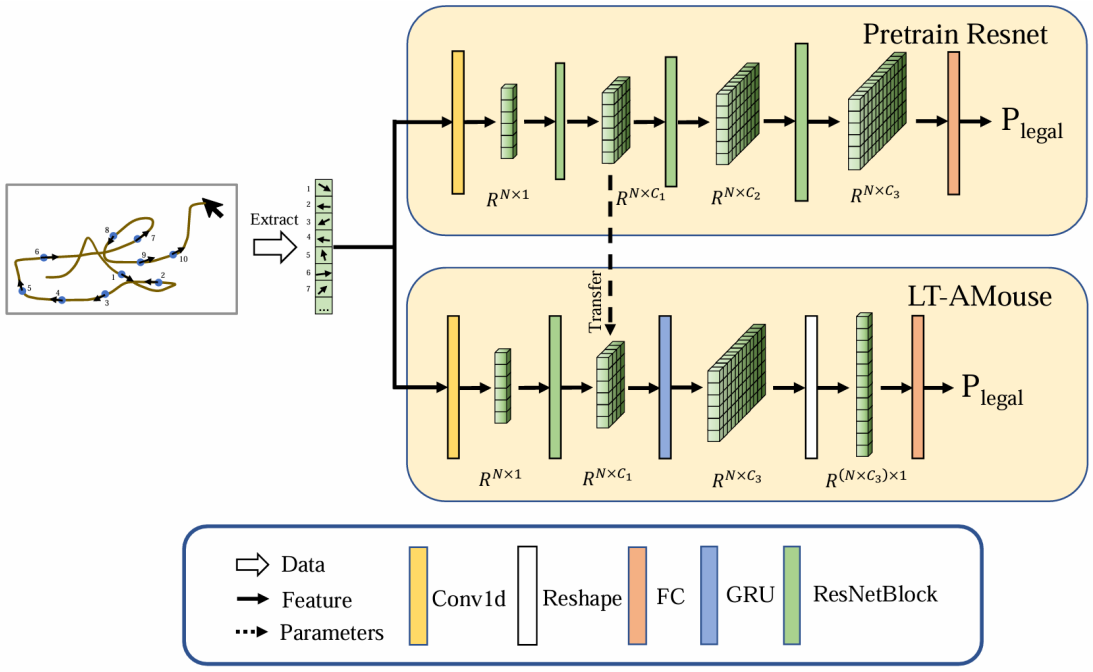


Fig. 5: User Authentication Model

and accuracy, as well as model performance. We first determine the appropriate amount of data for one individual user to be used as positive samples based on Equation 7, and then randomly select the remaining other users as negative samples to include in the training and testing sets. To verify whether the model is overfitting and to simulate a blind attack scenario, we randomly select unseen users from the remaining samples as unknown samples to be added to the testing set. Considering practical scenarios, user identification should be treated as an imbalanced classification problem, where the amount of positive sample data exceeds that of negative samples. For the DFL dataset, the ratio of positive to negative samples in the training set is 8:1, while for the Balabit dataset, the ratio is 5:1.

Metrics We employ the following evaluation metrics to assess the performance of our user authentication system based on mouse velocity sequences and its robustness against attacks:

