Feature Distribution-Based Touch Biometrics Using CNN and Siamese Networks

Gürkan Gündüz
Department of Data Informatics
Middle East Technical University
Ankara, Türkiye
gurkan.gunduz@metu.edu.tr

Muratcan Kaplan
Department of Data Informatics
Middle East Technical University
Ankara, Türkiye
muratcan.kaplan@metu.edu.tr

Tuğba Taşkaya Temizel
Department of Data Informatics
Middle East Technical University
Ankara, Türkiye
ttemizel@metu.edu.tr

Abstract

Ensuring secure and user-friendly authentication is important as mobile devices increasingly handle sensitive data. Traditional methods like PINs, fingerprints, and facial recognition have privacy limitations, whereas behavioral biometrics offer implicit, continuous authentication. This study presents a touch-based authentication framework leveraging feature distribution modeling with a Convolutional Neural Network (CNN)-based Siamese network. Using Kullback-Leibler (KL) divergence, we compare touch dynamics distributions across sessions to differentiate users. To address behavioral variability, we employ adaptive bandwidth tuning in kernel density estimation (KDE) for improved probability modeling. The CNN extracts embeddings from these feature distributions, while the Siamese network assesses session similarities. Unlike traditional handcrafted approaches using summary statistics, our method preserves the full statistical structure of touch interactions, improving authentication accuracy. Experimental results demonstrate competitive Equal Error Rates (EER), underscoring the potential of distribution-driven touch biometrics for mobile authentication.

I. Introduction

Mobile devices are essential for handling sensitive information, yet traditional authentication methods—passwords, PINs, and physiological biometrics—are vulnerable to attacks such as shoulder surfing and spoofing, and rely solely on entry-point authentication, exposing devices after initial access.

Continuous Authentication (CA) addresses these limitations by continuously verifying user identity through behavioral biometrics, leveraging unique interaction patterns like scrolling and swiping [1]. This work proposes a touch-based authentication framework integrating feature distribution learning with deep learning. Unlike methods relying on summary statistics [3], we model complete probability distributions of touch biometrics to capture richer interaction patterns. Features such as acceleration, movement speed, and curvature are extracted, and Kullback-Leibler (KL) divergence [4] is used to compare user-specific distributions across sessions and optimize Kernel Density Estimation (KDE) bandwidths. This refinement improves the modeling of individual behaviors, increasing authentication accuracy.

These distribution-based features are processed through a CNN-based Siamese network [5], which learns session similarities for authentication. Unlike prior approaches based on hierarchical models [6] or information-theoretic methods [7], our method integrates KL divergence into the feature extraction process, achieving improved differentiation among users.

This study addresses the following research questions: (1) What is the effect of using distribution-based feature representation on the performance and robustness of biometric authentication models? (2) How can feature representation and optimization strategies mitigate inconsistencies and noise in biometric data to improve model performance? These questions explore the advantages of modeling entire distributions over summary statistics and strategies for reducing variability in real-world data, advancing authentication reliability. We validate our framework on the BehavePassDB dataset [8]. Our method ensures generalization across different devices and populations by maintaining feature extraction consistency across screen sizes and demographic variations, contributing to more secure and scalable mobile authentication.

II. Related Work in Behavioral Biometric Authentication

Behavioral biometrics analyze user interaction patterns, such as gestures and movements. Gesture-based methods rely on features like trajectory, velocity, and micro-movements. Antal et al. [9] demonstrated that aggregating gesture sequences improves reliability, with micro-movement features proving more discriminative than traditional touch features. However, handcrafted feature extraction in such methods limits adaptability in real-world conditions.

Research in behavioral biometrics has explored various modeling techniques, including traditional machine learning, deep learning architectures like CNNs, and statistical approaches. CNN-based Siamese networks [10] have proven effective in improving scalability and robustness in touch-based authentication by learning discriminative embeddings. Similarly, Mitra et al. [6] introduced a Bayesian hierarchical random effects model to improve error rate prediction and generalizability, demonstrating the potential of statistical modeling in biometric systems.

