A MOBILE TOUCH-BASED CONTINUOUS AUTHENTICATION SYSTEM VIA USER-SPECIFIC DISTRIBUTION BASED LEARNING

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS OF
THE MIDDLE EAST TECHNICAL UNIVERSITY

BY

GÜRKAN GÜNDÜZ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE
OF
MASTER OF SCIENCE
IN
THE DEPARTMENT OF DATA INFORMATICS

AUGUST 2025

A Mobile Touch-Based Continuous Authentication System via User-Specific Distribution Based Learning

submitted by **GÜRKAN GÜNDÜZ** in partial fulfillment of the requirements for the degree of **Master of Science in Data Informatics Department, Middle East Technical University** by,

Date: 21.08.2025

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Gürkan GÜNDÜZ

Signature :

ABSTRACT

A MOBILE TOUCH-BASED CONTINUOUS AUTHENTICATION SYSTEM VIA USER-SPECIFIC DISTRIBUTION BASED LEARNING

GÜNDÜZ, Gürkan
M.S., Department of Data Informatics
Supervisor: Prof. Dr. Tuğba Taşkaya Temizel

August 2025, 50 pages

Mobile devices are exposed to security threats, even when protected by traditional authentication methods such as passwords, PINs, or physiological biometrics. These methods are vulnerable to attacks like shoulder surfing and spoofing, and they lack mechanisms for continuous identity verification after initial access. These issues are mitigated by Continuous Authentication (CA), which continuously validates identity through behavioral biometrics, such as scrolling and swiping patterns. In this work, a touch-based CA framework is introduced that models full probability distributions of interaction features (movement speed, acceleration, curvature, cumulative average speed) instead of relying on summary statistics. Kernel Density Estimation (KDE) is employed to generate feature distributions, and session- and user-specific bandwidths are optimized using Kullback–Leibler (KL) divergence, enhancing the fidelity of individual behavior models. These distribution-based features are then fed into a CNN-based Siamese network, where similarity embeddings across sessions are learned for authentication. When evaluated on the BehavePassDB dataset, the proposed method

is consistently shown to outperform both a fixed-bandwidth alternative and a baseline XGBoost model using handcrafted statistics, achieving higher ROC AUC, lower Equal Error Rates, and improved precision–recall metrics. Authentication accuracy, robustness, and generalization across devices and populations are demonstrated to be significantly boosted by the modeling of complete behavioral distributions and the tailoring of KDE parameters per user.

Keywords: Continuous Authentication (CA), Behavioral Biometrics, Touch-Based Authentication, Kernel Density Estimation (KDE), Kullback-Leibler Divergence

ÖZ

KULLANICIYA ÖZEL DAĞILIM TABANLI ÖĞRENME İLE DOKUNMATİK SÜREKLİ KİMLİK DOĞRULAMA SİSTEMİ

GÜNDÜZ, Gürkan
Yüksek Lisans, Veri Bilişimi Bölümü
Tez Yöneticisi: Prof. Dr. Tuğba Taşkaya Temizel

Ağustos 2025, 50 sayfa

Mobil cihazlar, şifre, PIN veya fizyolojik biyometri gibi geleneksel kimlik doğrulama yöntemleriyle korunsa bile güvenlik tehditlerine açıktır. Bu yöntemler, omuz üzerinden gözetleme ve sahtecilik gibi saldırılara karşı savunmasızdır ve ilk erişimden sonra sürekli kimlik doğrulama mekanizmalarından yoksundur. Bu sorunlar, kaydırma ve dokunma davranışları gibi davranışsal biyometri ile kimliği sürekli doğrulayan Sürekli Kimlik Doğrulama (CA) ile azaltılır. Bu çalışmada, özet istatistikler yerine etkileşim özelliklerinin (hareket hızı, ivme, eğrilik, kümülatif ortalama hız) tam olasılık dağılımlarını modelleyen dokunma tabanlı bir CA çerçevesi sunulmuştur. Çekirdek Yoğunluk Tahmini (KDE) ile özellik dağılımları oluşturulur ve Kullback-Leibler (KL) ıraksaması kullanılarak oturum ve kullanıcıya özel bant genişlikleri optimize edilir, böylece bireysel davranış modellerinin doğruluğu artırılır. Bu dağılım tabanlı özellikler, bir CNN tabanlı Siamese ağına beslenir ve kimlik doğrulama için oturumlar arası benzerlik gösterimleri öğrenilir. BehavePassDB veri seti üzerinde değerlendirildiğinde, önerilen yöntemin sabit bant genişlikli bir alternatifi ve elle hazırlanmış

istatistikler kullanan bir XGBoost temel modelini sürekli olarak geride bıraktığı gösterilmiştir. Daha yüksek ROC AUC, daha düşük Eşit Hata Oranları ve gelişmiş kesinlik–duyarlılık metrikleri elde edilmiştir. Tam davranışsal dağılımların modellenmesi ve KDE parametrelerinin kullanıcı bazında uyarlanması sayesinde, kimlik doğrulama doğruluğu, dayanıklılığı ve cihazlar ile popülasyonlar arasında genelleme yeteneği önemli ölçüde artırılmıştır.

Anahtar Kelimeler: Sürekli Kimlik Doğrulama (CA), Davranışsal Biyometri, Dokunma Tabanlı Kimlik Doğrulama, Çekirdek Yoğunluk Tahmini (KDE), Kullback-Leibler Iraksaması

To those who never give up

ACKNOWLEDGMENTS

I would like to thank my thesis advisor, Prof. Dr. Tuğba Taşkaya Temizel, for her guidance, time, and support during this study. Her advice was essential for the progress of both this thesis and my academic development. I am grateful to have worked under her supervision.

I also thank Middle East Technical University and the Informatics Institute for providing a supportive academic environment that helped me grow.

I am especially thankful to Muratcan Kaplan for our collaboration on related topics. His feedback and shared efforts were valuable throughout this process.

I am sincerely grateful to my mother, father, and brother for their continued support and encouragement.

Finally, I would like to thank my wife, Deniz Berfin Gündüz, for her constant support and motivation throughout this journey. Her presence has always been a source of strength.

This study was supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under the 2210-National Graduate Scholarship Program by the Presidency of Supporting Scientists (BİDEB). I gratefully acknowledge their financial support.

TABLE OF CONTENTS

ABSTRACT iv

ÖZ vi

DEDICATION viii

ACKNOWLEDGMENTS ix

TABLE OF CONTENTS x

LIST OF TABLES xiii

LIST OF FIGURES xiv

LIST OF ABBREVIATIONS xv

CHAPTERS

1 INTRODUCTION 1

1.0.1 Research Questions 2

1.0.2 Contributions of the Study 3

1.0.3 Organization of the Thesis 4

2 RELATED WORK 5

2.0.1 Literature on the Behavioral Biometrics 5

2.0.2 Touchalytics Dataset 7

2.0.3 Posture and Body Movement Dataset 9

2.0.4 HuMIdb Dataset 10

3 METHODOLOGY 13

3.1 Parameter Optimization and Data Preprocessing 14

3.2 Data Filtering 17

3.3 Bandwidth Selection and Distribution Generation 17

3.3.1 User-Specific Bandwidth Optimization 17

3.3.2 Session-Feature Bandwidth Optimization 19

3.3.3 Distribution Generation for Model Input 19

3.4 Model Architecture and Training 20

3.5 Model Evaluation with Unseen Data 22

4 EXPERIMENTS & RESULTS 23

4.1 Dataset 23

4.1.1 Exploratory Data Analysis and Preprocessing 24

4.2 Model Comparison 25

4.2.1 Dataset Preparation for Model Training and Evaluation 25

4.2.1.1 Training Dataset 25

4.2.1.2 Validation Dataset 26

4.2.1.3 Test Dataset 26

4.3 Experimental Setup 26

4.4 Results 27

4.5 Ablation Study 31

5 CONCLUSION AND DISCUSSION 33

5.1 Conclusion 33

5.2 Discussion 34

5.3 Limitations 36

5.4 Future Work 37

REFERENCES 39

APPENDICES

A Data Quality Findings 45

B Preprocessing and Model Architecture 49

LIST OF TABLES

Table 1 Optimized Feature-Specific Parameters 27

Table 2 Comparison of Baseline and Optimized Models on Test Set Metrics. ORT: Optimized ROC Threshold, OET: Optimized EER Threshold 28

Table 3 Comparison of Optimized (O) and Non-Optimized (NO) Model Performances 28

Table 4 Comparison of Optimized (O) and Non-Optimized (NO) Model Performances against the Unprocessed data with Filtering(F) and Unfiltering(UF) Users 31

LIST OF FIGURES

Figure 1 Example Distance matrix for comparing two session users on Movement Speed feature 16

Figure 2 Bandwidth selection process for distribution generation 21

Figure 3 DET Curve Comparison of Non-Optimized and Optimized Models. 29

Figure 4 Roc Curves for both Test and Validation Data 30

Figure 5 Acceleration feature distributions for User 31 with raw and processed data. 30

Figure A.1 Illustration of timestamp irregularities in sensor data. Linear interpolation was applied to mitigate these issues. 45

Figure A.2 Touch data plot for the user 31 in session g1 46

Figure A.3 Touch data plot for the user 31 in session g3 46

Figure A.4 Sensor Data Lengths for the users 47

Figure A.5 Sensor Data collection rates for the users 48

Figure B.1 Evolution of the session data after for the user 4 in session g2 for the movement speed feature 49

Figure B.2 Overall approach for the user comparison stage. 50

LIST OF ABBREVIATIONS

CA	Continuous Authentication
PIN	Personal Identification Number
ID	Identification
BehavePassDB	BehavePass Database
HuMIdb	Human Mobile Interaction Database
Touchalytics	Touch-Based Authentication Dataset
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
SVM	Support Vector Machine
RF	Random Forest
KNN	K-Nearest Neighbors
XGBoost	Extreme Gradient Boosting
KDE	Kernel Density Estimation
KL Divergence	Kullback–Leibler Divergence
JS Divergence	Jensen–Shannon Divergence
MCC	Matthews Correlation Coefficient
EER	Equal Error Rate
ROC	Receiver Operating Characteristic
AUC	Area Under the Curve
DET	Detection Error Tradeoff Curve

TPR	True Positive Rate
FPR	False Positive Rate
FNR	False Negative Rate
ORT	Optimized ROC Threshold
OET	Optimized EER Threshold
UF	Unfiltered
F	Filtered
O	Optimized
NO	Non-Optimized
PDF	Probability Density Function
AUC_max	Maximum Area Under the Curve

CHAPTER 1

INTRODUCTION

Mobile devices have become essential tools in daily life, used not only for communication and entertainment but also for storing sensitive personal and corporate information. However, traditional mobile authentication systems, such as passwords, PIN codes, graphical patterns, and physiological biometrics (e.g. fingerprint or facial recognition), face significant security and usability challenges. These approaches are vulnerable to attacks, including shoulder surfing, smudge analysis, and spoofing. In addition, they typically function as one-time entry-point authentication methods, offering no continuous protection against unauthorized access during device use. To address these shortcomings, behavioral biometrics have been investigated for continuous authentication. Publicly available datasets, such as BehavePassDB and MobileB2C, have been used in recent studies to evaluate the performance of these techniques. Additionally, advanced methods, such as CNN-based Siamese networks [1], have demonstrated the potential to enhance reliability and scalability in addressing these challenges.

To address these limitations, Continuous Authentication (CA) has been proposed as an alternative to entry-point authentication methods [2]. In CA systems, behavioral biometrics are used to analyze consistent patterns in user interactions with mobile devices, including typing, scrolling, and swiping. These methods operate passively, without requiring active user input. The continuous nature of authentication is supported by multimodal data collected from embedded sensors, such as accelerometers, gyroscopes, and touchscreens, which enable real-time verification of user identity.

Behavioral biometrics offer stronger security compared to physiological biometrics, as spoofing them often requires advanced technical skills. They have been extensively studied in the literature, with research focusing on unique patterns derived from user interactions, such as touch dynamics, keystroke patterns, and motion behavior. These studies have explored various modeling techniques, including traditional machine learning, deep learning architectures like CNNs, and feature distribution analysis, to improve the reliability and accuracy of authentication systems. Despite significant advancements, challenges remain in addressing variability in user behavior, environmental influences, and scalability across diverse scenarios, highlighting the need for continued innovation in this field.

1.0.1 Research Questions

To advance the understanding of behavioral biometrics, this study focuses on two main research questions:

1. **What is the impact of distribution-based methods for feature representation on the performance and robustness of biometric authentication models?** Unlike traditional approaches that summarize feature distributions with metrics such as the mean and standard deviation, this study explores how modeling the entire distribution can better represent user behavior, reduce information loss, and improve model accuracy.

2. **How can feature representation and optimization strategies effectively address inconsistencies and noise in biometric data to improve model performance?** This question aims to investigate methods for mitigating the negative effects of real-world data noise, which often challenges the reliability and accuracy of biometric authentication models.

1.0.2 Contributions of the Study

A new framework for mobile authentication is presented in this work. Feature distribution learning and deep learning techniques are used to model touch-based behavioral biometrics. Unlike traditional methods based on statistical features, the underlying distributions of touch interactions are analyzed. Features like acceleration, movement speed, curvature, and cumulative average speed are extracted. Rather than treating them as independent variables, their probability distributions are modeled, and a Convolutional Neural Network (CNN) is employed to learn representations from these distributions.