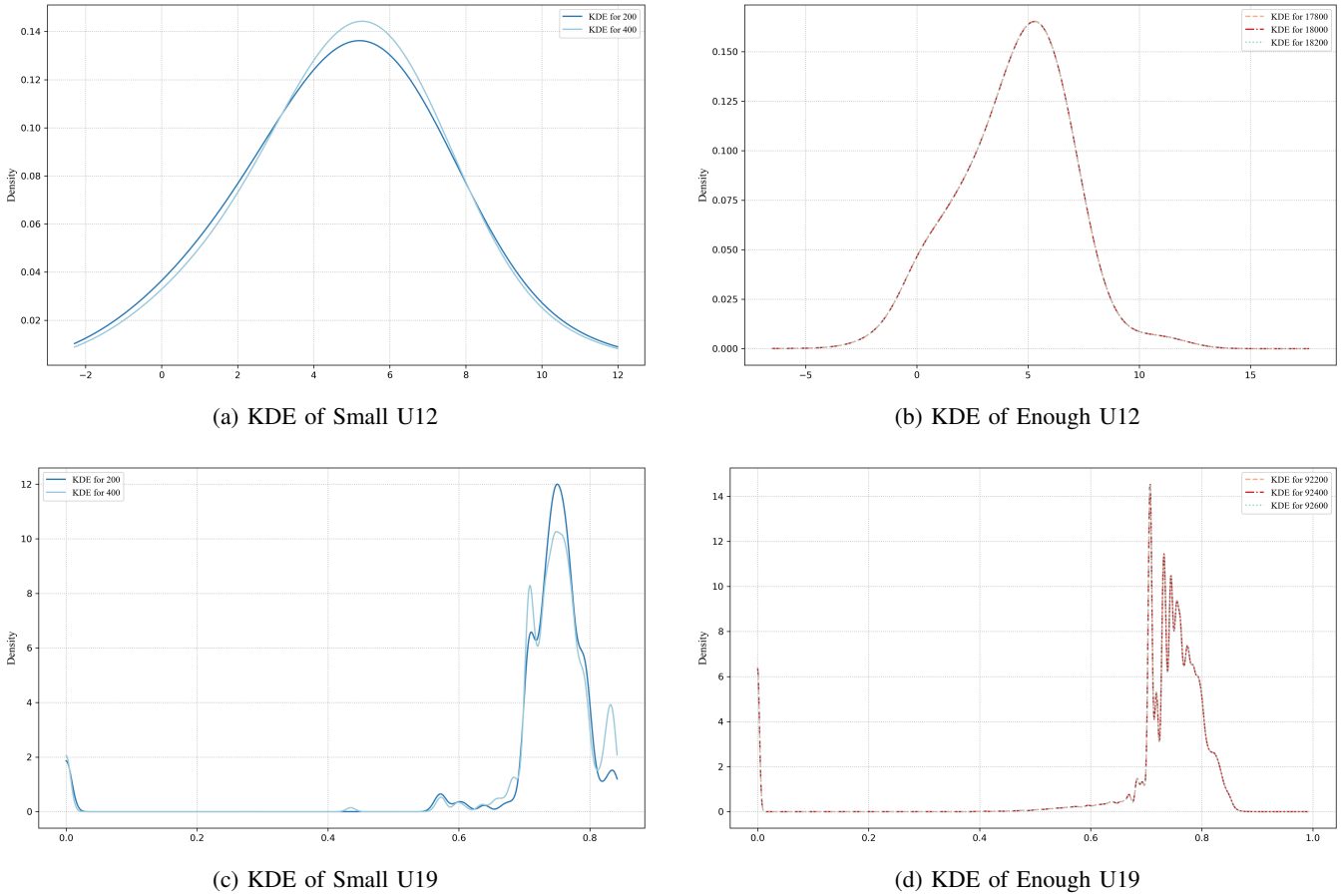
1) **F1 Score**: The harmonic mean of precision and recall, used to balance the trade-off between false positives and false negatives in imbalanced classification tasks.
2) **Area Under the Curve (AUC)**: The area under the Receiver Operating Characteristic (ROC) curve, which reflects the system's ability to differentiate between legitimate and unauthorized users across varying classification thresholds.
3) **Equal Error Rate (ERR)**: The point at which the False Acceptance Rate (FAR) and False Rejection Rate (FRR) are equal, representing a balance between security and usability in the authentication process.
4) **Defense Success Rate (DSR)**: A metric used to evaluate the effectiveness of the model in defending against adversarial attacks. It is defined as the percentage of attack attempts that fail to bypass the authentication system.

These metrics collectively ensure a comprehensive evaluation of the system's performance, particularly in the context of imbalanced classification tasks and its robustness under adversarial conditions.

B. Proper Volume of Data

In this study, mouse velocity sequences were extracted from two mouse dynamics datasets, Balabit and DFL mentioned in Section 3. Each user in the Balabit and DFL datasets is associated with multiple CSV files, each representing a distinct session unit. There is a clear discontinuity between the ending timestamp of one CSV file and the starting and ending timestamps of the subsequent file. Considering the potential variations in mouse operation habits across different time periods, which may result in differing data distributions, we first calculate the optimal data quantity for each individual CSV file and then sum these quantities to obtain the total data volume.

After calculating the mouse velocity sequences, due to the dataset being divided into multiple CSV files with clear temporal discontinuities, we computed the velocity sequences for each CSV file. To calculate the optimal data volume for a single session for each user, based on Equation4, we first generate KDEs for different data volumes within each session, using a step size of 200. As shown in Figure 6, taking User 12 in Balabit and User 19 in DFL as example, the kernel density changes significantly with increasing data when the mouse velocity data volume is relatively small. At



(a) KDE of Small U12

(b) KDE of Enough U12

(c) KDE of Small U19

(d) KDE of Enough U19

Fig. 6: Comparison of KDE of Different Volume of Mouse Velocity Data for Balabit and DFL Dataset Example

data volumes of $n = 200$ and $n = 400$, the kernel densities for two velocity sequence lengths differ markedly. However, when the data volume is large, even substantial data increases result in minimal kernel density differences. For example, when the data volume reaches sufficient value, mouse velocity data captures all variations, and the kernel density is not significantly different around it.

To better define the differences between kernel densities, we calculate the KL divergence between kernel densities represented by data of two different sample sizes using Equation6. Additionally, we use Equation7 to determine when the KL divergence converges to a sufficiently small value, indicating that the distributions of the two datasets exhibit only minimal changes. If the two thresholds in Equation7 are set to larger values, it results in smaller data volume; conversely, smaller thresholds yield larger data volume. To obtain conservative results, we set ϵ_1 to 1×10^{-4} in this study. For the Balabit dataset, considering the complexity of mouse dynamics in real-world scenarios, we adopt a more conservative threshold ϵ_2 as 1×10^{-7}; for the DFL dataset, we adopted a more aggressive strategy, setting ϵ_2 as 1×10^{-6}.

Taking User 12 from the Balabit dataset and User 9 from the DFL dataset as examples, both of which have relatively large original data volumes, we present the variation trends of the KL divergence values corresponding to Equation6 across different session data volumes. As illustrated in Figure 9, the KL divergence decreases and stabilizes as the data volume grows, suggesting diminishing distributional differences between the mouse dynamic datasets with different volume. Furthermore, adding more mouse velocity data contributes minimal additional information, indicating saturation in the data's informational content.