Recent studies show that deep learning techniques can greatly enhance gesture-based biometric authentication. For example, Wang et al. [11] propose an on-device behavioral authentication framework using deep metric learning (Siamese networks) to learn discriminative gesture patterns. Their system continuously adapts to users' behavioral changes by retraining on the device and even embeds noise in sensor data to thwart side-channel attacks [11]. Likewise, Abuhamad et al. [12] introduce *AUToSen*, a lightweight

LSTM-based continuous authentication scheme that leverages smartphone motion sensors; by analyzing short motion sequences (0.5–1 s) from the accelerometer, gyroscope, and magnetometer [12]. These deep learning approaches demonstrate improved biometric accuracy and adaptability in the face of changing user behavior and potential attacks, which closely aligns with our goals.

Our approach uses CNNs and Siamese networks to model full probability distributions, improving biometric accuracy, robustness, and adaptability while tackling domain adaptation and behavioral drift.

III. Dataset Preparation and Analysis

The BehavePassDB dataset [8] is a large-scale resource for behavioral biometrics and continuous authentication. It includes data from 51 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 diverse 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 [11].

This study utilizes touch data from the Text Reading task, capturing high-resolution time-series data 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.

A. Exploratory Data Analysis and Preprocessing

Exploratory Data Analysis (EDA) revealed inconsistencies in the collected sensor and touch data, which could affect the performance of authentication models. In accelerometer data, timestamp irregularities were identified, where certain intervals displayed constant values. These lags, likely caused by internal sensor processing delays, disrupted the temporal alignment of events, introducing artifacts that could misrepresent user behavior. To address this issue, linear interpolation was applied to ensure a continuous and consistent progression of timestamps, preserving the integrity of the time-series data.

Touch interaction data exhibited additional anomalies, particularly in swipe gestures. Some strokes displayed abrupt directional changes and irregular trajectories, deviating significantly from expected natural swiping patterns. Moreover, short strokes with very few data points lacked sufficient information to represent user behavior reliably, posing challenges for similarity-based authentication metrics.

These issues emphasize the need for robust preprocessing methods to mitigate the effects of noisy or inconsistent data. Ensuring high data quality before modeling is important, as poor-quality data can significantly impair a model's performance. Our approach addresses these challenges through filtering during feature distribution modeling. For instance, the sessions including few data points were discarded as a part of this process, as including them may distort the analysis.

B. Dataset Preparation for Model Training and Evaluation

We split the dataset into training, validation, and test sets while ensuring a balanced distribution.

1) Training Dataset: The training set utilizes three sessions (g_1, g_2, g_3) through iterative sampling, where the 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

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.

3) Test Dataset: An independent test set, separate from (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.

IV. Proposed 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.

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.
- **Acceleration**: Derived from changes in movement speed across consecutive points.
- **Curvature**: Computed using three consecutive and discrete points, based on the method proposed by [16], 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:

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

  for the parametric curve $(x_i(t), y_i(t))$ formed by the three points.
- **Cumulative Average Speed**: Captures average speed cumulatively across touch intervals, reflecting consistency in user movement.

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

A. Parameter Optimization and Data Preprocessing

To improve model performance, we first preprocess the touch data by estimating optimal feature distributions. This preprocessing directly influences model parameters by providing more reliable input representations. We employ KDE to characterize user touch patterns:

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

where x represents the point at which we want to estimate the density (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, we determine optimal parameters through an iterative process that maximizes user discrimination capability. We first defined the following preprocessing variables for each feature vector:

- *Minimum required data points* (τ_{min}) is 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

We performed a Divergence-Based KDE Parameter Optimization approach to select the optimum values for the aforementioned preprocessing parameters. For each feature, we construct a similarity matrix using 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 our parameter selection, we fixed the bandwidth parameter to 0.9 for KDE estimation during this step, as this parameter has been used as a default parameter across several studies [14]. For each parameter combination (τ, Q_{min}, Q_{max}), we compute pairwise KL-divergences between all user pairs, creating a feature based similarity matrix M where $M_{i,j}$ represents the KL-divergence score between distributions of users i and j. We evaluate parameter combination on the training data by treating this problem as a binary classification problem, with labels 0 and 1 for same-user and different-user pairs respectively (see Algorithm 1).

