IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 3 August 2023, accepted 28 August 2023, date of publication 31 August 2023, date of current version 7 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3310819

RESEARCH ARTICLE

Auxiliary Cross-Modal Representation Learning With Triplet Loss Functions for Online Handwriting Recognition

FELIX OTT[1,2,3], (Member, IEEE), DAVID RÜGAMER[2,3], LUCAS HEUBLEIN[1], BERND BISCHL[2,3], AND CHRISTOPHER MUTSCHLER[1]

[1]Fraunhofer IIS, Fraunhofer Institute for Integrated Circuits, 90411 Nuremberg, Germany
[2]LMU Munich, 80539 Munich, Germany
[3]Munich Center for Machine Learning (MCML), 80539 Munich, Germany

Corresponding author: Felix Ott (felix.ott@iis.fraunhofer.de)

This work was supported in part by the Research Program Human–Computer-Interaction through the Project "Schreibtrainer" under Grant 16SV8228; and in part by the Bavarian Ministry for Economic Affairs, Infrastructure, Transport and Technology through the Center for Analytics-Data-Applications (ADA-Center) within the Framework of "BAYERN DIGITAL II." The work of David Rügamer was supported by the Federal Ministry of Education and Research (BMBF) of Germany under Grant 01IS18036A.

ABSTRACT Cross-modal representation learning learns a shared embedding between two or more modalities to improve performance in a given task compared to using only one of the modalities. Cross-modal representation learning from different data types – such as images and time-series data (e.g., audio or text data) – requires a deep metric learning loss that minimizes the distance between the modality embeddings. In this paper, we propose to use the contrastive or triplet loss, which uses positive and negative identities to create sample pairs with different labels, for cross-modal representation learning between image and time-series modalities (CMR-IS). By adapting the triplet loss for cross-modal representation learning, higher accuracy in the main (time-series classification) task can be achieved by exploiting additional information of the auxiliary (image classification) task. We present a triplet loss with a dynamic margin for single label and sequence-to-sequence classification tasks. We perform extensive evaluations on synthetic image and time-series data, and on data for offline handwriting recognition (HWR) and on online HWR from sensor-enhanced pens for classifying written words. Our experiments show an improved classification accuracy, faster convergence, and better generalizability due to an improved cross-modal representation. Furthermore, the more suitable generalizability leads to a better adaptability between writers for online HWR.

INDEX TERMS Contrastive learning, cross-modal retrieval, online handwriting recognition, optical character recognition, representation learning, sensor-enhanced pen, sequence-based learning, triplet learning.

I. INTRODUCTION

Cross-modal retrieval (CMR) such as cross-modal representation learning [1] for learning across two or more modalities (i.e., image, audio, text and 3D data) has recently garnered substantial interest from the machine learning community. CMR can be applied in a wide range of applications, such as multimedia management [2] and identification [3]. Extracting information from several modalities and adapting the domain with cross-modal learning allows using the information in all domains [4]. Cross-modal representation learning, however, remains challenging due to the *heterogeneity gap* (i.e., inconsistent representation forms of different modalities) [5].

A limitation of cross-modal representation learning is that many approaches require the availability of all modalities at inference time. Image-to-caption CMR methods solve

The associate editor coordinating the review of this manuscript and approving it for publication was Jon Atli Benediktsson.

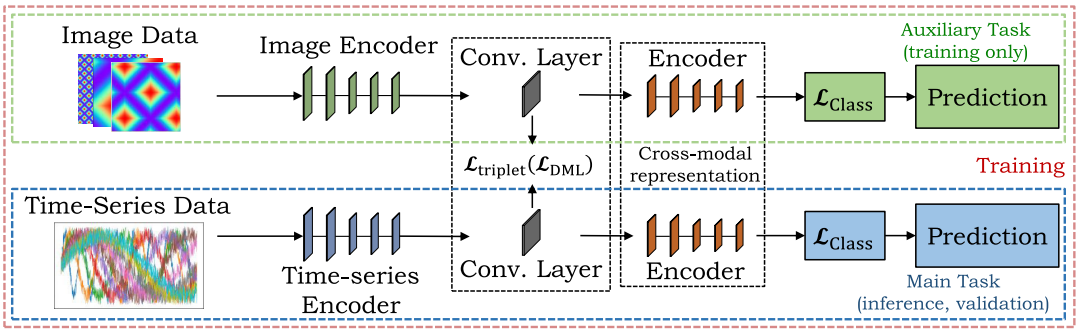


FIGURE 1. Method overview: Cross-modal representation learning between image and time-series data using the triplet loss based on metric learning functions to improve the time-series classification task.

this via a separate encoder [6], [7]. However, in many applications, certain data sources are only available during training by means of elaborate laboratory setups [8]. For instance, consider a human pose estimation task that uses inertial sensors together with color videos during training, where a camera setup might not be available at inference time due to bad lighting conditions or other application-specific restrictions. Here, a model that allows inference on only the main modality is required, while auxiliary modalities may only be used to improve the training process (as they are not available at inference time) [9]. *Learning using privileged information* [10] is one approach in the literature that describes and tackles this problem. During training, in addition to X, it is assumed that additional *privileged information* X^* is available. However, this *privileged information* is not present in the inference stage [11].

For cross-modal representation learning, we need a deep metric learning technique that aims to transform training samples into feature embeddings that are close for samples that belong to the same class and far apart for samples from different classes [12]. As deep metric learning requires no model update (simply fine-tuning for training samples of new classes), deep metric learning is an often applied approach for continual learning [13]. Typical deep metric learning methods use not only simple distances (e.g., Euclidean distance), but also highly complex distances (e.g., canonical correlation analysis [4] and maximum mean discrepancy [14]). While cross-modal representation learning learns representations from all modalities, single-modal learning commonly uses pair-wise learning. The triplet loss [15] selects a positive and negative triplet pair for a corresponding anchor and forces the positive pair distance to be smaller than the negative pair distance. While research of triplet selection for single-modal classification is very advanced [9], [13], [16], [17], [18], [19], [20], [21], [22], pair-wise selection for cross-modal representation learning has mainly been investigated for specific applications [2], [23], [24], i.e., visual semantic embeddings [7], [25], [26], [27].

One exemplary application for cross-modal learning is handwriting recognition (HWR), which can be categorized into offline and online HWR. Offline HWR – such as optical character recognition (OCR) – concerns only analysis of the visual representation of handwriting and cannot be applied for real-time recognition applications [28]. In contrast, online HWR works on different types of spatio-temporal signals and can make use of temporal information, such as writing speed and direction [29]. As an established real-world application of online HWR, many recording systems make use of a stylus pen together with a touch screen surface [30]. There also exist prototypical systems for online HWR when writing on paper [31], [32], [33], [34], but these are not yet suitable for real-world applications. However, a novel sensor-enhanced pen based on inertial measurement units (IMUs) may enable new online HWR applications for writing on normal paper. This pen has previously been used for single character [16], [35], [36], [37] and sequence [38] classification. However, the accuracy of previous online HWR methods is limited, due to the following reasons: (1) The size of datasets is limited, as recording larger amounts of data is time-consuming. (2) Extracting important spatio-temporal features is important. (3) Training a writer-independent classifier is challenging, as different writers can have notably different writing styles. (4) Evaluation performance drops for under-represented groups, i.e., left-handed writers. (5) The model overfits to seen words that can be addressed with generated models. A possible solution is to combine datasets of different modalities using cross-modal representation learning to increase generalizability. In this work, we combine offline HWR from generated images (i.e., OCR) and online HWR from sensor-enhanced pens by learning a common representation between both modalities. The aim is to integrate information on OCR – i.e., typeface, cursive or printed writing, and font thickness – into the online HWR task – i.e., writing speed and direction [39].

Our Contribution: Models that use rich data (e.g., images) usually outperform those that use a less rich modality (e.g., time-series). We therefore propose to train a shared representation using the triplet loss between pairs of image and time-series data to learn a cross-modal representation between both modality embeddings (cf. Figure 1). This allows for improving the accuracy of single-modal inference in the main task. Cross-modal learning between images and time-series data is rare. Furthermore, we propose a novel dynamic margin for the triplet loss based on the Edit distance.

We prove the efficacy of our metric learning-based triplet loss for cross-modal representation learning both with simulated data and in a real-world application. More specifically, our proposed cross-modal representation learning technique 1) improves the multivariate time-series classification accuracy and convergence, 2) results in a small time-series-only network independent from the image modality while allowing for fast inference, and 3) has better generalizability and adaptability [5]. Our approach shows that the recent methods ScrabbleGAN [40] and OrigamiNet [41] are applicable in the real-world setup of offline HWR to enhance the online HWR task. We provide an extensive overview and technical comparison of related methods. Code and datasets are available upon publication.[1]

The paper is organized as follows. Section II discusses related work followed by the mathematical foundation of our method in Section III. The methodology is described in Section IV and the results are discussed in Section V.

II. RELATED WORK

In this section, we discuss related work – particularly, methods of offline HWR (in Section II-A) and online HWR (in Section II-B). We summarize approaches for learning a cross-modal representation from different modalities (in Section II-C), pairwise and triplet learning (in Section II-D), and deep metric learning (in Section II-E) to minimize the distance between feature embeddings.

A. OFFLINE HANDWRITING RECOGNITION

In the following, we give a brief overview of offline HWR methods to select a suitable lexicon and language model-free method. For an overview of offline and online HWR datasets, see [29] and [42]. For a more detailed overview, see Table 7 in the Appendix B. Methods for offline HWR range from hidden Markov models (HMMs) – such as [43], [44], [45], [46], and [47] – to deep learning techniques that became predominant in 2014, such as convolutional neural networks (CNNs) as the methods by [48] and [49]. Furthermore, temporal convolutional networks (TCNs) employ the temporal context of the handwriting – such as the methods [50], [51]. More prominent became recurrent neural networks (RNNs) including long short-term memories (LSTMs), bidirectional LSTMs (BiLSTMs) [52], [53], [54], [55], and multidimensional RNNs [56], [57], [58], [59], [60], [61], [62]. These sequential architectures are perfect to fit text lines, due to the probability distributions over sequences of characters, and due to the inherent temporal aspect of text [63]. Pham et al. [64] showed that the performance of LSTMs can be greatly improved using dropout. The authors in [65] introduced the BiLSTM layer in combination with the connectionist temporal classification (CTC) loss. CTC adds up over the probability of possible alignments of the input to the target sequences, producing a loss value which is differentiable with respect to each input node. Additionally, th work by [66] proposed a CNN+BiLSTM architecture that uses the CTC loss. GCRNN [67] combines a convolutional encoder (aiming for generic and multilingual features) and a BiLSTM decoder predicting character sequences. Further methods that combine CNNs with RNNs are [68], [69], and [70], while BiLSTMs are utilized in [71] and [72].

The most recent method based on CNNs is the gated text recognizer [73] that aims to automate the feature extraction from raw input signals with a minimum required domain knowledge. The fully convolutional network without recurrent connections is trained with the CTC loss. Thus, the gated text recognizer module can handle arbitrary input sizes and can recognize strings with arbitrary lengths. This module has been used for OrigamiNet [41] which is a segmentation-free multi-line or full-page recognition system. OrigamiNet yields state-of-the-art results on the IAM-OffDB dataset, and shows improved performance of gated text recognizer over VGG and ResNet26. Hence, we use the gated text recognizer module as our visual feature encoder for offline HWR.

Recent methods are generative adversarial networks (GANs) and Transformers. The first approach by [74] was a method to synthesize online data based on RNNs. The technique HWGAN by [75] extends this method by adding a discriminator $\mathcal{D}$. DeepWriting [76] is a GAN that is capable of disentangling style from content and thus making digital ink editable. The authors in [77] proposed a method to generate handwriting based on a specific author with learned parameters for spacing, pressure, and line thickness. Alonso et al. [78] used a BiLSTM to obtain an embedding of the word to be rendered and added an auxiliary network as a recognizer $\mathcal{R}$. The model is trained with a combination of an adversarial loss and the CTC loss. ScrabbleGAN by [40] is a semi-supervised approach that can arbitrarily generate many images of words with arbitrary length from a generator $\mathcal{G}$ to augment handwriting data and uses a discriminator $\mathcal{D}$ and recognizer $\mathcal{R}$. The paper proposes results for original data with random affine augmentation using synthetic images and refinement.

B. ONLINE HANDWRITING RECOGNITION

Motion-based handwriting [31] and air-writing [79] from sensor-enhanced devices have been extensively investigated. While such motions are spacious, the hand and pen motions for writing on paper are comparatively small-scale [80]. Research for classifying text from sensor-enhanced pens has recently attracted substantial interest. He et al. [81] use acceleration and audio data of handwritten actions for character recognition. Furthermore, recent publications came up with similar developments that are only prototypical, for example, the works proposed by [82], [83], and [84]. Hence, there is already a lot of interest and future technical advancements will further boost the classification performance of online HWR methods. The novel sensor-enhanced pen based on IMUs [35] enables new applications for writing on paper.

[1]Code and datasets: https://www.iis.fraunhofer.de/de/ff/lv/dataanalytics/anwproj/schreibtrainer/onhw-dataset.html

Note that this pen is a finished product and is commercially available. Data collection and processing is straightforward and allows applications to be easy to implement in real-world. Ott et al. [35] published the OnHW-chars dataset containing single characters. Kla et al. [37] evaluated the aleatoric and epistemic uncertainty to show the domain shift between right- and left-handed writers. Reference [16] reduced this domain shift by adapting feature embeddings based on transformations from optimal transport techniques. In [85], the authors presented an approach for distributing the computational workload between a sensor pen and a mobile device (i.e., smartphone or tablet) for handwriting recognition, as interference on mobile devices leads to high system requirements. Ott et al. [36] reconstructed the trajectory of the pen tip for single characters written on tablets from IMU data and cameras pointing at the pen tip [86]. A more challenging task than single-character classification is the classification of sequences (i.e., words or equations). The authors in [38] proposed several sequence-based datasets and a large benchmark of convolutional, recurrent, and Transformer-based architectures, loss functions, and augmentation techniques. While [87] combined a binary random forest to classify the writing activity and a CNN for windows of single-label predictions, [88] highlighted the effectiveness of Transformers for classifying equations. Methods such as the one proposed by [89] cannot be applied to this online task, as these methods are designed for image-based (offline) HWR, and traditional methods such as [71] for online HWR are based on online trajectories written on tablets. Recently, Azimi et al. [90] evaluated further machine and deep learning models as well as deep ensembles on the single OnHW-chars dataset.

C. CROSS-MODAL REPRESENTATION LEARNING

For traditional methods that learn a cross-modal representation, a cross-modal similarity for the retrieval can be calculated with linear projections [91]. However, cross-modal correlation is highly complex, and hence, recent methods are based on a *modal-sharing network* to jointly transfer non-linear knowledge from a single modality to all modalities [12]. Huang et al. [5] use a *cross-modal network* between different modalities (image to video, text, audio and 3D models) and a *single-modal network* (shared features between images of source and target domains). They use two convolutional layers (similar to our proposed architecture) that allow the model to adapt by using more trainable parameters. However, while their auxiliary network uses the same modality, the auxiliary network of the proposed method in this paper is based on another modality. The work by [2] learns a cross-modal embedding between video frames and audio signals with graph clusters, but both modalities must be available at inference. Sarafianos et al. [3] proposed an image-text modality adversarial matching approach that learns modality-invariant feature representations, but their projection loss is only used for learning discriminative image-text embeddings. The authors in [9] propose a model for single-modal inference. However, they use image and depth modalities for person re-identification without a time-series component, which makes the problem considerably different. Lim et al. [8] handled multi-sensory modalities for 3D models only. For an overview of CMR, see [92]. An overview of relevant CMR methods is given in Table 8 in the Appendix C. With respect to the kind of the modality, the work by [16] and [93] is closest, while the applications in [16], [20], and [94] of handwriting recognition are relevant.

D. PAIRWISE AND TRIPLET LEARNING

Networks trained for a classification task can produce useful feature embeddings with efficient runtime complexity $\mathcal{O}(NC)$ per epoch, where N is the number of training samples and C is the number of classes. However, the classical cross-entropy (CE) loss is only partly useful for deep metric learning, as it ignores how close each point is to its class centroid (or how far apart each point is from other class centroids). CE variations (e.g., for face recognition) that learn angularly discriminative features have also been developed [95]. The *pairwise contrastive loss* [96] minimizes the distance between feature embedding pairs of the same class and maximizes the distance between feature embedding pairs of different classes depending on a margin parameter. The drawback is that the optimization of positive pairs is independent of negative pairs, but the optimization should force the distance between positive pairs to be smaller than negative pairs [13].

The *triplet loss* [97] addresses this by defining an anchor and a positive point as well as a negative point and forces the positive pair distance to be smaller than the negative pair distance by a certain margin. The runtime complexity of the triplet loss is $\mathcal{O}(N^3/C)$ and can be computationally challenging for large training sets. Hence, several approaches exist to reduce this complexity, such as hard or semi-hard triplet mining [15] and smart triplet mining [98]. Often, data evolve over time, and hence, [99] proposed a formulation of the triplet loss where the traditional static *margin* is superseded by a temporally adaptive maximum margin function. While the research by [21] and [94] combines the triplet loss with the CE loss, Guo et al. [100] use a triplet selection with L_2-normalization for language modeling, but considered all negative pairs for triplet selection with fixed similarity intensity parameter. The proposed method uses a triplet loss with a dynamic margin together with a novel word-level triplet selection. The TNN-C-CCA [101] also uses the triplet loss on embeddings between an anchor from audio data and positive and negative samples from visual data and the cosine similarity for the final representation comparison. In image-to-caption CMR tasks, the most common design is separated encoders that allow the separated inference without the other modality [6], [7]. We choose a similar separate cross-modal encoder for single-modal inference. CrossATNet [102], another triplet loss-based method that

uses single class labels, defines class sketch instances as the anchor, the same class image instance as the positive sample, and a different class image instance as the negative sample. While the previous methods are based on a triplet selection method using single-label classification, related work exists for using the triplet loss for sequence-based classification (i.e., from texts) [103], [104], [105], [106]. To the best of our knowledge, no approach so far has used triplet-based cross-modal learning based on the Edit distance between words. Most relevant are the works by [7], [9], [102], [107], and [108] that use the triplet loss, but without a dynamic margin.

E. DEEP METRIC LEARNING

As deep metric learning is a very broad and advanced field, only the most related work is described here. For an overview of deep metric learning, see Musgrave et al. [109]. Most of the related work uses the Euclidean metric as distance loss, although the triplet loss can be defined based on any other (sub-)differentiable distance metric. Wan et al. [20] proposed a method for offline signature verification based on a dual triplet loss that uses the Euclidean space to project an input image to an embedding function. While Rantzsch et al. [110] use the Euclidean metric to learn the distance between feature embeddings, the authors in [94] use the Cosine similarity. Hermans et al. [111] state that using the *non-squared* Euclidean distance is more stable, while the *squared* distance made the optimization more prone to collapsing. Recent methods extend the canonical correlation analysis (CCA) [4] that learns linear projection matrices by maximizing pairwise correlation of cross-modal data. To share information between the same modality (i.e., images), the maximum mean discrepancy (MMD) [14] is typically minimized.

