UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

Talles Brito Viana

Robust Handwritten Signature Representation with Multi-task Continual Learning of Synthetic Data over Predefined Real Feature Space and Contrastive Fine-Tuning

Recife

2025

Talles Brito Viana

Robust Handwritten Signature Representation with Multi-task Continual Learning of Synthetic Data over Predefined Real Feature Space and Contrastive Fine-Tuning

Tese apresentada ao Programa de Pós- Graduação em Ciências da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Ciências da Computação.

Área de Concentração: Inteligência Computacional.

Orientador: Adriano Lorena Inácio de Oliveira.

Coorientadores: Rafael Menelau Oliveira e Cruz (École de Technologie Supérieure - Université du Québec, Montreal), Robert Sabourin (École de Technologie Supérieure - Université du Québec, Montreal).

Recife

2025

.Catalogação de Publicação na Fonte. UFPE - Biblioteca Central

Viana, Talles Brito.
Robust handwritten signature representation with multi-task continual learning of synthetic data over predefined real feature space and contrastive fine-tuning / Talles Brito Viana. - Recife, 2025.
148f.: il.

Tese (Doutorado) - Universidade Federal de Pernambuco, Centro de Informática, Programa de Pós-Graduação em Ciências da Computação, 2025.
Orientação: Adriano Lorena Inácio de Oliveira.
Coorientação: Rafael Menelau Oliveira e Cruz.
Coorientação: Robert Sabourin.

1. Verificação de assinaturas; 2. Redes neurais convolucionais; 3. Aprendizagem profunda; 4. Aprendizagem contrastiva; 5. Aprendizagem contínua; 6. Destilação de conhecimento. I. Oliveira, Adriano Lorena Inácio de. II. Cruz, Rafael Menelau Oliveira e. III. Sabourin, Robert. IV. Título.

UFPE-Biblioteca Central

Talles Brito Viana

“Robust Handwritten Signature Representation with Multi-task Continual Learning of Synthetic Data over Predefined Real Feature Space and Contrastive Fine-Tuning”

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Aprovada em: 16/04/2025.

_____

Orientador: Prof. Dr. Adriano Lorena Inácio de Oliveira

BANCA EXAMINADORA

_____

Prof. Dr. Francisco de Assis Tenório de Carvalho - Centro de Informática/UFPE

_____

Prof. Dr. George Darmiton da Cunha Cavalcanti - Centro de Informática /UFPE

_____

Prof. Dr. Tsang Ing Ren
Centro de Informática / UFPE

_____

Prof. Dr. George Gomes Cabral
Departamento de Estatística e Informática / UFRPE

_____

Prof. Dr. Moisés Díaz Cabrera
Universidad de Las Palmas de Gran Canaria, Espanha

ACKNOWLEDGEMENTS

Agradeço a Deus por me trazer oportunidade e força necessárias para o desenvolvimento deste trabalho.

Agradeço a minha família, meu pai José Viana, minha mãe Socorro Abreu e às minhas irmãs Thais Brito e Viviane Brito pelo suporte emocional e logístico durante o desenvolvimento deste trabalho. Este trabalho foi em sua maior parte desenvolvido durante a época mais grave da pandemia de COVID-19, de forma que o suporte familiar foi mais do que essencial durante este período.

Agradeço aos professores Adriano Lorena, Rafael Cruz e Robert Sabourin pelo apoio e orientação na condução deste trabalho.

Agradeço a Victor Souza pela colaboração e sugestões durante o desenvolvimento deste trabalho.

Agradeço aos professores do Centro de Informática pelo ensinamento de conceitos básicos necessários para a construção deste trabalho.

Agradeço aos professores que fazem parte da banca examinadora pelas contribuições ao presente trabalho.

RESUMO

Apesar dos recentes avanços em visão computacional, o clássico problema de verificação de assinaturas manuscritas offline ainda permanece desafiador. A tarefa de verificação de assinaturas possui uma alta variabilidade intraclasse pois um usuário frequentemente apresenta alta variabilidade entre suas amostras. Além disso, a verificação de assinaturas é mais difícil na presença de falsificações habilidosas. Recentemente, com o objetivo de enfrentar esses desafios, pesquisadores têm investigado métodos de aprendizagem profunda para obter representações de características de assinaturas manuscritas. Ao mapear assinaturas para um espaço de características, deseja-se obter grupos densos de representações de assinaturas, a fim de lidar com a variabilidade intraclasse. Além disso, não apenas grupos densos são necessários, mas também uma melhor separação entre grupos de diferentes usuários no espaço de características. Por fim, também deseja-se afastar as representações de características de falsificações habilidosas em relação ao respectivo grupo denso de representações genuínas. Nesta tese, concentramos nossos esforços em alcançar essas propriedades no aprendizado de representações de assinaturas manuscritas, visando melhorias subsequentes no tempo de verificação. A pesquisa relatada nesta tese é desenvolvida em dois momentos. Inicialmente, propomos um framework baseado no aprendizado contrastivo de representações de assinaturas manuscritas. Com base na observação dos resultados experimentais da primeira parte desta pesquisa, estendemos este framework para lidar com dados de assinaturas sintéticas e reais durante o treinamento por meio da destilação de conhecimento, proporcionando maior robustez das representações aprendidas na verificação de diversos conjuntos de dados. Neste trabalho, levanta-se a hipótese de que as propriedades desejadas podem ser alcançadas por meio de uma estrutura multitarefa para aprender representações de características de assinaturas manuscritas com base em aprendizagem contrastiva profunda. A estrutura proposta é composta por duas tarefas de objetivos específicos. A primeira tarefa visa mapear exemplos de assinaturas do mesmo usuário mais próximo dentro do espaço de representação, enquanto separa as representações de características de assinaturas de diferentes usuários. A segunda tarefa visa ajustar as representações de falsificações habilidosas adotando funções de perda contrastivas com a capacidade de realizar mineração de exemplos negativos difíceis. Negativos difíceis são exemplos de diferentes classes com algum grau de similaridade que podem ser aplicados durante o treinamento. Demonstramos que nossos modelos multitarefa contrastivos têm melhor desempenho do que o uso da função de perda de entropia cruzada, o que indica experimentalmente um apri-

moramento na verificação de assinaturas fornecida pela estrutura proposta. Como resultados bem-sucedidos em modelos de aprendizagem profunda exigem uma quantidade significativa de dados de treinamento, nesses experimentos recorremos ao conjunto de dados GPDSsynthetic com dados de assinaturas sintéticas para treinar modelos. No entanto, como resultado de observação experimental, verificamos que as características intrínsecas de assinaturas genuínas reais não estão perfeitamente incorporadas nas imagens de assinatura sintética. Dessa forma, encontramos uma diferença no desempenho de verificação entre modelos treinados usando os conjuntos de dados GPDS-960 e GPDSsynthetic; e efeitos indesejados introduzidos pelo desvio excessivo da distribuição dos modelos para qualquer uma destas fontes de dados. Dado este problema, na segunda parte desta tese recorremos ao modelo SigNet pré-treinado com dados da GPDS-960 para fornecer conhecimento sobre assinaturas reais. Complementamos o espaço de características de dados reais predefinidos na SigNet usando dados sintéticos, minimizando a divergência na distribuição entre os conjuntos de dados. Isso é obtido por meio de aprendizado contínuo incremental de classe com base na destilação de conhecimento e ajuste fino subsequente do espaço de representação com um objetivo contrastivo. Além disso, propomos um novo conjunto de dados de destilação, invertido a partir da distribuição real pré-codificada na SigNet. Nosso método proposto obtém verificação dependente de escritor aprimorada e uma verificação independente de escritor mais equilibrada, produzindo modelos mais robustos na verificação dos diversos conjuntos de dados de script ocidental e não ocidental GPDSsynthetic, GPDS-300, CEDAR, MCYT-75, BHSig-H e BHSig-B. Por fim, demonstramos que o método proposto pode ser utilizado para melhorar a verificação de assinaturas no treinamento de novas arquiteturas de visão computacional quando o acesso ao conjunto de dados GPDS-960 não estiver disponível.

Palavras-chaves: Verificação de assinaturas. Redes neurais convolucionais. Aprendizagem de características. Aprendizagem profunda. Aprendizagem contrastiva. Aprendizagem contínua. Destilação de conhecimento.

ABSTRACT

In spite of recent advances in computer vision, the classic problem of offline handwritten signature verification still remains challenging. The signature verification task has a high intra-class variability because a given user often shows high variability between its samples. Besides, signature verification is harder in the presence of skilled forgeries. Recently, in order to tackle these challenges, the research community has investigated deep learning methods for learning feature representations of handwritten signatures. When mapping signatures to a feature space, it is desired to obtain dense clusters of signature's representations, in order to deal with intra-class variability. Besides, not only dense clusters are required but also a larger separation between different user's clusters in the feature space. Finally, it is also desired to move away feature representations of skilled forgeries in relation to the respective dense cluster of genuine representations. In this thesis, we concentrate our efforts on achieving these properties in the learning of handwritten signature representations, aiming at subsequent improvements in the verification time. The research reported in this thesis is developed in two moments. Initially, we propose a framework based on the contrastive learning of handwritten signature representations. Based on the observation of the experimental results of the first part of this research, we extend this framework to handle synthetic and real signature data during training through knowledge distillation, providing greater robustness of the learned representations in the verification of diverse datasets. In the first part of this thesis, we hypothesize that desired properties can be achieved by means of a multi-task framework for learning handwritten signature feature representations based on deep contrastive learning. The proposed framework is composed of two objective-specific tasks. The first task aims to map signature examples of the same user closer within the feature space, while separating the feature representations of signatures of different users. The second task aims to adjust the skilled forgeries representations by adopting contrastive losses with the ability to perform hard negative mining. Hard negatives are examples from different classes with some degree of similarity that can be applied for training. We demonstrate that our contrastive multi-task models perform better than using cross-entropy loss, which experimentally indicates a signature verification improvement provided by the proposed framework. As successful results in deep learning models require a significant amount of training data, in these experiments we resorted to the GPDSsynthetic dataset with synthetic signature data for training models. However, as a result of experimental observation, we found that the intrinsic characteristics of real genuine signatures are not perfectly embed-

ded in the synthetic signature images. This way, we have found a difference in verification performance between models trained using the GPDS-960 and GPDSsynthetic datasets; and undesired effects introduced by excessively shifting the model distribution for any of these data sources. Given this problem, in the second part of this thesis, we resort to the SigNet model pre-trained with GPDS-960 data to provide knowledge about real signatures. We complement the SigNet predefined real data feature space using synthetic data while minimizing the divergence in distribution between the datasets. This is achieved through class-incremental continual learning based on knowledge distillation with subsequent fine-tuning of the representation space adopting a contrastive objective. Furthermore, we propose a new distillation dataset inverted from the pre-coded real distribution in SigNet. Our proposed method obtains improved writer-dependent verification and a more balanced writer-independent verification, producing more robust models in the verification of the diverse Western and non-Western script datasets GPDSsynthetic, GPDS-300, CEDAR, MCYT-75, BHSig-H, and BHSig-B. Finally, we demonstrate that the proposed method can be used to improve signature verification in the training of new large vision architectures when access to the GPDS-960 dataset is unavailable.

Keywords: Signature verification. Convolutional neural networks. Feature learning. Deep learning. Contrastive learning. Continual learning. Knowledge distillation.

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$E(\cdot)$	Energy function
$g(\cdot)$	Fusion function
$P(\cdot\|\cdot)$	Conditional probability
$p(\cdot)$	Output probabilities
$f(\cdot)$	Feature representation
$sim(\cdot,\cdot)$	Cosine similarity
m	Margin hyper-parameter
τ	Temperature hyper-parameter
γ	Kernel coefficient hyper-parameter
C	Regularization parameter
$\mathcal{W}$	Weights
$\mathcal{D}$	Development set
$\mathcal{E}$	Exploitation set
$\mathcal{V}_v$	Validation set
$\mathbf{u}$	Dissimilarity vector
$\mathbf{x}_q$	Feature vector of the questioned signature
$\mathbf{x}_r$	Feature vector of the reference signature
μ	Mean
σ	Variance
$\mathbb{E}$	Expected value
$\mathrm{KL}(\cdot,\cdot)$	Kullback-Leibler divergence
$\mathrm{JS}(\cdot,\cdot)$	Jensen-Shannon divergence

CONTENTS

1 INTRODUCTION

1.1 PROBLEM STATEMENT

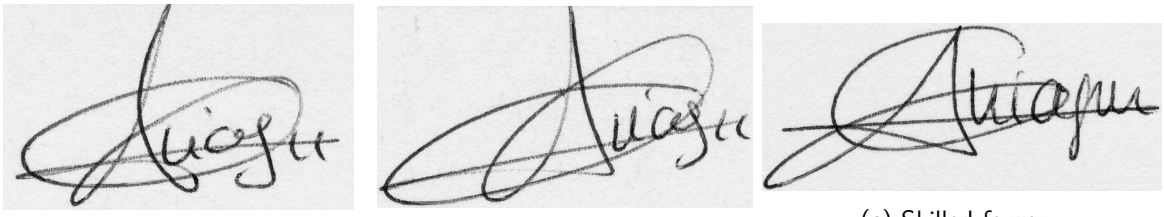
A Handwritten Signature Verification (HSV) system aims to distinguish whether a given signature provided by the user actually belongs to the claimed user (when the signature is genuine) or by an impostor (when the signature is a forgery) (MASOUDNIA et al., 2019). The handwritten signature is an essential form of biometric identification, primarily because of its widespread use in verifying a person's identity across legal, financial, and administrative domains (HAFEMANN; SABOURIN; OLIVEIRA, 2017b). Biometrics involves measuring and statistically analyzing an individual's biological characteristics for authentication purposes. These traits are universal, unique, permanent, acceptable, and reliable. Besides, they cannot be lost, stolen, or forgotten (BHAVANI; BHARATHI, 2024). One reason for the widespread adoption of handwritten signatures as biometric identification is that collecting them is a non-invasive process, and people are already accustomed to using signatures in their everyday lives (HAFEMANN; SABOURIN; OLIVEIRA, 2017b).

Handwritten signature is the most commonly used trait for verifying identity across financial institutions, law enforcement agencies, forensic departments, and governments worldwide. Its distinctive features make it a popular choice for use in banking, financial and business transactions, cheque processing, and access control (BHAVANI; BHARATHI, 2024). Besides, the growing reliance on mail-in ballots and automated signature verification systems has become increasingly evident. For instance, during the November 2020 U.S. election, the U.S. Census Bureau reported that over 154 million voters, 43% of the electorate, voted by mail (KENNEDY et al., 2025). Additionally, 27 states employed signature verification methods, many of which incorporated automated decision-making technologies (KENNEDY et al., 2025).

Manual verification of signatures by organizations and financial institutions has become increasingly difficult due to the rise in forgery cases and the large volume of samples requiring review (HAMEED et al., 2021). Determining the authenticity of a handwritten signature is a critical task (BHAVANI; BHARATHI, 2024). This challenge highlights the urgent need for a reliable automated system to quickly and accurately distinguish between genuine and forged signatures. Consequently, recent research has focused heavily on machine learning-based signature verification to support identity verification in access control, employment, finance, and security applications (HAMEED et al., 2021).

In handwritten signature verification systems, the signatures can be acquired offline or online. In the online handwritten signature verification method, signatures are captured using devices (like tablets, pens) that collects dynamic information such as pen inclination and pressure (HAMEED et al., 2021). On the other hand, in the offline handwritten signature verification method, signatures are acquired as static images by scanning (HAMEED et al., 2021). Thus, offline signature verification is a more complicated problem (HAMEED et al., 2021) due to not having dynamic information available for assisting in verification.

The forgeries can be categorized according to the knowledge of the impostor (MASOUDNIA et al., 2019): i) in the case of the *Skilled Forgeries*, the forger has information about the name and the original genuine signature pattern. ii) In the case of the *Random Forgeries*, the forger has no information about the writer. It is worth noting that genuine signatures present high intra-class variation because a given user often shows high variability between its samples (HAFEMANN; SABOURIN; OLIVEIRA, 2017b). Besides, signature verification is harder in the presence of skilled forgeries (MASOUDNIA et al., 2019), because they present low inter-class variability when considering skilled forgeries that resemble genuine signatures.



(a) Genuine example. (b) Other genuine example from the same writer. (c) Skilled forgery.

Figure 1 – Signature examples from the MCYT-75 dataset (ORTEGA-GARCIA et al., 2003).

For instance in Figures 1a and 1b, two genuine signatures obtained from the same user are shown. Note that there is a variation in the writing pattern of these two signatures due to differences in shape and size. On the other hand, a skilled forgery (shown in Figure 1c) can be carefully crafted to look like the genuine signatures (MASOUDNIA et al., 2019). Given this, the main challenge of the handwritten signature verification problem is in the design of methods that enable determining differences in the existing patterns between genuine signatures and skilled forgeries. Thus, the offline handwritten signature verification is a problem in which similarities between examples need to be estimated. When mapping signatures to a feature space, it is desired to obtain the following properties: i) Dense clusters of signature's representations for each user, in order to deal with intra-class variability. ii) A larger separation between different user's clusters in the feature space. Such a property is desirable since different user's

clusters can be easier separated by classifiers. iii) Move away feature representations of skilled forgeries in relation to the respective dense cluster of genuine representations of each user, for tackling low inter-class variability when considering skilled forgeries.

Recently, in order to tackle this problem, the research community has investigated deep learning methods for learning feature representations of handwritten signatures, instead of relying on hand-engineered feature extractor methods (HAFEMANN; SABOURIN; OLIVEIRA, 2017b). Representation learning in the context of offline signature verification systems has shown very effective results compared to hand-engineered methods (HAMEED et al., 2021). For instance, in Hafemann, Sabourin e Oliveira (2017a), deep convolutional neural networks are used to extract features from handwritten signatures images in a writer-independent way. A model with this mentioned architecture is called *SigNet* (HAFEMANN; SABOURIN; OLIVEIRA, 2017a). Subsequently, these features are employed to train *Writer-Dependent* (WD) (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) as well as *Writer-Independent* (WI) (SOUZA; OLIVEIRA; SABOURIN, 2018) models based on support vector machines in order to provide signature verification. A classifier is trained for each user in the writer-dependent verification approach, whereas in the writer-independent verification approach, a single classifier is trained for all users (HAMEED et al., 2021). Writer-dependent systems are more complex to maintain but provide better classification accuracies than writer-independent systems (SOUZA et al., 2020).

However, in spite of recent advances in computer vision and pattern recognition, offline handwritten signature verification still remains challenging (HAFEMANN; SABOURIN; OLIVEIRA, 2017b; MASOUDNIA et al., 2019). A key limitation is the presence of partial knowledge on training (HAFEMANN; SABOURIN; OLIVEIRA, 2017b) in real world scenarios, in which few genuine signature data for each user and no skilled forgeries are available for training (MASOUDNIA et al., 2019). Besides, successful results in deep learning models require a significant amount of training data (YAPıCı; TEKEREK; TOPALOĞLU, 2021) in such a way that models cannot be adequately trained with a limited number of signature examples (HAMEED et al., 2021).

In this sense, the GPDS-960 (VARGAS et al., 2007) dataset used to be the largest publicly available dataset of offline handwritten signatures for training deep learning models. For illustration, whereas GPDS-960 provides signatures of 881 users, other public datasets containing real signatures, such as CEDAR (KALERA; SRIHARI; XU, 2004) and MCYT-75 (ORTEGA-GARCIA et al., 2003), have 55 and 75 users. Nevertheless, the GPDS-960 dataset is no longer publicly available due to data protection regulatory issues stated by The General Data Protection Regulation (EU) 2016/679. Since 2016, the former researchers who possess the dataset have contin-

ued performing experiments with the GPDS-960 for research purposes (HAFEMANN; SABOURIN; OLIVEIRA, 2017a; YILMAZ; ÖZTÜRK, 2020; ARAB; NEMMOUR; CHIBANI, 2020; MARUYAMA et al., 2021; BOUAMRA et al., 2022; ARAB; NEMMOUR; CHIBANI, 2023). On the other hand, new investigators starting research in this field have suffered from the absence of a public large-scale Western script signature dataset with real signatures for training models. Therefore, as a replacement for the GPDS-960, new researchers have resorted (MAERGNER et al., 2019; YAPıCı; TEKEREK; TOPALOĞLU, 2021; ZHENG et al., 2021; VIANA et al., 2023; LI et al., 2024) to adopt the GPDSsynthetic (FERRER et al., 2017) dataset, which has a large-scale set of synthetically generated user signatures, to perform the training of deep models.

Despite this, it has been found a difference in verification performance (VIANA et al., 2023; YILMAZ; ÖZTÜRK, 2020) between models trained using real signature data from the GPDS-960 dataset and synthetic signature data from the GPDSsynthetic dataset. For instance, the *SigNet* (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) is a cross-entropy based model trained with GPDS-960 signature data whereas the *SigNet (Synthetic)* (VIANA et al., 2023) is a cross-entropy based model trained using GPDSsynthetic signature data. As a result of the performance evaluation of these models, we have observed that: i) A remarkably unsatisfactory performance for the verification of synthetic skilled forgeries and synthetic random forgeries is obtained using SigNet. ii) In the opposite situation, the SigNet (Synthetic) model has difficulty in the verification of real skilled forgeries. In other words, models trained with real data provide insufficient performance when tested in the face of a more diverse set of synthetic signatures, and models trained with synthetic data are not entirely sufficient for verifying real signatures.

Besides, we observed a trade-off in verification performance between the GPDS-960 and GPDSsynthetic datasets in such a way that a training configuration that balances verification between the two data sources must be encountered. We show that this divergence occurs due to side effects introduced by excessively shifting the model distribution for any of these data sources. Namely, we found empirical evidence that there is a distribution divergence between the GPDS-960 and GPDSsynthetic datasets, so handling this divergence should be taken into account in the training of new models.

1.2 OBJECTIVES

In this thesis, we concentrate our efforts on enhancing the learning of handwritten signature representations, aiming at subsequent improvements in the verification time. Thus, we modify

aspects of the SigNet learning method based on convolutional networks originally proposed by (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) through applying recent state-of-the-art methods in deep learning.

The research reported in this thesis is developed in two moments. Initially, we propose a framework based on the contrastive learning of handwritten signature representations. Based on the observation of the experimental results of the first part of this research, we extend this framework to handle synthetic and real signature data during training through SigNet knowledge distillation, providing greater robustness of the learned representations in the verification of diverse datasets. The two moments of this research are described below.

Firstly, we hypothesize that desired properties of handwritten signature feature representations can be achieved by means of a multi-task framework for learning representations based on deep contrastive learning. We propose a multi-task framework for offline handwritten signature feature learning comprised by a first task for writer identification followed by a second task that aims to improve representations with metric learning adopting contrastive losses optimized with hard training samples. In this scenario, we aim to investigate the following main research question:

- Do the proposed multi-task contrastive models provide significantly better feature representations for handwritten signatures than SigNet?

Secondly, we aim to obtain a more robust handwritten signature feature representation employing continual learning of synthetic data supported by SigNet knowledge distillation to minimize the divergence and combine the characteristics of real and synthetic datasets. Furthermore, we evaluate knowledge distillation performance in the training of other state-of-the-art architectures. Given this, we investigate the following main research questions:

- Can the characteristics of real and synthetic datasets be complemented using continual learning to offer a more balanced verification performance across diverse datasets?
- Is SigNet knowledge distillation helpful in training state-of-the-art large vision models for handwritten signature feature representation learning?

1.3 PROPOSED METHODOLOGY

The proposed model training framework[1] is composed of two objective-specific tasks. The first task aims to map signature examples of the same user closer within the feature space, while linearly separating the feature representations of signatures of different users. To solve this first task, the model is optimized in terms of a cross-entropy loss, following the *SigNet* approach proposed in Hafemann, Sabourin e Oliveira (2017a).

The second task aims to adjust the skilled forgery representations by adopting contrastive losses with the ability to perform hard negative mining (SCHROFF; KALENICHENKO; PHILBIN, 2015; WU et al., 2017). In the context of handwritten signature verification, a hard negative example consists of a pair of similar signatures that are obtained from different users. These hard examples can allow models to learn how to better discriminate skilled forgeries from genuine signatures even without the need for skilled forgeries during training phase. To solve this second task, two different contrastive losses having hard negative mining ability are investigated. Firstly, an Euclidean distance based metric loss (SCHROFF; KALENICHENKO; PHILBIN, 2015) in which the mining of hard examples is done in an explicit way (KHOSLA et al., 2020). Secondly, a probabilistic based metric loss (CHEN et al., 2020) in which hard negative mining is done in a implicit way (KHOSLA et al., 2020).

We demonstrate that our contrastive multi-task models that adopt the proposed framework perform better than using cross-entropy loss, which experimentally indicates a signature verification improvement provided by the proposed framework. Besides, we found that explicit hard negative mining is more effective than implicit mining.

Given these experimental results, we extend this multi-task training framework to a new scenario in which synthetic and real signature data are employed in model training. This way, we resort to a data-free knowledge transfer technique (YIN et al., 2020). Firstly, we generate inverted examples that have the same distribution as the real GPDS-960 examples. Then, we demonstrate that these inverted examples can be used alongside a pre-trained SigNet model as a foundation to provide knowledge about the characteristics of real signature data in the training of new models. In this sense, we complement the SigNet predefined real data feature space using synthetic data while minimizing the divergence in distribution between the representations provided by these two different types of data sources. This is achieved through class-incremental continual learning based on knowledge distillation, obtaining a more robust

[1] Software is available for download at <https://github.com/tallesbrito/contrastive_sigver>.

model for the verification of real and synthetic signatures.

We detail the properties of the inverted signature data and show that in the distillation process, the inverted data activates network regions of the SigNet corresponding to those of real genuine signatures with high certainty in such a way that skilled forgeries are detected from a classification uncertainty. Therefore, the examples inverted from the distribution pre-coded in the SigNet model can replace the GPDS-960 dataset to provide knowledge about real signatures in situations where this dataset is unavailable. Besides, we show that a model explicitly trained with GPDS-960 skilled forgeries over-adjusts the verification to the own GPDS-960 examples, actually making it more difficult for the model to generalize across different datasets. In contrast, the proposed method in this thesis provides a more balanced generalization across diverse datasets. In total, we evaluate the following synthetic, real, and non-Western script datasets: GPDSsynthetic (FERRER et al., 2017), GPDS-960 (VARGAS et al., 2007), CEDAR (KALERA; SRIHARI; XU, 2004) and MCYT-75 (ORTEGA-GARCIA et al., 2003); BHSig-H (Hindi) and BHSig-B (Bengali) (PAL et al., 2016).

As the examples are inverted using a data-free technique, these do not explicitly show the original signature shapes of the GPDS-960 examples, making it possible to maintain a protection level when publicly sharing the inverted dataset[2]. We evaluate the proposed method in training other state-of-the-art large vision architectures and compare it with the situation in which only synthetic data is available for training. We found that SigNet knowledge distillation is helpful in the training of new architectures as it improves verification performance. In this sense, our proposed approach can be applied to support the training of other deep backbones and representation learning schemes for signature verification so that the research community can significantly benefit from this work.

1.4 ORGANIZATION

The remainder of this thesis is organized as follows. This chapter presents a brief introduction to the investigated problem, the main objective of the work, and the main research questions.

[2] Dataset and software are available for download at <https://github.com/tallesbrito/continual_sigver>.

- Chapter 2: RELATED WORK

This chapter presents related concepts in signature verification, contrastive learning and generative methods required for a better comprehension of this work. Related methods for feature learning of handwritten signature data with writer-dependent and writer-independent verification are respectively presented in Sections 2.1 and 2.2. Besides, concepts primarily related to the contrastive losses used in deep contrastive learning and related work on contrastive learning of handwritten signature representations is discussed in Section 2.3. In Section 2.4 the related work on generative methods for handwritten signature representation learning is reviewed.

- Chapter 3: A MULTI-TASK APPROACH FOR CONTRASTIVE LEARNING OF HANDWRITTEN SIGNATURE FEATURE REPRESENTATIONS

This chapter describes the proposed framework for contrastive learning of handwritten signature representations.

The motivation for the proposed method is discussed in Section 3.1. Given this, the adopted contrastive losses in the proposed framework are defined and detailed in Section 3.2 . Besides, the training process with the proposed framework is described.

Section 3.3 presents the research questions and Section 3.4 details the associated experimental configuration and presents the results of the experiments. The experimental setup regarding the studied datasets and the training protocol are described. Besides, we performed a sensitivity analysis in order to understand the obtained effect of the contrastive losses hyper-parameters and through the proposed framework. Finally, the generalization performance of the proposed models is measured concerning different datasets in writer-dependent and writer-independent verification approaches. Given this, the obtained performance is compared to the state-of-the-art.

Section 3.5 summarizes the conclusions of this chapter.

- Chapter 4: EXPANDING GENERALIZATION OF HANDWRITTEN SIGNATURE FEATURE REPRESENTATION THROUGH DATA KNOWLEDGE DISTILLATION

This chapter describes a proposed continual learning mechanism of synthetic data based on knowledge distillation supported by inverted signature data over a predefined real feature space. The motivation for the proposed method is discussed in Section 4.1. Given this, the adopted

continual learning and data-free inversion losses in the proposed framework are formally defined and detailed in Section 4.2.

Section 4.3 presents the research questions and Section 4.4 outlines and discusses the associated experimental configuration and results obtained using the proposed method. Firstly, the properties of the inverted real data are detailed. Secondly, the experiments conducted using synthetic and real validation datasets, which was utilized to make design decisions for the proposed method, including the evaluation of hyper-parameters are detailed. Finally, the generalization performance across seven datasets is measured in writer-dependent and writer-independent verification approaches and compared with the state-of-the-art.

Section 4.5 summarizes the conclusions of this chapter.

- Chapter 5: CONCLUDING REMARKS

We present the key findings of the study and suggest directions for future research.

1.5 CONTRIBUTIONS

The present thesis stands out for presenting a state-of-the-art deep learning based approach for the learning of handwritten signature feature representations. The following papers were written during this research:

- (VIANA et al., 2022): VIANA, T. B.; SOUZA, V. L.; OLIVEIRA, A. L.; CRUZ, R. M.; SABOURIN, R. Contrastive learning of handwritten signature representations for writer-independent verification. In: 2022 International Joint Conference on Neural Networks (IJCNN). [S.l.: s.n.], 2022. p. 01–09.
  - Contribution of this paper: experiments with the proposed contrastive multi-task framework based on implicit hard negative mining are discussed and evaluated in a writer-independent verification approach.
- (VIANA et al., 2023): VIANA, T. B.; SOUZA, V. L.; OLIVEIRA, A. L.; CRUZ, R. M.; SABOURIN, R. A multi-task approach for contrastive learning of handwritten signature feature representations. Expert Systems with Applications, v. 217, p. 119589, 2023. ISSN 0957-4174.

  - Contribution of this paper: experiments with the proposed contrastive multi-task framework based on explicit and implicit hard negative mining are discussed and evaluated in a writer-dependent and writer-independent verification approaches.

- (VIANA et al., 2024): VIANA, T. B.; SOUZA, V. L. F.; OLIVEIRA, A. L. I.; CRUZ, R. M. O.; SABOURIN, R. Robust handwritten signature representation with continual learning of synthetic data over predefined real feature space. In: SMITH, E. H. B.; LIWICKI, M.; PENG, L. (Ed.). International Conference on Document Analysis and Recognition - ICDAR 2024. Cham: Springer Nature Switzerland, 2024. p. 233–249. ISBN 978-3-031-70536-6.

  - Contribution of this paper: experiments with the proposed continual learning mechanism based on knowledge distillation supported by inverted data are discussed and evaluated in a writer-dependent verification approach.

- VIANA, T. B.; SOUZA, V. L.; OLIVEIRA, A. L.; CRUZ, R. M.; SABOURIN, R. Expanding Generalization of Handwritten Signature Feature Representation through Data Knowledge Distillation. Under review.

  - Contribution of this paper: the studied multi-task contrastive framework is extended to accommodate the continual learning mechanism.

2 RELATED WORK

2.1 HANDWRITTEN SIGNATURE FEATURE LEARNING

2.1.1 Learning Feature Representations for Handwritten Signatures

Recently, deep convolutional networks have been applied to handle the offline handwritten signature verification problem. For instance, in Hafemann, Sabourin e Oliveira (2017a) are presented formulations based on deep convolutional networks for learning features of handwritten signatures in the offline scenario. Instead of relying on a handcrafted feature extraction process, this proposal focus in learning representations for handwritten signatures directly from raw data adopting a deep learning approach.

Specifically, a two-phase approach is adopted, including: a first phase for learning a feature space that captures the intrinsic properties of handwritten signatures. Such a feature representation is learned in a writer-independent way through a deep convolutional network model (adopting an architecture based on AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2017) and detailed in Table 1). A model with this architecture is called *SigNet* (HAFEMANN; SABOURIN; OLIVEIRA, 2017a). Once the SigNet model is obtained, in a second phase, features are extracted for a disjoint set of users and applied for classification using classifiers such as support vector machines in a writer-dependent way. Besides, in Souza, Oliveira e Sabourin (2018), these obtained features are applied for classification in a writer-independent way by transforming them to a dissimilarity space.

In this context, as proposed in Hafemann, Sabourin e Oliveira (2017a), given a *development* set $\mathcal{D}$ of genuine signatures of M users, a deep convolutional neural network is trained for this set of users. The last layer of this convolutional network (the *classification layer*) provides a softmax output with predicted probabilities $P(y_j|x_i)$ for each user $y_j \in \mathcal{D}$, given a genuine signature example $x_i \in \mathcal{D}$. The probabilities provided by the model estimate if signature x_i belongs to user y_j. By utilizing these probabilities, the model is trained to minimize the negative log likelihood $L_{cross\text{-}entropy}$ of the correct user by means of a *cross-entropy loss*, where $y_{ij} = 1$ if signature x_i belongs to user y_j or $y_{ij} = 0$ otherwise:

$$L_{Cross\text{-}entropy} = -\sum_{j=1}^{M} y_{ij} \log P(y_j|x_i)$$

where, i refers to a signature and j refers to a user. (2.1)

After the training process, the obtained model is applied for feature extraction of handwritten signatures belonging to a disjoint *exploitation* set $\mathcal{E}$ of users. In order to perform feature extraction, the classification layer of the model is thrown away and input signature data is propagated within the trained convolutional network until it achieves the last dense fully-connected layer (FC7 layer in Table 1). The output activations at this last dense layer are employed as feature vector for an input signature in classification tasks. Thus, the trained network (with $\mathcal{D}$ set) is used to project the input signatures of a disjoint set of users (the $\mathcal{E}$ set) onto the representation space learned by the convolutional neural network (HAFEMANN; SABOURIN; OLIVEIRA, 2017a). It is worth noting that this approach takes only genuine signatures into account when training the model. These obtained features are verified by linear classifiers in a representation space.

Layer	Size	Other Parameters
Input	$1 \times 150 \times 220$	
Convolution (C1)	$96 \times 11 \times 11$	Stride $= 4$, pad $= 0$
Pooling	$96 \times 3 \times 3$	Stride $= 2$
Convolution $(\mathrm{C2})$	$256 \times 5 \times 5$	Stride $= 1$, pad $= 2$
Pooling	$256 \times 3 \times 3$	Stride $= 2$
Convolution (C3)	$384 \times 3 \times 3$	Stride $= 1$, pad $= 1$
Convolution (C4)	$384 \times 3 \times 3$	Stride $= 1$, pad $= 1$
Convolution (C5)	$256 \times 3 \times 3$	Stride $= 1$, pad $= 1$
Pooling	$256 \times 3 \times 3$	Stride $= 2$
Fully Connected (FC6)	2048	
Fully Connected (FC7)	2048	
Fully Connected + Softmax $(P(\mathbf{y} \mid X))$	M	

Table 1 – Convolutional Neural Network layers of the *SigNet* model (HAFEMANN; SABOURIN; OLIVEIRA, 2017a).

2.1.2 Preprocessing of Signature Data

Given that the shape (considering the height and width) of signature samples belonging to the same dataset can vary, some preprocessing steps are required to obtain samples with

the input size of the convolutional neural network. To this end, in Hafemann, Sabourin e Oliveira (2017a), the signature images are firstly centered in a blank image of a maximum canvas size and subsequently resized to a fixed size. This alternative has empirically presented better results (HAFEMANN; OLIVEIRA; SABOURIN, 2018) compared to the approach of directly resizing all the signatures to a common size. Specifically in Hafemann, Sabourin e Oliveira (2017a), the following preprocessing steps are performed for each signature image: firstly the signature is centered in a large canvas by using the signatures' center of mass. The size of the large canvas is dependent on each dataset. Given this, the image background is removed using OTSU's algorithm (OTSU, 1979) which sets background pixels to white and foreground pixels in gray scale. After this, the image is inverted in order to zero-value the background. Finally, the image is resized to a fixed size.

In this thesis, we specifically intend to evaluate novel feature learning frameworks against the SigNet model (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) as a baseline. Therefore, we maintained the same preprocessing steps performed in Hafemann, Sabourin e Oliveira (2017a) in such a way that this stage cannot influence the feature learning process. Namely, all evaluated models in this work are trained with signature data preprocessed adopting the same steps originally defined in Hafemann, Sabourin e Oliveira (2017a). In the beginning of the preprocessing stage, images are centered in a large canvas size $\mathcal{S}_{canvas} = H \times W$ defined according to the maximum height and width dimensions encountered in all available image examples of the evaluated dataset. In the end of the preprocessing stage, images are resized to the 170×242 size. During the training of models, random crops of size 150×220 from the 170×242 image are taken from each signature sample as an input to the convolutional network model. In the case of testing, a center crop of size 150×220 is taken for each tested signature example. Finally, it is worth noting that we have not used data augmentation techniques to increase the dataset size for training models.

2.1.3 Datasets

The signatures in the GPDS-960 dataset (VARGAS et al., 2007) were collected from forms with 24 boxes for writing the signatures, each form being filled out by a real person. The forms for creating forgeries contain five randomly chosen genuine signatures from a given user forged three times by a human. The entire signing process was carried out under the supervision of an operator (VARGAS et al., 2007). The GPDS-960 was the largest publicly available Western

script dataset for offline signature verification with real samples of 881 users (HAFEMANN; SABOURIN; OLIVEIRA, 2017a).

A common challenge in the research of signature verification methods is the scarcity of handwritten signature samples. Moreover, legal concerns related to data protection hinder the sharing and dissemination of biometric data (FERRER et al., 2017). Motivated by these problems, a synthetic dataset named GPDSsynthetic is proposed by Ferrer et al. (2017). This synthetic dataset allows algorithm evaluation without incurring the time expenses associated with creating real datasets. Furthermore, the synthetic dataset can be shared freely without legal restrictions.

The GPDSsynthetic examples are generated from a human-like model for signature synthesis inspired by motor equivalence theory (FERRER et al., 2017). The proposed procedure for generating synthetic user signatures defines a signature morphology and a user grid based on a cognitive map. Besides, it encodes the name and flourishes as sequences of grid nodes, applies a motor model to design the signature trajectory, and generates dynamic signatures through lognormal sampling. Finally, it produces multiple genuine samples and skilled forgeries replicating real database characteristics (FERRER et al., 2017). The synthesizer is configured to mimic a real dataset by inputting morphology parameters and defining variability for different signature types. It iteratively adjusts these settings to minimize the error between the performance of the real and the synthetically generated dataset (FERRER et al., 2017). The authors of the synthesizer made publicly available a dataset with 10000 simulated users based on morphology parameters extracted from the BiosecureID-Signature UAM subcorpus dataset (FIERREZ et al., 2010).

The GPDS-960 and GPDSsynthetic datasets were produced by The Digital Signal Processing Group (GPDS) of the Universidad de Las Palmas de Gran Canaria (ULPGC-Spain). Due to this, the GPDS-960 and GPDSsynthetic datasets are named with the GPDS prefix. However, despite this similarity in the names of the datasets and the fact that both datasets contain examples with Western writing style, this does not imply that both datasets have the same distribution. Given the current difficulty of accessing the GPDS-960 dataset, which is no longer public, authors have recently performed experiments with GPDSsynthetic and have interchangeably compared (ZHENG et al., 2021; LIU et al., 2021) the performance of GPDSsynthetic with respect to previous works that performed experiments with the GPDS-960 dataset. In this thesis, we show empirical evidence for the first time that the distributions of the GPDS-960 and GPDSsynthetic datasets are not the same.

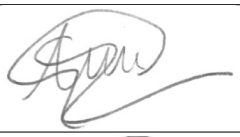
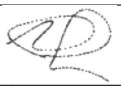
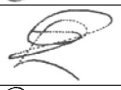
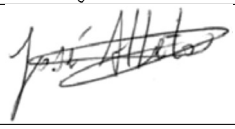
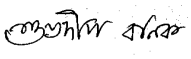
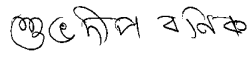
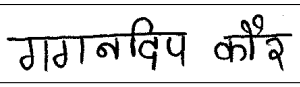
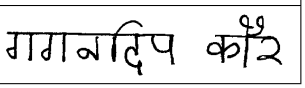
Dataset	Genuine signatures	Forged signatures
GPDS-960 (VARGAS et al., 2007)		
GPDSsynthetic (FERRER et al., 2017)		
CEDAR (KALERA; SRIHARI; XU, 2004)		
MCYT-75 (ORTEGA-GARCIA et al., 2003)		
BHSig-B (PAL et al., 2016)		
BHSig-H (PAL et al., 2016)		

Table 2 – Examples of genuine signatures and skilled forgeries from each dataset studied in this thesis. The GPDS-960 images were obtained from (HAFEMANN; SABOURIN; OLIVEIRA, 2017a). All the other images were obtained from the respective papers.

Other Western writing style datasets are evaluated in this thesis: the CEDAR (KALERA; SRIHARI; XU, 2004) and MCYT-75 (ORTEGA-GARCIA et al., 2003) datasets. The CEDAR dataset includes signatures from 55 individuals. Each genuine signer was instructed to sign within a designated area. To create skilled forgeries, unrelated individuals were asked to carefully imitate the authentic signatures of those in the dataset (KALERA; SRIHARI; XU, 2004). In the MCYT-75 dataset, forgers were given images of the genuine signatures they had to replicate. After multiple practice attempts, they were instructed to mimic the signature shapes using fluid, natural movements without pauses or slowdowns. Samples from 75 signers and their corresponding skilled forgeries were randomly chosen and digitized (ORTEGA-GARCIA et al., 2003).

Due to the lack of publicly available signature datasets for Bengali and Hindi scripts, Pal et al. (2016) developed an offline signature corpus for both languages. In the BHSig260 dataset, the signatures of 100 users were collected in the Bengali script (BHSig-B), while the signatures of 160 users were collected in the Hindi script (BHSig-H). The signatures were gathered in two separate sessions. In the first session, genuine signatures were collected. In the second session, skilled forgeries were obtained by presenting authentic signatures to the individuals and allowing them to practice and replicate the signatures (PAL et al., 2016). It is worth noting that evaluating non-Western datasets is important to measure the generalization of models

trained on Western script datasets since these datasets have more divergent writing styles.

Examples of genuine signatures and skilled forgeries from each dataset studied in this thesis are illustrated in Table 2. The number of users and the number of available genuine signature and skilled forgery examples per user in each dataset is detailed in Table 3.

Dataset Name	Users	Genuine signatures	Forgeries
GPDS-960	881	24	30
GPDSsynthetic	10000	24	30
CEDAR	55	24	24
MCYT-75	75	15	15
BHSig-B	100	24	30
BHSig-H	160	24	30

Table 3 – Summary of the datasets studied in this thesis.

2.2 HANDWRITTEN SIGNATURE VERIFICATION

2.2.1 Verifying Signatures in a Writer-Dependent Approach

In Hafemann, Sabourin e Oliveira (2017a), Support Vector Machines (SVM's) are applied to provided signature verification in a writer-dependent way. Verification can be seen as a binary classification problem, since given a questioned signature example $\mathbf{x}_q$, the classifier determines whether $\mathbf{x}_q$ is genuine or if it is a forgery. For each user, a training set is built consisting of genuine signatures from the user as positive examples and randomly obtained genuine signatures from other users as negative examples (HAFEMANN; SABOURIN; OLIVEIRA, 2017a). Therefore, in this case, a specific classifier is trained for each user of the system. The classifier establishes a decision boundary that separates a testing example from being classified as a genuine or a forgery signature concerning the considered user. In the verification process, the classifier can be tested against skilled forgeries, which are carefully created forgeries examples that resemble the genuine signatures of the target user. Skilled forgeries are harder to classify since they are intentionally created to resemble the genuine examples, and consequently their representations can be closer to the genuine signature representations in the feature space (SOUZA et al., 2019).

2.2.2 Verifying Signatures in a Writer-Independent Approach

Aiming to provide writer-independent verification, a method based on transforming handwritten signature feature representations obtained through *SigNet* to a dissimilarity space is proposed in Souza, Oliveira e Sabourin (2018). Dissimilarity vectors are obtained through a dichotomy transformation approach that converts a multi-class problem into a 2-class problem comprising (SOUZA et al., 2020): the *within class*, which constitutes the intra-class dissimilarity vectors computed from examples of the same user; and the *between class* which constitutes the inter-class dissimilarity vectors computed from samples of different users. In this way, a writer-independent classifier is trained with a set of dissimilarity vectors obtained from signatures of a development set $\mathcal{D}$ of users. Then after training such a classifier, signatures of a disjoint exploitation set $\mathcal{E}$ of users can be verified.

Formally, given that $\mathbf{x} = \mathbf{u}(\mathbf{x}_q, \mathbf{x}_r)$ is the dissimilarity vector between a questioned signature $\mathbf{x}_q \in \mathcal{E}$ and a reference signature $\mathbf{x}_r$, the writer-independent classifier determines whether the obtained dissimilarity vector $\mathbf{x}$ indicates that $\mathbf{x}_q$ and $\mathbf{x}_r$ belong to the same writer. Such a decision is based on the distance of the dissimilarity vector $\mathbf{x}$ to the support vector machine decision hyper-plane. When there are a set of reference signatures $\{\mathbf{x}_r\}_1^R$, the hyper-plane distances for each reference signature are combined through a fusion function $g(\mathbf{x}_q)$ (RIVARD; GRANGER; SABOURIN, 2011). Experiments reported in Souza, Oliveira e Sabourin (2018) showed that combining these distances by using the maximum obtained distance to the decision hyper-plane as fusion function provides better verification results.

The writer-independent classifier defines a decision boundary that delineates a region near to the origin for classifying dissimilarity vectors as belonging to the *within* class; or belonging to the *between* class, otherwise. When a questioned signature is similar to a reference signature, it is expected that the respective dissimilarity vector is positioned near to the origin in the *within* class region. Skilled forgeries are also hard to classify in a writer-independent approach since they are intentionally created to resemble the reference signatures, and consequently, their respective dissimilarity vectors are close to the origin.

2.2.3 Configuration of Support Vector Machine Classifiers

In in all experiments of this thesis, we employ soft margin Support Vector Machines (SVM) with Radial Basis Function (RBF) kernel as classifiers for writer-dependent and writer-

independent signature verification, following the experimental protocol defined in Hafemann, Sabourin e Oliveira (2017a), Souza, Oliveira e Sabourin (2018). In this work, we keep the hyperparameter configuration of SVM classifiers fixed in all experiments because our purpose is to determine improvements in feature representation learning. The SVM regularization parameter C is given by 1.0, and the RBF kernel coefficient hyper-parameter is given by $\gamma = 2^{-11}$. In case of writer-dependent verification, we adopt different weights for each class in the SVM formulation because the number of negatives examples is much greater than the number of positive examples. For the negative class the weight is given by $C^- = 1.0$, and for the positive class the weight is given by $C^+ = N/P$, where N is the number of negative training examples and P the number of positive training examples. In case of writer-independent verification, the number of dissimilarity vectors applied for training is balanced for each class. Then, $C^- = C^+ = 1.0$. Besides, we combine the hyper-plane distances for each reference signature through a *max* fusion function (SOUZA et al., 2020) in verification.