The parameter combination that maximizes the ROC-AUC score is selected. The optimization process can be similarly applied for EER score. This step ensures that we select parameters that maximize the discriminative power of our 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.

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

B. User and Session 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 is kept if it meets requirements for every feature and session. The users who do not meet this criteria are excluded from the analysis to maintain data integrity and consistency (see Algorithm 2).

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

C. Bandwidth Selection and Distribution Generation

1) User-Specific Bandwidth Optimization: In this paper, we argue that the bandwidth parameter significantly influences the quality of the generated distributions and, consequently, the discriminative power of the resulting authentication system. In the current literature, this bandwidth has been selected as fixed such as in [14]. To determine optimal bandwidth values for each user-feature combination, we developed

a novel approach inspired by triplet loss optimization. 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 the distribution generated with bandwidth b.

2) Distribution Comparison Metric: 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.

3) Session-Feature Bandwidth Optimization: The bandwidth selection process makes use of previously obtained user-specific bandwidths to optimize the distribution generation. For each session-feature pair, instead of exploring an arbitrary range of bandwidth values, we iterate through bandwidths that were found to be optimal for individual users in the prior optimization step. This approach ensures that we select bandwidths that have already demonstrated effectiveness in capturing user-specific characteristics.

Algorithm 4 Session-Feature Bandwidth Optimization Process

```
for each session-feature pair (s, f) do
    Initialize minimum divergence d_min ← ∞
    Let B_prev be the set of previously obtained user-specific bandwidths
    for each bandwidth value b ∈ 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} ← b
            d_min ← D_KL(P_b||P_ref)
        end if
    end for
    Log optimal bandwidth b_{s,f} and minimum divergence d_min
end for
```

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

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

$$D_{final}(u, f) = \mathcal{D}(X_{u,f}, b_{opt}(u, f)) \tag{5}$$

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

D. Model Architecture and Training

We used a CNN architecture to process user touch behavior patterns based on optimized feature distributions. The network transforms the distribution data into embeddings that capture essential user characteristics. We implemented a triplet loss framework 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 between samples from the same user while maximizing distances between different users' samples.

Our network architecture consists of two sequential convolutional blocks processing 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 dimensionality through shapes 2944, 1472, and finally 736, forming our 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 Euclidean distance between embeddings:

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

This architecture effectively learns discriminative user-specific embeddings while maintaining computational efficiency for mobile deployment.

E. Model Evaluation with Unseen Data

The test set consists of users with no identifiable links to the training or validation data. To determine the optimal bandwidth for each test user, we employed a similarity-based approach. Each user's data was compared against a set of candidate bandwidths derived from the training dataset. The bandwidth that minimized the KL divergence between the test and training distributions was selected. This ensured that the most representative bandwidth was used for feature extraction in unseen test users. While the matched distribution may not correspond to the exact user, our goal was to find an approximate bandwidth that best represents the selected user.

V. Model Comparison

We adopted a baseline model inspired by TouchAnalytics [3], [15], utilizing handcrafted statistical features that capture spatial, temporal, velocity, acceleration, and pressure-based characteristics of touch interactions. The model extracts 20+ features, including stroke duration, trajectory length, inter-stroke time, velocity percentiles (20%, 50%, 80%), acceleration percentiles, pressure dynamics, phone and finger orientation, and path-based metrics such as deviation from the end-to-end line and directionality.

Authentication is formulated as a binary classification task using XGBoost, where feature distributions from two user

samples are compared to determine if they originate from the same individual. The model processes spatial (e.g., start/stop coordinates, trajectory length), kinematic (e.g., velocity, acceleration percentiles), and pressure-based (e.g., mid-stroke pressure, pressure change) features to assess user-specific touch patterns.

To evaluate performance, we employ ROC AUC, Equal Error Rate (EER), Accuracy, Precision, and F1 Score, ensuring a comprehensive assessment of authentication effectiveness across varied user behaviors.

VI. Experimental Setup

We executed the Algorithm 1 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 steps of 2. We found the values in Table I as best parameters using our training dataset. After applying Algorithm 2, the number of eligible pairs decreased from 120 to 96. During the threshold optimization process, ROC threshold (1.58) was selected based on maximized difference between TPR and FPR. On the other hand, EER threshold (1.39) was selected on the point where FPR equals the FNR.