III. METHODOLOGICAL BACKGROUND

We define the problem of cross-mdoal representation learning and present deep metric learning loss functions in Section III-A. In Section III-B, we propose the triplet loss for cross-modal learning.

A. CROSS-MODAL RETRIEVAL FOR TIME-SERIES AND IMAGE CLASSIFICATION

A multivariate time-series $\mathbf{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_m\} \in \mathbb{R}^{m \times l}$ is an ordered sequence of $l \in \mathbb{N}$ streams with $\mathbf{u}_i = (u_{i,1}, \ldots, u_{i,l}), i \in \{1, \ldots, m\}$, where $m \in \mathbb{N}$ is the length of the time-series. The multivariate time-series training set is a subset of the array $\mathcal{U} = \{\mathbf{U}_1, \ldots, \mathbf{U}_{n_U}\} \in \mathbb{R}^{n_U \times m \times l}$, where n_U is the number of time-series. Let $\mathbf{X} \in \mathbb{R}^{h \times w}$ with entries $x_{i,j} \in [0, 255]$ represent an image from the image training set. The image training set is a subset of the array $\mathcal{X} = \{\mathbf{X}_1, \ldots, \mathbf{X}_{n_X}\} \in \mathbb{R}^{n_X \times h \times w}$, where n_X is the number of time-series. The aim of joint multivariate time-series and image classification tasks is to predict an unknown class label $y \in \Omega$ for single class prediction or $\mathbf{y} \in \Omega$ for sequence prediction for a given multivariate time-series or image (see also Section IV-B). The time-series samples denote the main training data, while the image samples represent the privileged information that is not used for inference. In addition to good prediction performance, the goal is to learn representative embeddings $f_c(\mathbf{U})$ and $f_c(\mathbf{X}) \in \mathbb{R}^{q \times t}$ to map multivariate time-series and image data into a feature space $\mathbb{R}^{q \times t}$, where f_c is the output of the convolutional layer(s) $c \in \mathbb{N}$ of the latent representation and $q \times t$ is the dimension of the layer output.

We force the embedding to live on the $q \times t$-dimensional hypersphere by using `softmax` – i.e., $||f_c(\mathbf{U})||_2 = 1$ and $||f_c(\mathbf{X})||_2 = 1 \, \forall c$ (see [112]). In order to obtain a small distance between the embeddings $f_c(\mathbf{U})$ and $f_c(\mathbf{X})$, we minimize deep metric learning functions $\mathcal{L}_{\text{DML}}(f_c(\mathbf{X}), f_c(\mathbf{U}))$. Well-known deep learning metric are the distance-based mean squared error (MSE) $\mathcal{L}_{\text{MSE}}$, the spatio-temporal cosine similarity (CS) $\mathcal{L}_{\text{CS}}$, the Pearson correlation (PC) $\mathcal{L}_{\text{PC}}$, and the distribution-based Kullback-Leibler (KL) divergence $\mathcal{L}_{\text{KL}}$. In our experiments, we additionally evaluate the kernalized maximum mean discrepancy (kMMD) $\mathcal{L}_{\text{kMMD}}$, Bray Curtis (BC) $\mathcal{L}_{\text{BC}}$, and Poisson $\mathcal{L}_{\text{PO}}$ losses. We study their performance in Section V. A combination of classification and cross-modal representation learning losses can be realized by dynamic weight averaging [113] as a multi-task learning approach that performs dynamic task weighting over time (see Appendix D).

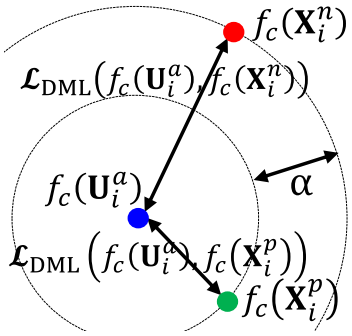
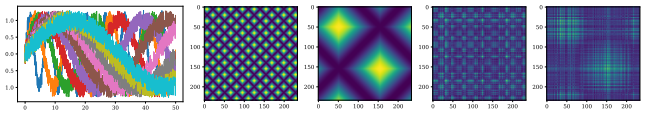


FIGURE 2. Triplet pair.

B. CONTRASTIVE LEARNING AND TRIPLET LOSS

While the training with the previous loss functions uses inputs where the image and multivariate time-series have the same label, pairs with similar but different labels can improve the training process. This can be achieved using the triplet loss [15], which enforces a margin between pairs of image and multivariate time-series data with the same identity to all other different identities. As a consequence, the convolutional output for one and the same label lives on a manifold, while still enforcing the distance – and thus, discriminability – to other identities.

Therefore, we seek to ensure that the embedding of the multivariate time-series $\mathbf{U}_i^a$ (*anchor*) of a specific label is closer to the embedding of the image $\mathbf{X}_i^p$ (*positive*) of the same label than it is to the embedding of any image $\mathbf{X}_i^n$ (*negative*) of another label (see Figure 2). Thus, we want



(a) Signal data. (b) Without noise. (c) With high noise.

FIGURE 3. Synthetic signal data (a) for 10 classes, and image data (b-c) for classes 0 (left) and 6 (right).

the following inequality to hold for all training samples $\left(f_c(\mathbf{U}_i^a), f_c(\mathbf{X}_i^p), f_c(\mathbf{X}_i^n)\right) \in \Phi$:

$$\mathcal{L}_{\text{DML}}\left(f_c(\mathbf{U}_i^a), f_c(\mathbf{X}_i^p)\right) + \alpha < \mathcal{L}_{\text{DML}}\left(f_c(\mathbf{U}_i^a), f_c(\mathbf{X}_i^n)\right), \quad (1)$$

where $\mathcal{L}_{\text{DML}}\left(f_c(\mathbf{X}), f_c(\mathbf{U})\right)$ is a deep metric learning loss, α is a margin between positive and negative pairs, and Φ is the set of all possible triplets in the training set. The *contrastive loss* minimizes the distance of the anchor to the positive sample and separately maximizes the distance to the negative sample. Instead, based on (1), we can formulate a differentiable loss function - the *triplet loss* - that we can use for optimization:

$$\mathcal{L}_{\text{trpl,c}}(\mathbf{U}^a, \mathbf{X}^p, \mathbf{X}^n) = \sum_{i=1}^{N} \max\Big[\mathcal{L}_{\text{DML}}\left(f_c(\mathbf{U}_i^a), f_c(\mathbf{X}_i^p)\right) - \mathcal{L}_{\text{DML}}\left(f_c(\mathbf{U}_i^a), f_c(\mathbf{X}_i^n)\right) + \alpha, 0\Big], \quad (2)$$

where $c \in \mathbb{N}$.[2] Selecting negative samples that are too close to the anchor (in relation to the positive sample) can cause slow training convergence. Hence, triplet selection must be handled carefully and with consideration for each specific application [13]. We choose negative samples based on the class distance (single labels) and on the Edit distance (sequence labels) (see Section IV-B).

IV. METHOD

We now demonstrate the efficacy of our proposal. In Section IV-A, we generate sinusoidal time-series with introduced noise (main task) and compute the corresponding Gramian angular summation field with different noise parameters (auxiliary task) (see Figure 1). In Section IV-B, we combine online (inertial sensor signals, main task) and offline data (visual representations, auxiliary task) for HWR with sensor-enhanced pens. This task is particularly challenging, due to different data representations based on images and multivariate time-series data. For both applications, our approach allows to only use the main modality (i.e., multivariate time-series) for inference. We further analyze and evaluate different deep metric learning functions to minimize the distance between the learned embeddings.

A. CROSS-MODAL LEARNING ON SYNTHETIC DATA

We first investigate the influence of the triplet loss for cross-modal learning between synthetic time-series and image-based data as a sanity check. For this, we generate signal data of 1,000 timesteps with different frequencies for 10 classes (see Figure 3a) and add noise from a continuous uniform distribution $U(a, b)$ for $a = 0$ and $b = 0.3$. We use a recurrent CNN with the CE loss to classify these signals. From each signal without noise, we generate a Gramian angular summation field [114]. For classes with high frequencies, this results in a fine-grained pattern, and for low frequencies in a coarse-grained pattern. We generate Gramian angular summation fields with different added noise between $b = 0$ (Figure 3b) and $b = 1.95$ (Figure 3c). A small CNN classifies these images with the CE loss. To combine both networks, we train each signal-image pair with the triplet loss. As the frequency of the sinusoidal signal is closer for more similar class labels, the distance in the manifold embedding should also be closer. For each batch, we select negative sample pairs for samples with the class label $CL = 1 + \lfloor \frac{\max_e - e - 1}{25} \rfloor$ as the lower bound for the current epoch e and the maximum epoch $\max_e$. We set the margin α in the triplet loss separately for each batch such that $\alpha = \beta \cdot (CL_p - CL_n)$ depends on the positive CL_p and negative CL_n class labels of the batch and is in the range $[1, 5]$ with $\beta = 0.1$. The batch size is 100 and $\max_e = 100$. Appendix E provides further details. This combination of the CE loss with the triplet loss can lead to a mutual improvement of the utilization of the classification task and embedding learning.

B. CROSS-MODAL LEARNING FOR HWR

1) METHOD OVERVIEW

Figure 4 gives a method overview. The main task is online HWR to classify words written with a sensor-enhanced pen and represented by multivariate time-series of the different pen sensors. To improve the classification task with a better generalizability, the auxiliary network performs offline HWR based on an image input. We pre-train ScrabbleGAN [40] on the IAM-OffDB [115] dataset. For all time-series word labels, we then generate the corresponding image as the positive time-series-image pair. Each multivariate time-series and each image is associated with $\mathbf{y}$ – a sequence of L class labels from a pre-defined label set Ω with K classes. For our classification task, $\mathbf{y} \in \Omega^L$ describes words. The multivariate time-series training set is a subset of the array $\mathcal{U}$ with labels $\mathcal{Y}_U = \{\mathbf{y}_1, \ldots, \mathbf{y}_{n_U}\} \in \Omega^{n_U \times L}$. The image training set is a subset of the array $\mathcal{X}$, and the corresponding labels are $\mathcal{Y}_X = \{\mathbf{y}_1, \ldots, \mathbf{y}_{n_X}\} \in \Omega^{n_X \times L}$. Offline HWR techniques are based on Inception, ResNet34, or gated text recognizer [73] modules. The architecture of the online HWR method consists of an IMU encoder with three 1D convolutional layers of size 400, a convolutional layer of size 200, a max pooling and batch normalization, and a dropout of 20%. The online method is improved by sharing layers with a common representation by minimizing the distance of the feature embedding of the convolutional layers $c \in \{1, 2\}$ (integrated in both networks) with a shared loss $\mathcal{L}_{\text{shared,c}}$. We set the embedding size $\mathbb{R}^{q \times t}$ to 400×200. Both networks

[2] To have a larger number of trainable parameters in the latent representation with a greater depth, we evaluate one and two stacked convolutional layers, each trained with a shared loss $\mathcal{L}_{\text{trpl,c}}$.

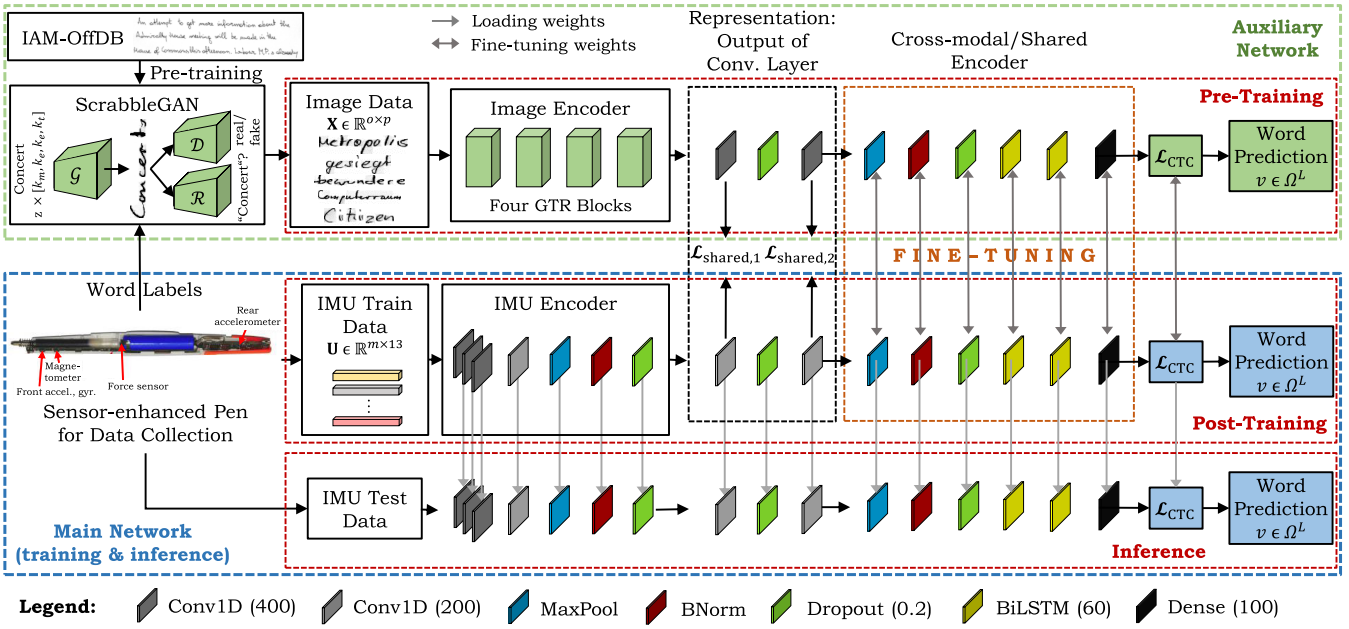


FIGURE 4. Detailed method overview: The middle pipeline consists of data recording with a sensor-enhanced pen, feature extraction of inertial multivariate time-series data, and word classification with CTC. We generate image data with the pre-trained ScrabbleGAN for corresponding word labels. The top pipeline (four gated text recognizer blocks) extracts features from images. The distances of the embeddings are minimized with the triplet loss and deep metric learning functions. The classification network with two BiLSTM layers are fine-tuned for the OnHW task for a cross-modal representation.

are trained with the connectionist temporal classification (CTC) [65] loss $\mathcal{L}_{\text{CTC}}$ to avoid pre-segmentation of the training samples by transforming the network outputs into a conditional probability distribution over label sequences.

2) DATASETS FOR ONLINE HWR

We make use of two word datasets proposed in [38]. These datasets are recorded with a sensor-enhanced pen that uses two accelerometers (3 axes each), one gyroscope (3 axes), one magnetometer (3 axes), and one force sensor at 100 Hz [35], [36]. One sample of size $m \times l$ represents an multivariate time-series of a written word of m timesteps from $l = 13$ sensor channels. One word is a sequence of small or capital characters (52 classes) or with mutated vowels (59 classes). The *OnHW-words500* dataset contains 25,218 samples where each of the 53 writers contributed the same 500 words. The *OnHW-wordsRandom* dataset contains 14,641 randomly selected words from 54 writers. For both datasets, 80/20 train/validation splits are available for writer-(in)dependent (WD/WI) tasks. We transform (zero padding, interpolation) all samples to 800 timesteps. For more information on the datasets, see [38].

3) IMAGE GENERATION FOR OFFLINE HWR

In order to couple the online time-series data with offline image data, we use a generative adversarial network (GAN) to arbitrarily generate many images. ScrabbleGAN [40] is a state-of-the-art semi-supervised approach that consists of a generator $\mathcal{G}$ that generates images of words with arbitrary length from an input word label, a discriminator $\mathcal{D}$, and a recognizer $\mathcal{R}$ that promotes style and data fidelity. While $\mathcal{D}$

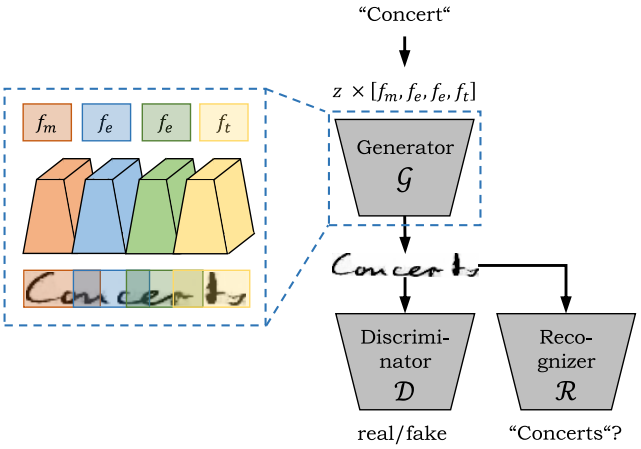
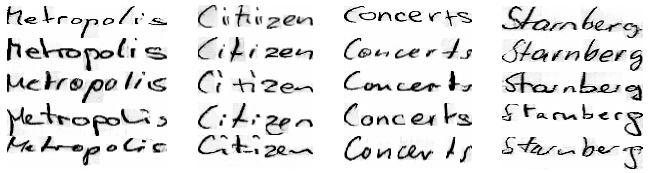


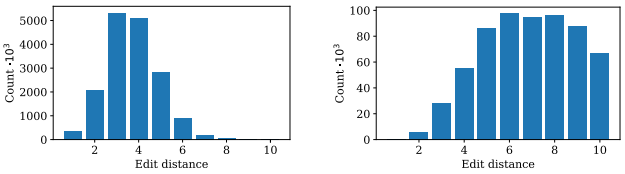
FIGURE 5. ScrabbleGAN concept by [40] of generating the word "Concerts".

promotes realistic-looking handwriting styles, $\mathcal{R}$ encourages the result to be readable. ScrabbleGAN minimizes a joint loss term $\mathcal{L} = \mathcal{L}_D + \lambda \mathcal{L}_R$ where $\mathcal{L}_D$ and $\mathcal{L}_R$ are the loss terms of $\mathcal{D}$ and $\mathcal{R}$, respectively, and the balance factor is λ. The generator $\mathcal{G}$ is designed such that each character is generated individually, using the property of the convolutions of overlapping receptive fields to account for the influence of nearby letters. Four character filters (k_m, k_e, k_e and k_t) are concatenated, multiplied by a noise vector z, and fed into a class-conditioned generator (see Figure 5). This allows for adjacent characters to interact and creates a smooth transition, e.g., enabling cursive text. The style of the image is controlled by a noise vector z given as the input to the network (being consistent for all characters of a word). The recognizer $\mathcal{R}$



(a) Metropolis. (b) Citizen. (c) Concerts. (d) Starnberg.

FIGURE 6. Overview of four generated words with ScrabbleGAN [40] with various text styles.



(a) OnHW-words500. (b) OnHW-wordsRandom.

FIGURE 7. Number image-time-series pairs dependent on substitutions.

discriminates between real and gibberish text by comparing the output of $\mathcal{R}$ to the one that was given as input to $\mathcal{G}$. $\mathcal{R}$ is trained only on real and labeled samples. $\mathcal{R}$ is inspired by CRNN [116] and uses the CTC [65] loss. The architecture of the discriminator $\mathcal{D}$ is inspired by BigGAN [117] consisting of four residual blocks and a linear layer with one output. $\mathcal{D}$ is fully convolutional, predicts the average of the patches, and is trained with a hinge loss [118]. We train ScrabbleGAN with the IAM-OffDB [115] dataset and generate three different datasets. Exemplary images are shown in Figure 6. First, we generate 2 million images randomly selected from a large lexicon (*OffHW-German*), and pre-train the offline HWR architectures. Second, we generate 100,000 images based on the same word labels for each of the OnHW-words500 and OnHW-wordsRandom datasets (*OffHW-words500*, *OffHW-wordsRandom*]) and fine-tune the offline HWR architectures.