The differences between user-specific feature distributions are measured using Kullback-Leibler (KL) divergence, which compares two probability distributions. This makes it useful for capturing individual behavioral patterns. The distribution embeddings are then passed through a Siamese network, which is trained to distinguish between input pairs for verification. This helps the model to generalize even with limited training data.

Existing biometric authentication methods focus mainly on extracting features from raw sensor data [3]. On the other hand, some studies apply statistical measures such as the mean, standard deviation, and percentiles on the touch data [4, 5]. Although these methods simplify data representation, they may discard useful information by reducing behavioral patterns to fixed summaries.

In contrast, the proposed method explicitly models feature distributions, capturing richer variability in user behavior. This approach avoids oversimplification and aligns better with deep learning techniques, which benefit from probabilistic representations. By focusing on distributions rather than aggregated statistics, the method improves discriminative power in verification tasks.

The approach is validated using the BehavePassDB mobile behavioral biometrics dataset [6], demonstrating the effectiveness of touch-based biometrics in achieving low Equal Error Rates (EERs). By combining the strengths of feature distribution

analysis with CNN-based Siamese networks, this study contributes to advancing mobile authentication systems toward more secure, user-friendly, and privacy-preserving solutions.

1.0.3 Organization of the Thesis

This thesis is organized as follows: Chapter 2 reviews related work and outlines the research gap addressed in this study. Chapter 3 describes the proposed architecture and the methods used. Chapter 4 presents the dataset, experimental setup, and results, including analysis and ablation studies. Chapter 5 concludes the thesis by summarizing the work, addressing limitations, and suggesting directions for future research.

CHAPTER 2

RELATED WORK

This chapter reviews prior studies and publicly available datasets relevant to continuous authentication (CA) through behavioral biometrics. The literature review first examines research on behavioral biometrics, covering approaches based on gesture and touch dynamics, multi-sensor fusion, and deep learning architectures for feature extraction. Following the methodological review, three datasets used in behavioral biometric researches are presented, highlighting their data collection protocols, modalities, and relevance to CA evaluation.

2.0.1 Literature on the Behavioral Biometrics

Behavioral biometrics involves the analysis of distinctive interaction patterns between users and their devices, including touch dynamics, motion patterns, and gesture-based behaviors. These traits have been widely studied for continuous authentication (CA) on smartphones, where identity is verified implicitly during device usage.

Gesture and touch-based authentication have often relied on constrained interaction patterns. In [7], horizontal slider movements were analyzed with device motion and swipe features using SVM and Random Forest classifiers, where fusion of thumb swipes improved accuracy, but fixed directions reduced behavioral variability. Similar reliance on predefined swipes was reported in [8], limiting applicability to free-form usage. Raw touchscreen interactions were modeled in [9]. One-class SVM and isolation forest models were applied in [10] to four controlled touch operation types, again restricting coverage. Direction-based separation was used in [11] with

Extra-Trees and Gradient Boosting classifiers, achieving improved results but constraining user input to predefined orientations. Touch dynamics were also explored in [12] through a handcrafted set of 21 behavioral features classified with neural networks, optimized using particle swarm optimization. Raw touch data were modeled through deep recurrent and convolutional networks in [13], and template protection was addressed in [14] using a fuzzy commitment scheme with CNN–RNN feature extraction. While these methods demonstrate effective classification, most depend on engineered features or constrained gestures, potentially obscuring fine-grained behavioral variation. The present work differs by modeling unconstrained touch patterns as complete probability distributions, preserving the full interaction profile.

Multi-sensor fusion approaches combine touch data with motion or other sensing modalities to increase robustness. Accelerometer, gyroscope, and magnetometer readings have been combined with touch events in [15] using Kalman filtering and Siamese networks, and in [16] through a two-tower transformer capturing temporal and cross-channel dependencies. Keystroke and swipe dynamics were fused in [4] using Random Forest classifiers on handcrafted statistics, while [17] and [18] combined motion sensors through serial/parallel fusion or augmentation. Similar integration was performed in [19] and [20], and multimodal scoring was applied in [21] for touchscreen and sensor modalities. Multimodal datasets containing swipes, keystrokes, motion sensor data, and gesture traces were introduced in [22]. A two-stage feature extraction combining handcrafted statistical descriptors and deep metric learning on sensor data was presented in [23]. Although fusion often improves accuracy, these methods require specific hardware configurations and preprocessing pipelines, making them dependent on device capabilities. In contrast, the proposed work uses only touchscreen interactions, avoiding reliance on additional sensors.

Deep learning architectures have been employed to improve feature representation and adapt to variability. Latent space mapping was used in [24] for user-specific tapping, while [25] applied CNNs with Monte Carlo Dropout to quantify prediction uncertainty. One-class autoencoding was explored in [26] for unsupervised modeling, and deep metric learning was used in [27] for privacy-preserving on-device

training. CNN–transformer hybrids were proposed in [28], and incremental learning was studied in [29] to address long-term drift. Handcrafted–learned feature integration was implemented in [30], while 3D CNN–RNN spatial–temporal modeling was applied in [31]. Few-shot CNN–Siamese models were explored in [32] and [33], attention-based fusion in [34], and adversarial robustness in [35] and [36]. Accelerometer–gyroscope fusion with deep feature concatenation was demonstrated in [37], and motion-based active authentication was studied in [38]. Keystroke dynamics were modeled in [39] using statistical features and both gradient boosting and deep neural networks. Activity-specific CNN models on motion data were developed in [40] to improve stability, with walking identified as a highly discriminative activity. While these works advance deep feature extraction, they typically operate on aggregated statistics or temporal sequences. The present work differs by directly modeling raw touch feature distributions and applying a CNN-based Siamese network for session-specific similarity learning.

Overall, prior research confirms the feasibility of behavioral biometrics for CA but shows recurring constraints: reliance on handcrafted or aggregated features, dependence on controlled gestures or device-specific sensors, and limited modeling of unconstrained behavioral variability. The proposed method addresses these limitations by representing user-specific probability distributions of touch features and applying deep similarity learning for robust, unconstrained, and personalized continuous authentication.

2.0.2 Touchalytics Dataset

The Touchalytics dataset is a publicly available collection of raw touch interaction data designed for studying behavioral biometrics on mobile devices [41]. Data were collected from 41 users interacting with Android smartphones through navigation tasks such as vertical and horizontal scrolling. For each interaction, 30 behavioral features were extracted, including pressure, finger orientation, stroke velocity, and contact area. These features, recorded at high resolution, enable the analysis of user-specific interaction patterns for authentication.

The dataset is organized into raw touch event logs and pre-computed feature matrices. Raw logs contain timestamps, coordinates, pressure values, and screen orientation. Feature matrices summarize stroke-level attributes for direct use in classification experiments. Intra-session, inter-session, and long-term data are provided, allowing performance evaluation under varying temporal conditions. Frank et al. [41] evaluated the feasibility of continuous authentication using this dataset and highlighted challenges in maintaining accuracy over long periods.

Several studies have applied machine learning and deep learning techniques to the Touchalytics dataset. In [11], touch interactions were analyzed using an omnidirectional approach that avoided separating strokes by direction, in contrast to traditional direction-based profiling. Multiple classifiers, including Extra-Trees and Gradient Boosting, were evaluated across five behavioral feature sets, with the Touchalytics feature set showing strong performance when combining multiple strokes. In [13], deep learning architectures such as LSTM, GRU, CNN-LSTM, and CNN-GRU were trained directly on Touchalytics data to automatically learn discriminative gesture features. The CNN-GRU model achieved the highest overall accuracy, while CNN-LSTM provided the lowest false acceptance and rejection rates. In [14], a fuzzy commitment scheme was combined with deep learning to protect touch-gesture templates. Features were extracted using a CNN–RNN architecture, and binary BCH codes were applied to handle intra-user variation. In [42], user recognition was performed using predefined Touchalytics features together with additional attributes derived from the raw data. Multiple classifiers, including SVM, k-NN, decision trees, and deep neural networks, were tested, with feature augmentation improving performance but relying on aggregated statistics and manually engineered attributes.

These studies demonstrate that the Touchalytics dataset supports both hand-crafted and learned feature-based approaches. However, most methods rely on aggregated stroke features or predefined interaction sequences, potentially omitting fine-grained temporal variations within gestures. In contrast, the present work uses raw interaction data to model complete probability distributions of touch features, preserving the

full behavioral signature without dependence on aggregated statistics or fixed gesture structures.

2.0.3 Posture and Body Movement Dataset

The Posture and Body Movement Dataset is a publicly available resource designed to explore the impact of body posture and motion on continuous smartphone authentication systems. The dataset comprises data from 39 users, collected in a 3D motion capture studio equipped with 30 cameras. Participants were instrumented with 41 reflective markers placed on their bodies and three markers on a smartphone, enabling precise tracking of body and device movements in a three-dimensional space [43].

The dataset includes 29,827 swipes and keystroke data, collected across two sessions with an average gap of one week. During the sessions, participants performed tasks such as swiping and typing in two primary postures: sitting and walking. Sensor data from accelerometers, gyroscopes, and 3D motion capture systems were recorded, capturing features such as velocity, acceleration, and relative angles of body parts. These features were used to analyze the relationship between posture, body movement, and authentication performance.

This dataset has been used in multiple studies that analyze the impact of body posture and motion on authentication accuracy. These studies extracted a variety of posture and motion-related features, such as distances, accelerations, and angular velocities of different body parts, using a 3D motion capture system with reflective markers placed on both the smartphone and the participant's body. The collected data was evaluated using classifiers such as Random Forest (RF), K-Nearest Neighbors (KNN), One-Class Support Vector Machine (OSVM), and Scaled Manhattan Distance (SM). The study demonstrated that integrating body posture and motion features with traditional swipe-based authentication significantly reduces authentication latency while maintaining low Equal Error Rates (EER). For example, using only swipe and phone movement features resulted in an EER of 15.7% at a 60-second authentication window, whereas incorporating body motion features reduced the EER to 6.4% with just

a 1-second latency and 5.4% with a 5-second latency when using a Random Forest classifier. These results highlight the feasibility of enhancing authentication performance through feature fusion, emphasizing the potential for real-world deployment using standard smartphone sensors and wearable devices [44].

2.0.4 HuMIdb Dataset

The HuMIdb (Human Mobile Interaction Database) [45] is a multimodal dataset designed to study human–mobile interaction through behavioral biometrics. It contains data from 600 participants interacting with their own mobile devices in unsupervised scenarios, including indoor and outdoor environments and varying times of day [45, 21]. Data were collected over five sessions with at least a one-day gap between sessions to capture temporal variability.

The dataset comprises 14 sensor modalities, including accelerometer, gyroscope, magnetometer, gravity, proximity, light, touchscreen, and microphone data. User activities include swiping, typing, drawing, and voice input. Touch-based tasks involve fixed and free-text keystroking, swipe gestures, air-drawn patterns such as circles and crosses, and digit handwriting on the touchscreen. This range of modalities and interaction types enables a detailed analysis of user behavior across various input channels. HuMIdb includes participants from 14 countries and age groups ranging from under-20 to over-50 years old, with more than 300 different devices represented. This diversity supports the evaluation of the generalizability and robustness of behavioral biometric methods. In [46], the dataset was used for applications such as bot detection and demographic modeling. In [47], unimodal and multimodal behavioral biometrics were compared for passive mobile authentication using touchscreen gestures and background sensor data from HuMIdb. Separate recurrent neural networks were trained for each modality using semi-hard triplet loss, and score-level fusion was applied to combine results across modalities. Although fusion reduced error rates, the method required multiple sensor streams and complex fusion procedures, which may limit deployment on resource-constrained devices. The present work also uses behavioral biometrics for continuous authentication but focuses exclusively on touch

interactions, modeling complete probability distributions of interaction features, and applying a CNN-based Siamese network for session-to-session similarity learning without requiring sensor fusion.

CHAPTER 3

METHODOLOGY

Let D be a dataset consisting of touch interaction records from a set of users U, collected across multiple sessions S. Each user $u \in U$ has touch interaction data recorded for features $f \in F$ where F represents the set of behavioral features (movement speed, acceleration, curvature, cumulative average speed). Let $D_{u,s,f}$ represent the feature data for user u in session s for feature f. On the other hand, i refers to the index of the current touch sample. x and y denote the screen coordinates (position) of the touch sensor data of the screen touch sensor data at that sample, and t is the corresponding timestamp indicating when the touch occurred.

The following features were extracted to represent user touch behavior:

- **Movement Speed**: Calculated as the ratio of movement distance to time difference between consecutive points.

$$speed(i) = \frac{\sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}}{(t_{i+1} - t_i)}$$

- **Acceleration**: Derived from changes in movement speed across consecutive points.

$$acc(i) = \frac{(speed_{i-1} - speed_i)}{(t_{i-1} - t_i)}$$

- **Curvature**: Computed using three consecutive and discrete points, based on the method proposed by [48], as:

$$\kappa = \frac{2(a_2 b_1 - b_2 a_1)}{(a_1^2 + b_1^2)^{1.5}}$$

where coefficients a_i and b_i are obtained by solving:

$$a_i = \frac{1}{2}\frac{d^2x_i}{dt^2}, \quad b_i = \frac{1}{2}\frac{d^2y_i}{dt^2}$$

for the parametric curve $(x_i(t), y_i(t))$ formed by the three points.