In the end, as shown in Figure 7 and Figure 8, the average suitable mouse dynamic dataset volume 73,778.7 of Balabit dataset is smaller than orginal one 114,114.2. For the DFL dataset, our method reduces the amount of data required for each user by a factor of ten, from 0.6726×10^7 to 0.0691×10^7.

C. Mouse Authentication Unit Length and Model Performance Trade-off

In this section, We analyzed the performance impact of different MAU (Mouse Action Unit) lengths on the authentication model LT-AMouse in Balabit and DFL datasets. The results show that as the MAU length increases, the AUC index of the model gradually increases and the EER decreases accordingly. This is because longer MAUs provide richer mouse dynamics

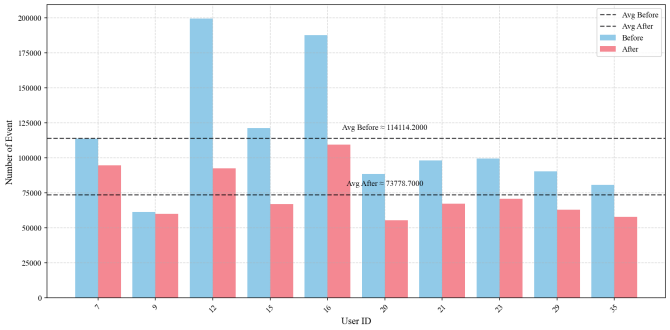


Fig. 7: Proper and Total Volume of Balabit Dataset

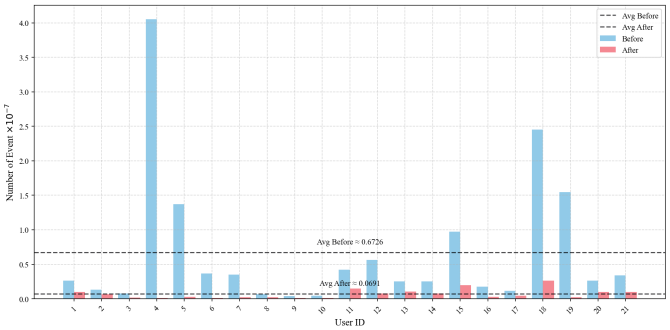


Fig. 8: Proper and Total Volume of DFL Dataset

features, which help to portray user behavior more comprehensively. However, the performance of the model does not increase linearly. In the short MAU length range, the AUC and EER improve significantly, and when the length exceeds a certain threshold, the performance improvement tends to slow down.

Through the analysis of Approximate Entropy (ApEn), we find that this trend is consistent with the pattern of sequence randomness reduction. With a short MAU length, the data is not enough to fully characterize the user; as the length increases, the uncertainty of the sequence decreases, and the model can quickly accumulate discriminative features. However, when the data volume reaches a certain scale, the additional information gain decreases, and the change amplitude of both ApEn and performance indicators tends to level off.

Therefore, the choice of MAU length needs to be a trade-off between security and real-time: high security requirement scenarios can appropriately increase the MAU length to improve accuracy, while applications with high real-time requirements need to shorten the MAU length to realize fast response. This study provides a reference basis for the practical deployment of mouse dynamics authentication system.

By further analyzing the slope of the change of entropy, we find that when the absolute value of the slope of the entropy is close to or less than 1×10^{-4}, the increase of the MAU length tends to moderate the enhancement of the model performance. At this point, the decrease in entropy is small, indicating that the accumulated feature information is close to saturation, and continuing to increase the MAU length has limited gain in model performance, while the computational cost may increase significantly. Therefore, an absolute value of the entropy slope less than 1×10^{-4} is defined as the optimal balance between efficiency and model performance.

It can be observed in Figure 10 that the optimal equilibrium point for the Balabit dataset lies between 90-130 MAU lengths, while the DFL dataset lies between 110-160. This analysis provides a theoretical basis for the selection of experimental data in Section 7.4, helping us to optimize the efficiency while ensuring the model performance.

D. Comparison with Other Models

To ensure a fair comparison with existing state-of-the-art methods, we evaluate our proposed user authentication model on the DFL and Balabit datasets under identical experimental settings. For consistency, we use the same hyper-parameter configurations, including the network architecture, loss function, and training epochs, as employed by baseline methods. This ensures that performance improvements are attributed solely to our model's design rather than differing experimental conditions.