4) METHODS FOR OFFLINE HWR

OrigamiNet [41] is a state-of-the-art multi-line recognition method using only unsegmented image and text pairs. Similar to OrigamiNet, our offline method is based on different encoder architectures with one or two additional 1D convolutional layers (each with filter size 200, `softmax` activation [94]) with 20% dropout for the latent representation, and a cross-modal representation decoder with BiLSTMs. For the encoder, we make use of Inception modules from GoogLeNet [119] and the ResNet34 [120] architectures, and we re-implement the newly proposed gated, fully-convolutional method termed the gated text recognizer [73]. See Appendix F for detailed information on the architectures. We train the networks on the generated OffHW-German dataset for 10 epochs and fine-tune on the OffHW-[500, wordsRandom] datasets for 15 epochs. For comparison with state-of-the-art techniques, we train OrigamiNet and compare with IAM-OffDB. For OrigamiNet, we apply interline spacing reduction via seam carving [121], resizing the images to 50% height, and random projective (rotating and resizing lines) and random elastic transform [122]. We augment the OffHW-German dataset with random width resizing and apply no augmentation for the OffHW-[words500, wordsRandom] datasets for fine-tuning.

5) OFFLINE/ONLINE CROSS-MODAL REPRESENTATION LEARNING

Our architecture for online HWR is based on [38]. The encoder extracts features of the inertial data and consists of three convolutional layers (each with filter size 400, `ReLU` activation) and one convolutional layer (filter size 200, `ReLU` activation), a max pooling, batch normalization and a 20% dropout layer. As for the offline architecture, the network then learns a latent representation with one or two convolutional layers (each with filter size 200, `softmax` activation) with 20% dropout and the same cross-modal representation decoder. The output of the convolutional layers of the latent representation are minimized with the $\mathcal{L}_{\text{shared,c}}$ loss. The layers of the common representation are fine-tuned based on the pre-trained weights of the offline technique. Here, two BiLSTM layers with 60 units each and `ReLU` activation extract the temporal context of the feature embedding. As for the baseline classifier, we train for 1,000 epochs. For evaluation, the main time-series network is independent of the image auxiliary network by using only the weights of the main network.

6) TRIPLET SELECTION

To ensure (fast) convergence, it is crucial to select triplets that violate the constraint from Equation 1. Typically, it is infeasible to compute the loss for all triplet pairs, or this leads to poor training performance (as poorly chosen pairs dominate hard ones). This requires an elaborate triplet selection [13]. We use the Edit distance to define the identity and select triplets. The Edit distance is the minimum number of substitutions S, insertions I, and deletions D required to change the sequences $\mathbf{d} = (d_1, \ldots, d_r)$ into $\mathbf{g} = (g_1, \ldots, g_z)$ with length r and z, respectively. We define two sequences with an Edit distance of 0 as the positive pair, and with an Edit distance larger than 0 as the negative pair. Based on preliminary experiments, we use only substitutions for triplet selection that lead to a higher accuracy compared to additional insertions and deletions (whereas these would also change the length difference of image and time-series pairs). We constrain $p - m/2$ (the difference in pixels p of the images and half the number of timesteps of the time-series) to be maximally ± 20. The goal is to achieve a small distance for positive pairs and a large distance for negative pairs that increases with a larger Edit distance (between 1 and 10). Furthermore, despite a limited number of word labels, there still exist a large number of image-time-series pairs per word label for every possible Edit distance (see Figure 7). For each batch, we search in a dictionary of negative sample pairs for samples with $Edit\_distance = 1 + \lfloor \frac{\max_e - e - 1}{100} \rfloor$ as the lower bound for the current epoch e and maximal epochs $\max_e$.

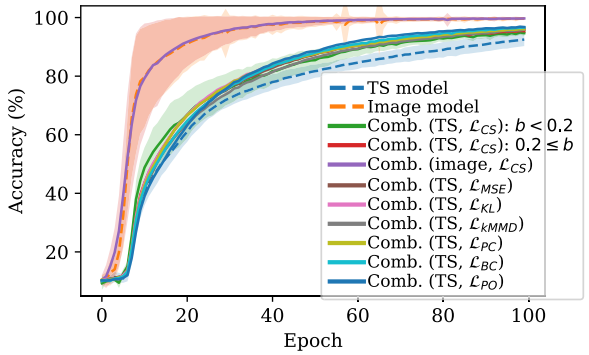


FIGURE 8. Accuracy of single- and cross-modal representation learning over all epochs.

For every label, we randomly pick one image. We let the margin α in the triplet loss vary for each batch such that $\alpha = \beta \cdot \overline{Edit\_distance}$ depends on the mean Edit distance of the batch and is in the range $[1, 11]$ with $\beta = 10^{-3}$ for MSE, $\beta = 0.1$ for CS and PC, and $\beta = 1$ for KL. The batch size is 100 and $\max_e = 1,000$.

V. EXPERIMENTAL RESULTS

a: HARDWARE AND TRAINING SETUP

For all experiments, we use Nvidia Tesla V100-SXM2 GPUs with 32 GB VRAM equipped with Core Xeon CPUs and 192 GB RAM. We use the vanilla Adam optimizer with a learning rate of 10^{-4}.

A. EVALUATION OF SYNTHETIC DATA

We train the time-series (TS) model 18 times with noise $b = 0.3$ and the combined model with the triplet loss for all 40 noise combinations $(b \in \{0, \ldots, 1.95\})$ with different deep metric learning functions. Figure 8 shows the validation accuracy averaged over all trainings as well as the combined cases separately for noise $b < 0.2$ and noise $0.2 \leq b < 2.0$ (for the $\mathcal{L}_{CS}$ loss). Table 1 summarizes the final classification results of all cases. The accuracy of the models that use only images and in combination with time-series during inference reach an accuracy of 99.7% (which can be seen as an unreachable upper bound for the TS-only models). The triplet loss improves the final TS baseline accuracy from 92.5% to 95.36% (averaged over all combinations), while combining TS and image data leads to a faster convergence. Conceptually similar to [14], we use the $\mathcal{L}_{kMMD}$ loss, which yields 95.83% accuracy. The $\mathcal{L}_{PC}$ (96.03%), $\mathcal{L}_{KL}$ (96.22%), $\mathcal{L}_{MSE}$ (96.25%), $\mathcal{L}_{BC}$ (96.62%), and $\mathcal{L}_{PO}$ (96.76%) loss functions can further improve the accuracy. We conclude that the triplet loss can be successfully used for cross-modal learning by utilizing negative identities.

B. EVALUATION OF HANDWRITING RECOGNITION

1) EVALUATION METRICS

A metric for sequence evaluation is the character error rate (CER), defined as $\text{CER} = \frac{S_c+I_c+D_c}{N_c}$, i.e., the Edit distance (the sum of character substitutions S_c, insertions I_c and deletions D_c) divided by the total number of characters in the set N_c. Similarly, the word error rate (WER) is defined as $\text{WER} = \frac{S_w+I_w+D_w}{N_w}$, which is computed with the sum of word operations S_w, I_w and D_w, divided by the number of words in the set N_w.

TABLE 1. Comparison of single- and cross-modal representation learning.

Method	Accuracy (%)
TS model	92.50
Combined (TS, $\mathcal{L}_{CS}$)	95.36
Combined (image, $\mathcal{L}_{CS}$)	99.70
Combined (TS, $\mathcal{L}_{MSE}$)	96.25
Combined (TS, $\mathcal{L}_{KL}$)	96.22
Combined (TS, $\mathcal{L}_{kMMD}$)	95.83
Combined (TS, $\mathcal{L}_{PC}$)	96.03
Combined (TS, $\mathcal{L}_{BC}$)	96.62
Combined (TS, $\mathcal{L}_{PO}$)	**96.76**

2) EVALUATION OF OFFLINE HWR METHODS

Table 2 shows offline HWR results on our generated OffHW-German dataset and on the IAM-OffDB [115] dataset. ScrabbleGAN [40] yields a WER of 23.61% on the IAM-OffDB dataset, while OrigamiNet [41] achieves a CER of 4.70% with 12 gated text recognizer modules. While OrigamiNet is trained for the multi-line classification, which is an easier task (as the image of the paragraph does not have to be segmented into lines), we trained OrigamiNet on single-lines with zero padding, which is closer to the OffHW-German dataset. While the images for the multi-line task are of approximately similar lengths, the image lengths of the single-line task varies strongly, and hence, zero padding has a high influence on the model performance, resulting in a CER of 15.67%. While [41] did not propose WER results, OrigamiNet yields only a WER of 90.40%. This problem does not appear for the OffHW-German dataset, as the dataset contains only single words with similar lengths. With our own implementation of four gated text recognizer modules and one convolutional layer for the common representation, our model achieves similar results. As the training takes more than one day for one epoch on the large OffHW-German dataset, we train OrigamiNet with four gated text recognizer modules, and achieve 0.11% CER on the generated dataset and 15.67% on the IAM-OffDB dataset. All our models yield low error rates on the generated OffHW-German dataset. Our approach with gated text recognizer blocks outperforms (0.24% to 0.44% CER) the models with Inception [119] (1.17% CER) and ResNet [120] (1.24% CER). OrigamiNet achieves the lowest error rates of 1.50% WER and 0.11% CER. Four gated text recognizer blocks yield the best results at a significantly lower training time compared to six or eight blocks. We fine-tune the model with four gated text recognizer blocks for one and two convolutional layers and achieve notably low error rates between 0.22% to 0.76% CER, and between 0.85% to 2.95% WER on the OffHW-[words500, wordsRandom] datasets (see Table 3). While results for OffHW-wordsRandom are

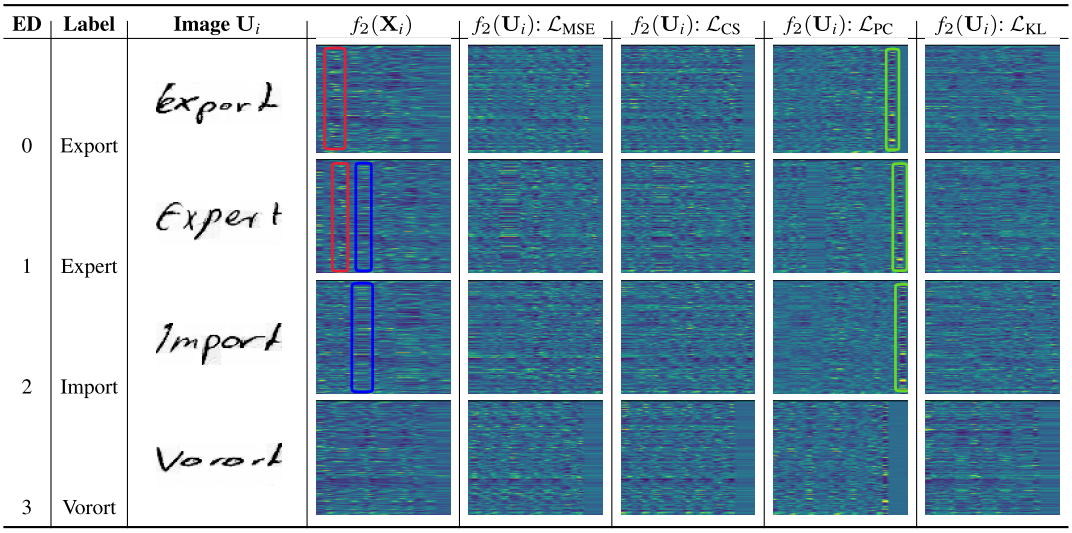
TABLE 2. Evaluation results (WER and CER in %) for the generated dataset with ScrabbleGAN [40] OffHW-German and the IAM-OffDB [115] dataset.

	Method	OffHW-German WER	OffHW-German CER	IAM-OffDB WER	IAM-OffDB CER
Related Work	ScrabbleGAN [40]	-	-	23.61	-
	OrigamiNet [41] (12 × gated text recognizer)	-	-	-	4.70
Our Implementation	OrigamiNet (ours, 4 × gated text recognizer)	1.50	0.11	90.40	15.67
	Inception	12.54	1.17	-	-
	ResNet	13.05	1.24	-	-
	Gated text recognizer (2 blocks), 1 conv. layer	4.34	0.39	-	-
	Gated text recognizer (2 blocks), 2 conv. layer	5.02	0.44	-	-
	Gated text recognizer (4 blocks), 1 conv. layer	3.35	0.34	89.37	15.60
	Gated text recognizer (4 blocks), 2 conv. layer	2.52	0.24	-	-
	Gated text recognizer (6 blocks)	2.85	0.26	-	-
	Gated text recognizer (8 blocks)	4.22	0.38	-	-

TABLE 3. Evaluation results (WER and CER in %) for the generated OffHW-words500 and OffHW-wordsRandom datasets for one and two convolutional layers (c). We propose writer-dependent (WD) and writer-independent (WI) results.

Method (4 × gated text recognizer)	OffHW-words500 WD WER	OffHW-words500 WD CER	OffHW-words500 WI WER	OffHW-words500 WI CER	OffHW-wordsRandom WD WER	OffHW-wordsRandom WD CER	OffHW-wordsRandom WI WER	OffHW-wordsRandom WI CER
$c = 1$	2.94	0.76	0.95	0.23	1.98	0.35	2.05	0.37
$c = 2$	2.51	0.69	0.85	0.22	1.82	0.34	1.95	0.38

TABLE 4. Feature embeddings $f_c(\mathbf{X}_i)$ and $f_c(\mathbf{U}_i)$ of exemplary image $\mathbf{X}_i$ and multivariate time-series $\mathbf{U}_i$ data of the convolutional layer $c = \text{conv}_2$ for different deep metric learning functions for positive pairs (*Edit_distance* = 0) and negative pairs (*Edit_distance* > 0) trained with the triplet loss. The feature embeddings are similar in the red box (character x) or blue box (character p) for $f_2(\mathbf{X}_i)$, or the last pixels (character t) of $f_2(\mathbf{U}_i)$ for $\mathcal{L}_{\text{PC}}$ marked green.



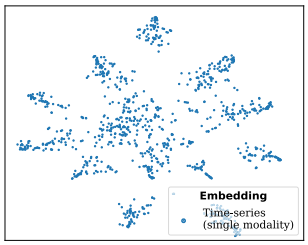
similar for writer-dependent (WD) and writer-independent (WI) tasks, WI results of the OffHW-words500 dataset are lower than WD results, as words with the same label appear in the training and test dataset. We use the weights of the fine-tuning as initial weights of the image model for the cross-modal representation learning.

3) EVALUATION OF REPRESENTATION LEARNING FEATURE EMBEDDINGS

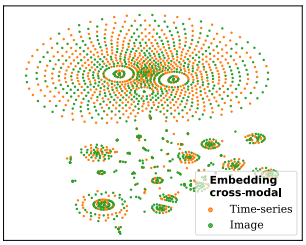
Table 4 shows the feature embeddings for image $f_2(\mathbf{X}_i)$ and time-series data $f_2(\mathbf{U}_i)$ of the *positive* sample `Export` and the two *negative* samples `Expert` (*Edit_distance* = 1) and `Import` (*Edit_distance* = 2) based on four deep metric learning loss functions. The pattern of characters are similar, as the words differ only in the fourth letter. In contrast, `Import` has a different feature embedding, as the replacement of `E` with `I` and `x` with `m` leads to a higher feature distance in the embedding hypersphere. Note that image and time-series data can vary in length for *Edit_distance* > 0. Figure 9 shows the feature embeddings of the output of the convolutional layers ($c = 1$) processed with t-SNE [123]. Figure 9a visualizes the multivariate time-series embeddings $f_1(\mathbf{U}_i)$ of the single modal network. The learned representation generalizes well, but misclassifications (e.g., of small and capital letters at the beginning of a word, which happen quite often) also introduce errors in the latent

TABLE 5. Evaluation results (WER and CER in %) averaged over five splits of the baseline time-series-only technique and our cross-modal learning technique for the inertial-based OnHW datasets [38] with and without mutated vowels (MV) for one convolutional layer $c = 1$. Best results are bold, and second best results are underlined. Arrows indicate improvements (↑) and degradation (↓) of baseline results (w/o MV).

		OnHW-words500				OnHW-wordsRandom			
		WD		WI		WD		WI	
	Method	WER	CER	WER	CER	WER	CER	WER	CER
Main Task	InceptionTime, $\mathcal{L}_{\text{CTC}}$, w/ MV	37.12	12.96	62.09	26.36	42.88	7.19	84.14	32.35
	IT+BiLSTM, $\mathcal{L}_{\text{CTC}}$, w/ MV	43.22	13.07	61.62	26.08	39.14	6.39	85.42	33.31
	CNN+BiLSTM, $\mathcal{L}_{\text{CTC}}$, w/ MV	42.81	13.04	60.47	28.30	37.13	6.75	83.28	35.90
	CNN+BiLSTM, $\mathcal{L}_{\text{CTC}}$, w/o MV	42.77	13.44	59.82	28.54	38.02	7.81	83.54	36.51
Baseline	$\mathcal{L}_{\text{MSE}}$	40.76 ↑	12.71 ↑	**55.54** ↑	25.97 ↑	37.31 ↑	7.01 ↑	82.25 ↑	33.85 ↑
	$\mathcal{L}_{\text{CS}}$	38.62 ↑	11.55 ↑	56.37 ↑	**25.90** ↑	38.85 ↓	7.35 ↑	82.48 ↑	35.67 ↑
	$\mathcal{L}_{\text{PC}}$	39.09 ↑	11.69 ↑	57.90 ↑	27.23 ↑	38.46 ↓	7.15 ↑	82.71 ↑	35.13 ↑
	$\mathcal{L}_{\text{KL}}$	38.36 ↑	11.28 ↑	60.23 ↓	27.99 ↑	38.76 ↓	7.49 ↑	**81.07** ↑	33.96 ↑
Contrastive Loss	$\mathcal{L}_{\text{contr},1}(\mathcal{L}_{\text{MSE}})$	38.34 ↑	11.57 ↑	56.81 ↑	25.98 ↑	38.25 ↓	7.31 ↑	82.09 ↑	34.03 ↑
	$\mathcal{L}_{\text{contr},1}(\mathcal{L}_{\text{CS}})$	39.68 ↑	11.73 ↑	58.03 ↑	27.13 ↑	**35.96** ↑	**6.67** ↑	81.22 ↑	33.11 ↑
	$\mathcal{L}_{\text{contr},1}(\mathcal{L}_{\text{PC}})$	37.82 ↑	11.34 ↑	57.45 ↑	26.18 ↑	39.22 ↓	7.39 ↑	82.45 ↑	34.21 ↑
	$\mathcal{L}_{\text{contr},1}(\mathcal{L}_{\text{KL}})$	**36.70** ↑	**10.84** ↑	61.72 ↓	29.16 ↓	38.92 ↓	7.51 ↑	83.54	35.52 ↑
Triplet Loss	$\mathcal{L}_{\text{trpl},1}(\mathcal{L}_{\text{MSE}})$	42.95 ↓	14.13 ↓	56.48 ↑	26.66 ↑	37.66 ↑	7.04 ↑	81.64 ↑	34.39 ↑
	$\mathcal{L}_{\text{trpl},1}(\mathcal{L}_{\text{CS}})$	38.01 ↑	11.29 ↑	58.50 ↑	27.10 ↑	37.12 ↑	6.98 ↑	82.71 ↑	**33.09** ↑
	$\mathcal{L}_{\text{trpl},1}(\mathcal{L}_{\text{PC}})$	40.43 ↑	12.41 ↑	58.20 ↑	27.48 ↑	37.40 ↑	7.01 ↑	81.90 ↑	33.89 ↑
	$\mathcal{L}_{\text{trpl},1}(\mathcal{L}_{\text{KL}})$	37.55 ↑	11.21 ↑	63.52 ↓	30.52 ↓	38.39 ↓	7.36 ↑	83.18 ↑	35.21 ↑



(a) Feature embedding of IMU samples for the single modality network.



