Learning Generalisable Representations for Offline Signature Verification

Xianmu Cairang[1], Duoji Zhaxi[2], Xiaolong Yang[2], Yan Hou[2],
Qijun Zhao[1,2,*], Dingguo Gao[2], Danzeng Pubu[2], and Dorji Gesang[2]
[1]College of Computer Science, Sichuan University, ChengDu, China
[2]School of Information Science and Technology, Tibet University, Lhasa, China
Email: cairangxianmu@stu.scu.edu.cn, qjzhao@scu.edu.cn, {zhaduo, yangxiaolong, houyan, gdg, pbdz, gsdj}@utibet.edu.cn

***Abstract*—Current offline signature verification methods based on deep learning have achieved promising results, but these methods degrade greatly in cross-domain settings. An efficient offline signature verification model with both high performance and for deployment cross-domain without any adaptation. In this paper, we propose a novel approach to learning generalisable representations for offline signature verification. Firstly, we use the Siamese network combined with Triplet loss and Cross Entropy (CE) loss to learn discriminative features. Secondly, we introduce Instance Normalization (IN) into the network to cope with cross-domain discrepancies and propose an Inference Layer Normalization Neck (ILNNeck) module to further improve model generalization. We evalute the method on our self-collected Multi-lingual Signature dataset (MLSig) and three public datasets: BHSig-H, BHSig-B, and CEDAR. Results show that while our method achieves comparable results in single-domain setting, it is obviously superior to state-of-the-art methods in cross-domain setting.**

***Index Terms*—Offline Signature Verification, Cross-Domain Generalization**

I. Introduction

Signature verification is an important biometric technology in banking, finance, and other industries. Compared with online signature, offline signature is more widely used due to its device independence. However, verifying offline signature is more challenging, because it lacks dynamic information of signature action. Existing signature verification methods face two major challenges. The first challenge is intra-class variation and inter-class ambiguity, because the arbitrariness of signatures can lead to large variation in signatures of the same persons, while forged signatures by skillful forgery have high similarity with the genuine ones. For example, as can be seen from Fig.1 (a) and Fig.1 (e). the genuine signatures in the first column look more different to other genuine signatures of the same persons in the second column than to the forgery ones in the third column. The second challenge is generalizable feature learning. Different datasets are collected according to different standards, which may lead to domain discrepancies between datasets due to differences in signature forgery capabilities, writing instruments, languages, image aspect ratios, etc. For example, Fig.1 (a) and Fig.1 (b) show two signature images from two datasets, which have obviously different backgrounds and writing styles. These differences are usually dataset specific or language specific. Consequently,

Fig. 1. Example signature images in different languages from different datasets. Left to right columns: Reference signature, genuine signature, forged signature.

models that are trained on one dataset may performing poorly on other datasets.

In the past decade, many solutions have been proposed based either on traditional methods [1], [2] or on deep learning methods [3]–[7], and achieved promising results in resolving the first challenge. However, little attention has been paid to the second challenge. Existing offline signature verification methods [3]–[5] trained on a single-domain dataset usually suffer from large performance drops in cross-domain settings, suggesting that these methods are not domain generalizable. Therefore, it is highly demanded to develop a generalisable signature verification model that can be directly deployed in unknown domains without adaptation after trained on a source domain. To this end, we propose in this paper a novel method that can learn generalisable and discriminative feature representations for cross-domain offline signature verification. Firstly, Siamese network [3] equipped with Triplet loss [8] and Cross Entropy(CE) loss is employed to learn discriminative features. Secondly, the cross-domain difference is alleviated through the Instance Normalization (IN) [9], which is initially proposed in low-level vision tasks [10], [11], and has demonstrated its effectiveness in eliminating pedestrian appearance differences in person re-identification [12]. Finally, an Inference Layer Normalization Neck (ILNNeck) module is proposed to further enhance the generalization ability of signature verification models. To evaluate the effectiveness of our method, we collect a Multi-lingual Signature dataset (MLSig) of Chinese, English, and Tibetan signatures by 200 persons. The results on MLSig and another three public datasets demonstrate the superiority of our method in cross-domain setting.