TABLE I: 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

VII. Results

We compared our approach, which utilizes user-specific bandwidth selection (referred to as the *optimized approach*), with a baseline XGBoost model. The evaluated user-specific bandwidths and their corresponding performance metrics are presented in Table II. Additionally, we developed an alternative model using a fixed bandwidth of 0.9, following [14], 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.

Our methodology demonstrated significant improvement over the XGBoost baseline across multiple performance metrics, as summarized in Table III. It achieved an ROC AUC score of 0.6134 (compared to baseline 0.4924) and improved precision from 0.2463 to 0.4286. The EER threshold selection strategy (OET) proved more effective than ROC thresholding (ORT), yielding higher balanced accuracy (0.6409 vs. 0.6025) and MCC scores (0.2674 vs. 0.1991). The *optimized approach* maintained consistent recall rates of 0.7742 across both threshold types, whereas the *non-optimized approach* showed limited recall performance (0.0968 ORT, 0.0645 OET).

The results indicate that while the *non-optimized approach* achieves higher accuracy, it performs worse in key biometric authentication metrics such as Equal Error Rate (EER) and ROC AUC, suggesting potential overfitting to specific patterns. The *optimized approach*, particularly under the EER threshold, shows improved performance by reducing both false acceptance and false rejection rates. The Detection Error Tradeoff (DET) curve further supports this, demonstrating that the optimized model maintains a lower false negative rate across different false positive rates.

TABLE II: Comparison of Optimized (O) and Non-Optimized (NO) Model Performances

Metrics	**Validation**		**Test**	
	O	NO	O	NO
ROC AUC	**0.7573**	0.6849	**0.6134**	0.5667
EER	**0.3258**	0.3623	0.4489	**0.4251**
ROC Threshold (ORT) Performance				
Accuracy	**0.5788**	0.5693	0.5417	**0.6458**
Precision	**0.1102**	0.1011	**0.3934**	0.3333
Recall	**0.8116**	0.7464	**0.7742**	0.0968
F1 Score	**0.1941**	0.1780	**0.5217**	0.1500
Balanced Accuracy	**0.6874**	0.6519	**0.6025**	0.5022
MCC	**0.1821**	0.1475	**0.1991**	0.0072
EER Threshold (OET) Performance				
Accuracy	**0.6744**	0.6377	0.5938	**0.6354**
Precision	**0.1213**	0.1050	**0.4286**	0.2500
Recall	**0.6739**	0.6377	**0.7742**	0.0645
F1 Score	**0.2055**	0.1803	**0.5517**	0.1026
Balanced Accuracy	**0.6742**	0.6377	**0.6409**	0.4861
MCC	**0.1771**	0.1374	**0.2674**	-0.0470

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

Measure	**Baseline**	**OET**	**ORT**
ROC AUC	0.4924	**0.6134**	**0.6134**
EER	0.4802	**0.4489**	**0.4489**
Accuracy	0.5310	**0.5938**	0.5417
Precision	0.2463	**0.4286**	0.3934
F1 Score	0.4238	**0.5517**	0.5217
Balanced Accuracy	0.5158	**0.6409**	0.6025
MCC	0.1843	**0.2674**	0.1991

The Wilcoxon Signed Rank test is used to test whether each user's performance differences on the test dataset between the two approaches are statistically significant. 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.

For the acceleration feature (Figure 2), the *optimized approach with bandwidth* ($b_{opt} = 0.05$) is able to 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.

VIII. Conclusion and Discussion

Our approach outperforms both the baseline and *non-optimized approach*, confirming the effectiveness of complete behavioral distribution modeling. While our feature-specific

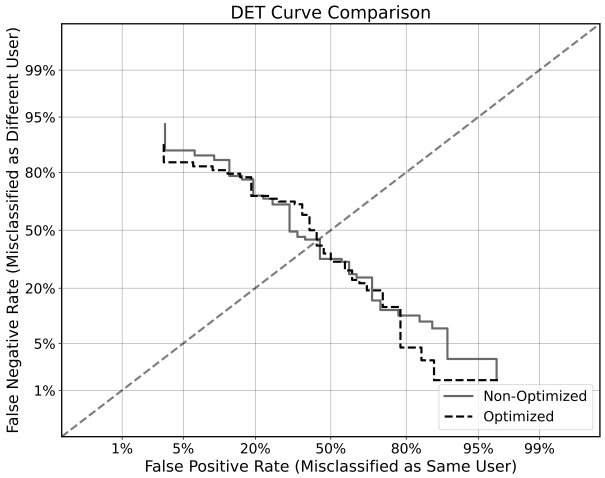

Fig. 1: DET Curve Comparison of Non-Optimized and Optimized Models.

