Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Holistic and Local Representation Learning for Online Signature Verification

João Pedro Felix de Almeida

Dissertação apresentada como requisito parcial para
conclusão do Mestrado em Informática

Orientador
Prof. Dr. Pedro Garcia Freitas

Coorientador
Prof. Dr. Bruno Luiggi Macchiavello Espinoza

Brasília
2025

Ficha Catalográfica de Teses e Dissertações

Está página existe apenas para indicar onde a ficha catalográfica gerada para dissertações de mestrado e teses de doutorado defendidas na UnB. A Biblioteca Central é responsável pela ficha, mais informações nos sítios:

http://www.bce.unb.br
http://www.bce.unb.br/elaboracao-de-fichas-catalograficas-de-teses-e-dissertacoes

Esta página não deve ser inclusa na versão final do texto.

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Holistic and Local Representation Learning for Online Signature Verification

João Pedro Felix de Almeida

Dissertação apresentada como requisito parcial para
conclusão do Mestrado em Informática

Prof. Dr. Pedro Garcia Freitas (Orientador)
CIC/UnB

Prof. Dr. Byron Leite Dantas Bezerra
Universidade de Pernambuco

Prof. Dr. Flavio Barros Vidal
Universidade de Brasília

Prof.a Dr.a Cláudia Nalon
Coordenadora do Programa de Pós-graduação em Informática

Brasília, 21 de Julho de 2025

Dedicatória

Dedico este trabalho aos meus pais e à minha irmã, que sempre me incentivaram a estudar. Também dedico aos meus amigos, cujo apoio foi essencial ao longo de toda a jornada.

Agradecimentos

Em primeiro lugar, agradeço a Deus pela saúde e pela oportunidade de estudar. Também sou grato a todos os professores que contribuíram para a minha formação, desde os primeiros passos na educação básica até o encerramento da minha trajetória no mestrado. Em especial, sou profundamente grato aos meus orientadores, Dr. Pedro Garcia e Dr. Bruno Machiavello, por toda a dedicação e tempo investidos em mim e em nossa pesquisa ao longo dos últimos anos. Levarei comigo para a vida todas as lições aprendidas, das mais técnicas às mais triviais e pessoais, que surgiram em meio às pautas de nossas reuniões.

Também gostaria de agradecer ao Lucas, meu coautor nesta pesquisa, que sempre se mostrou disposto a ouvir minhas inquietações e, muitas vezes, a me lembrar da importância do descanso. Além disso, sua contribuição técnica e suas críticas construtivas foram fundamentais para o desenvolvimento deste trabalho. Em grande parte graças a ele, posso dizer que estou satisfeito com o resultado final.

Partes deste trabalho, incluindo texto e algumas figuras, foram reproduzidas com a permissão da Springer Nature e são baseados em um artigo aceito para publicação na ICDAR 2025, atualmente em processo de publicação.

Por fim, é necessário destacar que este trabalho só foi possível graças à disponibilização do conjunto de dados DeepSignDB, gentilmente fornecido pela Universidad Autónoma de Madrid (UAM-ES).

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES), por meio do Acesso ao Portal de Periódicos.

Acknowledgement

First of all, I thank God for my health and for the opportunity to study. I am also grateful to all the teachers who contributed to my education, from my first steps in basic education to the completion of my master's degree. In particular, I am deeply grateful to my advisors, Dr. Pedro Garcia and Dr. Bruno Machiavello, for all the dedication and time they invested in me and in our research over the past years. I will carry with me for life all the lessons learned — from the most technical to the most trivial and personal — that emerged during the discussions in our meetings.

I would also like to thank Lucas, my co-author in this research, who was always willing to listen to my concerns and often reminded me of the importance of rest. Furthermore, his technical contributions and constructive feedback were essential for the development of this work. Largely thanks to him, I can say that I am satisfied with the final result.

Parts of this work, including text and some figures, were reproduced with permission from Springer Nature and are based on a paper accepted for publication at ICDAR 2025, currently in the publication process.

Finally, it is important to highlight that this work was only made possible thanks to the availability of the DeepSignDB dataset, kindly provided by the Universidad Autónoma de Madrid (UAM-ES).

This work was carried out with the support of the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) – Brazil, through access to the CAPES Journals Portal.

Aprendizagem de Representações Holísticas e Locais para a Verificação de Assinaturas Online

Resumo

Neste trabalho, abordamos o problema de minimizar a taxa de Equal Error Rate (EER) global em um sistema de verificação de assinaturas online independente de escritor, decompondo-a em duas componentes: a taxa de erro de separação e a taxa de erro de alinhamento. A taxa de separação surge da falta de uma separação correta entre assinaturas genuínas e falsificadas, enquanto a de alinhamento resulta da incapacidade de um único limiar global capturar de forma eficaz as separações específicas de cada escritor. Enquanto as abordagens tradicionais da literatura focam exclusivamente em aumentar a separação entre assinaturas genuínas e falsificadas, essa decomposição nos permite endereçar o problema também do ponto de vista da falta de alinhamento entre os limiares específicos de cada usuário.

Para isso, propomos HoLoSig, um novo arcabouço que integra duas populares representações profundas de assinaturas através de uma arquitetura convolucional 1D compartilhada que se bifurca em dois ramos especializados. Em um dos ramos, utilizamos a *Triplet Loss* com *Soft-DTW* para aprender representações locais de comprimento variável, cujas pontuações de dissimilaridade são deslocadas para uma região comum com a ajuda da Discrepância Média Máxima, a fim de melhorar o desempenho do sistema ao usar um limiar global. No outro ramo, empregamos como função de perda a expansão polinomial da entropia cruzada com correção no primeiro polinômio para aprender representações holísticas de comprimento fixo, que são usadas para reforçar ainda mais a separação criada pelo ramo de representações locais.

Para avaliar o método proposto nós conduzimos diversos experimentos de ablação que confirmam a redução tanto no erro de separação quanto de alinhamento e concluímos

que a Triplet-MMD desempenha papel fundamental na redução do último. Em seguida, estendemos a análise ao substituir a MMD por outras medidas de divergência estatística e comprovamos que elas alcançam resultados comparáveis—demonstrando que o erro de alinhamento pode ser reduzido ao garantir que as pontuações de dissimilaridade do sistema sigam uma distribuição consistente.

Como um arcabouço independente de escritor, HoLoSig não possui dependência de usuários e é portanto capaz de verificar assinaturas de escritores não vistos durante o treinamento sem a necessidade de qualquer adaptação ou ajuste-fino. No entanto, HoLoSig não está livre de vieses relacionados aos dispositivos de captura e protocolo de aquisição de assinaturas. Portanto, para avaliar a robustez do modelo a essas particularidades nós realizamos diversos experimentos onde o treinamento é realizado com dados oriundos de protocolos de aquisição e dispositivos de captura diferentes dos utilizados para medir o erro do sistema. Nossos resultados são promissores, pois eles não apenas mostram forte capacidade de generalização mas também que HoLoSig consegue superar métodos que figuram no estado da arte mesmo se treinada com apenas metade dos dados utilizados por eles.

HoLoSig foi idealizada para verificação de assinaturas online realizadas com canetas (*stylus*). No entanto, nós estendemos sua arquitetura para endereçar o problema de verificação de assinaturas online realizadas com dedo. Um dos maiores desafios na verificação de assinaturas feitas com dedo é o tamanho limitado dos conjuntos de dados disponíveis para pesquisa comparado com aqueles onde as assinaturas foram feitas com caneta, o que frequentemente inviabiliza o treinamento do zero. Para endereçar isso, nós adicionamos um ramo classificador de domínio na arquitetura de HoLoSig, transformando-a efetivamente em uma Rede Neural de Adaptação de Domínio. A arquitetura HoLoDANN resultante nos permite tirar proveito tanto das assinaturas feitas com caneta quanto das assinaturas feitas com dedo para melhor abordar o problema de assinaturas feitas com dedo.

HoLoSig e HoLoDANN alcançam resultados de ponta no DeepSignDB, o maior conjunto de verificação de assinaturas online atualmente. Os resultados (EER) de HoLoSig no cenário de assinaturas feitas com caneta contra falsificações habilidosas e aleatórias são: 1,73% (4vs1 habilidosas), 3,29% (1vs1 habilidosas), 0,43% (4vs1 aleatórias) e 0,89% (1vs1 aleatórias). Os resultados (EER) de HoLoDANN no cenário de assinaturas feitas com dedo contra falsificações habilidosas e aleatórias são: 5,65% (4vs1 habilidosas), 9.99% (1vs1 habilidosas), 0,55% (4vs1 aleatórias) e 1,78% (1vs1 aleatórias).

Palavras-chave: Verificação de Assinaturas Online, Aprendizado de Representações, Aprendizado de Métricas, Discrepância Média Máxima

Abstract

In this work, we address the problem of minimizing the Equal Error Rate (EER) in a Writer-Independent (WI) Online Signature Verification (OSV) system by decomposing it into two components: the separation error and the alignment error. The separation error arises from the lack of proper distinction between genuine and forged signatures, while the alignment error arises from the inability of a single global threshold to effectively capture the writer-specific separations. While traditional approaches in the literature focus solely on increasing the separation between genuine and forged signatures, this decomposition also enables us to tackle the problem from the perspective of writer-specific threshold misalignment.

To this end, we propose HoLoSig, a novel framework that integrates two popular deep signature representations via a shared 1D convolutional backbone, which bifurcates into two specialized branches. On one branch, we employ Triplet Loss with Soft-DTW to learn variable-length local representations, whose dissimilarity scores are shifted to a common region with the help of Maximum Mean Discrepancy (MMD), improving the system's performance when using a global threshold. On the other branch, we use Poly-1 Cross Entropy Loss to learn fixed-length holistic representations, which further enhance the separation achieved by the local representation branch.

To evaluate the proposed method, we conduct several ablation experiments that confirm the reduction of both separation and alignment errors, and show that Triplet-MMD plays a significant role in reducing the latter. We further extend Triplet-MMD by replacing MMD with other divergence metrics, and find that they achieve comparable results—demonstrating that alignment error can be reduced by ensuring the dissimilarity scores of the OSV system follow a consistent distribution.

As a WI framework, HoLoSig has no dependency on writers and is therefore capable of verifying signatures from unseen writers without the need for any adaptation or fine-tuning. However, HoLoSig is not free from biases related to acquisition devices and protocols. To assess the model's robustness to these particularities, we conduct several experiments in which training is performed using data acquired with different devices and protocols from those used to evaluate the system's EER. Our results are promising, as

they show not only strong generalization capability but also that HoLoSig can outperform state-of-the-art methods while using as little as half the training data they use.

HoLoSig is designed for stylus-written OSV. However, we extend its architecture to address the problem of finger-written OSV. One of the major challenges in finger-written OSV is the limited size of available datasets compared to those for stylus-written signatures, which often makes training from scratch difficult. To address this, we add a domain classifier branch to HoLoSig, effectively transforming it into a Domain Adaptation Neural Network (DANN). The resulting HoLoDANN architecture allows us to leverage knowledge from both stylus- and finger-written signatures to better tackle the finger-written OSV problem.

HoLoSig and HoLoDANN achieve state-of-the-art results on DeepSignDB, the largest OSV dataset to date. HoLoSig's EER results in the stylus scenario against skilled and random forgeries are: 1.73% (4vs1 skilled), 3.29% (1vs1 skilled), 0.43% (4vs1 random), and 0.89% (1vs1 random). HoLoDANN's EER results in the finger scenario against skilled and random forgeries are: 5.65% (4vs1 skilled), 9.99% (1vs1 skilled), 0.55% (4vs1 random), and 1.78% (1vs1 random).

Keywords: Online Signature Verification, Representation Learning, Metric Learning, Maximum Mean Discrepancy

Contents

List of Figures

List of Tables

Acronyms

BCE Binary Cross-Entropy.

CDF Cumulative Distribution Function.

CNN Convolutional Neural Network.

CRAN Convolutional Recurrent Adaptative Network.

DANN Domain Adaptation Neural Network.

DTW Dynamic Time Warping.

EER Equal Error Rate.

FAR False Accept Rate.

FRR False Reject Rate.

GARU Gated Auto Regressive Unit.

GRU Gated Recurrent Unit.

LODO Leave One Dataset Out.

LSTM Long Short Term Memory.

MMD Maximum Mean Discrepancy.

OSV Online Signature Verification.

RBF Radial Basis Function.

RKHS Reproducing Kernel Hilbert Space.

RNN Recurrent Neural Network.

SP Selective Pooling.

SVM Support Vector Machine.

WD Writer-Dependent.

WI Writer-Independent.

Chapter 1

Introduction

Biometrics is a scientific discipline focused on the verification and identification of individuals based on their physical or behavioral characteristics and it results from the premise that certain human traits are sufficiently distinguishable among different individuals [1]. As summarized in some recent works [2, 3], over the years several biometric traits were explored in order to build biometric systems, with particular success in physiological traits such as fingerprints, face and iris, and also behavioral traits, such as voice, gait and handwritten signatures.

A typical biometric system operates in one of two modes: verification or identification. A verification system aims to determine whether an individual is truly who they claim to be, and usually achieves this by comparing their biometric trait against the enrolled templates for that identity. An identification system, on the other hand, aims to determine whether someone is enrolled in a database and, if it is, who this person is. For this purpose, an identification system often compares the biometric input with all enrolled templates in the database. This work addresses handwritten signatures as the chosen biometric trait within the context of a verification system.

The effectiveness of a biometric system—whether operating in identification or verification mode— lies in its robustness against potential attacks. The nature of these attacks often varies depending on whether the system is positive, where it is expected that the user seeks to be accepted (e.g., gaining access to a bank account), or negative, where it is expected that the user aims to avoid acceptance (e.g., evading identification by authorities) [4]. In this work, we define an impostor as an user attempting to fool the system.

Therefore, the success of a biometric system is highly dependent on the specific application objectives and the inherent characteristics of the biometric trait being used. While physiological traits such as fingerprints can be used both in positive and negative systems (since they remain constant under ordinary conditions), behavioral traits such

as handwritten signatures are in general not suited for negative systems, since one can simply alter her behavior on purpose in order to fool the system. Furthermore, beyond the aforementioned aspect, we also consider the following seven criteria presented in [5] to analyze the applicability of a biometric trait:

1. Cooperative versus non-cooperative
2. Overt versus covert
3. Habituated versus non-habituated
4. Public versus private
5. Attended versus non-attended
6. Standard versus non-standard operating environment
7. Open versus closed

Cooperative versus non-cooperative refers to whether an impostor interacting with the system actively attempts to be accepted or not. Given that handwritten signatures are predominantly used in positive systems, where users seek authentication, signature-based biometric systems are typically designed for cooperative environments.

Overt versus covert refers to whether the user is aware she is being recognized by biometrics. Thus, handwritten signatures are necessarily classified as overt.

Habituated versus non habituated refers to how often the users interact with the system. In the context of handwritten signatures, this frequency can vary significantly from user to user and is one of the main aspects that makes signature verification a hard problem, as further discussed in section 1.1.

Public versus private, which refer whether the users are customers (public) or employees (private). In a private environment, for example, is possible to train the staff in order they get habituated with the system, thus enhancing the quality of the acquisition process, which can lead to lower error rates. Naturally, handwritten signatures can be used in both contexts.

Attended versus non-attended is related whether or not the process of data acquisition is guided by a human. Since signature-based systems are typically used in cooperative settings, data collection can occur in either attended or non-attended modes. This contrasts with non-cooperative systems, where data acquisition generally requires human supervision to ensure the quality and authenticity of the captured biometric data.

Standard versus non-standard environment refers whether or not the acquisition occurs in controlled conditions. Acquisition settings can range from well-controlled environments — such as offices where signatures are written on a stable desk surface — to more

unpredictable scenarios, like signing while standing and using a clipboard as a support surface.

Open versus closed, which refer whether or not the data used by the system will be shared with different systems. While not related to a specific biometric trait, the lack of a common acquisition protocol can lead to hardships during the design phase.

While these characteristics may turn handwritten signatures less suitable for certain applications—particularly within negative systems where users are expected to attempt to evade recognition—they also make signatures highly appropriate for other domains. As a necessarily overt biometric, the act of providing a handwritten signature inherently requires the user's conscious cooperation, i.e., the individual is fully aware that they are performing a biometric action. This property makes handwritten signatures especially well-suited for formal and legal agreements, where user awareness is essential. This stands in contrast to biometric traits that can operate in covert scenarios, such as face recognition, where biometric data can be captured passively and often without the subject's knowledge or consent.

Furthermore, the ability to acquire handwritten signatures in non-attended settings and under non-standard environmental conditions contributes to the simplicity and practicality of signature acquisition. This ease of collection facilitates straightforward deployment in a wide range of real-world applications. All of this combined with its widespread legal and social acceptance makes signature verification one of the most common forms of personal authentication [6].

As a form of behavioral biometric, handwritten signatures result from human movements and are thus influenced by various factors such as writing speed and emotional state [7]. These factors cause signatures from the same writer to exhibit dissimilarities [8], a phenomenon commonly referred to as intra-class variability.

Intra-class variability contrasts with inter-class variability, which arises from the high degree of similarity between signatures from different writers. Therefore, a signature verification system must be capable of accepting signatures from the same writer despite their variations while rejecting those from different writers, despite their similarities. This task becomes even more challenging when the system is susceptible to attacks that aim to reduce the inter-class variability between signatures from different writers.

There are two main types of attacks against a signature verification system. A skilled forgery attack involves an attempt to deceive the system by replicating a genuine signature. In this scenario, the impostor has access to one or more genuine signatures and possibly additional information about the writing process, such as a video recording. In contrast, a random forgery attack occurs when the impostor attempts to replicate a genuine signature without any prior knowledge of its appearance or characteristics. Random

forgery attacks are typically simulated by using genuine signatures from different writers as forgeries. The combination of high intra-class variability among genuine signatures and low inter-class variability—particularly in cases of skilled forgeries—makes signature verification an ongoing challenge, despite significant advancements in recent decades, as summarized in several works [6, 9, 10, 2].

Signature verification can be categorized into two main approaches based on how signatures are stored [8]: (1) *Offline* (or static) verification, where the signature is captured as an image, typically written using pen and paper and later digitized through scanning. (2) *Online* (or dynamic) verification, where the signature is recorded using a device that captures temporal information about the writing process, such as variations in spatial coordinates and pressure applied to the device's surface. Additionally, an online signature can be written with either a stylus or fingers according to the acquisition device. This work focuses on online (dynamic) signature verification. Figure 1.1 shows an example of four online signatures, including three genuine samples and a skilled forgery.

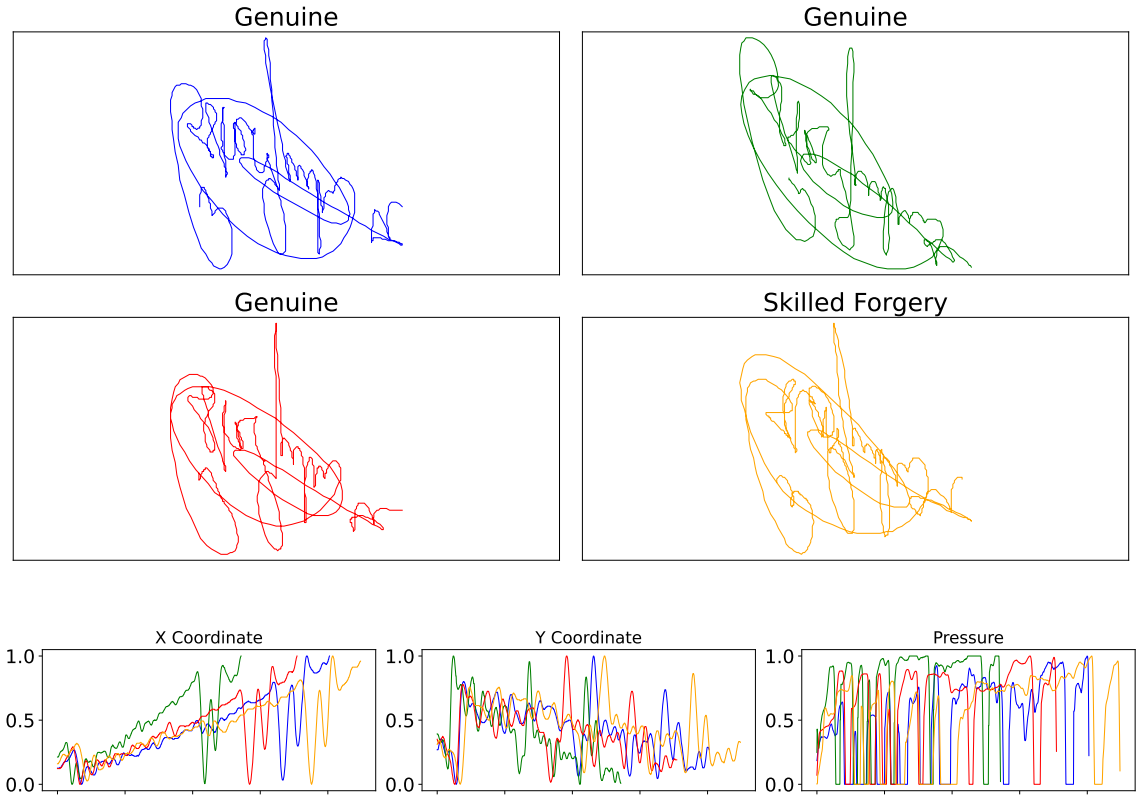
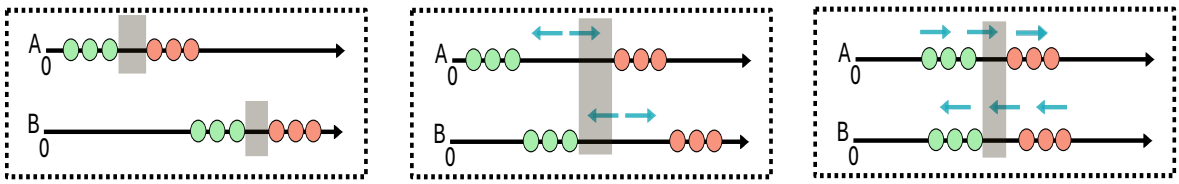


Figure 1.1: Example of stylus-written online signatures and their corresponding time series data for spatial coordinates (x, y) and pressure. The time series were scaled to the range $[0, 1]$ for better visualization.