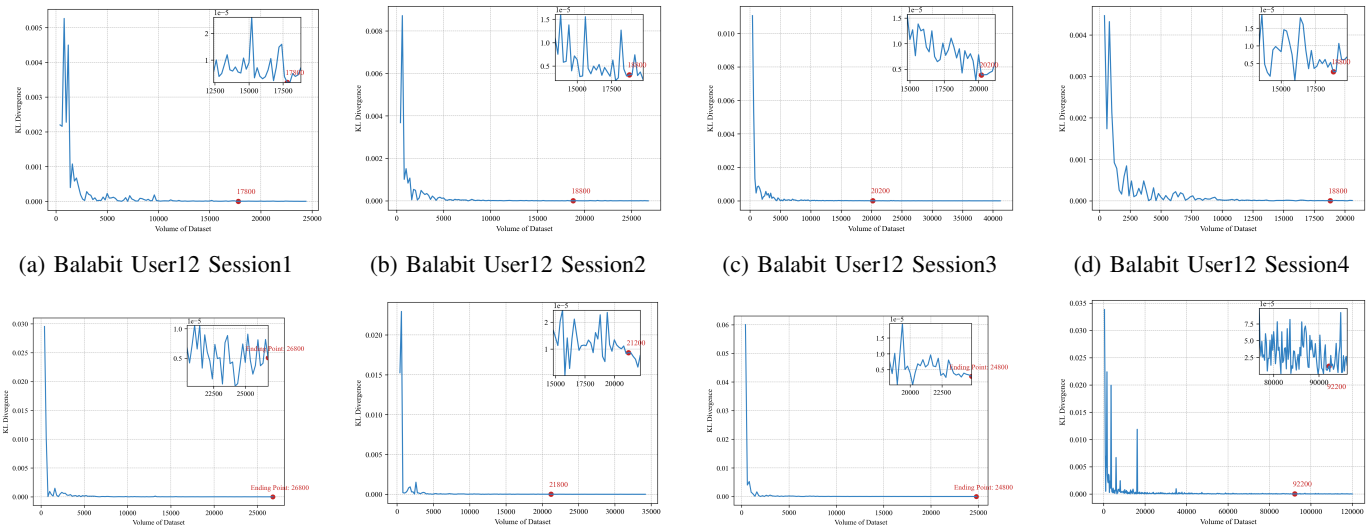
Table II and table III summarizes the performance of our model compared to other commonly used methods, including CNN, LSTM, RF, and SVM. On the DFL dataset, our model achieves the highest F1 score (97.24%) and AUC (98.52%), while maintaining a low equal error rate (EER) of 5.05%. Similarly, on the Balabit dataset, our model outperforms all baselines, achieving an F1 score of 94.65%, AUC of 97.73%, and an EER of 6.14%. These results demonstrate that our model not only provides state-of-the-art accuracy but also ensures robustness across different datasets.

Moreover, compared to traditional machine learning models such as RF and SVM, our model achieves significant improvements in both accuracy and efficiency. Specifically, our model is more than 8 times accurate than RF in terms of EER on the DFL dataset and reduces the error rate by over 40% compared to SVM on the Balabit dataset. These enhancements highlight the strength of our approach in capturing the underlying dynamics of user-specific mouse behavior.

Finally, our method exhibits superior generalizability across datasets, as evidenced by consistent improvements in both precision and recall. This capability is particularly critical for real-world applications, where datasets often vary significantly in terms of user behavior and interaction patterns.

E. Attack Model

We assume that the attacker is familiar with the authentication mechanism of LT-AMouse. Depending on whether the adversary can access the parameters of the LT-AMouse model and whether they can obtain partial mouse movement data from legitimate users (e.g., through phishing emails or other malicious means), we classify attacks into the following two types:



(a) Balabit User12 Session1 (b) Balabit User12 Session2 (c) Balabit User12 Session3 (d) Balabit User12 Session4

(e) Balabit User12 Session5 (f) Balabit User12 Session6 (g) Balabit User12 Session7 (h) DFL User9

Fig. 9: KL divergence convergence performance of User12 in the Balabit and User 9 in DFL datasets

TABLE II: User-Averaged Models Performance Comparison on DFL Dataset

Model	F1	AUC	EER
Our Model	97.24% (0.13%)	98.52% (0.01%)	5.05% (0.08%)
CNN	96.01% (0.68%)	97.07% (0.07%)	7.56% (0.18%)
LSTM	94.22% (0.07%)	83.48% (1.66%)	23.11% (1.45%)
RF	79.53% (2.56%)	89.85% (1.43%)	14.74% (1.71%)
SVM	88.92% (0.00%)	59.40% (2.77%)	42.97% (2.21%)

TABLE III: User-Averaged Models Performance Comparison on Balabit Dataset

Model	F1	AUC	EER
Our Model	94.65% (0.71%)	97.73% (0.03%)	6.14% (0.11%)
CNN	93.01% (0.28%)	93.15% (0.09%)	13.79% (0.20%)
LSTM	89.79% (0.24%)	80.18% (1.35%)	25.85% (1.21%)
RF	54.62% (0.43%)	72.18% (1.63%)	33.36% (1.21%)
SVM	86.10% (0.07%)	44.08% (2.47%)	54.04% (1.48%)

Blind Attack In a blind attack, the adversary possesses no prior knowledge of the legitimate user's mouse movement patterns. To carry out the attack, the adversary interacts with the system by controlling the mouse, attempting to bypass the authentication mechanism using their own mouse dynamics.