To sum up, our contributions in this paper are three fold: (i) a novel method that can enhance both discrimination and generalization of learned signature features; (ii) a multi-lingual offline signature dataset consisting of Chinese, English, and Tibetan signatures; (iii) comprehensive evaluation experiments proving that our method can achieve state-of-the-art performance for cross-domain offline signature verification.

TABLE I
COMPARISON BETWEEN DIFFERENT OFFLINE SIGNATURE DATASETS.

Dataset	Language	Signers	Genuine	Forged
CEDAR [13]	English	55	24	24
MCYT75 [14]	English	75	25	25
GPDS Synthetic [15]	English	4000	24	30
SigComp2011 [17]	Dutch	70	48	16
	Chinese	20	48	48
BHSig260 [16]	Hindi	160	24	30
	Bengali	100	24	30
MLSig (Ours)	Chinese	200	15	15
	English	200	15	15
	Tibetan	200	15	15

II. Related Work

A. Datasets

Many publicly available offline signature datasets have now been published, as shown in Table I. Most datasets are in a single language, such as the English dataset CEDAR [13], MCYT-74 [14] and GPDS synthesis [15]. There are also relatively few multilingual datasets, such as the BHSig260 dataset [16] containing Hindi and Bengali, and the competition dataset SigComp2011 [17] containing Dutch and Chinese. We can see that there are many English signature datasets, but fewer datasets for non-Western languages, and fewer multilingual datasets, so our proposed dataset is supplemented with more languages.

B. Single Domain Signature Verification

Early offline signature verification methods use geometric [18]–[20] or statistical [21]–[24] features. They are complex to implement and do not work well for signatures with complex backgrounds, noise interference, or skilled forgery. Deep learning based methods [25] alleviate these problems. Hafemann et al. [26] proposed a writer-dependent two-stage CNN network SigNet. Dey et al. [3] proposed a writer-independent method based on the Siamese network. Li et al. [4] proposed a 2-channel-2-logit single-branch network DeepHSV. Wei et al. proposed the Inverse Discriminative Network IDN. These methods require both genuine and forged signatures to train the networks. Hafemann et al. [6] proposed a meta-learning-based approach to constructing a model when only genuine signatures are available during training. Shaikh et al. [27] provided interpretable results through Attention. Li et al. [28] proposed Static-Dynamic Interaction Network (SDINet) to enhance dynamic feature extraction capabilities in static signatures. Liu et al. [7] proposed signature verification methods based on Regional Features. These methods have all achieved good performance for single-domain signature verification, but fail to consider the challenges in cross-domain settings.

C. Cross Domain Signature Verification

Das et al. [29] explored for the first time a method of combining multiple single-scripts into multi-scripts through a statistical analysis method. In addition, Single-domain-based methods [3]–[6] were tested at cross-domain settings and had significantly reduced performance, but they did not propose any specific design. We propose a generalizable feature learning task particularly for cross-domain signature verification.

D. Domain Generalization

Common methods to deal with the domain shift are domain transfer, domain adaptation, and domain generalization, but domain transfer and domain adaptation methods require fine-tuning on the target domain, which is unknown in real scenarios. Most domain generalization methods such as image classificationcite [30], [31] assume that the label spaces of the source and target domains are the same, but the person who train and test in the signature verification task is different, so these methods are not suitable for signature verification. The IN [9] is based on the normalization of a single channel of a single sample, which was originally used in style transfer to discard the difference information of a specific instance [10], [11]. Recently, some studies add IN to CNN to improve model generalization. Nam et al. [32] proposed a Batch Instance Normalization method, which effectively enhances the feature representation and generalization ability. Pan et al. [33] proposed IBN-Net for improving model generalizable in semantic segmentation. Zhou et al. [12] equipped the network with IN via differentiable architecture search to improve