1.1 Problem Statement

A typical signature verification system is evaluated by the Equal Error Rate (EER): the point where the False Reject Rate (FRR) and False Accept Rate (FAR), which are computed by varying different decision thresholds, are equal. It can be further divided into Writer-Dependent (WD) and Writer-Independent (WI) systems [11]. WD systems rely on specialized classifiers for each writer, which generally require a substantial number of template signatures. However, they produce good results due to the ability to set specific separation thresholds tailored for each writer [12, 13]. WI systems, on the other hand, use one classifier for all the writers and are generally preferred over WD systems [10], especially because they can be used with signatures from writers who had not provided samples during the system design phase, such as the training stage in machine-learning-based systems.

WI systems commonly operate with dissimilarity scores between a set of reference signatures and a query signature, allowing them to work even when only one reference is available. The classification is done through a global threshold: if the dissimilarity score is lower than the threshold, the signature is considered to be genuine and, otherwise, as a forgery. The use of a global threshold, however, accentuates the inherent challenge in signature verification: the high intra-class variability commonly observed among genuine samples and the low inter-class variability associated with skilled forgeries. This issue arises because forged signatures based on the signatures of consistent writers, i.e., writers whose signatures present low intra-class variability, may exhibit less dissimilarity compared to genuine signatures from writers whose signatures present high intra-class variability, which can result in the non-existence of a global threshold able to capture the separation *that exists* between genuine and forged signatures from different writers. Figure 1.2(a) shows an example of this situation.



(a) (b) (c)

Figure 1.2: Example of separation between genuine (green) and forged (red) signatures: (a) A separation exists between genuine and forged signatures using a specific threshold for User A and another for User B, but no single global threshold works well for both users. (b) Increasing the separation between genuine and forged signatures in (a) allows for the creation of a good global threshold. (c) Shifting the scores in (a) to a common region enables the establishment of a good global threshold.

The general approach to tackling this challenge in a WI system involves increasing the separation between genuine and forged signatures, as illustrated in Figure 1.2(b). However, this task becomes increasingly challenging, as it often requires maximizing the separation between samples with slight variations, i.e., genuine signatures and skilled forgeries, while minimizing the separation between samples with high variations, i.e., genuine signatures from inconsistent writers. Figure 1.3 illustrates how difficult this can be. In this work, we address this problem by creating a separation while shifting the dissimilarity scores to a common region—thus minimizing the error arising from the misalignment of writer-specific optimal thresholds—as shown in Figure 1.2(c), through the use of deep metric learning.

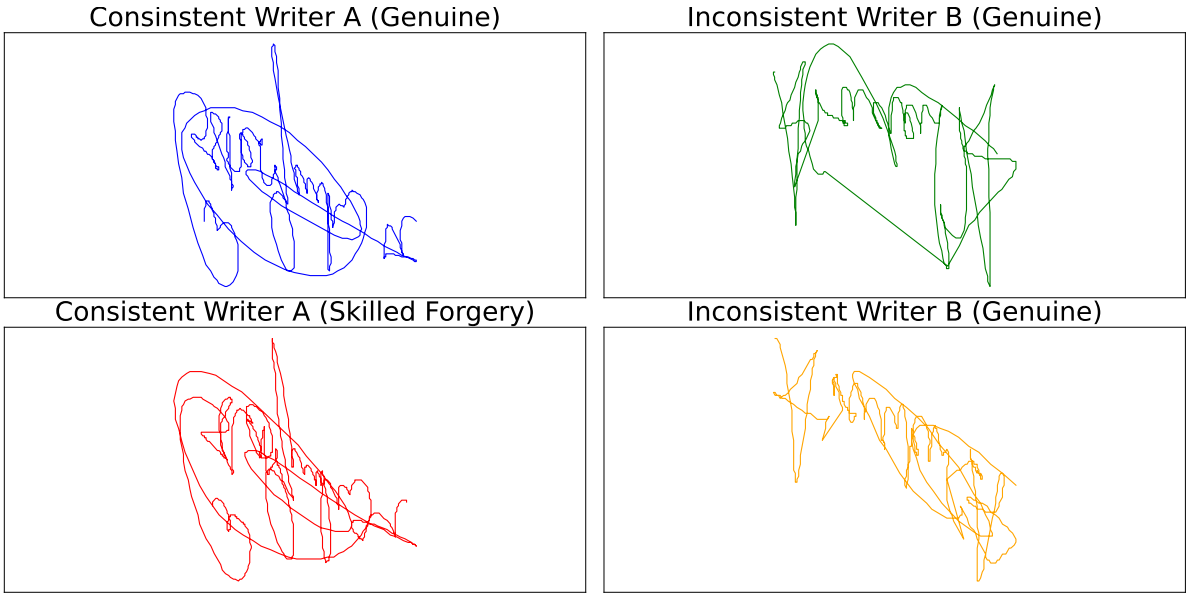


Figure 1.3: Illustration comparing the high similarity between a genuine signature and a skilled forgery from a consistent writer, with the high dissimilarity observed between two genuine signatures from an inconsistent writer.

Recent WI approaches to Online Signature Verification (OSV) commonly rely on deep metric learning [14, 15, 16, 17, 18, 19], where the network aims to learn a representation that better separates genuine and forged signatures. Formally, given a data object o and a representation learning method $f(\cdot)$, representation learning can be defined as the process of learning a low-dimensional vector o', such that $o'=f(o)$ [20]. Then, the dissimilarity between the learned representations can be computed using a dissimilarity function, such as a metric or semi-metric. In the context of deep metric learning, the function $f(\cdot)$ represents a deep neural network designed to generate representations that maximize the discriminability of the chosen dissimilarity function.

Although various types of signature representations have been explored in OSV over the years—such as symbolic [12, 13], spatial [21, 22], and sigma log-normal representations [23]—many of which do not rely on deep learning methods, deep representations have gained increasing attention in recent years, particularly holistic and local representations in deep metric learning systems. In OSV, local representations are generally associated with Dynamic Time Warping (DTW) [24] and its differentiable counterpart, namely Soft-DTW [25]. These algorithms serve as dissimilarity functions for time series that may vary in duration (length) by identifying the optimal alignment between corresponding points within a local window. Consequently, OSV systems based on local representations [14, 16, 17, 18] and DTW leverage the temporal alignment present in local signature properties (e.g., velocity, acceleration, and pressure) [26] to determine dissimilarity among a set of signatures.

Holistic representations in OSV, on the other hand, are often associated with the Euclidean distance and aim to learn a representation that describes a signature as a whole [15]. This approach differs from local representations in two key ways: (1) the learned representation is generally of fixed length, whereas local representations typically allow extracted features to have variable lengths, and (2) temporal alignment is usually lost. Consequently, OSV systems based on holistic representations [15, 19, 21, 27] aim to extract discriminative features from a set of signatures to determine their dissimilarity.

Few works, however, consider the combination of holistic and local representations. A recent study [22] proposed a network that integrates both holistic and local representation learning during the training stage, while also incorporating spatial information from the reconstructed image based on the signature time series. However, during the verification stage, the holistic representation extracted from the signature temporal data is not considered. The results presented two decades ago in [26], however, showed that holistic and local representations modeled by statistical models provided complementary information. In this work, we explore whether deep holistic and local representations provide complementary information and demonstrate that their combination achieves a lower EER compared to using each representation individually.

However, OSV becomes even more challenging when signatures are produced using fingers instead of a stylus. As most individuals are not habituated to signing with their fingers, intra-class variability tends to increase significantly. As demonstrated in this study, this high variability leads to a higher separation error, as the system must adopt a more tolerant threshold—consequently resulting in a greater margin of error for impostors.

Furthermore, most available finger-written online signature datasets contain relatively few users and signatures compared to stylus-written datasets, making it often infeasible to train deep learning architectures from scratch. Additionally, the domain gap between

stylus-written and finger-written signatures prevents simple transfer of models trained on stylus data to finger-based contexts. To address this challenge, we extend our stylus-based framework in this work by incorporating a Domain Adaptation Neural Network (DANN), aiming to mitigate the domain gap and leverage the knowledge learned from stylus-written signatures to improve the error rate of our finger-written OSV framework.

The main contributions of this work are:

1. We present HoLoSig, a novel framework for WI stylus-written OSV that combines deep holistic and local representations during both training and verification stage.
2. We decompose the EER as a sum of the separation error and the alignment error. This allows us to address the problem of minimizing the total EER not only as a separation problem, which is the common approach in the literature, but also from the perspective of optimal WD threshold misalignment.
3. We design Triplet-MMD, a novel loss function based on Triplet Loss [28], Soft-DTW [25] and Maximum Mean Discrepancy (MMD) [29] in order to, at the same time, separate genuine from forged signatures while shifting the dissimilarity scores to a common region to reduce the alignment error.
4. We extend the proposed loss function by replacing MMD with other divergence metrics between distributions, assessing their ability to minimize the alignment error.
5. We propose a signature verifier that combines both holistic and local representations with a novel normalization method that outperforms the common normalization widely used in literature.
6. We perform extensive experiments in the stylus-written scenario in order to assess and confirm the generalization capability of the proposed solution to different acquisition devices.
7. We present HoLoDANN, an extension of the HoLoSig framework tailored for the task of finger-written OSV.

The remainder of this work is organized as follows. Chapter 2 provides a concise literature review, highlighting relevant studies that support our research choices and contextualize HoLoSig within the literature. Chapter 3 outlines the theoretical foundations underlying key components of our method, though these elements are not direct contributions of this study. In Chapter 4, we detail the HoLoSig framework, including the pre-processing, network architecture, proposed loss function, signature verifier, and the HoLoDANN framework. In Chapter 5, we present the DeepSignDB dataset and its experimental protocol.

We also present and discuss our results and compare them with the state-of-the-art. Finally, in Chapter 6, we summarize the contributions presented in this work and discuss our future works.

Chapter 2

Related Work

Numerous approaches have been proposed for OSV, ranging from handcrafted feature extraction, statistical models, and classical machine learning algorithms to deep learning architectures tailored to capture significant characteristics of the signatures to improve verification. This section presents a structured review of some key contributions, focusing on three core aspects: the representation of signature data, the verification strategies and the challenges in finger-written OSV.

Most existing studies evaluate their models using well-known benchmark datasets, such as MCYT [30], SVC2004 [31], BiosecurID [32], or DeepSignDB [33]. While these datasets provide a useful basis for comparison, differences in experimental setups — particularly between user-dependent and user-independent protocols — can make direct comparison of results challenging. Thus, here we focus on the methods employed by different studies that achieved relevant results without delving into rigorous comparisons. We describe the dataset and experimental protocol used in this work in Chapter 5.

2.1 Signature Representations and Feature Modeling

OSV systems can be categorized into parameter-based and function-based approaches, depending on the nature of the input features. Parameter-based systems represent signatures as feature vectors that encapsulate various descriptive attributes, such as the number of pen-up movements, as well as the height and width of the signature [12, 13, 19, 34]. These feature vectors are then utilized as inputs to the OSV framework for verification. The effectiveness of parameter-based systems is highly dependent on the quality of the extracted features, as they predominantly rely on handcrafted descriptors [35]. Nevertheless, these systems often demonstrate strong performance in terms of authentication speed.

Function-based systems, by contrast, represent signatures as time-dependent functions—such as spatial trajectories and pressure. These approaches have gained wider popularity over parameter-based methods, as they typically achieve lower error rates and are less reliant on domain-specific knowledge [14, 15, 16, 18, 27, 22, 36, 37, 38]. However, they are not entirely free from the handcrafted aspect existing in the parameter-based systems, as it is a common practice to extract additional time-dependent features from the raw time-series data to further enhance system performance such as velocity and acceleration [35].

Whether an OSV system relies on parametric or function-based inputs, most approaches utilize these features to construct a signature representation that facilitates the verification process. Over the years, a variety of signature representation strategies have been proposed such as holistic, local, symbolic, Sigma LogNormal, and spatial representations.

Spatial representations in OSV share some similarities to the Offline Signature Verification field by reconstructing the signature image from the time-series describing the spatial coordinates and aim to take advantage from the advancements in the area of computer vision to enhance the system performance. However, works that rely on spatial representations [21, 22, 39] also often incorporate information not present in offline scenarios and frequently employ ensemble strategies that also account for the temporal aspects of online signatures.

Symbolic representations are typically based on parametric data, and OSV systems that adopt this approach often incorporate techniques inspired by clustering algorithms [12, 13, 40, 41]. The general idea involves identifying the most relevant parametric features by evaluating their consistency across genuine reference samples for each writer. Intervals are then defined to represent the acceptable range of intra-class variability for each parameter. These intervals guide the verification process, which can range from threshold-based decision mechanisms—assessing how many parameters fall within the defined ranges—to machine learning classifiers such as k-nearest neighbors and Support Vector Machine (SVM).

Sigma LogNormal representations are a special type of representation that can be used not only as a means for signature verification, but also to allow the generation of synthetic samples. Based on the kinematic theory of rapid human movements [42, 43, 44, 45], the Sigma LogNormal representation is hypothesized to enhance the characterization of a signee's hidden specificities while remaining invariant to both language and acquisition device [23].

The use of the Sigma LogNormal representation has been explored through various strategies over the years in OSV. In [46], an ensemble approach is proposed, composed

of a threshold-based classifier and a Sigma LogNormal classifier. The idea is that the Sigma LogNormal classifier determines whether a signature is genuine or forged based on its Sigma LogNormal characteristics, and this decision is used as a weighting factor to support the threshold-based classifier since the former is capable of revealing behaviors commonly found in skilled forgeries, such as hesitation and unnatural velocity. A similar ensemble strategy is employed in [47], where the Sigma LogNormal classifier is used to evaluate the consistency of strokes between two signatures based on factors such as stroke count, compatibility, and frequency.

However, the most prominent use of the Sigma LogNormal representation lies in the generation of synthetic samples in order to increase the amount of training data, a common problem in signature verification due to the limited size of the available datasets for the research community. In [48], an OSV system is designed with only one genuine sample available for each user, which is used as the base for generating new synthetic samples with relative variability. These synthetic samples are then compared using DTW against the query sample to determine if the query is genuine or forged. In [15], the Sigma LogNormal representation is extracted from the raw data, perturbed, and then reconstructed, thereby generating synthetic samples with slight variabilities that are further used as forgeries to train a deep metric learning system without relying on real skilled forgeries.

While these three representations have made significant contributions to OSV, most recent state-of-the-art approaches rely on deep holistic and local representations.

Along with using the Sigma LogNormal representation for generating synthetic samples, the study presented in [15] also extracts deep holistic representations from function-based time-series data using a Convolutional Neural Network (CNN), which produces fixed-length feature vectors to compute the dissimilarity between a query and a set of reference signatures. A similar approach is found in [27], where instead of a CNN, an autoencoder processes the function-based inputs to predict the probability that a query signature belongs to the same writer as a reference. Holistic representations are also extracted from signature strokes—defined by both image and temporal data—in [21], using a CNN-Long Short Term Memory (LSTM) network, with the final classification performed by an SVM. Additionally, deep holistic representations have been applied to parameter-based data as well, as shown in [19], where a CNN is used to process the handcrafted features.

Deep local representations are commonly associated with DTW and have been among the most widely used representations in OSV in recent years. A local representation necessarily relies on function-based input data. The works presented in [14, 22] extract local representations using a Convolutional Recurrent Adaptative Network (CRAN), which are then compared using DTW to measure the dissimilarity between a query signature and

a set of reference signatures. This approach is similar to the siamese CNN proposed in [18], where local representations are also extracted and later compared through DTW. Following a different approach, in [33], signatures are first aligned using DTW and then processed by a siamese recurrent network, which outputs a dissimilarity score between pairs of signatures.

In light of recent developments, we adopt a function-based approach aligned with recent advancements in OSV, and use the time-series data to extract both holistic and local representations within a siamese network. This setup enables the computation of a dissimilarity score between a reference set and a query signature.

2.2 Verification Strategies and General Contributions

In an OSV system, the standard method for determining whether a query signature is genuine or forged relies on a threshold-based classifier. Specifically, if the similarity score of the query signature falls below a predefined threshold, the signature is classified as genuine; otherwise, it is considered a forgery. By systematically varying the decision threshold, one can derive the curves representing the False Accept Rate (FAR) and False Reject Rate (FRR). The point at which these two rates are equal is referred to as the EER, a commonly used metric to evaluate the verification performance of a biometric system.

In practice, a threshold-based classifier offers flexibility in defining the decision boundary, allowing the threshold to be adjusted to prioritize either a lower FRR or FAR, depending on the specific requirements of the application. Moreover, this type of verifier can be easily configured for different operating modes: as a WI system, where a single global threshold is applied to all writers, or as a WD system, where each writer has an individual decision threshold tailored to their signature characteristics.

There are several strategies for determining the score of a query signature. Currently, most state-of-the-art systems employ deep metric learning approaches, in which a neural network extracts feature representations for both the query and a set of reference signatures. These representations are then compared using a dissimilarity function [14, 15, 18, 17, 22]. However, alternative verification strategies have also been explored. Some studies utilize SVM-based classifiers or rely on neural network architectures that directly produce a verification score without the use of an explicit dissimilarity function [33, 27]. In the latter case, the neural network is designed to integrate information from multiple signatures to assess their degree of similarity.

Template matching [36, 37] is also a widely used approach, particularly in systems that do not rely on deep learning. In such cases, a set of reference signatures is used

to construct a representative template that captures their shared characteristics. This strategy bears resemblance to methods based on symbolic representations and interval windows [13, 40], where similarity is assessed by evaluating the number of features that fall within an acceptable range of intra-class variability.

Beyond improvements in EER, many of these studies seek to advance the field of OSV by introducing key contributions that expand the understanding and scope of the problem. Some works focus on re-framing the task through alternative representations [12, 23] or by exploring novel learning paradigms, such as meta-learning [16]. Others investigate innovative data processing strategies, including the full integration of DTW into the training phase of deep learning models [14], as well as the use of multimodal approaches that combine complementary sources of information [22, 38, 39].

Some studies also address the practical feasibility of deploying OSV systems in real-world scenarios. For instance, certain works explore training deep neural networks without access to skilled forgeries [15, 21], or with only a single genuine sample per writer [48]. These approaches are motivated by real-world constraints, such as the difficulty of collecting large volumes of data and the ethical or legal limitations associated with obtaining skilled forgeries. Additionally, other studies investigate potential vulnerabilities in OSV systems, such as the threat posed by synthetic signature samples used in presentation attacks [49, 50].

In this sense, the present work aims to shed light on the challenges faced by threshold-based verifiers due to the misalignment of writer-specific optimal thresholds. To address this issue, we propose a decomposition of the overall EER in a WI system into two components: the alignment error and the separation error. We believe this decomposition can help the research community better design future OSV systems that explicitly consider the dispersion of writer-specific optimal thresholds. Additionally, it offers a more reliable way to evaluate system performance under WD settings, as an alternative to the commonly used average EER, which can be sensitive to the size of the evaluation set.

2.3 Challenges in finger-written OSV

Few studies have directly addressed the problem of finger-written OSV. In fact, most recent works in OSV focus on deep metric learning frameworks trained exclusively on stylus-written signatures, which are later evaluated in finger-written scenarios [33, 22, 15, 14]. In these studies, no fine-tuning or domain adaptation is performed when transitioning from stylus to finger-written input. Moreover, these specific works were evaluated using the DeepSignDB dataset—the same dataset employed in this study and described in Chapter 5—where finger-written signatures lack pressure information. Thus, to enable

the use of frameworks originally trained with pressure data, a constant pressure value (typically 0 or 1) is assumed for all finger-written signatures.

Naturally, it is expected that a stylus-trained OSV framework that incorporates pressure information will rely on this feature to make accurate predictions. Consequently, the absence of this time-dependent signal is likely to degrade the system's performance. Furthermore, the use of pressure information may prevent the architecture from learning alternative discriminative features that could compensate for its absence, which is the case in the finger-written signatures from DeepSignDB.

This gap in the literature is likely due to the limited number of publicly available datasets containing finger-written online signatures for the research community. Furthermore, most of these datasets, such as Mobisig [51], eBioSign DS1 [52], and eBioSign DS2 [33], contain significantly fewer samples compared to stylus-written signature datasets, making training from scratch often impractical. In addition, the higher intra-class variability observed among genuine finger-written samples further increases the challenge, making the task even more difficult than when using stylus-written signatures.

However, there have been some attempts to address the problem of finger-written OSV. In [53], a functional OSV system was developed to identify the most stable regions within a writer's signature. This system enables the computation of a dissimilarity score using DTW, incorporating a weighting mechanism that emphasizes stable regions while minimizing the influence of inconsistent ones. As the approach does not rely on machine learning algorithms, the authors were able to conduct experiments on a private dataset containing signatures from 25 users.

In [54], a WD functional OSV system based on deep learning was proposed. Initially, signatures are processed using a CNN to extract a holistic representation, which is then passed through an autoencoder to compute dissimilarity between signatures using the mean squared error. Notably, this study is among the few that explores training deep learning models specifically for this task, likely due to the scarcity of finger-written signature samples in public datasets. The authors conducted their experiments on a private dataset comprising 20 users.

In this work, we take a step further by adapting the proposed HoLoSig architecture—originally designed for stylus-written signatures—to the problem of finger-written OSV, by using both finger-written and stylus-written signatures during training.

Chapter 3

Theoretical Foundation

This chapter presents key aspects of the theoretical foundation necessary to understand the methods and techniques employed in this research. Although these elements do not constitute direct contributions of this work, they are fundamental to its development. The concepts addressed include deep metric learning, convolutional and recurrent neural networks, sequence alignment techniques, and loss functions relevant to the proposed method.