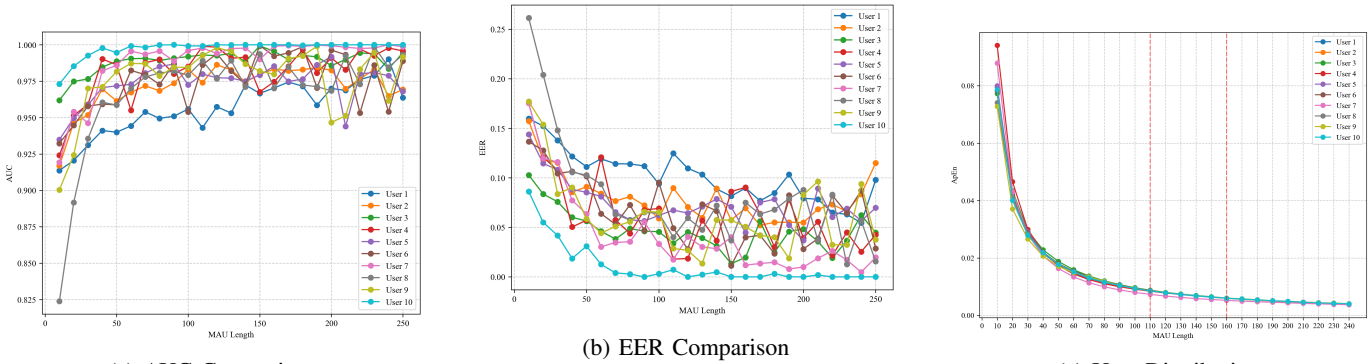
Result Blind attack results are summarized in Table II and Table III. By introducing unseen samples from the LT-AMouse dataset into the test set, which were not present in the training set, our model achieved notable performance across all evaluation metrics. Specifically, on , the model achieved an F1 Score of 97.24

Imitation Attack In an imitation attack, we assume that the adversary can observe, record, and analyze the dynamic mouse trajectories of legitimate users. Leveraging this data, the adversary employs generative models [55] to create highly realistic forged mouse trajectories, aiming to deceive the authentication system and impersonate a legitimate user.

In this study, we employ a tailored attack model designed to circumvent a mouse dynamics authentication system. This model leverages the Wasserstein Conditional Deep Convolutional Generative Adversarial Network (WCDCGAN) [54] to generate realistic and high-quality adversarial samples that closely mimic genuine mouse dynamics, making them challenging for the authentication system to distinguish from legitimate inputs.

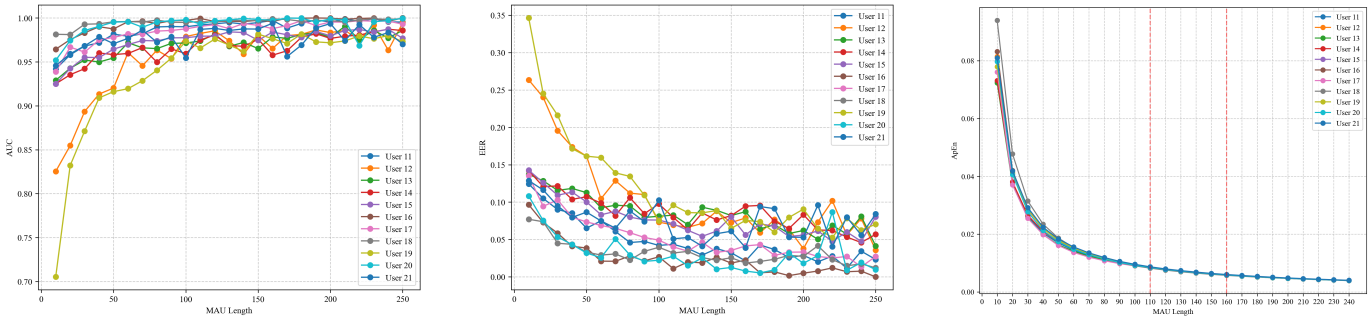
Result In the simulation attack scenario, this study employs DSR as the core evaluation metric to assess LT-AMouse's capability in distinguishing between legitimate users and generated fraudulent data under different parameter configurations. Table IV demonstrates the system's defensive performance against imitation attacks when the MAU length increases from 110 to 160.

As shown in Table IV, for the DFL dataset, when the MAU length is 120, the system achieves a defense success rate of up to 99.02%. At other lengths the defense success rate remains between 88.56% and 95.87%. For the Balabit dataset, our model achieves the best defense success rate of 79.69% at MAU length 110, which demonstrates a significant improvement compared to the original attack success rate of



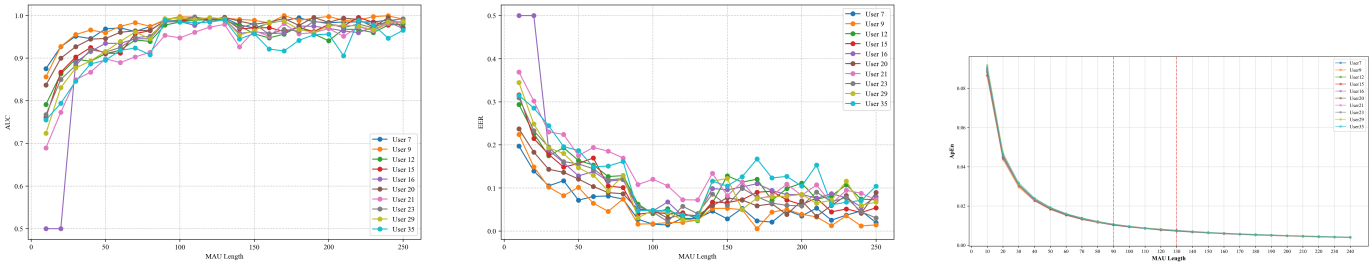
(a) AUC Comparison (b) EER Comparison (c) User Distribution