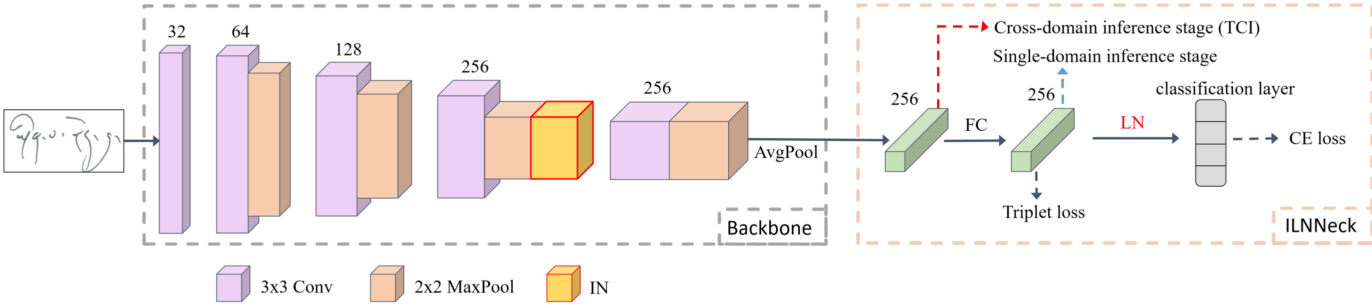


Fig. 2. The overall structure of our method. The components in red color are the novel modules we introduce: IN to learn generalisable features, LN to obtain discriminative and normalized features, and Trick of Cross-domain Inference (TCI) to improve performance in cross-domain setting.

the generalization of the model in cross-dataset person re-identification. However, the location, number, and associated parts of the IN in the network all affect the final result. Different fields have proposed special designs when applying the IN layer, which makes the existing structure cannot be directly applied to signature verification tasks. So we propose a new IN-equipped network suitable for signature verification tasks.

III. Proposed Method

A. Overview

The overall structure of our proposed method is shown in Fig. 2. Firstly, we extract features through the CNN-based Backbone module, in which the IN is incorporated to learn generalizable feature representations. Secondly, we propose the ILNNeck module to further improve the generalization. Finally, Triplet loss and CE loss are jointly imposed to enhance the discrimination of learned features. These modules are described in detail below.

B. Instance Normalization

Instance Normalization (IN) is based on the normalization of a single sample and a single channel, and is more suitable for scenarios where pixel-level and fine-grained features need to be considered, for example, signature verification. There is no unified standard for acquisition of signatures, resulting in differences in image resolution, aspect ratio, and writing paper for different datasets. Moreover, different datasets may focus on signatures of different languages, which usually differ largely in their writing habits. For example, English is usually a coherent signature, while in Chinese there is a space between each character. As a consequence, the feature representations learned by a model are quite possibly specific to the dataset used to train the model.

We believe that this is the main reason for the performance fluctuation across different datasets. Therefore, we propose to incorporate IN into the backbone network to alleviate the impact of domain differences between datasets on the generalization of learned feature representations. However, the location and number of IN in the network will directly affect the discrimination and generalization of the model. Generally, the introduction of IN into the shallow layer of the network can effectively eliminate the information of specific instances, while the introduction into the deep layer may lose some discriminant information and affect the performance of the model. Therefore, we determined the number and location of in through experiments.

C. Inference Layer Nnormalization Neck

In the person re-identification task, Batch Normalization Neck (BNNeck) [12] is proposed to normalize the feature distribution to solve the problem of inconsistency between Triplet loss and CE loss optimization spaces. Its basic idea is to introduce a Batch Normalization (BN) between the Fully Connected layer (FC) and the classification layer. We extend this idea for signature verification task by proposing the ILNNeck module as shown in Fig.2. Specifically, because BN conduct normalization across all samples in a batch and do not consider the detail of specific samples, we use Layer Normalization (LN) instead of BN, LN normalizes all channels of a sample, which makes it possible to achieve the purpose of feature normalization without losing the detail of specific samples. Because FC is a global feature, it is easy to overfit in the source domain and reduce the generalization of the model. Therefore, we recommend using the features after the first FC for inference when testing on a single domain, and using the features after Backbone for inference in a cross-domain setting, We call this Trick Cross-domain Inference (TCI).