2.2.4 Verification Performance Metrics

Following Hafemann, Sabourin e Oliveira (2017a), Souza et al. (2020), in this thesis we evaluate writer-dependent and writer-independent classifiers in terms of the EER - *Equal Error Rate*. This is the error rate obtained when FRR - *False Rejection Rate* (the fraction of genuine signatures rejected as forgeries (HAFEMANN; SABOURIN; OLIVEIRA, 2017a)) is equal to FAR - *False Acceptance Rate* (the fraction of forgeries accepted as genuine (HAFEMANN; SABOURIN; OLIVEIRA, 2017a)). Note that $EER_{skilled}$ is computed using genuine signatures and skilled forgeries. The $EER_{skilled}$ is an average measure of the separation of skilled forgeries in relation to each user cluster of genuine signature representations. Besides, in order to specifically measure how skilled forgeries are verified, we report the fraction of skilled forgeries accepted as genuine $FAR_{skilled}$ in this work. The FRR and $FAR_{skilled}$ metrics are measured considering the actual distance to the decision hyper-plane, then, the adopted threshold is zero. In some specific situations of this thesis, we also measure the EER_{random}: this is the error rate obtained when FRR - False Rejection Rate is equal to FAR_{random} - *False Acceptance Rate considering Random Forgeries* (the fraction of random forgeries accepted as genuine). The EER_{random} is an average measure of separation between the clusters of the evaluated users within the representation space.

Furthermore, following Hafemann, Sabourin e Oliveira (2017a), Souza et al. (2020) pro-

tocols, each experimental configuration in this thesis is performed ten times with randomly selected data, and all reported metrics are an average of these ten replications.

2.3 CONTRASTIVE LEARNING

Contrastive learning is enumerated as one of the most recent advances in deep learning techniques. It can be defined as follows (BENGIO; LECUN; HINTON, 2021): given a pair of examples (x_i, x_j), it is desired a model to indicate whether x_i is the same as x_j according to an energy function $E(x_i, x_j)$, which provides lower values when x_i and x_j are compatible, and higher values otherwise. $E(\cdot)$ can be computed by a deep learning model trained with pairs of compatible examples (providing low energy) and pairs of incompatible examples (providing high energy) (BENGIO; LECUN; HINTON, 2021). When the energy is defined as the spatial distance between two output vectors, contrastive learning is an interesting approach to obtain good feature vectors into an embedding space, since a deep network model can be trained to provide similar feature vectors for objects belonging to the same class, and dissimilar vectors otherwise (BENGIO; LECUN; HINTON, 2021).

In the deep contrastive learning context, the energy function $E(x_i, x_j)$ is materialized as an objective *contrastive loss function* that is minimized through a model training process. Another central component of a deep contrastive model is the architecture. In the literature of deep contrastive learning, models with different architectures (KRIZHEVSKY; SUTSKEVER; HINTON, 2017; SZEGEDY et al., 2015; HUANG et al., 2017; HE et al., 2016; SIMONYAN; ZISSERMAN, 2015) have been applied for classification problems, including image classification (CHEN et al., 2020), face verification (SCHROFF; KALENICHENKO; PHILBIN, 2015) and signature verification (RANTZSCH; YANG; MEINEL, 2016). Despite the beneficial impact of evolving architectures to better deal with image classification problems, some works (WU et al., 2017; BOUDIAF et al., 2020) have experimentally shown that maintaining the same architecture but alternating to a contrastive losses for optimization can influence the performance. Choosing simpler model architectures but using contrastive losses optimization can improve the state-of-the-art results on the image classification task (CHEN et al., 2020).

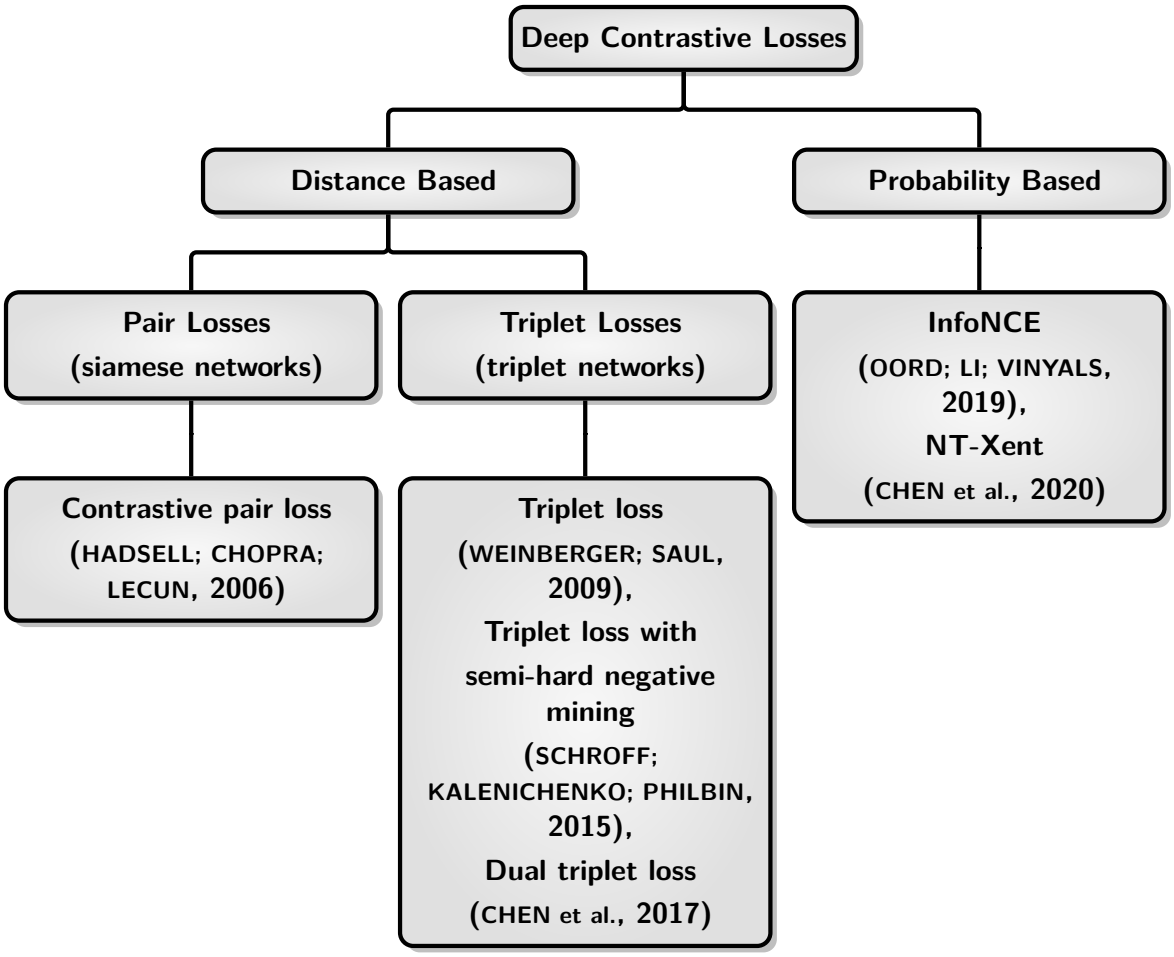


Figure 2 – Deep contrastive losses investigated in this work.

2.3.1 Contrastive Losses in Deep Learning Models

An overview of the investigated contrastive losses in this thesis is presented in this section (they are summarized in Figure 2). In Hadsell, Chopra e LeCun (2006) a loss function for mapping similar inputs to nearby points in an embedding space whereas mapping dissimilar inputs do distant points is defined, this function is referred to as contrastive *pair loss* (LE-KHAC; HEALY; SMEATON, 2020). The contrastive pair loss runs over pairs of examples (HADSELL; CHOPRA; LECUN, 2006), where the Euclidean distance is minimized for similar examples and maximized for dissimilar ones (until reach a predefined distance upper bound). Because of having to deal with pair of examples, the model architecture is a *siamese network*, consisting of two copies of the same model (sharing the same set of parameters (HADSELL; CHOPRA; LECUN, 2006)): each input is passed through each twin model and it is provided two outputs. Given these outputs, the loss is computed based on them and the model is updated through back-propagation.

The contrastive pair loss encourages examples of the same class to be projected at nearby points in the embedding space, but it does not impose a margin of separation between pairs of that class and samples of other classes (SCHROFF; KALENICHENKO; PHILBIN, 2015). Therefore, this implies that diverse classes are embedded in the same small space in such a way that the embedding space does not allow distortions, not enabling the space to tolerate outliers and adapt to different levels of intra-class variation (WU et al., 2017). Motivated by this problem, Weinberger e Saul (2009) proposed a metric optimized with the objective that, for a training set, $k-$nearest neighbors belong to the same class while examples from different classes are separated by a large margin. This metric is applied to learn a linear transformation that optimizes $k-$nearest neighbors classification for a testing set. In the deep learning context, this metric is commonly referred as *triplet loss*, and it is applied to optimize deep learning models. For instance, in (SCHROFF; KALENICHENKO; PHILBIN, 2015), the triplet loss is used to obtain feature representations for the face recognition problem. A triplet is composed of three examples: an anchor, a positive example of the same class, and a negative example of a different class. The objective of the triplet loss is to minimize the anchor-positive distance while maximizing the anchor-negative distance in the embedding space (SCHROFF; KALENICHENKO; PHILBIN, 2015). Thus, the model architecture is a *triplet network*, consisting of three instances of the same network with shared parameters (HOFFER; AILON, 2015).

It is experimentally shown in Wu et al. (2017) that adopting sampling strategies for obtaining good pairs/triplets from the training dataset leads to better feature representations, and consequently results in improved performance on classification, clustering and verification tasks. With the motivation of obtaining good triplets and to increase the difficulty of the sampled triplets as the training process runs, the authors in Schroff, Kalenichenko e Philbin (2015) proposed a strategy to obtain semi-hard negative examples when forming triplets. A semi-hard negative example is an example that resembles the positive example, but it belongs to another class than the class of the anchor and positive examples (the formal definition of a semi-hard triplet is further detailed in the Section 3.2.3.1 of this thesis). Experiments reported in Wu et al. (2017) show that effectiveness of triplet loss does not come just from the loss itself but due to the adopted sampling method. Besides, experiments reported in Schroff, Kalenichenko e Philbin (2015) showed that selecting semi-hard negatives for optimizing triplet loss improve state-of-the-art models for the face verification task.

Moreover in Chen et al. (2017), it is argued that the generalization performance of models trained with the triplet loss can be improved by reducing intra-class variations and enlarging the

inter-class variations. To this end, Chen et al. (2017) defined the *dual triplet loss* (also called quadruplet loss), which is an extension to the triplet loss that introduces a new constraint that pushes away a new pair of negative examples from the anchor and the positive example. Reported experiments by Chen et al. (2017) indicated that this new constraint can generate larger inter-class variations and smaller intra-class variations.

Probabilistic contrastive losses are generally based on *noise contrastive estimation*, in such a way that a nonlinear logistic regression is performed to discriminate between observed data and some artificially generated noise (GUTMANN; HYVäRINEN, 2010). This technique is based on comparing a target positive example with randomly sampled negative examples. Artificially generated noise is obtained from randomly obtaining negative examples from a proposal distribution (OORD; LI; VINYALS, 2019). Probabilistic contrastive losses are optimized to maximize some similarity measure between an anchor and a positive example whilst increasing the dissimilarity between this anchor and a set of negative examples. For instance, InfoNCE (OORD; LI; VINYALS, 2019) and NT-Xent (CHEN et al., 2020) loss adopt the cosine similarity as a similarity measure. Intuitively, given K negative examples, these losses are the log-loss of a $(K + 1)$-way softmax-based classifier that tries to classify as similar a questioned example as a positive example (HE et al., 2020), but the use of a normalized similarity measure adjusted by a temperature coefficient hyper-parameter enables the model to learn from hard negatives examples (JAISWAL et al., 2021). When adopting probabilistic contrastive losses, architectures with a memory bank can be adopted in order to sample negatives examples (WU et al., 2018). Recently, dictionaries that covers a rich set of negative examples instead of a memory bank have also been proposed (HE et al., 2020).

It is important to mention that the contrastive power of probabilistic losses increases with more negatives examples: the ability to discriminate between from a target class and a noise (examples from other classes) is improved when adding more negative examples in the training batches (KHOSLA et al., 2020), as empirically showed in He et al. (2020), Oord, Li e Vinyals (2019), Chen et al. (2020). Besides, probabilistic losses have an intrinsic ability to perform hard negative mining when used with normalized representations (KHOSLA et al., 2020): gradient contributions from hard negatives/positives in relation to an anchor example are larger than gradients obtained from easy negatives/positives. This property allows that probabilistic contrastive losses do not need explicit mechanisms (based on a filtering criteria) for mining negative examples, which, on the contrary, is critical when using the triplet loss (KHOSLA et al., 2020).

2.3.2 Contrastive Learning of Handwritten Signature Feature Representations

Recent state-of-the-art works have investigated the application of siamese and triplet networks in the context of offline handwritten signature verification. A proposal for learning handwritten signatures features in a writer-independent way using a siamese network is presented in Dey et al. (2017). The siamese network is composed by twin deep convolutional neural networks with shared weights adopting an architecture inspired by the AlexNet architecture (KRIZHEVSKY; SUTSKEVER; HINTON, 2017), which was previously applied for the image recognition problem. In this proposal, the siamese network is optimized through the contrastive pair loss originally defined in Hadsell, Chopra e LeCun (2006).

Siamese networks have also been applied for writer-independent feature extraction in Ruiz et al. (2020), but with a different loss formulation and adopted architecture. In this proposal, a convolutional neural network with an inception layer (SZEGEDY et al., 2015) concatenates stacks of convolutional layers in the architecture in order to increase the capacity of the network to capture details (RUIZ et al., 2020) of handwritten signatures. Real signatures, augmented signatures with simple transformations (morphological dilations, rotations and addition of noise) and synthetic generated signatures are used in training. As a result of experimental evaluation, it is concluded that lower error rates are obtained when combining these three types of signatures in training. Furthermore in Liu et al. (2021), siamese networks have been applied for feature extraction through learning from local regions of signature images, motivated by the fact that distinction of genuine signatures and forgeries lies in the details contained in local regions (LIU et al., 2021). In this case, a similarity measure obtained from different local regions are fused by average in order to obtain a final decision in signature verification. The DenseNet architecture (HUANG et al., 2017) with SE-blocks (squeeze-and-excitation-blocks) (HU; SHEN; SUN, 2018) is adopted in order to better account the difference between the two input signatures of the siamese network (LIU et al., 2021).

In Lai e Jin (2018), a deep convolutional neural network with spatial pyramid pooling layers (HE et al., 2015) is concatenated to a softmax and a siamese output layers. Lai e Jin (2018) argues that this additional contrastive layer can learn local signature similarities that are needed to refine the cross-entropy loss. Experimental results shows that adding a contrastive layer to the architecture can improve the performance of the softmax layer. However, in the experiments performed by Lai e Jin (2018), skilled forgery samples are used during the training of models. Even so, reported results are on the same level as those obtained by Hafemann,

Sabourin e Oliveira (2017a) with the SigNet model (which does not use skilled forgeries for training).

Triplet networks have recently been investigated as deep neural network models for extracting features of handwritten signatures in a writer-independent approach in Rantzsch, Yang e Meinel (2016). In this proposal, each member of the triplet networks is a deep convolutional network based on a reduced version of the VGG-16 (SIMONYAN; ZISSERMAN, 2015) architecture. Before optimizing the model using the triplet loss following the formulation defined in Hoffer e Ailon (2015), convolutional layers are pre-trained in order to distinguish the writers. The training process depends on skilled forgeries and strategies for mining semi-hard triplets during training are not investigated.

A multiple classifier system combining deep contrastive learning and a graph-based approach is proposed in Maergner et al. (2018). In order to obtain the dissimilarity score of two signatures in the graph based approach, signatures are skeletonized and compared using a graph edit distance. In addition, adopting a deep learning approach, a triplet convolutional network (based on ResNet18 architecture (HE et al., 2016)) is applied for obtaining feature representations. Given this, the distance between feature representations characterizes a complementary dissimilarity score between signatures. Experiments in Maergner et al. (2018) show that combining these approaches improves the signature verification performance. Specifically, graph based approach provides better results against skilled forgeries whereas triplet networks approach provides better results against random forgeries. However, mechanisms for mining semi-hard triplets during training are not investigated.

The formulation of dual triplets is adapted to specifically solve the offline signature verification problem in Wan e Zou (2021). In this formulation, beyond the positive and anchor examples, the dual triplets are also composed by two negative examples: the first is a random forgery and the second is a skilled forgery. The obtained performance with this contrastive loss is better than using skilled forgery examples during training with cross-entropy loss optimization with an additional binary neuron in the architecture that indicates whether a training example is a genuine or a skilled forgery. However, such an approach has the evident drawback of using skilled forgeries during model training.

In Tsourounis et al. (2022), a deep convolutional neural network using the same architecture of *SigNet* (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) model is trained using handwritten textual data. They state that learning features in the domain of handwritten textual data can be transferred to the domain of handwritten signature data. After training models in the aux-

iliary domain of textual data, features are adjusted to the specific domain using a finetuning step but with two different strategies. The first strategy is to finetune the auxiliary domain model with handwritten signatures images. In the second strategy, the features learned by the auxiliary domain model are combined in pairs and used to learn a mapping function through the contrastive pair loss defined in Hadsell, Chopra e LeCun (2006). Despite of interesting proposal, experimental results reported in Tsourounis et al. (2022) do not present statistically significant improvement in verification error rates when the proposed models are compared to the *SigNet* model. Even though absolute values of reported error rates in Tsourounis et al. (2022) are generally lower than those presented in Hafemann, Sabourin e Oliveira (2017a), this effect is attributed due to a user-specific grid search procedure in the training process of the support vector machine classifiers applied for signature verification. Therefore, improvements are associated with specific hyper-parameter adjustments for each user in signature verification and not to better-obtained feature representations.

Soleimani, Araabi e Fouladi (2016) proposed a neural network model for signature verification composed by a shared layer for all users, which is followed by separate layers that distinguish signatures of each specific user. The shared layer learns writer-independent characteristics of forged and genuine signatures from all individuals, whereas the separated layers for each user learns writer-dependent specific factors. During the training process, these both tasks are learned simultaneously with the objective to minimize the Euclidean distances of pair of signatures from the same user while maximizing the distance of genuine and random forgery samples. A disadvantage of this architecture is the computational cost for model maintenance in the addition of new users as the training of the writer-independent layers is dependent on the back-propagation coming from the user-specific layers.

The Table 4 summarizes in which aspects our proposed method differs from the related work already existent in the literature. We propose and evaluate a multi-task framework that adopts contrastive losses to better adjust the skilled forgery representations within the feature space. In order to separately understand the effect of the proposed framework, we maintained the same convolutional network architecture adopted by the *SigNet* model (HAFEMANN; SABOURIN; OLIVEIRA, 2017a), since this way we can do fair comparisons with the SigNet model. In Tsourounis et al. (2022), it is also kept the SigNet architecture with the similar purpose of evaluating a different framework for training deep convolutional models, but, in general, the model architectures adopted by related works in the literature vary.

In some works (DEY et al., 2017; RUIZ et al., 2020; LIU et al., 2021; WAN; ZOU, 2021), model

Table 4 – Comparison of related work on contrastive learning of handwritten signature representations.

	A	B	C	D	E	F	G	H	I	Present work
Network architecture										
AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2017)	X						X	X		X
Inception layers (SZEGEDY et al., 2015)		X								
DenseNet (HUANG et al., 2017)			X							
CNN with spatial pyramid pooling (HE et al., 2015)				X						
VGG-16 (SIMONYAN; ZISSERMAN, 2015)					X					
ResNet18 (HE et al., 2016)						X				
DMML (SOLEIMANI; ARAABI; FOULADI, 2016)									X	
Training tasks										
Single task	X	X	X				X			
Multiple tasks with transfer learning between sequential tasks					X			X		X
Multiple tasks combined with a multiple classifier system						X				
Multiple tasks learned simultaneously				X					X	
Optimization loss										
Cross-entropy loss				X				X		X
Contrastive pair loss (HADSELL; CHOPRA; LECUN, 2006)	X							X		
Binary cross-entropy loss over normalized distances of pair samples		X	X	X						
Mean squared error loss over normalized distances of triplet samples (HOFFER; AILON, 2015)					X					
Triplet loss (WEINBERGER; SAUL, 2009)						X				
Generalized logistic loss over Euclidean distances of pair samples (HU; LU; TAN, 2014)									X	
Dual triplet loss (CHEN et al., 2017)							X			
Triplet loss with semi-hard negative mining (SCHROFF; KALENICHENKO; PHILBIN, 2015)										X
NT-Xent loss (CHEN et al., 2020)										X
Signature verification										
Euclidean distance (L_2) threshold	X				X	X	X		X	
Manhattan distance (L_1) threshold		X	X							
Graph edit distance threshold (MAERGNER et al., 2017)						X				
Using auxiliary classifier (SVM)				X				X		X
Verification approach										
Writer-dependent (WD)			X	X				X	X	X
Writer-independent (WI)	X	X	X		X	X	X			X
Signature sampling										
Training using only genuine signatures		X				X		X	X	X
Testing with genuine signatures and skilled forgeries	X		X	X	X	X	X	X	X	X

Related works: [A] (DEY et al., 2017) [B] (RUIZ et al., 2020) [C] (LIU et al., 2021) [D] (LAI; JIN, 2018)
[E] (RANTZSCH; YANG; MEINEL, 2016) [F] (MAERGNER et al., 2018) [G] (WAN; ZOU, 2021)
[H] (TSOUROUNIS et al., 2022) [I] (SOLEIMANI; ARAABI; FOULADI, 2016)

training is performed to optimize a single objective (in Table 4, these works are categorized as single task approaches). Other works define optimization steps and goals for certain specific problems or domains (for simplicity, in Table 4, these works are categorized as multiple task approaches). For instance, in Tsourounis et al. (2022), models are initially optimized using data from an auxiliary domain, and in a following step, models are optimized in the specific domain of handwritten signatures. Besides, in Maergner et al. (2018), a metric learning approach is combined with a graph-based method. In Soleimani, Araabi e Fouladi (2016), writer-dependent and writer-independent characteristics are learned simultaneously.

Also adopting a multiple task approach, our proposed framework has different training stages with specific optimization goals. Initially, models are trained to distinguish between users. After that, models are adjusted to obtain better representations for skilled forgeries. To achieve this second goal, in our proposed framework we adopt optimization loss functions that have not been investigated in the related work. To the best of our knowledge, these losses have not been yet specifically investigated for the problem of obtaining handwritten signature representations. Moreover, we evaluated obtained feature representations with the proposed framework in writer-dependent and writer-independent approaches, which is an aspect that is not considered in the most of related works.

Although most of the works (DEY et al., 2017; RUIZ et al., 2020; RANTZSCH; YANG; MEINEL, 2016; MAERGNER et al., 2018; LIU et al., 2021; SOLEIMANI; ARAABI; FOULADI, 2016; WAN; ZOU, 2021) use some distance threshold to determine whether or not two signatures belong to the same user, in this work, we use an auxiliary support vector machine classifier for verifying signature representations. This is advantageous because signatures can be verified in relation to a set of reference signatures, as the support vector machine classifier is trained with a collection of signature samples. This way, we can evaluate and understand the performance of the proposed method when varying the number of reference signatures.

Besides, it is interesting to point out that we use only genuine signatures for training models in this work, with models being tested in the presence of genuine signatures and skilled forgeries. This design choice is more suitable to the real world applications, in which forgeries are not available in training but the methods must still be subject to being tested with forgeries. Note in Table 4 that some related works (DEY et al., 2017; RUIZ et al., 2020; LIU et al., 2021; RANTZSCH; YANG; MEINEL, 2016; WAN; ZOU, 2021; LAI; JIN, 2018) do not follow this premise. For instance in Ruiz et al. (2020), even though models are trained using only genuine signatures, they are not tested using skilled forgeries. Such a testing protocol is insufficient to

measure the performance of the investigated method in the presence of skilled forgeries.

2.4 GENERATIVE NETWORKS

Generative adversarial networks (GANs) (GOODFELLOW et al., 2014) have stood out as state-of-the-art methods for synthesizing new images (YIN et al., 2020). In the context of signature verification systems, generative adversarial networks have recently been investigated for generating new signature examples in order to deal with the limited availability of signature data when training new models. Besides, generative adversarial networks (and other deep learning methods) have been widely explored for signature image synthesis, seemingly surpassing conventional mathematical methods (DIAZ et al., 2025).

In this thesis, we propose and adopt a dataset with examples generated from the pre-coded signature distribution in the SigNet model, aiming to transfer knowledge about real signatures in the training of new models through knowledge distillation. Knowledge transfer by distillation was originally conceived to transfer learning between an expert teacher model to different architectures (HINTON; VINYALS; DEAN, 2015). However, these methods rely on the original dataset used in training. More recent research (YIN et al., 2020) has explored data-free distillation, in which pre-existing knowledge from a teacher model can be transferred without directly accessing the original dataset. To achieve this, training examples are inverted and generated from the distribution encoded in the pre-trained model. Data-free methods have a different approach from generative adversarial networks, which require access to the original dataset to capture the data distribution in order to generate new examples.

2.4.1 Knowledge Distillation Supported by Generated Inverted Data

In the absence of prior data to support knowledge distillation, researchers have explored recovering training data from pre-trained models for knowledge transfer. Methods like *DeepDream* (MORDVINTSEV; OLAH; TYKA, 2015) optimize input images to produce high responses for specific output classes. However, these generated images prove ineffective for knowledge transfer, as shown in the experiments reported in Yin et al. (2020).

Motivated by this problem and considering that high-performing convolutional neural networks use batch normalization layers that store running means and variances of activations, effectively capturing the history of past data. The *DeepInversion* (YIN et al., 2020) method

generates inverted images by introducing a regularization term based on layer-wise mean and variance, eliminating the need for training data or metadata. This approach facilitates knowledge transfer between networks, even with different architectures, achieving minimal accuracy difference compared to the situation where the original training data is employed in the distillation process. Besides, DeepInversion can be used for continual learning, allowing new classes to be added to a pre-trained neural network without requiring access to the original data (YIN et al., 2020).

2.4.2 Generative Methods for Handwritten Signature Representation Learning

Initial efforts to generate signature examples using generative adversarial networks are presented in Zhang, Liu e Cui (2016) so that generated training examples are used for training a discriminator based on convolutional networks (RADFORD; METZ; CHINTALA, 2015) in an unsupervised manner. The discriminator is employed to extract features verified by a hybrid writer-dependent and writer-independent classifier. In the experimental evaluation, the generative model is trained with GPDS-960 data, but tests are also performed with classifiers trained with GPDSsynthetic data. In such experiments, it is already possible to notice a more outstanding difficulty in generalizing the features learned with GPDS-960 for classifying GPDSsynthetic examples. In Yonekura e Guedes (2021), a similar approach is proposed, with the difference that the generative model is trained in a supervised manner, aiming to add an extra layer of control in the training of the generator and the discriminator. However, even with this additional restriction, the experimental results in Yonekura e Guedes (2021) are far below those presented by the state-of-the-art.

In Yapıcı, Tekerek e Topaloğlu (2021), a Cycle-GAN generative network (ZHU et al., 2017) is used to learn high-level signature features through an image-to-image translation process and thus create new augmented examples for a given user. These augmented examples are used to train classifiers based on Capsule Networks (SABOUR; FROSST; HINTON, 2017) that consider spatial relationships between input features in a writer-dependent approach. A drawback of the method is that a distinct generative model is trained for each user, introducing more complexity in systems with a large number of users.

Also using a Cycle-GAN but for an on-2-off scenario where offline signatures are generated from online signatures, in Jiang et al. (2022) online signature datasets (TOLOSANA et al., 2021) are used to train a single generative network that provides new augmented offline examples

Table 5 – Comparison of related work on generative methods for handwritten signature representation learning.

	A	B	C	D	E	Present work
Generative method						
Inversion (YIN et al., 2020)						X
Cycle-GAN (ZHU et al., 2017)	X	X				
Adversarial variation network (LI; WEI; HU, 2022)			X			
Generative adversarial network (GOODFELLOW et al., 2014)				X		
Conditional generative adversarial network (YONEKURA; GUEDES, 2021)					X	
Network architecture						
AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2017)						X
Capsule Network (SABOUR; FROSST; HINTON, 2017))	X					
SigCNN (JIANG et al., 2022)		X				
VGG-16 (SIMONYAN; ZISSERMAN, 2015)			X			
Deep convolutional generative adversarial network (RADFORD; METZ; CHINTALA, 2015)				X	X	
Signature verification						
SVM classifier						X
Euclidean distance (L_2) threshold	X	X				
Probability threshold			X		X	
Gentle Adaboost (FRIEDMAN; HASTIE; TIBSHIRANI, 2000)				X		
Verification approach						
Writer-dependent (WD)	X	X		X	X	X
Writer-independent (WI)			X	X		X
Signature sampling						
Training using only genuine signatures		X		X	X	X
Testing with genuine signatures and skilled forgeries	X	X	X	X	X	X

Related works: [A] (YAPıCı; TEKEREK; TOPALOĞLU, 2021) [B] (JIANG et al., 2022)
[C] (LI; WEI; HU, 2022) [D] (ZHANG; LIU; CUI, 2016) [E] (YONEKURA; GUEDES, 2021)

from a style transfer perspective. In this way, the generative model provides examples employed to train a convolutional model for feature representation extraction, and representations are verified based on distance in a writer-dependent approach. Furthermore, generated examples in lower distortion act as augmented genuine data, and those in higher distortion act as forgeries in the training of the feature extractor, eliminating the use of real forgeries.

Based on the hypothesis that colors or intensities of signature images should not change the verification decision in such a way that the model decision in verification must be focused on the signature stroke information, a framework based on a generative method called adversarial variation network is proposed in Li, Wei e Hu (2022). In this framework, a variator module produces variation maps applied in training examples to generate signature image variants. The augmented examples are employed to train a feature extractor and a discriminator, which

determines the decision in signature verification. Despite the interesting contribution, models in Li, Wei e Hu (2022) are trained and tested using partitions of the same dataset. Considering that examples from the same dataset have the same acquisition process, training and testing models with partitions from the same dataset can introduce specific dataset biases, which facilitates the solution of the problem. In contrast, in this thesis, we use prior knowledge about the GPDS-960 dataset with GPDSsynthetic data examples for training, and models are tested on a broad set of datasets with different acquisition processes and scripts such that different populations of writers are considered.

Table 5 summarizes the discussed related works and qualitatively compares them with this work. In our proposed work, we generate new examples that come from a real distribution pre-coded in a neural network. However, we use a data-free inversion method to achieve this. To the best of our knowledge, this is the first work in which a deep neural network inversion method is used to generate examples that represent characteristics of real signatures.

3 A MULTI-TASK APPROACH FOR CONTRASTIVE LEARNING OF HANDWRITTEN SIGNATURE FEATURE REPRESENTATIONS

3.1 MOTIVATION

Signature verification task has a high intra-class variability (HAFEMANN; SABOURIN; OLIVEIRA, 2017b) because users often show high variability between their samples. This implies that the distribution of feature representations for a given user can present a higher variance, providing scattered clusters that can overlap in the feature space. This situation is illustrated in Figure 3a. Due to that, in the case of writer-dependent verification, genuine signatures of other users can be wrongly accepted since a given user can have feature representations that overlap the clusters of other users. After transforming these feature representations into a dissimilarity space, vectors of *within* class tend to be more scattered around the region near to the origin. Furthermore, dissimilarity vectors of *between* class tends to be scattered into the whole space, also overlapping the *within* class vectors (as shown in Figure 3c).

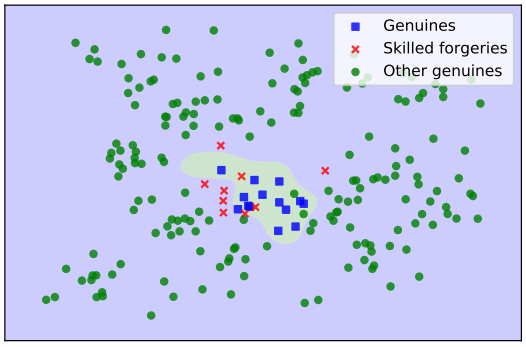
Signature verification is a harder task when skilled forgeries are present since they have low inter-class variability (HAFEMANN; SABOURIN; OLIVEIRA, 2017b). As illustrated in Figure 3a, in writer-dependent approach, skilled forgeries can be miss-classified as genuine signatures because they are generally closer to the decision hyper-plane around the cluster of genuine examples. In the case of writer-independent verification, as shown in Figure 3c, since skilled forgeries resemble reference signatures, the associated dissimilarity vectors obtained from comparing them can be closer to the region of *within* class, in the origin, resulting in miss-classification. Given these problems, the motivation of this work is to experiment a framework for obtaining offline handwritten signature feature representations, specifically aiming to achieve the following properties:

Property 1. *Obtain denser clusters of signature's representations for each user, dealing with high intra-class variability.*

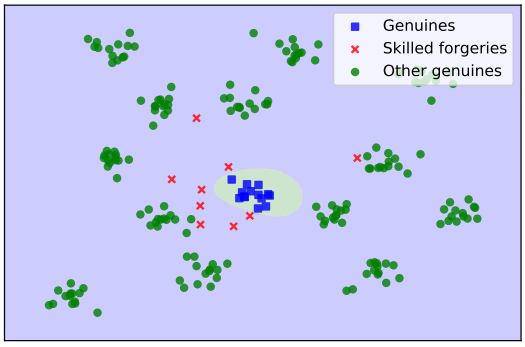
Property 2. *Not only dense clusters are required but also a larger separation between different user's clusters in the representation space. Such a property is desirable since different user's clusters can be easier separated by classifiers.*

Property 3. *Move away representations of skilled forgeries in relation to the respective dense cluster of genuine representations of each user, dealing with low inter-class variability when considering skilled forgeries and genuine signatures.*

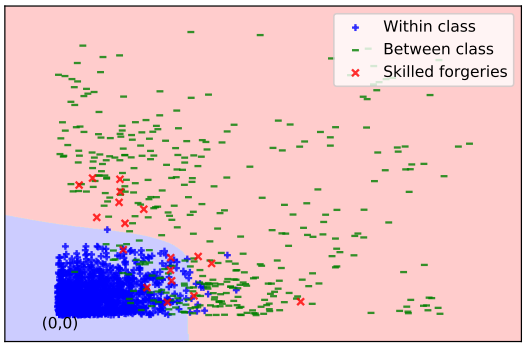
The effect of obtaining the properties (1) and (2) is illustrated in Figure 3b. Having these properties, features representations for each user are positioned in a dense cluster, and also



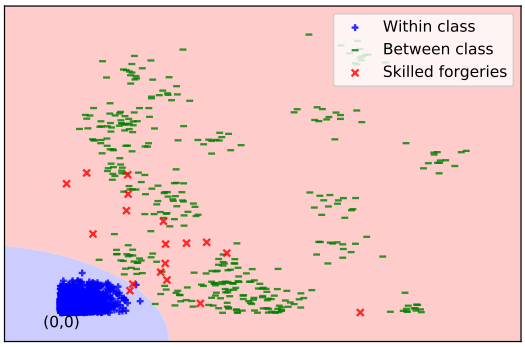
(a) In Writer-Dependent (WD) approach, an user-specific classifier determines whether a questioned signature is a genuine or a forgery.



(b) In Writer-Dependent (WD) approach, a better separated dense cluster of a given user is easier to be separated from the other user's clusters (achieving properties 1 and 2). Besides, better skilled forgery representations are further away from the respective dense cluster of genuine signature feature representations of a given user (achieving property 3).



(c) In Writer-Independent (WI) approach, a classifier determines whether a questioned signature is a genuine or a forgery when compared to a reference signature.



(d) In Writer-Independent (WI) approach, a better clearer separation of *within* and *between* dissimilarity vectors makes easier to classify dissimilarity vectors (achieving properties 1 and 2). Besides, better skilled forgery dissimilarity vectors are further away from the *within* class region (achieving property 3).

These examples are figurative, and we artificially generated the examples in Figures 3a and 3b. Random clusters with a default variance were generated in Figure 3a. We decreased the variance of clusters representing genuine signatures and increased the variance of the cluster representing skilled forgeries to obtain Figure 3b from Figure 3a. Applying the dichotomy transformation, we transformed the artificial data from Figure 3a to the data in the dissimilarity space illustrated in Figure 3c. Likewise, data from Figure 3b were transformed to Figure 3d.

Figure 3 – Hypothetical handwritten signatures represented in the feature space and in the dissimilarity space, respectively verified in writer-dependent and writer-independent ways.

these clusters are separated within the feature space. Then, in writer-dependent verification, a given cluster of genuine signatures is easier to be separated from the other clusters. In the case of writer-independent classification, as illustrated in Figure 3d, since clusters in the feature space are denser, consequently, the obtained vectors of *within* class in the dissimilarity space are also denser and nearer to the origin, with a clearer separation to the *between* class vectors. These, in turn, are less scattered within the dissimilarity space.

Secondly, the effect of obtaining the property (3) is also illustrated in Figure 3b. Having this property, skilled forgeries are further away from the respective cluster of genuine signatures for a given user. Thus, these skilled forgeries are easier to be classified in a writer-dependent approach. In the case of writer-independent classification, as illustrated in Figure 3d, since these representation are further away from the representations of reference signatures, it is obtained dissimilarity vectors that are further away from the region of *within* class vectors, and thus, improving the classification of skilled forgeries in this scenario.

3.2 MULTI-TASK HANDWRITTEN SIGNATURE REPRESENTATION LEARNING

In this work, we hypothesize that properties (1), (2) and (3) can be achieved by means of a multi-task framework for learning handwritten signature feature representations based on deep contrastive learning. Multi-task learning is a mechanism that objectives to improve generalization performance by taking advantage of learning information contained in different domain-specific related tasks (CARUANA, 1997). In the proposed framework, we adopt a transfer learning approach in which the representation space learned through a first task is used as a primary structure for learning a representation space adjusted by a second task, providing an information flow for learning feature representations. Specifically, in the first task, the properties (1) and (2) are achieved through learning a feature space that separates feature representations of genuine signatures of different users. In sequence, in the second task, property (3) is achieved by applying contrastive losses in order to better discriminate skilled forgeries. We hypothesize that solving the first task is beneficial because it can improve the generalization performance by defining a more favorable initial state for applying a second step that adjusts specific characteristics of the learned representation space (MASOUDNIA et al., 2019; RANTZSCH; YANG; MEINEL, 2016; TSOUROUNIS et al., 2022; SOLEIMANI; ARAABI; FOULADI, 2016). These two tasks are outlined in Figure 4 and described in the following subsections.

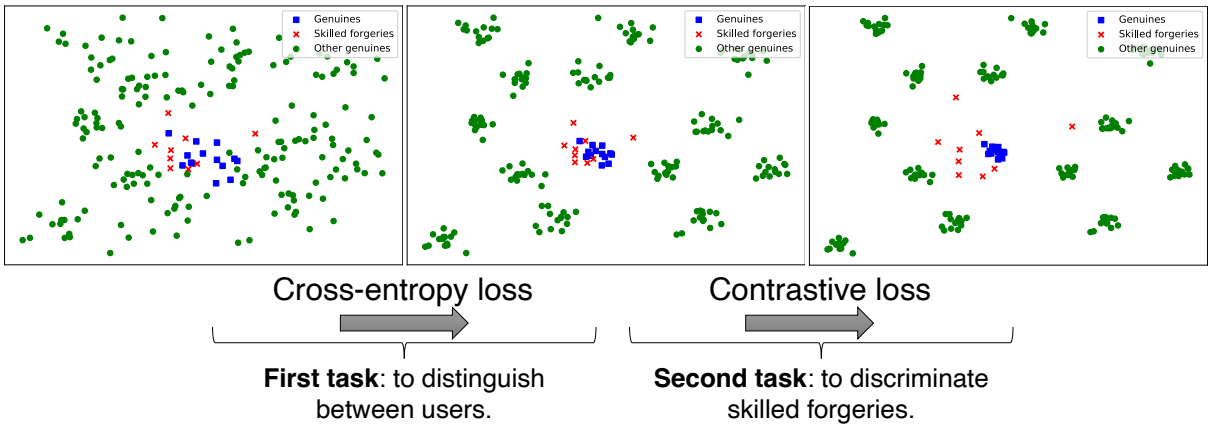


Figure 4 – An overview of the proposed multi-task framework for contrastive learning of handwritten signature feature representations.

3.2.1 First Task: to Distinguish between Users

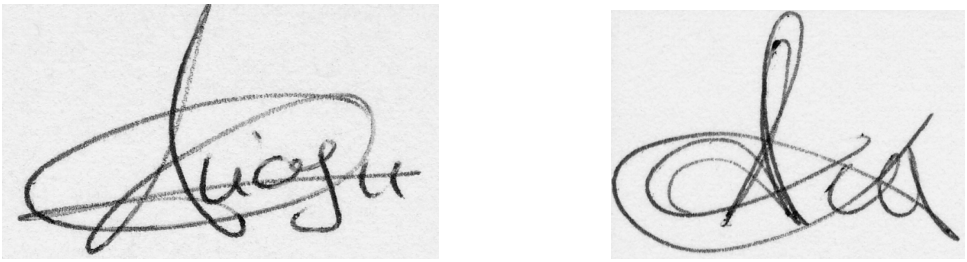
As proposed in Hafemann, Sabourin e Oliveira (2017a), a deep convolutional model is trained to distinguish between signatures of different users in such a way that representations of signatures from these users are separable in the representation space. The model is optimized in terms of a cross-entropy loss (following the Equation 2.1). Following this idea, the objective of the first task of the proposed framework is to encourage signature examples of the same user to be close to each other in the feature space, whereas pushing away feature representations of signatures obtained from different users. This enables reaching the properties (1) and (2). Besides, the learned feature space in this first task generalizes well for verification of signatures of other users in other datasets which are not used for training the model. It is experimentally shown in Hafemann, Sabourin e Oliveira (2017a), Souza et al. (2020) that this learned feature space generalizes well to other datasets in transfer learning scenarios.

3.2.2 Second Task: to Discriminate Skilled Forgeries

In the proposed framework, we hypothesize that applying contrastive losses with the ability to better separate similar examples of different classes can be advantageous for obtaining feature representations of handwritten signatures. It is experimentally shown in Hafemann, Sabourin e Oliveira (2017a) that adding skilled forgeries information during training can result in lowering the verification error rates, but this is not feasible in practice because usually, in real-world scenarios, skilled forgeries are not readily available. Following this idea, the objective of the second task in the proposed framework is to allow models to learn to differentiate better

skilled forgeries from genuine signatures when moving the representations of skilled forgeries from the respective genuine representations, even without requiring to provide skilled forgeries during training.

A hard negative pair consists of a pair of similar examples but from different classes (SCHROFF; KALENICHENKO; PHILBIN, 2015). In the context of handwritten signature verification, a hard negative pair can be seen as a pair of similar signatures that are obtained from different users. For instance, Figure 5 illustrates a pair of similar signatures (x^a, x^n), but the anchor example x^a and the negative example x^n are obtained from different users. In this scenario, contrastive losses with the ability to perform hard negative mining encourage feature representations of similar signatures obtained from different users (hard negatives) to be pushed away by a larger separation in the embedding space. We expect that this allows for refining properties (1) and (2) and consequently reaching property (3).



(a) Anchor example x^a. (b) Negative example x^n.

Figure 5 – A *hard negative pair of signatures*. This pair of signatures presents similarities but they were obtained from two different users of MCYT-75 dataset (ORTEGA-GARCIA et al., 2003).

In this sense, two different contrastive losses having the ability to perform hard negative mining are investigated in this work:

i) A distance based loss with explicit hard negative mining: *the triplet loss with semi-hard negative mining* (SCHROFF; KALENICHENKO; PHILBIN, 2015).

ii) A probabilistic based loss with implicit hard negative mining: *the normalized temperature-scaled cross entropy loss (NT-Xent loss)* (CHEN et al., 2020).

In the triplet loss with semi-hard negative mining, hard negative pairs that constitute the triplets applied for training are filtered based on the Euclidean distance of the negative example in relation to an positive example of another class. The purpose is to select negative examples near to a positive example within a predefined margin. Selecting these semi-hard negative pairs is based on some concrete spatial metric, in such way that mining these training examples is done in an explicit way (KHOSLA et al., 2020).

On the other hand, in the NT-Xent loss, hard negative pairs are not selected based on a explicit filtering criteria. The formulation of NT-Xent loss takes advantage of hard negative pairs, because gradient contributions of the hard negative pairs during training provide a stronger model correction in comparison to the easy negative pairs. Since an explicit filtering criteria is not performed in NT-Xent loss, it is considered that obtaining the contributions of hard negative examples is done in an implicit way (KHOSLA et al., 2020).

In this work, we investigate the effect of using explicit or implicit mechanisms for mining hard negative examples during training to obtain feature representations for handwritten signature verification. The investigated losses are further detailed in the following subsections.

3.2.3 Investigated Losses

3.2.3.1 Triplet Loss with Semi-hard Triplet Mining

A triplet network is composed of three instances of the same feed-forward network with shared weights (HOFFER; AILON, 2015). The network is fed with three examples: a anchor example x_i^a, a positive example x_i^p and a negative example x_i^n. Given this, the network encodes the pair of distances between the negative and positive example against the anchor example (HOFFER; AILON, 2015). The network models a function $f(x_i)$ that embeds an input example x_i into an Euclidean space. Namely, $f(x_i)$ is the feature representation obtained from the network for an input example x_i. With respect to the offline handwritten signature verification problem, for any triplet (x_i^a, x_i^p, x_i^n), the anchor example x_i^a and the positive example x_i^p are signatures of the same user, whereas the negative example x_i^n is a signature of a different user than the anchor user.

In the training phase, the objective of the triplet loss is to minimize the anchor-positive distance while maximizing the anchor-negative distance. Thus, given a training set of N triplets (x_i^a, x_i^p, x_i^n), the loss being minimized (SCHROFF; KALENICHENKO; PHILBIN, 2015) is:

$$L_{Triplet} = \frac{1}{N}\sum_{i}^{N}[m + \|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2)]_+$$

$$\text{where, } [\bullet]_+ = \max(0, \bullet) \quad (3.1)$$

In the Equation 3.1, m is the adopted margin when optimizing the network with the triplet loss. This margin controls how much the feature representation of a negative example should

be pushed away from the feature representation of a positive example, within a margin m. When $\|f(x_i^a) - f(x_i^n)\|_2^2 > m + \|f(x_i^a) - f(x_i^p)\|_2^2$, the network is not optimized because $L_{triplet} = 0$. Consequently, in order to ensure fast convergence when training the network (SCHROFF; KALENICHENKO; PHILBIN, 2015), it is necessary to select triplets that violate this constraint. In other words, triplets such that: $\|f(x_i^a) - f(x_i^n)\|_2^2 < m + \|f(x_i^a) - f(x_i^p)\|_2^2$ should be selected in order to improve the model optimization during training.

Therefore, the margin m is applied to narrow the selection of triplets during training. In this case, triplets such that:

$$\|f(x_i^a) - f(x_i^n)\|_2^2 < m + \|f(x_i^a) - f(x_i^p)\|_2^2 \tag{3.2}$$

are selected. Selecting the hardest negatives which are nearer to the anchor in relation to the positive examples can lead to bad local minima early on training (SCHROFF; KALENICHENKO; PHILBIN, 2015), resulting in a collapsed model. Preliminary experiments (which are not explored in this study) selecting those hardest negatives indicated that collapsed models such that $f(x_i) = 0$ were obtained. Due to this problem, only triplets in which the negative example is further away from the anchor than the positive example ($\|f(x_i^a) - f(x_i^p)\|_2^2 < \|f(x_i^a) - f(x_i^n)\|_2^2$) within a margin m are selected. Such triplets are called as *semi-hard* triplets (SCHROFF; KALENICHENKO; PHILBIN, 2015).