DFL Dataset User 1-10



(d) AUC Comparison (e) EER Comparison (f) User Distribution

DFL Dataset User 11-21



(g) AUC Comparison (h) EER Comparison (i) User Distribution

Balabit Dataset

Fig. 10: Performance Metrics and ApEn Analysis on DFL and Balabit Dataset

94% reported in the WC-DCGAN attack [54] . These findings suggest that the proposed LT-AMouse model can accurately differentiate genuine from forged mouse trajectories, even with high-fidelity imitation data, thereby demonstrating strong robustness and security.

VIII. Discussion

Environmental interference and minimal user input In practical applications, mouse dynamics-based authentication relies on the consistent capture of movement trajectories and click patterns. However, significant fluctuations in surface friction or hardware sensitivity caused by environmental factors, as well as minimal and low-amplitude mouse operations, pose challenges to accurate modeling of these sparse and weak dynamic signals, leading to performance degradation. Compared to continuous and pronounced mouse inputs, unstable or low-amplitude operations are more susceptible to environmental factors and usage posture, increasing the likelihood of errors in authentication.

Biometric Authentication Mouse operation habits are susceptible to variations in hardware, environment, and users' states (e.g. fatigue and emotions), leading to significant shifts in data distribution over time and weak generalization. Ad-

TABLE IV: Defense Success Rates for Different MAU Lengths

Dataset	MAU Length	Defense Success Rate (%)
DFL	110	89.22
	120	99.02
	130	88.56
	140	95.87
	150	94.44
	160	93.68
Balabit	90	47.51
	100	70.00
	110	79.69
	120	59.90
	130	60.00

ditionally, in cross-platform or cross-context deployments, mouse movement features can vary substantially due to differences in operating systems or the mouse's settings, posing challenges to model transferability. Furthermore, while mouse-dynamics-based approaches offer advantages in data collection and privacy protection compared to biometic methods, they remain vulnerable to some threats. For example, attackers could potentially infer user behavior patterns and compromise system security by intercepting portions of mouse trajectories through malicious software or remote monitoring. Lastly, to continuously enhance security and robustness, further research is needed in privacy-protection technologies, such as on-device computation, federated learning, and secure multiparty computation, along with in-depth model adaptation and optimization tailored to diverse application scenarios.

Attack Model In this study, we assessed our method's resilience against the advanced WCDCGAN [54] attack, which has demonstrated up to a 94% success rate in compromising mouse dynamics authentication systems. Notably, our approach can operate effectively with smaller datasets, thus enabling faster and more efficient defense performance—albeit without achieving complete immunity. Future research will focus on integrating advanced strategies and novel techniques to further fortify the system's resistance. Empirical results show that our method achieved defense success rates exceeding 88% on the DFL dataset, whereas the Balabit dataset peaked at 79.69%, suggesting that dataset properties significantly influence how well the system can withstand sophisticated adversarial threats.

IX. Conclusion

This study presents a robust and efficient mouse dynamics-based authentication framework, addressing challenges in data sufficiency, practicality, and security. By introducing the Mouse Authentication Unit (MAU) and leveraging Approximate Entropy, our method optimizes data segmentation for accurate behavioral representation. The Local-Time Mouse Authentication (LT-MAuthen) framework achieved state-of-the-art performance, with AUCs of 98.52% on the DFL dataset and 94.65% on the Balabit dataset, while demonstrating resilience against advanced adversarial attacks. These findings highlight the framework's potential for real-world applications, with future work focused on enhancing cross-platform adaptability and robust defense mechanisms.

References

References

[1] Z. Erlich and M. Zviran, "Authentication methods for computer systems security," in *Encyclopedia of Information Science and Technology, Second Edition*. IGI Global, 2009, pp. 288–293.

[2] R. W. Proctor, M.-C. Lien, K.-P. L. Vu, E. E. Schultz, and G. Salvendy, "Improving computer security for authentication of users: Influence of proactive password restrictions," *Behavior Research Methods, Instruments, & Computers*, vol. 34, pp. 163–169, 2002.

[3] K. Revett, H. Jahankhani, S. T. D. Magalhaes, and H. M. D. Santos, "A survey of user authentication based on mouse dynamics," in *Global E-Security: 4th International Conference, ICGeS 2008, London, UK, June 23-25, 2008. Proceedings*. Springer, 2008, pp. 210–219.

[4] D. Haussler, "A general minimax result for relative entropy," *IEEE Transactions on Information Theory*, vol. 43, no. 4, pp. 1276–1280, 1997.

[5] C. Villani, "A short proof of the "concavity of entropy power"," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1695–1696, 2000.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[7] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[8] K. Wang, C. Ma, Y. Qiao, X. Lu, W. Hao, and S. Dong, "A hybrid deep learning model with 1dcnn-lstm-attention networks for short-term traffic flow prediction," *Physica A: Statistical Mechanics and its Applications*, vol. 583, p. 126293, 2021.