3.1 Deep Metric Learning

Over recent years, numerous OSV systems based on Deep Metric Learning have been proposed [14, 15, 16, 17, 18, 19]. A common goal among these approaches is to employ neural networks to learn a feature representation that maps each signature into an embedding space where samples from the same writer are positioned close together, while those from different writers are placed farther apart. These learned representations, typically expressed as feature vectors, are subsequently compared using a dissimilarity function—such as DTW or Euclidean distance—to evaluate the dissimilarity between signatures. Rather than learning to directly classify a signature as genuine or forged, the network is trained to enhance the effectiveness of the dissimilarity function by improving feature separability.

This property is particularly advantageous for WI OSV systems, as it facilitates generalization to writers not seen during training. By focusing on learning how to measure dissimilarity rather than perform classification, the model is encouraged to capture the distinctive features that characterize each signature. As a result, the dissimilarity function can effectively measure the difference between an input signature and a set of reference signatures, enabling the system to verify signatures from previously unseen writers during the training stage. This setup constitutes a zero-shot learning scenario [55].

3.2 Convolutional Recurrent Adaptative Network

One of the primary advantages of using neural networks to extract feature representations for signature dissimilarity analysis is their ability to perform both linear and non-linear transformations on raw input data. This is exemplified by the CRAN architecture employed in this study (detailed in Chapter 4), which integrates both CNN and Recurrent Neural Network (RNN) components. CNNs are well-suited for capturing spatial relationships in the data and typically consist of linear operations, such as convolutional and fully connected layers, alongside non-linear operations, such as pooling layers and activation functions. Similarly, RNN, such as the Gated Recurrent Unit (GRU) [56], process input sequentially, enabling the model to learn temporal dependencies while also introducing non-linearity.

The combination of these components enables CRAN to learn a more expressive and discriminative feature space when compared to approaches based on hand-crafted features, which are typically restricted to linear transformations. CRAN was originally introduced in [14] and is constructed by placing a CNN module before a Gated Auto Regressive Unit (GARU) [16]. The GARU (Gated Additive Recurrent Unit) is a simplified variant of the traditional GRU, where the update gate is removed, meaning that the hidden state is always updated at each time step. Formally, the GARU can be defined as:

$$\begin{aligned} r_t &= sigm(W_r x_t + U_r y_{(t-1)} + b_r), \\ y_t &= tanh(W_{act} x_t + U_{act}(r_t \odot y_{t-1}) + b_{act}). \end{aligned} \tag{3.1}$$

where x_t and y_t are the input and output vectors at time step t, $W_r, W_{act}, U_r, U_{act}$ are parameter matrices, b_r, b_{act} are biases and $\odot$ denotes the element-wise multiplication. Figure 3.1 illustrates a GARU cell.

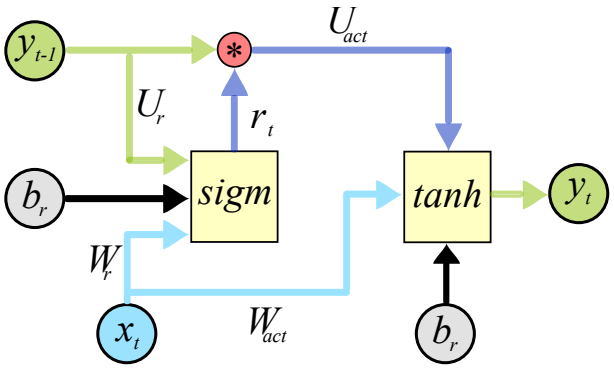


Figure 3.1: Example of GARU cell.

3.3 Selective Pooling

Time series data, such as online signatures in functional approaches as discussed in Chapter 2, often exhibit variable lengths. A common strategy to address this characteristic in OSV systems is to employ techniques designed to handle variable-length sequences. These techniques include elastic dissimilarity measures such as DTW, neural architectures such as RNN, padding-based methods, or transformations that convert the sequences into fixed-length representations through techniques such as interpolation.

More recently, Selective Pooling (SP) [15] has been introduced to aggregate variable-length sequences into fixed-size vectors specifically for the context of OSV. The SP mechanism is inspired by the multi-head attention framework and leverages learnable queries to guide the attention process. Given N_{SP} linear layers with D_{SP} hidden units, the objective is to project an input sequence F into N_{SP} subspaces ($F \rightarrow F_{i i=1}^{N_{SP}}$), where $F_i \in \mathcal{R}^{|F| \times D_{SP}}$ and $|F|$ is the length of sequence F.

For each subspace indexed by i, a learnable query vector w_i of dimension D_{SP} is defined. The sub-vector obtained from F_i through the SP mechanism, as described in [15], is computed using the following expression:

$$\text{vec}_i = \text{softmax}\left(\frac{w_i F_i^T}{\sqrt{D_{SP}}}\right) F_i. \tag{3.2}$$

The final output of the SP module is the concatenation of all resulting sub-vectors, namely [vec_1,...,$\text{vec}_{N_{SP}}$]. Figure 3.2 illustrates the complete SP architecture.

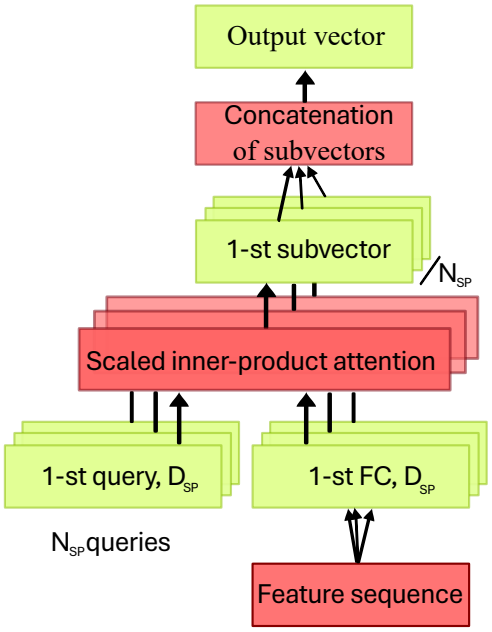


Figure 3.2: Selective Pooling Architecture.

3.4 Poly1 Cross-Entropy Loss

The Poly1 Cross-Entropy Loss [57] is a variant of the traditional Cross-Entropy Loss, motivated by the Taylor expansion of the original loss function. The findings presented in [57] indicate that, depending on the domain, modifying the first polynomial coefficient in the Cross-Entropy Loss can lead to improved performance. The Poly1 Cross-Entropy Loss is defined as:

$$\mathcal{L}_{Poly-1} = \mathcal{L}_{CE} + \epsilon(1 - p_t), \tag{3.3}$$

where $\mathcal{L}_{CE}$ is the traditional cross-entropy loss, $\epsilon \in \mathbb{R}^+$ is a tunable hyperparameter and p_t the prediction probability of the target class.

3.5 Sequence Alignment: DTW and Soft-DTW

The DTW, originally introduced in [58], is designed to measure the dissimilarity between two variable-length time series by comparing their points within a local neighborhood window. It determines the optimal alignment between sequences by allowing one point in a series to be mapped zero or more times to points in the other, making it an elastic dissimilarity measure. Over the years, DTW has demonstrated strong performance in the context of OSV [14, 18, 22, 33, 36, 37]. Let D be a dynamic programming matrix, the DTW can be determined by the following dynamic programming recursion:

$$D(i,j) = \|x_i - y_j\|^2 + \min \begin{cases} D(i-1,j), \\ D(i,j-1), \\ D(i-1,j-1). \end{cases} \tag{3.4}$$

However, since DTW uses the non-smooth operator min, it is not differentiable, which complicates its application in deep learning. To address this issue, the Soft-DTW was proposed in [25] as a differentiable variant. It is computed using the same recursion as DTW, but the non-smooth min operator is replaced with a differentiable version defined as:

$$\text{softmin}(a_1, ..., a_n) = \begin{cases} \min_{1,...,n} a_i, & \text{if } \gamma = 0, \\ -\gamma \log \sum_{i=1}^{n} \exp^{\frac{-a_i}{\gamma}}, & \text{if } \gamma > 0. \end{cases}, \tag{3.5}$$

where γ is a smoothing parameter that controls the degree of approximation to the standard min operator. Observe that when $\gamma = 0$ Soft-DTW works exactly as DTW.

Chapter 4

Methodology

The proposed HoLoSig framework for stylus-written OSV consists of three key components: signature pre-processing, representation extraction and verification process, as illustrated in Figure 4.1. First, time-dependent functions are extracted from the data describing each signature and serve as inputs to HoLoSig feature extractor. Next, the feature extractor processes these time-dependent functions to extract both holistic and local representations. Finally, using the representations from a set of reference signatures and a query signature, the signature verifier computes the dissimilarity score of the query relative to the references. By performing comparisons according to a defined experimental protocol (benchmark), we can determine the EER.

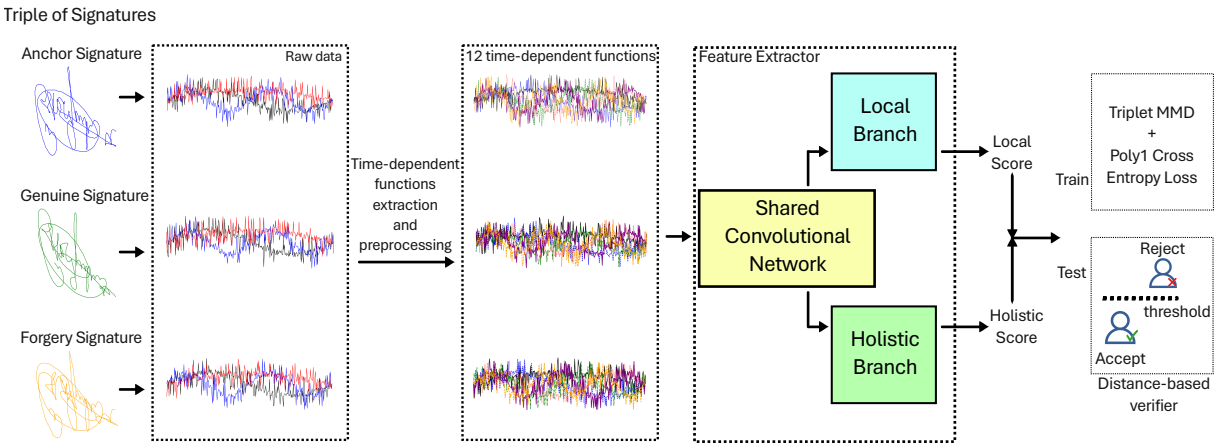


Figure 4.1: Overall HoLoSig framework.

Table 4.1: Input time-dependent functions for HoLoSig. Mean normalization is denoted as $\mathring{x}_i = \frac{x_i - \bar{x}}{x_{max} - x_{min}}$, while standard deviation normalization (z-score) is $x_i^* = \frac{x_i - \bar{x}}{s_x}$, with s_x as the standard deviation of x.

#	Description	Definition	Normalization
1.	Spatial X-coordinate	$x = x(t)$	$\mathring{x}_i$
2.	Spatial Y-coordinate	$y = y(t)$	$\mathring{y}_i$
3.	Velocity magnitude	$v = \sqrt{\dot{x}^2 + \dot{y}^2}$	v_i^*
4.	Path-tangent angle	$\theta = \arctan(\dot{x}/\dot{y})$	θ_i^*
5.	Cosine of path-tangent angle	$\mathbb{C} = \cos(\theta)$	$\mathbb{C}_i^*$
6.	Sin of path-tangent angle	$\mathbb{S} = \sin(\theta)$	$\mathbb{S}_i^*$
7.	Pressure	$p = p(t)$	p_i^*
8.	First-order derivative of velocity magnitude	$\dot{v}$	$\dot{v}_i^*$
9.	First-order derivative of path-tangent angle	$\dot{\theta}$	$\dot{\theta}_i^*$
10.	Log curvature radius	$\rho = \log(v/\dot{\theta})$	ρ_i^*
11.	Centripetal acceleration	$c = v \times \dot{\theta}$	c_i^*
12.	Total acceleration	$a = \sqrt{\dot{v}^2 + c^2}$	a_i^*

4.1 Input Time Functions

Let t_i represent a timestamp in the set $\{t_i, i = 1, ..., n_t\}$. The data describing each signature in this study consist of the time-dependent functions x, y and p, where $x(t_i)$, $y(t_i)$ and $p(t_i)$ denote, respectively, the values of spatial x-coordinate, spatial y-coordinate and pressure at time t_i. For each signature, we compute the 12 time-dependent functions presented in Table 4.1, which are commonly used in the literature [14, 15, 22, 33] and serve as inputs for the backbone. For simplicity, from now on, we will refer to the set of these 12 time-dependent functions as "signature".

4.2 Backbone Architecture

The backbone architecture is based on CRAN [14] to extract local representations, with the addition of SP [15] to extract holistic representations. The architecture consists of two convolutional blocks followed by a GARU [16] for nonlinear transformation and context modeling. We opt for GARU, a variant of GRU without the update gate, because it has shown good results when used together with DTW [14, 16, 22]. The architecture then bifurcates, as can be seen in Figure 4.2. On one branch, aimed at extracting the holistic representation, the output from the GARU passes through the SP, which pools the variable-length features into a fixed-length vector. It is noteworthy that each signature in a batch has its own length (duration). Thus, SP condenses the extracted information into

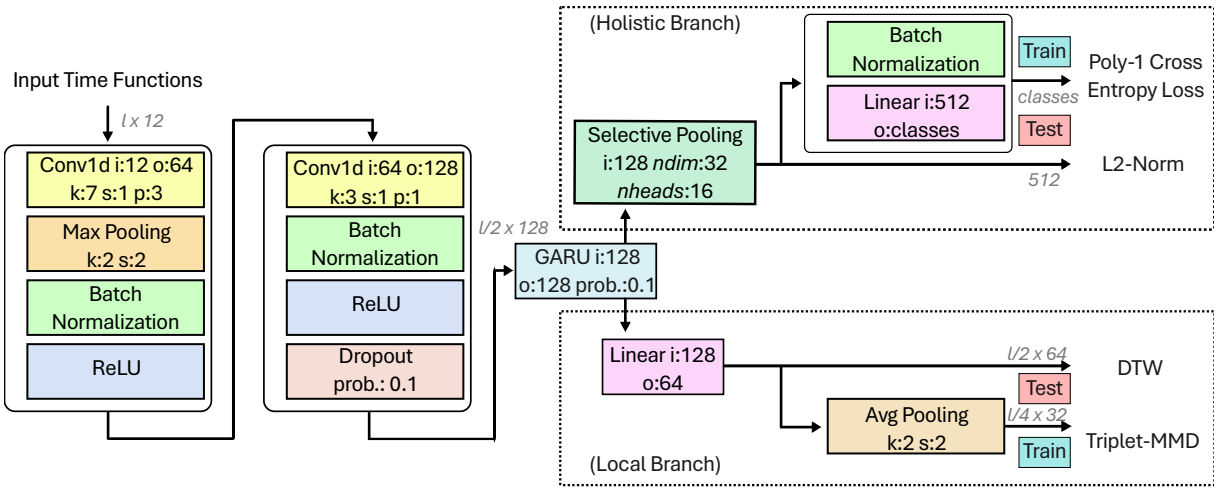


Figure 4.2: Backbone Architecture. "l" denotes the maximum length (duration) of the time-dependent functions in each batch. The "i", "o", "k", "s", "p", "prob.", "nheads" and "ndim" denotes respectively: input dimensionality, output dimensionality, kernel size, stride, padding, dropout probability, number of attention heads and heads dimensionality.

a fixed-length vector, enabling both the use of L2-Norm during the verification stage and the linear layer that follows, which maps each signature to a writer during the training stage, making the computation of Poly-1 Cross Entropy Loss [57] possible.

On the other branch, where we aim to extract a local representation, the output from the GARU passes through a Linear Layer for dimensional reduction. During the training, we adopt the same late average pooling strategy used in [14] to reduce memory consumption. Different from the holistic representation, each extracted local representation in a batch has its own length, allowing us to explore their alignment with DTW during verification stage and Soft-DTW during training stage since DTW is not differentiable. Figure 4.2 presents the detailed architecture.

4.3 Proposed Loss Function

The overall loss function for HoLoSig's backbone is the sum of the Triplet Loss, MMD, and the Poly-1 Cross-Entropy Loss. While the Triplet Loss and MMD directly affect both the shared convolutional block and the local representation branch, the Poly-1 Cross-Entropy Loss—computed on a per-mini-batch basis and averaged over the number of mini-batches in a batch—impacts both the shared convolutional block and the holistic representation branch. The complete loss function is defined as follows:

$$\mathcal{L} = \bar{\mathcal{L}}_{tri} \times \zeta + \overline{\text{MMD}} \times \alpha + \mathcal{L}_{Poly-1} \times \psi, \tag{4.1}$$

where $\zeta \in \mathbb{R}^+$, $\alpha \in \mathbb{R}^+$, $\psi \in \mathbb{R}^+$ are weighting factors that regulate the contribution of each loss function during optimization.

4.3.1 Triplet Loss to Enforce Writer Separation

In order to enforce writer separation, we employ Triplet Loss [28] with Soft-DTW [25], since the original DTW [58] is not differentiable as also done in [14]. Unlike other metric learning loss functions, such as Contrastive Loss [59] — which encourages same-class samples to be similar (zero dissimilarity) — Triplet Loss aims to create a separation between different classes of at least the margin value, regardless of the dissimilarity between same-class samples. In the domain of OSV, this behavior is desirable due to the high intra-class variability, allowing the model to learn representations where genuine and forged signatures of a given writer are separated, even if this separation is not centered around the same value for all writers.

Given a batch of n_w writers, let k be the index of each writer from 1 up to n_w. Therefore, each batch consists of n_w mini-batches, one for each writer. Each mini-batch is formed by an anchor signature $\mathcal{A}$, n_g genuine signatures $\{\mathcal{G}_i, i = 1, \cdots, n_g\}$ (i.e., signatures from the same writer as the anchor $\mathcal{A}$) and n_f forged signatures $\{\mathcal{F}_j, j = 1, \cdots, n_f\}$, which can be skilled or random forgeries with respect to the anchor signature. Figure 4.3 presents an illustration of a batch composed of two mini-batches.

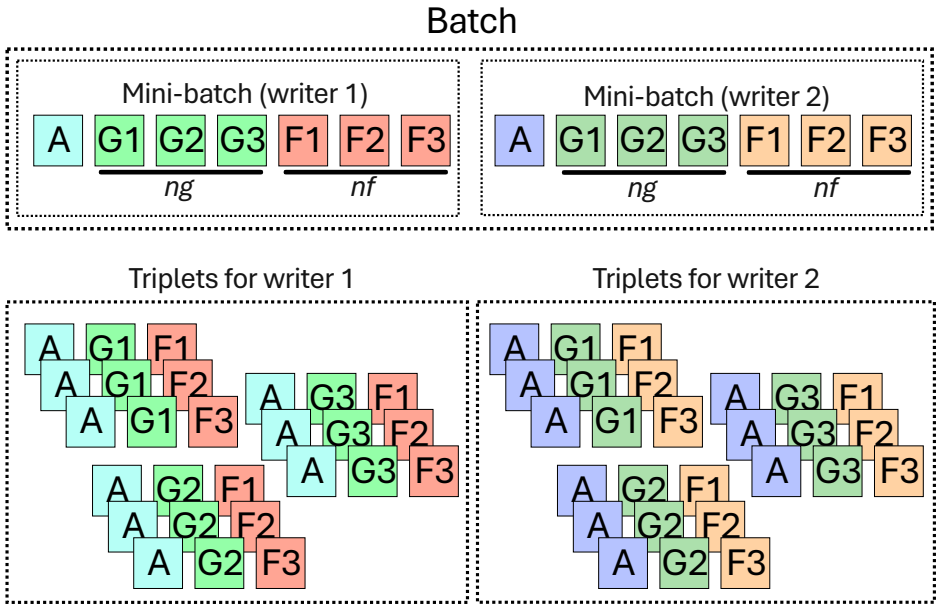


Figure 4.3: Example of batch when $n_w = 2, n_g = 3$, and $n_f = 3$, thus resulting in $n_g \times n_f = 3 \times 3 = 9$ triplets for each writer.

The loss of a triplet $(\mathcal{A}, \mathcal{G}, \mathcal{F})$ is defined as:

$$l(\mathcal{A}, \mathcal{G}, \mathcal{F}) = \max(0, d_{tr}(\mathcal{A}, \mathcal{G}) + m - d_{tr}(\mathcal{A}, \mathcal{F})), \tag{4.2}$$

in which $m \in \mathbb{R}^+$ represents a margin defining the minimum required separation between signature pairs from the same writer and those from different writers. The d_{tr} for any two input signatures $\mathcal{X}$ and $\mathcal{Y}$ is defined as:

$$d_{tr}(\mathcal{X}, \mathcal{Y}) = \frac{\mathrm{dtw}_\gamma(\varphi(f(\mathcal{X})), \varphi(f(\mathcal{Y})))}{\sqrt{|\varphi(f(\mathcal{X}))|^2 + |\varphi(f(\mathcal{Y}))|^2}}, \tag{4.3}$$

where dtw_γ is the Soft-DTW [25] with smoothing factor γ, $f(\cdot)$ denotes the output of the Local Branch on HoLoSig backbone architecture, φ the late average pooling and $|\cdot|$ returns the length of the input sequence (which is affected by the pooling layers). The Triplet Loss of writer k is defined as:

$$\mathcal{L}^k_{tri} = \frac{\sum_{i=1}^{n_g} \sum_{j=1}^{n_f} l(\mathcal{A}^k, \mathcal{G}^k_i, \mathcal{F}^k_j)}{1 + \sum_{i=1}^{n_g} \sum_{j=1}^{n_f} \mathbb{I}(l(\mathcal{A}^k, \mathcal{G}^k_i, \mathcal{F}^k_j) > 0)}, \tag{4.4}$$

in which $\mathbb{I}$ is an indicator function: if the condition is satisfied, the output is 1, otherwise it is 0. The loss of a batch is defined as:

$$\bar{\mathcal{L}}_{tri} = \frac{1}{n_w} \sum_{k=1}^{n_w} \mathcal{L}^k_{tri} + \max(0, \beta - \mathrm{abs}(\bar{g} - \bar{s})) \times \iota, \tag{4.5}$$

where $\iota \in \mathbb{R}^+$ is a weighting factor for the second term, which enforces a minimum separation of at least $\beta \in \mathbb{R}^+$ between the average d_{tr} values between the anchor and genuine samples ($\bar{g}$), and between the anchor and skilled forgeries ($\bar{s}$). Even when the triplet loss becomes zero, meaning that there is at least a margin m of separation between genuine and forged signatures (as defined in Equation 4.2), the distance between the average d_{tr} anchor-genuine and average d_{tr} anchor-forged signatures can still be further increased by introducing this second term. This additional separation is encouraged without significantly penalizing the overall loss when the network is struggling thanks to the balancing effect provided by the ι hyperparameter. Note that a positive β ensures that no trivial solution is allowed (where $\bar{g}$ is equal to $\bar{s}$).