(b) Feature embeddings of IMU and image samples for the cross-modal network.

FIGURE 9. Comparison of the naive method (left) and our proposed approach (right), where our method shows a much better behaved embedding space compared to the naive approach by learning a joint representation. Plot of 400 × 200 feature embeddings of image and IMU modalities with t-SNE.

representation. Figure 9b visualizes the multivariate time-series and image embeddings $\big(f_1(\mathbf{U}_i)$ and $f_1(\mathbf{X}_i)$, respectively$\big)$ in a cross-modal setup. While the embedding of the single modal network is unstructured, the embeddings of the cross-modal network are structured (distance of samples visualizes the Edit distance between words) with the embeddings of the time-series modality being close to the embeddings of the image modality, and hence, more distinctive clusters with better separation.

4) EVALUATION OF CROSS-MODAL REPRESENTATION LEARNING

Table 5 gives an overview of cross-modal representation learning (for $c = 1$). The first row shows baseline results by [38]: 13.04% CER on OnHW-words500 (WD) and 6.75% CER on OnHW-wordsRandom (WD) with mutated vowels. Compared to various time-series classification techniques, their benchmark results showed superior performance of CNN+BiLSTMs on these OnHW recognition tasks. Only InceptionTime [124] (a large time-series encoder network with *depth* = 11 and *nf* = 96) – with BiLSTM layers – yields partly better results or is on par with the CNN+BiLSTM model for sequence-based classification, while the CNN+BiLSTM model outperforms state-of-the-art techniques on single character-based classification tasks. Due to the faster training of the CNN+BiLSTM, we chose this network for the cross-modal task. In general, the word error rate (WER) can vary for a similar character error rate (CER). The reason is that a change of one character of a correctly classified word leads to a large change in the WER, while the change of the CER is marginal. We define the results trained without mutated vowels as baseline results, as ScrabbleGAN is pretrained on IAM-OffDB, which does not contain mutated vowels, and hence, such words cannot be generated. Nevertheless, the main model can be trained and is applicable to samples with mutated vowels.

For a fair comparison, we compare our results to the results of the models trained without mutated vowels. Here, the error rates are slightly higher for both datasets. As expected, cross-modal learning improves the baseline results up to 11.28% CER on the OnHW-words500 WD dataset and up to 7.01% CER on the OnHW-wordsRandom WD dataset. The contrastive loss shows the best results on the OnHW-words500 (WD) dataset with the Kullback-Leibler metric and on the OnHW-wordsRandom dataset (WD) with the cosine similarity metric. With the triplet loss, $\mathcal{L}_{\text{CS}}$ outperforms other metrics on the OnHW-wordsRandom dataset but is inconsistent on the OnHW-words500 dataset. The importance of the triplet loss is more significant for one convolutional layer ($c = 1$) than for two convolutional layers ($c = 2$) (see Appendix G). Furthermore, training with kMMD (implemented as in [14]) does not yield reasonable results. We assume that this metric cannot make use of the important time component in the HWR application. We proposed

TABLE 6. Evaluation results (WER and CER in %) averaged over five splits of the baseline time-series-only technique and our cross-modal techniques for the inertial-based left-handed writers OnHW datasets [38] with and without mutated vowels (MV) for one ($c = 1$) and two ($c = 2$) convolutional layers $c = 1$. Best results are bold, and second best results are underlined. Arrows indicate improvements (↑) and degradation (↓) of baseline results (w/o MV).

		OnHW-words500-L				OnHW-wordsRandom-L			
		WD		WI		WD		WI	
	Method	WER	CER	WER	CER	WER	CER	WER	CER
Main Task	InceptionTime, $\mathcal{L}_{\text{CTC}}$, w/ MV	49.70	14.02	100.0	96.06	48.10	8.63	100.0	95.93
	CNN+BiLSTM, $\mathcal{L}_{\text{CTC}}$, w/ MV	14.20	3.30	94.40	71.41	30.20	4.86	100.0	83.51
	CNN+BiLSTM, $\mathcal{L}_{\text{CTC}}$, w/o MV	12.94	3.33	<u>89.07</u>	62.07	30.89	5.26	100.0	81.15
Baseline	$\mathcal{L}_{\text{MSE}}$	<u>11.62</u> ↑	2.77 ↑	90.65 ↓	67.90 ↓	30.53 ↑	4.93 ↑	100.0	81.99 ↓
	$\mathcal{L}_{\text{CS}}$	14.92 ↓	3.53 ↓	94.14 ↓	65.10 ↓	29.06 ↑	4.87 ↑	100.0	83.94 ↓
	$\mathcal{L}_{\text{PC}}$	12.29 ↑	3.04 ↑	91.33 ↓	60.89 ↑	<u>27.32</u> ↑	**4.47** ↑	100.0	85.09 ↓
	$\mathcal{L}_{\text{KL}}$	**11.37** ↑	**2.57** ↑	93.02 ↓	66.64 ↓	29.61 ↑	4.91 ↑	100.0	81.28 ↓
Triplet Loss	$\mathcal{L}_{\text{trpl},1}(\mathcal{L}_{\text{MSE}})$	12.48 ↑	3.11 ↑	90.09 ↓	62.87 ↓	32.62 ↓	5.43 ↓	100.0	**80.41** ↑
	$\mathcal{L}_{\text{trpl},1}(\mathcal{L}_{\text{CS}})$	13.65 ↓	3.28 ↑	90.76 ↓	62.40 ↓	34.21 ↓	5.53 ↓	100.0	82.14 ↓
	$\mathcal{L}_{\text{trpl},1}(\mathcal{L}_{\text{PC}})$	13.71 ↓	3.23 ↑	91.55 ↓	65.95 ↓	31.59 ↓	5.32 ↓	100.0	81.77 ↓
	$\mathcal{L}_{\text{trpl},1}(\mathcal{L}_{\text{KL}})$	13.65 ↓	3.45 ↓	94.93 ↓	72.01 ↓	31.87 ↓	5.42 ↓	100.0	82.02 ↓
	$\mathcal{L}_{\text{trpl},2}(\mathcal{L}_{\text{MSE}})$	11.97 ↑	2.83 ↑	**84.34** ↑	**57.84** ↑	**27.19** ↑	4.79 ↑	**99.87** ↑	82.60 ↓
	$\mathcal{L}_{\text{trpl},2}(\mathcal{L}_{\text{CS}})$	11.65 ↑	<u>2.63</u> ↑	94.70 ↓	67.69 ↓	28.39 ↑	<u>4.62</u> ↑	100.0	83.44 ↓
	$\mathcal{L}_{\text{trpl},2}(\mathcal{L}_{\text{PC}})$	13.02 ↓	2.94 ↑	89.86 ↓	<u>60.26</u> ↑	30.22 ↑	4.81 ↑	100.0	84.29 ↓
	$\mathcal{L}_{\text{trpl},2}(\mathcal{L}_{\text{KL}})$	13.55 ↓	3.22 ↑	97.86 ↓	76.54 ↓	28.14 ↑	4.71 ↑	100.0	<u>80.81</u> ↑

our approach as learning with privileged information by exploiting a visual modality as an auxiliary task and improve the main task based on an inertial modality. The cross-modal learning would also work for the visual modality as the main task and a generated dataset for the inertial modality as an auxiliary task. However, the error rates are already low for the image-based classification task, as methods for offline HWR are very advanced and the image dataset is very large. Hence, we assume that fine-tuning the image encoder with inertial data would result in a minor improvement. Prior work [38] evaluated data augmentation techniques for multivariate time-series data (i.e., time warping, scaling, jittering, magnitude warping, and shifting). This approach was rather limited with only 2-3% points of improvement compared with augmentation with the auxiliary image-based task.

5) TRANSFER LEARNING ON LEFT-HANDED WRITERS

To adapt the model to left-handed writers (who are typically under-represented and hence marginalized in the real-world), we make use of the left-handed datasets OnHW-words500-L and OnHW-wordsRandom-L proposed by [38]. These datasets contain recordings of two writers who provided 1,000 and 996 samples. As a baseline, we pre-train the time-series-only model on the right-handed datasets and post-train the left-handed datasets for 500 epochs (see the second and third rows of Table 6). As these datsets are rather small, the models can overfit on these specific writers and achieve a very low CER of 3.33% on the OnHW-words500-L datasets and 5.26% CER on the OnHW-wordsRandom-L dataset without mutated vowels for the writer-dependent tasks. However, the models cannot generalize on the writer-independent tasks, as evidenced by 62.07% CER on the OnHW-words500-L dataset and 81.15% CER on the OnHW-wordsRandom-L dataset. Hence, we focus on the WD tasks. For comparison, we use the state-of-the-art time-series classification technique InceptionTime [124] with *depth* = 11 and *nf* = 96 (without pre-training). As shown, our CNN+BiLSTM outperforms InceptionTime by a considerable margin. We use the weights of the pre-training with the offline handwriting datasets and again post-train on the left-handed datasets with $c = 1$ and $c = 2$. Using the weights of the cross-modal learning without the triplet loss can decrease the error rates up to 2.57% CER with $\mathcal{L}_{\text{KL}}$ and 4.47% CER with $\mathcal{L}_{\text{PC}}$. Using the triplet loss $\mathcal{L}_{\text{trpl},2}(\mathcal{L}_{\text{MSE}})$ can further significantly decrease the WI OnHW-words500-L error rates. In conclusion, due to the use of the weights of the cross-modal setup, the model can adapt faster to new writers and generalize better to unseen words due to the triplet loss.

VI. CONCLUSION AND FUTURE RESEARCH

We evaluated metric learning-based triplet loss functions for cross-modal representation learning between image and time-series modalities with class label-specific triplet selection.

We perform experiments on synthetic data for learning a common representation between images and time-series data for single class prediction. The label-specific triplet selection in combination with a deep metric learning loss leads to an accuracy improvement from 92.5% to 96.76% by being more robust against noise present in the data.

Furthermore, we propose an extensive evaluation on handwriting datasets. We learn a common representation between offline handwriting data (image-based) and online handwriting data from sensor-enhanced pens (time-series-based). We generated two million images by employing ScrabbleGAN to imitate arbitrarily many writing styles. Our cross-modal triplet loss with dynamic triplet selection based on the Edit distance further yields a faster training convergence with better generalization on the main task. The representation of the feature embeddings between both modalities is more structured and the model is more robust

against different writing styles. This yields a notable accuracy improvement for the main time-series classification task (e.g., from 13.44% 10.84% CER for the OnHW-words500 dataset) that can be decoupled from the auxiliary image classification task at inference time. Our proposed method leads to a better adaptability to different writers, such as a better transfer learning from right-handed writers to the under-represented left-handed writers.

For future work, the influence of the generative model to augment offline handwriting data can be elaborated. Recent models include the approaches presented in [125], [126], [127], and [128]. The generator proposed by Kang et al. [126] conditions on both visual appearance and textual content, and it can produce text-line samples with diverse handwriting styles that visually outperform ScrabbleGAN. On the other hand, HiGAN+ [128] introduces a contextual loss to enhance style consistency and achieves better calligraphic style transfer. Furthermore, domain adaptation techniques such as higher-order moment matching (HoMM) by Chen et al. [129] can further improve the adaptability to left-handed writers.

APPENDICES

We provide more information about the broader impact, limitations, ethical concerns, and a comparison to writing on touch sceen surfaces in Section A. While Section B gives an overview of methods for offline handwriting recognition, Section C summarizes cross-modal retrieval methods, the corresponding modalities, pairwise learning, and deep metric learning. We present the multi-task learning technique in Section D, and show more details on learning with the triplet loss on synthetically generated signal and image data in Section E. We propose more details of our architectures in Section F. Section G presents results of representation learning for online HWR.

A. STATEMENTS

1) BROADER IMPACT STATEMENT

While research for offline handwriting recognition (HWR) is well-established, research for online HWR from sensor-enhanced pens only emerged in 2019. Hence, the methodological research for online HWR currently does not meet the requirements for real-world applications. Handwriting is still important in different fields, in particular graphomotoricity as a fine motor skill. The visual feedback provided by the pen helps young students to learn a new language. A well-known bottleneck for many machine learning algorithms is their requirement for large amounts of datasets, while data recording of handwriting data is time-consuming. This paper extends the online HWR dataset with generated images from offline handwriting and closes the gap between offline and online HWR by using offline HWR as an auxiliary task by learning with privileged information. One downside of training the offline architecture (consisting of gated text recognizer blocks) is its long training time. However, as this model is not required at inference time, processing the time-series is still fast. The cross-modal representation between both modalities (image and time-series) is achieved by using the triplet loss and a sample selection depending on the Edit distance. This approach is important in many applications of sequence-based classification, i.e., the triplet loss evolved recently for language processing applications such as visual semantic clustering, while pairwise learning is typically applied in fields such as image recognition.

2) LIMITATIONS

The limitation of the method is the requirement of multiple image-based datasets in the same language. As the OnHW-words and OnHW-wordsRandom datasets are written in German and contain word labels with mutated vowels, a similar image-based German dataset is required, which does not currently exist. The available dataset most similar to the OnHW dataset is the IAM-OffDB dataset, which does not contain mutated vowels. Hence, the OCR method cannot be pre-trained on words with mutated vowels. In conclusion, the method is not limited by ScrabbleGAN, but by the image-based dataset required for pre-training. The gated text recognizer could also be directly pre-trained on the IAM-OffDB dataset, but we assume less generalized results than for our generated dataset.

3) STATEMENT ON ETHICAL CONCERNS

Machine learning models face various challenges when classifying text with this sensor-enhanced pen. These challenges can appear if there is a domain shift between training and test datasets, e.g., specific writers have a unique writing style and accelerations, or they hold the pen differently. Also, some writers might have a unique writing environment (different writing surfaces such as a unique table or paper which leads to different magnetic fields). Another difficulty can appear through an under-represented group such as left-handed writers or a disabled person for which the model is not trained on. A well-generalized model trained on all possible pen movements is very challenging and requires a lot of training data. One solution is to record data for a unique writer and adapt the model, or augment the data for a better representation, e.g., as proposed with our method on left-handed writers. Hence, unique writers are not excluded and the task for classifying writing from under-represented groups is addressed in our paper, but domain shifts still remain a challenging problem. Ethical statement about collection consent and personal information: For data recording, the consent of all participants was collected. The datasets only contain the raw data from the sensor-enhanced pen and – for statistics – the age, gender, and handedness of the participants. The datasets are fully pseudonymized by assigning an ID to every participant. The datasets do not contain any personal identifying information. The approach proposed in this paper – in particular, when used for the application of online handwriting recognition

from sensor-enhanced pens – does not (1) facilitate injury to living beings, (2) raise safety or security concerns (due to the anonymity of the data), (3) raise human rights concerns, (4) have a detrimental effect on people's livelihood, (5) develop harmful forms of surveillance (as the data is pseudonymized), (6) damage the environment, and (7) deceive people in ways that cause harm.

4) COMPARISON TO WRITING ON TOUCH SCREEN SURFACES

Methods for writing on surfaces such as the iPad OS system and others require a tablet with a touch screen surface and stylus pens with integrated magnetometers or pressure sensitivity. These methods can easily reconstruct the trajectory of the pen tip through the magnetometer on the surface, and hence, can classify the written text. This is more challenging when using sensor-enhanced pens, as the classification task is performed directly on the sensor data. One drawback of methods used in the iPad OS is the requirement for writing on specific surfaces, which in turn can influence the writing style. Also, certain applications require writing on normal paper, or the availability of a touch screen surface is not always given, e.g., when writing a short list, but notes need to be digitized afterwards.

B. OFFLINE HANDWRITING RECOGNITION

In the following, we give a detailed overview of offline HWR methods to select a suitable lexicon and language model-free method. To our knowledge, there is no recent paper summarizing published work for offline HWR. For an overview of offline and online HWR datasets, see [29] and [42]. Table 7 presents related work. Methods for offline HWR range from hidden Markov models (HMMs) to deep learning techniques that became predominant in 2014, such as convolutional neural networks (CNNs), temporal convolutional networks (TCNs), and recurrent neural networks (RNNs). RNN techniques are well explored, including long short-term memories (LSTMs), bidirectional LSTMs (BiLSTMs), and multidimensional RNNs (MDRNN, MDLSTM). Recent methods are generative adversarial networks (GANs) and Transformers. In Table 7, we refer to the use of a language model as LM with k and identify the data level on which the method works – i.e., paragraph or full-text level (P), line level (L), and word level (W). We present evaluation results for the IAM-OffDB [115] and RIMES [130] datasets. We show the character error rate (CER) – the percentage of characters that were incorrectly predicted (the lower, the better) – and the word error rate (WER) – a common performance metric on word level instead of the phoneme level (the lower, the better).

1) LSTMs AND BiLSTMs

RNNs for HWR marked an important milestone in achieving impressive recognition accuracies. Sequential architectures are perfect to fit text lines, due to the probability distributions over sequences of characters, and due to the inherent temporal aspect of text [63]. [65] introduced the BiLSTM layer in combination with the CTC loss. Reference [64] showed that the performance of LSTMs can be greatly improved using dropout. Reference [134] investigated sequence-discriminative training of LSTMs using the maximum mutual information (MMI) criterion. While [135] utilized an RNN with an HMM and a language model, [133] combined an RNN with a sliding window Gaussian HMM. GCRNN [67] combines a convolutional encoder (aiming for generic and multilingual features) and a BiLSTM decoder predicting character sequences. Additionally, [66] proposed a CNN+BiLSTM architecture (CNN-1DLSTM-CTC) that uses the CTC loss. The start, follow, read (SFR) [136] model jointly learns text detection and segmentation. Reference [137] used synthetic data for pre-training and image normalization for slant correction. The methods by [52], [53], [54], and [55] also make use of BiLSTMs. While [139] uses a feature pyramid network (FPN), the adversarial feature deformation module (AFDM) [140] learns ways to elastically warp extracted features in a scalable manner. Further methods that combine CNNs with RNNs are [68], [69], and [70], while BiLSTMs are utilized in [71] and [72].