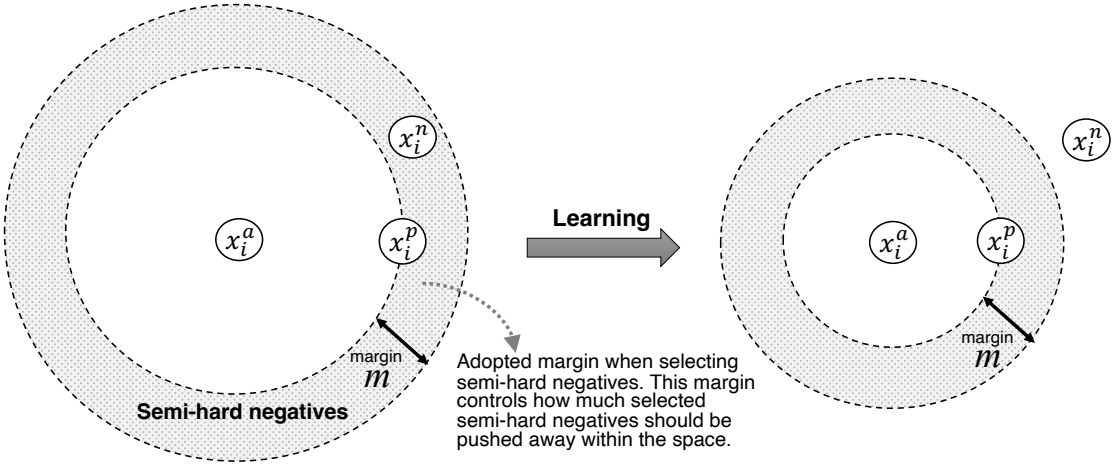


Figure 6 – A graphic example of a semi-hard triplet (x_i^a, x_i^p, x_i^n). It is shown where a semi-hard negative x_i^n is positioned in relation to an anchor x_i^a and a positive example x_i^p.

Figure 6 shows an graphic example of a semi-hard triplet (x_i^a, x_i^p, x_i^n). In this figure, it is illustrated where a semi-hard negative x_i^n is positioned in relation to an anchor x_i^a and a positive example x_i^p. Semi-hard negatives are further away within an margin m from the positive example. When optimizing the model, selected semi-hard negatives feature representations are pushed away within a margin m into the space.

3.2.3.2 Normalized Temperature-Scaled Cross Entropy Loss

The Normalized Temperature-Scaled Cross Entropy Loss (referred as NT-Xent loss in Chen et al. (2020)) is a contrastive loss for maximizing the similarity of representations of a pair composed by an anchor and a positive example whilst minimizing the similarity between this anchor and a set of negative examples. With respect to the offline handwritten signature verification problem, for any positive pair (x_i^a, x_j^p), the anchor example x_i^a and the positive example x_j^p are signatures obtained from the same user, whereas for any negative pair (x_i^a, x_k^n), the anchor example x_i^a and the negative example x_k^n are signatures obtained from different users. Therewith, the loss function for all positive pairs of examples (x_i^a, x_j^p) obtained from a training set with N positive pairs is defined as:

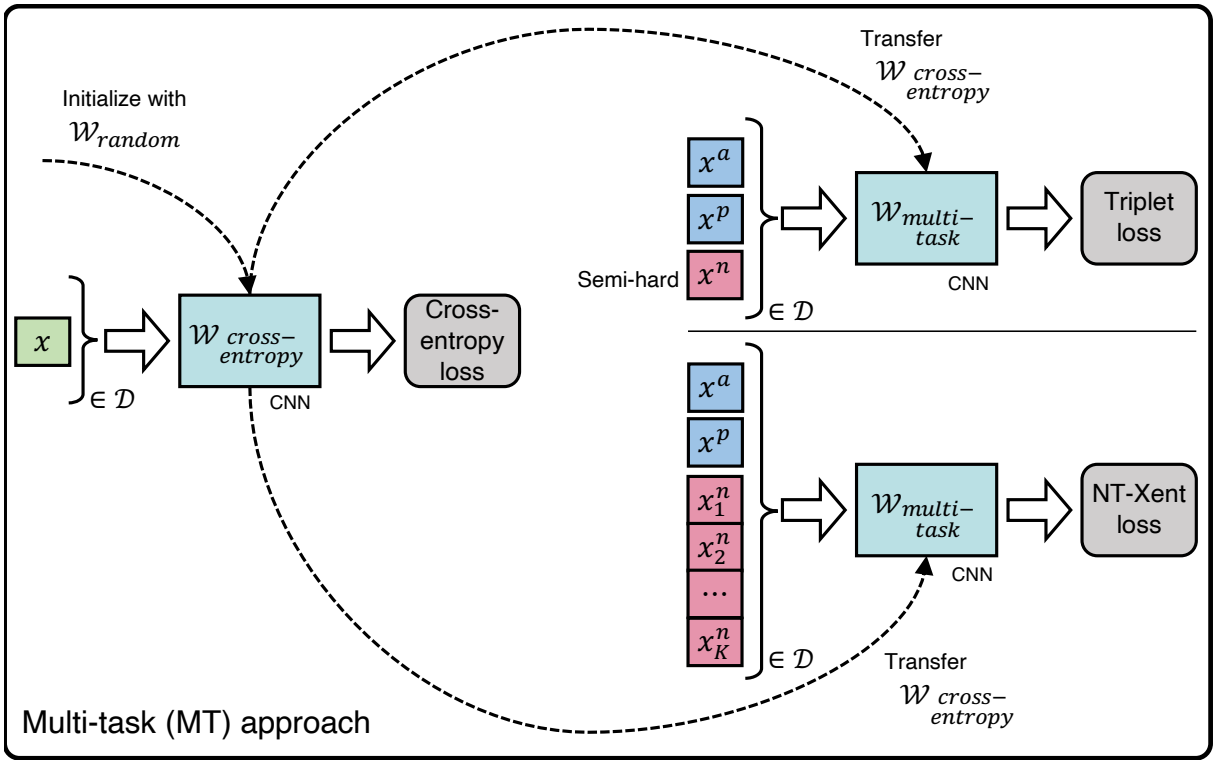
$$L_{NT\text{-}Xent} = \frac{1}{N}\sum_{i,j}^{N} -\log \frac{\exp(sim(f(x_i^a), f(x_j^p))/\tau)}{\exp(sim(f(x_i^a), f(x_j^p))/\tau) + \sum_{k=1}^{K}\exp(sim(f(x_i^a), f(x_k^n))/\tau)} \tag{3.3}$$

where, the sum $\sum_{k=1}^{K}$ is over all K negative examples in the training set for a given anchor example x_i^a; $sim(v, w) = \frac{v \cdot w}{\|v\|_2 \|w\|_2}$ is the cosine similarity between two feature vectors; and τ is a temperature hyper-parameter. The use of L_2 normalization (applied in the cosine similarity) with an appropriately adjusted τ temperature hyper-parameter weighs different examples and makes possible the model to learn from hard negative pair examples (JAISWAL et al., 2021; CHEN et al., 2020).

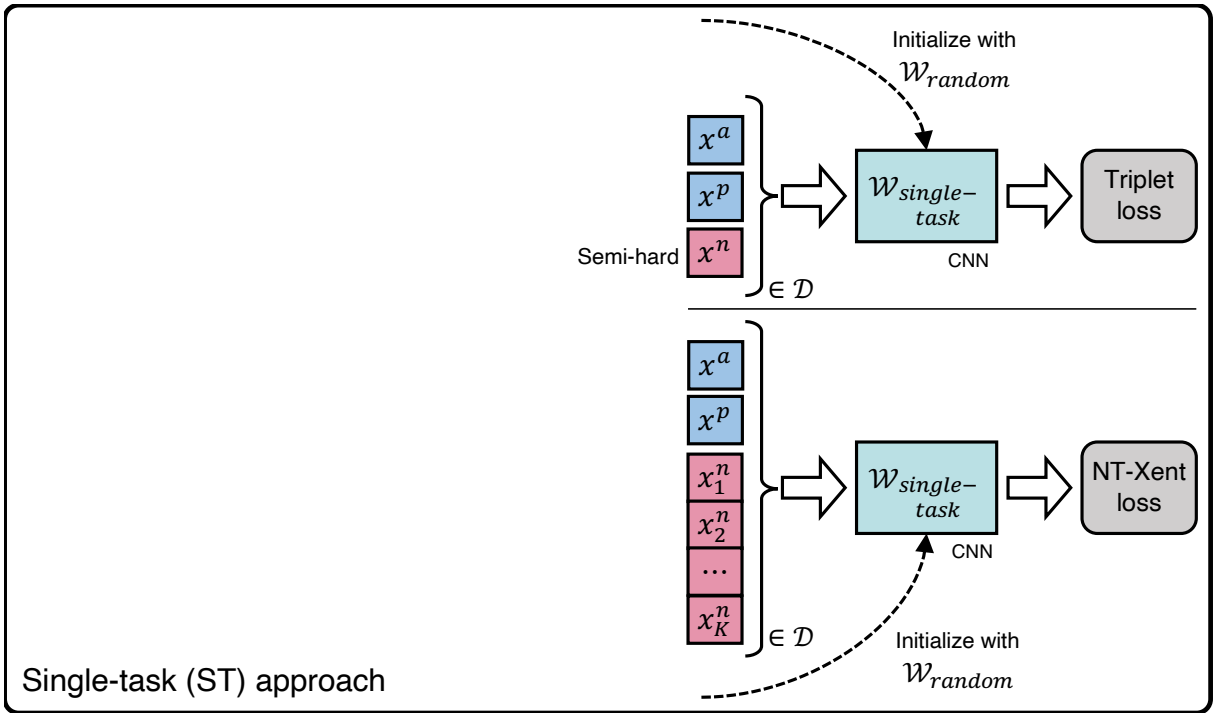
3.2.4 Training Process with the Proposed Framework

In the training process, we firstly optimize randomly initialized weights $\mathcal{W}_{random}$ of a convolutional neural network in terms of the cross-entropy loss using randomly obtained batches of data from the development set $\mathcal{D}$, resulting in the weights $\mathcal{W}_{cross-entropy}$ at the end of this first training stage. Thereafter, in a second training stage, the weights $\mathcal{W}_{cross-entropy}$ are optimized in terms of a contrastive loss (the NT-Xent loss or the Triplet loss with semi-hard negative mining) using balanced batches of data from the same development set $\mathcal{D}$, resulting in a second convolutional neural network with weights $\mathcal{W}_{multi-task}$. Therefore, the second task of the proposed framework is accomplished by transferring a preceding learned feature space by adjusting it with contrastive losses. For the sake of simplicity, in this work, we call these

models as *multi-task* (MT) contrastive models, since they are trained in a two-task approach. This approach is illustrated in Figure 7a.



(a) Multi-task contrastive training process.



(b) Single-task contrastive training process.

Figure 7 – Contrastive training approaches investigated in this work.

In the experiments, the proposed framework is compared to the situation in which contrastive losses are directly applied in a single-stage training approach. In this case, randomly

initialized weights $\mathcal{W}_{random}$ of a convolutional neural network are optimized in terms of the considered contrastive losses using balanced batches of data from the development set $\mathcal{D}$, resulting in the weights $\mathcal{W}_{single-task}$. For the sake of simplicity, in this work we call these models as *single-task* (ST) contrastive models. This approach is illustrated in Figure 7b. It is also important to note that $\mathcal{W}_{multi-task} \neq \mathcal{W}_{single-task}$.

3.3 RESEARCH QUESTIONS

The main objective of this chapter is to investigate whether adopting the proposed multi-task framework outperforms the situations in which contrastive losses can be directly applied in a single-task approach. Specifically, we investigate the obtained effect on model generalization when adopting the proposed framework for training models. Besides, we investigate whether obtained models with the proposed framework can significantly improve the signature verification in comparison to the state-of-the-art *SigNet* model. Yet, to our knowledge, a contrastive framework for representation learning based on hard negative mining of handwritten signatures has not been investigated in the context of handwritten signature verification problem. Therefore, the following Research Questions (RQs) are investigated in this chapter:

RQ1) How do the contrastive losses adopted in the proposed framework affect signature verification and model generalization?

RQ2) Does the proposed multi-task approach for learning representations provide signature verification improvement in comparison to a contrastive single-task approach?

RQ3) Do the proposed multi-task contrastive models provide significantly better feature representations for handwritten signatures than SigNet?

RQ4) Does NT-Xent loss (with implicit hard negative mining) provide better feature representations than Triplet loss (with explicit hard negative mining) for handwritten signature verification?

3.4 EXPERIMENTS

3.4.1 Experimental Setup

In this set of experiments, we adopt the GPDSsynthetic dataset (FERRER et al., 2017) for training and testing models. The GPDSsynthetic is a dataset containing synthetic genuine signatures and skilled forgeries generated with the objective of having similar characteristics to real signatures. Formerly, partitions of the GPDS Signature 960 dataset were extensively used for training and testing deep learning models in previous related works (HAFEMANN; SABOURIN; OLIVEIRA, 2017a; HAFEMANN; OLIVEIRA; SABOURIN, 2018; MARUYAMA et al., 2021; SOUZA et al., 2020; TSOUROUNIS et al., 2022), but GPDS Signature 960 dataset is no longer publicly available due to The General Data Protection Regulation (EU) 2016/679 and new investigators starting research in this field do not have access to this dataset. Thus, we expect to obtain an approximation of the performance on real signature data when training and testing models with the GPDSsynthetic dataset. Furthermore, the GPDSsynthetic dataset has a large number of users, which is an important requirement because it increases the likelihood of obtaining hard negative pairs during training.

It is worth mentioning that we performed experiments with the SigNet model for comparison purposes. The SigNet model was trained by Hafemann, Sabourin e Oliveira (2017a) with signature examples obtained from the GPDS Signature 960 dataset. In addition, we also trained a model following the same neural network architecture optimized with the cross-entropy loss as proposed by Hafemann, Sabourin e Oliveira (2017a), but using genuine signature examples obtained from the GPDSsynthetic dataset. In this work, this second model is called *SigNet (Synthetic)*.

3.4.1.1 Data Segmentation

In this set of experiments, we carried out investigation using the GPDSsynthetic (FERRER et al., 2017), MCYT-75 (ORTEGA-GARCIA et al., 2003) and CEDAR (KALERA; SRIHARI; XU, 2004) datasets. The number of users and the number of available genuine signature and skilled forgery examples per user in each dataset is detailed in Table 6.

We followed the GPDSsynthetic dataset segmentation proposed by Zheng et al. (2021) for training deep learning models. It is important to note that this segmentation allows a well

Dataset Name	Users	Genuine signatures	Forgeries
CEDAR	55	24	24
MCYT-75	75	15	15
GPDSsynthetic	10000	24	30

Table 6 – Summary of the datasets used in this set of experiments.

definition of disjoint sets for training, validation, and testing of models, allowing us to compare the error rates among related works, as it is guaranteed that the same subset of users is being verified. Given this, we split the GPDSsynthetic dataset into disjoint subsets of users: the development set $\mathcal{D}$, the validation set $\mathcal{V}_v$, and the exploitation sets $\mathcal{E}$ with different range of users (summarized in Table 7).

Subset Name	#Users	User Range
Development set $\mathcal{D}$	2000	5001 - 7000
Validation set $\mathcal{V}_v$	50	9951 - 10000
Exploitation set $\mathcal{E}_{150S}$ (GPDS-150S)	150	1 - 150
Exploitation set $\mathcal{E}_{300S}$ (GPDS-300S)	300	1 - 300

Table 7 – Segmentation of the GPDSsynthetic dataset following (ZHENG et al., 2021).

The signatures of users in the development set $\mathcal{D}$ are employed for training models. The validation set $\mathcal{V}_v$ is used for hyper-parameter search and validation of models, comprising signatures of the last 50 writers of the GPDSsynthetic dataset. Finally, different exploitation sets $\mathcal{E}$ are employed for testing models with different number of users. In this work, we tested models with the first 150 users (called as GPDS-150S) and with the first 300 users (called as GPDS-300S) of the GPDSsynthetic dataset. Besides, all the users of the MCYT-75 and CEDAR datasets are used to measure the generalization performance of features learned with the GPDSsynthetic development set. This way, two synthetic signature datasets (GPDS150S, GPDS150S) and two real signature datasets (MCYT-75, CEDAR) are evaluated in our experiments, balancing the types of datasets used for testing.

3.4.1.2 Experimental Protocol for Training Models

The signatures of the development set $\mathcal{D}$ are employed during model optimization, whereas $\mathcal{V}_v$ signatures are employed for evaluating the performance of trained models with different hyper-parameter configurations in order to obtain unbiased choices of best hyper-parameters. The $\mathcal{D}$ set has 24 genuine signatures per user. In training, we used 90% of these genuine

signatures to optimize the model during training, whereas we employed 10% of these genuine signatures for monitoring the early stopping of the training process up to a maximum of 60 epochs. It is important to note that skilled forgeries are not used for training.

We conducted model optimization as the same way as defined in Hafemann, Sabourin e Oliveira (2017a). For all contrastive losses studied in this work, those are minimized using Stochastic Gradient Descent with Nesterov Momentum with momentum factor given by 0.9. We trained models with an initial learning rate of 10^{-3} which is divided by 10 every 20 epochs. The adopted architecture of the deep convolutional neural network model is the same as defined in Hafemann, Sabourin e Oliveira (2017a) (described in Table 1), in such a way that the dimensionality of obtained signature feature representations is 2048.

In general, balanced batches of data are acquired (SCHROFF; KALENICHENKO; PHILBIN, 2015; WU et al., 2017) when deep learning models are optimized with contrastive losses, aiming to ensure that the number of positive and negative pairs are balanced for each class. Therefore, in the experiments, we sampled balanced batches of data during training when the optimization is done with contrastive losses. In this case, the number of signatures of each user is the same in any batch. Aiming to increase the diversity of samples from different users, we allocated 2 signature samples per user. The adopted batch size is 256, thus, 128 different users are randomly allocated per batch. In order to enable more pairwise combinations between signatures of different users, each signature example of the dataset is randomly allocated twice for two different batches.

3.4.2 Sensitivity Analysis

3.4.2.1 Experimental Protocol for the Validation Phase

The triplet loss with semi-hard negative mining is dependent on a margin hyper-parameter m (as defined in Equation 3.1). The margin hyper-parameter is usually set as $m = 0.2$ (SCHROFF; KALENICHENKO; PHILBIN, 2015; WU et al., 2017). However, margin values around $m = 0.2$ are investigated in this work. Thus, we trained models with $m = \{0.1, 0.2, 0.4, 0.8\}$, and signatures of $\mathcal{V}_v$ are verified in writer-dependent and writer-independent ways using feature representations obtained from these models. The NT-Xent loss is dependent on a temperature hyper-parameter τ (as defined in Equation 3.3). The temperature hyper-parameter is set as $\tau = 0.07$ in He et al. (2020) for an image classification problem. However, smaller hyper-

parameters around $\tau = 0.07$ as well as a hyper-parameter much greater than $\tau = 0.07$ are investigated in this work. Thus, we trained models with $\tau = \{0.01, 0.05, 0.07, 0.5\}$, and signatures of $\mathcal{V}_v$ are verified in writer-dependent and writer-independent ways using feature representations obtained from these models.

For writer-dependent verification, we trained a support vector machine classifier for each user of the validation set $\mathcal{V}_v$, following the protocol defined by Hafemann, Sabourin e Oliveira (2017a). To this end, 12 genuine reference signatures from the user of the validation set $\mathcal{V}_v$ are used as positive examples, whereas 14 randomly obtained signatures from each user of the $\mathcal{D}$ set are used as negative examples for training. We tested each writer-dependent classifier using 10 remaining genuine signatures and 10 skilled forgeries from the respective user in the validation set $\mathcal{V}_v$.

For writer-independent verification, we trained a support vector machine classifier with a set of dissimilarity vectors obtained from the signatures of all users in the $\mathcal{D}$ set, following the protocol defined by Souza, Oliveira e Sabourin (2018). Hence, for each user of the $\mathcal{D}$ set, we randomly selected 12 genuine signatures, which are pairwise combined, obtaining vectors of *within class*. Besides that, for each user of the $\mathcal{D}$ set, 11 genuine signatures are combined against 6 randomly selected genuine signatures from other users (random forgeries) in the $\mathcal{D}$ set. These are the dissimilarity vectors of *between class*. We tested the writer-independent classifier in face of each user of the validation set $\mathcal{V}_v$. In this case, for each user of the $\mathcal{V}_v$ set, we used 12 genuine signatures as reference signatures within the *max* fusion function, and the classifier is tested using 10 remaining genuine signatures and 10 skilled forgeries of the user.

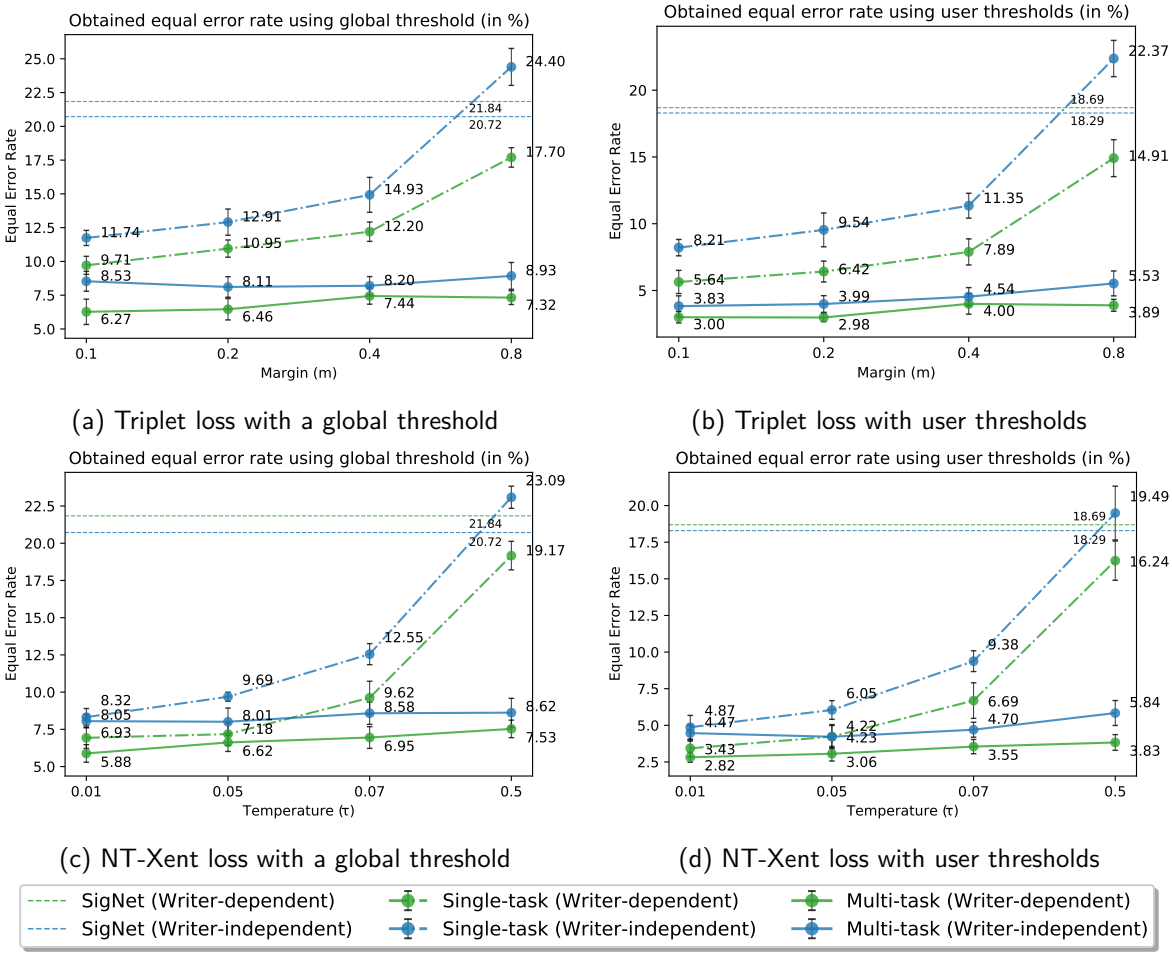
In our experiments, training of writer-independent classifiers has not converged in feasible time specifically with feature representations obtained through SigNet model to the GPDSsynthetic dataset examples. Thus, specifically in this situation, we decreased the number of sampled dissimilarity vectors from the development set $\mathcal{D}$ in order to obtain the performance of the classifier in a feasible time. In this case, for each user of the $\mathcal{D}$ set, we randomly selected 6 genuine signatures, which are pairwise combined, obtaining vectors of *within class*. Besides that, for each user of the $\mathcal{D}$ set, 5 genuine signatures are combined against 3 randomly selected genuine signatures from other users (random forgeries) in the $\mathcal{D}$ set, obtaining vectors of *between* class.

Finally, it is important to highlight that all comparisons between models performed in the experiments with the validation set $\mathcal{V}_v$ were done using Wilcoxon paired signed-rank tests with a 5% level of significance.

3.4.2.2 *Analyzing Contrastive Losses Varying the Hyper-parameters*

In this validation experiment, we analyze the obtained effect on equal error rate when varying the margin hyper-parameter in the Triplet loss and the temperature hyper-parameter in the NT-Xent loss. The obtained equal error rates when using global and user thresholds for each evaluated model configuration are shown in Figure 8.

As can be observed in the experimental results of validation experiments with the Triplet loss (Figures 8a and 8b), the representation with margin $m = 0.1$ presented lower error rates with a statistical difference to all the other margin configurations in the most of situations. In the specific situations where the average error adopting $m = 0.1$ is slightly higher than to the configuration with $m = 0.2$, these two configurations are statistically equivalent. Given



(a) Triplet loss with a global threshold

(b) Triplet loss with user thresholds

(c) NT-Xent loss with a global threshold

(d) NT-Xent loss with user thresholds

Firstly, we varied the margin hyper-parameter m of the Triplet loss, using a global threshold in 8a and using user thresholds in 8b. Secondly, we varied the temperature hyper-parameter τ of the NT-Xent loss, using a global threshold in 8c and using user thresholds in 8d.

Figure 8 – Performance of writer-dependent and writer-independent classifiers on the validation set $\mathcal{V}_v$ in terms of the equal error rate (errors and standard deviations in %).

this, adopting the margin hyper-parameter as $m = 0.1$ indicates the best hyper-parameter configuration for the Triplet loss in the studied problem. Note that a higher margin hyper-parameter value ($m = 0.8$) provided worse feature representations.

Besides, as can be observed in the experimental results of validation experiments with the NT-Xent loss (Figures 8c and 8d), the representation with $\tau = 0.01$ presented lower error rates with a statistical difference to all the other temperature configurations in the most of situations. In the specific situations where the average error adopting $\tau = 0.01$ is slightly higher than to the configuration with $\tau = 0.05$, these two configurations are statistically equivalent. Given this, adopting the temperature hyper-parameter as $\tau = 0.01$ indicates the best hyper-parameter configuration for the NT-Xent loss in the studied problem. Note that a higher temperature hyper-parameter value ($\tau = 0.5$) provided worse feature representations.

Finally, it is worth mentioning that the best-obtained configurations with the proposed framework adopting the Triplet loss with $m = 0.1$ and the NT-Xent loss with $\tau = 0.01$ are statistically equivalent in writer-dependent as well as in writer-independent verification with the validation set. This result indicates that despite being contrastive losses with different mechanisms for mining hard negative examples, these losses perform in a comparable way.

3.4.2.3 Understanding Obtained Representations with Contrastive Losses

In the experimental results, obtaining a lower FRR (False Rejection Rate) implies in better classification of genuine signatures with well-separated dense clusters of genuine signatures representations, which indicates reaching the Properties 1 and 2 (as previously defined in Section 3.1). Furthermore, obtaining a lower $FAR_{skilled}$ (False acceptance rate considering skilled forgeries) implies in better classification of skilled forgeries when moving away representations of skilled forgeries from the respective genuine signature representations, which indicates reaching the Property 3 (as previously defined in Section 3.1).

As can be seen from the results of such metrics (listed in Table 8): increasing the hyper-parameter values of the contrastive losses in the single-task approach increases both the $FAR_{skilled}$ and the FRR error rates. Increasing the margin and the temperature hyper-parameters worsens both the verification of genuine signatures and skilled forgeries as these representations are being moved in the space, which causes an overlap of these representations. However, skilled forgeries are better separated from the genuine signatures with this move, as the $FAR_{skilled}$ error rate is generally lower than the FRR error rate. Despite this

being a beneficial effect, moving skilled forgery representations tends to generate a side effect that increases the intra-class distance of genuine signatures. Hyper-parameter values that decrease these two metrics should be found for the problem. In the studied validation set, smaller hyper-parameters values such as $m = 0.1$ and $\tau = 0.01$ do it, providing lower equal error rates.

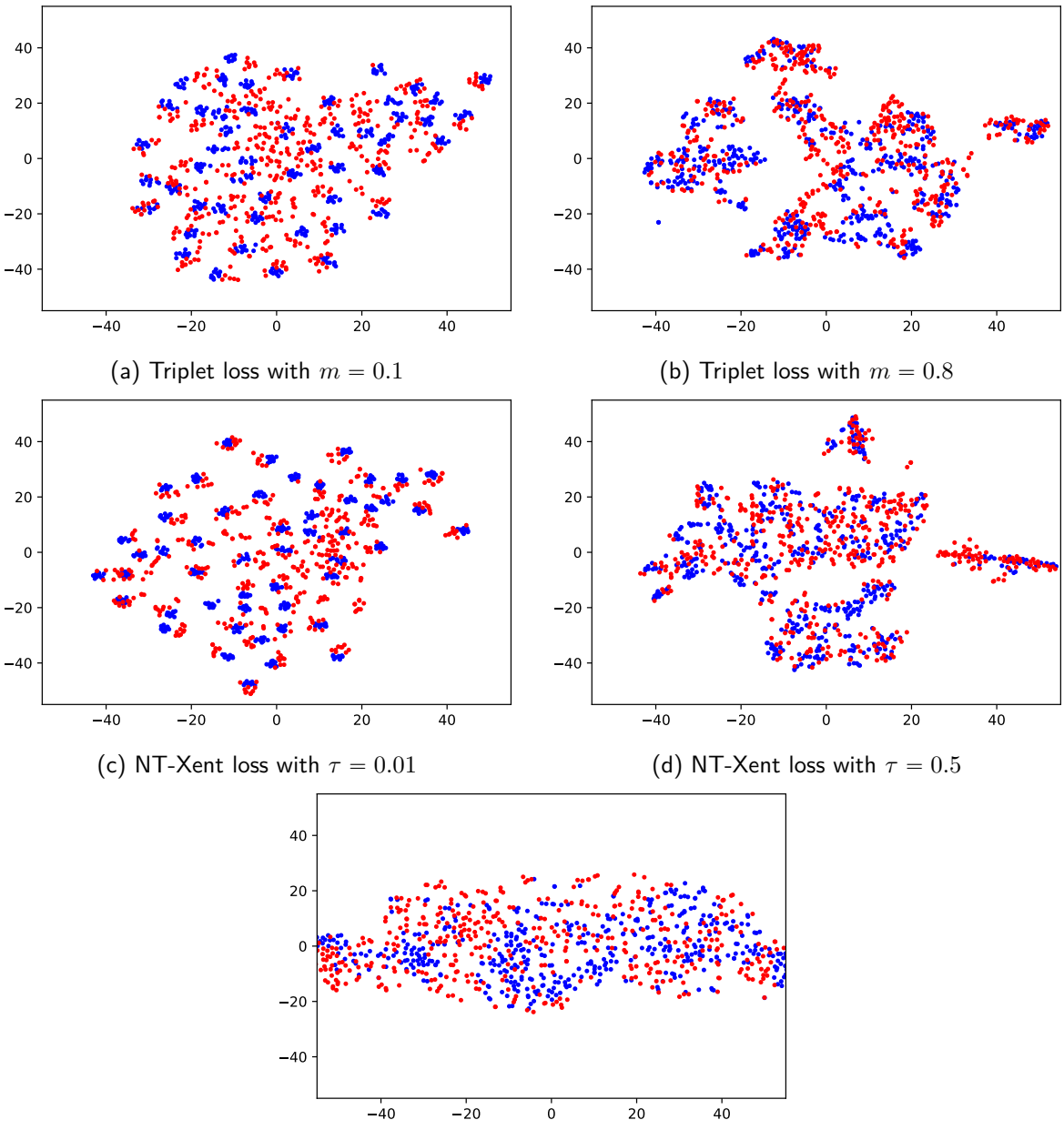
Table 8 – Obtained metrics for feature representations extracted from different single-task contrastive models adopting different loss configurations for the 50 users of the validation set $\mathcal{V}_v$.

These metrics were obtained in the feature space using writer-dependent classifiers. The FRR and $FAR_{skilled}$ metrics are measured considering the actual distance to the decision hyper-plane, then, the adopted threshold for these metrics is zero. The $*$ means that obtained experimental result is significantly different when compared to the same loss configuration but with smaller hyper-parameter value. Wilcoxon paired signed-rank tests were performed with 5% significance level.

	FRR	$FAR_{skilled}$	$EER_{globalthreshold}$	$EER_{userthresholds}$
Triplet loss $m = 0.1$ (single-task)	19.88 ±(1.04)	4.76 ±(0.73)	9.71 ±(0.67)	5.64 ±(0.87)
Triplet loss $m = 0.8$ (single-task)	26.90 ±(1.96)$*$	11.56 ±(1.22)$*$	17.70 ±(0.72)$*$	14.91 ±(1.38)$*$
NT-Xent loss $\tau = 0.01$ (single-task)	8.52 ±(1.09)	5.86 ±(0.88)	6.93 ±(0.71)	3.43 ±(0.57)
NT-Xent loss $\tau = 0.5$ (single-task)	27.94 ±(2.02)$*$	12.62 ±(1.96)$*$	19.17 ±(0.96)$*$	16.24 ±(1.34)$*$

When the margin m and temperature τ hyper-parameters are higher in the studied contrastive losses, these hyper-parameters have an influence in adjusting genuine signatures and skilled forgery representations. Higher hyper-parameter values increase the distance between representations of similar examples of different classes, which can be seen as an effect associated with moving the skilled forgery representations from the genuine ones during training. But on the other hand, as a side effect, higher hyper-parameters can increase the distance between representations of similar examples of the same class, which implies that genuine representations of different users overlap more in the feature space. This generates not-well separated signature clusters and decreases the ability to verify genuine signatures.

Figure 9 shows obtained t-SNE (MAATEN; HINTON, 2008) projections for feature representations of genuine signatures (in blue) and skilled forgeries (in red) for the users of the validation set $\mathcal{V}_v$. When adopting a small margin $m = 0.1$ (in Figure 9a) or a small temperature $\tau = 0.01$ (in Figure 9c) hyper-parameters, each user has genuine signatures projected in dense clusters that are in separated regions of the space. Namely, in this case genuine signatures have a more uniform distribution within the space. On the other hand, when adopting a large margin $m = 0.8$ (in Figure 9b) or a large temperature $\tau = 0.5$ (in Figure 9d) hyper-parameters, the skilled forgeries representations are moved away from the genuine signature representations within the feature space. However, in this situation, genuine signature clusters are not strictly dense and consequently are more scattered within the space in such way that genuine clusters



(a) Triplet loss with $m = 0.1$

(b) Triplet loss with $m = 0.8$

(c) NT-Xent loss with $\tau = 0.01$

(d) NT-Xent loss with $\tau = 0.5$

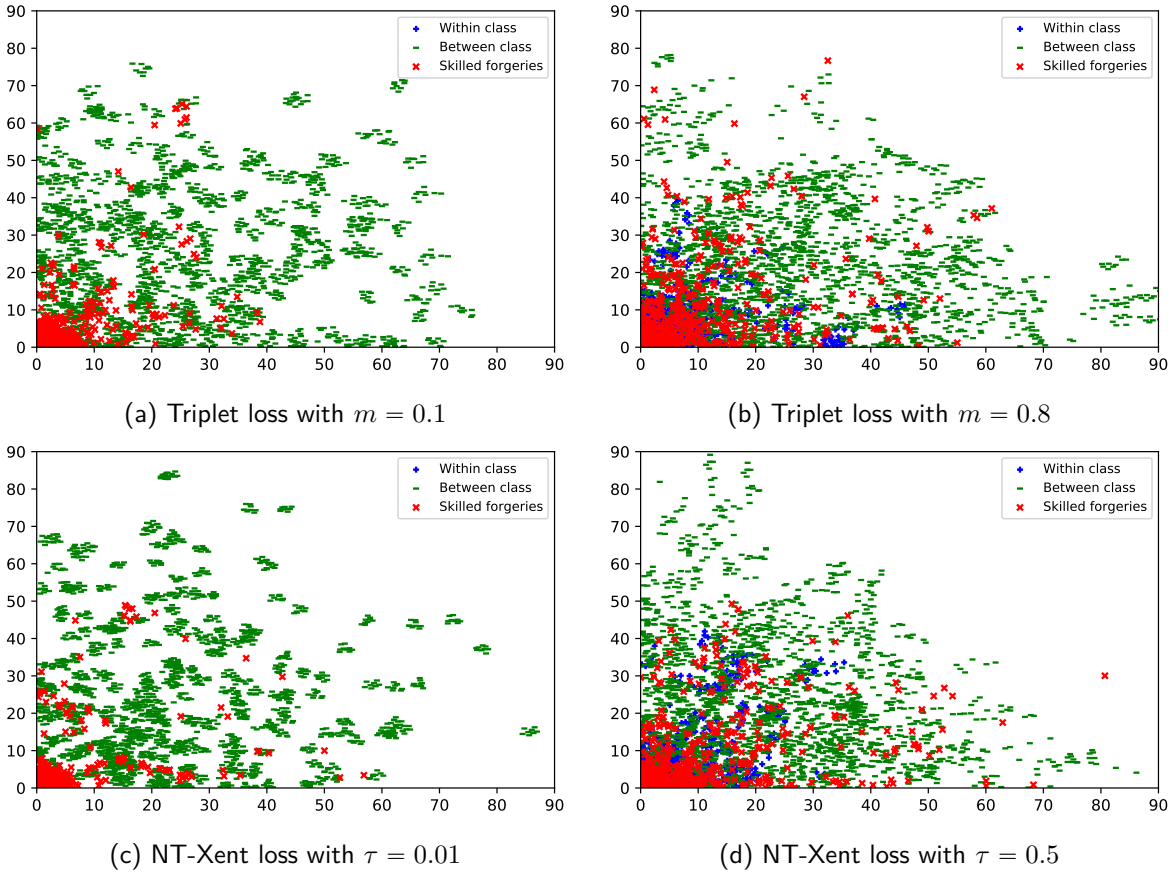
(e) SigNet

These features were extracted from different models adopting different configurations: in 9a and 9b we adopted the Triplet loss with two different margin hyper-parameter configurations; and in 9c and 9d we adopted the NT-Xent loss with two different temperature hyper-parameter configurations. Finally, features extracted from the *SigNet* model are shown in 9e.

Figure 9 – Obtained t-SNE projections for features representations of genuine signatures (in blue) and skilled forgeries (in red) for the 50 users of the validation set $\mathcal{V}_v$.

of different users can overlap each other.

The effect of moving skilled forgery representations is evidenced by transforming them into a dissimilarity space. The representations in Figure 9 were transformed through a dichotomy transformation, and obtained visualizations are shown in Figure 10. As can be observed in Figures 10a and 10c, when lower hyper-parameter values ($m = 0.1$ and $\tau = 0.01$) are adopted



(a) Triplet loss with $m = 0.1$

(b) Triplet loss with $m = 0.8$

(c) NT-Xent loss with $\tau = 0.01$

(d) NT-Xent loss with $\tau = 0.5$

We extracted dissimilarity vectors from different models adopting different configurations: in 10a and 10b we adopted triplet loss with two different margin hyper-parameter configurations; and in 10c and 10d we adopted NT-Xent loss with two different temperature hyper-parameter configurations.

Figure 10 – Obtained visual projections in the dissimilarity space for: the *between* class vectors (in green), the *within* class vectors (in blue), and skilled forgeries compared to one reference signature (in red). Dissimilarity vectors were obtained from signatures of the 50 users of the validation set $\mathcal{V}_v$.

in the studied contrastive losses, the *within* and *between* classes are better separated in the dissimilarity space. Due to the skilled forgeries tend to be similar to genuine signatures, a considerable amount of skilled forgery dissimilarity vectors are positioned in the within class region. On the other hand, when higher hyper-parameter values are adopted ($m = 0.8$ and $\tau = 0.5$ in Figures 10b and 10d), the dissimilarity vectors of skilled forgeries are moved further away from the within class region. However, as a side effect, the within and between class regions overlap more in the dissimilarity space.

Finally, as can be observed in Figure 9e, it is worth mentioning that the SigNet model does not provide good representations for the GPDSsynthetic samples as there is no tendency to represent genuine signatures in clusters adopting this model. Hence, this explains the obtained high error rates (reported in Figure 8) when using SigNet model for feature extraction in writer-dependent and writer-independent verification of GPDSsynthetic samples.

3.4.2.4 Understanding the Influence of the Proposed Framework

This experiment aims to understand the influence of the proposed framework (defined in Section 3.2) in the obtained feature representations for the validation set $\mathcal{V}_v$. To this end, contrastive models trained by directly applying contrastive losses in a single-task approach are compared to the models trained in a multi-task approach adopting the proposed framework. In these experiments, we defined the hyper-parameters values of the contrastive losses according to the best values found for the problem (as previously discussed in Section 3.4.2.2).

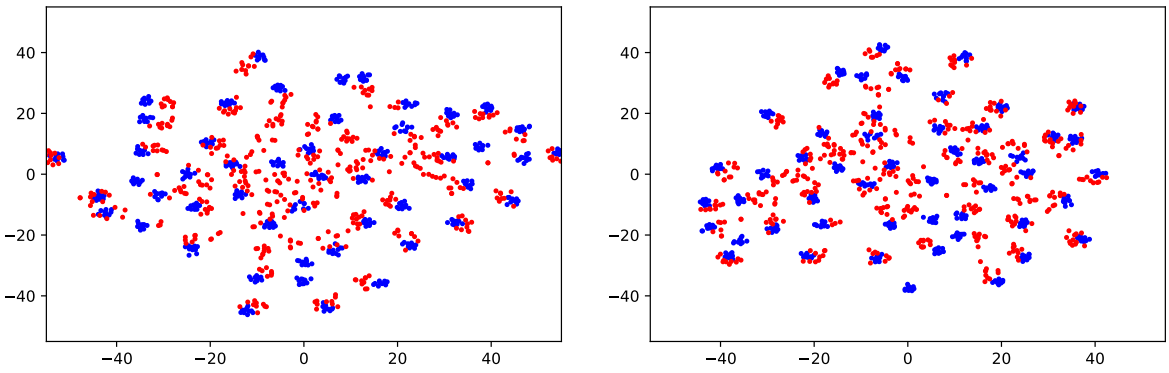
Table 9 – Obtained metrics for feature representations extracted from different contrastive models adopting the multi-task and single-task approaches for the 50 users of the validation set $\mathcal{V}_v$.

These metrics were obtained in the feature space using writer-dependent classifiers. The FRR and $FAR_{skilled}$ metrics are measured considering the actual distance to the decision hyper-plane, then, the adopted threshold for these metrics is zero. The $*$ means that obtained experimental result is significantly different when compared to the same configuration but following the proposed multi-task framework. Wilcoxon paired signed-rank tests were performed with 5% significance level.

	FRR	$FAR_{skilled}$	$EER_{globalthreshold}$	$EER_{userthresholds}$
Triplet loss $m = 0.1$ (multi-task)	7.38 $\pm$(1.12)	5.30 $\pm$(0.63)	6.27 $\pm$(0.94)	3.00 $\pm$(0.43)
Triplet loss $m = 0.1$ (single-task)	19.88 $\pm$(1.04)$*$	4.76 $\pm$(0.73)	9.71 $\pm$(0.67)$*$	5.64 $\pm$(0.87)$*$
NT-Xent loss $\tau = 0.01$ (multi-task)	7.98 $\pm$(0.71)	4.84 $\pm$(0.79)	5.88 $\pm$(0.59)	2.82 $\pm$(0.34)
NT-Xent loss $\tau = 0.01$ (single-task)	8.52 $\pm$(1.09)	5.86 $\pm$(0.88)$*$	6.93 $\pm$(0.71)$*$	3.43 $\pm$(0.57)$*$

As can be seen from the results of the metrics listed in Table 9: single-task contrastive models have worsened the ability to classify genuine signatures. In contrast, multi-task contrastive models can better classify genuine signatures while still maintaining the ability to verify skilled forgeries, which lowers the equal error rates. This occurs because a prior training step with cross-entropy loss implies in a denser and more uniform distribution of genuine clusters into the representation space, controlling the side effects later introduced by using the contrastive losses in the second task of the framework. The t-SNE projections for feature representations obtained from multi-task contrastive models are shown in Figure 11. As can be observed in this figure, dense clusters (in Figures 11a and 11b) obtained from the multi-task contrastive models are more dispersed and uniformly distributed into the space when compared to the respective single-task model adopting the same contrastive loss and same hyper-parameter configuration (visualizations previously shown in Figures 9a and 9c).

Besides, we collected the number of semi-hard triplets applied for training in each epoch with the Triplet loss (namely, the number of semi-hard triplets that explicitly satisfy the constraint defined in Equation 3.2). As can be observed when comparing the Figures 12a and 12b, training the contrastive model from a representation space initially optimized through

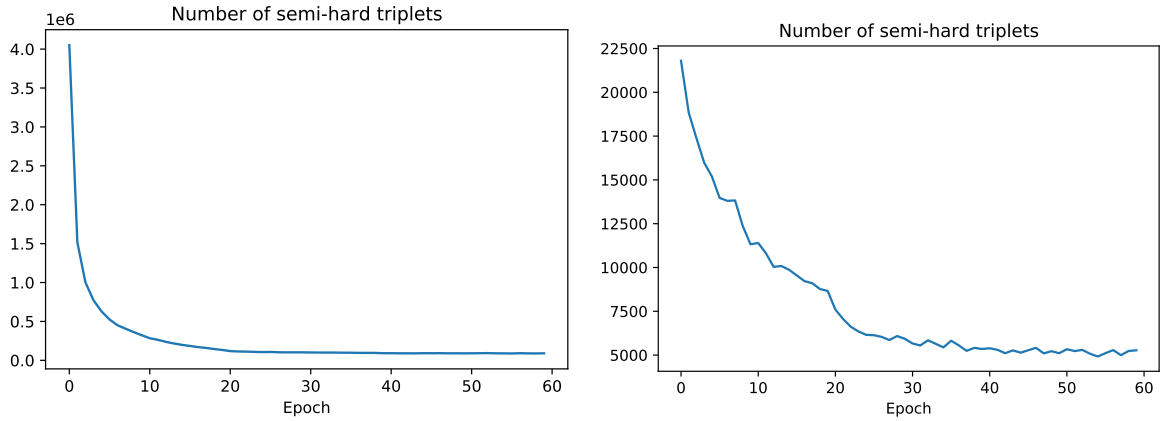


(a) Triplet loss with $m = 0.1$ (multi-task) (b) NT-Xent loss with $\tau = 0.01$ (multi-task)

The triplet loss is optimized with margin hyper-parameter given by $m = 0.1$ in a multi-task approach in 11a. The NT-Xent loss is optimized with temperature hyper-parameter given by $\tau = 0.01$ in a multi-task approach in 11b.