- **Cumulative Average Speed**: Calculates the average of all instantaneous movement speeds from the beginning of the gesture up to the current point P, providing a time-dependent metric that reflects the accumulated speed behavior throughout the interaction.

Current methodology includes the following steps: (1) Parameter Optimization and Data Preprocessing, (2) User and Session Filtering, (3) Bandwidth Selection and Distribution Generation, and (4) Model Architecture and Training.

3.1 Parameter Optimization and Data Preprocessing

To improve model performance, firstly touch data are preprocessed by estimating optimal feature distributions. This preprocessing directly influences the model parameters by providing more reliable input representations. Kernel Density Estimation (KDE) is a non-parametric technique for estimating the probability density function of a variable without assuming a predefined distributional form. The method constructs the density by placing a kernel function K at each observation x_i and averaging over all n data points. The bandwidth parameter h controls the smoothness of the estimate: smaller values capture local variations but increase sensitivity to noise, whereas larger values provide smoother curves at the expense of detail. In this study, KDE is applied to represent user touch patterns as continuous probability distributions:

$$f(y) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{1}$$

where x represents the intended density estimation (evaluation point), x_i represents the input data points (observed touch features), and h is the bandwidth parameter that controls the smoothness of the density estimate.

Algorithm 1 Divergence-Based KDE Parameter Optimization

for each feature f **do**
  Initialize $AUC_{max} \leftarrow 0$
  for (τ, Q_{min}, Q_{max}) combinations **do**
    Generate KDE distributions using Equation 1
    Compute KL-divergence matrix using Equation 2
    Evaluate classification AUC for same/different user pairs
    if current AUC $> AUC_{max}$ **then**
      $\tau_{opt} \leftarrow \tau$
      $Q_{min,opt} \leftarrow Q_{min}$
      $Q_{max,opt} \leftarrow Q_{max}$
      $AUC_{max} \leftarrow$ current AUC
    end if
  end for
end for

For each feature, the optimal parameters are determined through an iterative process that maximizes user discrimination capability. Firstly, the following preprocessing variables are defined for each feature vector:

- *Minimum required data points* (τ_{min}) are required to retain stroke data in a session for reliable distribution estimation.

- *Lower quantile threshold* (Q_{lower}) is used to remove potential outliers in the lower range of the feature vector of a session

- *Upper quantile threshold* (Q_{upper}) is used to filter extreme values in the upper range of a feature vector of a session

A divergence-based KDE Parameter Optimization approach is performed to select the optimal values for the mentioned preprocessing parameters. For each feature, a similarity matrix is constructed using the KL-divergence between user distributions generated with Equation 2:

$$D_{KL}(P||Q) = \sum_{x \in X} P(x) \log \left(\frac{P(x)}{Q(x)} \right) \tag{2}$$

To facilitate parameter selection, the bandwidth parameter is set to 0.9 for KDE estimation during this step, as this parameter has been used as a default parameter in several studies [49]. For each combination of parameters (τ, Q_{min}, Q_{max}), the KL differences between all pairs of users are calculated, creating a similarity matrix based on characteristics M where $M_{i,j}$ represents the KL differences between distributions of users i and j. The combination of parameters in the training data is evaluated by treating the problem as a binary classification task, where labels 0 and 1 are assigned to same-user and different-user pairs, respectively (see Algorithm 1). In addition, users of two sessions are compared based on the movement speed, as shown in Figure 1.

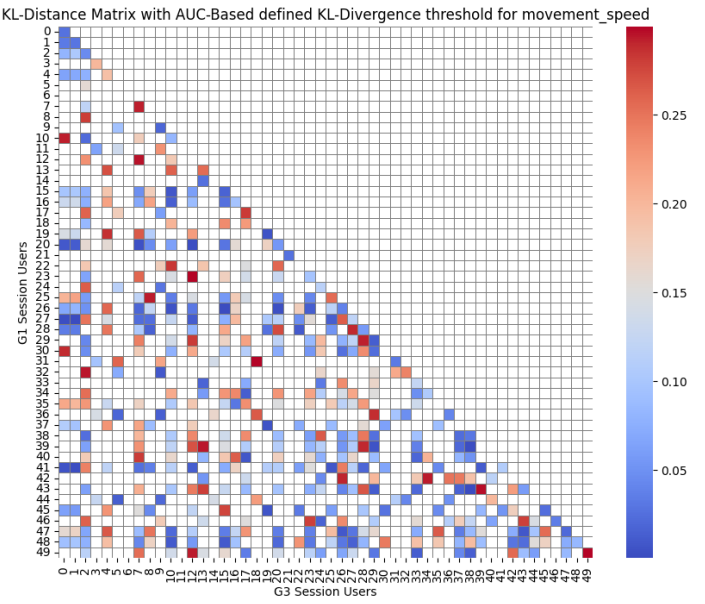


Figure 1: Example Distance matrix for comparing two session users on Movement Speed feature

The combination of parameters that maximizes the ROC-AUC score is selected. The optimization process can be similarly applied for the EER score. This step ensures that selected parameters maximize the discriminative power of feature distributions while maintaining data reliability through minimum point thresholds and outlier removal via quantile filtering. In an operational scenario, removing such instances helps ensure a more reliable representation of user behavior data collected under real-world conditions.

As shown in Figure B.1 in the appendix, Quantile-based filtering on the touch movement speed data removed extreme values. The raw data showed a wide spread with noticeable outliers, particularly near zero and above two. After applying the 30th to 90th percentile filter, the distribution became more compact and symmetric, and the number of data points was reduced from 178 to 106. This filtering step removed noise while preserving the main trend in the data, which became more apparent in the time-series plot. The result provides a cleaner and more stable input for further analysis.

3.2 Data Filtering

The filtering process aims to keep users to satisfy data requirements across all features and sessions, while systematically removing outliers through feature-specific quantile thresholds. The user's data are kept if they meet the requirements for every feature and session. User sessions that do not meet this criterion are excluded from the analysis to maintain data integrity and consistency (see Algorithm 2).

3.3 Bandwidth Selection and Distribution Generation

3.3.1 User-Specific Bandwidth Optimization

In this thesis, it is argued that the bandwidth parameter has a significant influence on the quality of the generated distributions and, consequently, the discriminative power of the resulting authentication system. In the current literature, this bandwidth has

Algorithm 2 Data Filtering Process

for each session s **do**
  for each user u **do**
    if any feature has $|D_{u,s,f}| < \tau_{min,f}$ **then**
      Mark user as ineligible
    end if
    Filter data: $D_{filtered} \leftarrow \{x \mid Q_{lower,f} \leq x \leq Q_{upper,f}\}$
  end for
end for

been selected as fixed, such as in [49]. To determine optimal bandwidth values for each user-feature combination, inspired by triplet loss optimization, an approach is developed. The methodology is formalized (see also Algorithm 3) as follows:

$$b_{opt}(u, f) = \underset{b \in B}{\operatorname{argmin}} \mathcal{L}_{triplet}(D_b(u, f)) \tag{3}$$

where $b_{opt}(u, f)$ represents the optimal bandwidth for user u and feature f, B is the set of candidate bandwidth values, and $D_b(u, f)$ refers to the distribution generated with bandwidth b. KL Divergence metric was selected for distribution comparison due to its theoretical foundations in information theory and its effectiveness in measuring differences between probability distributions.

Algorithm 3 Bandwidth Optimization Process

for each user u **do**
  for each feature f **do**
    for each candidate bandwidth $b \in [0, 1]$ with step 0.05 **do**
      Compute total KL divergence for user
      Update $b_{opt}(u, f)$ if current divergence is minimal
    end for
  end for
end for

Figure 2 illustrates a method for optimizing the bandwidth parameter used to estimate user-specific feature distributions. After suitable users are selected, the session-level

distributions are extracted. For each bandwidth value within a defined range (e.g., [0.05, 1]), KL divergence scores are calculated between session pairs. These include both same-user and different-user comparisons. The aim is to find a bandwidth that results in low KL divergence between sessions of the same user while maintaining a higher divergence between different users. This trade-off is evaluated using a loss function based on the difference between individual and mean different-user divergences, with a margin term added. The selected bandwidth minimizes this loss, ensuring that the estimated distributions are both stable for individual users and distinguishable between different users.

3.3.2 Session-Feature Bandwidth Optimization

The bandwidth selection process makes use of previously obtained user-specific bandwidths to optimize distribution generation. For each session-feature pair, instead of exploring an arbitrary range of bandwidth values, the bandwidth values that were found to be optimal for individual users in the prior optimization step are iterated. This approach ensures that selected bandwidths have already demonstrated effectiveness in capturing user-specific characteristics.

The optimal bandwidth for each session-feature pair is formally defined as:

$$b_{s,f} = \underset{b \in B_{u,f}}{\operatorname{argmin}} D_{KL}(P_b || P_{ref}) \tag{4}$$

where user-feature bandwidths are represented by $B_{u,f}$, P_b is the distribution generated using bandwidth b, and P_{ref} is the reference distribution for the current user (see Algorithm 4.

3.3.3 Distribution Generation for Model Input

The optimized bandwidths are subsequently utilized to generate the final distributions that serve as input to the authentication model. These distributions incorporate the

Algorithm 4 Unlabeled Session-Feature Bandwidth Optimization Process

for each session-feature pair (s, f) **do**
  Initialize minimum divergence $d_{min} \leftarrow \infty$
  Let B_{prev} be the set of previously obtained user-specific bandwidths
  for each bandwidth value $b \in B_{prev}$ **do**
    Generate distribution P_b using bandwidth b
    Compute $D_{KL}(P_b||P_{ref})$
    if $D_{KL}(P_b||P_{ref}) < d_{min}$ **then**
      $b_{s,f} \leftarrow b$
      $d_{min} \leftarrow D_{KL}(P_b||P_{ref})$
    end if
  end for
  Log optimal bandwidth $b_{s,f}$ and minimum divergence d_{min}
end for

learned optimal parameters for each user-feature combination, ensuring maximum discriminative power in the feature space:

$$D_{(u,f)} = \mathcal{KDE}(X_{\text{raw}}^{(u,f)};\, b_{\text{opt}}(u, f)) \tag{5}$$

where $\mathcal{KDE}$ represents the distribution generation function, $X_{u,f}$ is the raw feature data, and $b_{opt}(u, f)$ is the optimal bandwidth determined previously. $D_{(u,f)}$ represents the generated distribution.

3.4 Model Architecture and Training

A CNN architecture is used to process user touch behavior patterns based on optimized feature distributions generated in Equation 5. The network transforms the distribution data into embeddings that capture essential user characteristics. A triplet loss framework is implemented, where the CNN learns from groups of three samples: an anchor sample, a positive sample from the same user, and a negative sample from a different user. The training process adjusts these embeddings to minimize distances

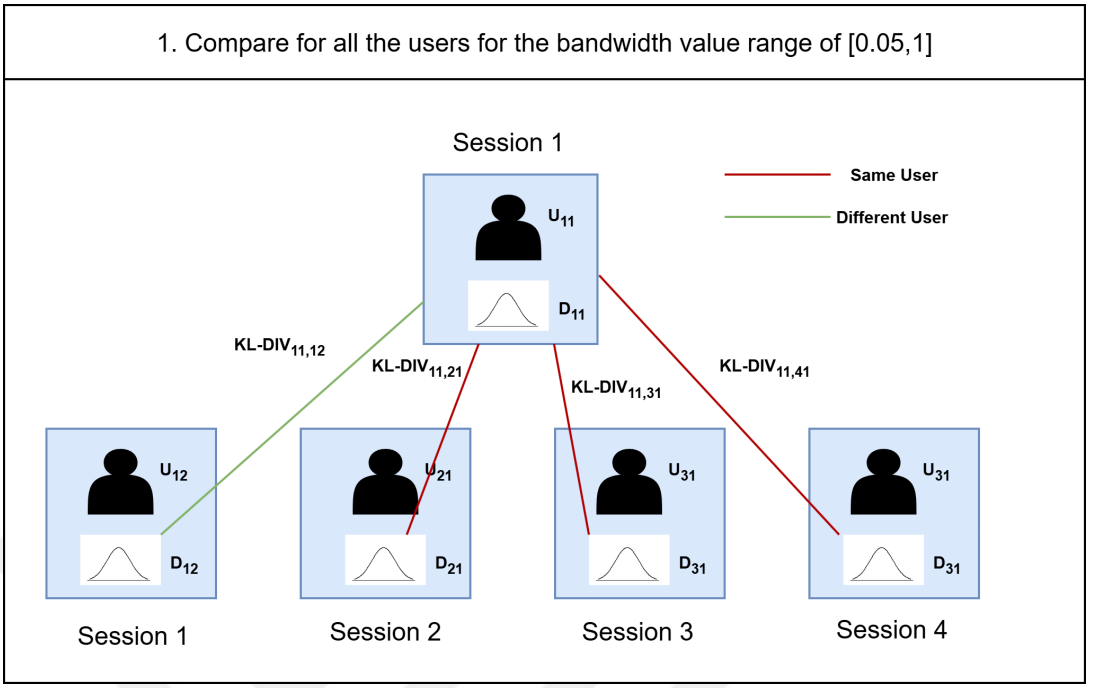


Figure 2: Bandwidth selection process for distribution generation

between samples from the same user while maximizing distances between different users' samples.