[9] S. Fu, D. Qin, D. Qiao, and G. T. Amariucai, "Rumba-mouse: Rapid user mouse-behavior authentication using a cnn-rnn approach," in *2020 IEEE Conference on Communications and Network Security (CNS)*. IEEE, 2020, pp. 1–9.

[10] P. Chong, Y. Elovici, and A. Binder, "User authentication based on mouse dynamics using deep neural networks: A comprehensive study," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1086–1101, 2019.

[11] Q. Yao, J. Zhao, Z. Yang, R. Fei, L. Yan, and Y. Wang, "Identity authentication based on user mouse behavior," in *2020 International Conference on Virtual Reality and Intelligent Systems (ICVRIS)*. IEEE, 2020, pp. 571–577.

[12] M. Antal, N. Fejér, and K. Buza, "Sapimouse: Mouse dynamics-based user authentication using deep feature learning," in *2021 IEEE 15th International Symposium on Applied Computational Intelligence and Informatics (SACI)*. IEEE, 2021, pp. 61–66.

[13] D. Polemi, "Biometric techniques: Review and evaluation of biometric techniques for identification and authentication, including an appraisal of the areas where they are most applicable," *Reported Prepared for the European Commission DG XIIIC*, vol. 4, 1997.

[14] R. Joyce and G. Gupta, "Identity authentication based on keystroke latencies," *Communications of the ACM*, vol. 33, no. 2, pp. 168–176, 1990.

[15] K. O. Bailey, J. S. Okolica, and G. L. Peterson, "User identification and authentication using multi-modal behavioral biometrics," *Computers & Security*, vol. 43, pp. 77–89, 2014.

[16] Y. Meng, D. S. Wong, R. Schlegel, and L. for Kwok, "Touch gestures based biometric authentication scheme for touchscreen mobile phones," in *Information Security and Cryptology: 8th International Conference, Inscrypt 2012, Beijing, China, November 28-30, 2012, Revised Selected Papers 8*. Springer, 2013, pp. 331–350.

[17] A. Buriro, B. Crispo, F. D. Frari, and K. Wrona, "Touchstroke: Smartphone user authentication based on touch-typing biometrics," in *New Trends in Image Analysis and Processing–ICIAP 2015 Workshops: ICIAP 2015 International Workshops, BioFor, CTMR, RHEUMA, ISCA, MADiMa, SBMI, and QoEM, Genoa, Italy, September 7-8, 2015, Proceedings 18*. Springer, 2015, pp. 27–34.

[18] A. M. Amarasinghe, I. Malassri, K. Weerasinghe, I. Jayasingha, P. K. Abeygunawardhana, and S. Silva, "Stress analysis and care prediction system for online workers," in *2021 3rd International Conference on Advancements in Computing (ICAC)*. IEEE, 2021, pp. 329–334.

[19] Q. Yi, W. Li, S. ping Yi, and J. dong Xie, "Trustworthy identity authentication based on joint time-frequency analysis of mouse behavior," *Journal of Beijing University of Posts and Telecommunications*, vol. 44, no. 4, p. 121.

[20] X. Ding, C. Peng, H. Ding, M. Wang, H. Yang, and Q. Yu, "User identity authentication and identification based on multi-factor behavior features," in *2019 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2019, pp. 1–6.

[21] S. J. Quraishi and S. Bedi, "Secure system of continuous user authentication using mouse dynamics," in *2022 3rd International Conference on Intelligent Engineering and Management (ICIEM)*. IEEE, 2022, pp. 138–144.

[22] N. Siddiqui, R. Dave, and N. Seliya, "Continuous user authentication using mouse dynamics, machine learning, and minecraft," in *2021 International Conference on Electrical, Computer and Energy Technologies (ICECET)*. IEEE, 2021, pp. 1–6.

[23] C. Shen, Z. Cai, X. Guan, C. Fang, and Y. Du, "User authentication and monitoring based on mouse behavioural features," *Journal on Communications*, vol. 31, no. 7, pp. 68–75, 2010.

[24] A. A. E. Ahmed and I. Traore, "A new biometric technology based on mouse dynamics," *IEEE Transactions on Dependable and Secure Computing*, vol. 4, no. 3, pp. 165–179, 2007.

[25] H. Jun and M. Kang, "Three-way identity authentication method based on mouse behavior," *Journal of Nanjing University of Science and Technology*, no. 4, pp. 474–480, 2019.

[26] M. Antal and N. Fejér, "Mouse dynamics based user recognition using deep learning," *Acta Universitatis Sapientiae, Informatica*, vol. 12, no. 1, pp. 39–50, 2020.

[27] O. V. Komogortsev, A. Karpov, and C. D. Holland, "Attack of mechanical replicas: Liveness detection with eye movements," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 716–725, 2015.

[28] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," *Advances in Neural Information Processing Systems*, vol. 29, 2016.

[29] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf, "Pruning filters for efficient convnets," *arXiv Preprint arXiv:1608.08710*, 2016.

[30] Y. Huang, C. Du, Z. Xue, X. Chen, H. Zhao, and L. Huang, "What makes multi-modal learning better than single (provably)," *Advances in Neural Information Processing Systems*, vol. 34, pp. 10 944–10 956, 2021.

[31] S. Pincus, "Approximate entropy (apen) as a complexity measure," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 5, no. 1, pp. 110–117, 1995.