Figure 11 – Obtained t-SNE projections for features representations of genuine signatures (in blue) and skilled forgeries (in red) for the 50 users of the validation set $\mathcal{V}_v$. We extracted these features from different models adopting the proposed multi-task framework.

cross-entropy loss with the multi-task approach diminishes the chances of generating semi-hard triplets. It happens because the clusters of genuine signatures for each user are already well defined in the representation space at the beginning of the second step of the training process. Thus, such triplets satisfy the constraint of being semi-hard but from a situation that makes this restriction more difficult to be satisfied. Adopting these triplets for optimization seems strong enough for adjusting the model in a second-step training refinement. Moreover, a more restricted choice for sampling triplets seems to better control the overfitting of the obtained model.



(a) Triplet loss with $m = 0.1$ (single-task) (b) Triplet loss with $m = 0.1$ (multi-task)

The vertical axis represents the number of mined semi-hard triplets. In the Figure 12a this number is described in million.

Figure 12 – The number of mined semi-hard triplets in each epoch of the training process considering the single-task and the multi-task approaches.

3.4.2.5 Findings of the Sensitivity Analysis

In this section, the findings of the performed analysis are summarized: **RQ1) How do the contrastive losses adopted in the proposed framework affect signature verification and model generalization?**

The tuning of hyper-parameters of the contrastive losses allows adjusting the trade-off between verifying genuine signatures and skilled forgeries. The Triplet and NT-Xent losses have the property of moving genuine signature and skilled forgery representations within the feature and the dissimilarity spaces. Specifically, in the Triplet loss, increasing the margin means including more semi-hard negative examples during the optimization, which moves the representations of skilled forgeries, but on the other hand, it introduces a side-effect that worsens the verification of genuine signatures. Such an effect also happens with higher temperature values in the NT-Xent loss. We encountered hyper-parameter values of the Triplet and NT-Xent losses that balance this trade-off.

Finally, we identified that applying contrastive losses in a pre-organized space generates a refinement of this original representation space. In this way, the proposed multi-task framework maintains the same generalization ability provided by the cross-entropy loss for the studied problem. A better control of the intra-cluster density and more uniform dispersion of clusters in the representation space provided by the proposed multi-task approach can help in the generalization of the models to subset of users with different distributions as well as to other datasets.

3.4.3 Measuring Generalization Performance on Other Datasets

In this section, we apply the feature space learned in the GPDSsynthetic development set $\mathcal{D}$ for extracting features on the GPDS-150S, GPDS-300S, CEDAR and MCYT-75 datasets as a transfer learning approach. The single-task and multi-task contrastive models optimized through Triplet loss and NT-Xent loss with the best hyper-parameter configurations obtained in the validation set are applied for feature extraction. We used these features for training writer-dependent and writer-independent classifiers, which had their performance measured.

From this point on, we refer to the evaluated models by their respective names. *SigNet* is the pre-trained model provided by Hafemann, Sabourin e Oliveira (2017a). Furthermore, we trained a model using the same cross-entropy optimized architecture as defined by Hafe-

mann, Sabourin e Oliveira (2017a) but using genuine signature samples obtained from the GPDSsynthetic development set. This second model is called *SigNet (Synthetic)*. Regarding the contrastive models, we evaluated the best single-task and multi-task models for each one of the contrastive losses studied in this work. Thus, the models optimized with the Triplet loss are called *ST-SigNet (Triplet)* and *MT-SigNet (Triplet)* for the single-task and multi-task versions, respectively. Finally, the models optimized with the NT-Xent loss are called *ST-SigNet (NT-Xent)* and *MT-SigNet (NT-Xent)* for the single-task and multi-task versions, respectively.

3.4.3.1 Experimental Protocol for Writer-Dependent Verification

In order to measure the generalization performance of models in a writer-dependent approach, we followed a signature segmentation method based on those proposed in Hafemann, Sabourin e Oliveira (2017a), Zheng et al. (2021), Tsourounis et al. (2022): for each user, we built a training set consisting of user's genuine signatures as positive examples and genuine signatures of other users as negative examples. By utilizing this training set, we trained a support vector machine for each user. For the GPDS-150S, GPDS-300S, CEDAR and MCYT-75 datasets, we used r genuine reference signatures of a given user from the exploitation set as positive examples, whereas we obtained r genuine signatures from the other users in the same exploitation set as negative examples. These negative examples can be seen as random forgeries with regard to the considered user. The number r of genuine reference signatures and the number of random forgeries used in training for each dataset is listed in Table 10.

Table 10 – Separation into training and testing sets in Writer-Dependent approach for each evaluated dataset.

Dataset		Training set		Testing set
Name	#Users	Genuine	Random Forgeries	
CEDAR	55	$r \in \{1, 2, 3, 5, 10, 12\}$	$r \times 54$	10 genuine, 10 skilled
MCYT-75	75	$r \in \{1, 2, 3, 5, 10\}$	$r \times 74$	5 genuine, 15 skilled
GPDS-150S	150	$r \in \{1, 2, 3, 5, 10, 12\}$	$r \times 149$	10 genuine, 10 skilled
GPDS-300S	300	$r \in \{1, 2, 3, 5, 10, 12\}$	$r \times 299$	10 genuine, 10 skilled

After training a writer-dependent classifier, we measured the performance over a testing set. To this end, a disjoint sample of remaining genuine signatures and skilled forgeries of each user from the exploitation set are used for testing. We verified these signatures and evaluation metrics (listed in Section 2.2.4) were applied considering them. The number of

genuine signatures and the number of skilled forgeries employed in testing for each dataset is listed in Table 10.

3.4.3.2 Experimental Protocol for Writer-Independent Verification

In order to measure the generalization performance of models in a writer-independent approach with the GPDS-150S and GPDS-300S datasets, we followed the signature segmentation proposed in Souza, Oliveira e Sabourin (2018). Writer-independent classifiers were trained with a set of dissimilarity vectors obtained from signatures of the development set $\mathcal{D}$. For each user, we randomly selected a subset of 12 genuine signatures. All genuine signatures of each subset were pairwise combined, obtaining 132,000 dissimilarity vectors of *within class*. Besides that, for each user, 11 genuine signatures were combined against 6 random forgeries. In this case, random forgeries consist of randomly selected genuine signatures from other users. In total, we obtained 132,000 dissimilarity vectors of *between class*. As the number of vectors of *within class* and *between class* is the same, then, dataset used for training is balanced.

In the case of measuring the generalization performance of models in a writer-independent approach with the CEDAR and MCYT-75 datasets, we followed the signature segmentation proposed in Souza et al. (2020). We evaluated writer-independent classifiers through a cross-validation procedure. In this scenario, half of the users of the dataset are used as development set, whereas the other half are used as exploitation set. In the experiments, we performed a random split of the dataset in a $5 \times 2-$fold cross-validation. Obtained metrics are an average of measuring the performance in each of these splits. The number of examples used for training of writer-independent classifiers for each dataset is summarized in Table 11.

For the CEDAR dataset, signatures of 27 or 28 users were employed for training writer-independent classifiers. For each user, we randomly selected a subset of 14 genuine signatures. All genuine signatures of each subset were pairwise combined, obtaining 2,457 dissimilarity vectors of *within class*. Besides that, for each one of these 27 or 28 users, 13 genuine signatures were combined against 7 random forgeries. In total, we obtained 2,457 dissimilarity vectors of *between class*. For the MCYT-75 dataset, signatures of 37 or 38 users were employed for training writer-independent classifiers. For each user, we randomly selected a subset of 10 genuine signatures. All genuine signatures of each subset were pairwise combined, obtaining 1,665 dissimilarity vectors of *within class*. Besides that, for each one of these 37 or 38 users, 9 genuine signatures were combined against 5 random forgeries. In total, we obtained 1,665

Table 11 – Separation into training and testing sets in Writer-Independent approach for each evaluated dataset.

Dataset		Training set		Testing set	
Name	#Users	Negative (*between*) class	Positive (*within*) class	Reference set	Questioned set
CEDAR	55	Distances between the 13 signatures for each user and 7 random signatures from other users (2,457 samples).	Distances between the 14 signatures for each user (2,457 samples).	$r \in \{1, 2, 3, 5, 10, 12\}$	10 genuine, 10 skilled
MCYT-75	75	Distances between the 9 signatures for each user and 5 random signatures from other users (1,665 samples).	Distances between the 10 signatures for each user (1,665 samples).	$r \in \{1, 2, 3, 5, 10\}$	5 genuine, 15 skilled
GPDS-150S	150	Distances between the 11 signatures for each user and 6 random signatures from other users (132,000 samples).	Distances between the 12 signatures for each user (132,000 samples).	$r \in \{1, 2, 3, 5, 10, 12\}$	10 genuine, 10 skilled
GPDS-300S	300	Distances between the 11 signatures for each user and 6 random signatures from other users (132,000 samples).	Distances between the 12 signatures for each user (132,000 samples).	$r \in \{1, 2, 3, 5, 10, 12\}$	10 genuine, 10 skilled

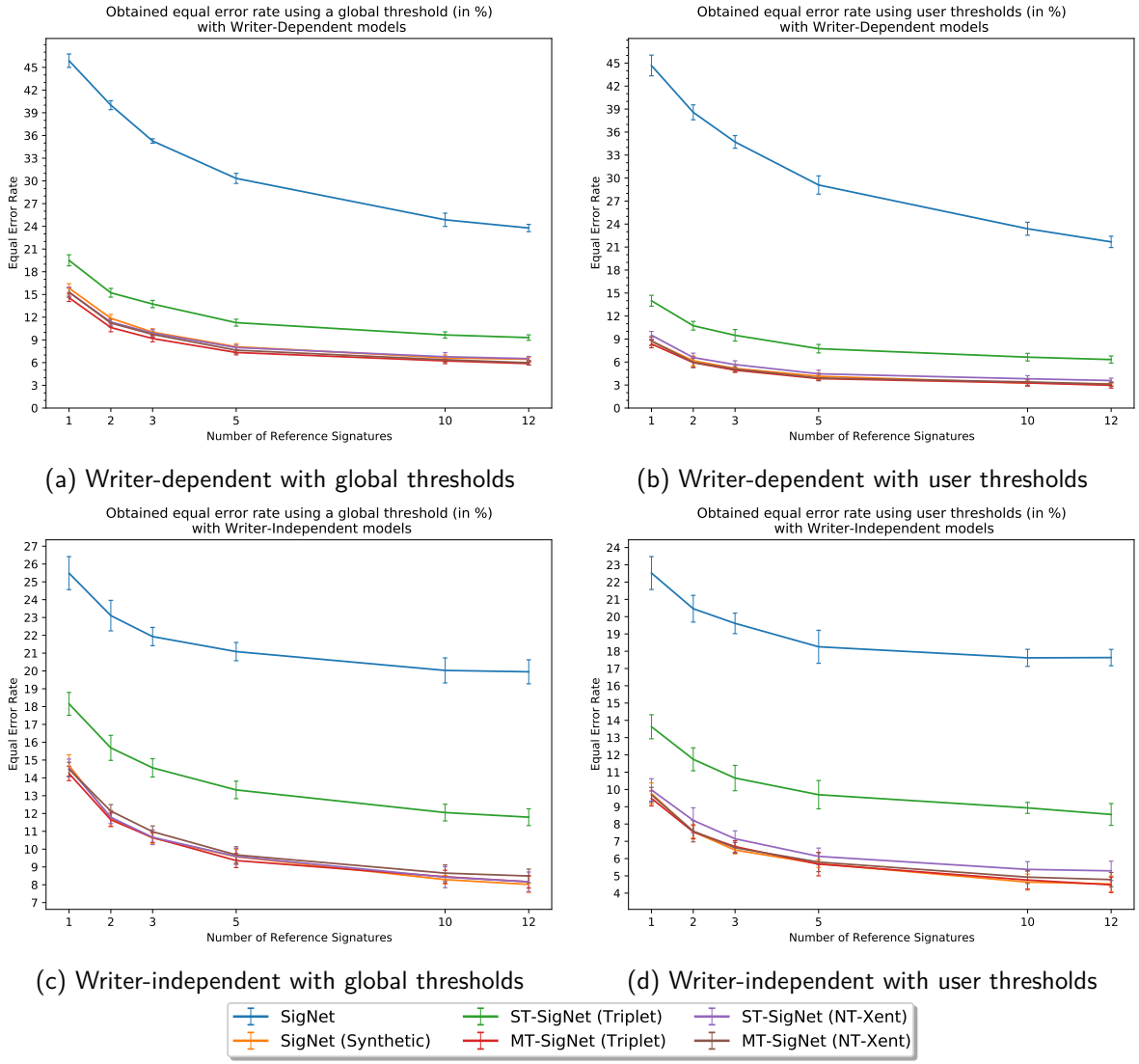
dissimilarity vectors of *between class*. The number of vectors of *within class* and *between class* is the same. Given this, the dataset used for training classifiers in the cross-validation procedure is balanced.

With respect to the evaluation of writer-independent classifiers with the GPDS-150S and GPDS-300S datasets, we respectively employed signatures of 150 and 300 users in the exploitation set $\mathcal{E}$ for testing models. In the case of the CEDAR and MCYT-75 datasets, signatures of the remaining half of users (not used in training) in the cross-validation procedure were employed as exploitation set for testing models. For each user, we randomly selected r genuine reference signatures. We employed these reference signatures to perform verification within fusion functions. Besides that, for each user, we applied the remaining genuine signatures and randomly selected skilled forgeries for verification. These questioned set of signatures were verified and evaluation metrics (listed in Section 2.2.4) were applied considering them. In the experiments, the number r of reference signatures is varied for each dataset and is listed in Table 11.

3.4.3.3 Experimental Results and Discussion

Figures 13, 14, 15 and 16 respectively show the average performance obtained in terms of equal error rate on GPDS-150S, GPDS-300S, CEDAR and MCYT-75 datasets.

As can be seen in the Figures 13 and 14, the performance of the contrastive single-task models is worse than contrastive multi-task models in the face of the GPDS-150S and GPDS-300S datasets in most situations. Besides, in the case of CEDAR and MCTY-75 datasets



(a) Writer-dependent with global thresholds

(b) Writer-dependent with user thresholds

(c) Writer-independent with global thresholds

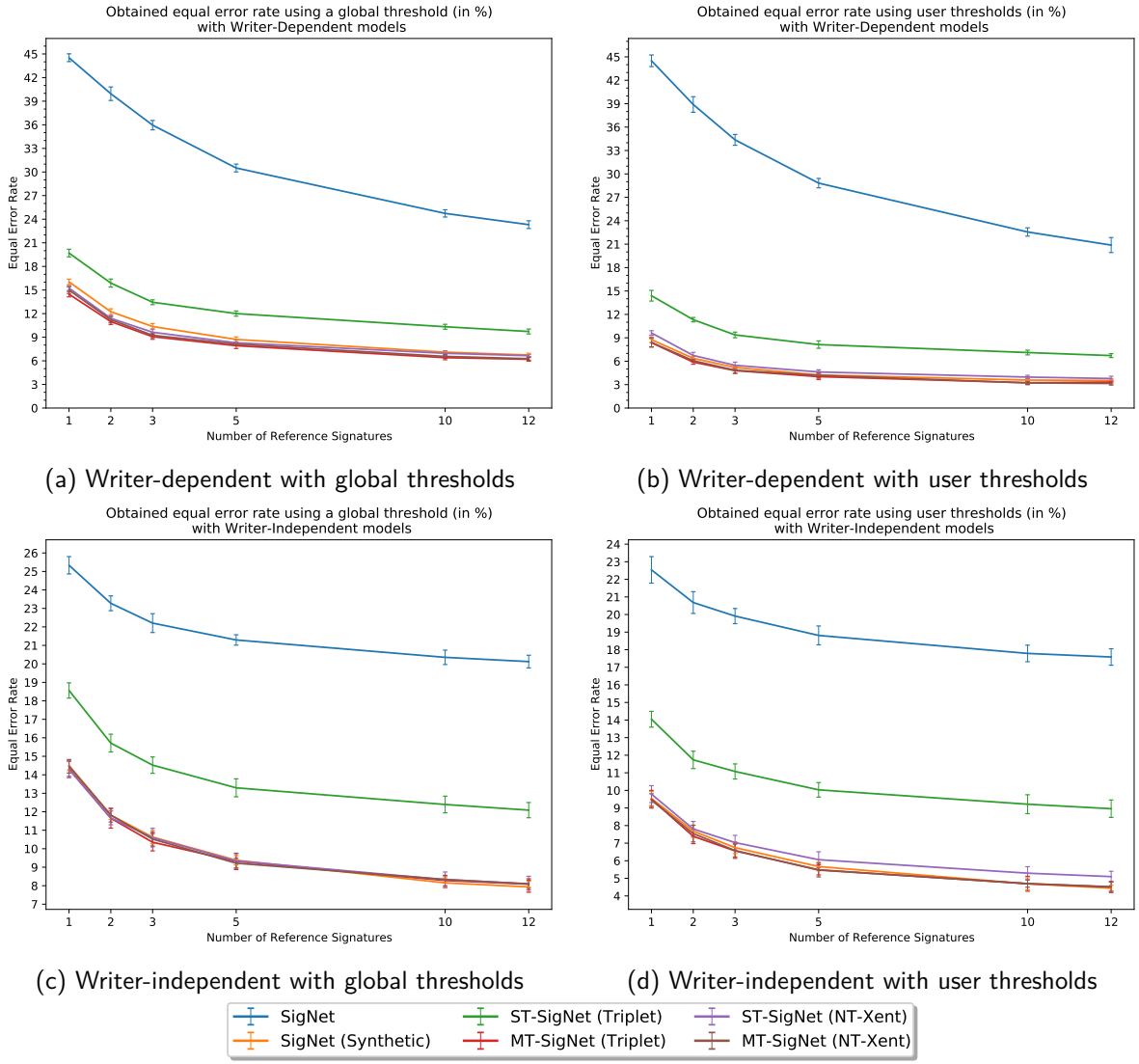
(d) Writer-independent with user thresholds

Figure 13 – Average performance of Writer-Dependent and Writer-Independent classifiers for the GPDS-150S dataset, as the number of reference genuine signatures (per user) available for training is varied.

(Figures 15 and 16), the performance of single-task contrastive models is clearly worse than that obtained through multi-task contrastive models. Thus, an improvement on verifying signatures of these datasets is obtained when using multi-task contrastive models following the proposed framework.

RQ2) Does the proposed multi-task approach for learning feature representations provide signature verification improvement in comparison to a contrastive single-task approach? This obtained effect suggests that single-task contrastive models over-adjust to specific characteristics of signatures belonging to the training dataset. Therefore, the proposed multi-task framework generates models with a better generalization ability when compared to applying contrastive methods directly in training following a single-task approach.

Furthermore, it is possible to observe in Figures 13, 14, 15 and 16 that there is no difference



(a) Writer-dependent with global thresholds

(b) Writer-dependent with user thresholds

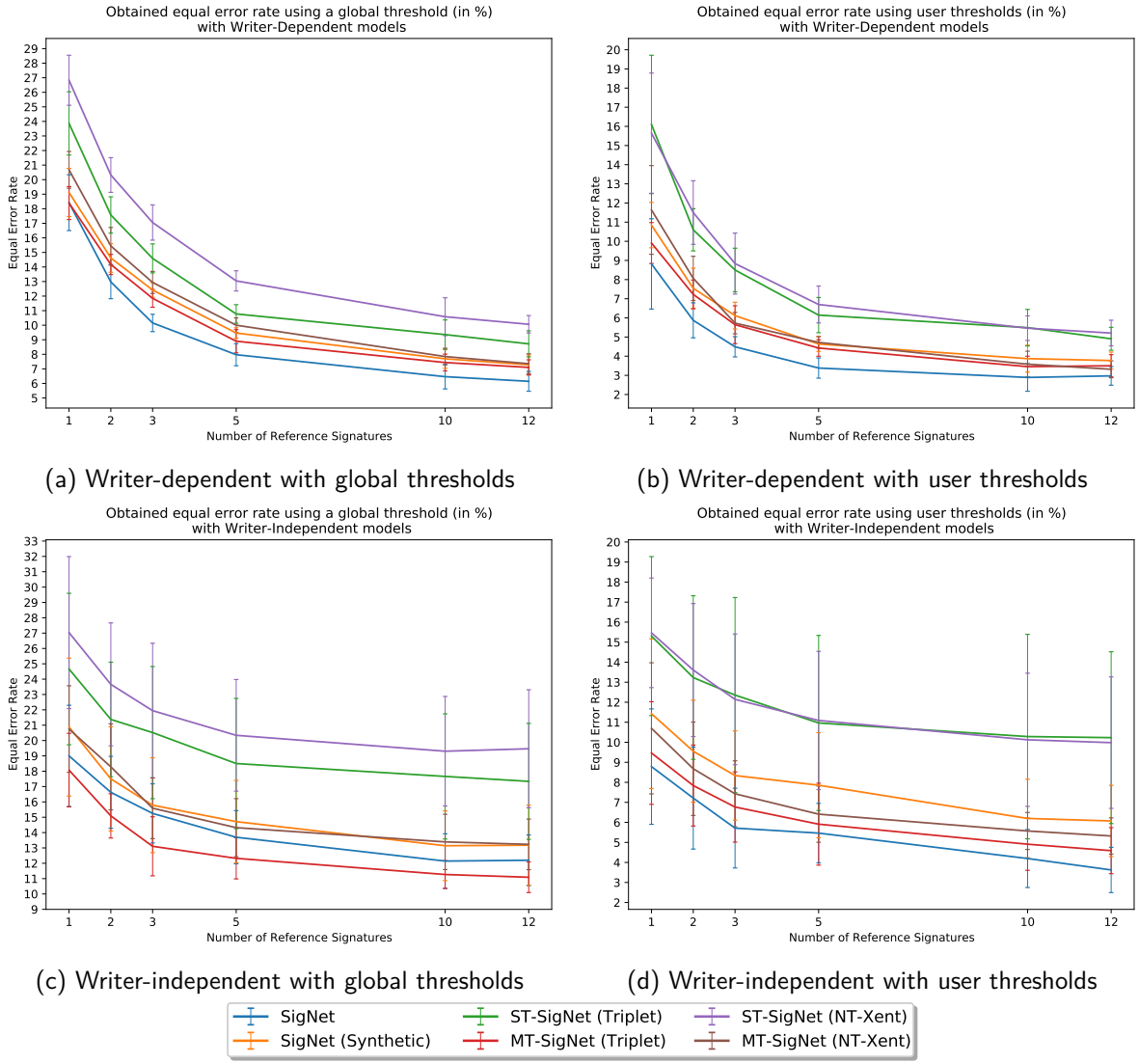
(c) Writer-independent with global thresholds

(d) Writer-independent with user thresholds

Figure 14 – Average performance of Writer-Dependent and Writer-Independent classifiers for the GPDS-300S dataset, as the number of reference genuine signatures (per user) available for training is varied.

in signature verification performance behavior among models with respect to the number of reference signatures (in writer-dependent or writer-independent verification). With contrastive single-task and multi-task models or using SigNet, when the number of reference signatures increases, the verification performance improves in all evaluated datasets.

Another important observation from the experiments is that the best writer-independent performance of models is generally inferior to the writer-dependent performance. Writer-independent classifiers are trained with a disjoint set of users different from those applied for testing. Therefore, the writer-independent model must adapt to a broader range of users even being a single model, leading to slightly higher error rates. Nevertheless, the writer-independent classifier is more scalable because it can be applied to verify signatures from a greater diversity of users without fine-tuning the model for each user.



(a) Writer-dependent with global thresholds

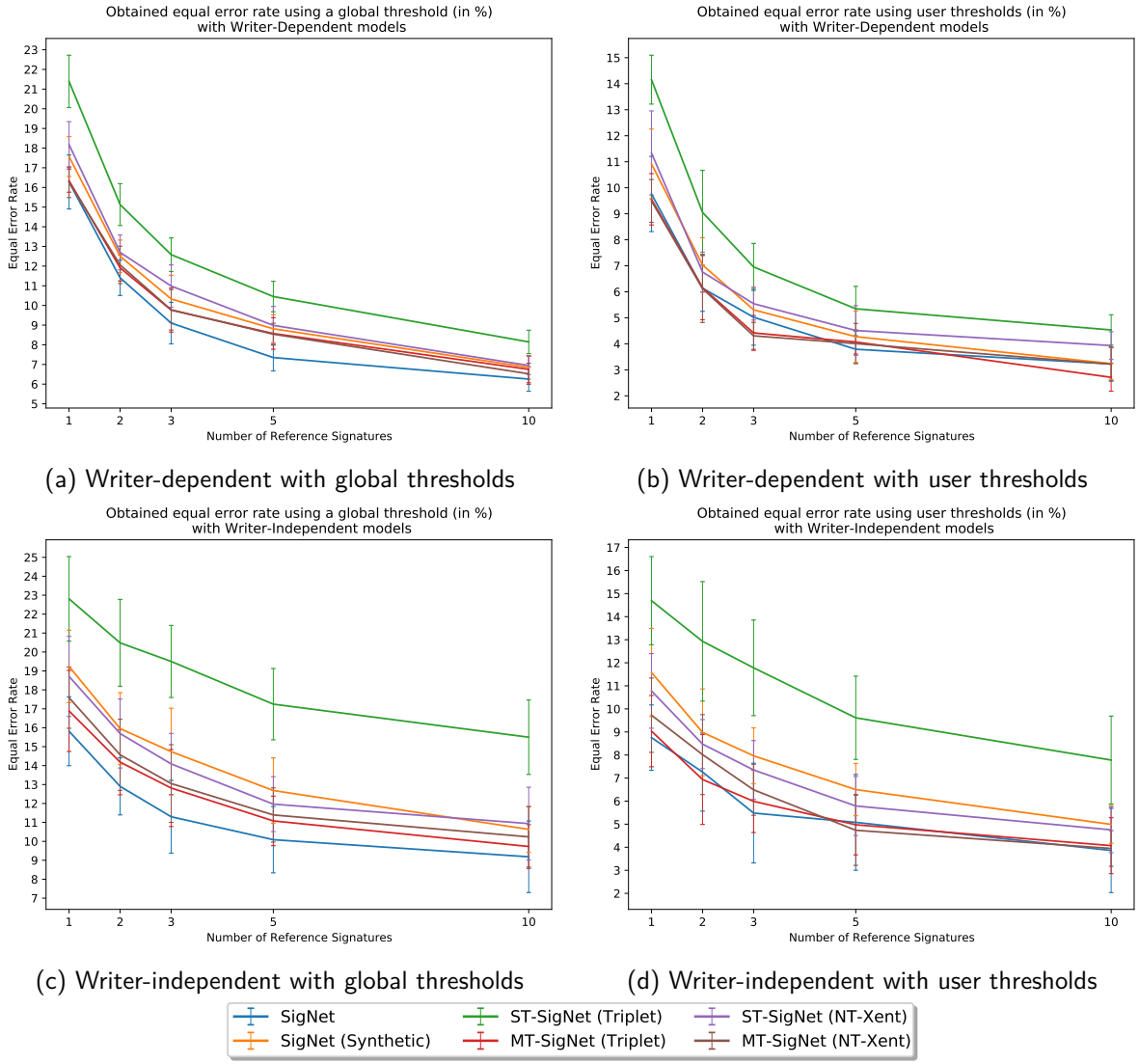
(b) Writer-dependent with user thresholds

(c) Writer-independent with global thresholds

(d) Writer-independent with user thresholds

Figure 15 – Average performance of Writer-Dependent and Writer-Independent classifiers for the CEDAR dataset, as the number of reference genuine signatures (per user) available for training is varied.

Besides, an interesting finding based on the analysis of the experimental results reported in Figures 13, 14, 15 and 16 is that the SigNet (trained with GPDS960Gray by Hafemann, Sabourin e Oliveira (2017a)) performed better than SigNet (Synthetic) (trained with GPDSsynthetic) on real signature data. However, SigNet (Synthetic) is still competitive in the majority of cases. It is explained by the fact that SigNet learned the intrinsic characteristics of genuine signature images more than SigNet (Synthetic), where this knowledge is not perfectly embedded in the synthetic signature images. On the other side, SigNet does not perform well on synthetic signature data compared to the SigNet (Synthetic). This analysis demonstrated that the distributions of the GPDS960Gray and GPDSsynthetic are different. However, learning from synthetic signature images can help to cope with the unavailability of real signature data for training.



(a) Writer-dependent with global thresholds

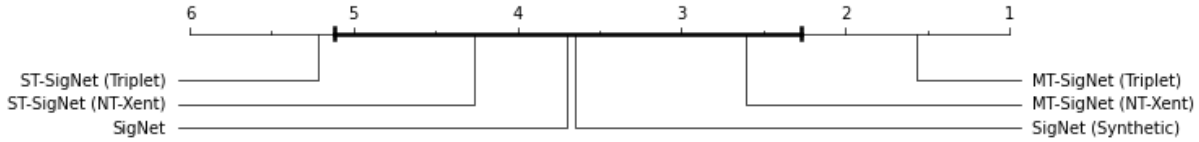
(b) Writer-dependent with user thresholds

(c) Writer-independent with global thresholds

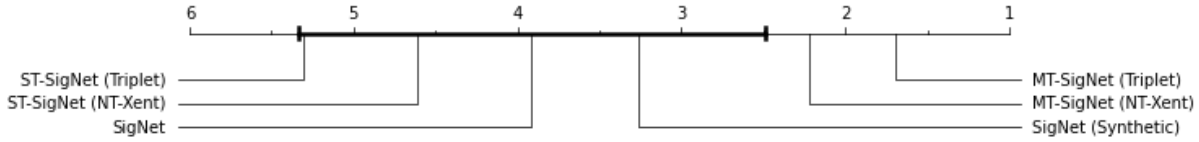
(d) Writer-independent with user thresholds

Figure 16 – Average performance of Writer-Dependent and Writer-Independent classifiers for the MCYT-75 dataset, as the number of reference genuine signatures (per user) available for training is varied.

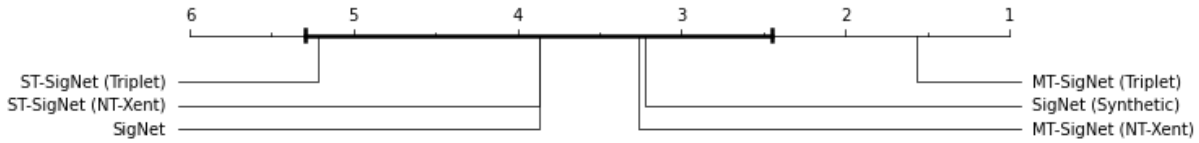
We performed Friedman tests in order to statistically compare the proposed models. The purpose of Friedman test is to rank the models in terms of obtained equal error rates in writer-dependent and writer-independent approaches considering global thresholds and user-specific thresholds. For the GPDS-150S, GPDS-300S, CEDAR and MCYT-75 datasets, we generated different signature segmentations with different number of reference signatures (according to the Tables 10 and 11), totaling twenty-three evaluated subjects. For each one of these subjects, different obtained feature representations (using the investigated models) are extracted in such a way that equal error rate is a repeated measure over each subject when varying each model. It is worth noting that we balanced the chosen datasets (when adopting two synthetic signature datasets and two real signature datasets), aiming to diminish the biases that one type of dataset would introduce in the performed statistical tests. All performed tests indicated



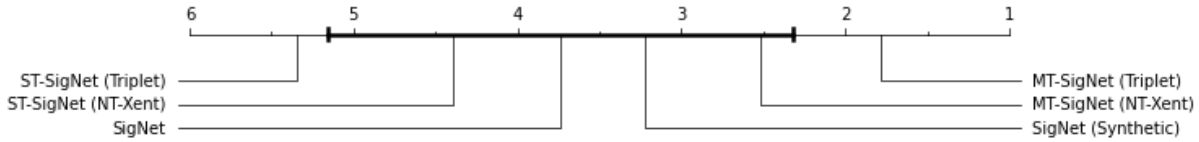
(a) Writer-dependent with global thresholds



(b) Writer-dependent with user thresholds



(c) Writer-independent with global thresholds



(d) Writer-independent with user thresholds

The GPDS-150S dataset provides synthetic signatures of 150 users, the GPDS-300S dataset provides synthetic signatures of 300 users, the MCYT-75 dataset provides real signatures of 75 users, and the CEDAR dataset provides real signatures of 55 users. All the signatures obtained from the two real signature datasets (MCYT-75 and CEDAR) and the two synthetic signature datasets (GPDS-150S and GPDS-300S) are applied together in the performed statistical tests. All models with ranks outside the marked interval are significantly different ($p < 0.05$) from the *SigNet* model.

Figure 17 – Comparison of the *SigNet* model against contrastive models with the Bonferroni-Dunn post hoc test. The models were tested using signature data obtained from the GPDS-150S, GPDS-300S, CEDAR and MCYT-75 datasets.

that all evaluated models are not equivalent (rejecting the null hypothesis) with 5% level of significance. Then, Bonferroni-Dunn post hoc tests were performed to compare the contrastive models against the *SigNet* as a baseline method. Critical difference diagrams (DEMsAR, 2006) are shown in Figure 17. In these diagrams, the *SigNet* model is compared against single-task and multi-task contrastive models. All classifiers with ranks outside the marked interval are significantly different ($p < 0.05$) from the *SigNet* model. Moreover, right-most placed classifiers provide better performance in terms of the equal error rate.

RQ3) Do the proposed multi-task contrastive models provide significantly better feature representations for handwritten signatures than SigNet? As can be observed

in Figure 17, single-task contrastive models provided worse performance than *SigNet*. In contrast, the multi-task contrastive models that adopt our proposed framework are better ranked in performance when compared to the *SigNet* model. Furthermore, when considering the multi-task model optimized with the Triplet loss, the experiments demonstrated a statistically significant improvement in signature verification error rates with different datasets when adopting the proposed multi-task contrastive framework. This result points towards a general guideline strategy when training contrastive models for extracting handwritten signature features. Adopting the proposed framework results in models that outperform the state-of-the-art *SigNet* (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) model, improving the signature verification task in writer-dependent and writer-independent approaches.

It is important to mention that SigNet tends to have a better performance specifically with the MCYT-75 and CEDAR datasets. This occurs because SigNet is a pre-trained model with data obtained from real users of the GPDS Signature 960 dataset, and the MCYT-75 and CEDAR datasets also contain real signature samples. In contrast, the models proposed in this work are trained with the GPDSsynthetic dataset, which includes synthetic signatures. Even with this, we identified situations in our experiments where the proposed multi-task models performed better than SigNet when tested with MCYT-75 and CEDAR datasets.

For comparison purposes, we also trained a model optimized with the cross-entropy (adopting the same architecture defined by Hafemann, Sabourin e Oliveira (2017a)) but using samples from the GPDSsynthetic dataset (labeled as *SigNet (Synthetic)* in Figure 17). It is observed that the SigNet (Synthetic) does not present a significant difference to the SigNet model (HAFEMANN; SABOURIN; OLIVEIRA, 2017a). In contrast, the MT-SigNet (Triplet) model, in addition to being the best-ranked model, presented a statistical difference in performance compared to the SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a). This result is due to the fact that the MT-SigNet (Triplet) model is only a little worse than SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) in the MCYT-75 and CEDAR datasets with real signatures, thus providing a better overall ranking of the MT-SigNet (Triplet) model when considering all the evaluated synthetic and real datasets. As a result, our contrastive multi-task models that adopt the proposed framework perform better than using cross-entropy loss, which experimentally indicates a signature verification improvement provided by the proposed framework.

RQ4) Does NT-Xent loss (with implicit hard negative mining) provide better feature representations than Triplet loss (with explicit hard negative mining) for handwritten signature verification? As also can be observed in Figure 17, in the writer-

dependent as well as writer-independent verification approaches (with global or user-specific thresholds), the Triplet loss multi-task optimized model presented statistically better performance than SigNet. However, the NT-Xent loss multi-task optimized model presented statistically better performance than SigNet only in writer-dependent approach with user thresholds, even though being better ranked than SigNet in all approaches. The better performance of the Triplet loss in verification is partially explained by the fact that the separation between skilled forgery and genuine signature representations seems to be stronger when adopting the explicit mechanism for mining hard negatives with the Triplet loss within the proposed framework. Note that the contrastive single-task model optimized with the Triplet loss is also the worst-ranked model, which suggests that the representation adjustments with the Triplet loss are more significant when compared to the NT-Xent loss.

3.4.3.4 Comparison with the State-of-the-art

In this section, the best-obtained results with the proposed multi-task framework are compared with the related state-of-the-art works in which deep learning, contrastive learning and hand-engineered methods are applied for offline handwritten signature verification. Among these works, we highlight Hafemann, Sabourin e Oliveira (2017a), Hafemann, Oliveira e Sabourin (2018), Souza, Oliveira e Sabourin (2018), Souza et al. (2020), which are previous works in which the SigNet model is trained and tested using data obtained from the GPDS960Gray dataset. In addition to these works, we highlight Zheng et al. (2021), a very recent work in which deep learning models are trained and tested using data obtained from the GPDSsynthetic dataset. It is important to note that we compare the performance of the methods with regard to different datasets with real and synthetic user data (respectively, the GPDS960Gray and the GPDSsynthetic) but keeping the same number of users. Therefore, the comparison presented in this section intends to place an overview of the performance of the existing methods in relation to the proposed framework.

Experimental results with the GPDS-150S and GPDS-300S datasets are respectively listed in Tables 12 and 13. The proposed framework for obtaining feature representations clearly outperforms very recent deep learning based methods for writer-dependent verification (ZHENG et al., 2021; YILMAZ; ÖZTÜRK, 2020), in which the models are tested using samples taken from the GPDSsynthetic dataset. The proposed framework also outperforms the contrastive method described in Soleimani, Araabi e Fouladi (2016) and hand-engineering methods pre-

Table 12 – Comparison with the state-of-the-art in GPDS-150S dataset (errors in %).

The *Exploitation $\mathcal{E}$ source* column indicates the type of the dataset in which the tested signature examples were obtained: from the GPDS960Gray (GPDS-150) in case of real signature data or from the GPDSsynthetic in case of synthetic signature data.

Reference	Exploitation $\mathcal{E}$ source	#Samples	Features	Type	EER
Hu e Chen (2013)	GPDS960Gray	10	LBP, HOG and GLCM	WD	7.66
Soleimani, Araabi e Fouladi (2016)	GPDSsynthetic	10	LBP + DMML	WD	12.67
Zheng et al. (2021)	GPDSsynthetic	5	Micro deformations	WD	6.87 ($\pm$ 0.35)
Zheng et al. (2021)	GPDSsynthetic	10	Micro deformations	WD	5.45 ($\pm$ 0.42)
Zheng et al. (2021)	GPDSsynthetic	12	Micro deformations	WD	4.82 ($\pm$ 0.38)
Hafemann, Sabourin e Oliveira (2017a) (our test)	GPDSsynthetic	5	SigNet	WD	29.10 ($\pm$ 1.19)
Hafemann, Sabourin e Oliveira (2017a) (our test)	GPDSsynthetic	10	SigNet	WD	23.39 ($\pm$ 0.85)
Hafemann, Sabourin e Oliveira (2017a) (our test)	GPDSsynthetic	12	SigNet	WD	21.68 ($\pm$ 0.74)
Present Work	GPDSsynthetic	5	SigNet (Synthetic)	WD	4.14 ($\pm$ 0.43)
Present Work	GPDSsynthetic	10	SigNet (Synthetic)	WD	3.26 ($\pm$ 0.42)
Present Work	GPDSsynthetic	12	SigNet (Synthetic)	WD	3.15 ($\pm$ 0.15)
Present Work	GPDSsynthetic	5	MT-SigNet (Triplet)	WD	3.84 ($\pm$ 0.23)
Present Work	GPDSsynthetic	10	MT-SigNet (Triplet)	WD	3.28 ($\pm$ 0.40)
Present Work	GPDSsynthetic	12	MT-SigNet (Triplet)	WD	2.96 ($\pm$ 0.37)
Present Work	GPDSsynthetic	5	MT-SigNet (NT-Xent)	WD	3.92 ($\pm$ 0.36)
Present Work	GPDSsynthetic	10	MT-SigNet (NT-Xent)	WD	3.41 ($\pm$ 0.42)
Present Work	GPDSsynthetic	12	MT-SigNet (NT-Xent)	WD	3.11 ($\pm$ 0.29)
Souza, Oliveira e Sabourin (2018) (our test)	GPDSsynthetic	5	SigNet	WI	18.26 ($\pm$ 0.96)
Souza, Oliveira e Sabourin (2018) (our test)	GPDSsynthetic	10	SigNet	WI	17.62 ($\pm$ 0.49)
Souza, Oliveira e Sabourin (2018) (our test)	GPDSsynthetic	12	SigNet	WI	17.63 ($\pm$ 0.47)
Present Work	GPDSsynthetic	5	SigNet (Synthetic)	WI	5.70 ($\pm$ 0.21)
Present Work	GPDSsynthetic	10	SigNet (Synthetic)	WI	4.64 ($\pm$ 0.46)
Present Work	GPDSsynthetic	12	SigNet (Synthetic)	WI	4.53 ($\pm$ 0.47)
Present Work	GPDSsynthetic	5	MT-SigNet (Triplet)	WI	5.68 ($\pm$ 0.68)
Present Work	GPDSsynthetic	10	MT-SigNet (Triplet)	WI	4.76 ($\pm$ 0.53)
Present Work	GPDSsynthetic	12	MT-SigNet (Triplet)	WI	4.48 ($\pm$ 0.45)
Present Work	GPDSsynthetic	5	MT-SigNet (NT-Xent)	WI	5.80 ($\pm$ 0.55)
Present Work	GPDSsynthetic	10	MT-SigNet (NT-Xent)	WI	4.93 ($\pm$ 0.36)
Present Work	GPDSsynthetic	12	MT-SigNet (NT-Xent)	WI	4.78 ($\pm$ 0.41)

sented in Zois, Alewijnse e Economou (2016), Serdouk, Nemmour e Chibani (2017), Hu e Chen (2013), Bhunia, Alaei e Roy (2019), Diaz et al. (2017), Hamadene e Chibani (2016). When the proposed framework is compared with models trained with the GPDS960Gray dataset, our new models performed at a comparable level to those reported in Hafemann, Sabourin e Oliveira (2017a) for writer-dependent verification. However, specifically in the case of writer-independent verification, our proposed models trained with the GPDSsynthetic dataset performed a little worse when compared with the experimental results reported by Souza, Oliveira e Sabourin (2018). The difference among distributions of the GPDSsynthetic users seems to be greater than among GPDS960Gray users, as the GPDSsynthetic dataset is much larger than the GPDS960Gray, making it more challenging to adapt writer-independent classifiers to a different set of users. Besides, methods based on the SigNet-F model (MARUYAMA et al.,

Table 13 – Comparison with the state-of-the-art in GPDS-300S dataset (errors in %).

The *Exploitation $\mathcal{E}$ source* column indicates the type of the dataset in which the tested signature examples were obtained: from the GPDS960Gray (GPDS-300) in case of real signature data or from the GPDSsynthetic in case of synthetic signature data.

Reference	Exploitation $\mathcal{E}$ source	#Samples	Features	Type	EER
Soleimani, Araabi e Fouladi (2016)	GPDS960Gray	10	LBP + DMML	WD	20.94
Zois, Alewijnse e Economou (2016)	GPDS960Gray	12	Poset-oriented grid	WD	3.24
Hamadene e Chibani (2016)	GPDS960Gray	5	Contourlet Transform	WI	18.42
Serdouk, Nemmour e Chibani (2017)	GPDS960Gray	10	HOT	WD	9.30
Diaz et al. (2017)	GPDS960Gray	8	Duplicator	WD	14.58
Hafemann, Sabourin e Oliveira (2017a)	GPDS960Gray	12	SigNet-F	WD	1.69 ($\pm$ 0.18)
Hafemann, Sabourin e Oliveira (2017a)	GPDS960Gray	5	SigNet	WD	3.92 ($\pm$ 0.18)
Hafemann, Sabourin e Oliveira (2017a)	GPDS960Gray	12	SigNet	WD	3.15 ($\pm$ 0.18)
Hafemann, Oliveira e Sabourin (2018)	GPDS960Gray	12	SigNet-SPP	WD	3.15 ($\pm$ 0.14)
Hafemann, Oliveira e Sabourin (2018)	GPDS960Gray	12	SigNet-SPP-F	WD	0.41 ($\pm$ 0.05)
Souza, Oliveira e Sabourin (2018)	GPDS960Gray	5	SigNet	WI	4.40 ($\pm$ 0.34)
Souza, Oliveira e Sabourin (2018)	GPDS960Gray	12	SigNet	WI	3.34 ($\pm$ 0.22)
Bhunia, Alaei e Roy (2019)	GPDS960Gray	12	Hybrid texture	WD	8.03
Zois et al. (2019)	GPDS960Gray	12	SR-K-SVD/OMP	WD	0.70
Zois, Alexandridis e Economou (2019)	GPDS960Gray	5	Poset-oriented grid	WI	3.06
Maruyama et al. (2021)	GPDS960Gray	$3 + (3 \times 22)$	SigNet-F + Duplicator	WD	0.20
Tsourounis et al. (2022)	GPDS960Gray	5	CNN-CoLL	WD	2.91
Tsourounis et al. (2022)	GPDS960Gray	12	CNN-CoLL	WD	2.12 ($\pm$ 0.76)
Yılmaz e Öztürk (2020)	GPDSsynthetic	5	RBP network	WD	31.16 ($\pm$ 0.31)
Yılmaz e Öztürk (2020)	GPDSsynthetic	12	RBP network	WD	24.22 ($\pm$ 0.21)
Zheng et al. (2021)	GPDSsynthetic	5	Micro deformations	WD	7.11 ($\pm$ 0.41)
Zheng et al. (2021)	GPDSsynthetic	10	Micro deformations	WD	5.38 ($\pm$ 0.36)
Zheng et al. (2021)	GPDSsynthetic	12	Micro deformations	WD	4.52 ($\pm$ 0.42)
Hafemann, Sabourin e Oliveira (2017a) (our test)	GPDSsynthetic	5	SigNet	WD	28.82 ($\pm$ 0.60)
Hafemann, Sabourin e Oliveira (2017a) (our test)	GPDSsynthetic	10	SigNet	WD	22.56 ($\pm$ 0.53)
Hafemann, Sabourin e Oliveira (2017a) (our test)	GPDSsynthetic	12	SigNet	WD	20.88 ($\pm$ 0.96)
Present Work	GPDSsynthetic	5	SigNet (Synthetic)	WD	4.24 ($\pm$ 0.43)
Present Work	GPDSsynthetic	10	SigNet (Synthetic)	WD	3.58 ($\pm$ 0.25)
Present Work	GPDSsynthetic	12	SigNet (Synthetic)	WD	3.53 ($\pm$ 0.23)
Present Work	GPDSsynthetic	5	MT-SigNet (Triplet)	WD	4.02 ($\pm$ 0.38)
Present Work	GPDSsynthetic	10	MT-SigNet (Triplet)	WD	3.24 ($\pm$ 0.21)
Present Work	GPDSsynthetic	12	MT-SigNet (Triplet)	WD	3.33 ($\pm$ 0.23)
Present Work	GPDSsynthetic	5	MT-SigNet (NT-Xent)	WD	4.19 ($\pm$ 0.33)
Present Work	GPDSsynthetic	10	MT-SigNet (NT-Xent)	WD	3.22 ($\pm$ 0.23)
Present Work	GPDSsynthetic	12	MT-SigNet (NT-Xent)	WD	3.15 ($\pm$ 0.21)
Souza, Oliveira e Sabourin (2018) (our test)	GPDSsynthetic	5	SigNet	WI	18.81 ($\pm$ 0.54)
Souza, Oliveira e Sabourin (2018) (our test)	GPDSsynthetic	10	SigNet	WI	17.79 ($\pm$ 0.47)
Souza, Oliveira e Sabourin (2018) (our test)	GPDSsynthetic	12	SigNet	WI	17.59 ($\pm$ 0.47)
Present Work	GPDSsynthetic	5	SigNet (Synthetic)	WI	5.67 ($\pm$ 0.24)
Present Work	GPDSsynthetic	10	SigNet (Synthetic)	WI	4.69 ($\pm$ 0.34)
Present Work	GPDSsynthetic	12	SigNet (Synthetic)	WI	4.44 ($\pm$ 0.16)
Present Work	GPDSsynthetic	5	MT-SigNet (Triplet)	WI	5.48 ($\pm$ 0.29)
Present Work	GPDSsynthetic	10	MT-SigNet (Triplet)	WI	4.69 ($\pm$ 0.43)
Present Work	GPDSsynthetic	12	MT-SigNet (Triplet)	WI	4.51 ($\pm$ 0.32)
Present Work	GPDSsynthetic	5	MT-SigNet (NT-Xent)	WI	5.47 ($\pm$ 0.39)
Present Work	GPDSsynthetic	10	MT-SigNet (NT-Xent)	WI	4.70 ($\pm$ 0.21)
Present Work	GPDSsynthetic	12	MT-SigNet (NT-Xent)	WI	4.52 ($\pm$ 0.26)

Table 14 – Comparison with the state-of-the-art in CEDAR dataset (errors in %).