2) TCNs

TCNs use dilated causal convolutions and have been applied to air-writing recognition by [142]. As RNNs are slow to train, [50] presented a faster system that is based on text line images and TCNs with the CTC loss. This method achieves 9.6% CER on the IAM-OffDB dataset. Reference [51] combined 2D convolutions with 1D dilated non-causal convolutions that offer high parallelism with a smaller number of parameters. They analyzed re-scaling factors and data augmentation and achieved comparable results for the IAM-OffDB and RIMES datasets.

3) CNNs

Reference [48] utilized a CNN with multiple fully connected branches to estimate its n-gram frequency profile (set of n-grams contained in the word). With canonical correlation analysis (CCA), the estimated profile can be matched to the true profiles of all words in a large dictionary. As most attention methods suffer from an alignment problem, [49] proposed a decoupled attention network (DAN) that has a convolutional alignment module that decouples the alignment operation from using historical decoding results based on visual features. The gated text recognizer [73] aims to automate the feature extraction from raw input signals with a minimum required domain knowledge. The fully convolutional network without recurrent connections is trained with the CTC loss. Thus, the gated text recognizer module can handle arbitrary input sizes and can recognize strings with arbitrary lengths. This module has been used for OrigamiNet [41] which is a segmentation-free multi-line or full-page recognition system. OrigamiNet yields state-of-the-art results on the IAM-OffDB dataset, and shows

TABLE 7. Evaluation results (WER and CER in %) of different methods on the IAM-OffDB [115] and RIMES [130] datasets. The table is sorted by year.

	Method	Information	LM size k	Level P	Level L	Level W	IAM-OffDB WER	IAM-OffDB CER	RIMES WER	RIMES CER
HMM	HMM+ANN [47]	Markov chain with MLP	w/ (5)				15.50	6.90	-	-
	Tandem GHMM [131]	GHMM and LSTM, writer adaptation	w/ (50)			×	13.30	5.10	13.70	4.60
	LSTM-HMM [132]	Combination of LSTM with HMM	w/ (50)		×		12.20	4.70	12.90	4.30
Multi-dim. LSTM	2DLSTM [56]	Combined MDLSTM with CTC	w/o				27.50	8.30	17.70	4.00
	MDLSTM-RNN [57]	150 dpi	w/o	×			29.50	10.10	13.60	3.20
		150 dpi	w/ (50)	×			16.60	6.50	-	-
		300 dpi	w/o	×			24.60	7.90	12.60	2.90
		300 dpi	w/ (50)	×			16.40	5.50	-	-
	[58]	GPU-based, diagonal MDLSTM					9.30	3.50	9.60	2.80
	SepMDLSTM [59]	Multi-task approach	w/o				34.55	11.15	30.54	8.29
	[60]	MDLSTM, attention	w/o	×			-	16,20	-	-
		Line segmentation 150 dpi	w/o		×		-	11.10	-	-
		Line segmentation 150 dpi	w/o		×		-	7.50	-	-
	MDLSTM [61]						10.50	3.60	-	-
RNN	BiLSTM [65]		w/ (20)				18.20	25.90	-	-
	HMM+RNN [133]	Sliding win. Gaussian HMM, RNN			×	×		-	4.75	-
	Dropout [64]	LSTMs with dropout	w/o				35.10	10.80	28.50	6.80
	[134]	Maximum mutual information					12.70	4.80	12.10	4.40
	[135]						10.90	4.40	11.20	3.50
			w/ (50)				13.60	5.10	12.30	3.30
	GCRNN [67]	CNN+BiLSTM	w/ (50)				10.50	3.20	7.90	1.90
	CNN-1DLSTM-CTC [66]	CNN+BiLSTM+CTC (128 × width)	w/o		×		18.40	5.80	9.60	2.30
		NN+BiLSTM+CTC	w/ (50)		×		12.20	4.40	9.00	2.50
	End2End [62]	Without line level	w/				16.19	6.34	-	-
		Line level	w/		×		32.89	9.78	-	-
	SFR [136]	Text detection and segmentation	w/o	×			23.20	6.40	9.30	2.10
	CNN-RNN [137]	Unconstrained	w/o				12.61	4.88	7.04	2.32
		Full-Lexicon	w/				4.80	2.52	**1.86**	**0.65**
		Text-Lexicon	w/				**4.07**	**2.17**		
		Unconstrained	w/o		×		17.82	5.70	9.60	2.30
	[52]	Seq2seq, w/o LN	w/o				25.50	17.40	19.10	12.00
		w/ LN	w/o				22.90	13.10	15.80	9.70
		w/ LN + Focal Loss	w/o				21.10	11.40	13.50	7.30
		w/ LN + Focal Loss + Beam Search	w/o				16.70	8.10	9.60	3.50
	[53]	LSTM encoder-decoder, attention					15.90	4.80	-	-
	[138]	ResNet+LSTM, segmentation	w/	×			-	8.50	-	-
	[54]	BiLSTM			×		30.70	12.80	-	-
		GRCL			×		35.20	14.10	-	-
	[55]	Seq2seq CNN+BiLSTM (64 × width)			×		-	5.24	-	-
	FPN [139]	Feature Pyramid Network, 150 dpi			×		-	15.60	-	-
	AFDM [140]	AFD module	w/				8.87	5.94	6.31	3.17
CNN	[48]	CNN + connected branches, CCA	w/				6.45	3.44	3.90	1.90
	Gated text recognizer [73]	CNN+CTC (32 × width)	w/o		×		-	4.90	-	-
	OrigamiNet [41]	VGG (500 × 500)	×	×			-	51.37	-	-
		VGG (500 × 500), w/o LN	w/o	×	×		-	34.55	-	-
		ResNet26 (500 × 500), w/o LN	w/o	×	×		-	10.03	-	-
		ResNet26 (500 × 500), w/ LN	w/o	×	×		-	7.24	-	-
		ResNet26 (500 × 500), w/o LN	w/o	×	×		-	8.93	-	-
		ResNet26 (500 × 500), w/ LN	w/o	×	×		-	6.37	-	-
		ResNet26 (500 × 500), w/o LN	w/o	×	×		-	76.90	-	-
		ResNet26 (500 × 500), w/ LN	w/o	×	×		-	6.13	-	-
		GTR-8 (500 × 500), w/o LN	w/o	×	×		-	72.40	-	-
		GTR-8 (500 × 500), w/ LN	w/o	×	×		-	5.64	-	-
		GTR-8 (750 × 750), w/ LN	w/o	×	×		-	5.50	-	-
		GTR-12 (750 × 750), w/ LN	w/o	×	×		-	4.70	-	-
	DAN [49]	Decoupled attention module	w/o		×		19.60	6.40	8.90	2.70
GAN	ScrabbleGAN [40]	Original data	w/o				25.10	-	12.29	-
		Augmentation	w/o				24.73	-	12.24	-
		Augmentation + 100k synth.	w/o				23.98	-	11.68	-
		Augmentation + 100k synth. + Refine	w/o				23.61	-	11.32	-
Trans-former	[63]	Self-attention for text/images	w/o		×		15.45	4.67	-	-
	FPHR [89]	CNN encoder, Transformer decoder	w/o	×			-	6.70	-	-
		With augmentation	w/o	×			-	6.30	-	-
Other	FST [141]	Finite state transducer (lexicon)	n-gram				19.10	-	13.30	-

Abbreviations. Size k of the language model (LM) (with (w/) or without (w/o) a LM). P: paragraph or full text level, L: line level, LN: layer normalization, CER/WER: character/word error rate, HMM: hidden markov model, GTR: gated text recognizer, seq2seq: sequence-to-sequence, GAN: generative adversarial network, CTC: connectionist temporal classification, RNN: recurrent neural network, LSTM: long short-term memory

TABLE 8. Overview of cross-modal and pairwise learning techniques using the modalities video (V), images (I), audio (A), text (T), sensors (S), or haptic (H). Data from sensors are represented by time-series from inertial, biological, or environmental sensors. We indicate cross-modal learning from the same modality with $\times^n$ with n modalities. If n is unspecified, the method can potentially work with an arbitrary number of modalities.

Method (sorted by year)	Modality V	I	A	T	S	H	Pairwise Learning	Deep Metric Learning Loss/Objective	Application
[96]		$\times^2$					Pairwise	L_1 similarity	Face verification
[91]		×		×			Pairwise	Canonical correlation analysis	Multimedia documents: emb. mapping to common space
DeViSE [143]		×		×			Hinge rank	Cosine similarity	Visual semantic embedding
OxfordNet [144]		×		×			Contrastive	Cosine similarity	Visual semantic embedding
[145]		×		×			Denotion graph	Pointwise MI	Visual semantic embedding
DAN [14]		$\times^2$					Pairwise	Kernelized MMD	Domain adaptation
ml-CCA [4]		×		×			Not pairwise	CCA extended	Multi-label annotations
FaceNet [15]		×					Triplet	Euclidean	Face recognition, clustering
deep-SM [12]		$\times^2$		×			Pairwise	CCA, T-V CCA semantic matching	Universal representation for various recognition tasks
[103]		×		×			Triplet	non-Mercer match kernel	Visual semantic embedding
TristouNet [106]			×				Triplet	Euclidean	Speech classification
Triplet+FANNG [98]		×					Smart triplet	Nearest neighbour graph	General
[94]		×					Triplet	CE, conditional random field	Handwritten Chinese characters recognition
[146]		×		×			Pairwise	Cosine similarity	Visual semantic embedding
GXN [147]		×		×			Triplet	Similarity: order--violation penalty	Visual semantic embedding
TDH [104]		$\times^2$		×			Triplet	Hamming space	Visual semantic embedding
VSE++ [6]		×		×			Triplet	Similarity: inner product	Visual semantic embedding
SCAN t-i [148]		×		×			Triplet	Similarity LSE	Visual semantic embedding
Discriminative [13]		×					Triplet	Class centroids	Image classification
VSRN [149]		×		×			Triplet	Similarity: inner product	Visual semantic embedding
PIE-Nets [150]	×	×		×			Pairwise	Diversity, MIL, MMD	Visual semantic embedding
LIWE [151]		×		×			Contrastive	Sum/Max of Hinges	Visual semantic embedding
[105]		×					STriplet+triplet	Cosine similarity	Relationship understanding
TIMAM [3]		×		×			Pairwise	Norm-softmax CE	Visual question answering
GMN [8]		$\times^n$	$\times^n$			$\times^n$	Pairwise	Cross-modal generation	Multisensory 3D scenes
CTM [100]	×			×			Triplet	CTC, CE , L_2 correlation	Sentence translation
UniVSE [152]		×		×			Contrastive	Alignment losses	Visual semantic embedding
ActiveSet+RRPB [97]		×					Smart triplet	Semidefinite constraint	General
PAN [153]		×		×			Pairwise	Cosine similarity	Visual semantic embedding
CM-GANs [154]		×		×			Adversarial	Inter/intra class	Visual semantic embedding
CPC [155]		×	×		×		Contrastive	CE, MI	One modality classification
CrossATNet [102]		$\times^2$		×			Triplet	MSE	Zero-shot learning, sketches
MHTN [5]	×	×	×	×			Pairwise, contr.	MMD, Euclidean	CMR
GCML [2]	$\times^2$		×				Triplet	Hierarchical relational graph clustering	Retrieval, search, video-to-video similarity
CSVE [156]		×		×			Bidirect. triplet	Correlation graph	Visual semantic embedding
TXS-Adapt [99]		×		×			Triplet (adaptive)	Recency-based correlation	Social media domain
Proxy-Anchor [22]		×					Pair+proxy	Cosine similarity	Image classification
TNN-C-CCA [101]	×		×				Triplet	CCA	Multimedia

improved performance of gated text recognizer over VGG and ResNet26. Hence, we use the gated text recognizer module as our visual feature encoder for offline HWR (see Section F).

4) GANs

Handwriting text generation is a relatively new field. The first approach by [74] was a method to synthesize online data based on RNNs. The technique HWGAN by [75] extends this method by adding a discriminator $\mathcal{D}$. DeepWriting [76] is a GAN that is capable of disentangling style from content and thus making digital ink editable. Reference [77] proposed a method to generate handwriting based on a specific author with learned parameters for spacing, pressure, and line thickness. Reference [78] used a BiLSTM to obtain an embedding of the word to be rendered and added an auxiliary network as a recognizer $\mathcal{R}$. The model is trained with a combination of an adversarial loss and the CTC loss. ScrabbleGAN by [40] is a semi-supervised approach that can arbitrarily generate many images of words with arbitrary length from a generator $\mathcal{G}$ to augment handwriting data and uses a discriminator $\mathcal{D}$ and recognizer $\mathcal{R}$. The paper proposes results for original data with random affine augmentation using synthetic images and refinement.

5) TRANSFORMERS

RNNs prevent parallelization, due to their sequential pipelines. Reference [63] introduced a non-recurrent model by the use of Transformer models with multi-head self-attention layers at the textual and visual stages. Their method works for any pre-defined vocabulary. For the feature encoder, they used modified ResNet50 models. The full page

TABLE 9. Table 8 continued.

Method (sorted by year)	Modality V	I	A	T	S	H	Pairwise Learning	Deep Metric Learning Loss/Objective	Application
AdapOffQuin [157]		×		×			Quintuplet	Cosine similarity	Visual semantic embedding
ROMA [21]		$\times^2$					Soft triplet (fixed margin)	CE, random perturbation	Unsupervised representation learning
CLIP [27]		×		×			Contrastive	CE, Cosine similarity	OCR, action/object recognition
SGRAF [26]		×		×			Pairwise	Vector similarity	Visual semantic embedding
PCME [158]		×		×			Triplet	Euclidean	Visual semantic embedding
MCN [159]	×		×	×			Contrastive	Similarity, reconstruction	Multimodal clustering
VATT [160]	×		×	×			Contrastive	CC, NCE, MIL-NCE	Transformer for CMR
[20]		×					Dual triplet	Euclidean	Signature verification
[161]	×		$\times^2$				Contrastive	Cosine similarity	Audio classification
[9]		$\times^2$					Triplet	Softmax, MSE	Person re-identification
AlignMixup [19]		$\times^2$					Pairwise	Sinkhorn transport	Data augmentation for interpolation
SAM [25]		×		×			Triplet	Cosine similarity	Visual semantic embedding
VSE_∞ [7]		×		×			Triplet	Similarity	Visual semantic embedding
AudioCLIP [162]		×	×	×			Contrastive	Cosine similarity, symmetric CE	Environmental sound classification
data2vec [163]	×		×	×			Predicts latent representations		Self-supervision with masks
ColloSSL [18]					$\times^n$		Contrastive	CE, Cosine similarity	Human-activity recognition
COCOA [17]					$\times^n$		Contrastive	Cosine similarity	General time-series
ELo [164]	×	×	×	×			Contrastive	L_2, evolutionary	Cross-modal, multi-task
[165]		×					Pairwise	-	Crowd counting
MM-ALT [93]	×		×		×		Pairwise	CTC, residual attention	Lyric transcription
FLAVA [166]		×		×			Contrastive	Cosine similarity, temperature scaling	Visual semantic embedding
ConceptBeam [108]		×	$\times^n$				Triplet	Cosine similarity	Target speech extraction
[167]	×			×			Contrastive	-	Text-video retrieval, kitchen
C^3CMR [107]		×		×			Triplet	CE, cosine similarity	Visual semantic embedding
[168]		×		×			Contrastive	Cosine similarity	VSE with graph embedding
[16]					$\times^2$		Classwise	CE, HoMM/CC/PC	Online HWR
CMR-IS (Ours, 2022)		×			×		Contr., triplet	CTC, MSE/CC/PC/KL	Online HWR

Abbreviations. CE: cross-entropy, CTC: connectionist temporal classification, MSE: mean squared error, CC: cross-correlation, PC: Pearson correlation, MMD: maximum mean discrepancy, HoMM: higher-order moment matching, CCA: canonical correlation, analysis, MIL: multiple-instance learning, MI: mutual information, NCE: noise contrastive estimation, VSE: visual semantic embedding

HTR (FPHR) method by [89] uses a CNN as an encoder and a Transformer as a decoder with positional encoding.

C. OVERVIEW OF CROSS-MODAL RETRIEVAL METHODS

We provide a summary of methods for cross-modal learning in Table 8 and Table 9. Typical modalities are video, image, audio, text, sensors (such as inertial sensors used for our method), and haptic modalities. We classify each method with the technique used for pairwise learning that utilizes an objective for deep metric learning. The overview contains a wide range of applications, while visual semantic embedding is a common field for cross-modal retrieval.

D. MULTI-TASK LEARNING

We simultaneously train the $\mathcal{L}_{\text{CTC}}$ loss for sequence classification combined with one or two shared losses $\mathcal{L}_{\text{shared},1}$ and $\mathcal{L}_{\text{shared},2}$ for cross-modal representation learning. As both losses are in different ranges, the naive weighting

$$\mathcal{L}_{\text{total}} = \sum_{i=1}^{|T|} \omega_i \mathcal{L}_i, \tag{3}$$

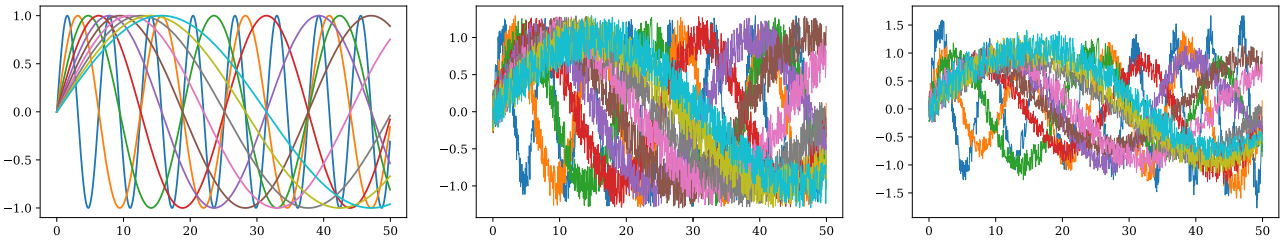
with pre-specified constant weights $\omega_i = 1, \forall i \in \{1, \ldots, |T|\}$ can harm the training process. Hence, we apply dynamic weight average (DWA) [113] as a multi-task learning approach that performs dynamic task weighting over time (i.e., after each batch).