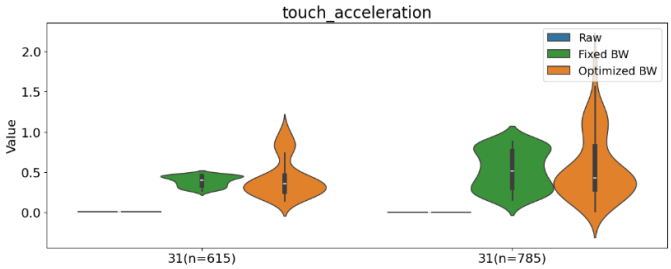

Fig. 2: Acceleration feature distributions for User 31 with raw and processed data. Parameters: minimum points=4, lower quantile=0.25, upper quantile=0.85, fixed bandwidth=0.9, optimized bandwidth=0.05, bandwidth range=0-1.

optimization showed promise with a 0.7742 recall in cross-session authentication, EER rates above 0.4 highlight ongoing challenges in touch interaction variability. Similar performances were also observed in the literature. The study of [8] achieved a 0.67 AUC score using an LSTM-based model, while the study of [18] reported a 0.732 AUC score with an LSTM-based benchmark. As demonstrated by [1] and [17], integrating additional sensor modalities could improve authentication robustness. Future work should explore symmetric measures such as Jensen-Shannon Divergence to address the KL divergence limitations, while developing computationally efficient optimization techniques for real-time processing. The practical implications suggest prioritizing adaptive preprocessing pipelines and robust temporal adaptation mechanisms for long-term deployment stability, as stated in [19]. Considering the research questions, distribution-based feature representation improves authentication performance by capturing complete behavioral patterns rather than just statistical summaries. In addition, methodology overcomes the data inconsistencies through adaptive bandwidth selection, quantile-based outlier removal, and minimum data point thresholds.

Acknowledgement: This work was supported by TÜBİTAK 2214-A.

REFERENCES

[1] 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.

[2] M. Mohamed and N. Saxena, "Gametrics: Towards Attack-Resilient Behavioral Authentication with Simple Cognitive Games," *Annual Computer Security Applications Conference (ACSAC)*, Los Angeles, CA, USA, pp. 195-206, 2016. DOI: 10.1145/2991079.2991096.

[3] 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.

[4] F. Perez-Cruz, "Kullback-Leibler divergence estimation of continuous distributions," in Proceedings of the 2008 IEEE International Symposium on Information Theory, Toronto, ON, Canada, 2008, pp. 1666-1670.

[5] S. Chopra, R. Hadsell and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), San Diego, CA, USA, 2005, pp. 539-546.

[6] S. Mitra, M. Savvides, and A. Brockwell, "Statistical performance evaluation of biometric authentication systems using random effects models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 517–530, 2007.

[7] J. Bhatnagar and A. Kumar, "On estimating performance indices for biometric identification," *Pattern Recognition*, vol. 42, no. 9, pp. 1803–1815, 2009. doi: https://doi.org/10.1016/j.patcog.2008.10.004.

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

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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/10.1145/3577923.358363710.1145/3577923.3583637.

[14] 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.

[15] 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

[16] 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.

[17] L. Yuan, J. Andrews, H. Mu, A. Vakil, R. Ewing, E. Blasch, and J. Li, "Interpretable Passive Multi-Modal Sensor Fusion for Human Identification and Activity Recognition," *Sensors*, vol. 22, no. 15, p. 5787, 2022. https://doi.org/10.3390/s22155787.

[18] 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.2024.10188810.1016/j.pmcj.2024.101888.

[19] 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.