Reference	#Samples	Features	Type	EER
Serdouk, Nemmour e Chibani (2016)	16	Gradient LBP + LRF	WD	3.54
Hamadene e Chibani (2016)	5	Contourlet Transform	WI	2.11
Hafemann, Sabourin e Oliveira (2017a)	12	SigNet-F	WD	4.63 ($\pm$ 0.42)
Hafemann, Sabourin e Oliveira (2017a)	4	SigNet	WD	5.87 ($\pm$ 0.73)
Hafemann, Sabourin e Oliveira (2017a)	12	SigNet	WD	4.76 ($\pm$ 0.36)
Hafemann, Oliveira e Sabourin (2018)	10	SigNet-SPP	WD	3.60 ($\pm$ 1.26)
Hafemann, Oliveira e Sabourin (2018)	10	SigNet-SPP (fine-tuned)	WD	2.33 ($\pm$ 0.88)
Lai e Jin (2018)	5	CNN with SPP	WD	4.37
Bhunia, Alaei e Roy (2019)	10	Hybrid texture	WD	6.66
Shariatmadari, Emadi e Akbari (2019)	12	HOCCNN	WD	4.94
Zois et al. (2019)	10	SR-K-SVD/OMP	WD	0.79
Zois, Alexandridis e Economou (2019)	5	Poset-oriented grid	WI	2.90
Souza et al. (2020)	12	SigNet	WI	5.86 ($\pm$ 0.50)
Ruiz et al. (2020)	1	SCINN	WI	4.84
Wan e Zou (2021)	5	SigCNN	WI	8.12 ($\pm$ 0.98)
Wan e Zou (2021)	12	SigCNN	WI	6.42 ($\pm$ 1.15)
Maruyama et al. (2021)	$3 + (3 \times 22)$	SigNet-F + Duplicator	WD	0.82
Liu et al. (2021)	10	MSDN	WD	1.75
Liu et al. (2021)	12	MSDN	WD	1.66
Liu et al. (2021)	1	MSDN	WI	6.74
Zheng et al. (2021)	5	Micro deformations	WD	3.89 ($\pm$ 0.45)
Zheng et al. (2021)	10	Micro deformations	WD	2.95 ($\pm$ 0.38)
Zheng et al. (2021)	12	Micro deformations	WD	2.76 ($\pm$ 0.43)
Tsourounis et al. (2022)	5	CNN-CoLL	WD	2.03
Tsourounis et al. (2022)	10	CNN-CoLL	WD	1.66 ($\pm$ 0.74)
Hafemann, Sabourin e Oliveira (2017a) (our test)	5	SigNet	WD	3.38 ($\pm$ 0.53)
Hafemann, Sabourin e Oliveira (2017a) (our test)	10	SigNet	WD	2.89 ($\pm$ 0.72)
Hafemann, Sabourin e Oliveira (2017a) (our test)	12	SigNet	WD	2.97 ($\pm$ 0.49)
Present Work	5	SigNet (Synthetic)	WD	4.64 ($\pm$ 0.38)
Present Work	10	SigNet (Synthetic)	WD	3.87 ($\pm$ 0.70)
Present Work	12	SigNet (Synthetic)	WD	3.77 ($\pm$ 0.44)
Present Work	5	MT-SigNet (Triplet)	WD	4.43 ($\pm$ 0.43)
Present Work	10	MT-SigNet (Triplet)	WD	3.45 ($\pm$ 0.55)
Present Work	12	MT-SigNet (Triplet)	WD	3.50 ($\pm$ 0.58)
Present Work	5	MT-SigNet (NT-Xent)	WD	4.72 ($\pm$ 0.31)
Present Work	10	MT-SigNet (NT-Xent)	WD	3.58 ($\pm$ 0.68)
Present Work	12	MT-SigNet (NT-Xent)	WD	3.32 ($\pm$ 0.44)
Souza et al. (2020) (our test)	5	SigNet	WI	5.46 ($\pm$ 1.48)
Souza et al. (2020) (our test)	10	SigNet	WI	4.20 ($\pm$ 1.45)
Souza et al. (2020) (our test)	12	SigNet	WI	3.63 ($\pm$ 1.13)
Present Work	5	SigNet (Synthetic)	WI	7.86 ($\pm$ 2.62)
Present Work	10	SigNet (Synthetic)	WI	6.20 ($\pm$ 1.96)
Present Work	12	SigNet (Synthetic)	WI	6.07 ($\pm$ 1.78)
Present Work	5	MT-SigNet (Triplet)	WI	5.91 ($\pm$ 2.05)
Present Work	10	MT-SigNet (Triplet)	WI	4.91 ($\pm$ 1.30)
Present Work	12	MT-SigNet (Triplet)	WI	4.59 ($\pm$ 1.15)
Present Work	5	MT-SigNet (NT-Xent)	WI	6.41 ($\pm$ 1.40)
Present Work	10	MT-SigNet (NT-Xent)	WI	5.57 ($\pm$ 0.92)
Present Work	12	MT-SigNet (NT-Xent)	WI	5.32 ($\pm$ 0.91)

2021; HAFEMANN; OLIVEIRA; SABOURIN, 2018) have presented better results than the proposed framework. This is expected because the SigNet-F model is trained using skilled forgeries from the GPDS960Gray dataset.

Experimental results with the CEDAR dataset are listed in Table 14. With the CEDAR

dataset, we obtained improvements for writer-dependent and writer-independent signature verification in relation to the works (HAFEMANN; SABOURIN; OLIVEIRA, 2017a; HAFEMANN; OLIVEIRA; SABOURIN, 2018; SOUZA et al., 2020) that adopt SigNet and SigNet-F models for feature extraction. Besides, we noticed improvements in writer-dependent verification regarding recent hand-engineering (BHUNIA; ALAEI; ROY, 2019; SERDOUK; NEMMOUR; CHIBANI, 2016)

Table 15 – Comparison with the state-of-the-art in MCYT-75 dataset (errors in %).

Reference	#Samples	Features	Type	EER
Soleimani, Araabi e Fouladi (2016)	5	LBP + DMML	WD	13.44
Soleimani, Araabi e Fouladi (2016)	10	LBP + DMML	WD	9.86
Serdouk, Nemmour e Chibani (2017)	10	HOT	WD	10.60
Diaz et al. (2017)	8	Duplicator	WD	9.12
Hafemann, Sabourin e Oliveira (2017a)	10	SigNet-F	WD	3.00 ($\pm$ 0.56)
Hafemann, Sabourin e Oliveira (2017a)	5	SigNet	WD	3.58 ($\pm$ 0.54)
Hafemann, Sabourin e Oliveira (2017a)	10	SigNet	WD	2.87 ($\pm$ 0.42)
Hafemann, Oliveira e Sabourin (2018)	10	SigNet-SPP	WD	3.64 ($\pm$ 1.04)
Hafemann, Oliveira e Sabourin (2018)	10	SigNet-SPP (fine-tuned)	WD	3.40 ($\pm$ 1.08)
Maergner et al. (2018)	5	CNN-Triplet + GED	WI	10.67
Maergner et al. (2018)	10	CNN-Triplet + GED	WI	10.13
Lai e Jin (2018)	5	CNN with SPP	WD	3.78
Masoudnia et al. (2019)	10	MLSE	WD	5.85 ($\pm$ 0.71)
Shariatmadari, Emadi e Akbari (2019)	10	HOCCNN	WD	5.46
Bhunia, Alaei e Roy (2019)	10	Hybrid texture	WD	9.26
Zois et al. (2019)	10	SR-K-SVD/OMP	WD	1.37
Zois, Alexandridis e Economou (2019)	5	Poset-oriented grid	WI	3.50
Ruiz et al. (2020)	1	SCINN	WI	2.06
Wan e Zou (2021)	5	SigCNN	WI	17.24 ($\pm$ 1.07)
Wan e Zou (2021)	12	SigCNN	WI	15.04 ($\pm$ 0.93)
Souza et al. (2020)	10	SigNet	WI	2.99 ($\pm$ 0.16)
Maruyama et al. (2021)	$3 + (3 \times 22)$	SigNet-F + Duplicator	WD	0.01
Tsourounis et al. (2022)	5	CNN-CoLL	WD	2.61
Tsourounis et al. (2022)	10	CNN-CoLL	WD	1.62
Hafemann, Sabourin e Oliveira (2017a) (our test)	5	SigNet	WD	3.79 ($\pm$ 0.52)
Hafemann, Sabourin e Oliveira (2017a) (our test)	10	SigNet	WD	3.23 ($\pm$ 0.65)
Present Work	5	SigNet (Synthetic)	WD	4.28 ($\pm$ 0.99)
Present Work	10	SigNet (Synthetic)	WD	3.24 ($\pm$ 0.60)
Present Work	5	MT-SigNet (Triplet)	WD	4.07 ($\pm$ 0.46)
Present Work	10	MT-SigNet (Triplet)	WD	2.71 ($\pm$ 0.53)
Present Work	5	MT-SigNet (NT-Xent)	WD	4.01 ($\pm$ 0.77)
Present Work	10	MT-SigNet (NT-Xent)	WD	3.22 ($\pm$ 0.65)
Souza et al. (2020) (our test)	5	SigNet	WI	5.08 ($\pm$ 2.08)
Souza et al. (2020) (our test)	10	SigNet	WI	3.86 ($\pm$ 1.83)
Present Work	5	SigNet (Synthetic)	WI	6.50 ($\pm$ 1.13)
Present Work	10	SigNet (Synthetic)	WI	4.99 ($\pm$ 0.82)
Present Work	5	MT-SigNet (Triplet)	WI	4.97 ($\pm$ 1.31)
Present Work	10	MT-SigNet (Triplet)	WI	4.07 ($\pm$ 1.21)
Present Work	5	MT-SigNet (NT-Xent)	WI	4.74 ($\pm$ 1.52)
Present Work	10	MT-SigNet (NT-Xent)	WI	3.95 ($\pm$ 0.77)

and deep-learning (SHARIATMADARI; EMADI; AKBARI, 2019) methods. Concerning the related contrastive methods, experimental results with the CEDAR dataset outperform those presented in Ruiz et al. (2020), Liu et al. (2021), Wan e Zou (2021) following a writer-independent approach. However, Liu et al. (2021) has presented better results than ours in writer-dependent verification. It is important to note that models in Liu et al. (2021) were trained using skilled forgeries obtained from the own CEDAR dataset. In contrast, skilled forgeries are not used in training in the present work. Besides, Zheng et al. (2021) presented lower error rates than ours for writer-dependent verification with the CEDAR dataset, but on the other hand, we reiterate that our models performed better than those presented in Zheng et al. (2021) in the face of the GPDS-150S and GPDS-300S datasets.

Experimental results regarding the MCYT-75 dataset are listed in Table 15. Our experimental results presented lower error rates than those reported in Hafemann, Sabourin e Oliveira (2017a), Hafemann, Oliveira e Sabourin (2018) when adopting SigNet and SigNet-F models for feature extraction in writer-dependent verification, even though obtained results do not outperform Souza et al. (2020) for writer-independent verification. Our results with the MCYT-75 dataset also outperform recent hand-engineering (BHUNIA; ALAEI; ROY, 2019; SERDOUK; NEMMOUR; CHIBANI, 2017; DIAZ et al., 2017) and deep learning (SHARIATMADARI; EMADI; AKBARI, 2019; SOLEIMANI; ARAABI; FOULADI, 2016) related methods for writer-dependent verification. Moreover, our proposed contrastive framework even with a simpler training process outperforms the method proposed in Masoudnia et al. (2019), in which three different losses (cross-entropy loss, CauchySchwarz divergence, and hinge loss) are combined as an ensemble framework for feature learning. When considering recent contrastive learning methods for writer-independent verification, our experimental results considerably outperform those presented in Maergner et al. (2018), Wan e Zou (2021) but do not outperform those presented in Ruiz et al. (2020). In Ruiz et al. (2020), tests are performed without considering skilled forgeries, which is an easier verification problem. Therefore, our results cannot be directly compared to Ruiz et al. (2020).

It is important mentioning that the contrastive method proposed in Tsourounis et al. (2022) has presented better results than ours in all evaluated datasets. However, in the experimental protocol defined by Tsourounis et al. (2022), the hyper-parameters of writer-dependent classifiers are specifically adjusted for each user through a cross-validation procedure. The implementation of such cross-validation procedure is not publicly available, then we could not reproduce it. It is unclear whether the good obtained performance is due to better feature representations or an exhaustive search process for achieving better results. Under these con-

ditions, the experimental results of this work with the GPDS-300S, CEDAR and MCYT-75 datasets can not be fairly compared to those presented in Tsourounis et al. (2022). In the experimental protocol conducted in this work, hyper-parameters of support vector machines are fixed to precisely evaluate the improvement on obtained feature representations.

3.5 CONCLUSION

One main contribution of this work is to propose a multi-task framework based on contrastive losses with hard negative mining for offline handwritten signature feature extraction in writer-dependent and writer-independent verification systems. Models optimized with the proposed framework were experimentally compared to the state-of-the-art model called SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a), in a effort to assess whether better suitable feature representations are obtained when following the proposed method. In Tsourounis et al. (2022), it is showed that the performance of the SigNet is obtainable with less training signature examples by appropriately pre-training the models with external textual based data. With a similar purpose, in this work, we identified that an external data source of synthetic signatures can reasonably be used for real signature verification in the lack of real signature data for training deep learning models. However, we found that the intrinsic characteristics of real genuine signatures are not perfectly embedded in the synthetic signature images.

Moreover, experiments indicated that:

i) Adopting larger margin (in the Triplet loss) and temperature (in the NT-Xent loss) hyper-parameters worsens the verification of genuine signatures and skilled forgeries as these representations overlap in the space. Higher hyper-parameter values move skilled forgeries representations from the genuine ones but introduce a side effect that increases the distance between genuine signature representations. Therefore, adopting smaller hyper-parameters balances these two factors, providing lower equal error rates.

ii) Single-task contrastive models have lost the ability to generalize to other datasets with a disjoint set of users. In transfer learning scenarios, adopting the proposed framework is required to provide a better generalization performance to other datasets. We demonstrated that contrastive losses do not replace the cross-entropy loss for the offline signature verification problem, but in fact, contrastive losses should be used in conjunction with the cross-entropy loss following the proposed framework.

iii) Finally, concerning the generalization performance of contrastive models, the multi-

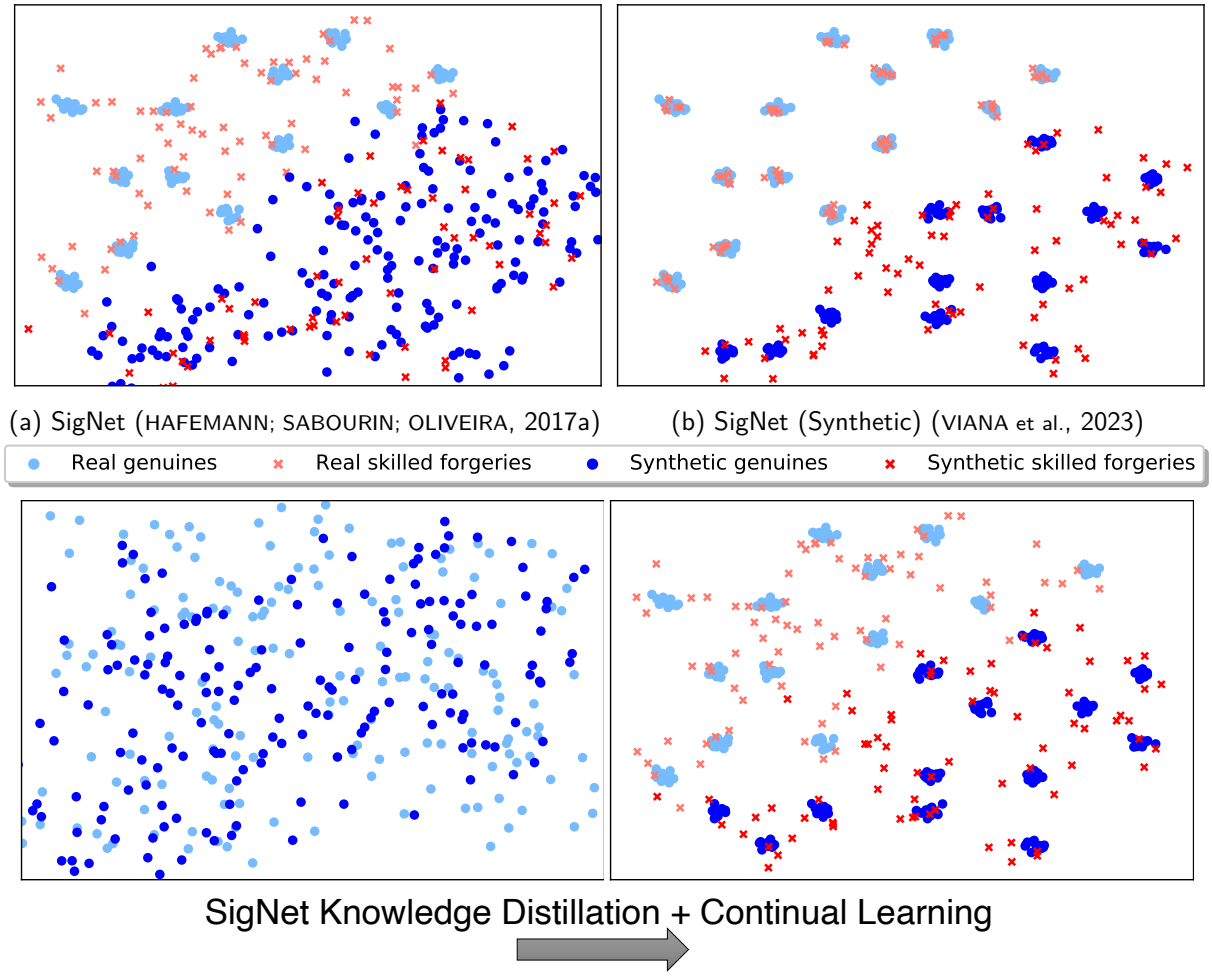
task contrastive model optimized with the Triplet loss provided better feature representations for writer-dependent and writer-independent verification. This result indicates that the explicit mining of hard negatives performed by the Triplet loss provides a more powerful model correction than the implicit mining mechanism of the NT-Xent loss.

4 EXPANDING GENERALIZATION OF HANDWRITTEN SIGNATURE FEATURE REPRESENTATION THROUGH DATA KNOWLEDGE DISTILLATION

4.1 MOTIVATION

The GPDS-960 dataset (VARGAS et al., 2007) used to be the largest publicly available Western script dataset of offline handwritten signatures for training deep learning models. For illustration, whereas GPDS-960 provides signatures of 881 users, other public datasets containing real signatures, such as CEDAR (KALERA; SRIHARI; XU, 2004) and MCYT-75 (ORTEGA-GARCIA et al., 2003), have 55 and 75 users. Nevertheless, the GPDS-960 dataset is no longer publicly available due to data protection regulatory issues stated by The General Data Protection Regulation (EU) 2016/679. Since 2016, the former researchers who possess the dataset have continued performing experiments with the GPDS-960 for research purposes (HAFEMANN; SABOURIN; OLIVEIRA, 2017a; YILMAZ; ÖZTÜRK, 2020; ARAB; NEMMOUR; CHIBANI, 2020; MARUYAMA et al., 2021; BOUAMRA et al., 2022; ARAB; NEMMOUR; CHIBANI, 2023). On the other hand, new investigators starting research in the offline signature verification field have suffered from the absence of a public large-scale Western script signature dataset with real signatures for training models. Therefore, as a replacement for the GPDS-960, new researchers have resorted (ZHENG et al., 2021; VIANA et al., 2023; YAPıCı; TEKEREK; TOPALOĞLU, 2021; MAERGNER et al., 2019) to adopt the GPDSsynthetic (FERRER et al., 2017) dataset, which has a large-scale set of synthetically generated user signatures, to perform the training of deep learning models.

However, we have found a difference in verification performance (VIANA et al., 2023; YILMAZ; ÖZTÜRK, 2020) between models trained using real signature data from the GPDS-960 dataset (VARGAS et al., 2007) and synthetic signature data from the GPDSsynthetic dataset (FERRER et al., 2017). For instance, the *SigNet* (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) is a cross-entropy based model trained with GPDS-960 signature data whereas the *SigNet (Synthetic)* (VIANA et al., 2023) is a cross-entropy based model trained using GPDSsynthetic signature data. As a result of the performance evaluation of these models, we have observed that: i) A remarkably unsatisfactory performance for the verification of synthetic skilled forgeries and synthetic random forgeries is obtained using SigNet (this scenario is visually represented in Figure 18a). This occurs because the SigNet model provides incorrect cluster separation when applied to synthetic signature data and thus has substantial difficulty in separating synthetic skilled forgeries from poorly formed genuine signature clusters. ii) In the opposite situation



(a) SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) (b) SigNet (Synthetic) (VIANA et al., 2023)

(c) Proposed Method

Figure 18 – Our motivation.

(illustrated in Figure 18b), the SigNet (Synthetic) model has difficulty in the verification of real skilled forgeries. However, the SigNet (Synthetic) model maintains suitable cluster separation in real and synthetic datasets, supporting the ability to verify real and synthetic random forgeries as the synthetic dataset provides more diversity with a large number of different users. This way, models trained with real signature data provide insufficient performance when tested in the face of a more diverse set of synthetic signatures, and models trained with synthetic signature data are not entirely sufficient for verifying real genuine signatures and skilled forgeries.

To deal with this problem, in our proposed method (illustrated in Figure 18c) we resort to a pre-trained SigNet model as a foundation to provide knowledge about the characteristics of real signature data in the training of new models. In this sense, we complement the SigNet predefined real data feature space using synthetic data while minimizing the divergence in distribution between the representations provided by these two different types of data sources. This is achieved through class-incremental continual learning based on knowledge distillation.

In this work, we demonstrate that knowledge distillation through SigNet enhances skilled forgery detection capability, which is provided by an inherent ability of the model to separate genuine representations from skilled forgery representations in a region of uncertainty within the representation space. This separation capability is helpful in the verification of various types of signature data sources. Besides, continual learning based on synthetic data enables the model to be more robust in the verification of signatures on other domains, for instance, signatures of non-Western scripts.

More than that, we empirically demonstrate a trade-off in verification performance between the GPDS-960 and GPDSsynthetic datasets in such a way that a training configuration that balances verification between the two data sources is encountered. We show that this divergence occurs due to side effects introduced by excessively shifting the model distribution for any of these data sources. Thus, aiming to alleviate such side effects, we extend the multi-task framework proposed in Chapter 3 to this new scenario and propose fine-tuning the combined representation space based on a contrastive objective. This objective minimizes intra-class distances between representations to reduce the variability of distances among different users, providing a more uniform verification decision boundary across them.

Finally, it is important to mention that an auxiliary dataset with examples having a similar distribution as the GPDS-960 data is required to be propagated during training in order to provide knowledge distillation from the SigNet model. To this end, we generate inverted examples that have the same distribution as the real GPDS-960 examples. We demonstrate that the examples inverted from the distribution pre-coded in the SigNet model can replace the GPDS-960 dataset to provide knowledge about real signatures in situations where this dataset is unavailable. As the examples are inverted using a data-free technique, these do not explicitly show the original signature shapes of the GPDS-960 examples, making it possible to maintain a protection level when publicly sharing the inverted dataset to the research community. As the research community can benefit from this data in the training of other deep backbones and representation learning schemes, in this work, we preliminarily evaluate our distillation mechanism in the training of other recent state-of-the-art deep backbones for signature verification.

4.2 COMBINED HANDWRITTEN SIGNATURE REPRESENTATION WITH CONTINUAL LEARNING OF SYNTHETIC DATA BASED ON KNOWLEDGE DISTILLATION SUPPORTED BY INVERTED SIGNATURE DATA

In the proposed method, we take advantage of the greater availability of synthetic signature data for model training whereas minimizing the distribution divergence of such data to real signature data during the training process. To this end, we employ examples inverted from SigNet (a pre-trained model with GPDS-960 dataset) that follow a real distribution, along with synthetic examples obtained from a pre-existing synthetic dataset (the GPDSsynthetic dataset). Initially, given that the SigNet model is pre-trained on a set of real classes $\mathcal{R}$, the

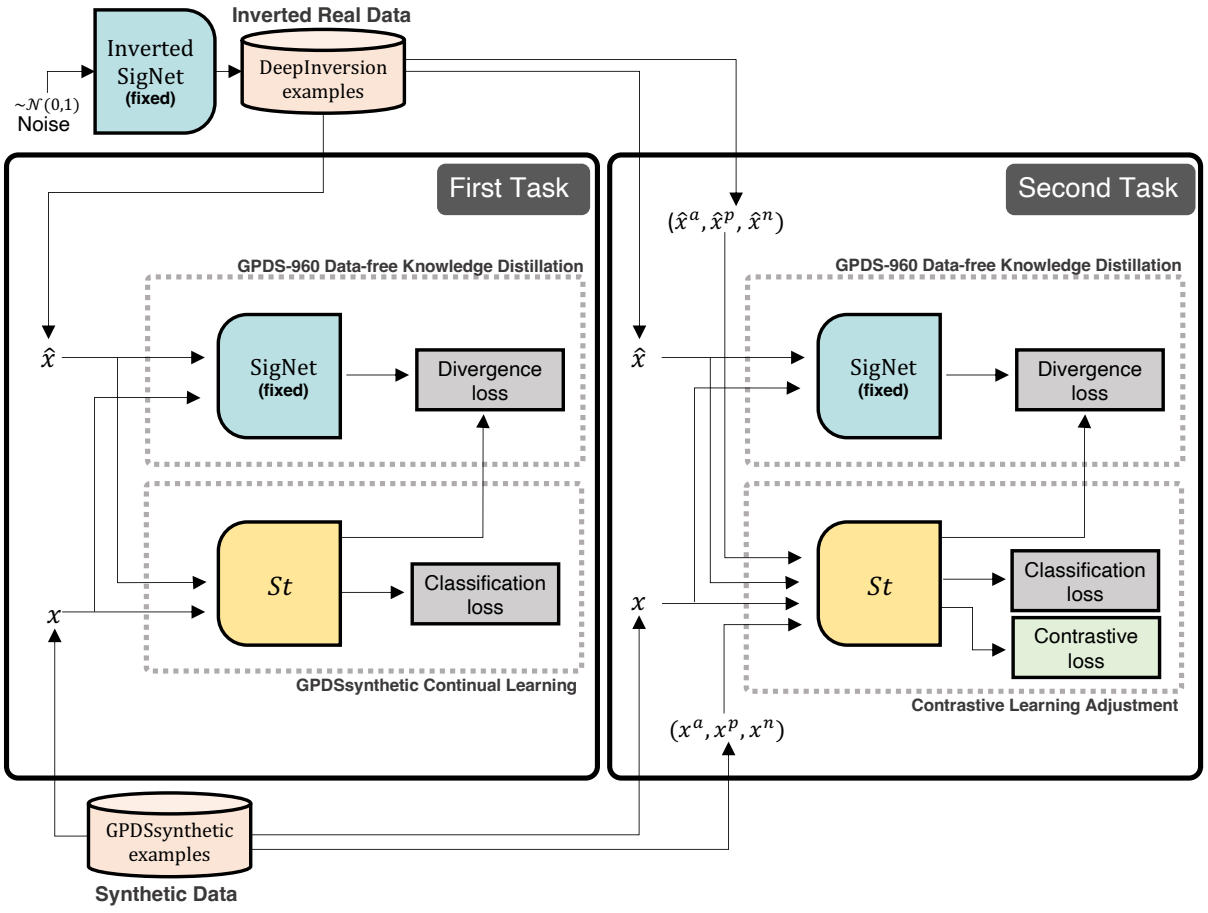


Figure 19 – Overview of the proposed method. The *Inverted SigNet* module inverts a set of examples $\hat{x} \sim \mathcal{N}(0,1)$ that follow a real distribution using *Deep Inversion*. Besides, synthetic examples x are obtained from a pre-existing synthetic dataset (the *GPDSsynthetic* dataset). The *SigNet* is a pre-trained model on a real dataset (the *GPDS-960* dataset). Given this, in the *First Task*, a new student model St is trained using synthetic data x while minimizing the divergence in distribution with the inverted data $\hat{x}$. After that, in the *Second Task*, the combined representation space provided by the student model St is adjusted according to a contrastive objective using synthetic triplets (x^a, x^p, x^n) and inverted triplets $(\hat{x}^a, \hat{x}^p, \hat{x}^n)$. The *SigNet* teacher has 531 probability outputs, whereas the student St has 8.481 probability outputs, since the latter is optimized using both inverted data of 531 users and synthetic data of 7950 users.

specific goal of the *first task* of the proposed method is to generate a new student model St trained on a set of synthetic classes $\mathcal{S}$, such that it produces predictions in a combined space $\mathcal{R} \cup \mathcal{S}$ that allows extracting features in a representation space that complements global characteristics of real and synthetic signatures. In this context, we denote $\hat{x} \in \mathcal{R}$ as an *inverted* example. Besides, examples directly obtained from the *synthetic* dataset are represented as $x \in \mathcal{S}$. After that, in a *second task*, the representation space provided by the student model St is adjusted according to a contrastive objective: we contrast synthetic triplet examples (x^a, x^p, x^n) with each other and inverted triplet examples $(\hat{x}^a, \hat{x}^p, \hat{x}^n)$ with each other. The proposed method is summarized in the framework illustrated in Figure 19.

4.2.1 Data-free Deep Inversion of Real Signature Data

The objective of the *Inverted SigNet* module (illustrated in Figure 19) is to synthesize a set of examples inverted from noise $\hat{x} \sim \mathcal{N}(0, 1)$ that maintain classification accuracy in the pre-trained SigNet model. To achieve this, we apply the *Deep Inversion* technique proposed in Yin et al. (2020) to our investigated problem. This way, the pre-trained SigNet model provides statistics that are used to generate new examples that have the same distribution as the real examples $\hat{x}' \in \mathcal{R}$ originally used to train the SigNet model. These generated examples can be subsequently applied to provide data-free knowledge distillation since they are generated without the need for direct access to the original data but only to a pre-trained model that encodes the original data distribution. An overview of the inversion method is illustrated in Figure 20.

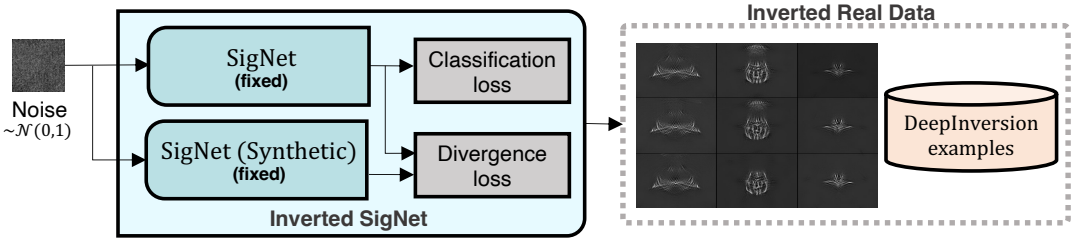


Figure 20 – Overview of the inversion method.

4.2.1.1 SigNet Deep Inversion

In this sense, the inverted real examples $\hat{x} \in \mathcal{R}$ are optimized according to the $\mathcal{L}_{DI}$ loss function (YIN et al., 2020) defined as:

$$\min_{\hat{x}} \mathcal{L}_{DI}(\hat{x}) \;=\; \min_{\hat{x}} \alpha_{\text{l}} \mathcal{L}_{\text{SigNet}}(\hat{x}, y) \;+\; \alpha_{\text{tv}} \mathcal{R}_{\text{TV}}(\hat{x}) \;+\; \alpha_{\ell_2} \mathcal{R}_{\ell_2}(\hat{x}) \;+\; \alpha_{\text{f}} \mathcal{R}_{\text{feature}}\,(\hat{x}) \quad (4.1)$$

where $\mathcal{L}_{\text{SigNet}}(\cdot)$ is the cross-entropy classification loss obtained with the SigNet, and $\mathcal{R}_{\text{TV}}(\cdot)$, $\mathcal{R}_{\ell_2}(\cdot)$ are regularization terms that respectively penalize the total variance and ℓ_2 norm of $\hat{x}$ with scaling factors α_{l}, α_{tv}, α_{ℓ_2}. Besides, $\mathcal{R}_{\text{feature}}\,(\hat{x})$ is the feature distribution regularization term scaled with α_{f} and defined by:

$$\mathcal{R}_{\text{feature}}\,(\hat{x}) \;=\; \sum_{l} \left\| \mu_l(\hat{x}) - \mathbb{E}\left(\mu_l(\hat{x}') \mid \mathcal{R}\right) \right\|_2 \;+\; \sum_{l} \left\| \sigma_l^2(\hat{x}) - \mathbb{E}\left(\sigma_l^2(\hat{x}') \mid \mathcal{R}\right) \right\|_2 \quad (4.2)$$

where $\mu_l(\hat{x})$ and $\sigma_l^2(\hat{x})$ are the batch-wise mean and variance estimates of feature maps corresponding to the l^{th} convolutional layer of the SigNet model, and $\mathbb{E}\left(\mu_l(\hat{x}') \mid \mathcal{R}\right)$ and $\mathbb{E}\left(\sigma_l^2(\hat{x}') \mid \mathcal{R}\right)$ are the running average statistics stored in the SigNet batch normalization layers. The feature distribution regularization term $\mathcal{R}_{\text{feature}}\,(\hat{x})$ effectively enforces feature similarities at all convolutional levels when minimizing the distance between feature map statistics for $\hat{x}$ and $\hat{x}'$ (YIN et al., 2020).

4.2.1.2 Introducing Competition on SigNet Deep Inversion

Inspired by the adaptive inversion loss proposed in Yin et al. (2020), we propose a new adapted objective function that introduces competition between models trained with real and synthetic signature data when synthesizing inverted examples. We evaluate the effect in the generation of inverted examples by competing the *SigNet* model (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) (trained on GPDS-960 data) with other model trained on synthetic data from the GPDSsynthetic dataset: the *SigNet (Synthetic)* model (VIANA et al., 2023).

In this scenario, the inverted examples $\hat{x} \in \mathcal{R}$ are optimized according to the deep inversion loss $\mathcal{L}_{DI}$ defined in Equation 4.1 with an additional loss term $\mathcal{R}_{\text{compete}}\,(\hat{x})$ that encourages the inverted synthesized examples to cause disagreement between two model distributions in

order to enhance the diversity of generated examples. The inverted real $\hat{x} \in \mathcal{R}$ examples are optimized as:

$$\min_{\hat{x}} \mathcal{L}_{DI}(\hat{x}) + \alpha_c \mathcal{R}_{\text{compete}}(\hat{x}) = \tag{4.3}$$

$$\min_{\hat{x}} \mathcal{L}_{DI}(\hat{x}) + \alpha_c (1 - \text{JS}\left(p_{\text{SigNet}}(\hat{x}), p_{\text{SigNet-Synthetic}}(\hat{x})\right)) \tag{4.4}$$

where JS is the measure of Jensen-Shannon divergence (FUGLEDE; TOPSOE, 2004) between the *SigNet* and *SigNet (Synthetic)* models:

$$\text{JS}\left(p_{\text{SigNet}}(\hat{x}), p_{\text{SigNet-Synthetic}}(\hat{x})\right) = \frac{1}{2}\left(\text{KL}\left(p_{\text{SigNet}}(\hat{x}), M\right) + \text{KL}\left(p_{\text{SigNet-Synthetic}}(\hat{x}), M\right)\right) \tag{4.5}$$

such that,

$$M = \frac{1}{2} \cdot \left(p_{\text{SigNet}}(\hat{x}) + p_{\text{SigNet-Synthetic}}(\hat{x})\right) \tag{4.6}$$

is the average of the real and synthetic distributions. We aim to promote the generation of inverted examples that produce greater divergence in relation to the average of the probability distributions provided by the SigNet and SigNet (Synthetic). To this end, we employ Jensen-Shannon divergence, which is a symmetric divergence measure (DOMENICO et al., 2015).

In Equation 4.3, the hyper-parameter α_c scales the influence of the competition between these models in the inversion process. Our goal is to ideally synthesize new inverted examples from a real distribution that cannot be correctly captured by the synthetic data distribution provided by the SigNet (Synthetic). Therefore, the purpose of Equation 4.3 is to induce the generated examples to cause disagreement between the SigNet and SigNet (Synthetic) outputs.

4.2.2 First Task: Combining Real and Synthetic Signature Representations through Continual Learning

Initially, in the proposed method (illustrated in Figure 19), we complement a predefined feature space based on real data using synthetic data while minimizing the divergence in distribution between the representations provided by these two different types of data sources by means of knowledge distillation. With a class-incremental continual learning configuration (YIN et al., 2020), a student model St is trained to learn new synthetic classes while simultaneously distilling knowledge from the SigNet to generate an improved model that combines signature

representations from the real and synthetic feature spaces, avoiding catastrophic forgetting (LI; HOIEM, 2018). To achieve this, we optimize the student model St using inverted ($\hat{x} \in \mathcal{R}$) and synthetic ($x \in \mathcal{S}$) data according to the following loss function:

$$\min_{\mathbf{W}_{St}} \sum_{\hat{x}\in\mathcal{R},x\in\mathcal{S}} \Big[\mathcal{L}_{Continual}\Big] = \min_{\mathbf{W}_{St}} \sum_{\hat{x}\in\mathcal{R},x\in\mathcal{S}} \Big[\mathrm{KL}_{\mathsf{real}}\left(p_{\mathsf{SigNet}}(\hat{x}), p_{St}(\hat{x})\right) + \mathrm{CE}(y, p_{St}(x)) + \lambda_p \mathrm{KL}_{\mathsf{synthetic}}(p_{\mathsf{SigNet}}(x), p_{St}(x))\Big] = \quad (4.7)$$

$$\min_{\mathbf{W}_{St}} \sum_{\hat{x}\in\mathcal{R},x\in\mathcal{S}} \Bigg[p_{\mathsf{SigNet}}(\hat{x}) \log\left(\frac{p_{\mathsf{SigNet}}(\hat{x})}{p_{St}(\hat{x})}\right) - y \log p_{St}(x) + \lambda_p p_{\mathsf{SigNet}}(x) \log\left(\frac{p_{\mathsf{SigNet}}(x)}{p_{St}(x)}\right)\Bigg] \quad (4.8)$$

We adapt a continual learning loss $\mathcal{L}_{Continual}$ (Equation 4.7, which is expanded into Equation 4.8) with a class incremental approach defined in Yin et al. (2020) to our investigated problem. Specifically, the loss is composed by a first distillation term (denoted as $\mathrm{KL}_{\mathsf{real}}$) that applies the Kullback-Leibler divergence (HINTON; VINYALS; DEAN, 2015) aiming to measure the difference in prediction score distributions between the SigNet and a new student model being trained considering inverted examples. In this case, the teacher output $p_{\mathsf{SigNet}}(\hat{x})$ with 531 probabilities is incremented with zeros for comparison with the student output $p_{St}(\hat{x})$ with 8.481 probabilities. Secondly, we adopt an additional cross-entropy term (denoted as CE) to distinguish between new incremental synthetic classes in $\mathcal{S}$ that are complementary to real classes in $\mathcal{R}$. Finally, we use a third distillation term (denoted as $\mathrm{KL}_{\mathsf{synthetic}}$) that measures the Kullback-Leibler divergence between the predictions generated by the incremented student with synthetic classes in relation to the original SigNet predictions. Thus, we compute this latter divergence considering the SigNet outputs $p_{\mathsf{SigNet}}(x)$ and the first 531 outputs of the student model $p_{St}(x)$. Note that the λ_p hyper-parameter enforces the intensity such that the divergence between synthetic and real signature characteristics is minimized into the new combined representation space.

It is worth noting that since synthetic data is used in training, the student classification layer has considerably more outputs than the teacher. As expected in our experiments, the student model occupies more memory (132.68 MB) compared to the SigNet teacher (67.52 MB). Furthermore, the optimization time for the continual learning loss (Equation 4.7) is higher (4 hours and 39 minutes) than the cross-entropy loss (Equation 2.1) of the teacher (25

minutes). Despite introducing additional computational overhead during the training phase of the student model, its classification layer is not used in testing. During inference, input signature data is propagated through the trained student up to the final dense fully connected layer, which occupies only 63.17 MB of memory. Consequently, when extracting features from a questioned signature, there is no difference in inference performance between the teacher and student models, assuming both adopt the same backbone architecture.

4.2.3 Second Task: Contrastive Adjustment of the Combined Representation Space

We found that the process of minimizing the divergence between real and synthetic data distributions in the first task increases the intra-class distances of the synthetic representations. Given this problem, we resort to a contrastive objective to refine the combined representation space of the first task. Contrastive losses applied in a pre-organized representation space provide a refinement of the representation space by generating denser genuine clusters and locally separating skilled forgery representations (VIANA et al., 2023). Following this, in the first task a combined feature space that map real and synthetic signature examples is generated, and after that, this space is locally adjusted in a second task (as illustrated in Figure 19). To this end, we adopt a triplet-based contrastive loss.

However, we desire to not undo the complementary $\mathcal{R} \cup \mathcal{S}$ representation space between inverted real and synthetic data obtained in the first task. Thus, the adjustment is applied separately for each data source type: contrasting synthetic examples (x^a, x^p, x^n) with each other, as well as contrasting inverted real examples $(\hat{x}^a, \hat{x}^p, \hat{x}^n)$ with each other. Furthermore, we desire to prevent model distribution shifts towards the synthetic or real data distributions. Therefore, in the contrastive adjustment we simultaneously optimize $\mathcal{L}_{Continual}$ (Equation 4.7) with a fixed λ_p aiming to hold the combined representation space. Given this, in the second task we adjust the representation space provided by the student model St according to the following loss function:

$$\min_{\mathbf{W}_{St}} \left[\sum_{(\hat{x}^a, \hat{x}^p, \hat{x}^n) \in \mathcal{R}} \left(\hat{\mathcal{L}}_{Triplet} \right) + \sum_{(x^a, x^p, x^n) \in \mathcal{S}} \left(\mathcal{L}_{Triplet} \right) + \sum_{\hat{x} \in \mathcal{R}, x \in \mathcal{S}} \left(\mathcal{L}_{Continual} \right) \right] \quad (4.9)$$

where $\hat{\mathcal{L}}_{Triplet} = \max(0, m + \|f(\hat{x}^a) - f(\hat{x}^p)\|_2^2 - \|f(\hat{x}^a) - f(\hat{x}^n)\|_2^2)$,

and $\mathcal{L}_{Triplet} = \max(0, m + \|f(x^a) - f(x^p)\|_2^2 - \|f(x^a) - f(x^n)\|_2^2)$,

adopting a defined margin $m = 0.1$ hyper-parameter using semi-hard triplets such that:

$\|f(\hat{x}^a) - f(\hat{x}^n)\|_2^2 < m + \|f(\hat{x}^a) - f(\hat{x}^p)\|_2^2$ with $\|f(\hat{x}^a) - f(\hat{x}^p)\|_2^2 < \|f(\hat{x}^a) - f(\hat{x}^n)\|_2^2$, and $\|f(x^a) - f(x^n)\|_2^2 < m + \|f(x^a) - f(x^p)\|_2^2$ with $\|f(x^a) - f(x^p)\|_2^2 < \|f(x^a) - f(x^n)\|_2^2$, for the optimization of the model (VIANA et al., 2023).

4.2.4 Distillation to Large Vision Architectures

We evaluate the distillation performance of the proposed method to other state-of-the-art deep learning backbones. Specifically, we assess the signature verification performance of feature representations obtained through models with large vision architectures. Firstly, the student St is modified to a deep architecture with a large number of convolutional layers: we adopt a 152-layer ResNet (HE et al., 2016) that through residual connections is expected to obtain accuracy gains with a large number of layers. Furthermore, as shown in (HE et al., 2016), the representations provided by ResNets have excellent generalization performance in recognition tasks. Secondly, we evaluate the distillation performance for a student St founded on a self-attention based architecture: the Vision Transformer (ViT) (DOSOVITSKIY et al., 2021). This architecture is inspired by the Transformers (VASWANI et al., 2017), which have become the standard state-of-the-art architecture for large-scale natural language processing tasks. In the case of vision transformers, a sequence of image patches provides a sequence of embeddings used as input to a transformer encoder.

4.3 RESEARCH QUESTIONS

The main objective of this chapter is to investigate whether the combined representation space generated from knowledge about synthetic and real signatures provides a more balanced signature verification across different datasets. Yet, to our knowledge, a framework for continual representation learning using synthetic handwritten signature data based on knowledge distillation using inverted real handwritten signature data has not been investigated in the context of handwritten signature verification problem. Given this, the following Research Questions (RQs) are investigated in this chapter:

RQ5) How can we obtain a more suitable signature representation with models trained using synthetic and inverted data in a situation where the access to the GPDS-960 dataset is unavailable?

RQ6) Is it feasible to rely solely on synthetic data from the GPDSsynthetic dataset for training models?

RQ7) Can the characteristics of real and synthetic datasets be complemented using continual learning to offer a more balanced verification performance across diverse datasets?

RQ8) Is SigNet knowledge distillation helpful in training state-of-the-art large vision models for handwritten signature feature representation learning?

4.4 EXPERIMENTS

4.4.1 Experimental Setup

4.4.1.1 Experimental Protocol for SigNet Deep Inversion

In order to determine the best hyper-parameter configuration for SigNet deep inversion, we generated inverted validation sets with 50 handpicked fixed users, such that each user has 24 inverted genuine signatures. In the case of SigNet deep inversion (Equation 4.1), each validation set is generated with an associated distribution regularization scaling factor hyper-parameter α_{f} around the values reported in Yin et al. (2020) with experiments concerning the ImageNet dataset: we proceeded with a grid search and evaluated $\alpha_{\mathrm{f}} \in \{0.001, 0.01, 0.1, 0.5, 1.0\}$ and found that $\alpha_{\mathrm{f}} = 0.1$ provided the best accuracy in the classification of the inverted examples using SigNet, obtaining an average accuracy of 99.92%. Besides, we set $\alpha_{\mathrm{tv}} = 10^{-4}$ and $\alpha_{\ell_2} = 10^{-7}$ following Yin et al. (2020). In the case of SigNet deep inversion with competition (Equation 4.3), we kept these best hyper-parameters initially found, and each validation set is generated with an associated competition scaling factor hyper-parameter α_{c} around the values reported in Yin et al. (2020) with experiments regarding the ImageNet dataset: we proceeded with a grid search and evaluated $\alpha_{\mathrm{c}} \in \{0.2, 0.5, 0.8, 1.0\}$ and found that a mid-term competition coefficient $\alpha_{\mathrm{f}} = 0.5$ provided the best accuracy in the classification of the inverted examples using SigNet, obtaining an average accuracy of 99.58%.