4.3.2 MMD to Achieve Better Threshold Alignment

In order to shift the dissimilarity scores to a common region, we employ MMD [29] as a loss function over the Soft-DTW discrepancies (d_{tr}) which are also used in Triplet Loss. Let $P = \{p_1, \cdots, p_m\}$, $Q = \{q_1, \cdots, q_m\}$ be independent random variables with distributions

$\mathcal{P}$ and $\mathcal{Q}$, respectively. The empirical estimate of the distance between $\mathcal{P}$ and $\mathcal{Q}$, as defined by MMD, can be calculated as:

$$\mathrm{MMD}^2(\mathcal{P}, \mathcal{Q}) = \frac{1}{(m)(m-1)} \sum_{i \neq j}^{m} \kappa(p_i, p_j) + \kappa(q_i, q_j) + \kappa(p_i, q_j) + \kappa(p_j, q_i), \tag{4.6}$$

where:

$$\kappa(p_i, q_j) = \sum_{u=0}^{\eta-1} \kappa_u(p_i, q_j). \tag{4.7}$$

We consider a Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}$ induced by a universal kernel function κ_u. Specifically, we use the Radial Basis Function (RBF) kernel, which has been proven to generate a universal RKHS [60]. The RBF kernel κ_u for any p_i and q_j is defined as:

$$\kappa_u(p_i, q_j) = \exp(\sigma_u \times d(p_i - q_j)^2), \tag{4.8}$$

where d is the Euclidean distance and $\sigma_u \in \mathbb{R}^+$ is a free parameter defined as:

$$\sigma_u = \frac{\sum D_{ij}}{(2m)^2 - 2m} \cdot \frac{\xi^u}{\xi^{\lfloor \eta/2 \rfloor}}, \tag{4.9}$$

where $\eta \in \mathbb{N}^+$ and $\xi \in \mathbb{R}^+$ are hyperparameters that represent, respectively, the number of kernels and kernel multiplier. $D_{ij} = d(\tau_i - \tau_j)^2$ is the squared Euclidean distance of the pair $\{\tau_i, \tau_j\}$, where $\{\tau_1, ..., \tau_m, ..., \tau_{2m}\}$ is the tensor created from the concatenation of the samples from P and Q. Finally, let W_i correspond to the set of all Soft-DTW discrepancies (d_{tr}) between the anchor and the remaining signatures for each writer in a mini-batch $\{W_l, l = 1, \cdots, n_w\}$. The final MMD loss is:

$$\overline{\mathrm{MMD}} = \max_{k \neq l}^{n_w} \mathrm{MMD}^2(W_k, W_l). \tag{4.10}$$

4.4 Alternatives to MMD

In this work, we also evaluate the performance in aligning the separation windows through the use of three distance metrics between distributions. Specifically, we consider the Cramér distance when **p**=1 and **p**=2 and the Energy Distance. As MMD, these functions are commonly used in the context of domain adaptation and generative models [61], but here act as a mean to move the d_{tr} scores to a common region.

4.4.1 Cramér Distance

The Cramér Distance, in the discrete case, can be computed by comparing the Cumulative Distribution Function (CDF) of two distributions. Specifically, let $P = \{p_1, \cdots, p_m\}$ and $Q = \{q_1, \cdots, q_m\}$ denote the CDF values of distributions $\mathcal{P}$ and $\mathcal{Q}$, respectively. The normalized Cramér Distance between $\mathcal{P}$ and $\mathcal{Q}$ is then defined as:

$$d_{\text{Cramér}}(\mathcal{P}, \mathcal{Q}) = \left(\frac{1}{m} \sum_{i=1}^{m} (p_i - q_i)^{\mathbf{p}} \right)^{1/\mathbf{p}}, \tag{4.11}$$

where $\mathbf{p} \in \{1, 2\}$ in this work.

In our application, we extend this notion by computing the Cramér Distance between the dissimilarity score distributions of two different writers. Since a CDF is, by definition, a monotonically increasing function, we first sort the dissimilarity scores for each writer before applying Equation 4.11. Finally, the Cramér Distance for a batch of writers is defined as the maximum pairwise distance between all distinct writer pairs:

$$\overline{D_{\text{Cramér}}} = \max_{k \neq l}^{n_w} d_{\text{Cramér}}(\text{sorted}(W_k), \text{sorted}(W_l)), \tag{4.12}$$

where n_w is the number of writers in the batch, and W_k and W_l represent the dissimilarity score vectors for writers k and l, respectively.

4.4.2 Energy Distance

The Energy Distance can be seen as a generalization of the Cramér Distance to the multivariate case [61]. In this work, however, we employ the Energy Distance in the one-dimensional setting. Let $P = \{p_1, \cdots, p_m\}$ and $Q = \{q_1, \cdots, q_m\}$ be independent random variables drawn from distributions $\mathcal{P}$ and $\mathcal{Q}$, respectively. The Energy Distance between $\mathcal{P}$ and $\mathcal{Q}$ is defined as:

$$D_{\text{Energy}}(\mathcal{P}, \mathcal{Q}) = \frac{2}{m^2} \sum_{i=1}^{m} \sum_{j=1}^{m} |p_i - q_j| - \frac{1}{m^2} \sum_{i=1}^{m} \sum_{j=1}^{m} |p_i - p_j| - \frac{1}{m^2} \sum_{i=1}^{m} \sum_{j=1}^{m} |q_i - q_j| . \tag{4.13}$$

Finally, the total Energy Distance for a batch of writers is defined as the maximum pairwise distance between all distinct writer pairs:

$$\overline{D_{\text{Energy}}} = \max_{k \neq l}^{n_w} d_{\text{Energy}}(\text{sorted}(W_k), \text{sorted}(W_l)), \tag{4.14}$$

where n_w is the number of writers in the batch, and W_k and W_l represent the dissimilarity score vectors for writers k and l, respectively.

4.5 Signature Verifier

During the verification stage, we apply the traditional DTW to the local representation — denoted by the function $f(\cdot)$ — and the L2-Norm to the holistic representation, denoted by the function $g(\cdot)$. Given a set of n_r reference signatures $\{\mathcal{R}_i, i = 1, \cdots, n_r\}$ and a query signature $\mathcal{S}$, the dissimilarity score of $\mathcal{S}$ with respect to the references $\{\mathcal{R}_i, i = 1, \cdots, n_r\}$ is defined as:

$$s_{total}(\mathcal{S}) = \frac{s_h(\mathcal{S}) + s_l(\mathcal{S})}{2}, \tag{4.15}$$

where s_h and s_l denotes, respectively, the holistic and local dissimilarity scores and are defined as follows:

$$s_h(\mathcal{S}) = \left(\frac{1}{n_r} \sum_{i=1}^{n_r} d_{l2}(\mathcal{R}_i, \mathcal{S}) + \min_{i=1,\ldots,n_r} d_{l2}(\mathcal{R}_i, \mathcal{S}) \right) \times \frac{1}{\sqrt{\bar{d}_h}}, \tag{4.16}$$

$$s_l(\mathcal{S}) = \left(\frac{1}{n_r} \sum_{i=1}^{n_r} d_{te}(\mathcal{R}_i, \mathcal{S}) + \min_{i=1,\ldots,n_r} d_{te}(\mathcal{R}_i, \mathcal{S}) \right) \times \frac{1}{\sqrt{\bar{d}_l}}, \tag{4.17}$$

where functions d_{l2} in Equation 4.16 and d_{te} in Equation 4.17 are defined as:

$$d_{l2}(\mathcal{X}, \mathcal{Y}) = \left\| \frac{g(\mathcal{X})}{nhead \times ndim} - \frac{g(\mathcal{Y})}{nhead \times ndim} \right\|_2^2, \tag{4.18}$$

$$d_{te}(\mathcal{X}, \mathcal{Y}) = \frac{\mathrm{dtw}(f(\mathcal{X}), f(\mathcal{Y}))}{\sqrt{|f(\mathcal{X})|^2 + |f(\mathcal{Y})|^2}} \cdot ldim^{-1}, \tag{4.19}$$

where $nhead$ and $ndim$ refer to the number of attention heads and their dimensionality in SP and $ldim$ is the output dimension size of the linear layer in the local branch.

Our local branch verifier is inspired by the one used in DsDTW [14] and the CGRN+CNN Ensemble [22]. However, while these studies normalized the d_{te} by dividing it by the average pairwise distance between only the reference signatures, our experiments in section 5 revealed the benefits of including all the pairwise distances, i.e., both reference and query signatures, in the normalization factor. Thus, we define $\bar{d}_l$ in Equation 4.17 as the average of all the pairwise d_{te}. Following a similar approach for the holistic branch, $\bar{d}_h$ in Equation 4.16 is given by the average pairwise d_{l2} considering both references and query signatures. If the dissimilarity score of a query signature $\mathcal{S}$ is lower than a predefined threshold, we classify the signature as genuine; otherwise, we classify it as a forgery.

4.6 HoLoDANN for finger-written OSV

Domain Adaptation Neural Network (DANN), originally introduced in [62], incorporates a domain classifier module into a deep neural network to address the problem of domain shift. This issue arises when two datasets—typically a labeled source domain and an unlabeled target domain—contain similar classes but differ in their data distributions due to factors such as varying acquisition devices or environmental conditions. DANN mitigates this discrepancy by encouraging the network to learn domain-invariant feature representations. As a result, the model can generalize more effectively to the target domain, enabling knowledge transfer without requiring labeled data from the target domain. In this work, we incorporate a DANN mechanism to facilitate knowledge transfer from stylus-written signatures to approach the problem of finger-written OSV.

However, unlike the original DANN setting, where the source and target domains share the same label space, here we employ DANN within a deep metric learning pipeline whose objective is to separate genuine from forged signatures. Specifically, we extend the HoLoSig architecture by integrating the key components from DANN: the introduction of a gradient reversal layer and a domain classifier. This new branch is designed to distinguish between finger-written and stylus-written signatures, as illustrated in Figure 4.4.

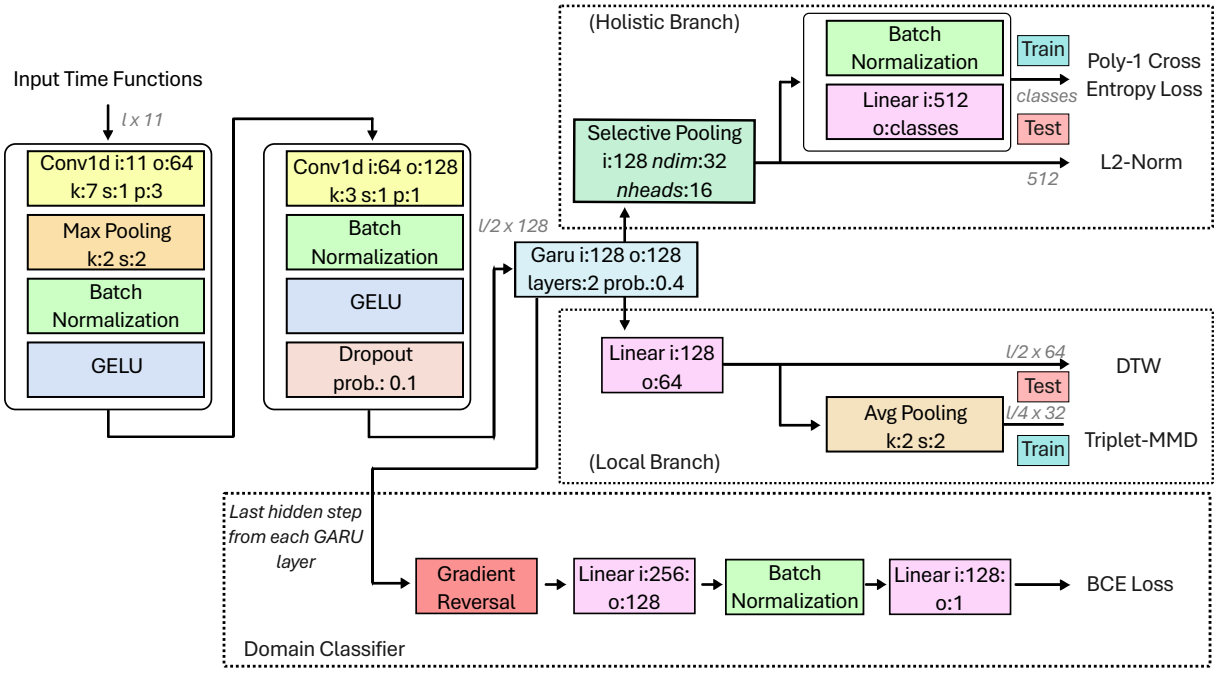


Figure 4.4: Overall HoLoDANN architecture.

The goal of the domain classifier is to minimize the Binary Cross-Entropy (BCE) loss

through the backpropagation algorithm. However, the gradient reversal layer, introduced in [62], multiplies the gradient by a negative constant during backpropagation, causing the shared backbone to maximize—rather than minimize—the BCE loss. This setup creates a min-max game: while the domain classifier branch aims to distinguish between signatures written with fingers and those written with a stylus, the shared backbone is encouraged to extract domain-invariant features that prevent such discrimination. In this way, we leverage the large number of stylus-written signatures available in DeepSignDB to address the problem of finger-written OSV.

Therefore, during the forward pass, the gradient reversal layer acts as an identity function outputting the exact same input. In the backward pass, however, the gradient is multiplied by a negative constant, in this work, we consider this constant as -1 to simply invert its direction. Thus, the gradient layer, as defined in [62], is defined in its forward and backward form respectively in equations 4.20 and 4.21.

$$R(\mathbf{x}) = \mathbf{x}, \tag{4.20}$$

$$\frac{\partial R}{\partial \mathbf{x}} = -\mathbf{I}, \tag{4.21}$$

where $\mathbf{x}$ is the input and $\mathbf{I}$ the identity matrix. Specifically, the input to the domain classifier consists of the last two hidden states of each GARU layer from the shared backbone. Furthermore, the inputs for the HoLoDANN architecture are the same from the HoLoSig architecture, with the exception of the pressure information, which is not present in DeepSignDB finger-written signatures. Thus, HoLoDANN overall loss function is the same of HoLoSig described in Section 4.3 with the addition of the BCE Loss and is defined as:

$$\mathcal{L} = \bar{\mathcal{L}}_{tri} \times \zeta + \overline{\text{MMD}} \times \alpha + \mathcal{L}_{Poly-1} \times \psi + \bar{\mathcal{L}}_{BCE} \times \nu, \tag{4.22}$$

where $\zeta \in \mathbb{R}^+$, $\alpha \in \mathbb{R}^+$, $\psi \in \mathbb{R}^+$ and $\nu \in \mathbb{R}^+$ are weighting factors that regulate the contribution of each loss function during optimization.

Moreover, the HoLoDANN architecture uses the GELU activation function in place of ReLU. Unlike ReLU, which outputs zero for any negative input, the GELU activation function applies a smooth suppression of negative values. This property allows GELU to retain small negative activations. In our experiments, presented in Chapter 5, GELU outperformed ReLU, contributing to better overall performance. Additionally, we observed that increasing the dropout rate in the GARU layer further enhanced model generalization. The GELU activation function is formally defined as:

$$\mathrm{GELU}(x) = x \cdot \Phi(x), \tag{4.23}$$

where $\Phi(x)$ denotes the CDF of the standard Gaussian distribution.

Finally, since both finger-written and stylus-written signatures are required for domain adaptation, each HoLoDANN batch must contain at least one mini-batch of each of these types. As discussed in Chapter 5, DeepSignDB contains significantly more stylus-written signatures than finger-written ones. Consequently, we opted to construct each epoch by sampling random stylus-written mini-batches without replacement and random finger-written mini-batches with replacement.

Chapter 5

Experiments and Results

5.1 Dataset and Experimental Protocol

DeepSignDB [33] is currently the largest online signature dataset, containing signatures from 1,526 writers. It is composed of the union of the MCYT [30], BiosecurID [32], Biosecure DS2 [63], e-BioSign DS1 [52], and e-BioSign DS2 datasets. To allow fair comparisons across different studies, DeepSignDB defines a strict experimental protocol, which states which combinations of reference and query signatures should be used to determine the system's EER. DeepSignDB experimental protocol is rigorously followed in this study.

Table 5.1: Specification of DeepSignDB regarding the number of users, genuine signatures, skilled forgeries and total amount of stylus-written signatures for each dataset of DeepSignDB.
* As previously mentioned, we do not have access to the Biosecure DS2 training set.
** Each user from eBioSign DS1 provided samples using five different devices. Therefore, each partition has 8 genuine and 6 skilled forgeries, totalizing 40 genuine signatures and 30 skilled forgeries per user.

Dataset	Signatures per user		Training Set		Evaluation Set	
	Genuine	Skilled Forgeries	#Writers	Total of Signatures	#Writers	Total of Signatures
MCYT	25	25	230	11500	100	5000
BiosecureID	16	12	268	7504	132	3696
Biosecure DS2*	30	20	510	25500	140	7000
eBioSign DS1**	8 (40)	6 (30)	30	2100	35	2450
eBioSign DS2	8	6	46	644	35	490
DeepSignDB	Varies	Varies	1084	47248 (21748)*	442	18636

Table 5.2: Specification of DeepSignDB regarding the number of users, genuine signatures, skilled forgeries and total amount of signatures for each dataset of DeepSignDB finger-written signatures. Each user from each dataset provided samples using two different devices.

Dataset	Signatures per user		Training Set		Evaluation Set	
	Genuine	Skilled Forgeries	#Writers	Total of Signatures	#Writers	Total of Signatures
eBioSign DS1**	8 (16)	6 (12)	30	840	35	980
eBioSign DS2	8 (16)	6 (12)	46	1288	35	980
DeepSignDB	8 (16)	6 (12)	76	2128	70	1960

It is noteworthy that DeepSignDB is divided into training and evaluation sets with no writer overlap, i.e., none of the writers used to determine the system's EER provide samples during the training stage. Furthermore, the experimental protocol considers scenarios with either one or four reference signatures and attacks made with either random or skilled forgeries. However, it is important to note that we do not use the Biosecure DS2 training set due to licensing constraints. As a result, we train HoLoSig using only the available signatures, following the same approach of the studies presented in [14, 15, 22] since they also do not use Biosecure DS2 training signatures due to the same reason. Nevertheless, we adhere to the experimental protocol and consider Biosecure DS2 signatures in the evaluation set. Table 5.1 and Table 5.2 presents some details about the DeepSignDB regarding stylus-written and finger-written signatures, respectively.

The experimental protocol considers two types of presentation attacks: skilled forgeries and random forgeries. Each presentation attack is evaluated using one of four reference signatures per writer. Specifically, the first four lexicographically ordered signatures from each writer in the evaluation set are selected as reference signatures for computing the Equal Error Rate (EER). In the 4vs1 scenario, the set of four reference signatures is compared against all remaining genuine signatures of the same writer to calculate the FRR. In the 1vs1 scenario, each reference signature is evaluated individually, being compared separately against each non-reference genuine signature from the same writer.

The FAR is computed based on the type of presentation attack. For skilled forgeries, all available skilled forgeries for a given writer are compared against the reference signatures, both in the 4vs1 and 1vs1 scenarios. For random forgeries, one genuine signature from every other writer (i.e., writers who are not the owner of the reference signature) is selected from the same reference-specific dataset and used as the random forgery, both in the 4vs1 and 1vs1 scenarios.

Following these steps, DeepSignDB defines an experimental protocol, provided as a

Table 5.3: Number of comparisons in each DeepSignDB stylus-written evaluation scenario and their respective percentage relative to the total number of comparisons.

Dataset	Skilled Forgeries		Random Forgeries		Total
	4vs1	1vs1	4vs1	1vs1	
MCYT	4600 (31.15%)	18400 (31.15%)	12000 (19.86%)	48000 (19.86%)	83000 (22.08%)
BiosecureID	3168 (21.45%)	12672 (21.45%)	18876 (31.24%)	75504 (31.24%)	110220 (29.32%)
Biosec. DS2	4900 (33.18%)	19600 (33.18%)	21560 (35.69%)	86240 (35.69%)	132300 (35.19%)
eBioDS1 w1	350 (2.37%)	1400 (2.37%)	1330 (2.20%)	5320 (2.20%)	8400 (2.23%)
eBioDS1 w2	350 (2.37%)	1400 (2.37%)	1330 (2.20%)	5320 (2.20%)	8400 (2.23%)
eBioDS1 w3	350 (2.37%)	1400 (2.37%)	1330 (2.20%)	5320 (2.20%)	8400 (2.23%)
eBioDS1 w4	350 (2.37%)	1400 (2.37%)	1330 (2.20%)	5320 (2.20%)	8400 (2.23%)
eBioDS1 w5	350 (2.37%)	1400 (2.37%)	1330 (2.20%)	5320 (2.20%)	8400 (2.23%)
eBioDS2 w2	350 (2.37%)	1400 (2.37%)	1330 (2.20%)	5320 (2.20%)	8400 (2.23%)
DeepSignDB	14768	59072	60416	241664	375920

text file distributed alongside the dataset. Each dataset within DeepSignDB has its own dedicated protocol file. The overall DeepSignDB protocol is constructed by concatenating the individual protocols from all datasets. By varying the decision threshold, we can

Table 5.4: Number of comparisons in each DeepSignDB finger-written evaluation scenario and their respective percentage relative to the total number of comparisons.