E. TRAINING SYNTHETIC DATA WITH THE TRIPLET LOSS

1) SIGNAL AND IMAGE GENERATION

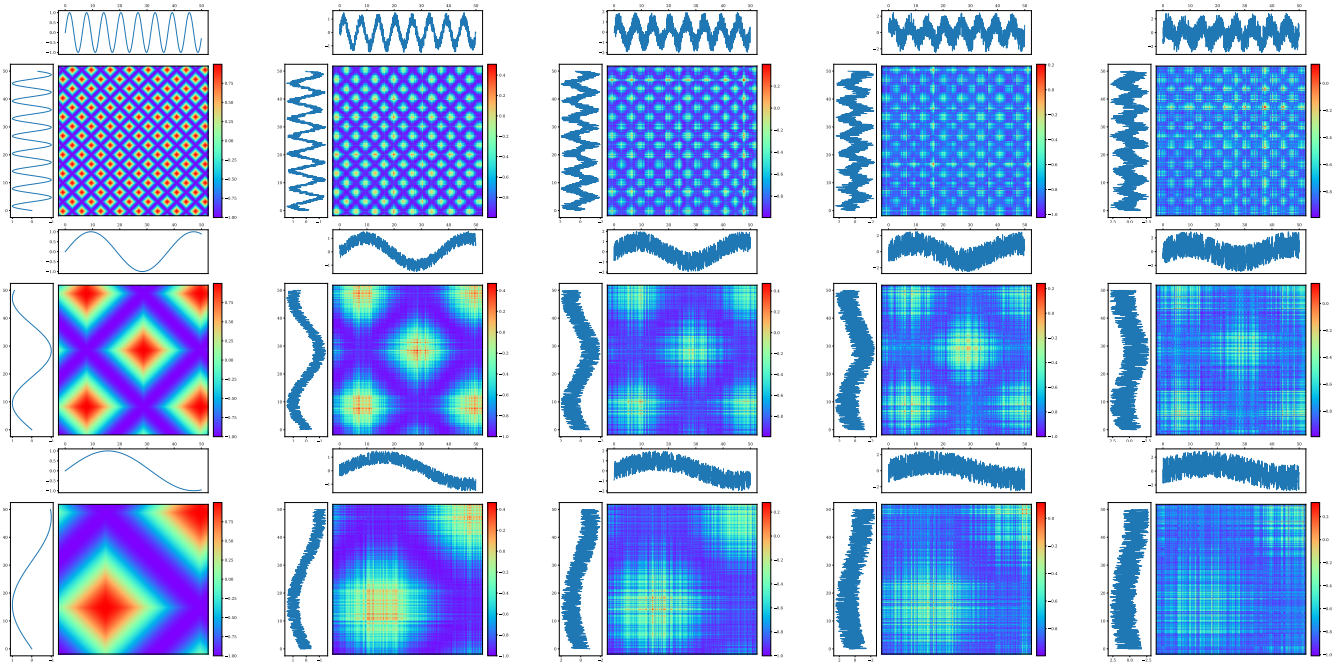
We combine the networks for both signal and image classification to improve the classification accuracy over each single-modal network. The aim is to show that the triplet loss can be used for such a cross-modal setting in the field of cross-modal representation learning. Hence, we generate synthetic data in which the image data contains information of the signal data. We generate signal data $\mathbf{x}$ with $x_{i,k} = \sin\left(0.05 \cdot \frac{t_i}{k}\right)$ for all $t_i \in \{1, \ldots, 1,000\}$ where t_i is the timestep of the signal. The frequency of the signal is dependent on the class label k. We generate signal data for 10 classes (see Figure 10a). We add noise from a continuous uniform distribution $U(a, b)$ for $a = 0$ and $b = 0.3$ (see Figure 10b) and add time and magnitude warping (see Figure 10c). We generate a signal-image pair such that the image is based on the signal data. We make use of the Gramian angular field that transforms time-series into images. The time-series is defined as $\mathbf{x} = (x_1, \ldots, x_n)$ for $n = 1,000$. The Gramian angular field creates a matrix of temporal correlations for each (x_i, x_j) by rescaling the time-series in the range $[p, q]$ with $-1 \leq p < q \leq 1$ by

$$\hat{x}_i = p + (q - p) \cdot \frac{x_i - \min(\mathbf{x})}{\max(\mathbf{x}) - \min(\mathbf{x})}, \quad \forall i \in \{1, \ldots, n\}, \tag{4}$$



(a) Signal data of 10 classes. (b) Signal data with noise $b = 0.3$. (c) Applying augmentation techniques.

FIGURE 10. Plot of the 1D signal data for 10 classes.



(a) Without noise. (b) Noise $b = 0.5$. (c) Noise $b = 1.0$. (d) Noise $b = 1.5$. (e) Noise $b = 1.95$.

FIGURE 11. Plot of the Gramian angular summation field based on 1D signal data with added noise for the classes 0 (top row), 5 (middle row) and 9 (botton row).

and computes the cosine of the sum of the angles for the Gramian angular summation field [114] by

$$\text{GASF}_{i,j} = \cos(\phi_i + \phi_j), \quad \forall i, j \in 1, \ldots, n, \tag{5}$$

with $\phi_i = \arccos(\hat{x}_i), \forall i \in \{1, \ldots, n\}$ being the polar coordinates. We generate image datasets based on signal data with different noise parameters ($b \in \{0.0, \ldots, 1.95\}$) to show the influence of the image data on the classification accuracy. As an example, Figure 11 shows the Gramian angular summation field plots for the noise parameters $b = [0, 0.5, 1.0, 1.5, 1.95]$. We present the Gramian angular summation field for the classes 0, 5, and 9 to show the dependency of the frequency of the signal data on the Gramian angular summation field.

2) MODELS

We use the following models for classification. Our encoder for time-series classification consists of a 1D convolutional layer (filter size 50, kernel 4), a max pooling layer (pool size 4), batch normalization, and a dropout layer (20%). The image encoder consists of a layer normalization and 2D convolutional layer (filter size 200), and batch normalization with `ELU` activation. After that, we add a 1D convolutional layer (filter size 200, kernel 4), max pooling (pool size 2), batch normalization, and 20% dropout. For both models, after the dropout layer follows a cross-modal representation – i.e., an LSTM with 10 units, a Dense layer with 20 units, a batch normalization layer, and a Dense layer of 10 units (for 10 sinusoidal classes). These layers are shared between both models.

F. DETAILS ON ARCHITECTURES FOR OFFLINE HWR

In this section, we provide details about the integration of Inception [119], ResNet [120] and gated text recognizer [73] modules into the offline HWR system. All three architectures are based on publicly available implementations, but we changed or adapted the first layer for the image input and

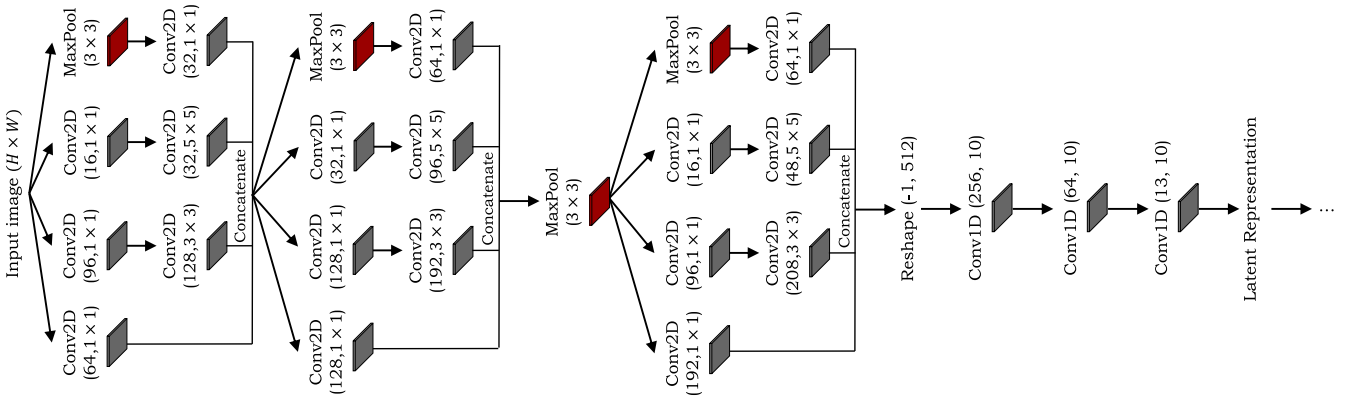


FIGURE 12. Offline HWR method based on Inception modules [119].

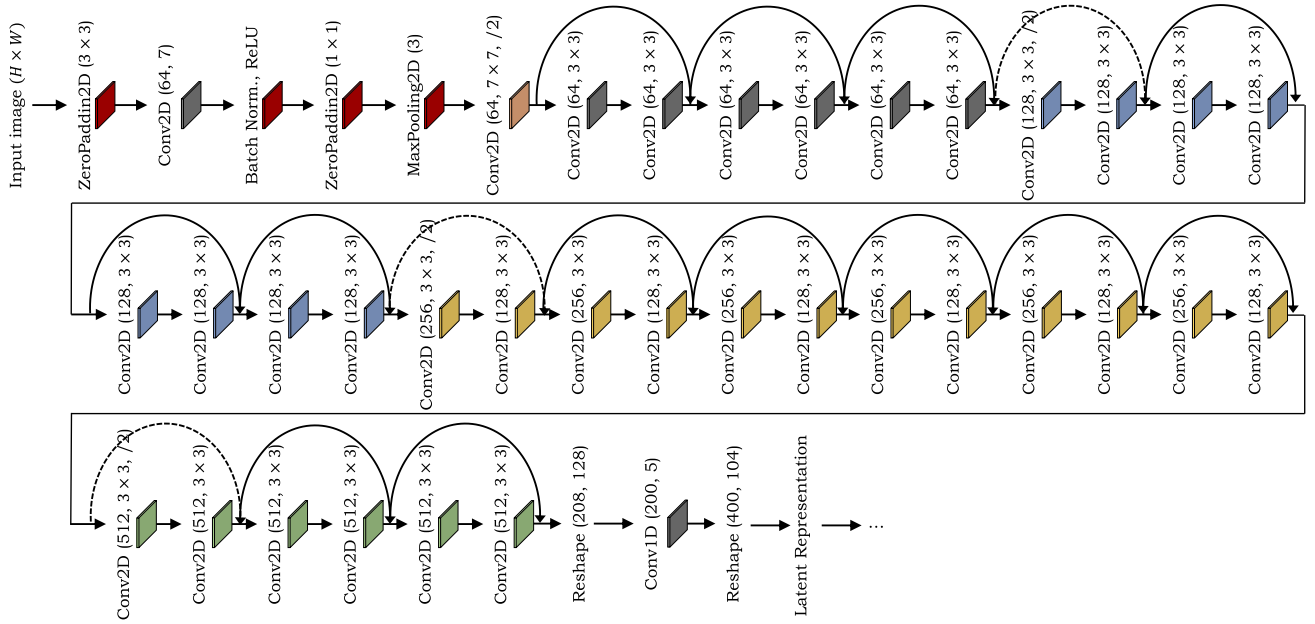


FIGURE 13. Offline HWR method based on the ResNet34 architecture [120].

the last layer for a proper input for our latent representation module.

1) INCEPTION

Figure 12 gives an overview of the integration of the Inception module. The Inception module is part of the well-known GoogLeNet architecture. The main idea is to consider how an optimal local sparse structure can be approximated by readily available dense components. As the merging of pooling layer outputs with convolutional layer outputs would lead to an inevitable increase in the number of output and would lead to a high computational increase, we apply the Inception module with dimensionality reduction to our offline HWR approach [119]. The input image is of size $H \times W$. What follows is the Inception (3a), Inception (3b), a max pooling layer (3×3) and Inception (4a). We add three 1D convolutional layers to obtain an output dimensionality of 400×200 as the input for the latent representation.

2) ResNet34

Figure 13 provides an overview of the integration of the ResNet34 architecture. Instead of learning unreferenced functions, [120] reformulated the layers as learning residual functions with reference to the layer inputs. This residual network is easier to optimize and can gain accuracy from considerably increased depth. The ResNet block allows the layers to fit a residual mapping denoted as $\mathcal{H}(\mathbf{x})$ with identity $\mathbf{x}$ and fits the mapping $\mathcal{F}(\mathbf{x}) := \mathcal{H}(\mathbf{x}) - \mathbf{x}$. The original mapping is recast into $\mathcal{F}(\mathbf{x}) + \mathbf{x}$. We reshape the output of ResNet34, add a 1D convolutional layer, and reshape the output for the latent representation.

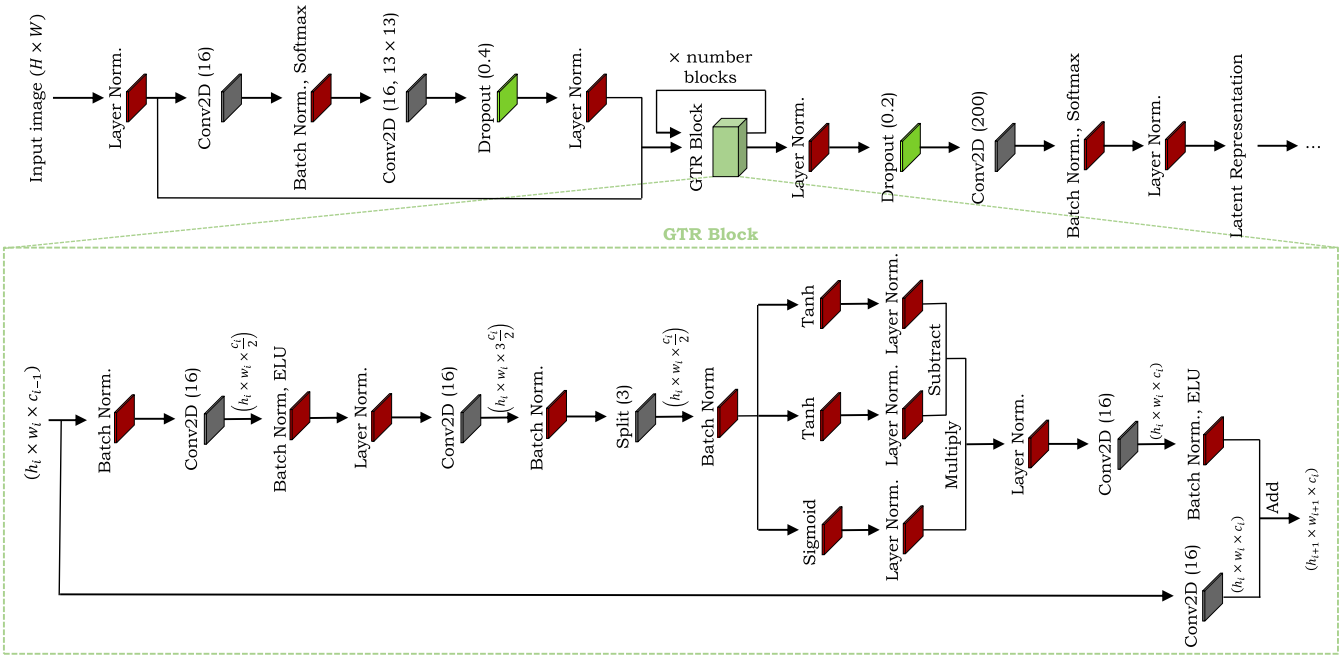


FIGURE 14. Offline HWR method based on the gated text recognizer architecture [73].

TABLE 10. Evaluation results (WER and CER in %) averaged over five splits of the baseline time-series-only technique and our cross-modal learning technique for the inertial-based OnHW datasets [38] with and without mutated vowels (MV) for two convolutional layers $c = 2$. We propose writer-(in)dependent (WD/WI) results. Best results are bold, and second best results are underlined. Arrows indicate improvements (↑) and degradation (↓) of baseline results (CNN+BiLSTM, w/o MV).

	OnHW-words500				OnHW-wordsRandom			
	WD		WI		WD		WI	
Method	WER	CER	WER	CER	WER	CER	WER	CER
Small CNN+BiLSTM, $\mathcal{L}_{\text{CTC}}$, w/ MV	51.95	17.16	60.91	27.80	41.27	7.87	84.52	35.22
CNN+BiLSTM (ours), $\mathcal{L}_{\text{CTC}}$, w/ MV	42.81	13.04	60.47	28.30	37.13	6.75	83.28	35.90
CNN+BiLSTM (ours), $\mathcal{L}_{\text{CTC}}$, w/o MV	42.77	13.44	59.82	28.54	41.52	7.81	83.54	36.51
$\mathcal{L}_{\text{MSE}}$	39.79 ↑	12.14 ↑	60.35 ↓	28.48 ↑	<u>39.98</u> ↑	7.79 ↑	<u>83.50</u> ↑	36.92 ↓
$\mathcal{L}_{\text{CS}}$	43.40 ↓	13.70 ↓	59.31 ↑	<u>27.99</u> ↑	40.31 ↑	<u>7.68</u> ↑	83.68 ↓	<u>36.30</u> ↑
$\mathcal{L}_{\text{PC}}$	38.90 ↑	<u>11.60</u> ↑	60.77 ↓	28.45 ↑	**39.93** ↑	**7.60** ↑	**83.19** ↑	**35.83** ↑
$\mathcal{L}_{\text{KL}}$	**37.25** ↑	**11.29** ↑	65.10 ↓	31.26 ↓	41.81 ↓	8.22 ↓	84.40 ↓	38.93 ↓
$\mathcal{L}_{\text{trpl,2}}(\mathcal{L}_{\text{MSE}})$	41.16 ↑	12.71 ↑	<u>58.65</u> ↑	28.19 ↑	41.16 ↑	8.03 ↓	85.38 ↓	39.49 ↓
$\mathcal{L}_{\text{trpl,2}}(\mathcal{L}_{\text{CS}})$	42.74 ↑	13.43 ↑	**58.13** ↑	**27.62** ↑	41.49 ↑	8.18 ↓	85.24 ↓	38.75 ↓
$\mathcal{L}_{\text{trpl,2}}(\mathcal{L}_{\text{PC}})$	39.94 ↑	12.19 ↑	62.76 ↓	30.68 ↓	41.58 ↓	8.18 ↓	85.18 ↓	38.53 ↓
$\mathcal{L}_{\text{trpl,2}}(\mathcal{L}_{\text{KL}})$	<u>38.34</u> ↑	11.77 ↑	67.08 ↓	33.84 ↓	41.87 ↓	8.33 ↓	86.34 ↓	40.37 ↓

3) GATED TEXT RECOGNIZER

Figure 14 gives an overview of the integration of the gated text recognizer [73] module – a fully convolutional network that uses batch normalization and layer normalization to regularize the training process and increase convergence speed. The module uses batch renormalization [169] on all batch normalization layers. Depthwise separable convolutions reduce the number of parameters at the same/better classification performance. The gated text recognizer uses spatial dropout instead of regular unstructured dropout for better regularization. After the input image of size $H \times W$ that is normalized follows a convolutional layer with `Softmax` normlization, a 13×13 filter, and dropout (40%). After the dropout layer, a stack of 2, 4, 6 or 8 gate blocks follows that models the input sequence. Similar to [73], we add a dropout of 20% after the last gated text recognizer block. Lastly, we add a 2D convolutional layer of 200, a batch normalization layer and a layer normalization layer that is the input for our latent representation.

G. DETAILED ONLINE HWR EVALUATION

Table 10 gives an overview of cross-modal representation learning results based on two convolutional layers ($c = 2$) for the cross-modal representation. Our CNN+BiLSTM contains three additional convolutional layers and outperforms the smaller CNN+BiLSTM by [38] on the WD classification tasks. Without triplet loss, $\mathcal{L}_{\text{PC}}$ yields the best results on the OnHW-wordsRandom dataset. The triplet loss

partly decreases results and partly improves results on the OnHW-words500 dataset. In conclusion, two convolutional layers for the cross-modal representation has a negative impact, while here the triplet loss has no impact.

REFERENCES

[1] Y. Peng, X. Huang, and Y. Zhao, "An overview of cross-media retrieval: Concepts, methodologies, benchmarks, and challenges," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 9, pp. 2372–2385, Sep. 2018.

[2] H. Lee, J. Lee, J. Y. Ng, and P. Natsev, "Large scale video representation learning via relational graph clustering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 6806–6815.