Finally, given the definition of the best hyper-parameter configurations, we generated two definitive inverted sets: $\hat{\mathcal{D}}_r$ (without competition) and $\hat{\mathcal{D}}_{r-compete}$ (with competition) considering all the 531 users from the GPDS-960 dataset formerly employed in the SigNet training. For each user, 24 genuine signatures were inverted. For all the inversion experiments described

in this work, we used Adam for gradient optimization with learning rate given by 0.25 (YIN et al., 2020). To obtain the final result, we performed 9000 gradient updates for each data batch. The two final inverted datasets containing 12744 examples each provide a SigNet classification accuracy given by 99.40% when using deep inversion and accuracy given by 99.17% when using deep inversion with competition. Note that the competition scheme between SigNet and SigNet (Synthetic) models slightly degraded the overall accuracy compared to the case in which the examples were synthesized without competition. This result partially indicates that the competition term promotes a greater variability of generated inverted examples that may be out-of-distribution of the SigNet model, and thus expanding the inverted data distribution coverage.

4.4.1.2 Experimental Protocol for Training Models

In our experimental setup, we combine the distribution of inverted real signature data with the distribution of synthetic signature data using the proposed method. In order to obtain the characteristics of real signatures through knowledge distillation, we evaluate three different datasets: firstly, the GPDS-960 development dataset $\mathcal{D}_r$ as an oracle that allows measuring performance in the situation such that this dataset is available to support distillation. Secondly, we replace the GPDS-960 development dataset to the examples obtained through SigNet deep inversion with and without competition ($\hat{\mathcal{D}}_{r-compete}$ and $\hat{\mathcal{D}}_r$) to evaluate the performance when using these datasets as surrogates to support knowledge distillation. Thus, our intention is to perform an ablation experiment to compare the performance of the examples inverted without (Equation 4.1) and with competition (Equation 4.3).

Therefore, the real data is obtained from a development dataset $\mathcal{D}_r$ with segmentation detailed in Table 16. Besides, in order to obtain the characteristics of synthetic data, the GPDSsynthetic development set $\mathcal{D}_s$ with segmentation detailed in Table 17 is used for training. In both development sets, each user has 24 genuine signatures. With these training datasets, we conducted model optimization adopting the evaluated architectures: ResNet-152, Vision Transformer and the SigNet architecture as defined in Hafemann, Sabourin e Oliveira (2017a).

The adopted Vision Transformer follows the architectural configuration proposed in Dosovitskiy et al. (2021): we adopt patches of size 16×16 that are linearly embedded in representations of dimensionality $D = 768$ that feed a stacked architecture of $L = 12$ transformers encoders having $h = 12$ attention heads and hidden layer dimensionality given by 3072. In the

Table 16 – Segmentation for the GPDS-960 dataset following Hafemann, Sabourin e Oliveira (2017a).

Subset Name	Users	User Range
Development set $\hat{\mathcal{D}}_r$ or $\hat{\mathcal{D}}_{r-compete}$ or $\mathcal{D}_r$	531	351 - 881
Validation set $\mathcal{V}_{vr}$	50	301 - 350
Exploitation set $\mathcal{E}_{300}$	300	1 - 300

Table 17 – Proposed segmentation for the GPDSsynthetic dataset.

Subset Name	Users	User Range
Development set $\mathcal{D}_s$	7950	2001 - 9950
Validation set $\mathcal{V}_{vs}$	50	9951 - 10000
Exploitation set $\mathcal{E}_{150S}$	150	1 - 150
Exploitation set $\mathcal{E}_{300S}$	300	1 - 300

ResNet-152 and Vision Transformer architectures, we added a classification head that follows the same original configuration of the architecture adopted by SigNet: we append two fully connected layers of size 2048 to the end of the architectures. In this way, all models studied in this work provide feature representations of size 2048 that can be used in the training of classifiers for signature verification.

The studied losses are minimized with Stochastic Gradient Descent with Nesterov Momentum with momentum factor given by 0.9. We trained models with an initial learning rate of 10^{-3} which is divided by 10 every 20 epochs (following Hafemann, Sabourin e Oliveira (2017a)). In the first task, the adopted batch size is 64, we use balanced data batches in which 32 elements are obtained from the real dataset and the other 32 from the synthetic dataset. As the second task is contrastive based, we allocated 2 signature samples per user in each batch specifically in the second task. In this case, the adopted batch size is 256 with 128 synthetic examples and 128 real examples, thus, 128 different users are allocated per batch. Specifically in the large vision architectures (ResNet-152 and Vision Transformer), due to memory constraints in the second task, we use data batches of size 128, with 64 real examples and 64 synthetic examples.

In training, we used 90% of the total genuine signatures (real and synthetic ones respectively from the development sets $\mathcal{D}_r$ and $\mathcal{D}_s$) to optimize the model, whereas we employed the remaining 10% of the total genuine signatures for monitoring the early stopping of the training process. We respectively call the real and synthetic genuine signature monitoring sets as $\mathcal{M}_r$ and $\mathcal{M}_s$. Training is performed up to a maximum of 60 epochs in the case of the SigNet and ResNet-152 architectures. Specifically in the case of the Vision Transformer architecture, we

found better model convergence when the training process is conducted up to a maximum of 120 epochs.

In the first task, the final obtained model is the one that generates the best accuracy performance on the classification of the monitoring sets $\mathcal{M}_r$ and $\mathcal{M}_s$, considering an average between real and synthetic performance. As probability-based and contrastive distance-based loss functions are simultaneously optimized in the second task, in this situation the final obtained model is the one that provides the best performance considering an average of the classification accuracy of the $\mathcal{M}_r$, $\mathcal{M}_s$ signature examples by the model; and the accuracy that considers whether the first nearest neighbors of the $\mathcal{M}_r$, $\mathcal{M}_s$ signature examples correspond to the respective users of the $\mathcal{D}_r$, $\mathcal{D}_s$ development sets. Besides, it is important to note that neither real nor synthetic skilled forgeries are used for training.

Given this, the trained models are evaluated in terms of the Equal Error Rate (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) using synthetic and real validation sets in writer-dependent and writer-independent approaches. A disjoint set of real signatures $\mathcal{V}_{vr}$ (Table 16) from the GPDS-960 dataset in conjunction with a disjoint set of synthetic signatures $\mathcal{V}_{vs}$ (Table 17) from the GPDSsynthetic dataset are employed for evaluating the performance of trained models with different λ_p hyper-parameter (Equation 4.8) configurations (we adopted $\lambda_p = \{0.0, 0.1, \ldots, 1.9, 2.0\}$ in writer-dependent approach, and $\lambda_p = \{0.0, 0.1, \ldots, 1.0\}$) in writer-independent approach) in order to obtain unbiased choices of best λ_p hyper-parameters. In this validation scenario, the objective is to determine the optimal value of λ_p that maintains a balanced verifiability of real and synthetic signatures. In the first task, we discover the best λ_p value. In the second task, we keep the previously found hyper-parameter value fixed, aiming to focus on the contrastive adjustment of the model. Furthermore, it is important to mention that we search for the best λ_p hyper-parameters using SigNet architecture, which are later employed in the training of other evaluated architectures. This way, we can obtain a comprehension of how the use of these hyper-parameters generalizes to different student architectures.

Given the best choices for λ_p hyper-parameters, we apply the combined feature space learned with the real $\mathcal{D}_r$ and synthetic $\mathcal{D}_s$ development sets in the extraction of features for disjoint exploitation sets ($\mathcal{E}_{150S}$, $\mathcal{E}_{300S}$, $\mathcal{E}_{300}$) containing a large number of users and other datasets in order to measure the generalization performance of the trained models in a transfer learning scenario. Thus, we evaluated writer-dependent and writer-independent classifiers using the synthetic dataset GPDSsynthetic (GPDS-150S and GPDS-300S); on the Western script

real datasets GPDS-960 (GPDS-300), CEDAR (KALERA; SRIHARI; XU, 2004) and MCYT-75 (ORTEGA-GARCIA et al., 2003); and on the non-Western script real datasets BHSig-H (Hindi) (PAL et al., 2016) and BHSig-B (Bengali) (PAL et al., 2016) datasets. The experimental protocol for the training of the classifiers is described as follows.

4.4.1.3 Experimental Protocol for Writer-Dependent Verification

For writer-dependent verification of the validation sets $\mathcal{V}_{vr}$ and $\mathcal{V}_{vs}$ we followed the protocol defined in Hafemann, Sabourin e Oliveira (2017a): we trained a support vector machine classifier for each user of the validation set. To this end, 12 genuine reference signatures from the verified user are used as positive examples, whereas 14 randomly obtained signatures from each user of the respective development set are used as negative examples. We tested each classifier using 10 remaining genuine signatures, 10 skilled forgeries and 10 random forgeries (randomly obtained from other users).

Table 18 – Separation of the training and testing sets for the evaluation of each dataset in Writer-Dependent approach.

Dataset		Training set		Testing set
Name	Users	Genuine	Random Forgeries	
GPDS-150S	150	$r \in \{1, 2, 3, 5, 10, 12\}$	$r \times 149$	10 genuine, 10 skilled
GPDS-300S	300	$r \in \{1, 2, 3, 5, 10, 12\}$	$r \times 299$	10 genuine, 10 skilled
GPDS-300	300	$r \in \{1, 2, 3, 5, 10, 12\}$	$r \times 299$	10 genuine, 10 skilled
CEDAR	55	$r \in \{1, 2, 3, 5, 10, 12\}$	$r \times 54$	10 genuine, 10 skilled
MCYT-75	75	$r \in \{1, 2, 3, 5, 10\}$	$r \times 74$	5 genuine, 15 skilled
BHSig-B	100	$r \in \{1, 2, 3, 5, 10, 12\}$	$r \times 99$	10 genuine, 10 skilled
BHSig-H	160	$r \in \{1, 2, 3, 5, 10, 12\}$	$r \times 159$	10 genuine, 10 skilled

In order to measure the generalization performance in a writer-dependent approach on other datasets, we followed a protocol based on those adopted in Hafemann, Sabourin e Oliveira (2017a), Viana et al. (2023), Zheng et al. (2021), Tsourounis et al. (2022): we used r genuine reference signatures of a given user from the dataset as positive examples, whereas we obtained r genuine signatures from the other users (random forgeries) in the same dataset as negative examples. The number r of signatures used in training for each dataset is listed in Table 18. By utilizing this training set, we trained a support vector machine for each user and measured the performance over the testing set: a disjoint sample of remaining genuine signatures and skilled forgeries of each user are used for testing. We verified these signatures

and the number of tested signatures is listed in Table 18).

4.4.1.4 Experimental Protocol for Writer-Independent Verification

For writer-independent verification of the validation sets $\mathcal{V}_{vr}$ and $\mathcal{V}_{vs}$, we trained a support vector machine classifier with a set of dissimilarity vectors obtained from the signatures of all users in the development set, following the protocol defined by Souza, Oliveira e Sabourin (2018). Hence, for each user of the development set, we randomly selected 12 genuine signatures, which are pairwise combined, obtaining vectors of *within class*. Besides that, for each user of the development set, 11 genuine signatures are combined against 6 randomly selected genuine signatures from other users (random forgeries) in the development set. These are the dissimilarity vectors of *between class*. We tested the writer-independent classifier in face of each user of the validation set (50 users). In this case, for each user of the validation set, we used 12 genuine signatures as reference signatures within the *max* fusion function, and the classifier is tested using 10 remaining genuine signatures, 10 skilled forgeries from the user and 10 random forgeries (randomly obtained from other users).

Table 19 – Separation into training and testing sets in Writer-Independent approach for each evaluated dataset.

Dataset		Training set		Testing set	
Name	#Users	Negative (*between*) class	Positive (*within*) class	Reference set	Questioned set
GPDS-150S	150	Distances between the 5 signatures for each user and 3 random signatures from other users (119,250 samples).	Distances between the 6 signatures for each user (119,250 samples).	$r \in \{1, 2, 3, 5, 10, 12\}$	10 genuine, 10 skilled
GPDS-300S	300	Distances between the 5 signatures for each user and 3 random signatures from other users (119,250 samples).	Distances between the 6 signatures for each user (119,250 samples).	$r \in \{1, 2, 3, 5, 10, 12\}$	10 genuine, 10 skilled
GPDS-300	300	Distances between the 11 signatures for each user and 6 random signatures from other users (38,346 samples).	Distances between the 12 signatures for each user (38,346 samples).	$r \in \{1, 2, 3, 5, 10, 12\}$	10 genuine, 10 skilled
CEDAR	55	Distances between the 13 signatures for each user and 7 random signatures from other users (2,457 samples).	Distances between the 14 signatures for each user (2,457 samples).	$r \in \{1, 2, 3, 5, 10, 12\}$	10 genuine, 10 skilled
MCYT-75	75	Distances between the 9 signatures for each user and 5 random signatures from other users (1,665 samples).	Distances between the 10 signatures for each user (1,665 samples).	$r \in \{1, 2, 3, 5, 10\}$	5 genuine, 15 skilled
BHSig-B	100	Distances between the 13 signatures for each user and 7 random signatures from other users (4,550 samples).	Distances between the 14 signatures for each user (4,550 samples).	$r \in \{1, 2, 3, 5, 10, 12\}$	10 genuine, 10 skilled
BHSig-H	160	Distances between the 13 signatures for each user and 7 random signatures from other users (7,280 samples).	Distances between the 14 signatures for each user (7,280 samples).	$r \in \{1, 2, 3, 5, 10, 12\}$	10 genuine, 10 skilled

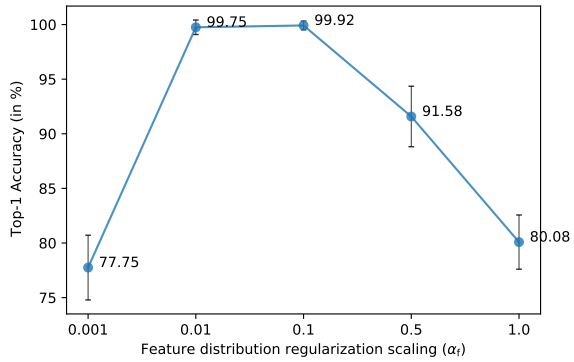
In order to measure the generalization performance of models in a writer-independent approach on other datasets we followed the protocol proposed in Souza, Oliveira e Sabourin (2018). In the case of GPDS-150S, GPDS-300S and GPDS-300 datasets, writer-independent classifiers are trained with a set of dissimilarity vectors obtained from the signatures of the respective development set. In the case of measuring the performance of models with the CEDAR, MCYT-75, BHSig-B and BHSig-H datasets, we evaluated writer-independent classifiers through a cross-validation procedure. In this scenario, half of the users of the dataset are used as development set, whereas the other half are used as exploitation set. In the experiments, we performed a random split of the dataset in a $5 \times 2-$fold cross-validation. The number of examples used for training and testing of writer-independent classifiers for each dataset is summarized in Table 19.

4.4.2 Sensitivity Analysis

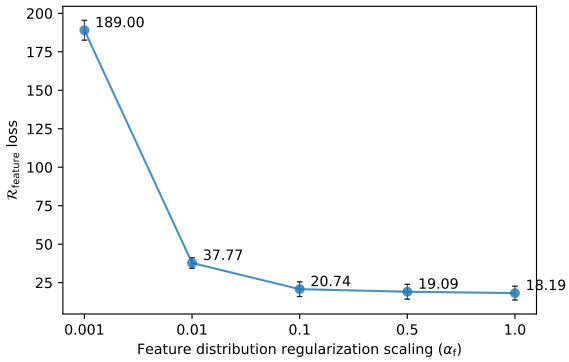
4.4.2.1 Grid-search of DeepInversion Hyper-parameters

We considered finding an α_{f} scaling factor that maintains the classification ability of the inverted examples by the SigNet. In the performed grid-search process, we monitored the $\mathcal{R}_{\mathsf{feature}}$ loss (Equation 4.1). This loss term indicates how the feature statistics of the inverted examples are close to the running average statistics stored in the SigNet batch normalization layers. The Figures 21a and 21b respectively show the average Top-1 accuracy across batches of inverted examples and the levels of $\mathcal{R}_{\mathsf{feature}}$ loss. When the feature distribution regularization scaling factor is high ($\alpha_{\mathrm{f}} = 1.0$), this generates an overfitting with the batch normalization layers statistics, providing a lower $\mathcal{R}_{\mathsf{feature}}$ loss but with an undesired low accuracy. In the opposite situation, adopting a low feature distribution regularization scaling factor ($\alpha_{\mathrm{f}} = 0.001$) implies no match with these statistics, also providing a low accuracy. In this way, an in-between scaling factor $\alpha_{\mathrm{f}} = 0.1$ balances the two previously mentioned opposite situations so that a high accuracy of inverted examples is obtained, maintaining the match with the GPDS-960 distribution.

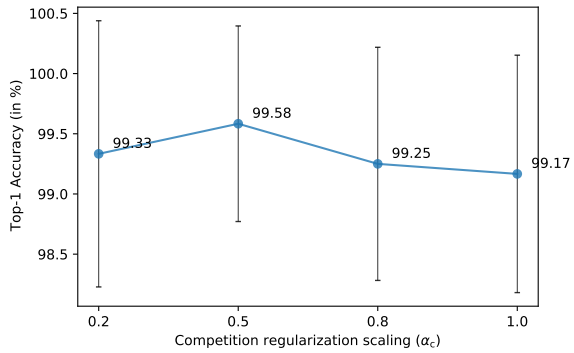
Regarding the additional competition term $\mathcal{R}_{\mathsf{compete}}$ (Equation 4.3), increasing the competition scaling factor α_{c} enforces obtaining a smaller value of the $\mathcal{R}_{\mathsf{compete}}$ term loss (Figure 21d), indicating a more significant disagreement between the SigNet and SigNet (Synthetic) outputs when propagating inverted examples. This competition term leads to further inverted



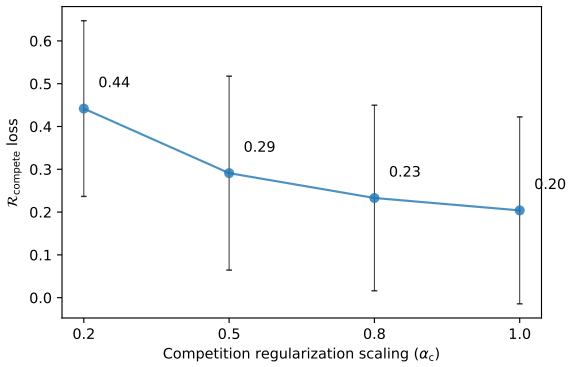
(a) Average Top-1 accuracy across batches of inverted examples.



(b) Average feature distribution regularization loss $\mathcal{R}_{\text{feature}}$ levels across batches of inverted examples.



(c) Average Top-1 accuracy across batches of inverted examples with competition.



(d) Average competition regularization loss $\mathcal{R}_{\text{compete}}$ levels across batches of inverted examples with competition.

Figure 21 – The level of the investigated losses and associated Top-1 accuracy for inverted validation sets with 50 handpicked fixed users, such that each user has 24 genuine examples.

examples outside the SigNet (Synthetic) distribution, aiming to increase the variability of inverted examples. However, choosing a high competition scale factor decreases the obtained Top-1 accuracy in the classification of inverted examples (Figure 21c), which indicates an excessive disagreement of the inverted examples with the SigNet distribution. Thus, a middle ground competition coefficient ($\alpha_c = 0.5$) appears to introduce a suitable level of variability in the inverted examples compared to other configurations with competition enabled.

4.4.2.2 Properties of the Inverted Data

In Figure 22 is shown four obtained inverted genuine signature examples (across the lines) for each one of eight randomly sampled users (across the columns). As can be observed, inverted examples generated from the same user appear to maintain a slight intra-class variation, whereas examples from different users have more significant inter-class variation. Besides, the

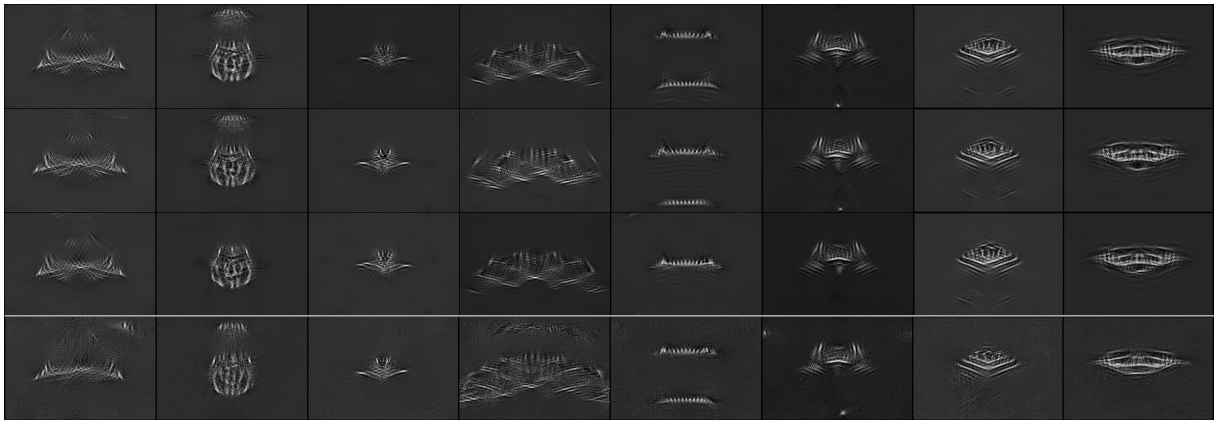
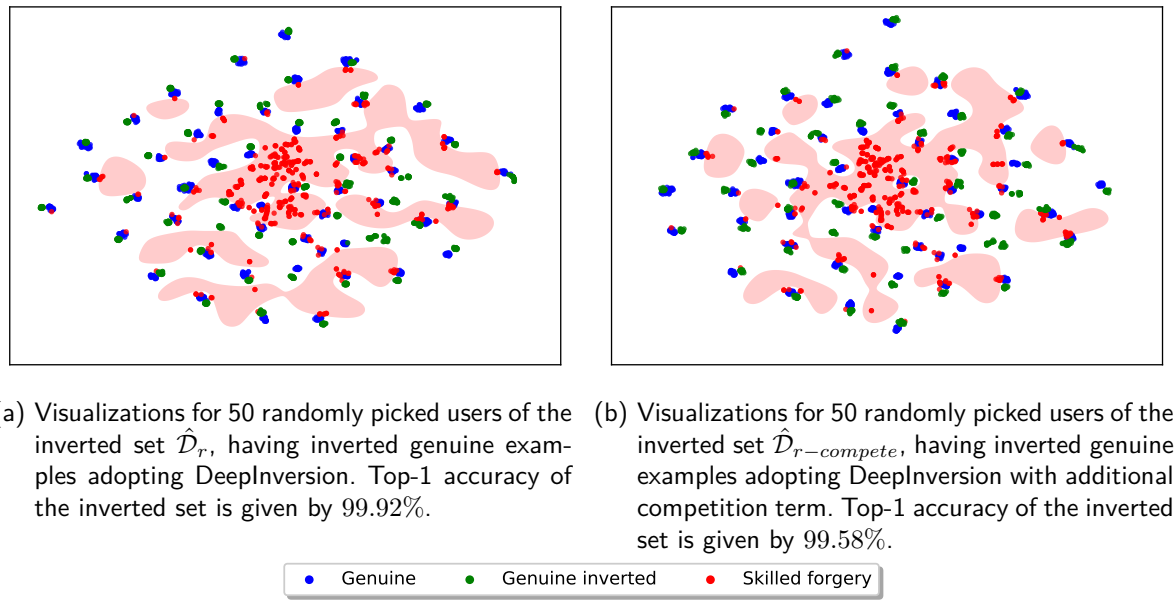


Figure 22 – Four inverted genuine signature examples (across the lines) for each one of eight randomly sampled users (across the columns). The last line shows examples generated using the additional competition term.

generated examples appear to have a darker background and present shapes in gray-scale, generally more positioned in the center. The presented shapes may be related to the writing intensity and shape of the users' signatures and may represent motor aspects concerning movements the users perform when producing signature examples. However, note that the inverted examples do not explicitly show the original traits of the GPDS-960 signatures. As inverted examples are not visually the same as the original examples, it is possible to maintain a data protection level when publicly sharing the inverted dataset.



(a) Visualizations for 50 randomly picked users of the inverted set $\hat{\mathcal{D}}_r$, having inverted genuine examples adopting DeepInversion. Top-1 accuracy of the inverted set is given by 99.92%.

(b) Visualizations for 50 randomly picked users of the inverted set $\hat{\mathcal{D}}_{r-compete}$, having inverted genuine examples adopting DeepInversion with additional competition term. Top-1 accuracy of the inverted set is given by 99.58%.

Figure 23 – Obtained t-SNE visualizations for feature representations of genuine inverted examples from $\hat{\mathcal{D}}_r$ and $\hat{\mathcal{D}}_{r-compete}$; real genuine examples and real skilled forgeries from the respective original users of the GPDS-960 development dataset $\mathcal{D}_r$. Critical regions where skilled forgeries are located are highlighted in light red. To generate visualizations of critical regions, we trained a support vector machine with a radial basis function kernel for a binary classification problem. We considered skilled forgeries as the positive class, and genuine and DeepInversion-generated examples as negative examples. Given this, we plotted the decision boundaries generated by separating these examples.

Furthermore, as illustrated in Figure 23, we generated t-SNE (MAATEN; HINTON, 2008) visualizations for the genuine inverted examples (in green) as well as real genuine examples (in blue) and real skilled forgeries (in red) for 50 randomly picked users from the two inverted sets $\hat{\mathcal{D}}_r$ and $\hat{\mathcal{D}}_{r-compete}$; and the GPDS-960 development set $\mathcal{D}_r$. The obtained visualizations for inverted signatures with deep inversion and additional competition term are respectively illustrated in Figure 23a and Figure 23b. As can be observed, in both approaches, the inverted genuine signature clusters tend to maintain inter-class separation with intra-class variability. It is worth noting that each cluster of inverted examples tends to be positioned close to the respective original cluster, indicating that the inverted samples keep the distribution of the original GPDS-960 development set data. Finally, it is also possible to observe a region that concentrates representations of skilled forgeries, which tend to be clustered together as a whole, moving away from the clusters of genuine and inverted signatures.

We analyzed the entropy of inverted signature examples; and the entropy of real genuine signature and skilled forgery examples to complement the visualizations presented in Figure 23 and reinforce the hypothesis that skilled forgeries tend to be generally positioned in a contained region of the representation space. The entropy $\mathrm{H}(\hat{x})$ is a measure of the amount of uncertainty (MACêDO et al., 2022) generated in the classification of a given example $\hat{x}$, defined by:

$$\mathrm{H}(\hat{x}) = -\sum_{u \in \mathcal{D}_r = 1}^{531} p_{\mathsf{SigNet}_{(u)}}(\hat{x}) \log p_{\mathsf{SigNet}_{(u)}}(\hat{x}) \tag{4.10}$$

The average entropy of the genuine signature and skilled forgery examples from the GPDS-960 development set and the average entropy of the inverted genuine examples are listed in Table 20. As expected, the original examples from the GPDS-960 development set have entropy close to zero. It occurs because the SigNet model is originally trained with such examples which generate probabilities much closer to the limit, providing high certainty in the classification. On the contrary, the GPDS-960 skilled forgeries present a very high entropy value, implying that skilled forgeries are in a region of uncertainty: the SigNet model does not assign a specific class to the skilled forgeries examples, but it generates an uncertain class assignment. This result indicates that the SigNet model learns the global characteristics of genuine signatures, holistically differentiating them from the global attributes of skilled forgeries. Thus, the skilled forgery representations are placed in a region of uncertainty within the representation space.

Concerning the inverted genuine examples using deep inversion, these produce a low entropy

Table 20 – Average entropy of genuine signatures, skilled forgeries, and inverted genuine signatures. Genuine signatures and skilled forgeries were obtained from the 531 users of the GPDS-960 development set. Inverted genuine signatures were obtained through deep inversion of examples from the same 531 users. The entropy measures the uncertainty across a probability distribution. In this scenario, the maximum value of entropy is $\log_{10} 531 = 2.7250$, as 531 is the number of users contained in the development set.

Source	**Entropy**
GPDS-960 genuine signatures in $\mathcal{D}_r$	0.0343 ($\pm$ 0.0598)
GPDS-960 skilled forgeries in $\mathcal{D}_r$	1.1241 ($\pm$ 0.5556)
Inverted genuine signatures in $\hat{\mathcal{D}}_r$	0.2426 ($\pm$ 0.4524)
Inverted genuine signatures with competition in $\hat{\mathcal{D}}_{r-compete}$	0.2335 ($\pm$ 0.4558)

providing network activations associated to the real GPDS-960 genuine signatures. Therefore, inverted examples are in a region of classification certainty and can be employed to distill knowledge about real genuine signatures through the SigNet model. The inverted examples using competition provide slightly better convergence of probability outputs with slightly lower entropy. The reason for this is explained by the promotion of a greater intra-class variability introduced by competition.

4.4.2.3 Signature Verification System Design

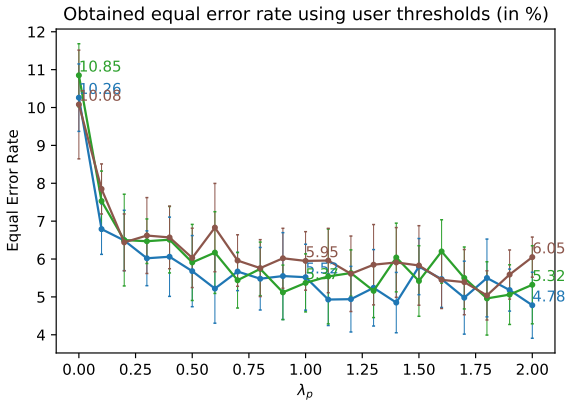
We adopted the GPDS-960 dataset as an oracle measure concerning the distillation process in order to delimit a performance baseline in the case that this dataset is available. In this way, we compared the performance of the models obtained by distillation using the alternative datasets based on SigNet deep inversion instead of the original dataset. Besides, we considered the situations in which student models are initialized from scratch (Figures 24a, 24b, 25a, 25b) as well as from the SigNet weights (Figures 24c, 24d, 25c, 25d).

We vary the λ_p hyper-parameter aiming to verify the obtained effect when the divergence between the distributions of real and synthetic data sources is minimized. As shown in Figure 24a, when the λ_p value is increased in the writer-dependent approach, the error rate in the real validation set decreases by approximately less than half. On the other hand, the error rate in the synthetic validation set increases (Figure 24b). This same result is observed when the student model is optimized from the original SigNet weights (Figures 24c and 24d).

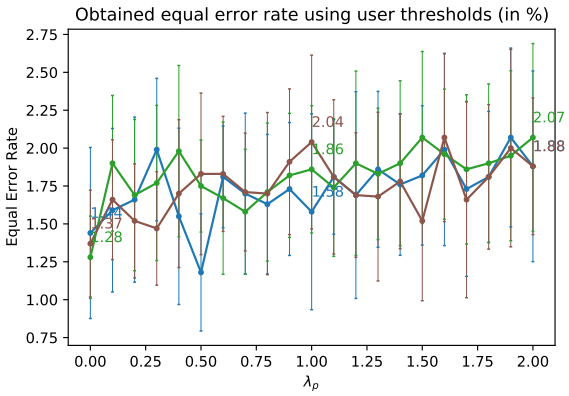
Analogous to the writer-dependent verification case, when the λ_p value is increased, the writer-independent error rate in the real validation set decreases by approximately less than half (as shown in Figures 25a and 25c). However, the writer-independent error rate in the

synthetic validation set slightly increases (Figure 25b) when the student models are initialized from scratch and presents a more stable variation when the student models are initialized from the SigNet weights (Figure 25d).

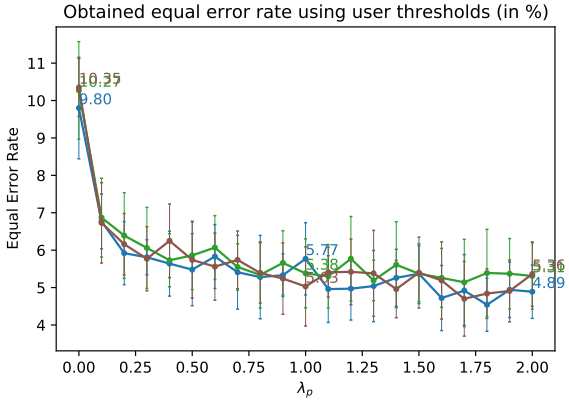
Given this scenario, we observe a trade-off in verification performance concerning the obtained equal error rate on the real validation set from the GPDS-960 dataset and the synthetic validation set from the GPDSsynthetic dataset. It is important to mention that this effect is observed not only when the GPDS-960 dataset is used but also when the datasets obtained through SigNet deep inversion are used to guide the distillation.



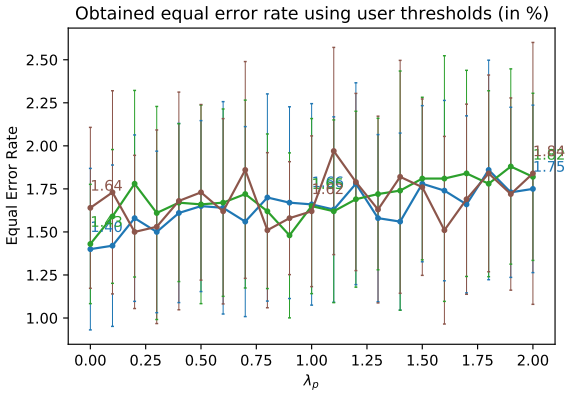
(a) Real validation set $\mathcal{V}_{vr}$ writer-dependent performance with features extracted from students trained from scratch.



(b) Synthetic validation set $\mathcal{V}_{vs}$ writer-dependent performance with features extracted from students trained from scratch.



(c) Real validation set $\mathcal{V}_{vr}$ writer-dependent performance with features extracted from students initialized with SigNet weights.



(d) Synthetic validation set $\mathcal{V}_{vs}$ writer-dependent performance with features extracted from students initialized with SigNet weights.

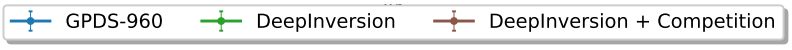
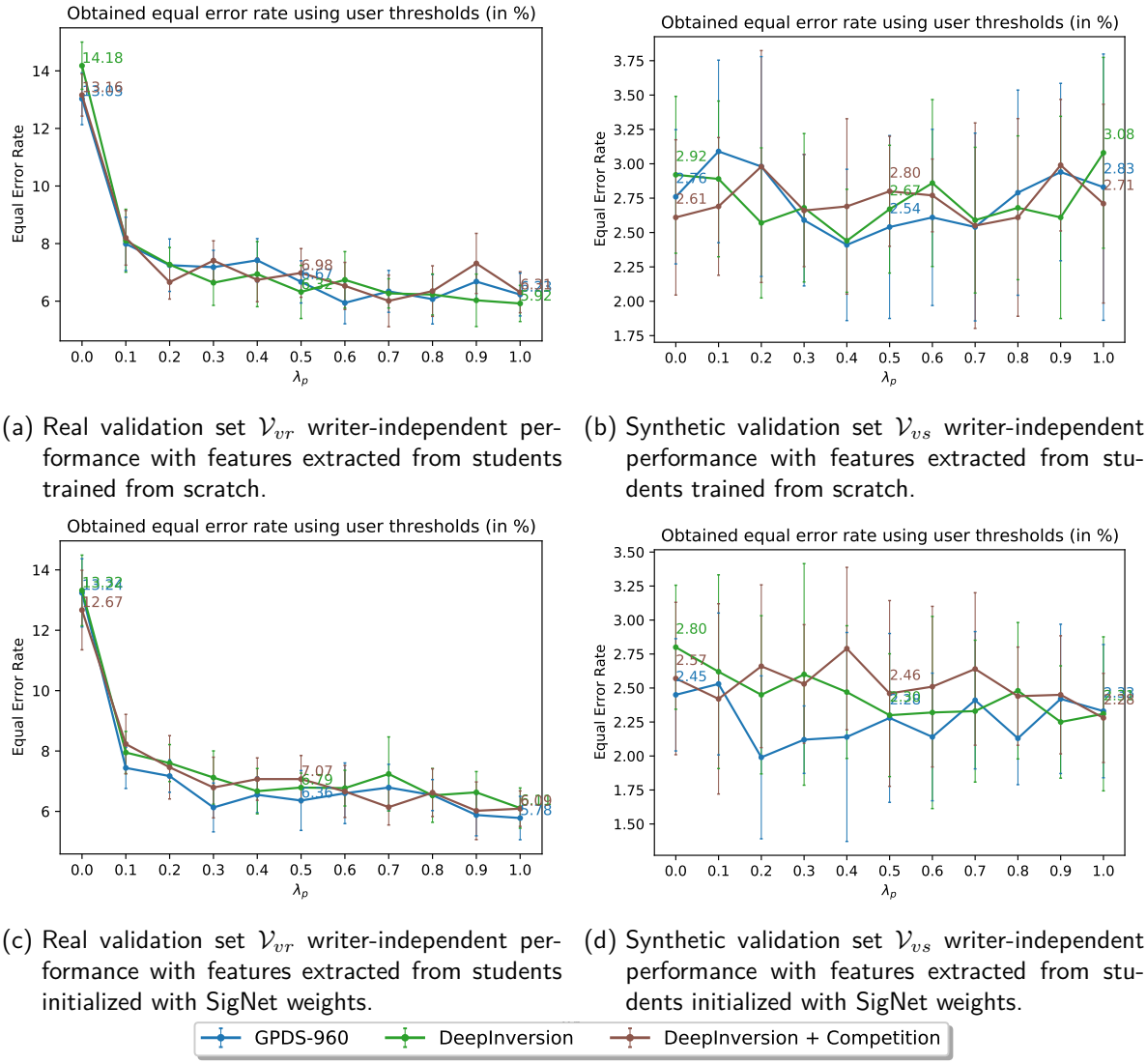


Figure 24 – Real $\mathcal{V}_{vr}$ and synthetic $\mathcal{V}_{vs}$ validation sets Writer-Dependent performance with features extracted from students trained with the proposed continual loss (Equation 4.7) using three different distillation datasets: GPDS-960 examples, DeepInversion examples and DeepInversion examples obtained with competition.

4.4.2.4 *Summarizing the Best λ_p Configurations*

For each evaluated dataset in the distillation process, we describe a comparison regarding the best student model generated with the λ_p hyper-parameter that provides the best average performance considering the synthetic $\mathcal{V}_{vs}$ and real $\mathcal{V}_{vr}$ validation sets.

RQ5) How can we obtain a more suitable signature representation with models trained using synthetic and inverted data in a situation where the access to the GPDS-960 dataset is unavailable? In the Tables 21 and 22 are listed the best λ_p hyper-parameter configurations associated with the lowest obtained error rate regarding the studied data sources for distillation in the writer-dependent and writer-independent approaches. We



(a) Real validation set $\mathcal{V}_{vr}$ writer-independent performance with features extracted from students trained from scratch.

(b) Synthetic validation set $\mathcal{V}_{vs}$ writer-independent performance with features extracted from students trained from scratch.

(c) Real validation set $\mathcal{V}_{vr}$ writer-independent performance with features extracted from students initialized with SigNet weights.

(d) Synthetic validation set $\mathcal{V}_{vs}$ writer-independent performance with features extracted from students initialized with SigNet weights.

Figure 25 – Real $\mathcal{V}_{vr}$ and synthetic $\mathcal{V}_{vs}$ validation sets Writer-Independent performance with features extracted from students trained with the proposed continual loss (Equation 4.7) using three different distillation datasets: GPDS-960 examples, DeepInversion examples and DeepInversion examples obtained with competition.

compared the performance of the proposed method on the real $\mathcal{V}_{vr}$ and synthetic $\mathcal{V}_{vs}$ validation sets in relation to the previous works, SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) and SigNet (Synthetic) (VIANA et al., 2023). As can be seen in the Table 21, the proposed method has a better average performance than previous methods when considering the average writer-dependent performance between real and synthetic data sources. Besides in the Table 22, all distillation-based models provide a considerably lower average error rate considering synthetic and real validation data than previous SigNet-based models (HAFEMANN; SABOURIN; OLIVEIRA, 2017a; VIANA et al., 2023) in the writer-independent verification scenario. It is worth noting that regardless of the dataset used to guide the distillation process (GPDS-960, DeepInversion or DeepInversion with competition), all average performance results are better than previous SigNet based models in both verification approaches. This indicates that the proposed method has greater robustness concerning different types of data sources.

Furthermore, as can be seen in the Table 21, when considering the $EER_{skilled}$ metric, the models obtained from deep inversion provided an average writer-dependent performance that is only slightly worse than those obtained with the original data. However, by adopting the data

Table 21 – Average Writer-Dependent performance of the proposed method on the real $\mathcal{V}_{vr}$ and synthetic $\mathcal{V}_{vs}$ validation sets in relation to previous SigNet based models.

		$\mathcal{V}_{vr}$ **Real Performance**		$\mathcal{V}_{vs}$ **Synthetic Performance**		**Average Performance**	
Model	**Initialization**	$EER_{skilled}$	EER_{random}	$EER_{skilled}$	EER_{random}	$EER_{skilled}$	EER_{random}
Previous models:							
SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a)	-	3.63 (± 0.77)	0.02 (± 0.06)	18.69 (± 1.27)	5.65 (± 1.04)	11.16 (± 7.53)	2.83 (± 2.81)
SigNet (Synthetic) (VIANA et al., 2023)	-	12.17 (± 0.99)	0.02 (± 0.06)	2.87 (± 0.62)	0.00 (± 0.00)	7.52 (± 4.65)	0.01 (± 0.01)
Finetuning of previous models:							
SigNet finetuned with synthetic data	-	11.33 (± 1.17)	0.03 (± 0.06)	1.52 (± 0.47)	0.00 (± 0.00)	6.43 (± 4.91)	0.02 ± (0.02)
SigNet (Synthetic) finetuned with real data	-	5.85 (± 0.86)	0.00 (± 0.00)	6.93 (± 1.02)	0.72 (± 0.31)	6.39 (± 0.54)	0.36 (± 0.36)
Distillation based models studied in this work:							
Oracle: GPDS-960	Random ($\lambda_p = 1.4$)	4.85 (± 0.80)	0.00 (± 0.00)	1.76 (± 0.47)	0.00 (± 0.00)	3.31 (± 1.55)	0.00 (± 0.00)
	SigNet ($\lambda_p = 1.8$)	4.54 (± 0.71)	0.00 (± 0.00)	1.86 (± 0.64)	0.00 (± 0.00)	**3.20 (± 1.34)**	**0.00 (± 0.00)**
DeepInversion	Random ($\lambda_p = 1.8$)	4.96 (± 0.97)	0.00 (± 0.00)	1.90 (± 0.52)	0.00 (± 0.00)	3.43 (± 1.53)	0.00 (± 0.00)
	SigNet ($\lambda_p = 1.3$)	5.19 (± 0.54)	0.00 (± 0.00)	1.72 (± 0.44)	0.00 (± 0.00)	3.46 (± 1.74)	0.00 (± 0.00)
DeepInversion with Competition	Random ($\lambda_p = 1.8$)	5.04 (± 0.65)	0.00 (± 0.00)	1.81 (± 0.48)	0.00 (± 0.00)	3.43 (± 1.61)	0.00 (± 0.00)
	SigNet ($\lambda_p = 1.7$)	4.70 (± 1.00)	0.00 (± 0.00)	1.69 (± 0.55)	0.00 (± 0.00)	**3.20 (± 1.50)**	**0.00 (± 0.00)**

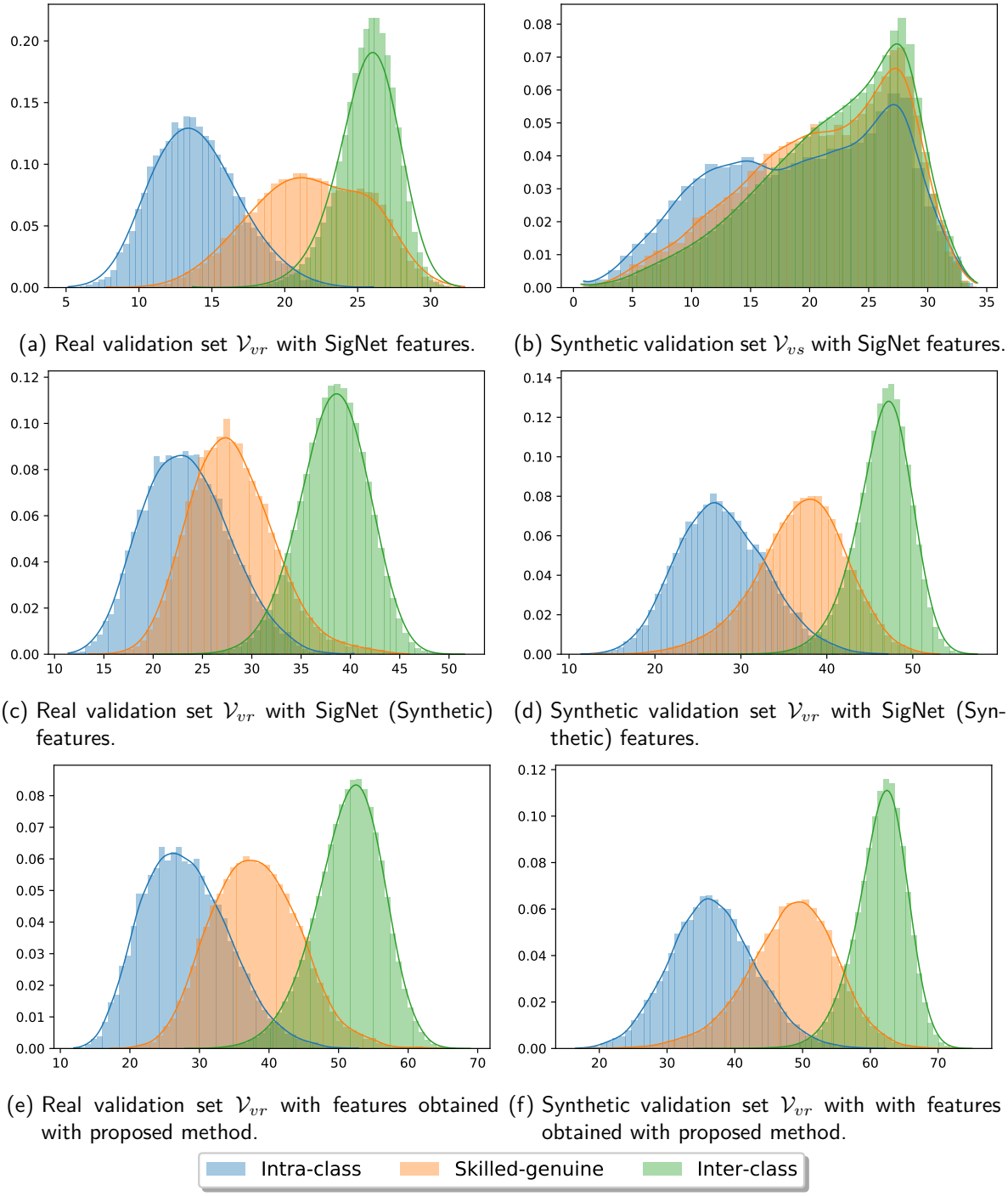
Table 22 – Average Writer-Independent performance of the proposed method on the real $\mathcal{V}_{vr}$ and synthetic $\mathcal{V}_{vs}$ validation sets in relation to previous SigNet based models.