Dataset	Skilled Forgeries		Random Forgeries		Total
	4vs1	1vs1	4vs1	1vs1	
eBioDS1 w4	350 (25.14%)	1400 (25%)	1330 (25%)	5320 (25%)	8400 (25.01%)
eBioDS1 w5	350 (25.14%)	1400 (25%)	1330 (25%)	5320 (25%)	8400 (25.01%)
eBioDS2 w5	350 (25.14%)	1400 (25%)	1330 (25%)	5320 (25%)	8400 (25.01%)
eBioDS2 w6	342 (24.58%)	1400 (25%)	1330 (25%)	5320 (25%)	8392 (24.97%)
DeepSignDB	1392	5600	5320	21280	33592

compute the FAR and FRR curves. The EER corresponds to the point where these two curves intersect and is the default metric to measure the verification performance in DeepSignDB experiment protocol. Table 5.3 and 5.4 presents the number of comparisons in each stylus-written and finger-written experiment protocol, respectively.

5.2 Data Augmentation

Neither CNNs nor DTW demonstrate robustness to rotations. Therefore, considering that signatures may not always be written in a straight line, we apply random rotations to the signatures within a range of $-30°$ to $30°$ using a 2D rotation matrix on the x and y input time-dependent functions. By preserving the original pressure information, we can create a new signature by computing the 12 input time-dependent functions presented in Section 4.1. In this study, each signature was augmented by generating a rotated variant, effectively doubling the size of the training set. The augmentation process was performed offline, ensuring that all experiments were conducted using an identical set of signature samples.

5.3 Implementation Details

In each experiment, we repeat the training process using five different random seeds (111, 222, 333, 444, and 555) over 35 epochs and report the results from the epoch that achieved the lowest average EER across the evaluation scenarios for the best model (i.e., the best-performing seed). We adopt several hyperparameters from the setup described in [14] since they demonstrated good results. Specifically, we set the γ in soft-dtw to 5 and the margin m in Equation 4.2 to 1. Also, we set $n_w = 4$, $n_g = 5$, and $n_f = 10$, resulting in four random mini-batches with 16 signatures in each batch, resampled to 100 Hz using a low-pass Butterworth filter, as also done in previous works [14, 15]. Among the n_f forgeries for each writer, we consider 5 skilled and 5 random forgeries. Furthermore, we also use SGD as optimizer with a momentum of 0.9 and an initial learning rate of 0.01, exponentially decayed by a factor of 0.9 after each epoch.

Due to the varying number of genuine and forged signatures in each dataset, it is not always possible to form new mini-batches without repetition. Thus, to avoid repetition, we opt to not include all signatures in the training set in each epoch. The values of η and ξ in Equation 4.9 are both empirically set to 3 and ζ in Equation 4.1 is set to 1.0 as writer separation is our priority during optimization. Similarly, the ν weighting factor in Equation 4.22 is empirically set to 1.0. The following hyperparameters were tuned using a bayesian search: We add $\epsilon = 0.0918$ to the first polynomial coefficient in Poly-1

Cross-Entropy Loss. The weighting factors in Equation 4.1, α, and ψ, are set to 0.3, and 0.171, respectively. Finally, in Equation 4.5, we set $\beta = 1.8$ and $\iota = 0.297$. The baysean search was performed using wandb sweep, the space search considered intervals empirically defined for each hyperaparameter with goal to minimize the EER.

To evaluate the performance of the other alignment losses, we kept the previously mentioned hyperparameters fixed and repeated the experiments using the same random seed. For each loss function, we varied the weighting factor within an empirically defined interval specific to that loss. The selected weighting factors for $\overline{D_{\text{Cramér}}}(p = 1)$, $\overline{D_{\text{Cramér}}}(p = 2)$, and D_{Energy} are 0.09, 0.15, and 0.18, respectively

All linear layers in our model are configured without bias terms. Weight parameters are initialized using a Kaiming uniform distribution, while the biases of the convolutional layers are set to zero. Following the approach of [15], the queries in the SP module are initialized orthogonally. All experiments were conducted using an NVIDIA RTX 4090 GPU with Python and PyTorch taking around 3 minutes to train an epoch which includes rotated signatures.

5.4 HoLoSig Results on stylus-written signatures

5.4.1 Ablation Study

We first assess the impact of the proposed methods in this work. Our experiments consider eight different training scenarios, resulting from the combination of three components that affect the training stage: the use of MMD, rotated signatures, and the time-dependent functions x and y mean-normalized, as opposed to the commonly used velocities ($\dot{x}$ and $\dot{y}$) that are normalized to zero mean and unit variance, as described in the literature [14, 15]. Table 5.5 presents the results for each combination using three different verifiers, evaluating the effectiveness of the proposed score normalization method (Chapter 4) and the combination of holistic and local representations during the verification stage.

Table 5.5 shows that, in general, the experiments with MMD (lines 4-7) present better EER compared to their equivalent versions without MMD (lines 0-3), especially for verifier V_3, where this is always the case. Furthermore, the use of the proposed inputs and rotated signatures alone (lines 1-2) yields similar results with the same verifier. However, when both techniques are used together (line 3), we observe a considerable performance improvement against skilled forgery attacks (compared to lines 0-2). The results also show that MMD contributes significantly against random forgery attacks (lines 4-7), especially when only one reference is available. Finally, the combination of MMD and rotated

Table 5.5: Equal Error Rate (EER) (in %) for each of the four DeepSignDB experimental protocols with stylus input, across training scenarios and verifiers V_i. V_1 uses only the local branch and normalizes the score based solely on the pairwise scores of the reference signatures. No normalization is performed when only one reference is available. V_2 also uses only the local branch but normalizes scores based on the pairwise scores of both the references and query signatures. V_3 uses both branches with the same normalization method as V_2.

Line #	MMD	Rotation	Inputs	Skilled Forgeries						Random Forgeries					
				4vs1			1vs1			4vs1			1vs1		
				V_1	V_2	V_3	V_1	V_2	V_3	V_1	V_2	V_3	V_1	V_2	V_3
0	×	×	×	2.55	2.31	*2.21*	4.08	4.08	*3.83*	0.90	0.62	*0.54*	1.26	1.26	*1.22*
1	×	×	✓	2.33	2.23	*2.16*	3.95	3.95	*3.83*	0.86	0.65	*0.61*	1.35	1.35	*1.25*
2	×	✓	×	2.25	2.17	*2.10*	4.08	4.08	*3.88*	0.83	0.59	*0.56*	1.28	1.28	*1.21*
3	×	✓	✓	2.15	1.99	*1.94*	3.73	3.73	*3.66*	0.90	0.74	*0.64*	1.40	1.40	*1.32*
4	✓	×	×	2.42	2.28	*2.19*	4.05	4.05	*3.79*	0.75	0.55	*0.48*	0.94	0.94	***0.84***
5	✓	×	✓	2.35	2.30	*2.14*	3.83	3.83	*3.51*	0.68	0.55	*0.48*	1.02	1.02	*0.89*
6	✓	✓	×	2.18	2.00	*1.88*	3.65	3.65	*3.35*	0.69	0.47	*0.44*	0.95	0.95	*0.85*
7	✓	✓	✓	1.95	1.88	***1.73***	3.47	3.47	***3.29***	0.72	0.47	***0.43***	0.94	0.94	*0.89*

signatures (line 6) results in a considerable improvement against skilled forgery attacks, which is further enhanced by the addition of the proposed inputs.

It is noteworthy that the normalization method does not affect verifiers V_1 and V_2 when only a single reference is available, which explains their identical results in this scenario, since in V_1 no normalization is performed in this case and in V_2 the normalization consists solely on dividing the score by its square root. The results presented by V_2 are better than those of V_1 across all experiments with four references, demonstrating the benefits of the proposed normalization method. Similarly, the results presented by V_3 are better than those of V_2 across all experiments – including when only one reference is available – highlighting the advantages of combining both holistic and local representations during the verification stage, particularly when only one reference is available. All remaining results reported in this paper use V_3 as the signature verifier.

To assess the reproducibility of the proposed method, Table 5.6 reports the results obtained by repeating the training process with five different random seeds. On average, the inclusion of MMD appears to slightly reduce the standard deviation compared to the variant without it. Nevertheless, the results demonstrate the overall consistency of the proposed method across all ablation experiments.

Table 5.6: Reproducibility of HoLoSig. For each experiment we report the average Equal Error Rate (EER) (in %) and standard deviation achieved in each of the four DeepSingDB experiment protocols in the stylus scenario across the repetition of the experiment with 5 different random seeds.

Line #	MMD	Rotation	Inputs	Skilled Forgeries		Random Forgeries		Average
				4vs1	1vs1	4vs1	1vs1	
0	×	×	×	2.34 ± 0.07	0.58 ± 0.05	3.97 ± 0.08	1.30 ± 0.07	2.05 ± 0.06
1	×	×	✓	2.27 ± 0.07	0.60 ± 0.04	4.02 ± 0.14	1.36 ± 0.10	2.06 ± 0.07
2	×	✓	×	2.17 ± 0.09	0.63 ± 0.06	4.03 ± 0.16	1.39 ± 0.10	2.06 ± 0.08
3	×	✓	✓	1.98 ± 0.03	0.61 ± 0.05	3.76 ± 0.11	1.33 ± 0.02	1.92 ± 0.03
4	✓	×	✓	2.25 ± 0.07	0.50 ± 0.04	3.75 ± 0.05	0.94 ± 0.08	1.86 ± 0.02
5	✓	✓	×	2.14 ± 0.08	0.47 ± 0.03	3.57 ± 0.11	0.94 ± 0.04	1.78 ± 0.03
6	✓	✓	✓	1.98 ± 0.10	0.45 ± 0.01	3.47 ± 0.08	0.88 ± 0.06	1.70 ± 0.06
7	✓	×	×	1.79 ± 0.06	0.47 ± 0.04	3.35 ± 0.09	0.93 ± 0.03	1.64 ± 0.04

5.4.2 Error Analysis: Separation and Alignment

To evaluate the impact of the proposed changes in writer separation and threshold alignment, we decompose the EER into the sum of the separation error and alignment error in Table 5.7, following the process described in Algorithm 1. To determine the separation error, we first compute the EER and the optimal threshold for each writer, following the process used in a WD system with a specific threshold for each writer. Next, we compute the global EER by adjusting each signature score: we subtract the optimal threshold (computed in the previous step) associated with the writer from the reference signature. This subtraction shifts the separation from around the local threshold to zero for all writers, without affecting writer separation. As a result, a signature that was previously classified as genuine using the local threshold will now receive a negative score, since the requirement to be classified as genuine is a score lower than the threshold. Conversely, a signature that was previously classified as a forgery will now receive a non-negative score, since the requirement to be classified as forgery is a score greater than or equal to the threshold. Therefore, we enforce all local optimal thresholds to be zero, and as a consequence, the global threshold is also set to zero. We call this the separation error since it reflects only the error related to the method's inability to distinguish between genuine and forged signatures. The alignment error, on the other hand, is given by subtracting the separation error from the total error and reflects the error resulting from misalignment in the optimal writer-specific thresholds.

Table 5.7 shows a reduction in alignment error when comparing a model with MMD

Table 5.7: Separation and Alignment Equal Error Rate (EER) (in %).

Line #	MMD	Rotation	Inputs	Separation EER				Alignment EER			
				Skilled		Random		Skilled		Random	
				4vs1	1vs1	4vs1	1vs1	4vs1	1vs1	4vs1	1vs1
0	×	×	×	0.82	1.75	0.18	0.46	1.39	2.08	0.36	0.76
1	×	×	✓	0.97	1.90	0.18	0.52	1.19	1.93	0.43	0.73
2	×	✓	×	0.83	1.77	0.17	0.46	1.27	2.11	0.39	0.75
3	×	✓	✓	0.86	1.71	0.21	0.52	1.08	1.95	0.43	0.80
4	✓	×	×	0.92	1.85	0.15	0.30	1.27	1.94	0.33	0.54
5	✓	×	✓	0.89	1.70	0.11	0.26	1.25	1.81	0.37	0.60
6	✓	✓	×	0.86	1.62	0.12	0.27	1.02	1.73	0.32	0.58
7	✓	✓	✓	0.91	1.59	0.10	0.28	0.82	1.70	0.33	0.61

Algorithm 1 Compute Separation Error and Alignment Error

1: **Input:** Signature scores for all comparisons
2: **Output:** Separation Error, Alignment Error
3: **for** each writer w **do**
4: Compute writer-specific EER and optimal threshold th_w
5: **end for**
6: **for** each signature score s **do**
7: Identify associated writer w
8: Adjust score: $s' \leftarrow s - th_w$
9: **end for**
10: Define global threshold $th_{global} \leftarrow 0$
11: Compute Global EER using adjusted scores s' and threshold th_{global}
12: Separation Error = Global EER with $th_{global} = 0$
13: Compute Total Global EER using the input scores (without adjustment or fixed threshold)
14: Alignment Error = Total Global EER − Separation Error
15: **return** Separation Error, Alignment Error

(lines 4-7) to its equivalent version without MMD (lines 0-3), with the exception of the 4vs1 skilled scenario when the proposed inputs are used without rotation (lines 1 and 5). The use of rotated signatures and proposed inputs (line 3) demonstrates better alignment against skilled forgeries compared to when none of the evaluated techniques are used (line 0). However, this improvement also leads to an increase in the alignment error against random forgeries, which is mitigated by the presence of MMD (lines 4 and 7). Comparing lines 0 and 7, the results indicate that the proposed techniques positively contribute by reducing the alignment error while also achieving good separation. Notice that, in general,

the alignment error is greater than the separation error.

To better visualize the effect of MMD, Figure 5.1 presents the distribution of dissimilarity scores for models trained with and without MMD across the four DeepSignDB stylus experimental protocols. The figure shows that models trained with MMD exhibit a lower standard deviation in the dissimilarity scores, suggesting that the dissimilarity scores are being effectively shifted to a common region, reducing the overlap between the scores of genuine and forged query signatures, thereby enhancing discriminability with a global threshold.

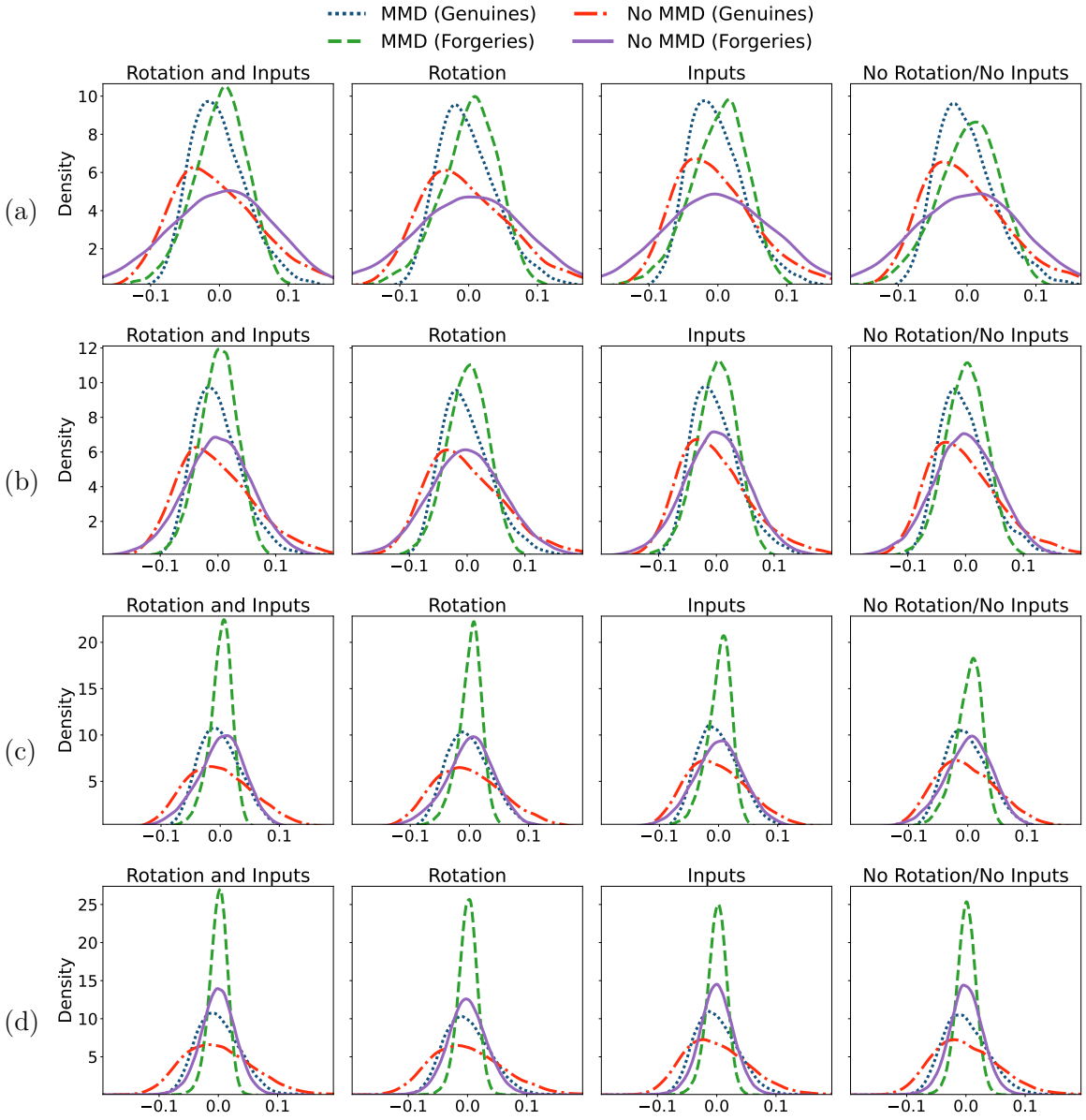


Figure 5.1: Distribution of dissimilarity scores for different experiments with or without MMD on DeepSignDB: (a) 4vs1 skilled, (b) 4vs1 random, (c) 1vs1 skilled, (d) 1vs1 random. The distributions were normalized to have zero mean for better visualization.

5.4.3 Beyond MMD: Evaluation of Alternative Alignment Losses

To evaluate the performance of the alternative alignment losses proposed as substitutes for MMD in Chapter 4, Table 5.8 reports the EER obtained in the stylus scenario. Although the results confirm the superiority of MMD, the overall performance across the different losses remains relatively similar, demonstrating the viability of using statistical divergence metrics over the dissimilarity scores to address the alignment error. Similarly, Table 5.9 reports the separation and alignment errors associated with each alignment loss, while Figure 5.2 compares the distributions of dissimilarity scores obtained using MMD with those from each of the alternative metrics. The remaining results in this work consider MMD as the alignment loss.

Table 5.8: EER (in %) achieved by each alignment loss evaluated in this study under the stylus scenario.

Alignment Loss	Skilled Forgeries		Random Forgeries		Average
	4vs1	1vs1	4vs1	1vs1	
Cramér (p=1)	1.83	3.29	0.49	0.95	1.64
Cramér (p=2)	1.89	3.42	0.52	1.01	1.71
Energy	1.86	3.37	0.48	0.93	1.66
MMD	1.73	3.29	0.43	0.89	1.59

Table 5.9: Separation Error and Alignment Error (in %) achieved by each alignment loss evaluated in this study in the stylus scenario.

Alignment Loss	Seperation Error				Alignment Error			
	Skilled		Random		Skilled		Random	
	4vs1	1vs1	4vs1	1vs1	4vs1	1vs1	4vs1	1vs1
Cramer (p=1)	0.84	1.61	0.13	0.35	1.05	1.80	0.39	0.66
Cramer (p=2)	0.79	1.60	0.10	0.24	1.04	1.68	0.39	0.71
Energy	0.92	1.73	0.15	0.33	0.93	1.63	0.33	0.60
MMD	0.91	1.59	0.10	0.28	0.82	1.70	0.33	0.61

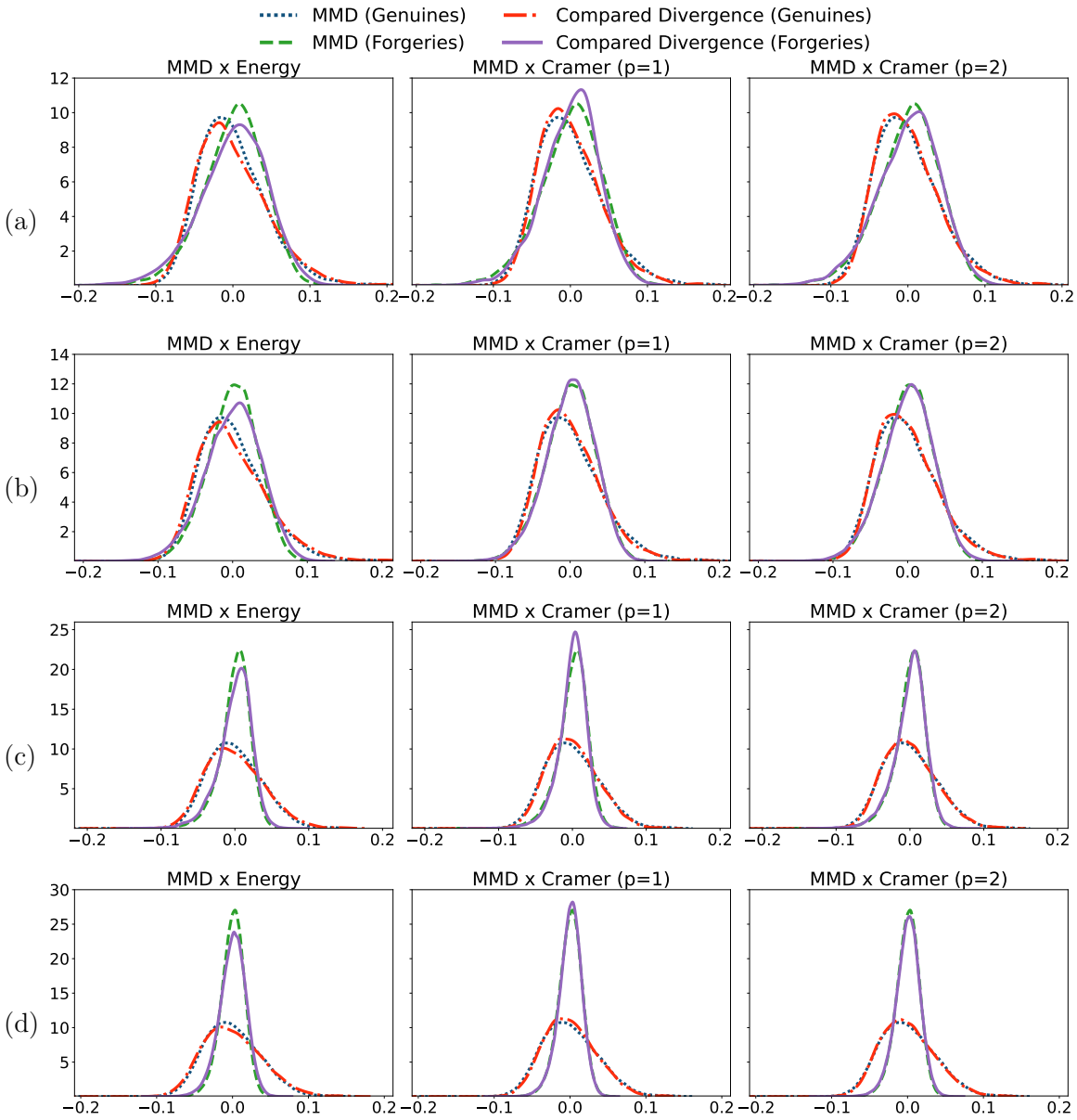


Figure 5.2: Distribution of dissimilarity scores for different experiments comparing MMD with Energy Distance and Cramer Distance with $\mathbf{p} = 1$ and $\mathbf{p} = 2$: (a) 4vs1 skilled, (b) 4vs1 random, (c) 1vs1 skilled, (d) 1vs1 random. The distributions were normalized to have zero mean for better visualization.