The proposed network architecture (see Figure B.2a) consists of two sequential convolutional blocks processing the touch behavior distributions of shape $(\text{Batch Size}, 4, 100)$. Each block comprises a 1D convolutional layer (kernel size=3, channels increasing from 16 to 32), MaxPool1D layer (kernel size = 3), ReLU activation and dropout regularization. Three fully connected layers progressively reduce the dimensionality through shapes 2944, 1472, and finally 736, forming the embedding space. The model training employs a triplet loss framework defined as:

$$\mathcal{L}_{\text{triplet}} = \max\left(0, |f(x_a) - f(x_p)|_2^2 - |f(x_a) - f(x_n)|_2^2 + \alpha\right) \tag{6}$$

where $f(x_a)$, $f(x_p)$, and $f(x_n)$ represent anchor, positive, and negative sample embeddings respectively, and α controls the margin between mated and non-mated samples. Authentication decisions utilize the Euclidean distance between embeddings:

$$d(f(x_i), f(x_j)) = \sqrt{\sum_k (f(x_i)_k - f(x_j)_k)^2} \tag{7}$$

See Figure B.2b for the comparison architecture. This architecture effectively learns discriminative user-specific embeddings while maintaining computational efficiency for mobile deployment.

3.5 Model Evaluation with Unseen Data

The test set consists of users with no identifiable links to training or validation data. To determine the optimal bandwidth for each test user, the bandwidth that minimized the KL divergence between the test and training distributions was selected, which is summarized in Algorithm 4. This ensured that the most representative bandwidth was used for feature extraction in unseen test users. Although the matched distribution may not correspond to the exact user, the purpose is to find an approximate bandwidth that best represents the selected user.

CHAPTER 4

EXPERIMENTS & RESULTS

4.1 Dataset

The BehavePassDB dataset [6] is a large-scale resource for behavioral biometrics and continuous authentication, originally introduced in the IJCB 2022 Mobile Behavioral Biometrics Competition [50]. It includes data from 81 participants across four sessions, each separated by at least 24 hours to capture intra-subject variability. The dataset includes touch interaction data along with optional sensor modalities such as accelerometer, gyroscope, and magnetometer, collected from various mobile devices. This multi-device data acquisition supports cross-platform analysis, enabling robust multi-modal studies and sensor fusion for improved authentication performance. BehavePassDB captures four tasks—keystroke, vertical/horizontal swiping, and tapping—recording touch dynamics such as timing, pressure, and movement patterns. It is divided into development, validation, and evaluation subsets, enabling standardized training and benchmarking. Studies have shown that the integration of motion sensors with touch data improves authentication accuracy [27].

This study utilizes touch data from the Text Reading task, capturing high-resolution time-series at millisecond granularity. Key features include timestamps, normalized x/y coordinates, and touch type (press "0", lift "1", move "2"). Only dragging touch points (touch type = 2) were retained for consistency. This task was selected because its continuous vertical swiping produces longer and more consistent touch trajectories, making it more representative of natural interaction patterns. In this task, the user first reads a text by swiping vertically, then answers a text-related question, and

finally completes a drag-and-drop gesture. See Figures A.2 and A.3 in Appendix that display mobile touch data for two sessions. Both sessions feature a main cluster of activity on the right side of the screen, indicating the primary interaction area. The temporal progression within the main clusters is visible through color changes, with earlier events in darker shades and later events in lighter shades.

4.1.1 Exploratory Data Analysis and Preprocessing

Exploratory data analysis (EDA) was performed to assess the quality and characteristics of sensor and touch data, identifying anomalies that could impact modeling and authentication. Sensor data collection is inherently prone to noise, inconsistencies, and inaccuracies due to environmental factors, device variability, and user behavior. The analysis showed inconsistencies in sensor timestamps, particularly in accelerometer readings across five sessions from two users. See Figure A.1 in the appendix for these irregularities. The x-axis represents the sample index and the y-axis represents the timestamp values. In the upper sub-figure, intervals can be observed where the timestamp values remain constant, indicating temporary lags in the sensor's process of recording events. These lags are likely the result of internal sensor processing delays, disrupting temporal alignment, and potentially affecting authentication accuracy. To address this issue, linear interpolation is applied to reconstruct a continuous timestamp sequence, ensuring consistent temporal progression. This preprocessing step is essential to preserve the integrity of time-series data and to prevent artifacts that could misrepresent user behavior.

Furthermore, as shown in Figure A.5 in the appendix, the sensor data was not collected at a fixed sampling frequency. Although this does not impact the core approach presented in this thesis, it becomes relevant when extending the system to include additional sensors. In such cases, upsampling and downsampling strategies should be considered to maintain consistency.

These anomalies pose challenges in user authentication by distorting feature distributions and affecting similarity metrics. Distance-based models may misinterpret

noisy strokes as distinct user behaviors, reducing system robustness. These anomalies might distort the computation of summary metrics such as the mean and standard deviation of touch data in a user session, leading to inaccuracies in classification. The proposed approach inherently deals with such inconsistencies.

4.2 Model Comparison

A baseline model is applied, inspired by TouchAnalytics [41, 12], using start and end coordinates, trajectory length, stroke duration, deviation from end-to-end line, along with velocity and acceleration percentiles (20th, 50th, 80th) and their arithmetic means. These handcrafted features capture basic geometric and temporal characteristics of touch interactions while avoiding complex transformations.

To evaluate performance, the ROC AUC, the equal error rate (EER), accuracy, precision, and the F1 score are used, ensuring a comprehensive assessment of the effectiveness of authentication in various user behaviors.

4.2.1 Dataset Preparation for Model Training and Evaluation

The dataset was split into training, validation, and test sets, while ensuring a balanced distribution. Sessions (g_1, g_2, g_3, g_4) were collected from users on their own devices, while the independent test set also covers cases where impostors try to authenticate on the same device as genuine users.

4.2.1.1 Training Dataset

The training set utilizes three sessions (g_1, g_2, g_3) through iterative sampling, where combinations are created as (g_1, g_2), (g_2, g_3), (g_1, g_3):

- For each (g_i, g_j) pair:
  - **Mated:** Same user samples from g_i and g_j

– **Non-Mated:** Different user samples within g_i

4.2.1.2 Validation Dataset

Using all four sessions (g_1, g_2, g_3, g_4):

- **Mated:** One g_4 sample paired with same user's (g_1, g_2, g_3) samples
- **Non-Mated:** g_4 samples paired with different users

Validation dataset used to determine optimal authentication thresholds.

4.2.1.3 Test Dataset

An independent test set, separate from the (g_1, g_2, g_3, g_4) sessions, includes pairs of enrollment samples and verification samples. This setup prevents data leakage and enables a reliable assessment of the model's generalization. Although the original dataset includes a separate test set, its evaluation is limited to AUC, which is insufficient to examine the results from multiple perspectives and metrics.

4.3 Experimental Setup

The Algorithm 1 is executed for the following ranges: lower threshold (Q_{lower}) from 0 to 0.3 with 0.05 increments, higher threshold (Q_{upper}) from 0.7 to 1.0 with 0.05 increments, and minimum data points (τ_{min}) from 4 to 30 with a step size of 2. The values in Table 1 are obtained as the best parameters using the training dataset. After applying Algorithm 2, the number of eligible test sessions decreased from 120 to 96. During the threshold optimization process, the ROC threshold (1.58) was selected based on the maximum difference between TPR and FPR. However, the EER threshold (1.39) was selected at the point where FPR equals FNR.

Table 1: Optimized Feature-Specific Parameters

Feature	τ_{min}	Q_{lower}	Q_{upper}
Movement Speed	4	0.30	0.90
Acceleration	4	0.25	0.85
Curvature	12	0.20	1.00
Cumulative Avg. Speed	10	0.00	0.85

4.4 Results

The optimized approach, using user-specific bandwidths, was compared to a standard XGBoost model. Furthermore, an alternative model is developed using a fixed bandwidth of 0.9, following [49], while applying the same user and session filtering criteria. This alternative model (referred to as the *non-optimized approach*) serves to highlight the impact of user-specific bandwidth selection by contrasting it against a standardized setting.
In addition, quantile filtering processes were also investigated. Without applying those quantiles, the filtered users and the entire population were compared.

The methodology demonstrated a significant improvement over the XGBoost baseline across multiple performance metrics based on [51], as summarized in Table 2. It achieved a ROC AUC score of 0.6695 (compared to baseline 0.4924) and improved precision from 0.2463 to 0.4359. Although there is improvement compared to the baseline, there is no predominant threshold selection method for ORT and OET. Accuracy and precision are better for the ROC threshold selection strategy (ORT); on the other hand, the F1 score, balanced accuracy, and MCC scores are better for the EER thresholding (OET). Higher precision can be considered as higher protection against the imposter users.

Table 2: Comparison of Baseline and Optimized Models on Test Set Metrics. ORT: Optimized ROC Threshold, OET: Optimized EER Threshold

Measure	**Baseline**	**ORT**	**OET**
ROC AUC	0.4924	**0.6695**	**0.6695**
EER	0.4802	**0.3697**	**0.3697**
Accuracy	0.5310	**0.6250**	0.6146
Precision	0.2463	**0.4359**	0.4318
F1 Score	0.4238	0.4857	**0.5067**
Balanced Accuracy	0.5158	0.6050	**0.6141**
MCC	0.1843	0.1999	**0.2142**

Table 3: Comparison of Optimized (O) and Non-Optimized (NO) Model Performances

Metrics	**Validation**		**Test**	
	NO	O	NO	O
ROC	0.7372	**0.8176**	0.596	**0.6695**
EER	0.3343	**0.2536**	0.4489	**0.3697**
ROC Threshold (ORT) Performance				
Accuracy	**0.745**	0.7437	0.625	0.625
Precision	0.1404	**0.1677**	0.4324	**0.4359**
Recall	0.6014	**0.7826**	0.5161	**0.5484**
F1	0.2277	**0.2762**	0.4706	**0.4857**
Balanced Accuracy	0.678	**0.7618**	0.5965	**0.605**
MCC	0.1947	**0.2789**	0.1855	**0.1999**
EER Threshold (OET) Performance				
Accuracy	0.6649	**0.7468**	0.5521	**0.6146**
Precision	0.117	**0.1643**	0.3696	**0.4318**
Recall	0.6667	**0.7464**	0.5484	**0.6129**
F1	0.1991	**0.2693**	0.4416	**0.5067**
Balanced Accuracy	0.6657	**0.7466**	0.5511	**0.6141**
MCC	0.1657	**0.2648**	0.0957	**0.2142**

Table 3 details the performance of these user-specific bandwidths, and there are some important patterns. The optimized model exhibits consistent improvements over the non-optimized version, with higher ROC scores and reduced EER values across both validation and test sets. An apparent performance gap between validation and test results hints at potential overfitting, but the optimized model retains its competitive edge in the test environment. Under the EER threshold, the enhancements in the F1 and MCC scores reflect the robustness of the model when calibrated for error balance. While test precision sees a modest rise for the optimized model, recall experiences a

slight dip, a trend potentially related to the limited sample size of the test set. The contrast between ORT and OET configurations is clear, with OET-driven metrics that showcase clearer advances, emphasizing the value of threshold-specific tuning.