D. Loss Function

Triplet loss can constrain the distance between positive and negative sample pairs, while CE loss provides global constraints to distinguish different classes. The combination of these two losses can achieve more discriminative feature representations. Triplet loss is calculated with non-zero loss [34]. The overall loss is as follows:

$$Loss = L_{triplet} + \alpha L_{ce} \tag{1}$$

α is a hyperparameter that balances Triplet loss and CE loss.

IV. Experiments

A. Experimental Settings

We conduct evaluation experiments on CEDAR, BHSig-H, BHSig-B, and our dataset MLSig, whose scale and language are shown in Table I. The evaluation is done in both single-domain (i.e., single-dataset and single-language) and cross-domain settings (i.e., cross-dataset and cross-language). In the single-dataset and single-language setting (single domain) of CEDAR, BHSIg-B, and BHSig-H, the dataset is divided according to [7]. Specifically, for CEDAE, we use the samples of the first 50 individuals for training and the last 5 individuals for testing. For BHSig-H, the signatures of the first 100 individuals are used for training, and the signatures of the last 60 individuals are used for testing. For BHSig-B, the signatures of the first 50 individuals are used for training, and the signatures of the last 50 individuals are used for testing. For MLSig, the signatures of the first 150 people for training and the last 50 people for testing. In the cross-dataset and cross-language settings (cross-domain) of CEDAR, BHSIg-B, and BHSig-H, we train the training set samples of one dataset and test all samples of other datasets. In the sigle-language or cross-language settings of MLSig, we use the samples in one language of the first 150 individuals for training and the samples in the the rest 50 individuals for testing.

For the sampling of sample pairs, each mini-batch randomly selects K sample pairs from the sample pairs of P individuals to construct triplets. For example, for BHSig-H, the positive sample pairs of a person have $C_{24}^2 = 276$, and the negative sample pairs have $24 \times 40 = 960$. In order to construct triplets, we select $PK/2$ positive sample pairs and negative sample pairs respectively. Note that during the testing phase we evaluate with all pairs of samples from signers.

When comparing with the existing methods, we use three metrics: Accuracy (Acc), Area Under Curve (AUC), and Equal Error Rate (EER). In other experiments, we only report AUC and EER, because the Acc is usually infinitely close to $1-$EER. These indicators can be obtained by adjusting the decision threshold. We report the results obtained based on a global decision threshold ($global\ T$) and a user-specific decision threshold ($user\ T$). Generally, the result of $global\ T$ is worse than that of $user\ T$. We compare the result of $global\ T$ with the existing writer independent methods.

For data preprocessing, we adopt the OTSU algorithm [35] to remove the image background and normalize the data with a mean of 0.5 and a standard deviation of 0.5. We adopt the Adam optimizer with a learning rate of $5e-5$. During training, we empirically set $P = 2, K = 128$. The hyperparameter $\alpha = 0.03$ is determined experimentally, and we use cosine distance metric in triplet loss.

B. Multi-lingual Signature Dataset

As shown in Table I, most of the publicly available datasets are Western script signatures, and there are few non-Western script signature datasets. Therefore, we propose a Multilingual Signature dataset (MLSig) containing Chinese, English, and Tibetan. It effectively complements some of the missing languages, and having three languages at the same time makes it more suitable for cross-language research and analysis, which to our knowledge is the first signature dataset to contain three languages at the same time. The signatures in MLSig were written by 200 students from our university, each providing signatures in Chinese, English, and Tibetan. Each language has 15 genuine signatures and 15 forged signatures. We use the signatures of the first 150 people as the training set and the signatures of the last 50 people as the test set.

TABLE II
RESULTS (%) OF OUR METHOD AND IDN METHOD ON THE MLSIG DATASET.