5.4.4 Experimental Results on the datasets of DeepSignDB

Table 5.10 presents the EER achieved by HoLoSig across the subsets of DeepSignDB. The results show that HoLoSig performance on the eBioSign datasets is worse compared to MCYT, BiosecurID, and Biosecure DS2. We attribute this to two factors: 1) the eBioSign datasets have significantly fewer samples in the training set than the other datasets, and 2) each eBioSign dataset was acquired using different devices, suggesting the need for proper domain adaptation. It is worth noting that while we do not have access to the

BioSecure DS2 training dataset, its signatures were collected with the same device model used in BiosecurID.

Table 5.10: EER (%) obtained by HoLoSig after training on the full training set with all proposed techniques, evaluated separately on each DeepSignDB stylus-written dataset.

subset	Skilled Forgeries		Random Forgeries		Average
	4vs1	1vs1	4vs1	1vs1	
MCYT	1.41	2.75	0.14	0.47	1.19
BiosecureID	0.82	1.41	0.12	0.33	0.67
Biosecure DS2	1.59	3.14	0.72	1.20	1.66
eBioSign DS1 w1	1.79	6.94	0.17	1.44	2.59
eBioSign DS1 w2	2.38	4.49	0.57	1.01	2.11
eBioSign DS1 w3	3.45	5.77	0.36	0.66	2.56
eBioSign DS1 w4	2.62	5.27	0.08	0.75	2.18
eBioSign DS1 w5	4.88	5.65	0.74	1.60	3.22
eBioSign DS2 w2	1.67	3.99	0.36	2.51	2.13
DeepSignDB	1.73	3.29	0.43	0.89	1.59

5.4.5 Generalization Capability: Leaving One Dataset Out

To assess the generalization capability of HoLoSig, we adopt a Leave One Dataset Out (LODO) evaluation strategy using the datasets from DeepSignDB. For each experiment, we retrain HoLoSig excluding all signatures from one specific dataset. The trained model is then evaluated on the evaluation set corresponding to the excluded dataset. It is important to note that DeepSignDB contains no writer overlap between its training and evaluation sets. Therefore, this experiment is not intended to study overfitting effects. Instead, our goal is to evaluate the hyperparameters and the framework's ability to generalize across different devices and acquisition protocols, as each dataset corresponds to a distinct capture condition. For further details on the specific characteristics of each dataset, we refer the reader to [30, 32, 63, 52, 33].

Table 5.11 presents the results of our LODO strategy. It is important to note that, as mentioned in Section 5.1, we do not have access to the Biosecure DS2 training set. Consequently, none of our experiments include signatures from this dataset. As a result, each LODO experiment effectively excludes the training signatures from two datasets from DeepSignDB: Biosecure DS2 and the target dataset being evaluated.

Naturally, the results presented in Table 5.11 show a degradation in performance when each sub-dataset is excluded from the training set. It is particularly interesting to observe

that the average performance when removing the eBioSign DS2 dataset remains almost unchanged, varying only from 2.16 to 2.13. However, the results reveal that this stability is largely due to the performance on skilled forgeries. While the LODO strategy presented a better EER against skilled forgeries, it led to a notable error improvement in the 1vs1 scenario for random forgeries. In fact, this trade-off was a recurring phenomenon observed throughout this study: improvements against skilled forgeries often came at the cost of worse performance against random forgeries. As presented in Section 5.4.6, this trade-off is also evident in some state-of-the-art methods.

Table 5.11: Comparison of the LODO strategy results with the results obtained when training on all available signatures in our training set. The reported values correspond to the global EER (in %). We also report the average degradation in error. Note: eBio stands for eBioSign.

Evaluated Dataset	Train w/o the evaluated dataset					Train with all available data					Avg. Degra.
	Skilled		Random		Avg.	Skilled		Random		Avg.	
	4vs1	1vs1	4vs1	1vs1		4vs1	1vs1	4vs1	1vs1		
MCYT	1.83	3.35	0.13	0.70	1.50	1.41	2.75	0.14	0.47	1.19	0.31
BiosecureID	1.11	1.70	0.20	0.42	0.86	0.82	1.41	0.12	0.33	0.67	0.19
eBio DS1 w1	2.62	6.79	0.78	1.41	2.90	1.79	6.94	0.17	1.44	2.59	0.31
eBio DS1 w2	3.10	3.72	0.57	0.96	2.09	2.38	4.49	0.57	1.01	2.11	-0.02
eBio DS1 w3	2.86	6.07	0.36	0.75	2.51	3.45	5.77	0.36	0.66	2.56	-0.05
eBio DS1 w4	3.69	5.71	0.82	1.05	2.82	2.62	5.27	0.08	0.75	2.18	0.64
eBio DS1 w5	5.95	7.38	1.09	2.48	4.23	4.88	5.65	0.74	1.60	3.22	1.01
eBio DS2 w2	1.43	3.75	0.40	3.07	2.16	1.67	3.99	0.36	2.51	2.13	0.03

A similar behavior of the eBioSign DS2 dataset can be seen in the eBioSign DS1 w2 and w3 partitions, where the average performance under the LODO strategy remains close to that obtained when using all available data. In these cases, the stability in the average is explained by performance gains in specific scenarios: the 1vs1 skilled scenario in w2 and the 4vs1 skilled scenario in w3, with nearly identical performance in the random forgery scenarios. In contrast, for partitions w4 and w5 of eBioSignDB, a considerable degradation in performance is observed when applying the LODO strategy.

However, we highlight that the average performance degradation—measured as the increase in EER—of the LODO strategy on the MCYT and BiosecureID datasets is particularly promising. As discussed in Section 5.1, MCYT and BiosecurID are the training

datasets with the largest number of signatures available to us, representing 52.88% and 34.5% of the total training data, respectively. Remarkably, even after excluding more than half of the training data, the average performance degradation of HoLoSig on MCYT is only 0.31%, despite the fact that its acquisition device, protocol, and writers were completely unseen during training. Similarly, for BiosecurID, the degradation is just 0.19% on average, even after removing one third of the training data. These results suggest that, while HoLoSig may indeed benefit from having training data acquired under conditions similar to the evaluated sets, the total quantity of available signatures for training plays a more significant role in overall performance than domain-specific acquisition conditions. These results highlight HoLoSig's ability to generalize not only to unseen writers, as expected from a WI system, but also to entirely unseen devices and acquisition protocols. Furthermore, they demonstrate the robustness and generalization capability of the various hyperparameters discussed in Section 5.3. For completeness, Table 5.12, Table 5.13, Table 5.14, and Table 5.15 present the detailed results of each LODO experiment across the different DeepSignDB stylus evaluation protocols.

Table 5.12: EER (in %) for each experimental protocol on the DeepSignDB datasets in the LODO setting, with training performed without the MCYT dataset.

DTW	Skilled Forgeries		Random Forgeries		Average
	4vs1	1vs1	4vs1	1vs1	
MCYT	1.83	3.35	0.13	0.70	1.50
BiosecureID	0.95	1.64	0.20	0.42	0.80
Biosecure DS2	2.00	4.14	0.64	1.41	2.05
eBioSign DS1 w1	3.10	6.28	0.04	1.58	2.75
eBioSign DS1 w2	2.62	4.70	0.13	0.71	2.04
eBioSign DS1 w3	3.45	6.10	0.40	1.21	2.79
eBioSign DS1 w4	3.10	5.71	0.17	0.69	2.42
eBioSign DS1 w5	4.41	7.08	0.57	1.71	3.44
eBioSign DS2 w2	1.43	4.70	0.36	2.89	2.35
DeepSignDB	2.07	3.78	0.47	0.95	1.82

Table 5.13: EER (in %) for each experimental protocol on the DeepSignDB datasets in the LODO setting, with training performed without the BiosecurID dataset

DTW	Skilled Forgeries		Random Forgeries		Average
	4vs1	1vs1	4vs1	1vs1	
MCYT	1.65	2.77	0.11	0.61	1.29
BiosecureID	1.11	1.70	0.20	0.42	0.86
Biosecure DS2	1.96	3.84	0.73	1.25	1.95
eBioSign DS1 w1	2.86	7.56	0.74	1.49	3.16
eBioSign DS1 w2	2.38	3.93	0.13	0.71	1.79
eBioSign DS1 w3	2.62	6.07	0.44	0.63	2.44
eBioSign DS1 w4	3.69	6.28	0.08	0.86	2.73
eBioSign DS1 w5	3.10	5.57	0.44	1.80	2.73
eBioSign DS2 w2	0.83	4.08	0.36	2.11	1.85
DeepSignDB	2.01	3.56	0.45	1.00	1.75

Table 5.14: EER (in %) for each experimental protocol on the DeepSignDB datasets in the LODO setting, with training performed without the eBioSign DS1

DTW	Skilled Forgeries		Random Forgeries		Average
	4vs1	1vs1	4vs1	1vs1	
MCYT	1.52	3.06	0.23	0.59	1.35
BiosecureID	0.88	1.31	0.15	0.33	0.67
Biosecure DS2	1.53	3.39	0.83	1.19	1.74
eBioSign DS1 w1	2.62	6.79	0.78	1.41	2.90
eBioSign DS1 w2	3.10	3.72	0.57	0.96	2.09
eBioSign DS1 w3	2.86	6.07	0.36	0.75	2.51
eBioSign DS1 w4	3.69	5.71	0.82	1.05	2.82
eBioSign DS1 w5	5.95	7.38	1.09	2.48	4.23
eBioSign DS2 w2	2.26	4.29	0.36	2.71	2.41
DeepSignDB	1.92	3.74	0.52	1.09	1.81

Table 5.15: EER (in %) for each experimental protocol on the DeepSignDB datasets in the LODO setting, with training performed without the eBioSign DS2

DTW	Skilled Forgeries		Random Forgeries		Average
	4vs1	1vs1	4vs1	1vs1	
MCYT	1.41	2.92	0.13	0.39	1.21
BiosecureID	0.98	1.56	0.18	0.33	0.76
Biosecure DS2	1.49	3.23	0.56	1.00	1.57
eBioSign DS1 w1	3.45	6.79	0.04	1.62	2.98
eBioSign DS1 w2	3.45	4.79	0.17	1.04	2.36
eBioSign DS1 w3	3.69	6.43	0.44	1.16	2.93
eBioSign DS1 w4	4.29	5.27	0.17	1.09	2.70
eBioSign DS1 w5	5.48	6.79	1.09	1.92	3.82
eBioSign DS2 w2	1.43	3.75	0.40	3.07	2.16
DeepSignDB	1.83	3.51	0.50	0.95	1.70

5.4.6 Comparison with the State-of-the-Art

Table 5.16 presents the performance of different methods on DeepSignDB. In this table, the baseline HoLoSig framework refers to the experiment where HoLoSig is trained on the full training set, but without MMD, the proposed input features, and rotated signatures. We also report the results of our LODO experiments. Notably, all of our configurations outperform the state-of-the-art on average, demonstrating the robustness of combining holistic and local representations—even when trained without additional components such as alignment loss, data augmentation, or when using reduced training data. The baseline framework presents particularly strong results against skilled forgeries, especially when only one reference is available.

Interestingly, the LODO experiments—which include MMD, proposed inputs and rotated signatures, but with less training data—outperform the baseline, showing that the inclusion of these removed components has a stronger impact on performance than simply increasing the amount of training data.

It is also worth highlighting that the average performance of the LODO experiment on MCYT is almost identical to that of the LODO experiment on eBioSign DS1, despite MCYT representing more than half (52.88%) of the total training data and eBioSign DS1 accounting for only 9.66%. This outcome is a consequence from a discrepancy in performance when comparing random and skilled forgeries. Specifically, the results obtained without MCYT are better than those without eBioSign DS1 in the random forgery sce-

Table 5.16: EER (in %) of the proposed method compared with the state-of-the-art on DeepSignDB stylus scenario. The baseline version of HoLoSig was not trained using MMD, proposed inputs and rotated signatures. We also report our LODO experiments. Arrows denote the decrease or increase in EER compared to the full proposed method.

Method	Skilled Forgeries		Random Forgeries		Average
	4vs1	1vs1	4vs1	1vs1	
TA-RNN [33]	3.33 (1.60 ↓)	4.02 (0.73 ↓)	0.60 (0.17 ↓)	1.50 (0.61 ↓)	2.40 (0.81 ↓)
DsDTW [14]	2.54 (0.81 ↓)	4.04 (0.75 ↓)	0.97 (0.54 ↓)	1.69 (0.80 ↓)	2.31 (0.72 ↓)
SynSig2Vec [15]	2.54 (0.81 ↓)	4.08 (0.79 ↓)	0.48 (0.05 ↓)	0.84 (0.05 ↑)	1.99 (0.40 ↓)
CGRN+CNN [22]	2.24 (0.51 ↓)	4.19 (0.90 ↓)	0.50 (0.07 ↓)	1.00 (0.11 ↓)	1.98 (0.39 ↓)
HoLoSig (baseline)*	2.21 (0.48 ↓)	3.83 (0.54 ↓)	0.54 (0.11 ↓)	1.22 (0.33 ↓)	1.95 (0.36 ↓)
HoLoSig (No MCYT)	2.07 (0.34 ↓)	3.78 (0.49 ↓)	0.47 (0.04 ↓)	0.95 (0.06 ↓)	1.82 (0.23 ↓)
HoLoSig (No BiosecurID)	2.01 (0.28 ↓)	3.56 (0.27 ↓)	0.45 (0.02 ↓)	1.00 (0.11 ↓)	1.75 (0.16 ↓)
HoLoSig (No ebioDS1)	1.92 (0.19 ↓)	3.74 (0.45 ↓)	0.52 (0.09 ↓)	1.09 (0.20 ↓)	1.81 (0.22 ↓)
HoLoSig (No ebioDS2)	1.83 (0.10 ↓)	3.51 (0.22 ↓)	0.50 (0.07 ↓)	0.95 (0.06 ↓)	1.70 (0.11 ↓)
HoLoSig (Full)	***1.73***	***3.29***	***0.43***	***0.89***	***1.59***

narios. Conversely, the results without eBioSign DS1 outperform those without MCYT in the skilled forgery scenarios. This suggests that eBioSign DS1 signatures provide valuable information that helps the model distinguish random forgeries, while MCYT signatures contribute more significantly to the identification of skilled forgeries.

This observation is further supported by the LODO experiment in which BiosecurID signatures are not used, when we observe an average better performance, likely resulting from a more balanced trade-off between random and skilled forgery detection. For example, in the 4vs1 skilled scenario, the LODO experiment without MCYT (2.07%) performs considerably worse than the LODO without eBioSign DS1 (1.92%), while the LODO without BiosecureID (which includes signatures from both MCYT and eBioSign DS1) shows intermediate performance between the two (2.01%). A similar trend appears in the 1vs1 random scenario, where the LODO without MCYT (0.95%) performs better than the LODO without eBioSign DS1 (1.09%), again with the LODO without BiosecureID falling in between (1.00%).

The LODO experiment excluding eBioSign DS2 presents the best average EER among all LODO experiments. However, despite representing only 2.96% of the total training set, the inclusion of eBioSign DS2 signatures allows HoLoSig to achieve significantly better results across all evaluated scenarios and surpass the state-of-the-art by a considerable margin, specially against skilled forgeries.

Overall, these LODO experiments suggest that carefully selecting training signatures—specifically those providing the most valuable information for the model—can have a greater positive impact on performance than simply increasing the quantity of training data. This is an encouraging finding, especially considering that online signature data is typically difficult to collect due to privacy and sensitivity concerns and the most datasets available to the research community have much fewer samples than datasets for other biometric traits. Therefore, we believe that by removing less informative signatures and retaining only those that contribute meaningfully to model learning, HoLoSig could achieve even better results.

5.5 HoLoDANN Results on finger-written signatures

5.5.1 Ablation Study

We first evaluate the performance of HoLoDANN compared to HoLoSig through an ablation study presented in Table 5.17. Line 0 reports the results obtained by the original HoLoSig framework described in Section 5.4, i.e., HoLoSig trained exclusively on stylus-based signatures, without any domain adaptation or fine-tuning, and with the pressure input fixed at a constant value of 1, since DeepSignDB finger-written signatures lack pressure information. This setup aligns with the standard evaluation protocol used in most recent state-of-the-art works on DeepSignDB [14, 15, 22, 33], and serves as our baseline for evaluating HoLoDANN. All subsequent models in the ablation study are trained without pressure information, utilizing only the remaining 11 time-dependent functions as input features.

Next, in line 1, we present the results of HoLoSig when trained solely on finger-written signatures. Interestingly, compared to line 0, the Equal Error Rate (EER) for skilled forgery attacks decreases, while the opposite trend is observed for random forgeries. On average, HoLoSig trained on the smaller set of finger-written signatures (2,128 samples) achieves slightly better overall performance than the model trained on the much larger set of stylus-based signatures (21,748 samples).

In line 2, we train HoLoSig using a combination of both finger-written and stylus-written signatures. This setup presents a considerable improvement over the previous

Table 5.17: EER (%) results for different experiments under the DeepSignDB protocols in the finger scenario.

	Parameters	Training details	Skilled		Random		Avg.
			4vs1	1vs1	4vs1	1vs1	
0	Same as HoLoSig	Stylus only, no DANN	7.93	13.72	0.36	1.44	5.86
1	Same as HoLoSig	Finger only, no DANN	6.40	11.70	1.57	3.20	5.72
2	Same as HoLoSig	Stylus+Finger, no DANN	5.68	10.77	0.81	1.95	4.80
3	Same as HoLoSig	Stylus+Finger+DANN	5.80	10.52	0.20	2.01	4.63
4	GELU instead ReLU $\eta = 2, \xi = 5, \alpha = 0.13$ GARU dropout=0.4	Finger only, no DANN	6.25	11.89	1.97	3.20	5.83
5	GELU instead ReLU $\eta = 2, \xi = 5, \alpha = 0.13$ GARU dropout=0.4	Stylus+Finger, no DANN	5.53	10.19	1.06	2.41	4.80
6	GELU instead ReLU $\eta = 2, \xi = 5, \alpha = 0.13$ GARU dropout=0.4	Stylus+Finger+DANN	5.65	9.99	0.55	1.78	4.49

two configurations (lines 0 and 1), especially in against skilled forgeries. We believe this improvement is due to the model's ability to leverage the specific characteristics of finger-written signatures while also benefiting from the larger quantity of stylus-written samples. When DANN is incorporated, as shown in line 3, we observe a further performance gain, particularly in the 4vs1 random forgery scenario.

Our experiments revealed that certain hyperparameters can be better defined when the domain classifier is present, as well as the advantage of using GELU over ReLU activations, as discussed in Chapter 4. Lines 4 and 5 present the same experiments as lines 1 and 2, respectively, but now with the number of kernels (η), the kernel multiplier (ξ) in the MMD, and its weighting factor empirically tuned for HoLoDANN (reported in line 6). While line 4 shows a slight performance degradation on average compared to line 1—especially in the 4vs1 random scenario—the average performance between lines 2 and 5 remains the same. However, when DANN is introduced (line 6), we observe a performance improvement with these hyperparameter and activation function adjustments.

Finally, by comparing line 0 with line 6, we conclude that with slight adaptations to a stylus-based framework like HoLoSig, it is possible to leverage finger-written signatures—even when only a small number of samples are available— to significantly improve system performance compared to the standard approach in the literature, which typically does

not explicitly address finger-written signatures.

5.5.2 HoLoDANN results on DeepSignDB

Table 5.18 presents the HoLoDANN results for each individual dataset within DeepSignDB. As expected, the EERs are considerably higher than those observed in the stylus-based scenario, mainly due to the increased difficulty of the finger-written signature verification task.

Table 5.18: EER (in %) for finger-written signatures across DeepSignDB experiment protocols.