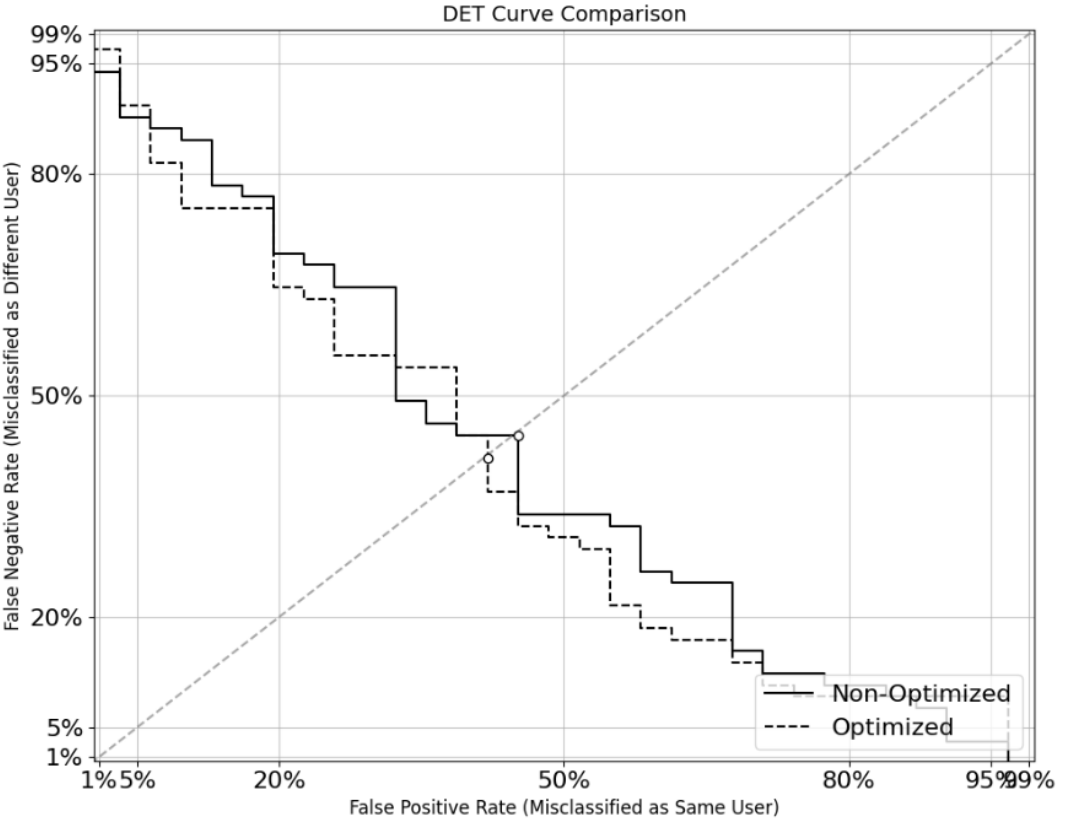
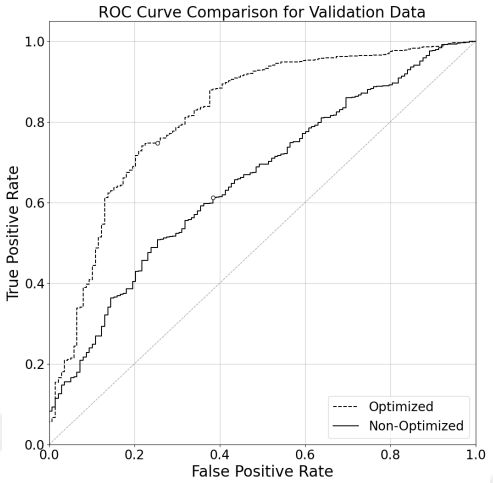


Figure 3: DET Curve Comparison of Non-Optimized and Optimized Models.

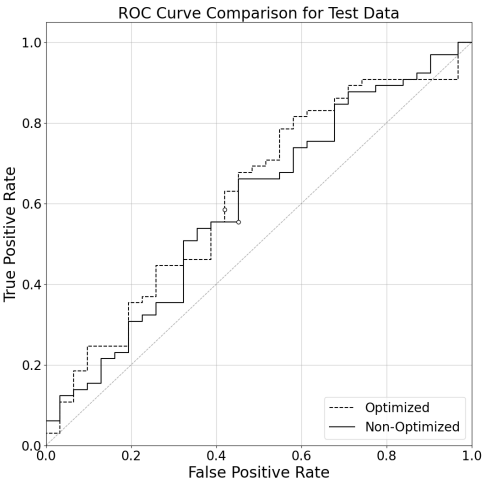
The Detection Error Tradeoff Curve (DET) highlights the reliability of the optimized model, particularly in minimizing false positives, a critical area where the risks of misclassification are severe. By achieving tighter control over false acceptance errors, the optimized model demonstrates a performance increase.

The Wilcoxon signed-rank test was applied to compare performance between optimized and non-optimized authentication models. Paired comparisons were conducted for three key metrics (false acceptance rate, false rejection rate, and Matthews correlation coefficient) for different threshold types. The results show that the *optimized approach* outperforms the *non-optimized approach* based on False Positive Rate (FPR) *(Z = -0.06, p = 0.016, N = 17)*, Matthews Correlation Coefficient (MCC) *(Z = -0.14,*

p = 0.016, N = 17), and False Negative Rate (FNR) *(Z = -0.14, p = 0.021, N = 17)*, confirming that the improvements are statistically significant. In Figure 4, the opti-



(a) ROC Curve Comparison for Validation Data



(b) ROC Curve Comparison for Test Data

Figure 4: Roc Curves for both Test and Validation Data

mized bandwidth setting achieves a sharper rise in the true positive rate at lower false positive rates, indicating a stronger early discrimination capability. This steep initial increase highlights better separation between classes under optimized settings. On the validation set, the optimized curve consistently dominates the non-optimized one, reflecting improved threshold learning.

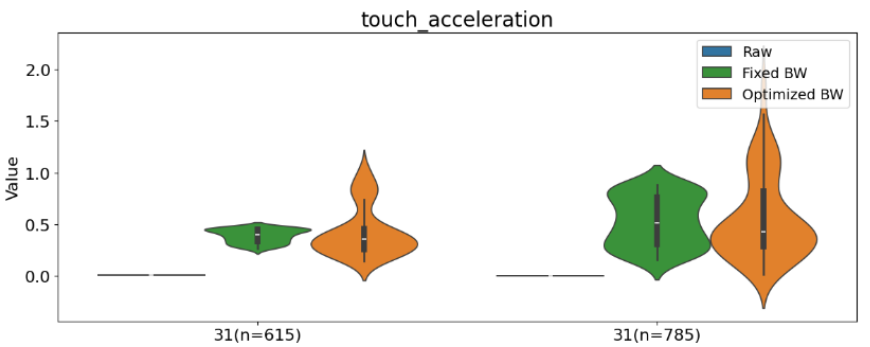


Figure 5: Acceleration feature distributions for User 31 with raw and processed data.

For the acceleration feature (Figure 5), the optimized approach with (minimum points=4, lower quantile=0.25, upper quantile=0.85, fixed bandwidth=0.9, optimized bandwidth=0.05,

bandwidth range=0-1) can simulate distributions, particularly in the sessions g_2 and g_4. This suggests that a smaller bandwidth may be more suitable for capturing fine-grained acceleration patterns.

4.5 Ablation Study

An ablation study was conducted to evaluate quantile filtering. The models were tested with filtered and unfiltered data using ROC AUC, EER, and F1 score metrics. The filtering step's effect on KDE-based feature modeling and Siamese network similarity learning was examined. Experiments were performed on the BehavePassDB dataset. The study demonstrated the contribution of quantile filtering to authentication performance across different preprocessing conditions.

Table 4: Comparison of Optimized (O) and Non-Optimized (NO) Model Performances against the Unprocessed data with Filtering(F) and Unfiltering(UF) Users

Metrics	**Noquantile**		**Test**	
	UF	F	NO	O
ROC	**0.5793**	0.5708	0.596	**0.6695**
EER	**0.4539**	0.4637	0.4489	**0.3697**
ROC Threshold (ORT) Performance				
Accuracy	0.4737	**0.51**	0.625	0.625
Precision	0.3429	**0.3455**	0.4324	**0.4359**
Recall	**0.6316**	0.5938	0.5161	**0.5484**
F1	**0.4444**	0.4368	0.4706	**0.4857**
Balanced Accuracy	0.5132	**0.5322**	0.5965	**0.605**
MCC	0.0255	**0.0603**	0.1855	**0.1999**
EER Threshold (OET) Performance				
Accuracy	0.4825	**0.49**	0.5521	**0.6146**
Precision	**0.3478**	0.3333	0.3696	**0.4318**
Recall	**0.6316**	0.5938	0.5484	**0.6129**
F1	**0.4486**	0.427	0.4416	**0.5067**
Balanced Accuracy	**0.5197**	0.5175	0.5511	**0.6141**
MCC	**0.0381**	0.0329	0.0957	**0.2142**

Table 4 shows the performance metrics of the raw data. All metrics except recall are improved when processed data are compared to unprocessed data. Recall represents genuine misclassified users, while precision indicates misclassified impostors. For unprocessed data, better ROC and EER results are observed in UF(Unfiltered users)

compared to F(Filtered users), suggesting that behavioral patterns are better captured for unseen test data. In filtered cases, ORT threshold selection yields higher balanced accuracy and MCC scores than OET. The precision in UF-OET slightly exceeds F-ORT. As in Table 3, superior ORT-based performance is demonstrated.

CHAPTER 5

CONCLUSION AND DISCUSSION

5.1 Conclusion

A touch-based mobile authentication system based on feature distribution learning was developed and evaluated within a Siamese convolutional framework. Instead of summary statistics, feature sequences (acceleration, speed, curvature, and cumulative average speed) were represented as probability distributions. The similarity between distributions was calculated using a divergence measure and user-specific bandwidth optimization was applied.

Across datasets, the optimized model outperformed the non-optimized baseline. In validation, the ROC increased from 0.7372 to 0.8176 (+0.0804; +10.9%), while EER decreased from 0.3343 to 0.2536 (0.0807; 24.1%). In the test, the ROC increased from 0.5960 to 0.6695 (+0.0735; +12.3%), and EER decreased from 0.4489 to 0.3697 (0.0792; 17.6%). At the equal-error operating point (OET), test-set F1 improved from 0.4416 to 0.5067 and MCC from 0.0957 to 0.2142, with accuracy increasing from 0.5521 to 0.6146. At the ROC-selected threshold (ORT), the accuracy in the test remained 0.625, while F1, balanced accuracy, and MCC showed modest gains.

The ablation results indicated that the distributional preprocessing and optimization improved performance over the unprocessed inputs. Compared to unfiltered, unprocessed data, the test ROC increased from 0.5793 to 0.6695 (+0.0902; +15.6%), and EER decreased from 0.4539 to 0.3697 (0.0842; 18.6%). The improvements were greater at the OET operating point than at ORT, indicating that threshold choice meaningfully affects reported gains.

Taken together, these results show that modeling entire feature distributions, together with user-specific bandwidth selection, yields consistent gains over summary-based baselines. For deployment, the findings imply that (i) divergence-based matching with distributional features can raise verification accuracy, (ii) user-specific bandwidths and quantile-based filtering provide measurable benefits, and (iii) calibration at the equal-error operating point produces stronger improvements in F1 and MCC than ROC-based thresholding.

5.2 Discussion

The results provide support for distribution-based representations in behavioral biometrics. By retaining the shape of the feature distributions rather than compressing the sequences into a few statistics, the variability of user interactions was preserved and the sensitivity to outliers was reduced. The observed gains over the summary-based baseline align with earlier reports in the literature that describe advantages for density- or histogram-level features when intra-user variability and skewed event counts are present. The present findings are consistent with those reports and extend them by showing that distribution inputs can be integrated with a Siamese encoder and a divergence-based objective.

The role of bandwidth selection is central in this setting. Kernel bandwidth governs the bias–variance trade-off in density estimation; user-specific optimization produced distributions that were better aligned with enrollment data, which was reflected in lower EER and higher ROC. The ablation comparing optimized distributions to unprocessed inputs further indicates that improvements arise from both the representation and the learned bandwidths. Quantile-based filtering improved most metrics relative to unfiltered inputs, while the decline in recall suggests that discarding sparse regions of the distribution can remove genuine events; this trade-off should be considered when setting filtering rules.

Operating-point analysis clarifies how gains appear in practice. Larger improvements were obtained under the OET configuration than under ORT. This pattern is consistent

with the expectation that systems calibrated to equalize false acceptance and false rejection rates benefit more from distributionally smoother scores. The unchanged ORT accuracy on the test, alongside higher F1 and MCC, shows that even when the overall accuracy is steady, the quality of positive decisions and rank correlation can improve.

From a practical perspective, real-life deployments such as mobile banking require stricter control of false acceptances, since misclassifying an impostor as a genuine user poses a higher risk than rejecting a legitimate user. In such scenarios, the proposed approach could operate in the background during normal usage and trigger stronger verification before authorizing sensitive operations like fund transfers.

A gap was observed between validation and test was observed (e.g., ROC 0.8176 vs. 0.6695; EER 0.2536 vs. 0.3697). This pattern indicates distributional differences between splits and suggests that the learned parameters were partly tuned to the validation distribution. In behavioral biometrics, such gaps are common due to session effects and context shifts; the current results follow that trend while retaining a consistent advantage for the optimized model. The following questions were asked and answered in the study:

RQ1 What is the impact of distribution-based methods for feature representation on the performance and robustness of biometric authentication models?
Distribution-based methods, instead of summary-statistics, are proposed to improve the accuracy of identification in behavioral biometrics. The distribution-based approach was found to perform better than a baseline model using summary statistics of features. Sensitivity to outliers was identified as a limitation in summary statistics, since measures such as the mean can be affected by extreme values. In distribution-based representations, the overall shape of the data was retained, which reduced the impact of outliers and allowed more detailed variation in user interactions to be considered.

RQ2 How can feature representation and optimization strategies effectively address inconsistencies and noise in biometric data to improve model performance?

It is hypothesized that the use of proper preprocessing techniques in the data improves the distribution that was generated. The main proposed approach for preprocessing was used to discard outliers using quantile-based filtering and users who contain a few data points. The results showed that the use of quantile filtering improved overall performance, which is in line with the methodology. However, omitting users with very few data points decreased the evaluation metrics, suggesting that there is some information to extract even from the minimal data points.

Finally, comparison with earlier studies indicated that the equal error rate (EER) obtained in this work—above 30% on the test set—remains higher than the values reported in [47] (24.99%) and [52] (23.93%). This difference suggests that further refinements are required in the proposed approach. The present results suggest that density-level representations, when combined with a Siamese metric, may offer competitive accuracy and provide a practical means for threshold calibration and user-specific adaptation.