Mehod	MLSig-CN		MLSig-EN		MLSig-TB	
	AUC	EER	AUC	EER	AUC	EER
IDN($global\ T$) [5]	77.57	27.58	81.96	25.94	77.50	29.03
IDN($user\ T$) [5]	82.29	23.37	83.40	23.81	79.40	26.95
Ours ($global\ T$)	**86.57**	**20.93**	**90.26**	**16.42**	**91.61**	**16.02**
Ours($user\ T$)	**92.84**	**13.21**	**92.46**	**13.49**	**93.92**	**11.98**

TABLE III
RESULTS (%) IN CROSS-LANGUAGE SETTING ON MLSIG DATASET

Train\Test	Method	MLSig-CN		MLSig-EN		MLSig-TB	
		AUC	EER	AUC	EER	AUC	EER
MLSig-CN	Baseline	80.91	26.38	85.32	20.93	88.54	19.20
	Ours	**88.10**	**19.90**	**87.51**	**20.15**	**92.00**	**15.17**
MLSig-EN	Baseline	81.78	26.43	84.99	21.90	89.96	17.91
	Ours	89.57	18.67	90.26	16.42	89.85	18.37
MLSig-TB	Baseline	81.46	25.85	85.95	21.55	86.37	20.50
	Ours	**87.72**	**20.47**	**87.89**	**20.14**	**91.61**	**16.02**

We compare the performance of our method and the IDN method [5] on the MLSig dataset. The results are shown in Table II. As can be seen, our method performs better than the counterpart method. Moreover, compared with other datasets (refer to results in Table V), the performance on the MLSig dataset is lower, indicating that our proposed dataset is more challenging.

To analyze the impact of language domain differences on signature verification performance and to verify the performance of our method in a cross-language setting, we compare our method with the baseline method (i.e., remove the IN, LN, and TCI modules from the framework shown in Fig. 2.) in the cross-language setting on MLSig. According to the obtained results in Table III, our method achieves similar performance in the cross-language setting as in the single language setting, suggesting that our approach is highly generalizable to differences across language domains. It is worth noting that the baseline approach also achieves high performance at cross-language setting. This shows that language might not be the most important factor affecting the performance of cross-domain signature verification. In contrast, the collection standard has a greater impact.

TABLE IV
RESULTS (%) OF CROSS-DOMAIN COMPARISONS WITH EXISTING METHODS

Train\Test	Method	CEDAR		BHSig-H		BHSig-B	
		Acc	AUC	Acc	AUC	Acc	AUC
CEDAR	SigNet [3]	**100.00**	-	55.61	-	64.15	-
	DeepHSV [4]	-	**100.00**	-	74.00	-	76.00
	IDN [5]	95.98	-	50.36	-	50.01	-
	Cut and Compare [25]	95.66	-	60.92	-	61.93	-
	Ours	90.79	96.75	**80.83**	**89.50**	**84.93**	**93.31**
BHSig-H	SigNet [3]	59.57	-	84.64	-	60.65	-
	DeepHSV	-	53.00	-	94.00	-	87.00
	IDN	50.00	-	93.04	-	74.12	-
	Cut and Compare	70.51	-	94.03	-	85.34	-
	Ours	**79.76**	**87.70**	**95.79**	**99.28**	**87.57**	**95.29**
BHSig-B	SigNet [3]	50.00	-	52.78	-	86.81	-
	DeepHSV	-	49.00	-	82.00	-	95.00
	IDN	50.00	-	74.30	-	95.32	-
	Cut and Compare	66.29	-	69.59	-	**96.04**	-
	Ours	**80.34**	**88.08**	**79.21**	**87.16**	94.28	98.78