		$\mathcal{V}_{vr}$ **Real Performance**		$\mathcal{V}_{vs}$ **Synthetic Performance**		**Average Performance**	
Model	**Initialization**	$EER_{skilled}$	EER_{random}	$EER_{skilled}$	EER_{random}	$EER_{skilled}$	EER_{random}
Previous works:							
SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a)	-	3.79 (± 0.56)	0.00 (± 0.00)	18.29 (± 1.60)	5.23 (± 1.16)	11.04 (± 7.25)	2.62 (± 2.62)
SigNet (Synthetic) (VIANA et al., 2023)	-	14.73 (± 0.86)	0.07 (± 0.11)	3.37 (± 0.68)	0.00 (± 0.00)	9.05 (± 5.68)	0.04 (± 0.04)
Finetuning of previous models:							
SigNet finetuned with synthetic data	-	14.32 (± 0.80)	0.17 (± 0.14)	2.78 (± 0.58)	0.00 (± 0.00)	8.55 (± 5.77)	0.09 ± (0.09)
SigNet (Synthetic) finetuned with real data	-	7.37 (± 0.91)	0.00 (± 0.00)	7.03 (± 0.81)	0.12 (± 0.15)	7.20 (± 0.17)	0.06 (± 0.06)
Distillation based models studied in this work:							
Oracle: GPDS-960	Random ($\lambda_p = 0.6$)	5.94 (± 0.73)	0.00 (± 0.00)	2.61 (± 0.64)	0.00 (± 0.00)	4.28 (± 1.67)	0.00 (± 0.00)
	SigNet ($\lambda_p = 1.0$)	5.78 (± 0.72)	0.00 (± 0.00)	2.33 (± 0.49)	0.00 (± 0.00)	**4.05 (± 1.73)**	**0.00 (± 0.00)**
DeepInversion	Random ($\lambda_p = 0.9$)	6.03 (± 0.91)	0.04 (± 0.08)	2.61 (± 0.74)	0.00 (± 0.00)	4.32 (± 1.71)	0.02 (± 0.02)
	SigNet ($\lambda_p = 1.0$)	6.11 (± 0.67)	0.02 (± 0.06)	2.31 (± 0.57)	0.00 (± 0.00)	4.21 (± 1.90)	0.01 (± 0.01)
DeepInversion with Competition	Random ($\lambda_p = 0.7$)	6.01 (± 0.90)	0.02 (± 0.06)	2.55 (± 0.75)	0.00 (± 0.00)	4.28 (± 1.73)	0.01 (± 0.01)
	SigNet ($\lambda_p = 1.0$)	6.09 (± 0.59)	0.02 (± 0.06)	2.28 (± 0.33)	0.00 (± 0.00)	**4.18 (± 1.91)**	**0.01 (± 0.01)**

obtained by deep inversion using competition, it is possible to obtain equivalent performance to the original GPDS-960 data in the situation such that the distillation process is initialized from the SigNet weights. This writer-dependent performance advantage of the data generated by deep inversion with competition can be partially explained by a better coverage of the inverted data distribution compared to the method without competition. In the writer-independent scenario (Table 22), the best average rates obtained with deep inversion data sources are only slightly higher than when using the GPDS-960 dataset to guide the distillation process. Besides, when considering EER_{random} in both verification approaches, all evaluated datasets provide better performance than the baselines. Therefore, the inverted data is a viable replacement for the original GPDS-960 data in situations where this data is unavailable and can effectively be used as a support in the process of transferring the characteristics of real signatures in the learning of new models.

Finally, it is important to note that the simple process of fine-tuning the SigNet and SigNet (Synthetic) models, respectively, with synthetic and real data, does not provide better average error rates than with the proposed method (as observed in Tables 21 and 22). This result

indicates that fine-tuning the pre-trained models tends to excessively skew the distribution of the models for the type of data source employed in fine-tuning.



(a) Real validation set $\mathcal{V}_{vr}$ with SigNet features.

(b) Synthetic validation set $\mathcal{V}_{vs}$ with SigNet features.

(c) Real validation set $\mathcal{V}_{vr}$ with SigNet (Synthetic) features.

(d) Synthetic validation set $\mathcal{V}_{vr}$ with SigNet (Synthetic) features.

(e) Real validation set $\mathcal{V}_{vr}$ with features obtained with proposed method.

(f) Synthetic validation set $\mathcal{V}_{vr}$ with with features obtained with proposed method.

Figure 26 – Obtained distributions of Euclidean distances between examples of the real and synthetic validation sets $\mathcal{V}_{vr}$ and $\mathcal{V}_{vs}$, considering representations obtained from the SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a), SigNet (Synthetic) (VIANA et al., 2023) models and student trained with the proposed continual loss (Equation 4.7) with $\lambda_p = 1.7$.

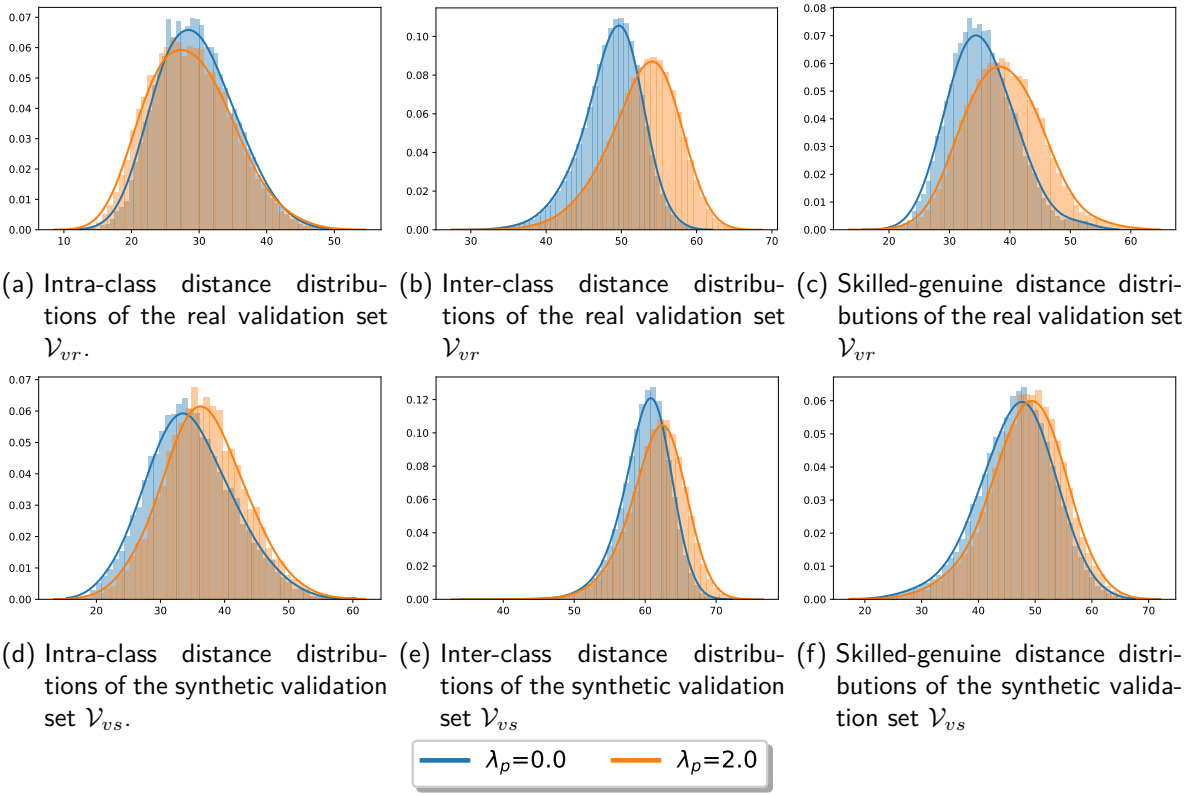
4.4.2.5 *Understanding the Influence of the Proposed Framework*

In Figure 26 is shown the distributions of intra-class, inter-class and skilled-genuine distances of the representations obtained for the synthetic $\mathcal{V}_{vs}$ and real $\mathcal{V}_{vr}$ validation sets using features obtained from SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a), SigNet (Synthetic) (VIANA et al., 2023) and with the proposed method.

RQ6) Is it feasible to rely solely on synthetic data from the GPDSsynthetic dataset for training models? As can be observed in Figure 26a, although the SigNet model correctly groups representations of real signatures, it provides incorrect cluster separation when applied to synthetic signature data (Figure 26b) and thus has substantial difficulty in separating synthetic skilled forgeries from poorly formed genuine signature clusters. On the other hand, the SigNet (Synthetic) is a model with a useful capability for grouping the different real users in the space (Figure 26c). However, the representations of skilled forgeries are positioned closer to the genuine representations. This result indicates that SigNet (Synthetic) can produce a good capacity to group different real or synthetic users (Figure 26d), but there is a lack in relation to the ability to provide acceptable separation of real skilled forgeries (Figure 26c). Therefore, it is unfeasible to rely solely on synthetic data from the GPDSsynthetic dataset for training models.

Given this scenario, by forcing the minimization of the divergence between synthetic and real representations in the student model with the proposed method, a separation force that moves real skilled forgeries away from their respective genuine clusters is created (as can be seen in Figure 26e). Despite this beneficial effect, by minimizing the divergence with the distribution of real data, a slight disturbance is generated in the representation space, providing less dense clusters (namely, with greater variance) of synthetic genuine representations that can justify the slight increase in the error rate in the synthetic validation set when λ_p is increased. Obtained effects are evidenced by the visualizations (Figure 27) of intra-class and inter-class distances; and between skilled forgeries and genuine signatures (skilled-genuine) representations of examples from the real and synthetic validation sets. As can be observed, although the desired increase in inter-class and skilled-genuine distances occurs in both validation sets when $\lambda_p = 2.0$ (Figures 27b, 27c, 27e, 27f), by emphasizing the effect of this hyper-parameter the intra-class distances of examples from the real validation set decrease (Figure 27a), but on the other hand, the intra-class distances of examples from the synthetic validation set increase (Figure 27d). This result justifies the occurrence of a verification performance trade-off

between the real and synthetic validation sets of the GPDS-960 and GPDSsynthetic datasets (initially reported in Section 4.4.2.3).



(a) Intra-class distance distributions of the real validation set $\mathcal{V}_{vr}$.

(b) Inter-class distance distributions of the real validation set $\mathcal{V}_{vr}$

(c) Skilled-genuine distance distributions of the real validation set $\mathcal{V}_{vr}$

(d) Intra-class distance distributions of the synthetic validation set $\mathcal{V}_{vs}$.

(e) Inter-class distance distributions of the synthetic validation set $\mathcal{V}_{vs}$

(f) Skilled-genuine distance distributions of the synthetic validation set $\mathcal{V}_{vs}$

Figure 27 – Obtained distributions of Euclidean distances between examples of the real and synthetic validation sets $\mathcal{V}_{vr}$ and $\mathcal{V}_{vs}$, considering representations obtained from students trained from scratch with the proposed continual loss (Equation 4.7) with $\lambda_p = 0.0$ and $\lambda_p = 2.0$. For this analysis, we used the representations obtained from student models trained from scratch, as such models have a more discrepant average performance. Wilcoxon paired signed-rank tests were performed with a 5% significance level to compare distributions with $\lambda_p = 0.0$ and $\lambda_p = 2.0$. The distribution of distances obtained with different λ_p is statistically different in all measurements listed in this figure.

4.4.3 Measuring Generalization Performance on Other Datasets

We evaluated the generalization of the best-obtained continual learning based models with the configurations highlighted in Tables 21 and 22. From this point on, in the writer-dependent verification context we call the best continual learning model trained with $\lambda_p = 1.7$ using inverted data as *SigNet-CL (DeepInversion + Competition)* and the best continual learning model trained with $\lambda_p = 1.8$ using the original GPDS-960 data as *SigNet-CL (GPDS-960)*. Besides, in the writer-independent verification context, we call the best continual learning model trained with $\lambda_p = 1.0$ using inverted data as *SigNet-CL (DeepInversion + Competition)* and the best continual learning model trained with $\lambda_p = 1.0$ using the original GPDS-960 data

as *SigNet-CL (GPDS-960)*. The models adjusted in the second contrastive task of the proposed framework (Figure 19) are named with the prefix MT (multi-task). Finally, models based on large vision architectures (ResNet-152 and Vision Transformers) are trained with the same hyper-parameter configuration and training datasets previously mentioned and are respectively called as MT-ResNet-152-CL and MT-ViT-CL. In the following three subsections, we discuss the research question: **RQ7) Can the characteristics of real and synthetic datasets be complemented using continual learning to offer a more balanced verification performance across diverse datasets?**

4.4.3.1 Writer-Dependent Generalization Performance

We observed the writer-dependent performance of the models provided by the first task of the proposed training framework (Figure 19). The writer-dependent generalization performance for the GPDS-150S, GPDS-300S, GPDS-300, CEDAR, MCYT-75, BHSig-B and BHSig-H datasets is respectively listed in Table 23.

As can be seen, the performance of the SigNet is clearly better when the model is tested on the GPDS-300 (a disjoint partition of the GPDS-960 dataset). On the other hand, the SigNet performance is noticeably very poor when tested on the GPDS-150S and GPDS-300S synthetic datasets, following results recently reported in the literature. Conversely, the proposed SigNet-CL based models perform much better on synthetic data from the GPDS-150S and GPDS-300S datasets.

The proposed SigNet-CL (DeepInversion + Competition) model has a better performance compared to the SigNet (Synthetic) model trained only with synthetic data in the face of GPDS-150S, GPDS-300S, GPDS-300, CEDAR, MCYT-75 and BHSig-H datasets. It is worth mentioning that unlike SigNet (Synthetic), SigNet-CL (DeepInversion + Competition) provides a more stable error rate when increasing the number of tested users from 150 to 300 in the GPDS-150S and GPDS-300S synthetic datasets. Furthermore, the proposed SigNet-CL (DeepInversion + Competition) model provides a lower average error on the synthetic GPDS-150S, GPDS-300S, and on the real CEDAR, MYCT-75, BHSig-B, BHSig-H datasets when compared to the version of SigNet-CL (GPDS-960) model with distillation performed over GPDS-960 data. This result indicates a better generalization capacity of models trained using DeepInversion-based data for knowledge distillation than the original GPDS-960 data.

When specifically analyzing the non-Western script datasets, the signature data from the

Table 23 – Writer-Dependent verification performance of the models studied in this work. Users of the GPDS-150S, GPDS-300S, GPDS-300, CEDAR, BHSig-B and BHSig-H datasets are tested using 12 reference signatures. Users of the MCYT-75 dataset are tested using 10 reference signatures.

	GPDS-150S	GPDS-300S	GPDS-300	CEDAR	MCYT-75	BHSig-B	BHSig-H
SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a)	21.68 (±0.74)	20.88 (±0.96)	3.24 (±0.25)	2.97 (±0.49)	3.23 (±0.65)	1.73 (±0.39)	4.64 (±0.40)
SigNet (Synthetic) (VIANA et al., 2023)	3.15 (±0.15)	3.53 (±0.23)	9.97 (±0.38)	3.77 (±0.44)	3.24 (±0.60)	3.71 (±0.55)	3.35 (±0.43)
SigNet-F (HAFEMANN; SABOURIN; OLIVEIRA, 2017a)	24.37 (±0.77)	23.64 (±0.41)	**1.67 (±0.10)**	4.76 (±0.70)	3.00 (±0.65)	1.97 (±0.33)	6.22 (±0.47)
SigNet-CL (GPDS-960)	2.77 (±0.32)	2.79 (±0.20)	4.39 (±0.35)	2.72 (±0.43)	3.39 (±0.39)	2.35 (±0.35)	3.66 (±0.37)
SigNet-CL (DeepInversion + Competition)	2.75 (±0.28)	2.71 (±0.20)	4.58 (±0.28)	**2.65 (±0.39)**	3.06 (±0.53)	2.33 (±0.49)	3.12 (±0.49)
MT-SigNet-CL (GPDS-960)	**2.27 (±0.15)**	2.36 (±0.16)	4.31 (±0.24)	2.75 (±0.43)	**2.95 (±0.51)**	1.87 (±0.35)	2.65 (±0.46)
MT-SigNet-CL (DeepInversion + Competition)	2.42 (±0.32)	**2.31 (±0.14)**	4.44 (±0.28)	2.94 (±0.41)	**2.95 (±0.73)**	**1.55 (±0.33)**	**2.43 (±0.41)**

BHSig-B (Bengali) dataset appears to be more aligned with the distribution of the GPDS-960 dataset. However, on the contrary, the SigNet model trained with GPDS-960 data delivers a higher error rate for the non-Western script dataset BHSig-H (Hindi). In this way, the proposed model SigNet-CL (DeepInversion + Competition) provides a more balanced performance between non-Western datasets, as it provides the lowest error rate in the BHSig-H dataset while delivering a more satisfactory performance than the Signet (Synthetic) model trained with synthetic data in the face of the BHSig-B dataset.

Given this scenario, we observed the effect on model generalization after applying contrastive adjustments to the SigNet-CL proposed models in the second task of the proposed training framework (Figure 19), with the obtained performance of MT-SigNet-CL models listed in Table 23. In writer-dependent verification, the adjusted MT-SigNet-CL (DeepInversion + Competition) model provides an improvement over the SigNet-CL (DeepInversion + Competition) model on the GPDS-150S, GPDS-300S, MCYT-75, BHSig-B, and BHSig-H datasets, obtaining a lower error rate on these datasets. Furthermore, the MT-SigNet-CL (DeepInversion + Competition) model performs better than SigNet and SigNet (Synthetic) models on all evaluated datasets except the GPDS-300 dataset. We believe that SigNet model is over-adjusted to the signature examples from the GPDS-960 sub-partitions. Therefore, although SigNet works well on the GPDS-300 dataset, this model has more difficulty to generalize to

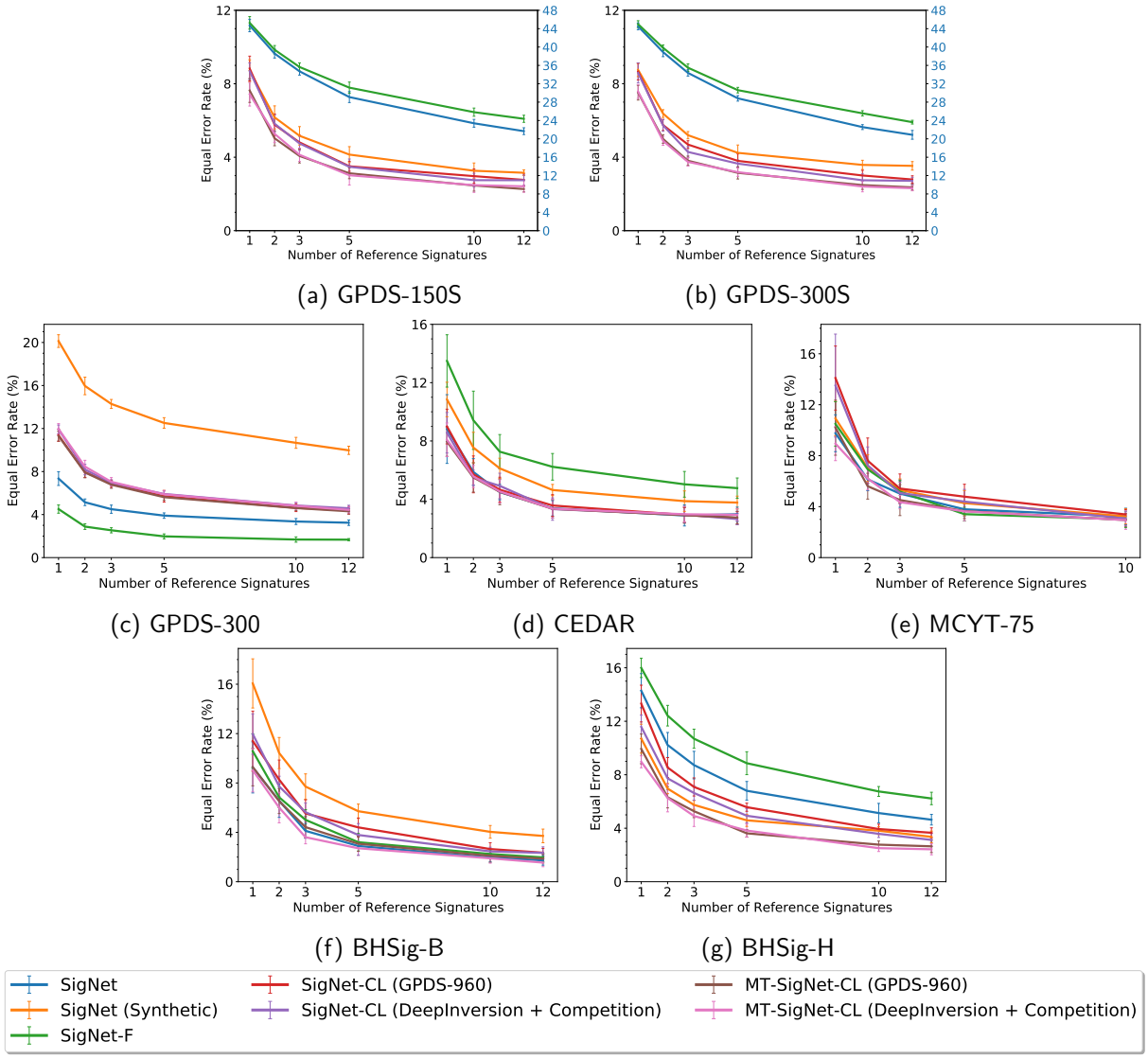


Figure 28 – Average performance of writer-dependent classifiers using user thresholds for the GPDS-150S, GPDS-300S, GPDS-300, CEDAR, MCYT-75, BHSig-B and BHSig-H datasets, as the number of reference genuine signatures (per user) available for training is varied. Given that SigNet and SigNet-F performance errors are much greater than the other models' errors specifically in the GPDS-150S and GPDS-300S datasets, the SigNet and SigNet-F performances are represented in a secondary axis positioned to the right in Figures 28a and 28b.

a range of more diverse datasets. On the other hand, the MT-Signet-CL models which are trained with real and synthetic data that come from different distributions, promote greater writer-dependent generalization on miscellaneous datasets. In conclusion, we observe that MT-SigNet-CL (DeepInversion + Competition) is a more robust model compared to the SigNet and SigNet (Synthetic) models when evaluating a broader range of real and synthetic datasets in a writer-dependent approach.

The Figure 28 shows the Equal Rate Error rate obtained when the number of reference signatures r used to train writer-dependent classifiers varies. For all models, there is a tendency for the error rate to decrease when more reference signatures are used.

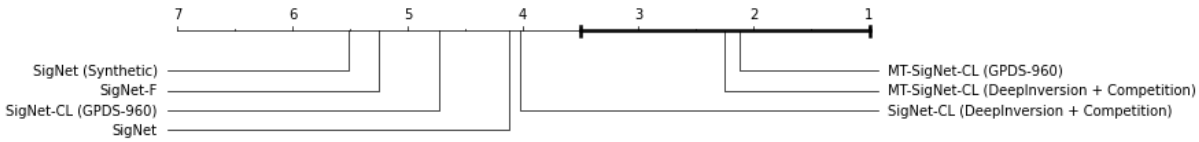


Figure 29 – Comparison of the proposed *MT-SigNet-CL (DeepInversion + Competition)* model against the other models studied in this work with the Bonferroni-Dunn post hoc tests. The models were tested in writer-dependent approach with user thresholds using signature data obtained from the GPDS-150S, GPDS-300S, GPDS-300, CEDAR, MCYT-75, HSig-B and BHSig-H datasets. The GPDS-150S dataset provides synthetic signatures of 150 users, the GPDS-300S dataset provides synthetic signatures of 300 users, the GPDS-300 dataset provides real signatures of 300 users, the CEDAR dataset provides real signatures of 55 users, the MCYT-75 dataset provides real signatures of 75 users, the HSig-B dataset provides real signatures of 100 users and the HSig-H dataset provides real signatures of 160 users. All the signatures obtained from these datasets are applied together in the performed statistical tests. All the models with ranks outside the marked interval are significantly different ($p < 0.05$) from the *MT-SigNet-CL (DeepInversion + Competition)* model.

We performed Friedman tests (DEMsAR, 2006) in order to statistically compare the studied models. For the all evaluated datasets, we generated different signature segmentations with different number of reference signatures r (according to the Table 18), totaling forty-one evaluated subjects. For each one of these subjects, different obtained feature representations (using the investigated models) are extracted in such a way that equal error rate is a repeated measure over each subject when varying each model. All performed tests indicated that all evaluated models are not equivalent (rejecting the null hypothesis) with 5% level of significance. Then, Bonferroni-Dunn post hoc tests were performed to compare the studied models against the proposed MT-SigNet-CL (DeepInversion + Competition) model (illustrated in Figure 29). All models with ranks outside the marked interval are significantly different ($p < 0.05$) from the MT-SigNet-CL (DeepInversion + Competition) model. Besides, the right-most placed models are better ranked regarding the writer-dependent verification performance. As observed in Figure 29, the proposed MT-SigNet-CL (DeepInversion + Competition) model provides a statistically significant improvement in the generalization of writer-dependent verification compared to the previous related models SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) and SigNet (Synthetic) (VIANA et al., 2023). Furthermore, the MT-SigNet-CL (DeepInversion + Competition) model is statistically equivalent to the MT-SigNet-CL (GPDS-960), which indicates that the inverted data can replace the original GPDS-960 data in model training.

4.4.3.2 *Writer-Independent Generalization Performance*

The writer-independent generalization performance for the GPDS-150S, GPDS-300S, GPDS-300, CEDAR, MCYT-75, BHSig-B and BHSig-H datasets is listed in Table 24.

Table 24 – Writer-Independent verification performance of the models studied in this work. Users of the GPDS-150S, GPDS-300S, GPDS-300, CEDAR, BHSig-B and BHSig-H datasets are tested using 12 reference signatures. Users of the MCYT-75 dataset are tested using 10 reference signatures.

	GPDS-150S	GPDS-300S	GPDS-300	CEDAR	MCYT-75	BHSig-B	BHSig-H
SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a)	17.63 (±0.47)	17.59 (±0.47)	3.65 (±0.16)	**3.63 (±1.13)**	3.86 (±1.83)	3.02 (±1.58)	6.92 (±0.74)
SigNet (Synthetic) (VIANA et al., 2023)	4.53 (±0.47)	4.44 (±0.16)	12.48 (±0.44)	6.07 (±1.78)	4.99 (±0.82)	7.56 (±1.14)	4.54 (±0.94)
SigNet-F (HAFEMANN; SABOURIN; OLIVEIRA, 2017a)	17.60 (±0.89)	17.55 (±0.60)	**2.03 (±0.26)**	7.70 (±2.71)	4.73 (±1.24)	3.17 (±0.77)	8.04 (±0.69)
SigNet-CL (GPDS-960)	3.55 (±0.40)	3.47 (±0.26)	5.42 (±0.28)	3.89 (±1.38)	4.29 (±0.66)	4.39 (±1.38)	4.26 (±0.80)
SigNet-CL (DeepInversion + Competition)	3.65 (±0.44)	3.56 (±0.16)	5.41 (±0.40)	4.38 (±1.37)	5.14 (±1.95)	4.91 (±2.39)	4.68 (±0.95)
MT-SigNet-CL (GPDS-960)	**3.35 (±0.42)**	**3.32 (±0.26)**	5.60 (±0.22)	3.98 (±1.25)	**3.47 (±1.09)**	3.24 (±1.32)	**3.29 (±0.75)**
MT-SigNet-CL (DeepInversion + Competition)	3.38 (±0.42)	3.42 (±0.28)	5.75 (±0.20)	4.43 (±1.58)	4.34 (±1.30)	**3.01 (±1.31)**	3.56 (±0.79)

We observed the writer-independent performance of the models provided by the first task of the proposed training framework (Figure 19): on the GPDS-150S and GPDS-300S synthetic datasets, the proposed SigNet-CL (DeepInversion + Competition) and SigNet-CL (GPDS-960) models provide lower error rates in writer-independent verification than the SigNet and SigNet (Synthetic) models. However, SigNet model provides better writer-independent verification results on GPDS-300, CEDAR, MCYT-75, and BHSig-B real datasets when compared to SigNet-CL based models. Despite this, the proposed SigNet-CL models have intermediate performance, performing better than the SigNet (Synthetic) model trained only with synthetic data on the GPDS-150S, GPDS-300S, GPDS-300, CEDAR, MCYT-75, BHSig-B and BHSig-H datasets. When specifically considering the non-Western script BHSig-H dataset, the SigNet model provided a higher writer-independent verification error rate. On the contrary, the proposed SigNet-CL models have an improved performance for writer-independent verification in this dataset.

Given this, we observed the effect on model generalization after applying contrastive adjust-

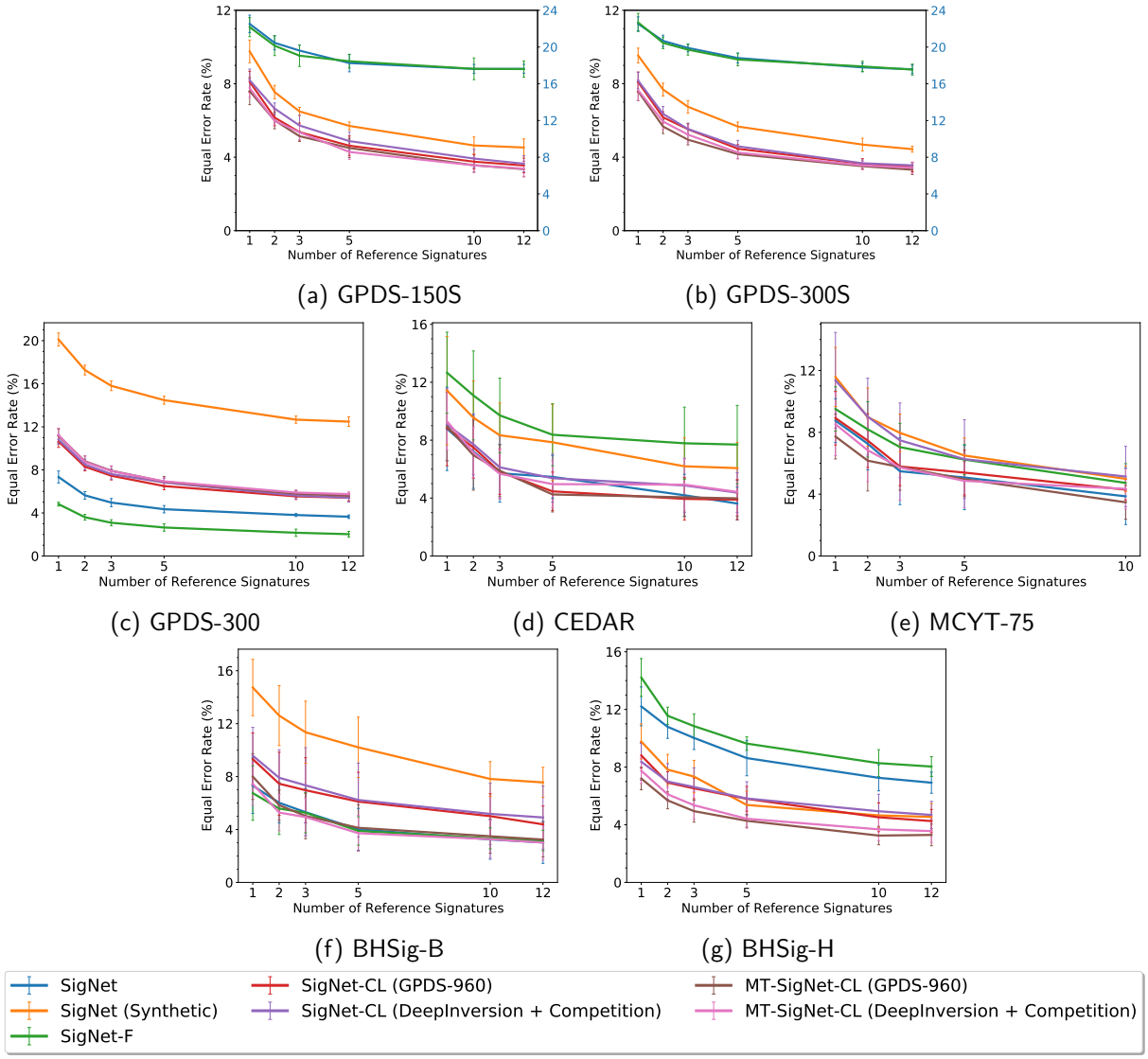


Figure 30 – Average performance of writer-independent classifiers using user thresholds for the GPDS-150S, GPDS-300S, GPDS-300, CEDAR, MCYT-75, BHSig-B and BHSig-H datasets, as the number of reference genuine signatures (per user) available for training is varied. Given that SigNet and SigNet-F performance errors are much greater than the other models' errors specifically in the GPDS-150S and GPDS-300S datasets, the SigNet and SigNet-F performances are represented in a secondary axis positioned to the right in Figures 30a and 30b.

ments to the SigNet-CL proposed models in the second task of the proposed training framework (Figure 19). In the writer-independent verification scenario, we observed that the contrastive adjustment over the SigNet-CL (DeepInversion + Competition) model that produces the MT-Signet-CL (DeepInversion + Competition) model decreases the writer-independent error rate in the verification of the GPDS-150S, GPDS-300S, MCYT-75, BHSig-B, and BHSig-H datasets, generating an improved model for writer-independent verification. Especially on the BHSig-B dataset, the adjustment diminished the error rate by 1.9%.

The proposed MT-Signet-CL models performs better than SigNet for writer-independent verification on the GPDS-150S, GPDS-300S, MCYT-75, BHSig-B, and BHSig-H datasets.

But on the contrary, the SigNet model has presented a better writer-independent performance on the GPDS-300 and CEDAR datasets. Nevertheless, it is worth noting that MT-SigNet-CL models performed better than SigNet (Synthetic) in all evaluated datasets.

The Figure 30 displays the Equal Error Rate achieved as the number of reference signatures r used for training writer-independent classifiers varies. In all models, the error rate generally decreases as the number of reference signatures increases.

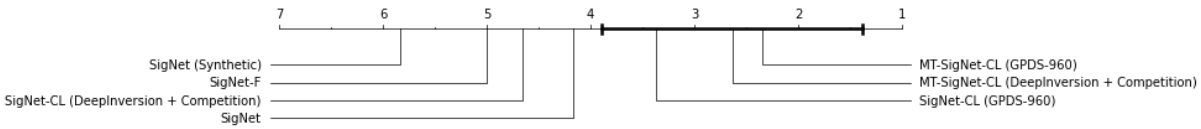


Figure 31 – Comparison of the proposed *MT-SigNet-CL (DeepInversion + Competition)* model against the other models studied in this work with the Bonferroni-Dunn post hoc tests. The models were tested in writer-independent approach with user thresholds using signature data obtained from the GPDS-150S, GPDS-300S, GPDS-300, CEDAR, MCYT-75, HSig-B and BHSig-H datasets. The GPDS-150S dataset provides synthetic signatures of 150 users, the GPDS-300S dataset provides synthetic signatures of 300 users, the GPDS-300 dataset provides real signatures of 300 users, the CEDAR dataset provides real signatures of 55 users, the MCYT-75 dataset provides real signatures of 75 users, the HSig-B dataset provides real signatures of 100 users and the HSig-H dataset provides real signatures of 160 users. All the signatures obtained from these datasets are applied together in the performed statistical tests. All the models with ranks outside the marked interval are significantly different ($p < 0.05$) from the *MT-SigNet-CL (DeepInversion + Competition)* model.

We performed Friedman tests (DEMsAR, 2006) in order to statistically compare the studied models in the writer-independent scenario. For the all evaluated datasets, we generated different signature segmentations with different number of reference signatures r (according to the Table 19), totaling forty-one evaluated subjects. All performed tests indicated that all evaluated models are not equivalent (rejecting the null hypothesis) with 5% level of significance. Then, Bonferroni-Dunn post hoc tests were performed to compare the studied models against the proposed MT-SigNet-CL (DeepInversion + Competition) model (illustrated in Figure 31). The right-most placed models are better ranked regarding the writer-independent verification performance. As observed, the proposed MT-SigNet-CL (DeepInversion + Competition) model provides a statistically significant improvement in the generalization of writer-independent verification compared to the previous related models SigNet (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) and SigNet (Synthetic) (VIANA et al., 2023). Additionally, the MT-SigNet-CL (DeepInversion + Competition) performs equivalently to the MT-SigNet-CL (GPDS-960), indicating that the inverted data can effectively substitute the original GPDS-960 dataset during model training in the writer-independent context.

4.4.3.3 *Comparing Performance with a Model Explicitly Trained with Skilled Forgeries*

The SigNet-F (HAFEMANN; SABOURIN; OLIVEIRA, 2017a) is a model trained with genuine signatures and skilled forgeries from the GPDS-960 development set, aiming to provide a more effective separation force between the representations of genuine signatures and skilled forgeries. We observed the performance of SigNet-F model explicitly trained with skilled forgeries in relation to the performance of SigNet trained only with genuine signatures. In writer-dependent verification (Table 23), SigNet-F performs better than SigNet only on the MCYT-75 and GPDS-300 datasets. In writer-independent verification (Table 24), SigNet-F presents only a slightly better performance on the GPDS-150S and GPDS-300S datasets and a more noticeable performance improvement on the GPDS-300. These results indicate that SigNet-F presents a degree of overfitting to the skilled forgeries from the GPDS-960 dataset.

Comparing the performance of SigNet-F with the proposed MT-SigNet-CL models, we observe that in writer-dependent verification (Table 23), improved performance is obtained with MT-SigNet-CL models in all datasets except the GPDS-300 dataset. In writer-independent verification (Table 24), enhanced performance with MT-SigNet-CL models is obtained on all datasets except the GPDS-300. Besides, the results of statistical tests considering different numbers of reference signatures in the writer-dependent and writer-independent approaches (respectively illustrated in Figures 29 and 31) indicate that the MT-SigNet-CL models provide a significant generalization improvement over the SigNet-F model. Given this, these results corroborate that the proposed models provide better generalization capacity for diverse datasets than a model explicitly trained with forgeries.

4.4.3.4 *Evaluating Performance in Large Vision Architectures*

RQ8) Is SigNet knowledge distillation helpful in training state-of-the-art large vision models for handwritten signature feature representation learning? In Tables 25 and 26, we respectively indicate the results obtained with the proposed method in large vision architectures for writer-dependent and writer-independent verification.

In both verification approaches, we demonstrate that employing the proposed method using the proposed inverted data source together with synthetic data for training large vision models significantly improves verification across all evaluated datasets. Thus, the proposed distillation mechanism is helpful in the training of new architectures instead of using only synthetic data

Table 25 – Writer-Dependent verification performance of models based on the ResNet-152 and Vision Transformer architectures. Users of the GPDS-150S, GPDS-300S, GPDS-300, CEDAR, BHSig-B and BHSig-H datasets are tested using 12 reference signatures. Users of the MCYT-75 dataset are tested using 10 reference signatures.

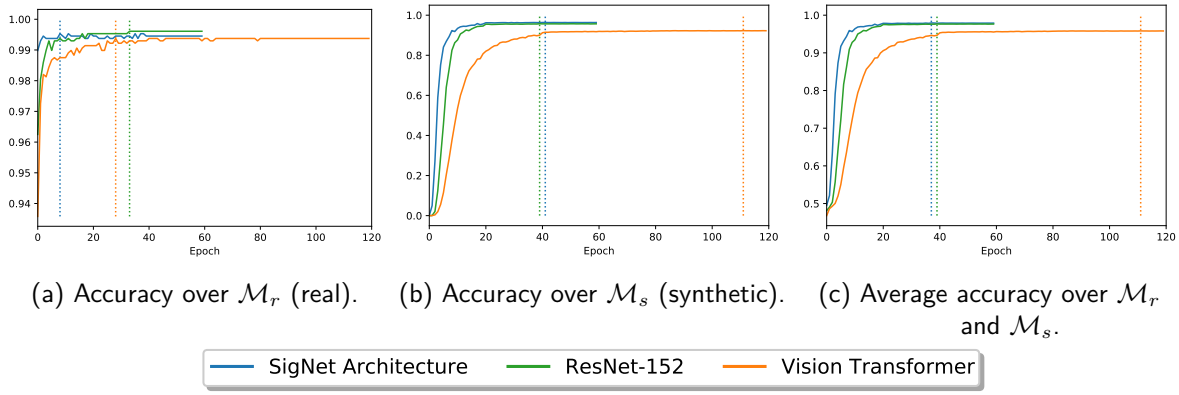
	GPDS-150S	GPDS-300S	GPDS-300	CEDAR	MCYT-75	BHSig-B	BHSig-H
ResNet-152 (Synthetic)	3.75 (±0.37)	3.89 (±0.38)	13.86 (±0.48)	6.57 (±0.88)	7.49 (±0.91)	6.88 (±0.68)	7.60 (±0.65)
ViT (Synthetic)	5.94 (±0.50)	6.33 (±0.40)	13.47 (±0.46)	7.32 (±0.41)	9.98 (±0.89)	7.04 (±0.87)	6.05 (±0.71)
MT-ResNet-152-CL (DeepInversion + Competition)	**2.19 (±0.33)**	**2.22 (±0.20)**	4.85 (±0.44)	3.13 (±0.60)	**4.59 (±0.29)**	**2.20 (±0.32)**	4.71 (±0.32)
MT-ViT-CL (DeepInversion + Competition)	2.90 (±0.42)	3.03 (±0.35)	**4.79 (±0.33)**	**2.97 (±0.56)**	4.76 (±0.87)	2.37 (±0.42)	**3.65 (±0.43)**

Table 26 – Writer-Independent verification performance of models based on the ResNet-152 and Vision Transformer architectures. Users of the GPDS-150S, GPDS-300S, GPDS-300, CEDAR, BHSig-B and BHSig-H datasets are tested using 12 reference signatures. Users of the MCYT-75 dataset are tested using 10 reference signatures.

	GPDS-150S	GPDS-300S	GPDS-300	CEDAR	MCYT-75	BHSig-B	BHSig-H
ResNet-152 (Synthetic)	4.65 (±0.25)	4.62 (±0.23)	16.44 (±0.42)	8.04 (±1.72)	11.39 (±2.89)	11.08 (±1.06)	9.07 (±1.33)
ViT (Synthetic)	7.36 (±0.62)	7.54 (±0.24)	14.80 (±0.41)	8.77 (±1.80)	16.95 (±1.96)	11.24 (±1.89)	7.22 (±1.04)
MT-ResNet-152-CL (DeepInversion + Competition)	**3.13 (±0.43)**	**3.12 (±0.26)**	5.18 (±0.38)	5.25 (±1.53)	7.73 (±1.91)	**2.50 (±0.89)**	5.43 (±0.91)
MT-ViT-CL (DeepInversion + Competition)	4.39 (±0.30)	4.49 (±0.17)	**5.13 (±0.33)**	**4.91 (±1.43)**	**7.61 (±1.59)**	3.10 (±0.56)	**4.41 (±0.86)**

(as done in Viana et al. (2023), Zheng et al. (2021).

However, despite the benefit provided by knowledge distillation in the training of large vision models, these new architectures present more prominent difficulty in the learning of synthetic signature representations compared to the SigNet architecture (which is a classical deep convolutional network based on the AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2017) architecture). Figure 32 shows the classification accuracy of the real and synthetic monitoring examples $\mathcal{M}_r$ and $\mathcal{M}_s$ over the epochs in model training. The vertical bars indicate the epoch in which the highest classification among the examples of the sets $\mathcal{M}_r$ and $\mathcal{M}_s$ is obtained for each architecture. The best obtained classification accuracy of monitored examples during the training of models based on the Vision Transformer, ResNet-152 and SigNet architectures in the same experimental configuration (only the architecture is changed) is respectively 95.83%, 97.67%, 97.93% (Figure 32c). This specifically occurs due to a discrepancy in the classification of synthetic examples (Figure 32b) that is more evident in the Vision Transformer architecture.



(a) Accuracy over $\mathcal{M}_r$ (real). (b) Accuracy over $\mathcal{M}_s$ (synthetic). (c) Average accuracy over $\mathcal{M}_r$ and $\mathcal{M}_s$.

Figure 32 – Classification accuracy of the real and synthetic monitoring examples $\mathcal{M}_r$ and $\mathcal{M}_s$ over the epochs in the first task of model training using DeepInversion + Competition data with the evaluated architectures.

4.4.3.5 Comparison with the State-of-the-art

In this section, the MT-SigNet-CL (DeepInversion + Competition) model is compared with the related state-of-the-art works in which deep learning generative methods and large vision architectures are applied for offline handwritten signature verification (listed in Table 27). It is important to note that the protocols for training feature extractors and classifiers can differ between related works since no single definitive protocol has been adopted by all works. Besides, none of the related works verify the performance on the same datasets analyzed in this work, but only on some of them. Therefore, the comparison presented in this section intends to place an overview of the performance of the existing methods in relation to the proposed framework.

In writer-dependent verification approach, the proposed method provides better verification performance on the GPDSsynthetic dataset when compared to recent works based on new architectures (MERSA et al., 2019) and generative networks (YAPıCı; TEKEREK; TOPALOĞLU, 2021). Furthermore, the proposed method provides verification improvement on the CEDAR dataset compared to generative methods (YONEKURA; GUEDES, 2021; JIANG et al., 2022). However, on the MCYT dataset, the related generative methods (YAPıCı; TEKEREK; TOPALOĞLU, 2021; JIANG et al., 2022) have presented better performance. Regarding the MCYT-75 dataset, in the experiments with Cycle-GAN performed in Yapıcı, Tekerek e Topaloğlu (2021), the signature data is partitioned in relation to the signatures instead of users; it simplifies the problem because examples from the same user are used to train and test generative models but introduces a higher cost in system maintenance as each user needs an associated generative model. In the experiments with Cycle-GAN on-2-off performed in Jiang et al. (2022), only

the performance obtained on the MCYT-75 dataset stands out. This result can be justified because the generative model proposed in Jiang et al. (2022) is trained with samples provided from the MCYT online dataset (with 330 users) and tested with a derived subcorpus with offline signatures of 75 users. On the contrary, in this work, the set of users used for model training and verification is entirely disjoint, indicating a competitive generalization capability of the features learned by the proposed model for a more diverse range of users and datasets.

When considering a model based on SigNet distillation to the ResNet architecture, the method proposed in Tsourounis et al. (2023) delivers improved performance on the CEDAR

Table 27 – Comparison with state-of-the art generative methods. The proposed method is the MT-SigNet-CL (DeepInversion + Competition) model.