5.3 Limitations

Threshold selection was found to be sensitive to context, with optimal configurations varying between ORT and OET approaches based on evaluation metrics. Data requirements led to the exclusion of 20% of test session pairs due to minimum point thresholds, reducing the number from 120 to 96. Additionally, during the model development, the number of pairs was reduced from 2500 to 2208.

In this study, unlike the training and validation sets, where each user interacted with their own device, the test set includes cases in which impostors use the same device as genuine users. This setup may reduce device-level variability and could underestimate the difficulty of cross-device generalization in real-world scenarios.

In addition, KL divergence was implemented in this study; however, its asymmetric property makes the computed value sensitive to the order in which the two distributions are compared. Specifically, reversing the roles of the reference and comparison distributions can yield different divergence scores. In this work, the divergence was

consistently computed in a fixed order, which may have influenced the results. The use of a symmetric measure, such as Jensen–Shannon divergence, could mitigate this sensitivity. The fixed-bandwidth baseline used in this work relied on a single value, $h = 0.9$, which has been commonly adopted in the literature. Although this provides a reasonable reference point, it does not capture the potential variability that could arise from alternative fixed bandwidths. The study is limited due to the use of only one dataset for evaluation. To generalize the results, HuMIdb [45] can be used.

In the test set, user identifiers were not available. Bandwidth values were therefore approximated by selecting the closest training user distributions based on the KL divergence. This indirect matching may not fully capture user-specific characteristics, as bandwidths were assigned heuristically rather than from labeled test data.

5.4 Future Work

The limitations identified in this study point to several directions for further research. First, the sensitivity of the KL divergence to the input order suggests that symmetric divergence measures could be explored. Jensen–Shannon divergence, for example, would treat the two distributions equally, potentially providing more stable comparisons in user verification.

Second, the current implementation employs a centralized model, where user data must be collected and processed in a single location. This setup raises privacy concerns, as raw biometric data can be exposed to unauthorized access during transmission or storage. An alternative is to investigate federated learning frameworks in which the model is trained locally on user devices and only model updates, not raw data, are shared. This approach could reduce privacy risks while preserving accuracy if aggregation and communication overhead are properly managed.

Third, temporal adaptation methods should be examined to address performance degradation over extended usage periods. For instance, gradual updates to the user's reference distribution based on recent authenticated sessions can help maintain accuracy as touch behavior changes over time. Techniques such as incremental learning with

a decay factor could balance responsiveness to new data with retention of stable patterns reported in earlier sessions. An additional future direction is to apply Optimal Transport [53] metrics to capture evolving user behaviors. By computing distances between enrollment and later session distributions, OT can quantify behavioral drift over time.

Fourth, context-aware threshold adjustment could be implemented to adapt decision boundaries to the conditions under which authentication is performed. Examples include lowering the threshold in high-security scenarios (e.g., financial transactions) or relaxing it when rapid access is prioritized (e.g., unlocking in low-risk environments). The context can be inferred from device usage patterns, geolocation, or application type.

Fifth, testing multiple fixed bandwidth values (e.g., 0.05, 0.5, 1.0) instead of a single baseline would make the comparison more solid and show whether the improvements come from user-specific optimization or could also be achieved with other fixed values.

Finally, multimodal extensions could be explored by integrating additional behavioral and sensor modalities, such as accelerometer and gyroscope signals, into the distributional framework. This may improve resilience to environmental variations and increase discrimination between users under diverse usage conditions.

REFERENCES

[1] C. Yuan, Z. Xu, X. Li, Z. Zhou, J. Huang, and P. Guo, “An Interpretable Siamese Attention Res-CNN for Fingerprint Spoofing Detection,” *IET Biometrics*, vol. 2024, Article ID 6630173, 2024. Available: https://doi.org/10.1049/2024/6630173.

[2] C. Bo, L. Zhang, T. Jung, J. Han, X.-Y. Li, and Y. Wang, “Continuous user identification via touch and movement behavioral biometrics,” in *2014 IEEE 33rd International Performance Computing and Communications Conference (IPCCC)*, 2014, pp. 1–8. Available: https://doi.org/10.1109/PCCC.2014.7017067.

[3] Alawami, M. A., Abuhmed, T., Abuhamad, M. and Kim, H. (2024). MotionID: Towards practical behavioral biometrics-based implicit user authentication on smartphones. Pervasive and Mobile Computing, 101. https://doi.org/10.1016/j.pmcj.2024.101922

[4] Al-Saraireh, J., & AlJa’afreh, M. R. (2023). Keystroke and swipe biometrics fusion to enhance smartphones authentication. Computers and Security, 125. https://doi.org/10.1016/j.cose.2022.103022

[5] Chen, Y., Liu, G., Yu, L., Kang, H., Meng, L. and Wang, T. (2025). TBAuth: A continuous authentication framework based on tap behavior for smartphones. Expert Systems with Applications, 264. https://doi.org/10.1016/j.eswa.2024.125811

[6] G. Stragapede, R. Vera-Rodriguez, R. Tolosana, and A. Morales, "BehavePassDB: Public database for mobile behavioral biometrics and benchmark evaluation," *Pattern Recognition*, 2022.

[7] M. Antal and L. Z. Szabó, "Biometric authentication based on touchscreen swipe patterns," *Procedia Technology*, vol. 22, 2016, pp. 862–869. Available: https://doi.org/10.1016/j.protcy.2016.01.061.

[8] J. Chao, M. S. Hossain, and L. Lancor, "Swipe gestures for user authentication in smartphones," *Journal of Information Security and Applications*, vol. 74, 103450, 2023. Available: https://doi.org/10.1016/j.jisa.2023.103450.

[9] A. Mahfouz, A. Hamdy, M. A. Eldin, and T. M. Mahmoud, "B2auth: A Contextual Fine-grained Behavioral Biometric Authentication Framework for Real-world Deployment," *Pervasive and Mobile Computing*, vol. 99, p. 101888, 2024. DOI: https://doi.org/10.1016/j.pmcj.2024.101888.

[10] Yang, Y., Guo, B., Wang, Z., Li, M., Yu, Z., & Zhou, X. (2019). BehaveSense: Continuous authentication for security-sensitive mobile apps using behavioral biometrics. Ad Hoc Networks, 84, 9–18. https://doi.org/10.1016/j.adhoc.2018.09.015

[11] Aaby, P., Giuffrida, M. V., Buchanan, W. J. and Tan, Z. (2023). An omnidirectional approach to touch-based continuous authentication. Computers and Security, 128. https://doi.org/10.1016/j.cose.2023.103146

[12] W. Meng, D. Wong, R. Schlegel, and L. Kwok, "Touch Gestures Based Biometric Authentication Scheme for Touchscreen Mobile Phones," in *Proceedings*, Nov. 2012, ISBN: 978-3-642-38518-6. https://doi.org/10.1007/978-3-642-38519-3_21

[13] Bajaber, A., Fadel, M., & Elrefaei, L. (2021). Evaluation of Deep Learning Models for Person Authentication Based on Touch Gesture. Computer Systems Science and Engineering, 42(2), 465–481. https://doi.org/10.32604/csse.2022.022003

[14] Bajaber, A., & Elrefaei, L. (2022). Biometric Template Protection for Dynamic Touch Gestures Based on Fuzzy Commitment Scheme and Deep Learning. Mathematics, 10(3). https://doi.org/10.3390/math10030362

[15] Zhang, J., Li, Z., Zhang, H., Zhang, W., Ling, Z., & Yang, M. (2023). Sensor-based implicit authentication through learning user physiological and behavioral characteristics. Computer Communications, 208, 244–255. https://doi.org/10.1016/j.comcom.2023.06.016

[16] Zhao, C., Gao, F. and Shen, Z. (2024). Multi-motion sensor behavior based continuous authentication on smartphones using gated two-tower transformer fusion networks. Computers and Security, 139. https://doi.org/10.1016/j.cose.2023.103698

[17] Y. Li, B. Zou, S. Deng, and G. Zhou, "Using Feature Fusion Strategies in Continuous Authentication on Smartphones," *IEEE Internet Computing*, vol. 24, no. 2, pp. 49–56, 2020. https://doi.org/10.1109/MIC.2020.2971447.

[18] Li, Y., Hu, H., & Zhou, G. (2019). Using Data Augmentation in Continuous Authentication on Smartphones. IEEE Internet of Things Journal, 6(1), 628–640. https://doi.org/10.1109/JIOT.2018.2851185

[19] Tang, G., Zhao, J., Xu, Z., Wang, Y., Yang, H., & Zhang, Z. (2024). Continuous Smartphone User Authentication Based on Gesture-Sensor Fusion. Proceedings - 2024 International Conference on Networking and Network Applications, NaNA 2024, 516–522. https://doi.org/10.1109/NaNA63151.2024.00091

[20] Gattulli, V., Impedovo, D., Pirlo, G., & Volpe, F. (2023). Touch events and human activities for continuous authentication via smartphone. Scientific Reports, 13(1). https://doi.org/10.1038/s41598-023-36780-3

[21] M. Papaioannou, G. Zachos, G. Mantas, A. Essop, A. Karasuwa and J. Rodriguez, "Generating Datasets Based on the HuMIdb Dataset for Risk-based User Authentication on Smartphones," 2022 IEEE 27th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), 2022, pp. 134-139.

[22] A. Ray-Dowling, A. A. Wahab, D. Hou, and S. Schuckers, "Multi-Modality Mobile Datasets for Behavioral Biometrics Research," in *Proc. of the 13th ACM Conf. on Data and Application Security and Privacy (CODASPY '23)*, Charlotte, NC, USA, Apr. 2023, pp. 1–6. DOI: https://doi.org/10.1145/3577923.3583637.

[23] Hu, M., Zhang, K., You, R., & Tu, B. (2023). Multisensor-Based Continuous Authentication of Smartphone Users With Two-Stage Feature Extraction. IEEE Internet of Things Journal, 10(6), 4708–4724. https://doi.org/10.1109/JIOT.2022.3219135

[24] Heo, J., Kim, J., Chung, E., Kim, S., Kang, P., Shin, D., Shin, J., & Han, D. (2025). Normalizing flow-based latent space mapping for implicit pattern authentication on mobile devices. Applied Soft Computing, 169. https://doi.org/10.1016/j.asoc.2024.112469

[25] Ahmadian, R., Ghatee, M., Wahlstrom, J., & Zare, H. (2024). Uncertainty Quantification to Enhance Probabilistic Fusion Based User Identification Using Smartphones. IEEE Internet of Things Journal. https://doi.org/10.1109/JIOT.2024.3429011

[26] Hu, M., Wang, D., Li, C., Xu, Y., & Tu, B. (2024). Behavioral Biometrics-based Continuous Authentication Using A Lightweight Latent Representation Masked One-Class Autoencoder. IEEE Transactions on Dependable and Secure Computing. https://doi.org/10.1109/TDSC.2024.3472631

[27] C. Wang, Y. Xiao, X. Gao, L. Li, and J. Wang, "A framework for behavioral biometric authentication using deep metric learning on mobile devices," *IEEE Transactions on Mobile Computing*, vol. 22, no. 1, pp. 19–36, 2023. Available: https://doi.org/10.1109/TMC.2022.3145673.

[28] Hu, M., Zhang, K., You, R., & Tu, B. (2023). AuthConFormer: Sensor-based Continuous Authentication of Smartphone Users Using A Convolutional Transformer. Computers and Security, 127. https://doi.org/10.1016/j.cose.2023.103122

[29] Shen, Z., Li, S., Zhao, X., & Zou, J. (2024). IncreAuth: Incremental-Learning-Based Behavioral Biometric Authentication on Smartphones. IEEE Internet of Things Journal, 11(1), 1589–1603. https://doi.org/10.1109/JIOT.2023.3289935

[30] Song, Y., & Cai, Z. (2022). Integrating Handcrafted Features with Deep Representations for Smartphone Authentication. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 6(1). https://doi.org/10.1145/3517332

[31] Shen, Z., Li, S., Zhao, X., & Zou, J. (2024). CT-Auth: Capacitive Touchscreen-Based Continuous Authentication on Smartphones. IEEE Transactions on Knowledge and Data Engineering, 36(1), 90–106. https://doi.org/10.1109/TKDE.2023.3277879

[32] Fereidooni, H., König, J., Rieger, P., Chilese, M., Gökbakan, B., Finke, M., Dmitrienko, A., & Sadeghi, A. R. (2023). AuthentiSense: A Scalable Behavioral Biometrics Authentication Scheme using Few-Shot Learning for Mobile Platforms. 30th Annual Network and Distributed System Security Symposium, NDSS 2023. https://doi.org/10.14722/ndss.2023.23194

[33] Wagata, K., & Teoh, A. B. J. (2022). Few-Shot Continuous Authentication for Mobile-Based Biometrics. Applied Sciences (Switzerland), 12(20). https://doi.org/10.3390/app122010365

[34] Zhao, C., Gao, F., & Shen, Z. (2024). AttAuth: An Implicit Authentication Framework for Smartphone Users Using Multimodality Data. IEEE Internet of Things Journal, 11(4), 6928–6942. https://doi.org/10.1109/JIOT.2023.3314717

[35] Agrawal, M., Mehrotra, P., Kumar, R., & Shah, R. R. (2022). GANTouch: An Attack-Resilient Framework for Touch-based Continuous Authentication System. http://arxiv.org/abs/2210.01594

[36] Zou, S., Sun, H., Xu, G., Wang, C., Zhang, X., & Quan, R. (2023). A Robust Continuous Authentication System Using Smartphone Sensors and Wasserstein Generative Adversarial Networks. Security and Communication Networks, 2023. https://doi.org/10.1155/2023/3673113

[37] Li, Y., Tao, P., Deng, S., & Zhou, G. (2022). DeFFusion: CNN-based Continuous Authentication Using Deep Feature Fusion. ACM Transactions on Sensor Networks, 18(2). https://doi.org/10.1145/3485060

[38] M. Abuhamad, T. Abuhmed, D. Mohaisen, and D. Nyang, "AUToSen: Deep-learning-based implicit continuous authentication using smartphone sensors," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 5008–5020, June 2020. DOI: 10.1109/JIOT.2019.2952913.