Dataset	Skilled Forgeries		Random Forgeries		Average
	4vs1	1vs1	4vs1	1vs1	
eBioSign DS1 w4	10.00	14.23	0.08	1.04	3.84
eBioSign DS1 w5	6.55	11.73	1.26	2.86	3.10
eBioSign DS2 w5	3.10	7.29	0.17	1.44	3.00
eBioSign DS2 w6	2.91	5.66	0.17	0.86	2.40
DeepSignDB	5.65	9.99	0.55	1.78	4.50

Table 5.19: Alignment and Separation Error (in %) across DeepSignDB experiment protocols

Dataset	Seperation Error				Alignment Error			
	Skilled		Random		Skilled		Random	
	4vs1	1vs1	4vs1	1vs1	4vs1	1vs1	4vs1	1vs1
eBioSign DS1 w4	5.71	7.65	0.00	0.54	4.29	3.31	0.08	0.50
eBioSign DS1 w5	3.45	9.23	0.00	0.87	3.10	6.58	1.26	1.99
eBioSign DS2 w5	0.83	4.35	0.00	0.15	2.26	2.50	0.17	1.29
eBioSign DS2 w6	0.00	1.13	0.00	0.41	2.91	2.95	0.17	0.45
DeepSignDB	3.53	6.67	0.00	0.85	2.12	4.52	0.55	0.93

Table 5.19 presents the separation and alignment errors for each dataset within DeepSignDB. Notably, the separation error in the 4vs1 random scenario is zero, indicating that HoLoDANN successfully creates a clear separation between genuine signatures and random forgeries in this case. However, a single global threshold is not able to capture all the separation windows across different users.

Unlike the stylus-based scenario, in the finger-written scenario we observe that the separation error is often considerably higher than the alignment error. This highlights that the model is struggling more with distinguishing between genuine and forged signatures, especially in the case of skilled forgeries.

5.5.3 Comparison with the state-of-art

Finally, we compare the HoLoDANN results with the state-of-the-art in Table 5.20. HoLoDANN outperforms previous approaches in detecting skilled forgeries across both the 1vs1 and 4vs1 scenarios, as well as on average. Additionally, it maintains competitive performance in the random forgery scenario. These results validate that holistic and local representations can also be effectively applied to finger-written OSV. Moreover, they demonstrate that the combination of the DANN strategy with targeted adaptations to the model architecture—designed to better handle the characteristics of finger-written signatures, even with limited training data—can lead to significant performance improvements.

Table 5.20: EER (in %) of the proposed method compared with the state-of-the-art on DeepSignDB stylus scenario.

Method	Skilled Forgeries		Random Forgeries		Average
	4vs1	1vs1	4vs1	1vs1	
TA-RNN [33]	11.30(5.65↓)	14.74(4.75↓)	1.00(0.45↓)	1.8(0.02↓)	6.98(2.49↓)
DsDTW [14]	6.99(1.34↓)	11.84(1.85↓)	1.81(1.26↓)	2.89(1.11↓)	5.88(1.39↓)
SynSig2Vec [15]	6.97(1.32↓)	10.87(0.88↓)	0.79(0.24↓)	1.86(0.08↓)	5.12(0.63↓)
CGRN+CNN[22]	6.71(1.06↓)	11.64(1.65↓)	0.25(0.30↑)	1.12(0.66↑)	4.93(0.44↓)
HoLoSig (stylus only)	7.93(2.28↓)	13.72(3.73↓)	0.36(0.19↑)	1.44(0.34↑)	5.86(1.37↓)
HoLoDANN (ours)	***5.65***	***9.99***	***0.55***	***1.78***	***4.49***

5.6 Real-Time Feasibility

To evaluate the practical applicability of HoLoSig and HoLoDANN in real-world scenarios, Figure 5.3 presents the distribution of inference times for the DeepSignDB experimental protocols against skilled forgeries, using either four or one reference signatures. These distributions are derived from 14,768 and 60,414 verification attempts, respectively, executed on an RTX 4090 GPU under a low-demand computational environment. Similarly,

Figure 5.4 shows the corresponding results for the finger scenario, based on 1,392 verification attempts for the 4vs1 setting and 5,600 for the 1vs1 setting. The results indicate that most verifications in the 4vs1 configuration are completed in under 0.02 seconds, while the 1vs1 scenario typically requires less than 0.01 seconds.

It is also worth noting that, although the number of trainable parameters varies according the number of writers—owing to the dependency of the final linear layer (used in the Poly-1 Cross Entropy Loss) on the number of classes in the training set—the total number of parameters required during inference remains fixed at 303,424 for both HoLoSig and HoLoDANN. This is because the final linear classification layer and the domain classifier (present only in HoLoDANN) are excluded from the inference stage.

These results support the viability of deploying HoLoSig and HoLoDANN in real-time, operational environments.

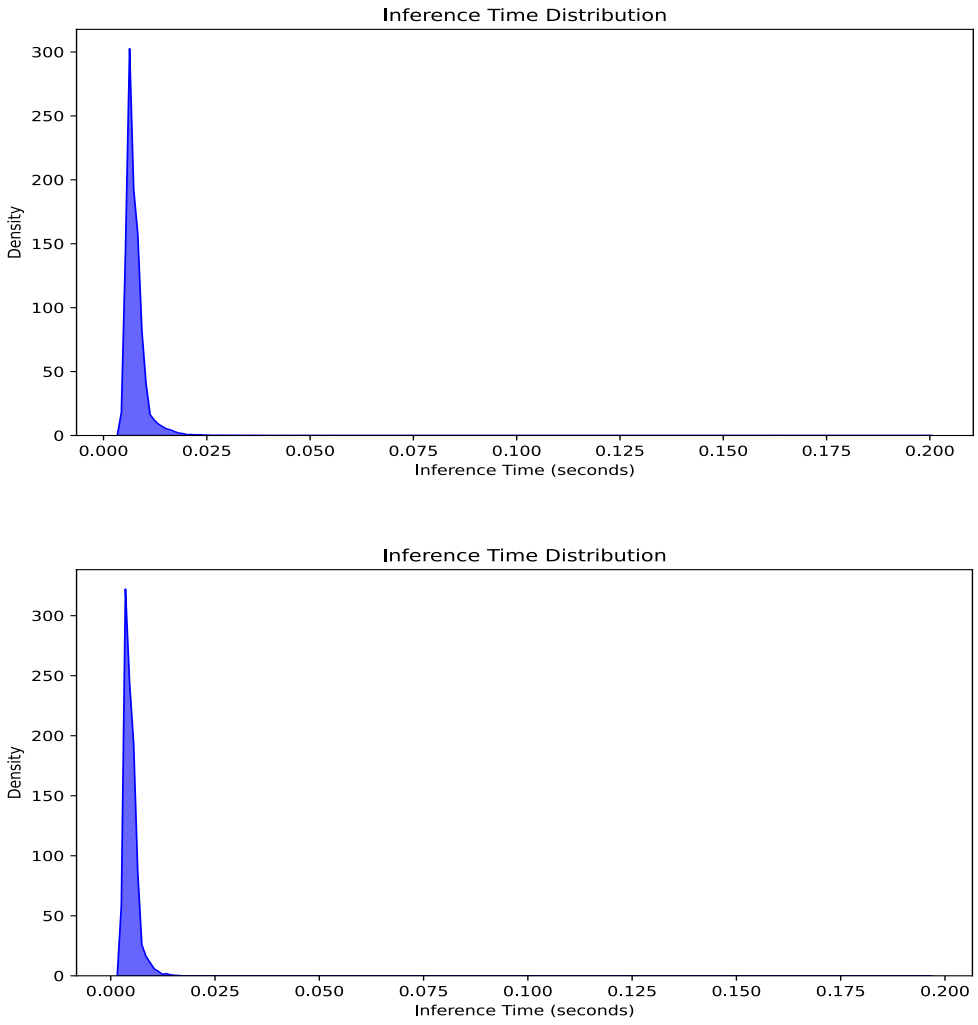


Figure 5.3: Distribution of inference times for HoLoSig (stylus scenario) across the DeepSignDB experimental protocols against skilled forgeries, using either four (4vs1) or one (1vs1) reference signatures.

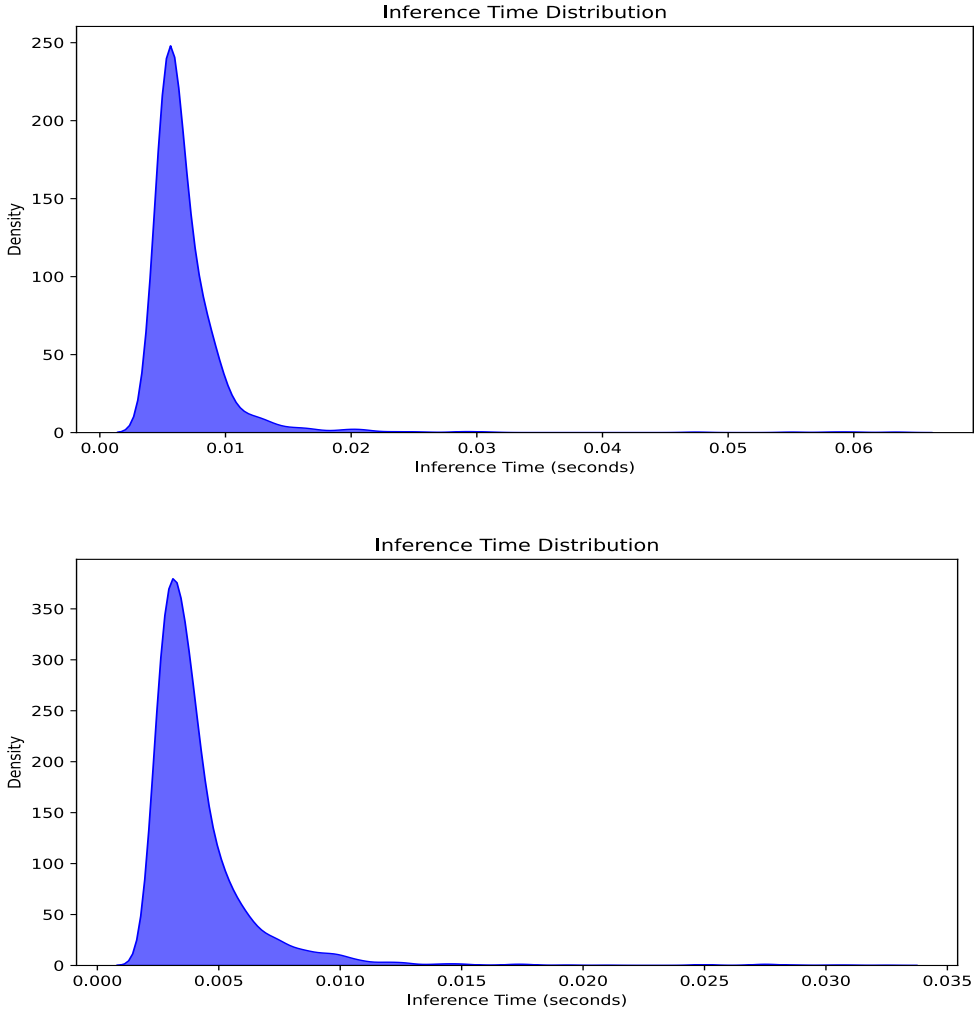


Figure 5.4: Distribution of inference times for HoLoDANN (finger scenario) across the DeepSignDB experimental protocols against skilled forgeries, using either four (4vs1) or one (1vs1) reference signatures.

5.7 Error Analysis According Signature Duration

To better understand which signatures HoLoSig struggles to verify, Figure 5.5 presents an analysis relating the signatures that were misclassified using a global threshold to the length of the reference signature across the four DeepSignDB experimental protocols in the stylus scenario. Figure 5.6 shows the same for HoLoDANN in the finger scenario. For the 4vs1 scenario, we consider the average duration of the template signatures. Figure 5.5 shows that most verification errors occur with signatures of shorter duration, which is expected since longer signatures tend to be more complex and therefore harder to imitate. This observation suggests that verification strategies which apply stricter acceptance for users with shorter reference signature durations may lead to improved overall performance.

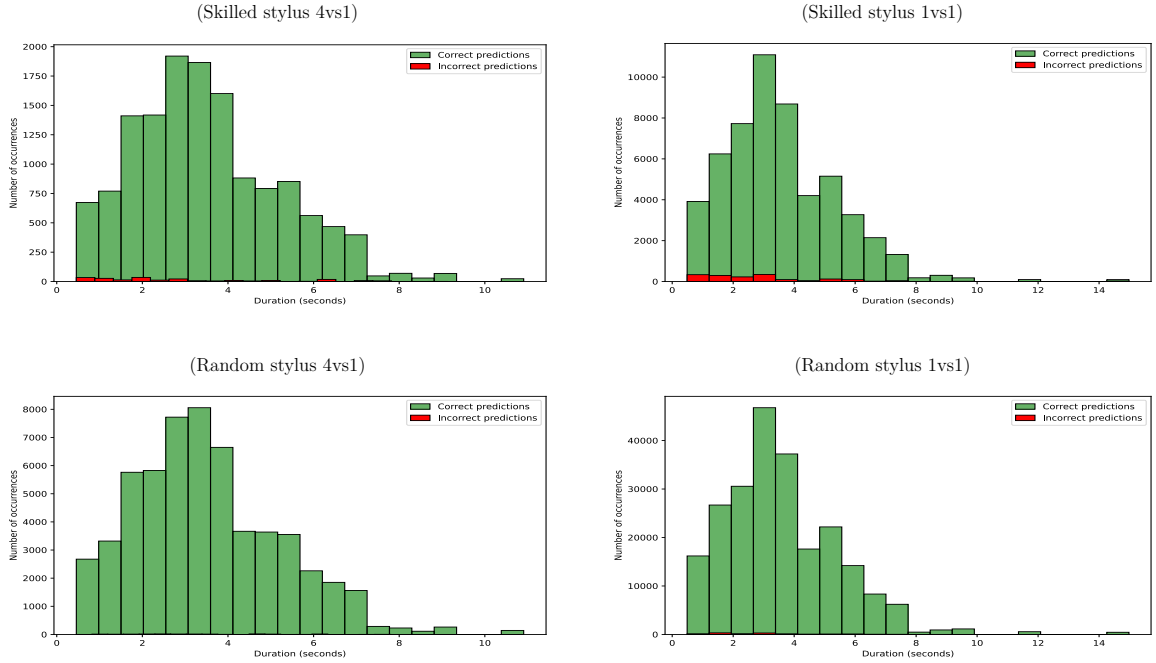


Figure 5.5: Error analysis based on signature duration in the stylus scenario.

However, Figure 5.6 shows that, in the 1vs1 skilled forgery scenario for finger-written signatures, even signatures with longer durations exhibit a considerable error rate. This result highlights that, although HoLoDANN outperforms the state-of-the-art in the skilled forgery scenario, there is still a significant gap before the performance with finger-written signatures can reach levels comparable to those achieved with stylus-written signatures.

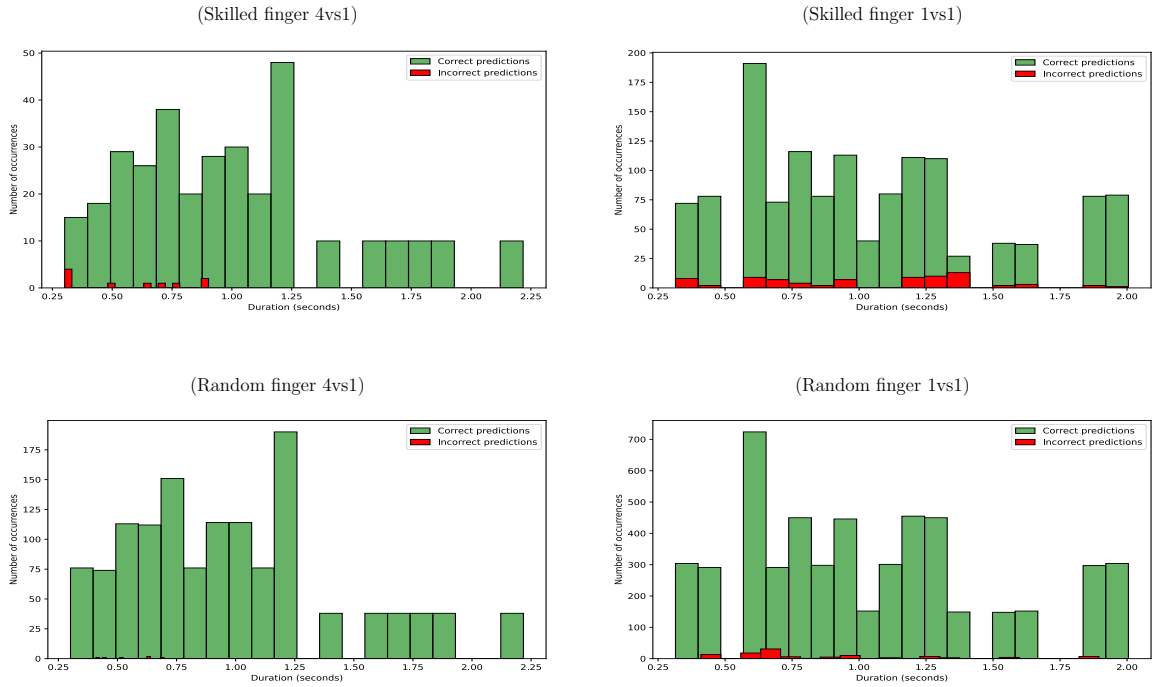


Figure 5.6: Error analysis based on signature duration in the finger scenario.

Chapter 6

Conclusion and Future Works

We introduced HoLoSig, a novel WI OSV framework that integrates both holistic and local representations during the training and verification stages. As a WI framework, HoLoSig addresses the challenge of minimizing the global EER using a single threshold, enabling it to effectively distinguish between genuine and forged signatures, even from writers who did not provide samples during the training stage. A key challenge in OSV is the high intra-variability among signatures from the same writer (genuine signatures) and the low inter-variability among skilled forgeries. This often results in genuine signatures from inconsistent writers exhibiting higher dissimilarity than skilled forgeries created from consistent writer signatures.

While the traditional approach in the literature focuses on increasing the separation between signatures from different writers — an effort that becomes increasingly difficult — HoLoSig creates a separation while shifting the dissimilarity scores to a common region through the novel Triplet MMD introduced in this study. This approach enhances the intersection of separation windows across different writers, ultimately improving the EER with a global threshold. As a result, HoLoSig achieves the new state-of-the-art results on DeepSignDB. Notably, the results demonstrate that the alignment EER is just as important as the separation EER, highlighting that the overall performance can be significantly enhanced by improving writer-specific threshold alignment.

Furthermore, we extended the proposed loss function by incorporating alternative divergence metrics in place of MMD, achieving comparable results while also reducing alignment error. This finding supports the view that the misalignment of writer-specific optimal thresholds can be addressed by ensuring that the dissimilarity scores computed from the extracted representations follow a consistent distribution.

Finally, we extended the HoLoSig framework to more effectively address the finger-written OSV problem by integrating a DANN mechanism, with minor architectural modifications. The resulting HoLoDANN framework showed a significant performance im-

provement over the original HoLoSig trained solely on stylus-based signatures—a common practice in recent state-of-the-art methods that typically do not account for finger-written data. This enhancement enabled us to achieve new state-of-the-art results for finger-written OSV on the DeepSignDB dataset.

6.1 Future Works

The conclusions drawn by this study reveal that it is worthwhile to explore alternative methods for moving the dissimilarity scores to a common region, thus reducing the alignment EER. While we addressed the problem by ensuring the model dissimilarities scores to respect a consistent distribution, other approaches may be also be able to induce the alignment in writer-specific optimal thresholds and may be even used together with our proposed approach.

Our LODO results revealed that HoLoSig outperforms the state-of-the-art in stylus-written OSV, even when trained with only half the amount of data used in the compared studies. Moreover, the results suggest that performance in specific evaluation scenarios can be more influenced by the choice of training datasets than by the total volume of training data. These findings are encouraging. The strong performance achieved under a reduced-data training setup, along with the observed sensitivity to the presence or absence of specific datasets during training, indicates that a thoughtful selection of training data could enhance system performance. Consequently, triplet-mining strategies tailored specifically to the OSV problem appear to be a promising direction for further improving HoLoSig's performance.

The HoLoDANN architecture may also benefit from these techniques. However, in this scenario, separation error appears to be more critical than alignment error, particularly due to the high variability observed in finger-written signatures. Although the DANN module improved overall performance—enabling the effective use of stylus-written signatures from DeepSignDB to address the finger-written OSV task—further research in transfer learning and domain adaptation is needed to fully leverage the potential of stylus-written data.

Finally, the decomposition of the EER into separation and alignment errors can also be beneficial for better understanding and addressing other biometric systems that operate with global thresholds. Specifically, systems that use gait or voice as biometric traits often deal with time series data and exploit local properties through DTW. Therefore, it is worth investigating whether HoLoSig can be adapted to tackle these types of problems.