[3] N. Sarafianos, X. Xu, and I. Kakadiaris, "Adversarial representation learning for text-to-image matching," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, South Korea, Oct. 2019, pp. 5813–5823.

[4] V. Ranjan, N. Rasiwasia, and C. V. Jawahar, "Multi-label cross-modal retrieval," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago de Chile, CL, USA, Dec. 2015, pp. 4094–4102.

[5] X. Huang, Y. Peng, and M. Yuan, "MHTN: Modal-adversarial hybrid transfer network for cross-modal retrieval," *IEEE Trans. Cybern.*, vol. 50, no. 3, pp. 1047–1059, Mar. 2020.

[6] F. Faghri, D. J. Fleet, H. R. Kiros, and S. Fidler, "VSE++: Improving visual-semantic embeddings with hard negatives," in Proc. Brit. Mach. Vis. Conf. (BMVC), 2018, pp. 1–14. [Online]. Available: http://www.bmva.org/bmvc/2018/contents/papers/0344.pdf

[7] J. Chen, H. Hu, H. Wu, Y. Jiang, and C. Wang, "Learning the best pooling strategy for visual semantic embedding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 15784–15793.

[8] J. H. Lim, P. O. O. Pinheiro, N. Rostamzadeh, C. Pal, and S. Ahn, "Neural multisensory scene inference," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 32, 2019, pp. 8996–9006.

[9] F. M. Hafner, A. Bhuyian, J. F. P. Kooij, and E. Granger, "Cross-modal distillation for RGB-depth person re-identification," *Comput. Vis. Image Understand.*, vol. 216, Feb. 2022, Art. no. 103352.

[10] V. Vapnik and R. Izmailov, "Learning using privileged information: Similarity control and knowledge transfer," in *J. Mach. Learn. Res.*, vol. 16, pp. 2023–2049, Sep. 2015.

[11] A. Momeni and K. Tatwawadi. (2018). *Understanding LUPI (Learning Using Privileged Information)*. [Online]. Available: https://web.stanford.edu/~kedart/files/lupi.pdf

[12] Y. Wei, Y. Zhao, C. Lu, S. Wei, L. Liu, Z. Zhu, and S. Yan, "Cross-modal retrieval with CNN visual features: A new baseline," *IEEE Trans. Cybern.*, vol. 47, no. 2, pp. 449–460, Feb. 2017.

[13] T.-T. Do, T. Tran, I. Reid, V. Kumar, T. Hoang, and G. Carneiro, "A theoretically sound upper bound on the triplet loss for improving the efficiency of deep distance metric learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10396–10405.

[14] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. ICML*, Feb. 2015, pp. 97–105.

[15] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 815–823.

[16] F. Ott, D. Rügamer, L. Heublein, B. Bischl, and C. Mutschler, "Domain adaptation for time-series classification to mitigate covariate shift," in *Proc. 30th ACM Int. Conf. Multimedia*, Lisboa, Portugal, Oct. 2022, pp. 5934–5943.

[17] S. Deldari, H. Xue, A. Saeed, D. V. Smith, and F. D. Salim, "COCOA: Cross modality contrastive learning for sensor data," in *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol. (IMWUT)*, vol. 6, Sep. 2022, pp. 1–28.

[18] Y. Jain, C. Ian Tang, C. Min, F. Kawsar, and A. Mathur, "ColloSSL: Collaborative self-supervised learning for human activity recognition," 2022, *arXiv:2202.00758*.

[19] S. Venkataramanan, E. Kijak, L. Amsaleg, and Y. Avrithis, "AlignMixup: Improving representations by interpolating aligned features," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 19174–19183. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2022/html/Venkataramanan_AlignMixup_Improving_Representations_by_Interpolating_Aligned_Features_CVPR_2022_paper.html

[20] Q. Wan and Q. Zou, "Learning metric features for writer-independent signature verification using dual triplet loss," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 3853–3859.

[21] W. Li, X. Yang, M. Kong, L. Wang, J. Huo, Y. Gao, and J. Luo, "Triplet is all you need with random mappings for unsupervised visual representation learning," 2021, *arXiv:2107.10419*.

[22] S. Kim, D. Kim, M. Cho, and S. Kwak, "Proxy anchor loss for deep metric learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3235–3244. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2020/html/Kim_Proxy_Anchor_Loss_for_Deep_Metric_Learning_CVPR_2020_paper.html

[23] Y. Zhen, P. Rai, H. Zha, and L. Carin, "Cross-modal similarity learning via pairs, preferences, and active supervision," in *Proc. AAAI*, Feb. 2015, pp. 3203–3209.

[24] D. Zhang and Z. Zheng, "Joint representation learning with deep quadruplet network for real-time visual tracking," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Glasgow, U.K., Jul. 2020, pp. 1–8.

[25] A. F. Biten, A. Mafla, L. Gómez, and D. Karatzas, "Is an image worth five sentences? A new look into semantics for image-text matching," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Waikoloa, HI, USA, Jan. 2022, pp. 2483–2492.

[26] H. Diao, Y. Zhang, L. Ma, and H. Lu, "Similarity reasoning and filtration for image-text matching," in *Proc. Assoc. Advancement Artif. Intell. (AAAI)*, vol. 35, no. 2, 2021, pp. 1218–1226.

[27] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, vol. 139, 2021, pp. 8748–8763. [Online]. Available: https://proceedings.mlr.press/v139/radford21a.html

[28] M. M. M. Fahmy, "Online signature verification and handwriting classification," *J. Ain Shams Eng.*, vol. 1, no. 1, pp. 59–70, Sep. 2010.

[29] R. Plamondon and S. N. Srihari, "Online and off-line handwriting recognition: A comprehensive survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 63–84, Jan. 2000.

[30] F. Alimoglu and E. Alpaydin, "Combining multiple representations and classifiers for pen-based handwritten digit recognition," in *Proc. IAPR Intl. Conf. Document Anal. Recognit. (ICDAR)*, Germany, vol. 2, Aug. 1997, pp. 1–9.

[31] J. Kevin Chen, W. Xie, and Y. He, "Motion-based handwriting recognition," 2021, *arXiv:2101.06022*.

[32] M. Schrapel, M.-L. Stadler, and M. Rohs, "Pentelligence: Combining pen tip motion and writing sounds for handwritten digit recognition," in *Proc. CHI Conf. Human Factors Comput. Syst.*, Apr. 2018, pp. 1–11.

[33] W. Jeen-Shing, H. Yu-Liang, and C. Cheng-Ling, "Online handwriting recognition using an accelerometer-based pen device," in *Proc. 2nd Int. Conf. Adv. Comput. Sci. Eng.*, 2013, pp. 1–9.

[34] T. Deselaers, D. Keysers, J. Hosang, and H. A. Rowley, "GyroPen: Gyroscopes for pen-input with mobile phones," *IEEE Trans. Hum.-Mach. Syst.*, vol. 45, no. 2, pp. 263–271, Apr. 2015.

[35] F. Ott, M. Wehbi, T. Hamann, J. Barth, B. Eskofier, and C. Mutschler, "The OnHW dataset: Online handwriting recognition from IMU-enhanced ballpoint pens with machine learning," in *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol. (IMWUT)*, Cancún, Mexico, Sep. 2020, vol. 4, no. 3, p. 92.

[36] F. Ott, D. Rügamer, L. Heublein, B. Bischl, and C. Mutschler, "Joint classification and trajectory regression of online handwriting using a multi-task learning approach," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Waikoloa, HI, USA, Jan. 2022, pp. 1244–1254.

[37] A. Klaß, S. M. Lorenz, M. W. Lauer-Schmaltz, D. Rügamer, B. Bischl, C. Mutschler, and F. Ott, "Uncertainty-aware evaluation of time-series classification for online handwriting recognition with domain shift," in *Proc. IJCAI-ECAI Intl. Workshop Spatio-Temporal Reasoning Learn. (STRL)*, Vienna, Austria, vol. 3190, Jul. 2022, pp. 1–5. [Online]. Available: http://ceur-ws.org/Vol-3190/paper3.pdf

[38] F. Ott, D. Rügamer, L. Heublein, T. Hamann, J. Barth, B. Bischl, and C. Mutschler, "Benchmarking online sequence-to-sequence and character-based handwriting recognition from IMU-enhanced pens," *Int. J. Document Anal. Recognit.*, vol. 25, no. 4, pp. 385–414, Sep. 2022.

[39] A. Vinciarelli and M. P. Perrone, "Combining online and offline handwriting recognition," in *Proc. IAPR Intl. Conf. Document Anal. Recognit. (ICDAR)*, Edinburgh, U.K., Aug. 2003, pp. 844–848.

[40] S. Fogel, H. Averbuch-Elor, S. Cohen, S. Mazor, and R. Litman, "ScrabbleGAN: Semi-supervised varying length handwritten text generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 4323–4332.

[41] M. Yousef and T. E. Bishop, "OrigamiNet: Weakly-supervised, segmentation-free, one-step, full page text recognition by learning to unfold," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 14698–14707.

[42] R. Hussain, A. Raza, I. Siddiqi, K. Khurshid, and C. Djeddi, "A comprehensive survey of handwritten document benchmarks: Structure, usage and evaluation," *EURASIP J. Image Video Process.*, vol. 2015, no. 1, p. 46, Dec. 2015.

[43] R. Bertolami and H. Bunke, "Hidden Markov model-based ensemble methods for offline handwritten text line recognition," *Pattern Recognit.*, vol. 41, no. 11, pp. 3452–3460, Nov. 2008.

[44] P. Dreuw, P. Doetsch, C. Plahl, and H. Ney, "Hierarchical hybrid MLP/HMM or rather MLP features for a discriminatively trained Gaussian HMM: A comparison for offline handwriting recognition," in *Proc. 18th IEEE Int. Conf. Image Process.*, Brussels, Belgium, Sep. 2011, pp. 3541–3544.

[45] N. Li, J. Chen, H. Cao, B. Zhang, and P. Natarajan, "Applications of recurrent neural network language model in offline handwriting recognition and word spotting," in *Proc. 14th Int. Conf. Frontiers Handwriting Recognit.*, Hersonissos, Greece, Sep. 2014, pp. 134–139.

[46] J. Pastor-Pellicer, S. España-Boquera, M. J. Castro-Bleda, and F. Zamora-Martínez, "A combined convolutional neural network and dynamic programming approach for text line normalization," in *Proc. 13th Int. Conf. Document Anal. Recognit. (ICDAR)*, Tunis, Tunisia, Aug. 2015, pp. 341–345.

[47] S. España-Boquera, M. J. Castro-Bleda, J. Gorbe-Moya, and F. Zamora-Martinez, "Improving offline handwritten text recognition with hybrid HMM/ANN models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 4, pp. 767–779, Apr. 2011.

[48] A. Poznanski and L. Wolf, "CNN-N-gram for handwriting word recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 2305–2314.

[49] T. Wang, Y. Zhu, L. Jin, C. Luo, X. Chen, Y. Wu, Q. Wang, and M. Cai, "Decoupled attention network for text recognition," in *Proc. Assoc. Advancement Artif. Intell. (AAAI)*, Apr. 2020, vol. 34, no. 7, pp. 12216–12224.

[50] A. Sharma, R. Ambati, and D. B. Jayagopi, "Towards faster offline handwriting recognition using temporal convolutional networks," in *Proc. NCVPRIPG Commun. Comput. Inf. Sci. (CCIS)*. Singapore: Springer, vol. 1249, Nov. 2020, pp. 344–354.

[51] A. Sharma and D. B. Jayagopi, "Towards efficient unconstrained handwriting recognition using dilated temporal convolution network," *Exp. Syst. Appl.*, vol. 164, Feb. 2021, Art. no. 114004.

[52] A. Chowdhury and L. Vig, "An efficient end-to-end neural model for handwritten text recognition," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2018, pp. 1–11. [Online]. Available: http://www.bmva.org/bmvc/2018/contents/papers/0606.pdf

[53] J. Sueiras, V. Ruiz, A. Sanchez, and J. F. Velez, "Offline continuous handwriting recognition using sequence to sequence neural networks," *Neurocomputing*, vol. 289, pp. 119–128, May 2018.

[54] R. R. Ingle, Y. Fujii, T. Deselaers, J. Baccash, and A. C. Popat, "A scalable handwritten text recognition system," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sydney, NSW, Australia, Sep. 2019, pp. 17–24.

[55] J. Michael, R. Labahn, T. Grüning, and J. Zöllner, "Evaluating sequence-to-sequence models for handwritten text recognition," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sydney, NSW, Australia, Sep. 2019, pp. 1286–1293.

[56] A. Graves and J. Schmidhuber, "Offline handwriting recognition with multidimensional recurrent neural networks," in *Proc. 21st Int. Conf. Neural Inf. Process. Syst.*, Dec. 2008, pp. 545–552.

[57] T. Bluche, "Joint line segmentation and transcription for end-to-end handwritten paragraph recognition," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Dec. 2016, pp. 838–846.

[58] P. Voigtlaender, P. Doetsch, and H. Ney, "Handwriting recognition with large multidimensional long short-term memory recurrent neural networks," in *Proc. 15th Int. Conf. Frontiers Handwriting Recognit. (ICFHR)*, Shenzhen, China, Oct. 2016, pp. 228–233.

[59] Z. Chen, Y. Wu, F. Yin, and C.-L. Liu, "Simultaneous script identification and handwriting recognition via multi-task learning of recurrent neural networks," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Kyoto, Japan, vol. 1, Nov. 2017, pp. 525–530.

[60] T. Bluche, J. Louradour, and R. Messina, "Scan, attend and read: End-to-end handwritten paragraph recognition with MDLSTM attention," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Kyoto, Japan, vol. 1, Nov. 2017, pp. 1050–1055.

[61] D. Castro, B. L. D. Bezerra, and M. Valença, "Boosting the deep multidimensional long-short-term memory network for handwritten recognition systems," in *Proc. 16th Int. Conf. Frontiers Handwriting Recognit. (ICFHR)*, Niagara Falls, NY, USA, Aug. 2018, pp. 127–132.

[62] P. Krishnan, K. Dutta, and C. V. Jawahar, "Word spotting and recognition using deep embedding," in *Proc. 13th IAPR Int. Workshop Document Anal. Syst. (DAS)*, Vienna, Austria, Apr. 2018, pp. 1–6.

[63] L. Kang, P. Riba, M. Rusiñol, A. Fornés, and M. Villegas, "Pay attention to what you read: Non-recurrent handwritten text-line recognition," *Pattern Recognit.*, vol. 129, Sep. 2022, Art. no. 108766.

[64] V. Pham, T. Bluche, C. Kermorvant, and J. Louradour, "Dropout improves recurrent neural networks for handwriting recognition," in *Proc. 14th Int. Conf. Frontiers Handwriting Recognit.*, Hersonissos, Greece, Sep. 2014, pp. 285–290.

[65] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 855–868, May 2009.

[66] J. Puigcerver, "Are multidimensional recurrent layers really necessary for handwritten text recognition?" in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Kyoto, Japan, Nov. 2017, pp. 67–72.

[67] T. Bluche and R. Messina, "Gated convolutional recurrent neural networks for multilingual handwriting recognition," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Kyoto, Japan, vol. 1, Nov. 2017, pp. 646–651.

[68] D. Liang, W. Xu, and Y. Zhao, "Combining word-level and character-level representations for relation classification of informal text," in *Proc. 2nd Workshop Represent. Learn. for NLP*, Vancouver, BC, Canada, 2017, p. 43.

[69] S. Sudholt and G. A. Fink, "Attribute CNNs for word spotting in handwritten documents," *Int. J. Document Anal. Recognit.*, vol. 21, pp. 199–218, Feb. 2018.

[70] Y. Xiao and K. Cho, "Efficient character-level document classification by combining convolution and recurrent layers," 2016, *arXiv:1602.00367*.

[71] V. Carbune, P. Gonnet, T. Deselaers, H. A. Rowley, A. Daryin, M. Calvo, L.-L. Wang, D. Keysers, S. Feuz, and P. Gervais, "Fast multi-language LSTM-based online handwriting recognition," *Int. J. Document Anal. Recognit.*, vol. 23, pp. 89–102, Feb. 2020.

[72] B. Tian, Y. Zhang, J. Wang, and C. Xing, "Hierarchical inter-attention network for document classification with multi-task learning," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 3569–3575.

[73] M. Yousef, K. F. Hussain, and U. S. Mohammed, "Accurate, data-efficient, unconstrained text recognition with convolutional neural networks," *Pattern Recognit.*, vol. 108, Dec. 2020, Art. no. 107482. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0031320320302855

[74] A. Graves, "Generating sequences with recurrent neural networks," 2013, *arXiv:1308.0850*.

[75] B. Ji and T. Chen, "Generative adversarial network for handwritten text," 2019, *arXiv:1907.11845*.

[76] E. Aksan, F. Pece, and O. Hilliges, "DeepWriting: Making digital ink editable via deep generative modeling," in *Proc. CHI Conf. Human Factors Comput. Syst.*, Apr. 2018, pp. 1–14.

[77] T. S. F. Haines, O. Mac Aodha, and G. J. Brostow, "My text in your handwriting," *ACM Trans. Graph.*, vol. 35, no. 3, pp. 1–18, May 2016.

[78] E. Alonso, B. Moysset, and R. Messina, "Adversarial generation of handwritten text images conditioned on sequences," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sydney, NSW, Australia, Sep. 2019, pp. 481–486.

[79] H. Zhang, L. Chen, Y. Zhang, R. Hu, C. He, Y. Tan, and J. Zhang, "A wearable real-time character recognition system based on edge computing-enabled deep learning for air-writing," *J. Sensors*, vol. 2022, pp. 1–12, May 2022.

[80] Y. Bu, L. Xie, Y. Yin, C. Wang, J. Ning, J. Cao, and S. Lu, "Handwriting-assistant: Reconstructing continuous strokes with millimeter-level accuracy via attachable inertial sensors," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 5, no. 4, pp. 1–25, Dec. 2021.

[81] G. He, Z. Wu, Y. Wu, P. Lin, and J. Huangfu, "Online handwriting recognition based on microphone and IMU," in *Proc. IEEE 5th Int. Conf. Electron. Technol. (ICET)*, Chengdu, China, May 2022, pp. 1–10.

[82] S. K. Singh and A. Chaturvedi, "Leveraging deep feature learning for wearable sensors based handwritten character recognition," *Biomed. Signal Process. Control*, vol. 80, Feb. 2023, Art. no. 104198.

[83] Q. He, Z. Feng, X. Wang, Y. Wu, and J. Wang, "A smart pen based on triboelectric effects for handwriting pattern tracking and biometric identification," *ACS Appl. Mater. Interfaces*, vol. 14, no. 43, pp. 49295–49302, Oct. 2022.

[84] T. T. Alemayoh, M. Shintani, J. H. Lee, and S. Okamoto, "Deep-Learning-Based character recognition from handwriting motion data captured using IMU and force sensors," *Sensors*, vol. 22, no. 20, p. 7840, Oct. 2022.