[39] Chang, H.-C., Li, J., Wu, C.-S., & Stamp, M. (2021). Machine Learning and Deep Learning for Fixed-Text Keystroke Dynamics. http://arxiv.org/abs/2107.00507

[40] Rukkanchanunt, T., Petchkuliinda, S., & Ochiai, H. (2024). Behavioral Authentication Using Activity-Based Model Segregation in Convolutional Neural Networks for Mobile Sensor Data. Proceedings - 21st International Joint Conference on Computer Science and Software Engineering, JCSSE 2024, 146–152. https://doi.org/10.1109/JCSSE61278.2024.10613628

[41] M. Frank, R. Biedert, E. Ma, I. Martinovic, and D. Song, "Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous

authentication," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 1, pp. 136–148, 2012.

[42] M. Montgomery, M., Chatterjee, P., Jenkins, J., & Roy, K. (n.d.). Touch Analysis: An Empirical Evaluation of Machine Learning Classification Algorithms on Touch Data.

[43] NYIT-LAMP Research Group, "Datasets for mobile authentication research," *NYIT-LAMP*, 2019. [Online]. Available: https://www.nyit-lamp.com/datasets/. [Accessed: 19-Jan-2025].

[44] N. Cariello, R. Eslinger, R. Gallagher, I. Kurtzer, P. Gasti, and K. S. Balagani, "Posture and body movement effects on behavioral biometrics for continuous smartphone authentication," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. xx, no. xx, Mar. 2024, pp. xx–xx. [Online]. Available: https://www.nyit-lamp.com/dataset/dataset4/.

[45] BiDAlab, "HuMIdb: Human Mobile Interaction Database," *GitHub Repository*, 2025. [Online]. Available: https://github.com/BiDAlab/HuMIdb.

[46] A. Acien, A. Morales, J. Fierrez, R. Vera-Rodriguez, and O. Delgado-Mohatar, "BeCAPTCHA: Behavioral bot detection using touchscreen and mobile sensors benchmarked on HuMIdb," *arXiv preprint*, arXiv:2005.13655, 2020. [Online]. Available: https://arxiv.org/abs/2005.13655.

[47] G. Stragapede, R. Vera-Rodriguez, R. Tolosana, A. Morales, A. Acien and G. Le Lan, "Mobile Passive Authentication through Touchscreen and Background Sensor Data," 2022 International Workshop on Biometrics and Forensics (IWBF), 2022, pp. 1-6.

[48] P. J. Zhang, C. B. Wang, and L. Ye, "A type III radio burst automatic analysis system and statistic results for a half solar cycle with Nançay Decameter Array data" *Astronomy & Astrophysics*, vol. 618, p. A165, 2018.

[49] P. Delgado-Santos, R. Tolosana, R. Guest, P. Lamb, A. Khmelnitsky, C. Coughlan, and J. Fierrez, "SwipeFormer: Transformers for mobile touchscreen biometrics," Expert Systems with Applications, vol. 237, 2024, pp. 121537.

[50] G. Stragapede, R. Vera-Rodriguez, R. Tolosana, A. Morales, J. Fierrez, J. Ortega-Garcia et al., "IJCB 2022 Mobile Behavioral Biometrics Competition (MobileB2C)," *Proc. of the 2022 International Joint Conference of Biometrics (IJCB)*, Abu Dhabi, 2022.

[51] Gündüz, G., Kaplan, M., & Taşkaya Temizel, T. (2025). Feature Distribution-Based Touch Biometrics Using CNN and Siamese Networks.Paper presented at the 2025 IEEE International Workshop on Biometrics and Forensics (IWBF), Germany.

[52] Stragapede, G., Vera-Rodriguez, R., Tolosana, R., Morales, A., Acien, A., & le Lan, G. (2022). Mobile behavioral biometrics for passive authentication. Pattern Recognition Letters, 157, 35–41. https://doi.org/10.1016/j.patrec.2022.03.014

[53] Peyré, G., & Cuturi, M. (2019). Computational Optimal Transport: With Applications to Data Science. Foundations and Trends® in Machine Learning, 11(5–6), 355–607. https://doi.org/10.1561/2200000073

APPENDICES

A Data Quality Findings

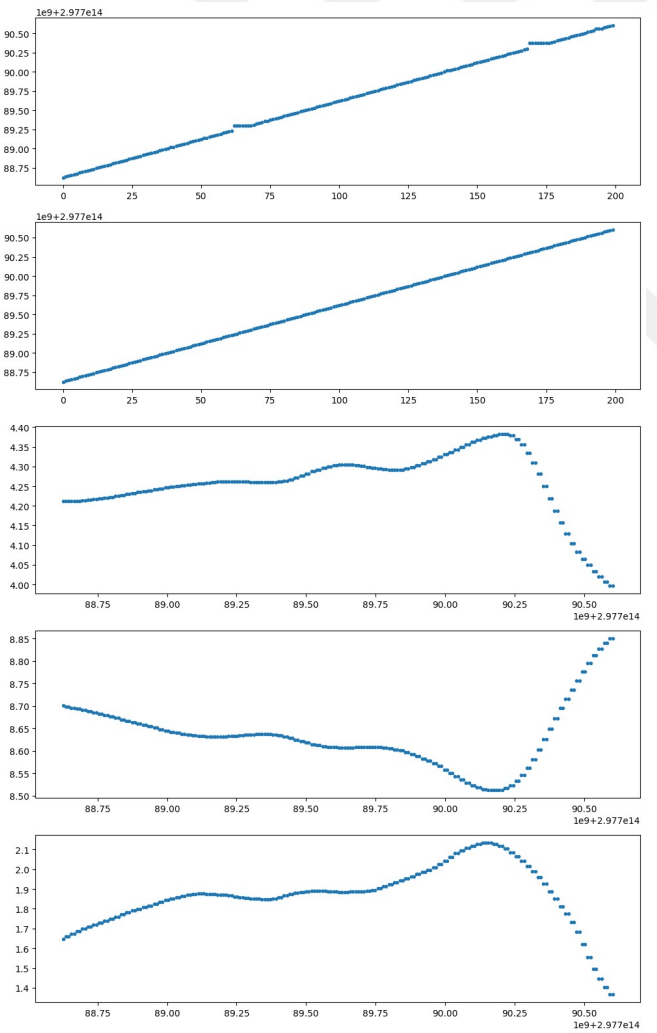


Figure A.1: Illustration of timestamp irregularities in sensor data. Linear interpolation was applied to mitigate these issues.