[32] E. Chiu, J. Lin, B. McFerron, N. Petigara, and S. Seshasai, "Mathematical theory of claude shannon. a study of the style and context of his work up to the genesis of information theory," *Submitted for The Structure of Engineering Revolutions (MIT course 6.933 J/STS. 420J), nd*, 2018.

[33] P. Paysarvi-Hoseini and N. C. Beaulieu, "Optimal wideband spectrum sensing framework for cognitive radio systems," *IEEE Transactions on Signal Processing*, vol. 59, no. 3, pp. 1170–1182, 2010.

[34] E. Fredkin, "An introduction to digital philosophy," *International Journal of Theoretical Physics*, vol. 42, pp. 189–247, 2003.

[35] M. Boedihardjo, T. Strohmer, and R. Vershynin, "Privacy of synthetic data: A statistical framework," *IEEE Transactions on Information Theory*, vol. 69, no. 1, pp. 520–527, 2022.

[36] J. A. O'Sullivan, R. E. Blahut, and D. L. Snyder, "Information-theoretic image formation," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2094–2123, 1998.

[37] A. Bradlev, "The use of the area under the roc curve in the evaluation of machine learning algorithms," *Pattern Recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.

[38] L. Araujo, L. Sucupira, M. Lizarraga, L. Ling, and J. Yabu-Uti, "User authentication through typing biometrics features," *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 851–855, 2005.

[39] K. W. Fülöp, Á., Kovács, L., "Balabit mouse dynamics challenge data set," 2016, https://github.com/balabit/Mouse-Dynamics-Challenge/, Last accessed on 03-01-2025.

[40] M. Antal and L. Denes-Fazakas, "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, 2019, pp. 143–148.

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

[42] B. W. Silverman, *Density estimation for statistics and data analysis*. Routledge, 2018.

[43] R. E. Heyman, B. R. Chaudhry, D. Treboux, J. Crowell, C. Lord, D. Vivian, and E. B. Waters, "How much observational data is enough? an empirical test using marital interaction coding," *Behavior Therapy*, vol. 32, no. 1, pp. 107–122, 2001. [Online]. Available: https://doi.org/10.1016/S0005-7894(01)80047-2

[44] A. H. Wortley, P. J. Rudall, D. J. Harris, and R. W. Scotland, "How much data are needed to resolve a difficult phylogeny?: case study in lamiales," *Systematic biology*, vol. 54, no. 5, pp. 697–709, October 2005. [Online]. Available: https://academic.oup.com/sysbio/article-pdf/54/5/697/26543993/10635150500221028.pdf

[45] K. D. Splinter, I. L. Turner, and M. A. Davidson, "How much data is enough? the importance of morphological sampling interval and duration for calibration of empirical shoreline models," *Coastal Engineering*, vol. 77, pp. 14–27, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378383913000495

[46] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, vol. 2, 1999, pp. 246–252 Vol. 2.

[47] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*. Chapman and Hall/CRC, 1986.

[48] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[49] C. Shen, Z. Cai, and X. Guan, "Continuous authentication for mouse dynamics: A pattern-growth approach," in *IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2012)*, 2012, pp. 1–12.

[50] J. Xu, M. Li, F. Zhou, and R. Xue, "Identity authentication method based on user's mouse behavior," *Computer Science*, vol. 43, no. 2, pp. 148–154, 2016.

[51] S. Bleha, C. Slivinsky, and B. Hussien, "Computer-access security systems using keystroke dynamics," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 12, pp. 1217–1222, 1990.

[52] M. S. Obaidat and B. Sadoun, "Verification of computer users using keystroke dynamics," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 27, no. 2, pp. 261–269, 1997.

[53] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, June 2017. [Online]. Available: https://doi.org/10.1145/3065386

[54] A. Roy, K. Wong, and R. C.-W. Phan, "Attacking mouse dynamics authentication using novel wasserstein conditional dcgan," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 3622–3631, 2023.

[55] Y. X. Marcus Tan, A. Iacovazzi, I. Homoliak, Y. Elovici, and A. Binder, "Adversarial attacks on remote user authentication using behavioural mouse dynamics," in *2019 International Joint Conference on Neural Networks (IJCNN)*, 2019, pp. 1–10.

[56] Y. X. M. Tan, A. Iacovazzi, I. Homoliak, Y. Elovici, and A. Binder, "Adversarial attacks on remote user authentication using behavioural mouse dynamics," 2019.

[57] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proceedings of the 34th International Conference on Machine Learning (ICML 2017)*, ser. ICML'17. JMLR.org, 2017, pp. 214–223.

[58] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014.

[59] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based fingerprint matching," *IEEE Transactions on Image Processing*, vol. 9, no. 5, pp. 846–859, 2000.

[60] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: From features to supervectors," *Speech Communication*, vol. 52, no. 1, pp. 12–40, 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167639309001289

[61] A. Gupta, A. Khanna, A. Jagetia, D. Sharma, S. Alekh, and V. Choudhary, "Combining keystroke dynamics and face recognition for user verification," in *Proceedings of the 2015 IEEE 18th International Conference on Computational Science and Engineering (CSE)*, 2015, pp. 294–299.