Bibliography

[1] Anil K. Jain, Arun A. Ross: *Handbook of Biometrics*, volume 01. Springer Science+Business Media, LLC, 233 Spring Street, New York, NY 10013, USA, 2008. 1

[2] Minaee, Shervin, Amirali Abdolrashidi, Hang Su, Mohammed Bennamoun, and David Zhang: *Biometrics recognition using deep learning: a survey.* Artificial Intelligence Review, 56(8):8647–8695, August 2023, ISSN 1573-7462. https://doi.org/10.1007/s10462-022-10237-x, visited on 2025-01-28. 1, 4

[3] Adil, Muhammad, Ahmed Farouk, Aitizaz Ali, Houbing Song, and Zhanpeng Jin: *Securing tomorrow of next-generation technologies with biometrics, state-of-the-art techniques, open challenges, and future research directions.* Computer Science Review, 57:100750, 2025, ISSN 1574-0137. https://www.sciencedirect.com/science/article/pii/S1574013725000267. 1

[4] Maltoni, Davide, Dario Maio, Anil K. Jain, and Salil Prabhakar: *Handbook of Fingerprint Recognition.* Springer Publishing Company, Incorporated, 2nd edition, 2009, ISBN 1848822537. 1

[5] WAYMAN, JAMES L.: *Fundamentals of biometric authentication technologies.* International Journal of Image and Graphics, 01(01):93–113, 2001. https://doi.org/10.1142/S0219467801000086. 2

[6] Plamondon, R. and S.N. Srihari: *Online and off-line handwriting recognition: a comprehensive survey.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(1):63–84, January 2000, ISSN 1939-3539. https://ieeexplore.ieee.org/document/824821, visited on 2024-12-16, Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence. 3, 4

[7] Faundez-Zanuy, Marcos, Julian Fierrez, Miguel A. Ferrer, Moises Diaz, Ruben Tolosana, and Réjean Plamondon: *Handwriting Biometrics: Applications and Future*

Trends in e-Security and e-Health. Cognitive Computation, 12(5):940–953, September 2020, ISSN 1866-9964. https://doi.org/10.1007/s12559-020-09755-z, visited on 2024-12-16. 3

[8] Diaz, Moises, Miguel A. Ferrer, Donato Impedovo, Muhammad Imran Malik, Giuseppe Pirlo, and Réjean Plamondon: *A Perspective Analysis of Handwritten Signature Technology*. ACM Comput. Surv., 51(6):117:1–117:39, January 2019, ISSN 0360-0300. https://dl.acm.org/doi/10.1145/3274658, visited on 2024-12-16. 3, 4

[9] Impedovo, Donato and Giuseppe Pirlo: *Automatic Signature Verification: The State of the Art*. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 38(5):609–635, September 2008, ISSN 1558-2442. https://ieeexplore.ieee.org/document/4603099, visited on 2024-12-16, Conference Name: IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews). 4

[10] Impedovo, D., G. Pirlo, and R. Plamondon: *Handwritten Signature Verification: New Advancements and Open Issues*. In *2012 International Conference on Frontiers in Handwriting Recognition*, pages 367–372, September 2012. https://ieeexplore.ieee.org/document/6424421, visited on 2024-12-17. 4, 5

[11] Srihari, S.N., Aihua Xu, and M.K. Kalera: *Learning strategies and classification methods for off-line signature verification*. In *Ninth International Workshop on Frontiers in Handwriting Recognition*, pages 161–166, October 2004. https://ieeexplore.ieee.org/document/1363904, visited on 2025-01-28, ISSN: 1550-5235. 5

[12] Guru, D.S. and H.N. Prakash: *Online Signature Verification and Recognition: An Approach Based on Symbolic Representation*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(6):1059–1073, June 2009, ISSN 1939-3539. https://ieeexplore.ieee.org/document/4731270, visited on 2024-12-19, Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence. 5, 7, 10, 11, 14

[13] Guru, D. S., K. S. Manjunatha, S. Manjunath, and M. T. Somashekara: *Interval valued symbolic representation of writer dependent features for online signature verification*. Expert Systems with Applications, 80:232–243, September 2017, ISSN 0957-4174. https://www.sciencedirect.com/science/article/pii/S0957417417301756, visited on 2025-04-15. 5, 7, 10, 11, 14

[14] Jiang, Jiajia, Songxuan Lai, Lianwen Jin, and Yecheng Zhu: *DsDTW: Local Representation Learning With Deep soft-DTW for Dynamic Signature Verification.* IEEE Transactions on Information Forensics and Security, 17:2198–2212, 2022, ISSN 1556-6021. `https://ieeexplore.ieee.org/document/9787558`, visited on 2024-12-10, Conference Name: IEEE Transactions on Information Forensics and Security. 6, 7, 11, 12, 13, 14, 16, 17, 19, 21, 22, 23, 27, 32, 34, 35, 47, 48, 51

[15] Lai, Songxuan, Lianwen Jin, Yecheng Zhu, Zhe Li, and Luojun Lin: *SynSig2Vec: Forgery-Free Learning of Dynamic Signature Representations by Sigma Lognormal-Based Synthesis and 1D CNN.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(10):6472–6485, October 2022, ISSN 1939-3539. `https://ieeexplore.ieee.org/document/9448392#full-text-header`, visited on 2024-12-10, Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence. 6, 7, 11, 12, 13, 14, 16, 18, 21, 32, 34, 35, 47, 48, 51

[16] Lai, Songxuan and Lianwen Jin: *Recurrent Adaptation Networks for Online Signature Verification.* IEEE Transactions on Information Forensics and Security, 14(6):1624–1637, June 2019, ISSN 1556-6021. `https://ieeexplore.ieee.org/document/8543636`, visited on 2024-12-12, Conference Name: IEEE Transactions on Information Forensics and Security. 6, 7, 11, 14, 16, 17, 21

[17] Wu, Xiaomeng, Akisato Kimura, Seiichi Uchida, and Kunio Kashino: *Prewarping Siamese Network: Learning Local Representations for Online Signature Verification.* In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2467–2471, May 2019. `https://ieeexplore.ieee.org/document/8683036`, visited on 2024-12-12, ISSN: 2379-190X. 6, 7, 13, 16

[18] Wu, Xiaomeng, Akisato Kimura, Brian Kenji Iwana, Seiichi Uchida, and Kunio Kashino: *Deep Dynamic Time Warping: End-to-End Local Representation Learning for Online Signature Verification.* In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1103–1110, September 2019. `https://ieeexplore.ieee.org/document/8978093`, visited on 2024-12-12, ISSN: 2379-2140. 6, 7, 11, 13, 16, 19

[19] Vorugunti, Chandra Sekhar, Guru Devanur S, Prerana Mukherjee, and Viswanath Pulabaigari: *OSVNet: Convolutional Siamese Network for Writer Independent Online Signature Verification.* In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1470–1475, September 2019. `https://ieeexplore.ieee.org/document/8978082`, visited on 2024-12-16, ISSN: 2379-2140. 6, 7, 10, 12, 16

[20] Yang, Peilun, Hanchen Wang, Jianye Yang, Zhengping Qian, Ying Zhang, and Xuemin Lin: *Deep Learning Approaches for Similarity Computation: A Survey.* IEEE Transactions on Knowledge and Data Engineering, 36(12):7893–7912, December 2024, ISSN 1558-2191. `https://ieeexplore.ieee.org/document/10584318`, visited on 2025-02-22, Conference Name: IEEE Transactions on Knowledge and Data Engineering. 6

[21] Park, Chan Yong, Han Gyu Kim, and Ho Jin Choi: *Robust Online Signature Verification Using Long-term Recurrent Convolutional Network.* In *2019 IEEE International Conference on Consumer Electronics (ICCE)*, pages 1–6, January 2019. `https://ieeexplore.ieee.org/document/8662005`, visited on 2024-12-16, ISSN: 2158-4001. 7, 11, 12, 14

[22] Yu, Hewei and Pengfei Shi: *A Novel Deep Ensemble Framework for Online Signature Verification Using Temporal and Spatial Representation.* In Wang, Ding, Moti Yung, Zheli Liu, and Xiaofeng Chen (editors): *Information and Communications Security*, pages 534–549, Singapore, 2023. Springer Nature, ISBN 978-981-99-7356-9. 7, 11, 12, 13, 14, 19, 21, 27, 32, 47, 48, 51

[23] O'Reilly, Christian and Réjean Plamondon: *Development of a Sigma–Lognormal representation for on-line signatures.* Pattern Recognition, 42(12):3324–3337, December 2009, ISSN 0031-3203. `https://www.sciencedirect.com/science/article/pii/S0031320308004470`, visited on 2025-02-22. 7, 11, 14

[24] Sakoe, H. and S. Chiba: *Dynamic programming algorithm optimization for spoken word recognition.* IEEE Transactions on Acoustics, Speech, and Signal Processing, 26(1):43–49, February 1978, ISSN 0096-3518. `https://ieeexplore.ieee.org/document/1163055`, visited on 2024-12-19, Conference Name: IEEE Transactions on Acoustics, Speech, and Signal Processing. 7

[25] Cuturi, Marco and Mathieu Blondel: *Soft-DTW: a differentiable loss function for time-series.* In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 894–903, Sydney, NSW, Australia, August 2017. JMLR.org. 7, 8, 19, 23, 24

[26] Fierrez-Aguilar, Julian, Loris Nanni, Jaime Lopez-Peñalba, Javier Ortega-Garcia, and Davide Maltoni: *An On-Line Signature Verification System Based on Fusion of Local and Global Information.* In Kanade, Takeo, Anil Jain, and Nalini K. Ratha (editors): *Audio- and Video-Based Biometric Person Authentication*, pages 523–532, Berlin, Heidelberg, 2005. Springer, ISBN 978-3-540-31638-1. 7

[27] Ahrabian, Kian and Bagher BabaAli: *Usage of autoencoders and Siamese networks for online handwritten signature verification*. Neural Computing and Applications, 31(12):9321–9334, December 2019, ISSN 1433-3058. https://doi.org/10.1007/s00521-018-3844-z, visited on 2024-12-16. 7, 11, 12, 13

[28] Hoffer, Elad and Nir Ailon: *Deep Metric Learning Using Triplet Network*. In Feragen, Aasa, Marcello Pelillo, and Marco Loog (editors): *Similarity-Based Pattern Recognition*, pages 84–92, Cham, 2015. Springer International Publishing, ISBN 978-3-319-24261-3. 8, 23

[29] Gretton, Arthur, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola: *A kernel two-sample test*. J. Mach. Learn. Res., 13(null):723–773, March 2012, ISSN 1532-4435. 8, 24

[30] Ortega-Garcia, J., J. Fierrez-Aguilar, D. Simon, J. Gonzalez, M. Faundez-Zanuy, V. Espinosa, A. Satue, I. Hernaez, J. J. Igarza, C. Vivaracho, D. Escudero, and Q. I. Moro: *MCYT baseline corpus: a bimodal biometric database*. IEE Proceedings - Vision, Image and Signal Processing, 150(6):395–401, December 2003. https://digital-library.theiet.org/doi/10.1049/ip-vis%3A20031078, visited on 2025-01-21, Publisher: The Institution of Engineering and Technology. 10, 31, 42

[31] Yeung, Dit Yan, Hong Chang, Yimin Xiong, Susan George, Ramanujan Kashi, Takashi Matsumoto, and Gerhard Rigoll: *SVC2004: First International Signature Verification Competition*. In Zhang, David and Anil K. Jain (editors): *Biometric Authentication*, pages 16–22, Berlin, Heidelberg, 2004. Springer, ISBN 978-3-540-25948-0. 10

[32] Fierrez, J., J. Galbally, J. Ortega-Garcia, M. R. Freire, F. Alonso-Fernandez, D. Ramos, D. T. Toledano, J. Gonzalez-Rodriguez, J. A. Siguenza, J. Garrido-Salas, E. Anguiano, G. Gonzalez-De-Rivera, R. Ribalda, M. Faundez-Zanuy, J. A. Ortega, V. Cardeñoso-Payo, A. Viloria, C. E. Vivaracho, Q. I. Moro, J. J. Igarza, J. Sanchez, I. Hernaez, C. Orrite-Uruñuela, F. Martinez-Contreras, and J. J. Gracia-Roche: *BiosecurID: a multimodal biometric database*. Pattern Anal. Appl., 13(2):235–246, May 2010, ISSN 1433-7541. https://doi.org/10.1007/s10044-009-0151-4, visited on 2025-01-21. 10, 31, 42

[33] Tolosana, Ruben, Ruben Vera-Rodriguez, Julian Fierrez, and Javier Ortega-Garcia: *DeepSign: Deep On-Line Signature Verification*. IEEE Transactions on Biometrics, Behavior, and Identity Science, 3(2):229–239, April 2021, ISSN 2637-6407.

https://ieeexplore.ieee.org/document/9335993, visited on 2024-12-10, Conference Name: IEEE Transactions on Biometrics, Behavior, and Identity Science. 10, 13, 14, 15, 19, 21, 31, 42, 47, 48, 51

[34] Vorugunti, Chandra Sekhar, Viswanath Pulabaigari, Prerana Mukherjee, and Avinash Gautam: *COMPOSV: compound feature extraction and depthwise separable convolution-based online signature verification.* Neural Computing and Applications, 34(13):10901–10928, July 2022, ISSN 1433-3058. https://doi.org/10.1007/s00521-022-07018-6, visited on 2025-04-19. 10

[35] Martinez-Diaz, Marcos, Julian Fiérrez, Ram P. Krish, and Javier Galbally: *Mobile signature verification: feature robustness and performance comparison.* IET Biom., 3(4):267–277, 2014. https://doi.org/10.1049/iet-bmt.2013.0081, visited on 2025-04-19. 10, 11

[36] Okawa, Manabu: *Template Matching Using Time-Series Averaging and DTW With Dependent Warping for Online Signature Verification.* IEEE Access, 7:81010–81019, 2019, ISSN 2169-3536. https://ieeexplore.ieee.org/document/8736875, visited on 2025-04-18. 11, 13, 19

[37] Okawa, Manabu: *Time-series averaging and local stability-weighted dynamic time warping for online signature verification.* Pattern Recognition, 112:107699, April 2021, ISSN 0031-3203. https://www.sciencedirect.com/science/article/pii/S0031320320305021, visited on 2025-04-17. 11, 13, 19

[38] Shi, Zhaosen, Fagen Li, Dong Hao, and Qinshuo Sun: *Handwritten Signature Verification via Multimodal Consistency Learning.* IEEE Transactions on Information Forensics and Security, pages 1–1, 2025, ISSN 1556-6021. https://ieeexplore.ieee.org/document/10950350, visited on 2025-04-19. 11, 14

[39] Li, Qixiang, Zhaoya Wang, Lianwen Jin, Nurbiya Yadikar, and Kurban Ubul: *MMHSV: A Multimodal Handwritten Signature Verification Fusing Dynamic and Static Feature.* In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4730–4734, April 2024. https://ieeexplore.ieee.org/document/10447071/references, visited on 2025-04-19, ISSN: 2379-190X. 11, 14

[40] Alaei, Alireza, Srikanta Pal, Umapada Pal, and Michael Blumenstein: *An Efficient Signature Verification Method Based on an Interval Symbolic Representation and a Fuzzy Similarity Measure.* IEEE Transactions on Information Forensics and Security,

12(10):2360–2372, October 2017, ISSN 1556-6021. https://ieeexplore.ieee.org/document/7932929, visited on 2025-04-19. 11, 14

[41] Sarvabhatla, Mrudula, K. Upendra Raju, Srinivasu Nulaka, and Atul Negi: *A Writer Adaptation Approach to Online Signature Verification Through Feature Clustering and Classifier Selection*. In *2024 IEEE International Conference on Computer Vision and Machine Intelligence (CVMI)*, pages 1–6, October 2024. https://ieeexplore.ieee.org/document/10781673?denied=, visited on 2025-03-29. 11

[42] Plamondon, Réjean: *A kinematic theory of rapid human movements: Part I. Movement representation and generation*. Biological Cybernetics, 72(4):295–307, March 1995, ISSN 1432-0770. https://doi.org/10.1007/BF00202785, visited on 2025-04-19. 11

[43] Plamondon, Réjean: *A kinematic theory of rapid human movements: Part II. Movement time and control*. Biological Cybernetics, 72(4):309–320, March 1995, ISSN 1432-0770. https://doi.org/10.1007/BF00202786, visited on 2025-04-19. 11

[44] Plamondon, Réjean: *A kinematic theory of rapid human movements: Part III. Kinetic outcomes*. Biological Cybernetics, 78(2):133–145, February 1998, ISSN 1432-0770. https://doi.org/10.1007/s004220050420, visited on 2025-04-19. 11

[45] Plamondon, Réjean, Chunhua Feng, and Anna Woch: *A kinematic theory of rapid human movement. Part IV: a formal mathematical proof and new insights*. Biological Cybernetics, 89(2):126–138, August 2003, ISSN 1432-0770. https://doi.org/10.1007/s00422-003-0407-9, visited on 2025-04-19. 11

[46] Gomez-Barrero, Marta, Javier Galbally, Julian Fierrez, Javier Ortega-Garcia, and Réjean Plamondon: *Enhanced on-line signature verification based on skilled forgery detection using Sigma-LogNormal Features*. In *2015 International Conference on Biometrics (ICB)*, pages 501–506, May 2015. https://ieeexplore.ieee.org/document/7139065, visited on 2025-04-18, ISSN: 2376-4201. 11

[47] Fischer, Andreas and Réjean Plamondon: *Signature Verification Based on the Kinematic Theory of Rapid Human Movements*. IEEE Transactions on Human-Machine Systems, 47(2):169–180, April 2017, ISSN 2168-2305. https://ieeexplore.ieee.org/document/7797210, visited on 2025-04-18. 12

[48] Diaz, Moises, Andreas Fischer, Réjean Plamondon, and Miguel A. Ferrer: *Towards an automatic on-line signature verifier using only one reference per signer*. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages

631–635, August 2015. https://ieeexplore.ieee.org/document/7333838, visited on 2025-04-18. 12, 14

[49] Galbally, Javier, Julian Fierrez, Marcos Martinez-Diaz, and Javier Ortega-Garcia: *Evaluation of Brute-force Attack to Dynamic Signature Verification Using Synthetic Samples*. In *2009 10th International Conference on Document Analysis and Recognition*, pages 131–135, July 2009. https://ieeexplore.ieee.org/document/5277761, visited on 2025-04-21, ISSN: 2379-2140. 14

[50] Ferrer, Miguel A., Moises Diaz, Cristina Carmona-Duarte, and Réjean Plamondon: *A Biometric Attack Case Based on Signature Synthesis*. In *2018 International Carnahan Conference on Security Technology (ICCST)*, pages 1–6, October 2018. https://ieeexplore.ieee.org/document/8585714, visited on 2025-04-18, ISSN: 2153-0742. 14

[51] Antal, Margit, László Zsolt Szabó, and Tünde Tordai: *Online signature verification on mobisig finger-drawn signature corpus*. Mobile Information Systems, 2018(1):3127042, 2018. https://onlinelibrary.wiley.com/doi/abs/10.1155/2018/3127042. 15

[52] Tolosana, Ruben, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia: *Benchmarking desktop and mobile handwriting across cots devices: The e-biosign biometric database*. PLOS ONE, 12(5):1–17, May 2017. https://doi.org/10.1371/journal.pone.0176792. 15, 31, 42

[53] Ren, Yanzhi, Chen Wang, Yingying Chen, Mooi Choo Chuah, and Jie Yang: *Signature Verification Using Critical Segments for Securing Mobile Transactions*. IEEE Transactions on Mobile Computing, 19(3):724–739, March 2020, ISSN 1558-0660. https://ieeexplore.ieee.org/document/8634928, visited on 2025-05-08. 15

[54] Nam, Seungsoo, Hosung Park, Changho Seo, and Daeseon Choi: *Forged Signature Distinction Using Convolutional Neural Network for Feature Extraction*. Applied Sciences, 8(2):153, February 2018, ISSN 2076-3417. https://www.mdpi.com/2076-3417/8/2/153, visited on 2025-05-08, Number: 2 Publisher: Multidisciplinary Digital Publishing Institute. 15

[55] Wang, Wei, Vincent W. Zheng, Han Yu, and Chunyan Miao: *A Survey of Zero-Shot Learning: Settings, Methods, and Applications*. ACM Trans. Intell. Syst. Technol., 10(2):13:1–13:37, January 2019, ISSN 2157-6904. https://doi.org/10.1145/3293318, visited on 2025-04-22. 16

[56] Cho, KyungHyun, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio: *On the properties of neural machine translation: Encoder-decoder approaches*. CoRR, abs/1409.1259, 2014. http://arxiv.org/abs/1409.1259. 17

[57] Leng, Zhaoqi, Mingxing Tan, Chenxi Liu, Ekin Dogus Cubuk, Jay Shi, Shuyang Cheng, and Dragomir Anguelov: *Polyloss: A polynomial expansion perspective of classification loss functions*. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. https://openreview.net/forum?id=gSdSJoenupI. 19, 22

[58] Sakoe, Hiroaki and Seibi Chiba: *A Dynamic Programming Approach to Continuous Speech Recognition*. In *Proceedings of the Seventh International Congress on Acoustics, Budapest*, volume 3, pages 65–69, Budapest, 1971. Akadémiai Kiadó. 19, 23

[59] Chopra, Sumit, Raia Hadsell, and Yann LeCun: *Learning a similarity metric discriminatively, with application to face verification*. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA*, pages 539–546. IEEE Computer Society, 2005. https://doi.org/10.1109/CVPR.2005.202. 23

[60] Steinwart, Ingo: *On the influence of the kernel on the consistency of support vector machines*. J. Mach. Learn. Res., 2:67–93, March 2002, ISSN 1532-4435. https://dl.acm.org/doi/10.1162/153244302760185252, visited on 2025-01-27. 25

[61] Bellemare, Marc G., Ivo Danihelka, Will Dabney, Shakir Mohamed, Balaji Lakshminarayanan, Stephan Hoyer, and Remi Munos: *The Cramer Distance as a Solution to Biased Wasserstein Gradients*. February 2018. https://openreview.net/forum?id=S1m6h21Cb, visited on 2025-06-19. 25, 26

[62] Ganin, Yaroslav, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky: *Domain-adversarial training of neural networks*. J. Mach. Learn. Res., 17(1):2096–2030, January 2016, ISSN 1532-4435. 28, 29

[63] Ortega-Garcia, Javier, Julian Fierrez, Fernando Alonso-Fernandez, J. Galbally, M. R. Freire, J. Gonzalez-Rodriguez, C. Garcia-Mateo, J. L. Alba-Castro, E. Gonzalez-Agulla, E. Otero-Muras, S. Garcia-Salicetti, L. Allano, B. Ly-Van, B. Dorizzi, J. Kittler, T. Bourlai, N. Poh, F. Deravi, M. W. R. Ng, M. Fairhurst, J. Hennebert, A. Humm, M. Tistarelli, L. Brodo, J. Richiardi, A. Drygajlo, H. Ganster, F. M. Sukno, S. K. Pavani, A. Frangi, L. Akarun, and A. Savran: *The Multi-Scenario Multi-Environment BioSecure Multimodal Database (BMDB)*. IEEE Trans. on Pattern

Analysis and Machine Intelligence, 32(6):1097–1111, 2009. https://urn.kb.se/resolve?urn=urn:nbn:se:hh:diva-21221, visited on 2025-01-21. 31, 42