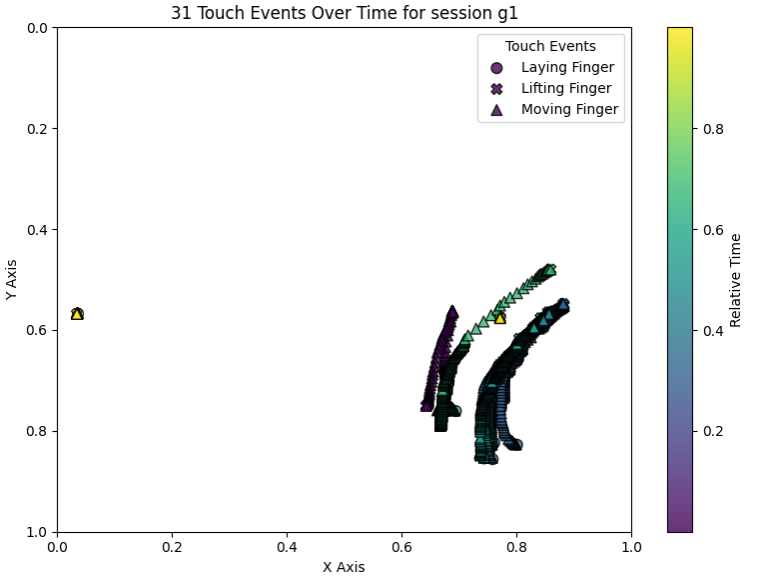


Figure A.2: Touch data plot for the user 31 in session g1

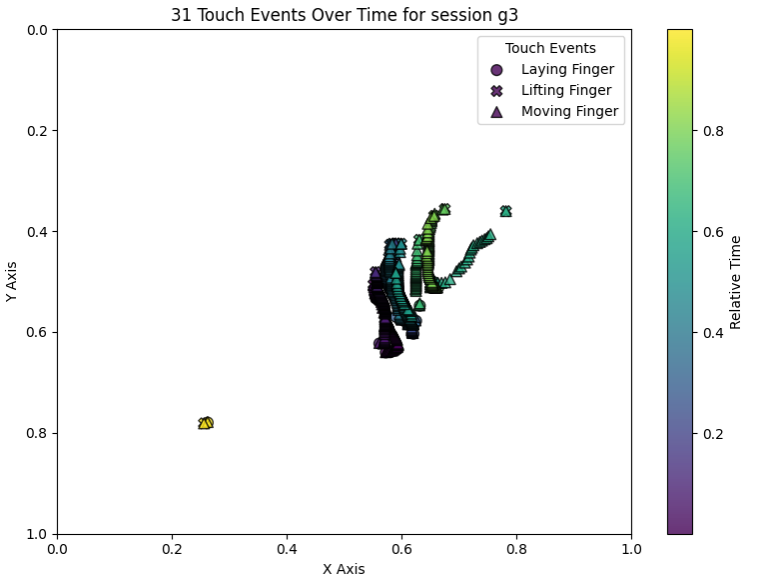


Figure A.3: Touch data plot for the user 31 in session g3

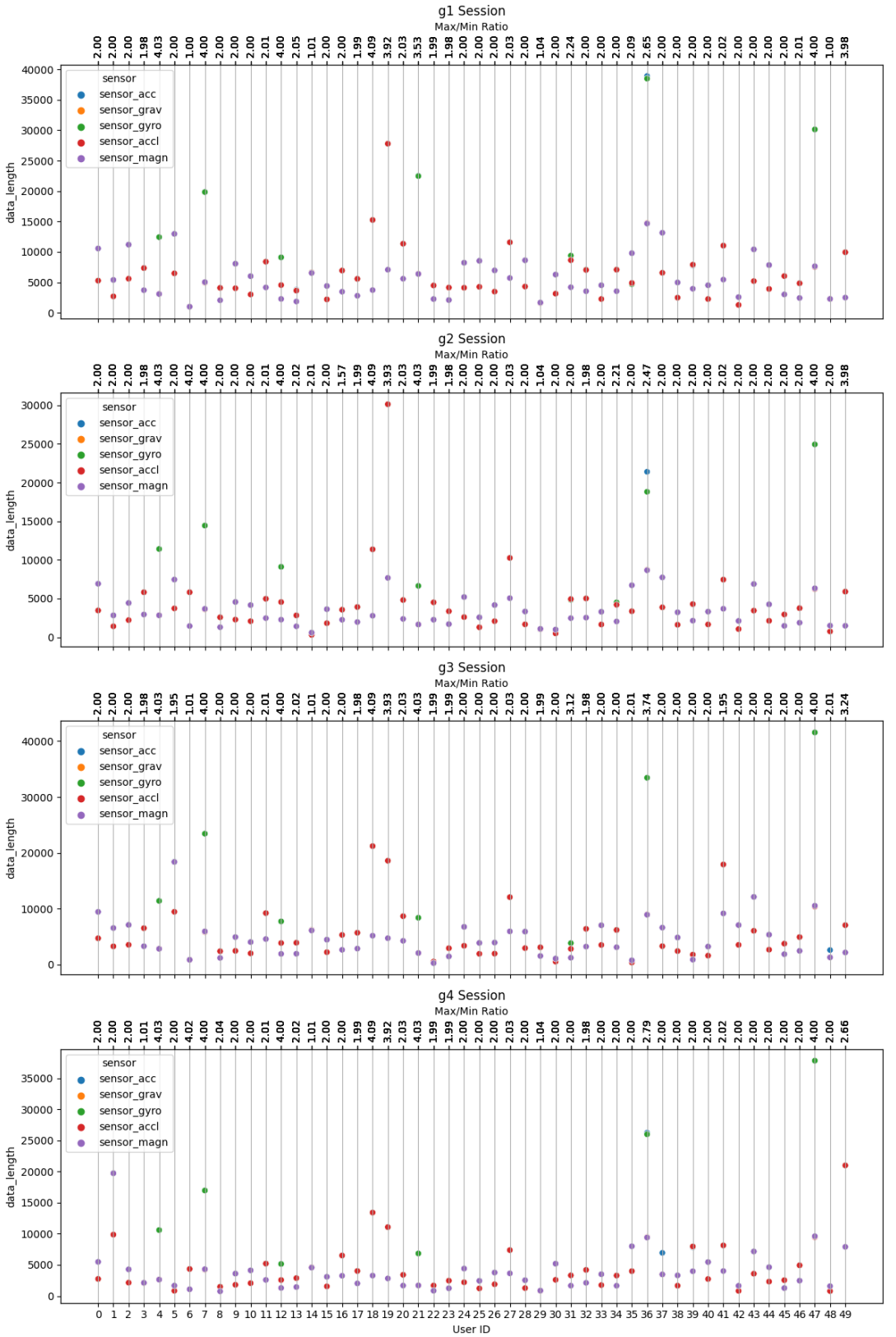


Figure A.4: Sensor Data Lengths for the users

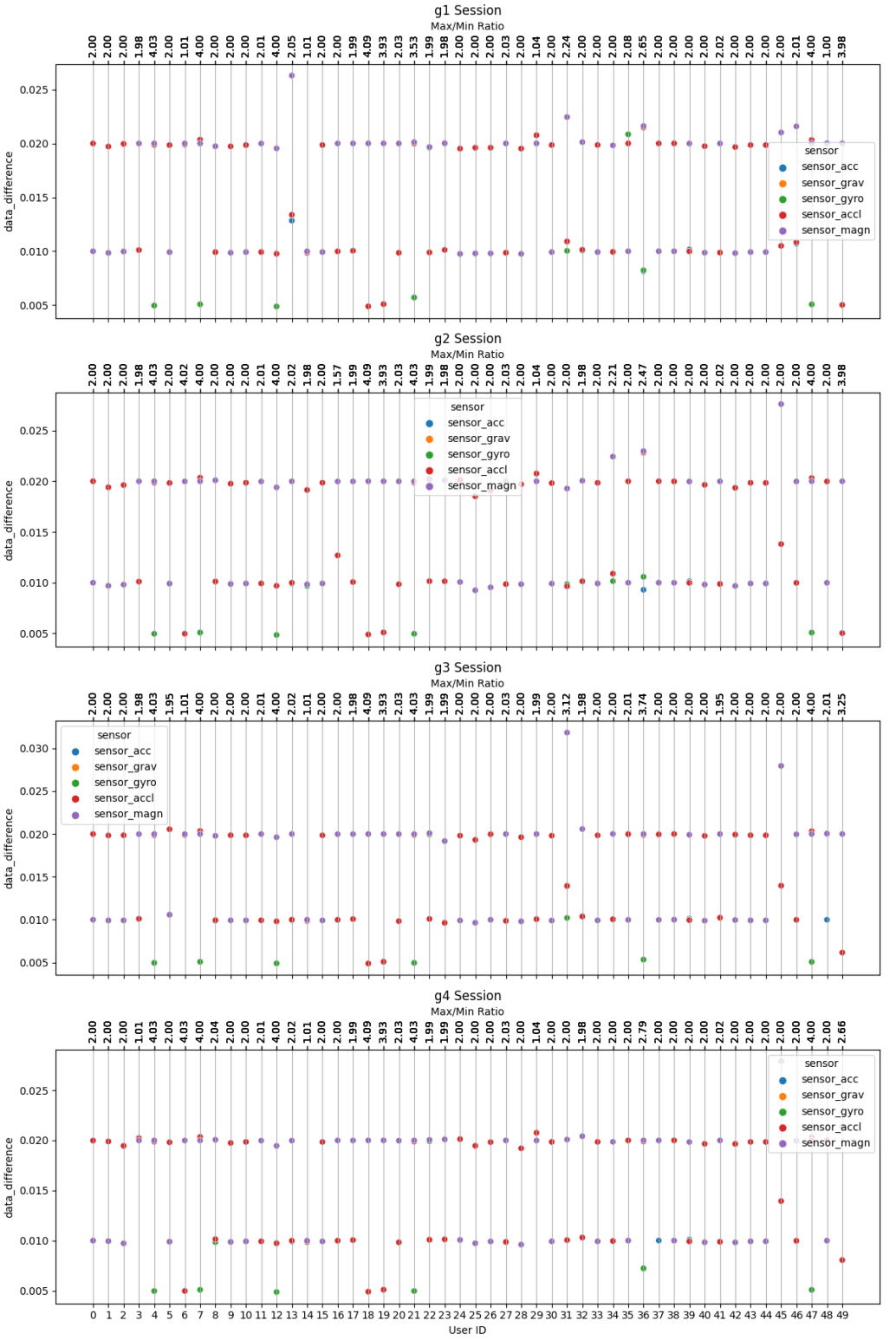


Figure A.5: Sensor Data collection rates for the users

B Preprocessing and Model Architecture

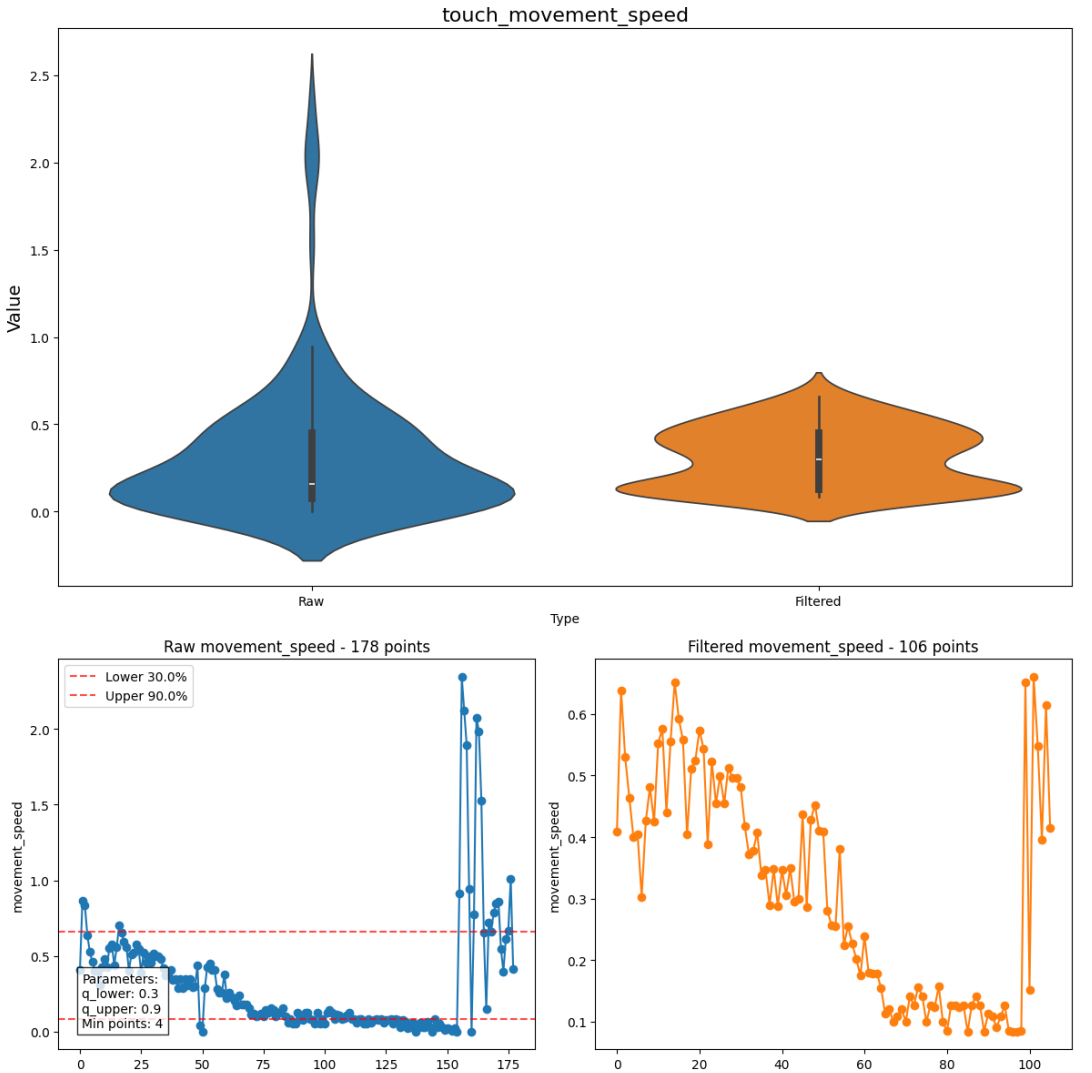
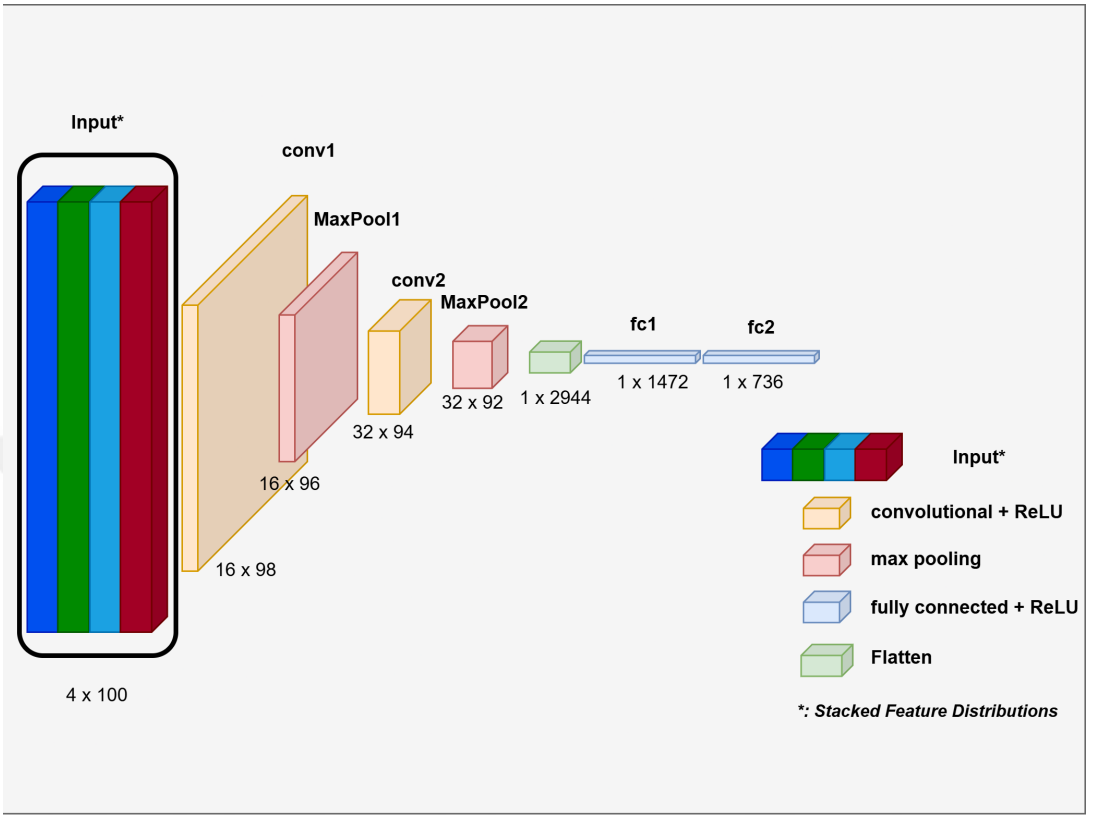
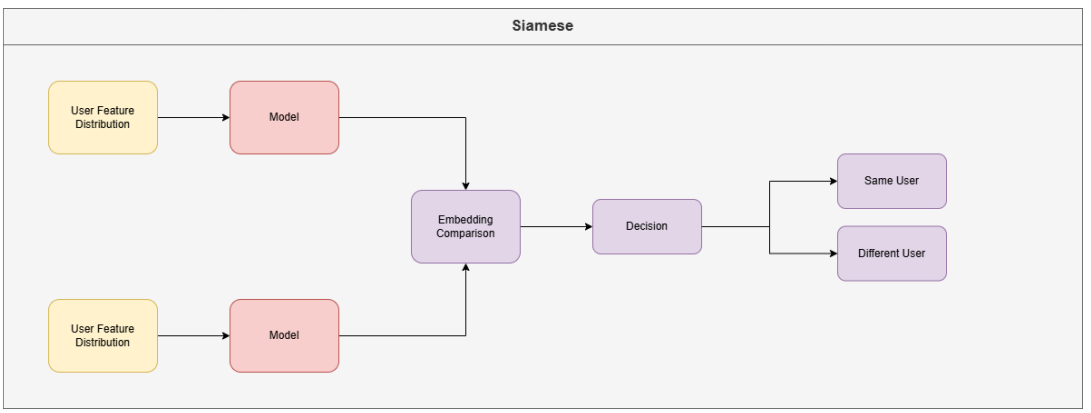


Figure B.1: Evolution of the session data after for the user 4 in session g2 for the movement speed feature



(a) The architecture of a deep learning model with convolutional layers for feature learning.



(b) Siamese network using embeddings to classify the same or different users.

Figure B.2: Overall approach for the user comparison stage.