[85] F. Kreß, A. Serdyuk, T. Hotfilter, J. Hoefer, T. Harbaum, J. Becker, and T. Hamann, "Hardware-aware workload distribution for AI-based online handwriting recognition in a sensor pen," in *Proc. Medit. Conf. Embedded Comput. (MECO)*, Budva, Montenegro, Jun. 2022, pp. 1–4. [Online]. Available: https://ieeexplore.ieee.org/document/9797131

[86] F. Ott, D. Rügamer, L. Heublein, B. Bischl, and C. Mutschler, "Representation learning for tablet and paper domain adaptation in favor of online handwriting recognition," 2023, *arXiv:2301.06293*.

[87] L. Wegmeth, A. Hoelzemann, and K. Van Laerhoven, "Detecting handwritten mathematical terms with sensor based data," 2021, *arXiv:2109.05594*.

[88] M. Bronkhorst, "A pen is all you need," in *Proc. 20th Student Conf. IT*, Enschede, The Netherlands, 2021, pp. 1–7.

[89] S. S. Singh and S. Karayev, "Full page handwriting recognition via image to sequence extraction," in *Proc. IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Lausanne, Switzerland, Mar. 2021, pp. 55–69.

[90] H. Azimi, S. Chang, J. Gold, and K. Karabina, "Improving accuracy and explainability of online handwriting recognition," 2022, *arXiv:2209.09102*.

[91] N. Rasiwasia, J. Costa Pereira, E. Coviello, G. Doyle, G. R. G. Lanckriet, R. Levy, and N. Vasconcelos, "A new approach to cross-modal multimedia retrieval," in *Proc. 18th ACM Int. Conf. Multimedia*, Oct. 2010, pp. 251–260.

[92] S. Deldari, H. Xue, A. Saeed, J. He, D. V. Smith, and F. D. Salim, "Beyond just vision: A review on self-supervised representation learning on multimodal and temporal data," 2022, *arXiv:2206.02353*.

[93] X. Gu, L. Ou, D. Ong, and Y. Wang, "MM-ALT: A multimodal automatic lyric transcription system," in *Proc. 30th ACM Int. Conf. Multimedia*, Oct. 2022, pp. 3328–3337.

[94] X. Zeng, D. Xiang, L. Peng, C. Liu, and X. Ding, "Local discriminant training and global optimization for convolutional neural network based handwritten Chinese character recognition," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Kyoto, Japan, vol. 1, Nov. 2017, pp. 382–387.

[95] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "SphereFace: Deep hypersphere embedding for face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 6738–6746.

[96] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. IEEE/CVF Intl. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, San Diego, CA, USA, Jun. 2005, pp. 1–9.

[97] T. Yoshida, I. Takeuchi, and M. Karasuyama, "Safe triplet screening for distance metric learning," *Neural Comput.*, vol. 31, no. 12, pp. 2432–2491, Dec. 2019.

[98] B. Harwood, V. Kumar B. G., G. Carneiro, I. Reid, and T. Drummond, "Smart mining for deep metric learning," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 2840–2848.

[99] D. Semedo and J. Magalhães, "Adaptive temporal triplet-loss for cross-modal embedding learning," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 1152–1161.

[100] D. Guo, S. Tang, and M. Wang, "Connectionist temporal modeling of video and language: A joint model for translation and sign labeling," in *Proc. Twenty-Eighth Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 751–757.

[101] D. Zeng, Y. Yu, and K. Oyama, "Deep triplet neural networks with cluster-CCA for audio-visual cross-modal retrieval," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 16, no. 3, pp. 1–23, Aug. 2020.

[102] U. Chaudhuri, B. Banerjee, A. Bhattacharya, and M. Datcu, "CrossATNet—A novel cross-attention based framework for sketch-based image retrieval," *Image Vis. Comput.*, vol. 104, Dec. 2020, Art. no. 104003.

[103] A. Gordo and D. Larlus, "Beyond instance-level image retrieval: Leveraging captions to learn a global visual representation for semantic retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 5272–5281.

[104] C. Deng, Z. Chen, X. Liu, X. Gao, and D. Tao, "Triplet-based deep hashing network for cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3893–3903, Aug. 2018.

[105] J. Zhang, Y. Kalantidis, M. Rohrbach, A. Elgammal, and M. Elhoseiny, "Large-scale visual relationship understanding," in *Proc. Assoc. Advancement Artif. Intell. (AAAI)*, Jul. 2019, vol. 33, no. 1, pp. 9185–9194.

[106] H. Bredin, "TristouNet: Triplet loss for speaker turn embedding," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 5430–5434.

[107] J. Wang, T. Gong, Z. Zeng, C. Sun, and Y. Yan, "C^3 CMR: Cross-modality cross-instance contrastive learning for cross-media retrieval," in *Proc. 30th ACM Int. Conf. Multimedia*, Oct. 2022, pp. 4300–4308.

[108] Y. Ohishi, M. Delcroix, T. Ochiai, S. Araki, D. Takeuchi, D. Niizumi, A. Kimura, N. Harada, and K. Kashino, "ConceptBeam: Concept driven target speech extraction," in *Proc. 30th ACM Int. Conf. Multimedia*, Oct. 2022, pp. 4252–4260.

[109] K. Musgrave, S. Belongie, and S.-N. Lim, "A metric learning reality check," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 681–699.

[110] H. Rantzsch, H. Yang, and C. Meinel, "Signature embedding: Writer independent offline signature verification with deep metric learning," in *Proc. Adv. Vis. Comput. (ISVC)*, Dec. 2016, pp. 616–625.

[111] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," 2017, *arXiv:1703.07737*.

[112] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Dec. 2005, pp. 1473–1480.

[113] S. Liu, E. Johns, and A. J. Davison, "End-to-end multi-task learning with attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 1871–1880.

[114] Z. Wang and T. Oates, "Imaging time-series to improve classification and imputation," in *Proc. 24th Int. Conf. Artif. Intell. (IJCAI)*, Buenos Aires, Argentinia, Jul. 2015, pp. 3939–3945.

[115] M. Liwicki and H. Bunke, "IAM-OnDB—An on-line English sentence database acquired from handwritten text on a whiteboard," in *Proc. 8th Int. Conf. Document Anal. Recognit. (ICDAR)*, Seoul, South Korea, 2005, pp. 956–961.

[116] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2298–2304, Nov. 2017.

[117] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," in *Intl. Conf. Learn. Represent. (ICLR)*, 2019, pp. 1–9. [Online]. Available: https://openreview.net/forum?id=B1xsqj09Fm

[118] J. H. Lim and J. C. Ye, "Geometric GAN," 2017, *arXiv:1705.02894*.

[119] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1–9.

[120] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.

[121] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," in *Proc. ACM SIGGRAPH Papers*, Jul. 2007, vol. 26, no. 3, p. 10.

[122] C. Wigington, S. Stewart, B. Davis, B. Barrett, B. Price, and S. Cohen, "Data augmentation for recognition of handwritten words and lines using a CNN-LSTM network," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Kyoto, Japan, vol. 1, Nov. 2017, pp. 639–645.

[123] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008. [Online]. Available: https://www.jmlr.org/papers/v9/vandermaaten08a.html

[124] H. I. Fawaz, B. Lucas, G. Forestier, C. Pelletier, D. F. Schmidt, J. Weber, G. I. Webb, L. Idoumghar, P.-A. Müller, and F. Petitjean, "InceptionTime: Finding AlexNet for time series classification," *Data Mining Knowl. Discovery*, vol. 34, no. 6, pp. 1936–1962, Nov. 2020.

[125] A. Mattick, M. Mayr, M. Seuret, A. Maier, and V. Christlein, "SmartPatch: Improving handwritten word imitation with patch discriminators," in *Proc. IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Lausanne, Switzerland, Sep. 2021, pp. 268–283. [Online]. Available: https://dl.acm.org/doi/abs/10.1007/978-3-030-86549-8_18

[126] L. Kang, P. Riba, M. Rusinol, A. Fornés, and M. Villegas, "Content and style aware generation of text-line images for handwriting recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 12, pp. 8846–8860, Oct. 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9585646

[127] C. Luo, Y. Zhu, L. Jin, Z. Li, and D. Peng, "SLOGAN: Handwriting style synthesis for arbitrary-length and out-of-vocabulary text," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Feb. 28, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9722567

[128] J. Gan, W. Wang, J. Leng, and X. Gao, "HiGAN+: Handwriting imitation GAN with disentangled representations," *ACM Trans. Graph.*, vol. 42, no. 1, pp. 1–17, Feb. 2023, doi: 10.1145/3550070.

[129] C. Chen, Z. Fu, Z. Chen, S. Jin, Z. Cheng, X. Jin, and X. S. Hua, "HoMM: Higher-order moment matching for unsupervised domain adaptation," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, Apr. 2020, vol. 34, no. 4, pp. 3422–3429. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/5745

[130] E. Grosicki and H. El-Abed, "ICDAR 2011–French handwriting recognition competition," in *Proc. Int. Conf. Document Anal. Recognit.*, Beijing, China, Sep. 2011, pp. 1459–1463.

[131] M. Kozielski, P. Doetsch, and H. Ney, "Improvements in RWTH's system for off-line handwriting recognition," in *Proc. 12th Int. Conf. Document Anal. Recognit.*, Washington, DC, USA, Aug. 2013, pp. 935–939.

[132] P. Doetsch, M. Kozielski, and H. Ney, "Fast and robust training of recurrent neural networks for offline handwriting recognition," in *Proc. 14th Int. Conf. Frontiers Handwriting Recognit.*, Hersonissos, Greece, Sep. 2014, pp. 279–284.

[133] F. Menasri, J. Louradour, A.-L. Bianne-Bernard, and C. Kermorvant, "The A2iA French handwriting recognition system at the rimes-ICDAR2011 competition," *Proc. SPIE*, vol. 8295, p. 51, Jan. 2012.

[134] P. Voigtlaender, P. Doetsch, S. Wiesler, R. Schlüter, and H. Ney, "Sequence-discriminative training of recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Brisbane, QLD, Australia, Apr. 2015, pp. 2100–2104.

[135] T. Bluche, "Deep neural networks for large vocabulary handwritten text recognition," Doctorat these, Ecole Doctorale Informatique de Paris-Sud, Université Paris-Sud, Orsay, France, May 2015. [Online]. Available: https://tel.archives-ouvertes.fr/tel-01249405/document

[136] C. Wigington, C. Tensmeyer, B. Davis, W. Barrett, B. Price, and S. Cohen, "Start, follow, read: End-to-end full-page handwriting recognition," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, vol. 11210, Oct. 2018, pp. 372–388.

[137] K. Dutta, P. Krishnan, M. Mathew, and C. V. Jawahar, "Improving CNN-RNN hybrid networks for handwriting recognition," in *Proc. 16th Int. Conf. Frontiers Handwriting Recognit. (ICFHR)*, Niagara Falls, NY, USA, Aug. 2018, pp. 80–85.

[138] J. Chung and T. Delteil, "A computationally efficient pipeline approach to full page offline handwritten text recognition," in *Proc. Int. Conf. Document Anal. Recognit. Workshops (ICDARW)*, Sydney, NSW, Australia, vol. 5, Sep. 2019, pp. 35–40.

[139] M. Carbonell, J. Mas, M. Villegas, A. Fornés, and J. Lladós, "End-to-end handwritten text detection and transcription in full pages," in *Proc. Int. Conf. Document Anal. Recognit. Workshops (ICDARW)*, Sydney, NSW, Australia, vol. 5, Sep. 2019, pp. 29–34.

[140] A. K. Bhunia, A. Das, A. K. Bhunia, P. S. R. Kishore, and P. P. Roy, "Handwriting recognition in low-resource scripts using adversarial learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 4762–4771.

[141] R. Messina and C. Kermorvant, "Over-generative finite state transducer N-gram for out-of-vocabulary word recognition," in *Proc. 11th IAPR Int. Workshop Document Anal. Syst.*, Tours, France, Apr. 2014, pp. 212–216.

[142] G. Bastas, K. Kritsis, and V. Katsouros, "Air-writing recognition using deep convolutional and recurrent neural network architectures," in *Proc. 17th Int. Conf. Frontiers Handwriting Recognit. (ICFHR)*, Dortmund, Germany, Sep. 2020, pp. 7–12.

[143] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. A. Ranzato, and T. Mikolov, "DeViSE: A deep visual-semantic embedding model," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2013, pp. 1–10. [Online]. Available: https://papers.nips.cc/paper/2013/hash/7cce53cf90577442771720a370c3c723-Abstract.html

[144] R. Kiros, R. Salakhutdinov, and R. S. Zemel, "Unifying visual-semantic embeddings with multimodal neural language models," 2014, *arXiv:1411.2539*.

[145] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, "From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions," in *Proc. Trans. Assoc. Comput. Linguistics*, Cambridge, MA, USA, vol. 2, pp. 67–78, Dec. 2014.

[146] Y. Huang, Q. Wu, C. Song, and L. Wang, "Learning semantic concepts and order for image and sentence matching," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 6163–6171.

[147] J. Gu, J. Cai, S. Joty, L. Niu, and G. Wang, "Look, imagine and match: Improving textual-visual cross-modal retrieval with generative models," in *Proc. IEEE/CVF Intl. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 1–9.

[148] K.-H. Lee, X. Chen, G. Hua, H. Hu, and X. He, "Stacked cross attention for image-text matching," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, vol. 11208, Oct. 2018, pp. 1–16.

[149] K. Li, Y. Zhang, K. Li, Y. Li, and Y. Fu, "Visual semantic reasoning for image-text matching," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, South Korea, Oct. 2019, pp. 4653–4661.

[150] Y. Song and M. Soleymani, "Polysemous visual-semantic embedding for cross-modal retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 1979–1988.

[151] J. Wehrmann, M. A. Lopes, D. Souza, and R. Barros, "Language-agnostic visual-semantic embeddings," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seould, South Korea, Oct. 2019, pp. 5803–5812.

[152] H. Wu, J. Mao, Y. Zhang, Y. Jiang, L. Li, W. Sun, and W.-Y. Ma, "Unified visual-semantic embeddings: Bridging vision and language with structured meaning representations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 6602–6611.

[153] L. Zhen, P. Hu, X. Wang, and D. Peng, "Deep supervised cross-modal retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 10386–10395.

[154] Y. Peng and J. Qi, "CM-GANs: Cross-modal generative adversarial networks for common representation learning," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 15, no. 1, pp. 1–24, Feb. 2019.

[155] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.

[156] H. Wang, Y. Zhang, Z. Ji, Y. Pang, and L. Ma, "Consensus-aware visual-semantic embedding for image-text matching," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 1–15. [Online]. Available: https://www.ecva.net/papers/eccv_2020/papers_ECCV/papers/123690018.pdf

[157] T. Chen, J. Deng, and J. Luo, "Adaptive offline quintuplet loss for image-text matching," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 1–5. [Online]. Available: https://www.springerprofessional.de/adaptive-offline-quintuplet-loss-for-image-text-matching/18635224

[158] S. Chun, S. J. Oh, R. Sampaio de Rezende, Y. Kalantidis, and D. Larlus, "Probabilistic embeddings for cross-modal retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, Jun. 2021, pp. 8411–8420.

[159] B. Chen, A. Rouditchenko, K. Duarte, H. Kuehne, S. Thomas, A. Boggust, R. Panda, B. Kingsbury, R. Feris, D. Harwath, J. Glass, M. Picheny, and S.-F. Chang, "Multimodal clustering networks for self-supervised learning from unlabeled videos," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, USA, Oct. 2021, pp. 7992–8001.

[160] H. Akbari, L. Yuan, R. Qian, W.-H. Chuang, S.-F. Chang, Y. Cui, and B. Gong, "VATT: Transformers for multimodal self-supervised learning from raw video, audio and text," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, May 2021, pp. 1–15. [Online]. Available: https://openreview.net/forum?id=RzYrn625bu8

[161] L. Wang, P. Luc, A. Recasens, J.-B. Alayrac, and A. van den Oord, "Multimodal self-supervised learning of general audio representations," 2021, *arXiv:2104.12807*.
[162] A. Guzhov, F. Raue, J. Hees, and A. Dengel, "Audioclip: Extending clip to image, text and audio," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Singapore, May 2022, pp. 976–980.
[163] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "data2vec: A general framework for self-supervised learning in speech, vision and language," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 162, 2022, pp. 1298–1312. [Online]. Available: https://proceedings.mlr.press/v162/baevski22a.html
[164] A. Piergiovanni, A. Angelova, and M. S. Ryoo, "Evolving losses for unsupervised video representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 130–139.
[165] H. Lin, Z. Ma, X. Hong, Y. Wang, and Z. Su, "Semi-supervised crowd counting via density agency," in *Proc. 30th ACM Int. Conf. Multimedia*, Oct. 2022, pp. 1416–1426.
[166] A. Singh, R. Hu, V. Goswami, G. Couairon, W. Galuba, M. Rohrbach, and D. Kiela, "FLAVA: A foundational language and vision alignment model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 15617–15629.
[167] A. Falcon, G. Serra, and O. Lanz, "A feature-space multimodal data augmentation technique for text-video retrieval," in *Proc. 30th ACM Int. Conf. Multimedia*, Oct. 2022, pp. 4385–4394.
[168] D. Chen, M. Wang, H. Chen, L. Wu, J. Qin, and W. Peng, "Cross-modal retrieval with heterogeneous graph embedding," in *Proc. ACM Int. Conf. Multimedia (ACMMM)*, Oct. 2022, pp. 3291–3300.
[169] S. Ioffe, "Batch renormalization: Towards reducing minibatch dependence in batch-normalized models," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 1942–1950.

FELIX OTT (Member, IEEE) received the M.Sc. degree in computational engineering from Friedrich-Alexander-Universität (FAU) Erlangen-Nürnberg, Germany, in 2019. He is currently pursuing the Ph.D. degree with the Probabilistic Machine and Deep Learning Group, Ludwig-Maximilians-Universität (LMU), Munich. He joined the Hybrid Positioning & Information Fusion Group, Locating and Communication Systems Department, Fraunhofer IIS. His research interest includes multimodal information fusion for self-localization.

DAVID RÜGAMER received the Ph.D. degree from LMU Munich, in 2018. Before joining the LMU Munich as an Associate Professor in 2023, he was Interim Professor at the TU Dortmund and RWTH Aachen. His research focuses on statistical modeling with neural networks as well as their uncertainty quantification and sparsification.

LUCAS HEUBLEIN received the M.Sc. degree in integrated life science with FAU Erlangen-Nürnberg, where he is currently pursuing the degree in computer science. He joined the Hybrid Positioning & Information Fusion Group, Fraunhofer IIS, in 2020, as a Student Assistant.

BERND BISCHL is currently a Full Professor in statistical learning and data science with LMU Munich and the Director of the Munich Center of Machine Learning. His research interests include AutoML, interpretable machine learning, and machine learning benchmarking.

CHRISTOPHER MUTSCHLER received the Diploma and Ph.D. degrees from FAU Erlangen-Nürnberg, in 2010 and 2014, respectively. He leads the Precise Positioning and Analytics Department, Fraunhofer IIS. Prior to that, he headed the Machine Learning & Information Fusion Group. He gives lectures on machine learning with FAU Erlangen-Nürnberg. His research interest includes machine learning with radio-based localization.