Dataset	Method	#Ref	EER
Writer-Dependent			
GPDSsynthetic	ResNet (MERSA et al., 2019)	10	6.81
	Cycle-GAN (YAPICI; TEKEREK; TOPALOĞLU, 2021)	10	12.34 ($\pm$ 0.20)
	Proposed Method	12	2.31 ($\pm$ 0.14)
GPDS-300	ResNet S-T FKD (TSOUROUNIS et al., 2023)	12	2.74 ($\pm$ 0.28)
	Proposed Method	12	4.44 ($\pm$ 0.28)
CEDAR	Conditional GAN (YONEKURA; GUEDES, 2021)	1	18.53
	Cycle-GAN on-2-off (JIANG et al., 2022)	10	3.48
	ResNet S-T FKD (TSOUROUNIS et al., 2023)	12	2.25 ($\pm$ 0.24)
	Proposed Method	12	2.94 ($\pm$ 0.41)
MCYT-75	ResNet (MERSA et al., 2019)	10	3.98
	Cycle-GAN (YAPICI; TEKEREK; TOPALOĞLU, 2021)	10	2.58 ($\pm$ 0.43)
	Cycle-GAN on-2-off (JIANG et al., 2022)	10	2.01
	ResNet S-T FKD (TSOUROUNIS et al., 2023)	10	3.29 ($\pm$ 0.62)
	Proposed Method	10	2.95 ($\pm$ 0.73)
BHSig-B (Bengali)	TransOSV (ViT) (LI et al., 2024)	12	0.95 ($\pm$ 0.24)
	Proposed Method	12	1.55 ($\pm$ 0.33)
BHSig-H (Hindi)	TransOSV (ViT) (LI et al., 2024)	12	3.43 ($\pm$ 0.21)
	Proposed Method	12	2.43 ($\pm$ 0.41)
Writer-Independent			
GPDSsynthetic	Adversarial variation network (LI; WEI; HU, 2022)	1	9.77
	TransOSV (ViT) (LI et al., 2024)	1	10.64
	Proposed Method	12	3.42 ($\pm$ 0.28)
CEDAR	Adversarial variation network (LI; WEI; HU, 2022)	1	3.77
	Proposed Method	12	4.43 ($\pm$ 1.58)
BHSig-B (Bengali)	Adversarial variation network (LI; WEI; HU, 2022)	1	6.14
	TransOSV (ViT) (LI et al., 2024)	1	9.90
	Proposed Method	12	3.01 ($\pm$ 1.31)
BHSig-H (Hindi)	Adversarial variation network (LI; WEI; HU, 2022)	1	5.65
	TransOSV (ViT) (LI et al., 2024)	1	3.24
	Proposed Method	12	3.56 ($\pm$ 0.79)

and GPDS-300 datasets. However, on the contrary, our proposed method demonstrates better performance than ResNet-based models (TSOUROUNIS et al., 2023; MERSA et al., 2019) on the MCTY-75 dataset. Furthermore, in Tsourounis et al. (2023) the resulting models are not tested on the GPDSsynthetic, BHSig-H, and BHSig-B datasets in such a way that it is not possible to understand whether such models would continue to maintain suitable verification on more diverse datasets (especially the GPDSsynthetic dataset, which we demonstrated that presents very discrepant divergence with the GPDS-960 distribution). When considering the BHSig-H and BHSig-B datasets, the Li et al. (2024) method based on Vision Transformers provides better performance on BHSig-B (Bengali), but in contrast, the method proposed in this work offers superior performance on the BHSig-H (Hindi) dataset, indicating that our method provides competitive performance when compared to the large architecture method proposed by Li et al. (2024).

In the writer-independent verification scenario, the Li et al. (2024), Li, Wei e Hu (2022) methods are designed to adopt a single reference signature for verification. In this work, we use multiple reference signatures to make the decision. Our proposed method provides improved writer-independent verification performance over the generative method Li, Wei e Hu (2022) on the GPDSsynthetic, BHSig-B, and BHSig-H datasets; and over the Vision Transformer based method Li et al. (2024) on the GPDSsynthetic and BHSig-B datasets. It is worth noting that in Li et al. (2024), Li, Wei e Hu (2022) methods, the models for feature extraction are trained with a sub-partition of the same dataset used in the tests and employing skilled forgeries. Thus, we can assume that these related models are over-adjusted to identify the forgeries of the test dataset.

4.4.3.6 Adaptation of the Proposed Method to other Applications

In this section, we discuss how the proposed method can be adapted to generate models trained with datasets from other divergent scripts. For instance, suppose that a new student model is trained with a divergent script dataset but minimizing the divergence with the Western script knowledge provided by the SigNet without directly depending on the GPDS-960 data. In this case, the SigNet model for knowledge distillation and an inverted data source is needed, and thus, one of the following approaches can be adopted.

A straightforward and less computationally expensive approach is to train the new student model through the loss function defined in Equation 4.8 using the λ_p hyper-parameter defined

according to the values described in the Tables 21 and 22. We demonstrated in our experiments that the adoption of the aforementioned hyper-parameter configuration is generalizable in the verification of different datasets (and with different architectures) so that the values found for λ_p are suitable for the distillation of GPDS-960 knowledge through the SigNet using inverted data source.

A second approach if a particular tuning of the λ_p hyper-parameter is desired is to perform a hyper-parameter search regarding the verification performance between two disjoint validation sets: one of the Western script and another of the specific application. In the case of the Western script validation set, an alternative is to adopt partitions from the MCYT-75 dataset as a validation set, since we verified in our experiments an improvement in the verification of this Western script datasets with the proposed method.

4.5 CONCLUSION

In this work, we propose a method to combine the distribution of a large number of synthetic signatures from the GPDSsynthetic dataset with the distribution of real signatures from the GPDS-960 dataset. Knowledge about real signatures is obtained through the distillation from the pre-trained SigNet model. We also propose a dataset obtained with a generative technique that inverts examples from the pre-coded distribution in SigNet. The inverted examples do not exhibit the original traits of the GPDS-960 signatures, so we consider they can be publicly shared. Thus, the introduction of a large synthetic variety, together with the minimization of the distribution divergence with real data, produces more robust models to verify examples from different datasets. Moreover, we found improvement in the verification of non-Western script datasets.

Employing the inverted data in distillation is beneficial for writer-dependent verification compared to using the original GPDS-960 data. In writer-independent verification, the inverted data maintained a competitive verification performance. Presumably, the inverted data produces a more prominent adaptability of the representations to the testing datasets. This is advantageous in the writer-dependent scenario but provides less uniform decision boundaries among users for the writer-independent scenario. In our experiments, we found that this problem is minimized by a contrastive adjustment of the models that produce more homogeneous representations. Furthermore, in both verification approaches, we identified that the SigNet-F model trained with skilled forgeries has an excessively deviated distribution to identify sub-

partitions of the GPDS-960 dataset. In contrast, the proposed models trained with inverted data present a more balanced verification considering the seven assessed datasets.

Finally, we demonstrated that the proposed method can be used to improve verification results in the training of new architectures, especially when access to the GPDS-960 dataset is unavailable. Therefore, we expect this work to encourage the experimentation of new architectures integrated with the proposed distillation method for the learning of handwritten signature feature representations.

5 CONCLUDING REMARKS

In this thesis, we propose a framework founded on contrastive optimization for learning handwritten signature representations based on three demanded properties for the feature representations: obtaining denser and more separated clusters of genuine signatures (smaller intra-class distances and larger inter-class distances) together with a local distancing of the skilled forgery representations with respect to each respective genuine cluster. In this thesis, we demonstrate that achieving these properties in the representation space implies an improvement in the subsequent verification.

In addition to this aspect, we verified that the divergence among the distribution of distances related to the mentioned properties obtained in different datasets generates difficulties in generalization, and especially such challenges are more experimentally evident when a feature representation is tested against a more diverse range of datasets.

For illustration, a model explicitly trained with GPDS-960 skilled forgeries generates separation distances between genuine signatures and skilled forgeries over-adjusted for subsets of the own GPDS-960 dataset in such a way that this separation force is extremely strong in separating forgeries from other different datasets. On the other hand, in this work, we demonstrate that introducing a more considerable synthetic variety together with a more balanced separation force of skilled forgeries obtained through SigNet knowledge distillation contributes to a better generalization ability.

We consider this thesis can influence in the development of new handwritten signature feature representation learning methods based on these mentioned properties, introducing an interpretative nuance on the quality of the obtained representations and their association with the error metrics that consider the separation between genuine signatures and skilled forgeries.

To conclude, we enumerate below the answers to the research questions investigated in this thesis:

RQ1) How do the contrastive losses adopted in the proposed framework affect signature verification and model generalization?

- The proposed multi-task framework refines the original representation space by applying contrastive losses while maintaining the generalization ability of the cross-entropy loss. This approach improves intra-cluster density and ensures more uniform cluster dispersion, benefiting model generalization to different datasets.

RQ2) Does the proposed multi-task approach for learning representations provide signature verification improvement in comparison to a contrastive single-task approach?

• Single-task contrastive models over-adjust to the training dataset. In contrast, the proposed multi-task framework improves generalization compared to directly using contrastive methods.

RQ3) Do the proposed multi-task contrastive models provide significantly better feature representations for handwritten signatures than SigNet?

• The experiments indicated a significant improvement in signature verification using the proposed multi-task contrastive framework compared to the SigNet (a cross-entropy based model), suggesting the proposed framework as a general strategy for training contrastive models for handwritten signature feature extraction.

RQ4) Does NT-Xent loss (with implicit hard negative mining) provide better feature representations than Triplet loss (with explicit hard negative mining) for handwritten signature verification?

• The more satisfactory verification performance of the Triplet loss compared to the NT-Xent loss is explained by its more significant separation ability between skilled forgery and genuine signatures, facilitated by the explicit mechanism for mining hard negatives within the proposed framework.

RQ5) How can we obtain a more suitable signature representation with models trained using synthetic and inverted data in a situation where the access to the GPDS-960 dataset is unavailable?

• The proposed knowledge distillation mechanism using inverted data has a better average performance than previous models (SigNet and SigNet (Synthetic)) when considering real and synthetic data sources. In situations where the original GPDS-960 data is unavailable, inverted data serves as an effective substitute for transferring real signature characteristics to new models.

RQ6) Is it feasible to rely solely on synthetic data from the GPDSsynthetic dataset for training models?

• A model trained with GPDSsynthetic data (as the SigNet (Synthetic)) effectively groups real and synthetic users but fails to adequately separate real skilled forgeries. As

a result, relying solely on synthetic data from the GPDSsynthetic dataset for training models is unfeasible.

RQ7) Can the characteristics of real and synthetic datasets be complemented using continual learning to offer a more balanced verification performance across diverse datasets?

- Models trained with both real and synthetic data distributions offer better generalization across various datasets. In contrast, a model trained with GPDS-960 skilled forgeries over-fits to the own GPDS-960 examples, hindering its ability to generalize to other datasets. The proposed method in this work provides more balanced generalization across both Western and non-Western script datasets.

RQ8) Is SigNet knowledge distillation helpful in training state-of-the-art large vision models for handwritten signature feature representation learning?

- Employing the proposed method with inverted and synthetic data for training large vision models enhances verification. The proposed distillation mechanism is beneficial for training new architectures, offering an alternative instead of using only synthetic data.

5.1 FUTURE WORK

This thesis presents a promising approach for handwritten signature feature representation learning. However, there are opportunities for further enhancement. The findings and analyses presented in this work provide inspiration for future research. Potential directions for future work include:

We identified in our experiments a more prominent difficulty in classifying synthetic examples by large vision models compared to the SigNet architecture, which is based on classical deep convolutional neural networks. Thus, an important research direction is the development of new architectures integrated with the proposed framework that can obtain better accuracy in classifying an extensive collection of synthetic examples.

In this thesis, we invert the genuine signatures of the GPDS-960 development set. One direction of research is to extrapolate and generate more inverted examples by: possibly simulating the distribution of new users from the existing users, as well as increasing the number of inverted examples for each user through augmentations.

REFERENCES

ARAB, N.; NEMMOUR, H.; CHIBANI, Y. Multiscale fusion of histogram-based features for robust off-line handwritten signature verification. In: *2020 IEEE/ACS 17th International Conference on Computer Systems and Applications (AICCSA)*. [S.l.: s.n.], 2020. p. 1–5.

ARAB, N.; NEMMOUR, H.; CHIBANI, Y. A new synthetic feature generation scheme based on artificial immune systems for robust offline signature verification. *Expert Systems with Applications*, v. 213, p. 119306, 2023. ISSN 0957-4174.

BENGIO, Y.; LECUN, Y.; HINTON, G. Deep learning for ai. *Commun. ACM*, Association for Computing Machinery, New York, NY, USA, v. 64, n. 7, p. 58–65, jun. 2021. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/3448250>.

BHAVANI, S. D.; BHARATHI, R. K. A multi-dimensional review on handwritten signature verification: strengths and gaps. *Multimedia Tools and Applications*, v. 83, n. 1, p. 2853–2894, 2024. Disponível em: <https://doi.org/10.1007/s11042-023-15357-2>.

BHUNIA, A. K.; ALAEI, A.; ROY, P. P. Signature verification approach using fusion of hybrid texture features. *Neural Computing and Applications*, v. 31, n. 12, p. 8737–8748, 2019. Disponível em: <https://doi.org/10.1007/s00521-019-04220-x>.

BOUAMRA, W.; DIAZ, M.; FERRER, M. A.; NINI, B. Spiral based run-length features for offline signature verification. In: CARMONA-DUARTE, C.; DIAZ, M.; FERRER, M. A.; MORALES, A. (Ed.). *Intertwining Graphonomics with Human Movements*. Cham: Springer International Publishing, 2022. p. 26–41. ISBN 978-3-031-19745-1.

BOUDIAF, M.; RONY, J.; ZIKO, I. M.; GRANGER, E.; PEDERSOLI, M.; PIANTANIDA, P.; AYED, I. B. A unifying mutual information view of metric learning: Cross-entropy vs. pairwise losses. In: VEDALDI, A.; BISCHOF, H.; BROX, T.; FRAHM, J.-M. (Ed.). *Computer Vision – ECCV 2020*. Cham: Springer International Publishing, 2020. p. 548–564. ISBN 978-3-030-58539-6.

CARUANA, R. Multitask learning. *Machine learning*, Springer, v. 28, n. 1, p. 41–75, 1997.

CHEN, T.; KORNBLITH, S.; NOROUZI, M.; HINTON, G. A simple framework for contrastive learning of visual representations. In: III, H. D.; SINGH, A. (Ed.). *Proceedings of the 37th International Conference on Machine Learning*. PMLR, 2020. (Proceedings of Machine Learning Research, v. 119), p. 1597–1607. Disponível em: <http://proceedings.mlr.press/v119/chen20j.html>.

CHEN, W.; CHEN, X.; ZHANG, J.; HUANG, K. Beyond triplet loss: A deep quadruplet network for person re-identification. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2017.

DEMsAR, J. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.*, JMLR.org, v. 7, p. 1–30, dec 2006. ISSN 1532-4435.

DEY, S.; DUTTA, A.; TOLEDO, J. I.; GHOSH, S. K.; LLADOS, J.; PAL, U. *SigNet: Convolutional Siamese Network for Writer Independent Offline Signature Verification*. 2017.

DIAZ, M.; FERRER, M. A.; ESKANDER, G. S.; SABOURIN, R. Generation of duplicated off-line signature images for verification systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 39, n. 5, p. 951–964, 2017.

DIAZ, M.; MENDOZA-GARCíA, A.; FERRER, M. A.; SABOURIN, R. A survey of handwriting synthesis from 2019 to 2024: A comprehensive review. *Pattern Recognition*, p. 111357, 2025. ISSN 0031-3203. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0031320325000172>.

DOMENICO, M. D.; NICOSIA, V.; ARENAS, A.; LATORA, V. Structural reducibility of multilayer networks. *Nature Communications*, v. 6, n. 1, p. 6864, 2015. Disponível em: <https://doi.org/10.1038/ncomms7864>.

DOSOVITSKIY, A.; BEYER, L.; KOLESNIKOV, A.; WEISSENBORN, D.; ZHAI, X.; UNTERTHINER, T.; DEHGHANI, M.; MINDERER, M.; HEIGOLD, G.; GELLY, S.; USZKOREIT, J.; HOULSBY, N. An image is worth 16x16 words: Transformers for image recognition at scale. In: *International Conference on Learning Representations*. [s.n.], 2021. Disponível em: <https://openreview.net/forum?id=YicbFdNTTy>.

FERRER, M. A.; DIAZ, M.; CARMONA-DUARTE, C.; MORALES, A. A behavioral handwriting model for static and dynamic signature synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 39, n. 6, p. 1041–1053, 2017.

FIERREZ, J.; GALBALLY, J.; ORTEGA-GARCIA, J.; FREIRE, M. R.; ALONSO-FERNANDEZ, F.; RAMOS, D.; TOLEDANO, D. T.; GONZALEZ-RODRIGUEZ, J.; SIGUENZA, J. A.; GARRIDO-SALAS, J.; ANGUIANO, E.; RIVERA, G. Gonzalez-de; RIBALDA, R.; FAUNDEZ-ZANUY, M.; ORTEGA, J. A.; CARDEÑOSO-PAYO, V.; VILORIA, A.; VIVARACHO, C. E.; MORO, Q. I.; IGARZA, J. J.; SANCHEZ, J.; HERNAEZ, I.; ORRITE-URUÑUELA, C.; MARTINEZ-CONTRERAS, F.; GRACIA-ROCHE, J. J. Biosecurid: a multimodal biometric database. *Pattern Analysis and Applications*, v. 13, n. 2, p. 235–246, 2010. Disponível em: <https://doi.org/10.1007/s10044-009-0151-4>.

FRIEDMAN, J.; HASTIE, T.; TIBSHIRANI, R. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, Institute of Mathematical Statistics, v. 28, n. 2, p. 337–407, 2000.

FUGLEDE, B.; TOPSOE, F. Jensen-shannon divergence and hilbert space embedding. In: *International Symposium onInformation Theory, 2004. ISIT 2004. Proceedings.* [S.l.: s.n.], 2004. p. 31–.

GOODFELLOW, I.; POUGET-ABADIE, J.; MIRZA, M.; XU, B.; WARDE-FARLEY, D.; OZAIR, S.; COURVILLE, A.; BENGIO, Y. Generative adversarial nets. In: GHAHRAMANI, Z.; WELLING, M.; CORTES, C.; LAWRENCE, N.; WEINBERGER, K. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2014. v. 27. Disponível em: <https://proceedings.neurips.cc/paper_files/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf>.

GUTMANN, M.; HYVäRINEN, A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In: TEH, Y. W.; TITTERINGTON, M. (Ed.). *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. Chia Laguna Resort, Sardinia, Italy: PMLR, 2010. (Proceedings of Machine Learning Research, v. 9), p. 297–304. Disponível em: <http://proceedings.mlr.press/v9/gutmann10a.html>.

HADSELL, R.; CHOPRA, S.; LECUN, Y. Dimensionality reduction by learning an invariant mapping. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. [S.l.: s.n.], 2006. v. 2, p. 1735–1742.

HAFEMANN, L. G.; OLIVEIRA, L. S.; SABOURIN, R. Fixed-sized representation learning from offline handwritten signatures of different sizes. *International Journal on Document Analysis and Recognition (IJDAR)*, v. 21, n. 3, p. 219–232, 2018. Disponível em: <https://doi.org/10.1007/s10032-018-0301-6>.

HAFEMANN, L. G.; SABOURIN, R.; OLIVEIRA, L. S. Learning features for offline handwritten signature verification using deep convolutional neural networks. *Pattern Recognition*, v. 70, p. 163 – 176, 2017. ISSN 0031-3203. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0031320317302017>.

HAFEMANN, L. G.; SABOURIN, R.; OLIVEIRA, L. S. Offline handwritten signature verification — literature review. In: *2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*. [S.l.: s.n.], 2017. p. 1–8.

HAMADENE, A.; CHIBANI, Y. One-class writer-independent offline signature verification using feature dissimilarity thresholding. *IEEE Transactions on Information Forensics and Security*, v. 11, n. 6, p. 1226–1238, 2016.

HAMEED, M. M.; AHMAD, R.; KIAH, M. L. M.; MURTAZA, G. Machine learning-based offline signature verification systems: A systematic review. *Signal Processing: Image Communication*, v. 93, p. 116139, 2021. ISSN 0923-5965. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0923596521000047>.

HE, K.; FAN, H.; WU, Y.; XIE, S.; GIRSHICK, R. Momentum contrast for unsupervised visual representation learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2020.

HE, K.; ZHANG, X.; REN, S.; SUN, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 37, n. 9, p. 1904–1916, 2015.

HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2016. p. 770–778.

HINTON, G.; VINYALS, O.; DEAN, J. *Distilling the Knowledge in a Neural Network*. 2015.

HOFFER, E.; AILON, N. Deep metric learning using triplet network. In: FERAGEN, A.; PELILLO, M.; LOOG, M. (Ed.). *Similarity-Based Pattern Recognition*. Cham: Springer International Publishing, 2015. p. 84–92. ISBN 978-3-319-24261-3.

HU, J.; CHEN, Y. Offline signature verification using real adaboost classifier combination of pseudo-dynamic features. In: *2013 12th International Conference on Document Analysis and Recognition*. [S.l.: s.n.], 2013. p. 1345–1349.

HU, J.; LU, J.; TAN, Y.-P. Discriminative deep metric learning for face verification in the wild. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2014. p. 1875–1882.

HU, J.; SHEN, L.; SUN, G. Squeeze-and-excitation networks. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2018. p. 7132–7141.

HUANG, G.; LIU, Z.; MAATEN, L. van der; WEINBERGER, K. Q. Densely connected convolutional networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2017.

JAISWAL, A.; BABU, A. R.; ZADEH, M. Z.; BANERJEE, D.; MAKEDON, F. A survey on contrastive self-supervised learning. *Technologies*, v. 9, n. 1, 2021. ISSN 2227-7080. Disponível em: <https://www.mdpi.com/2227-7080/9/1/2>.

JIANG, J.; LAI, S.; JIN, L.; ZHU, Y.; ZHANG, J.; CHEN, B. Forgery-free signature verification with stroke-aware cycle-consistent generative adversarial network. *Neurocomputing*, v. 507, p. 345–357, 2022. ISSN 0925-2312.

KALERA, M. K.; SRIHARI, S.; XU, A. Offline signature verification and identification using distance statistics. *International Journal of Pattern Recognition and Artificial Intelligence*, v. 18, n. 07, p. 1339–1360, 2004. Disponível em: <https://doi.org/10.1142/S0218001404003630>.

KENNEDY, R.; TIEDE, L. B.; OZER, A. L.; MARINOV, N. Signed, sealed, counted? an experimental study of mail-in ballot signature verification. *Political Research Quarterly*, SAGE Publications Inc, p. 10659129251319298, 2025/06/06 2025. Disponível em: <https://doi.org/10.1177/10659129251319298>.

KHOSLA, P.; TETERWAK, P.; WANG, C.; SARNA, A.; TIAN, Y.; ISOLA, P.; MASCHINOT, A.; LIU, C.; KRISHNAN, D. Supervised contrastive learning. In: LAROCHELLE, H.; RANZATO, M.; HADSELL, R.; BALCAN, M. F.; LIN, H. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2020. v. 33, p. 18661–18673. Disponível em: <https://proceedings.neurips.cc/paper/2020/file/d89a66c7c80a29b1bdbab0f2a1a94af8-Paper.pdf>.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, Association for Computing Machinery, New York, NY, USA, v. 60, n. 6, p. 84–90, maio 2017. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/3065386>.

LAI, S.; JIN, L. Learning discriminative feature hierarchies for off-line signature verification. In: *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*. [S.l.: s.n.], 2018. p. 175–180.

LE-KHAC, P. H.; HEALY, G.; SMEATON, A. F. Contrastive representation learning: A framework and review. *IEEE Access*, v. 8, p. 193907–193934, 2020.

LI, H.; WEI, P.; HU, P. Avn: An adversarial variation network model for handwritten signature verification. *IEEE Transactions on Multimedia*, v. 24, p. 594–608, 2022.

LI, H.; WEI, P.; MA, Z.; LI, C.; ZHENG, N. Transosv: Offline signature verification with transformers. *Pattern Recognition*, v. 145, p. 109882, 2024. ISSN 0031-3203. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0031320323005800>.

LI, Z.; HOIEM, D. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 40, n. 12, p. 2935–2947, 2018.

LIU, L.; HUANG, L.; YIN, F.; CHEN, Y. Offline signature verification using a region based deep metric learning network. *Pattern Recognition*, v. 118, p. 108009, 2021. ISSN 0031-3203. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0031320321001965>.

MAATEN, L. van der; HINTON, G. Visualizing data using t-sne. *Journal of Machine Learning Research*, v. 9, n. 86, p. 2579–2605, 2008. Disponível em: <http://jmlr.org/papers/v9/vandermaaten08a.html>.

MACêDO, D.; REN, T. I.; ZANCHETTIN, C.; OLIVEIRA, A. L. I.; LUDERMIR, T. Entropic out-of-distribution detection: Seamless detection of unknown examples. *IEEE Transactions on Neural Networks and Learning Systems*, v. 33, n. 6, p. 2350–2364, 2022.

MAERGNER, P.; PONDENKANDATH, V.; ALBERTI, M.; LIWICKI, M.; RIESEN, K.; INGOLD, R.; FISCHER, A. Offline signature verification by combining graph edit distance and triplet networks. In: BAI, X.; HANCOCK, E. R.; HO, T. K.; WILSON, R. C.; BIGGIO, B.; ROBLES-KELLY, A. (Ed.). *Structural, Syntactic, and Statistical Pattern Recognition*. Cham: Springer International Publishing, 2018. p. 470–480. ISBN 978-3-319-97785-0.

MAERGNER, P.; PONDENKANDATH, V.; ALBERTI, M.; LIWICKI, M.; RIESEN, K.; INGOLD, R.; FISCHER, A. Combining graph edit distance and triplet networks for offline signature verification. *Pattern Recognition Letters*, v. 125, p. 527–533, 2019. ISSN 0167-8655.

MAERGNER, P.; RIESEN, K.; INGOLD, R.; FISCHER, A. A structural approach to offline signature verification using graph edit distance. In: *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*. [S.l.: s.n.], 2017. v. 01, p. 1216–1222.

MARUYAMA, T. M.; OLIVEIRA, L. S.; BRITTO, A. S.; SABOURIN, R. Intrapersonal parameter optimization for offline handwritten signature augmentation. *IEEE Transactions on Information Forensics and Security*, v. 16, p. 1335–1350, 2021.

MASOUDNIA, S.; MERSA, O.; ARAABI, B. N.; VAHABIE, A.-H.; SADEGHI, M. A.; AHMADABADI, M. N. Multi-representational learning for offline signature verification using multi-loss snapshot ensemble of cnns. *Expert Systems with Applications*, v. 133, p. 317 – 330, 2019. ISSN 0957-4174. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0957417419301666>.

MERSA, O.; ETAATI, F.; MASOUDNIA, S.; ARAABI, B. N. Learning representations from persian handwriting for offline signature verification, a deep transfer learning approach. In: *2019 4th International Conference on Pattern Recognition and Image Analysis (IPRIA)*. [S.l.: s.n.], 2019. p. 268–273.

MORDVINTSEV, A.; OLAH, C.; TYKA, M. *Inceptionism: Going Deeper into Neural Networks*. 2015. Disponível em: <https://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html>.

OORD, A. van den; LI, Y.; VINYALS, O. *Representation Learning with Contrastive Predictive Coding*. 2019.

ORTEGA-GARCIA, J.; FIERREZ-AGUILAR, J.; SIMON, D.; GONZALEZ, J.; FAUNDEZ-ZANUY, M.; ESPINOSA, V.; SATUE, A.; HERNAEZ, I.; IGARZA, J.-J.; VIVARACHO, C. et al. Mcyt baseline corpus: a bimodal biometric database. *IEE Proceedings-Vision, Image and Signal Processing*, IET, v. 150, n. 6, p. 395–401, 2003.

OTSU, N. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, v. 9, n. 1, p. 62–66, 1979.

PAL, S.; ALAEI, A.; PAL, U.; BLUMENSTEIN, M. Performance of an off-line signature verification method based on texture features on a large indic-script signature dataset. In: *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*. [S.l.: s.n.], 2016. p. 72–77.

RADFORD, A.; METZ, L.; CHINTALA, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

RANTZSCH, H.; YANG, H.; MEINEL, C. Signature embedding: Writer independent offline signature verification with deep metric learning. In: BEBIS, G.; BOYLE, R.; PARVIN, B.; KORACIN, D.; PORIKLI, F.; SKAFF, S.; ENTEZARI, A.; MIN, J.; IWAI, D.; SADAGIC, A.; SCHEIDEGGER, C.; ISENBERG, T. (Ed.). *Advances in Visual Computing*. Cham: Springer International Publishing, 2016. p. 616–625. ISBN 978-3-319-50832-0.

RIVARD, D.; GRANGER, E.; SABOURIN, R. Multi-feature extraction and selection in writer-independent off-line signature verification. *International Journal on Document Analysis and Recognition (IJDAR)*, v. 16, p. 83–103, 03 2011.

RUIZ, V.; LINARES, I.; SANCHEZ, A.; VELEZ, J. F. Off-line handwritten signature verification using compositional synthetic generation of signatures and siamese neural networks. *Neurocomputing*, v. 374, p. 30–41, 2020. ISSN 0925-2312. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0925231219313098>.

SABOUR, S.; FROSST, N.; HINTON, G. E. Dynamic routing between capsules. In: GUYON, I.; LUXBURG, U. V.; BENGIO, S.; WALLACH, H.; FERGUS, R.; VISHWANATHAN, S.; GARNETT, R. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. v. 30. Disponível em: <https://proceedings.neurips.cc/paper_files/paper/2017/file/2cad8fa47bbef282badbb8de5374b894-Paper.pdf>.

SCHROFF, F.; KALENICHENKO, D.; PHILBIN, J. Facenet: A unified embedding for face recognition and clustering. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2015. p. 815–823.

SERDOUK, Y.; NEMMOUR, H.; CHIBANI, Y. New off-line handwritten signature verification method based on artificial immune recognition system. *Expert Systems with Applications*, v. 51, p. 186–194, 2016. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417416000026>.

SERDOUK, Y.; NEMMOUR, H.; CHIBANI, Y. Handwritten signature verification using the quad-tree histogram of templates and a support vector-based artificial immune classification. *Image and Vision Computing*, v. 66, p. 26–35, 2017. ISSN 0262-8856. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0262885617301208>.

SHARIATMADARI, S.; EMADI, S.; AKBARI, Y. Patch-based offline signature verification using one-class hierarchical deep learning. *International Journal on Document Analysis and Recognition (IJDAR)*, v. 22, n. 4, p. 375–385, 2019. Disponível em: <https://doi.org/10.1007/s10032-019-00331-2>.

SIMONYAN, K.; ZISSERMAN, A. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015.

SOLEIMANI, A.; ARAABI, B. N.; FOULADI, K. Deep multitask metric learning for offline signature verification. *Pattern Recognition Letters*, v. 80, p. 84–90, 2016. ISSN 0167-8655. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0167865516301076>.

SOUZA, V. L.; OLIVEIRA, A. L.; CRUZ, R. M.; SABOURIN, R. Characterization of handwritten signature images in dissimilarity representation space. In: SPRINGER. *International Conference on Computational Science*. [S.l.], 2019. p. 192–206.

SOUZA, V. L.; OLIVEIRA, A. L.; CRUZ, R. M.; SABOURIN, R. A white-box analysis on the writer-independent dichotomy transformation applied to offline handwritten signature verification. *Expert Systems with Applications*, v. 154, p. 113397, 2020. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417420302219>.

SOUZA, V. L. F.; OLIVEIRA, A. L. I.; SABOURIN, R. A writer-independent approach for offline signature verification using deep convolutional neural networks features. In: *2018 7th Brazilian Conference on Intelligent Systems (BRACIS)*. [S.l.: s.n.], 2018. p. 212–217.

SZEGEDY, C.; LIU, W.; JIA, Y.; SERMANET, P.; REED, S.; ANGUELOV, D.; ERHAN, D.; VANHOUCKE, V.; RABINOVICH, A. Going deeper with convolutions. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2015. p. 1–9.

TOLOSANA, R.; VERA-RODRIGUEZ, R.; FIERREZ, J.; ORTEGA-GARCIA, J. Deepsign: Deep on-line signature verification. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, v. 3, n. 2, p. 229–239, 2021.

TSOUROUNIS, D.; THEODORAKOPOULOS, I.; ZOIS, E. N.; ECONOMOU, G. From text to signatures: Knowledge transfer for efficient deep feature learning in offline signature verification. *Expert Systems with Applications*, v. 189, p. 116136, 2022. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417421014652>.

TSOUROUNIS, D.; THEODORAKOPOULOS, I.; ZOIS, E. N.; ECONOMOU, G. *Leveraging Expert Models for Training Deep Neural Networks in Scarce Data Domains: Application to Offline Handwritten Signature Verification*. 2023. Disponível em: <https://arxiv.org/abs/2308.01136>.

VARGAS, F.; FERRER, M.; TRAVIESO, C.; ALONSO, J. Off-line handwritten signature gpds-960 corpus. In: *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*. [S.l.: s.n.], 2007. v. 2, p. 764–768.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L. u.; POLOSUKHIN, I. Attention is all you need. In: GUYON, I.; LUXBURG, U. V.; BENGIO, S.; WALLACH, H.; FERGUS, R.; VISHWANATHAN, S.; GARNETT, R. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. v. 30. Disponível em: <https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf>.

VIANA, T. B.; SOUZA, V. L.; OLIVEIRA, A. L.; CRUZ, R. M.; SABOURIN, R. Contrastive learning of handwritten signature representations for writer-independent verification. In: *2022 International Joint Conference on Neural Networks (IJCNN)*. [S.l.: s.n.], 2022. p. 01–09.

VIANA, T. B.; SOUZA, V. L.; OLIVEIRA, A. L.; CRUZ, R. M.; SABOURIN, R. A multi-task approach for contrastive learning of handwritten signature feature representations. *Expert Systems with Applications*, v. 217, p. 119589, 2023. ISSN 0957-4174.

VIANA, T. B.; SOUZA, V. L. F.; OLIVEIRA, A. L. I.; CRUZ, R. M. O.; SABOURIN, R. Robust handwritten signature representation with continual learning of synthetic data over predefined real feature space. In: SMITH, E. H. B.; LIWICKI, M.; PENG, L. (Ed.). *International Conference on Document Analysis and Recognition - ICDAR 2024*. Cham: Springer Nature Switzerland, 2024. p. 233–249. ISBN 978-3-031-70536-6.

WAN, Q.; ZOU, Q. Learning metric features for writer-independent signature verification using dual triplet loss. In: *2020 25th International Conference on Pattern Recognition (ICPR)*. [S.l.: s.n.], 2021. p. 3853–3859.

WEINBERGER, K. Q.; SAUL, L. K. Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.*, JMLR.org, v. 10, p. 207–244, jun. 2009. ISSN 1532-4435.

WU, C.; MANMATHA, R.; SMOLA, A. J.; KRäHENBüHL, P. Sampling matters in deep embedding learning. In: *2017 IEEE International Conference on Computer Vision (ICCV)*. [S.l.: s.n.], 2017. p. 2859–2867.

WU, Z.; XIONG, Y.; YU, S. X.; LIN, D. Unsupervised feature learning via non-parametric instance discrimination. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, 2018. p. 3733–3742. Disponível em: <https://doi.ieeecomputersociety.org/10.1109/CVPR.2018.00393>.

YAPıCı, M. M.; TEKEREK, A.; TOPALOĞLU, N. Deep learning-based data augmentation method and signature verification system for offline handwritten signature. *Pattern Analysis and Applications*, v. 24, n. 1, p. 165–179, 2021.

YILMAZ, M. B.; ÖZTÜRK, K. Recurrent binary patterns and cnns for offline signature verification. In: ARAI, K.; BHATIA, R.; KAPOOR, S. (Ed.). *Proceedings of the Future Technologies Conference (FTC) 2019*. Cham: Springer International Publishing, 2020. p. 417–434. ISBN 978-3-030-32523-7.

YIN, H.; MOLCHANOV, P.; ALVAREZ, J. M.; LI, Z.; MALLYA, A.; HOIEM, D.; JHA, N. K.; KAUTZ, J. Dreaming to distill: Data-free knowledge transfer via deepinversion. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2020.

YONEKURA, D.; GUEDES, E. Offline handwritten signature authentication with conditional deep convolutional generative adversarial networks. In: *Anais do XVIII Encontro Nacional de Inteligência Artificial e Computacional*. Porto Alegre, RS, Brasil: SBC, 2021. p. 482–491. ISSN 2763-9061. Disponível em: <https://sol.sbc.org.br/index.php/eniac/article/view/18277>.

ZHANG, Z.; LIU, X.; CUI, Y. Multi-phase offline signature verification system using deep convolutional generative adversarial networks. In: *2016 9th International Symposium on Computational Intelligence and Design (ISCID)*. [S.l.: s.n.], 2016. v. 2, p. 103–107.

ZHENG, Y.; IWANA, B. K.; MALIK, M. I.; AHMED, S.; OHYAMA, W.; UCHIDA, S. Learning the micro deformations by max-pooling for offline signature verification. *Pattern Recognition*, v. 118, p. 108008, 2021. ISSN 0031-3203. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0031320321001953>.

ZHU, J.-Y.; PARK, T.; ISOLA, P.; EFROS, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. [S.l.: s.n.], 2017.

ZOIS, E. N.; ALEWIJNSE, L.; ECONOMOU, G. Offline signature verification and quality characterization using poset-oriented grid features. *Pattern Recognition*, v. 54, p. 162–177, 2016. ISSN 0031-3203. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0031320316000133>.

ZOIS, E. N.; ALEXANDRIDIS, A.; ECONOMOU, G. Writer independent offline signature verification based on asymmetric pixel relations and unrelated training-testing datasets. *Expert Systems with Applications*, v. 125, p. 14–32, 2019. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417419300636>.

ZOIS, E. N.; TSOUROUNIS, D.; THEODORAKOPOULOS, I.; KESIDIS, A. L.; ECONOMOU, G. A comprehensive study of sparse representation techniques for offline signature verification. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, v. 1, n. 1, p. 68–81, 2019.

APPENDIX A – EXECUTION TIME OF EXPERIMENTS

A.1 EXPERIMENT ENVIRONMENT

The experiments in Chapter 3 were performed on the cluster provided by the Northeast Strategic Technologies Center - CETENE on machines with the following configuration: 32GB of RAM using NVIDIA GeForce GTX 980 GPUs with 4 GB memory size.

The experiments in Chapter 4 were performed on the cluster belonging to the advanced research computing platform provided by the Digital Research Alliance of Canada using machines with the following configuration: 48GB of RAM using NVIDIA V100 GPUs with 32 GB memory size.

A.2 EXECUTION TIMES

A.2.1 Inversion

The runtimes for the inversion of the development set from SigNet with the methods discussed in Section 4.2.1 are listed in Table 28.

Table 28 – Runtimes for the inversion of the development set in days(d), hours(h), minutes(m) and seconds(s).

Inversion method	Inversion time
DeepInversion	7d:10h:48m:25s
DeepInversion + Competition	14d:10h:33m:11s

A.2.2 Validation Experiments

The runtimes for the contrastive models studied in Section 3.4.2 with tests performed on the validation set $\mathcal{V}_v$ are presented in Table 29.

The runtimes for the proposed models adopting the continual learning-based framework described in Section 4.2 are presented in Tables 30, 31, and 32. Table 30 lists the model training execution times. Respectively, Tables 31 and 32 present the testing times on the real $\mathcal{V}_{vr}$ and synthetic $\mathcal{V}_{vs}$ validation sets (described in Section 4.4.2.3) using writer-dependent and writer-independent approaches.

Table 29 – Runtimes for the training of models and testing on the validation set $\mathcal{V}_v$ in days(d), hours(h), minutes(m) and seconds(s).

Model configuration	Training time	Writer-dependent test time	Writer-independent test time
Triplet loss $m = 0.1$ (multi-task)	0d:2h:41m:50s	0d:1h:1m:46s	0d:11h:24m:24s
Triplet loss $m = 0.8$ (multi-task)	0d:2h:41m:4s	0d:0h:52m:8s	1d:0h:41m:11s
NT-Xent loss $\tau = 0.01$ (multi-task)	0d:2h:29m:20s	0d:1h:7m:33s	0d:14h:47m:0s
NT-Xent loss $\tau = 0.5$ (multi-task)	0d:2h:28m:44s	0d:0h:53m:20s	0d:23h:58m:43s
SigNet	0d:0h:24m:39s	0d:3h:32m:32s	0d:23h:50m:22s
SigNet (Synthetic)	0d:0h:43m:53s	0d:1h:2m:17s	0d:20h:39m:3s

Table 30 – Runtimes for the training of models with proposed continual learning framework in days(d), hours(h), minutes(m) and seconds(s).

Model configuration	Training time
SigNet-CL (DeepInversion + Competition) $\lambda_p = 0.0$	0d:4h:42m:39s
SigNet-CL (DeepInversion + Competition) $\lambda_p = 1.0$	0d:4h:41m:13s
SigNet-CL (DeepInversion + Competition) $\lambda_p = 1.7$	0d:4h:38m:22s
MT-SigNet-CL (DeepInversion + Competition) $\lambda_p = 1.0$	0d:14h:4m:20s
MT-SigNet-CL (DeepInversion + Competition) $\lambda_p = 1.7$	1d:0h:4m:5s
MT-ResNet-152-CL (DeepInversion + Competition) $\lambda_p = 1.0$	3d:18h:58m:41s
MT-ResNet-152-CL (DeepInversion + Competition) $\lambda_p = 1.7$	3d:19h:16m:25s
MT-ViT-CL (DeepInversion + Competition) $\lambda_p = 1.0$	6d:21h:11m:28s

Table 31 – Runtimes for the writer-dependent testing on the real validation set $\mathcal{V}_{vr}$ and on the synthetic validation set $\mathcal{V}_{vs}$ in days(d), hours(h), minutes(m) and seconds(s).

Model configuration	Writer-dependent test time with real data	Writer-dependent test time with synthetic data
SigNet-CL (DeepInversion + Competition) $\lambda_p = 0.0$	0d:0h:32m:49s	0d:6h:37m:59s
SigNet-CL (DeepInversion + Competition) $\lambda_p = 1.0$	0d:0h:30m:1s	0d:8h:40m:9s
SigNet-CL (DeepInversion + Competition) $\lambda_p = 1.7$	0d:0h:29m:11s	0d:8h:40m:23s

Table 32 – Runtimes for the writer-independent testing on the real validation set $\mathcal{V}_{vr}$ and on the synthetic validation set $\mathcal{V}_{vs}$ in days(d), hours(h), minutes(m) and seconds(s).

Model configuration	Writer-independent test time with real data	Writer-independent test time with synthetic data
SigNet-CL (DeepInversion + Competition) $\lambda_p = 0.0$	0d:7h:55m:43s	0d:21h:19m:41s
SigNet-CL (DeepInversion + Competition) $\lambda_p = 0.5$	0d:6h:5m:46s	1d:1h:31m:40s
SigNet-CL (DeepInversion + Competition) $\lambda_p = 1.0$	0d:5h:18m:33s	1d:4h:45m:10s

A.2.3 Generalization Experiments

The runtimes of the generalization measurement experiments (Section 3.4.3) of the models proposed in Chapter 3 are listed in Tables 33, and 34. The execution times of the generalization measurement experiments (Section 4.4.3) of the models proposed in Chapter 4 are listed in Tables 35, 36, 37, 38, 39, and 40. In these reported results, the users of the GPDS-150S, GPDS-300S, GPDS-300, CEDAR, BHSig-B and BHSig-H datasets are tested using 12 reference signatures. Users of the MCYT-75 dataset are tested using 10 reference signatures.

Table 33 – Testing times for the MT-SigNet (Triplet) model.

Dataset	Writer-dependent test	Writer-independent test
GPDS-300S	0d:0h:43m:21s	0d:12h:47m:28s
GPDS-150S	0d:0h:11m:38s	0d:11h:45m:13s
CEDAR	0d:0h:1m:40s	0d:0h:3m:54s
MCYT-75	0d:0h:2m:37s	0d:0h:3m:52s

Table 34 – Testing times for the MT-SigNet (NT-Xent) model.

Dataset	Writer-dependent test	Writer-independent test
GPDS-300S	0d:0h:45m:9s	0d:16h:21m:11s
GPDS-150S	0d:0h:10m:44s	0d:15h:8m:38s
CEDAR	0d:0h:1m:48s	0d:0h:4m:27s
MCYT-75	0d:0h:2m:52s	0d:0h:4m:4s

Table 35 – Testing times for the SigNet model.

Dataset	Writer-dependent test	Writer-independent test
GPDS-150S	0d:0h:24m:32s	1d:5h:8m:14s
GPDS-300S	0d:1h:31m:35s	1d:13h:57m:14s
GPDS-300	0d:2h:50m:18s	0d:2h:8m:46s
CEDAR	0d:0h:1m:36s	0d:0h:3m:50s
MCYT-75	0d:0h:2m:31s	0d:0h:3m:19s
BHSig-B	0d:0h:4m:30s	0d:0h:14m:23s
BHSig-H	0d:0h:17m:24s	0d:0h:46m:7s

Table 36 – Testing times for the SigNet (Synthetic) model.

Dataset	Writer-dependent test	Writer-independent test
GPDS-150S	0d:0h:11m:38s	0d:22h:14m:5s
GPDS-300S	0d:0h:43m:7s	0d:23h:11m:18s
GPDS-300	0d:0h:53m:12s	1d:2h:45m:17s
CEDAR	0d:0h:1m:52s	0d:0h:4m:37s
MCYT-75	0d:0h:3m:2s	0d:0h:4m:21s
BHSig-B	0d:0h:5m:24s	0d:0h:14m:8s
BHSig-H	0d:0h:14m:26s	0d:0h:30m:4s

Table 37 – Testing times for the SigNet-F model.

Dataset	Writer-dependent Test	Writer-independent test
GPDS-150S	0d:0h:31m:16s	4d:7h:52m:10s
GPDS-300S	0d:2h:3m:55s	5d:7h:43m:0s
GPDS-300	0d:0h:56m:40s	0d:12h:18m:42s
CEDAR	0d:0h:3m:10s	0d:0h:9m:22s
MCYT-75	0d:0h:2m:3s	0d:0h:6m:16s
BHSig-B	0d:0h:5m:12s	0d:0h:19m:22s
BHSig-H	0d:0h:22m:39s	0d:1h:19m:16s

Table 38 – Testing times for the MT-SigNet-CL (DeepInversion + Competition) model.

Dataset	Writer-dependent test	Writer-independent test
GPDS-150S	0d:0h:8m:59s	0d:7h:45m:31s
GPDS-300S	0d:0h:45m:4s	0d:8h:56m:10s
GPDS-300	0d:0h:42m:28s	0d:8h:10m:1s
CEDAR	0d:0h:0m:50s	0d:0h:3m:56s
MCYT-75	0d:0h:1m:44s	0d:0h:4m:48s
BHSig-B	0d:0h:3m:27s	0d:0h:20m:43s
BHSig-H	0d:0h:9m:7s	0d:0h:28m:11s

Table 39 – Testing times for the MT-ResNet-152-CL (DeepInversion + Competition) model.

Dataset	Writer-dependent test	Writer-independent test
GPDS-150S	0d:0h:9m:5s	0d:7h:51m:41s
GPDS-300S	0d:0h:45m:8s	0d:8h:38m:43s
GPDS-300	0d:0h:38m:40s	0d:8h:22m:5s
CEDAR	0d:0h:1m:31s	0d:0h:4m:3s
MCYT-75	0d:0h:6m:18s	0d:0h:5m:27s
BHSig-B	0d:0h:2m:49s	0d:0h:16m:38s
BHSig-H	0d:0h:11m:30s	0d:0h:51m:35s

Table 40 – Testing times for the MT-ViT-CL (DeepInversion + Competition) model.

Dataset	Writer-dependent test	Writer-independent test
GPDS-150S	0d:0h:11m:20s	0d:9h:53m:25s
GPDS-300S	0d:0h:44m:23s	0d:11h:55m:42s
GPDS-300	0d:0h:34m:13s	0d:11h:0m:2s
CEDAR	0d:0h:1m:13s	0d:0h:5m:16s
MCYT-75	0d:0h:2m:32s	0d:0h:7m:30s
BHSig-B	0d:0h:8m:42s	0d:0h:14m:59s
BHSig-H	0d:0h:33m:44s	0d:0h:45m:48s

APPENDIX B – MODEL PARAMETERS

In Table 41 is listed the number of total parameters and memory size of the models studied in this thesis.

Table 41 – Number of parameters and memory size of the studied models in this thesis.

Model	Total parameters	Parameters size (MB)
SigNet	16,880,179	67.52
SigNet (Synthetic)	19,890,160	79.56
MT-SigNet (Triplet)	15,792,160	63.17
MT-SigNet (NT-Xent)	15,792,160	63.17
MT-SigNet-CL (DeepInversion + Competition)	33,169,729	132.68
MT-ResNet-152-CL (DeepInversion + Competition)	83,911,905	335.65
MT-ViT-CL (DeepInversion + Competition)	108,497,697	433.62