TABLE V
RESULTS (%) OF SINGLE-DOMAIN COMPARISON WITH EXISTING METHODS

Dataset	Method	Venue	Acc	AUC	EER
CEDAR	Surroundedness Feature [1]	PRL'2012	-	-	8.33
	SigNet [3]	PRL'2017	**100.00**	**100.00**	**0.00**
	Meta-learning Based [6]	TIFS'2019	-	-	10.21
	DeepHSV [4]	ICDAR'2019	**100.00**	**100.00**	**0.00**
	IDN [5]	CVPR'2019	-	-	3.62
	Region Based [7]	PR'2021	-	-	6.74
	Cut and Compare [25]	ICPR'2021	-	-	**4.34 / 0.00** [1]
	SDINet [28]	AAAI'2021	-	-	1.75
	Ours (*global* T)	-	90.79 / 97.00	96.75 / 99.51	9.21 / 3.00 [1]
	Ours (*user* T)	-	93.79 / 98.79	96.85 / 99.55	6.20 / 1.20 [1]
BHSig-H	SigNet	PRL'2017	84.64	-	-
	DeepHSV	ICDAR'2019	86.66	94.00	13.34
	IDN	CVPR'2019	93.40	-	-
	Attention [27]	ICFHR'2020	92.37	-	-
	Cut and Compare	ICPR'2021	-	-	5.97
	SDINet	AAAI'2021	95.00	-	-
	Ours(*global* T)	-	**95.79**	**99.28**	**4.21**
	Ours(*user* T)	-	97.75	99.37	2.25
BHSig-B	SigNet	PRL'2017	86.11	-	-
	DeepHSV	ICDAR'2019	88.08	95.50	11.92
	IDN	CVPR'2019	95.32	-	-
	Cut and Compare	ICPR'2021	-	-	**3.96**
	SDINet	AAAI'2021	94.42	-	-
	Ours(*global* T)	-	94.28	98.78	5.72
	Ours(*user* T)	-	96.85	98.49	3.14

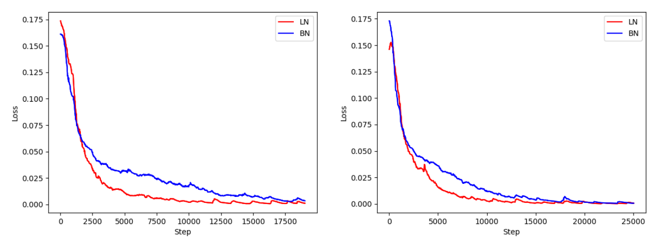
[1] Results on CEDAR with/without removal background.

C. Cross Domain Comparisons with Existing Methods

The comparison results in cross-domain setting are presented in Table IV. Our method greatly exceeds all the counterpart methods for cross-domain signature verification. An obvious phenomenon is that the existing methods have good performance between BHSig-H and BHSig-B, while their performance between BHSig and CEDAR is very poor. This is because the signature images in BHSig and CEDAR are quite different. It is worth noting that although our method does not overwhelm the counterpart methods on CEDAR and BHSig-B in single-dataset setting, it achieves the best performance in cross-domain setting. According to these results, on the one hand, our method does improve the generalization of extracted signature features; on the other hand, it is very challenging to simultaneously enhance both generalization and discrimination.

TABLE VI
ABLATION STUDY RESULTS (%) OF OUR METHOD

Method	CEDAR		CEDAR-> BHSig-B		BHSig-B		BHSig-B-> CEDAE	
	AUC	EER	AUC	EER	AUC	EER	AUC	EER
Baseline	92.92	15.56	84.81	23.63	96.13	11.51	79.31	28.57
+IN	94.76	12.29	87.29	21.34	97.65	8.66	84.07	23.99
+IN+BN	93.41	14.43	91.59	16.72	97.67	8.25	85.90	22.26
+IN+LN	**96.75**	**9.21**	86.18	22.24	**98.78**	**5.72**	86.76	21.10
+IN+LN+ TCI (Ours)	**96.75**	**9.21**	**93.31**	**15.07**	**98.78**	**5.72**	**88.08**	**19.66**



(a) CEDAR (b) BHSig-B

Fig. 3. Convergence curves of the loss when training LN and BN on (a) CEDAR and (b) BHSig-B datasets.

D. Single Domain Comparisons with Existing Methods

We further compare our method with more existing methods for single-domain setting, and the results are shown in Table V. We list the results of both *global* T and *user* T for our method. On CEDAR, the SigNet and DeepHSV methods implement an EER of 0%, IDN implements an EER of 3.62%, and SDINet implements an EER of 1.75%, This is due to the uneven gray level of the true and false signature background when removing the background without OTSU [7], [25]. Our method obtains 9.21% and 3.00% EER with/without OTSU. The relatively lower performance of our method on CEDAR in single-dataset setting is probably because our method does not focus on background noise but on generalized signature features. Moreover, our method is comparable to the state-of-the-arts on BHSig-B, and advances the state-of-the-art performance on BHSig-H, proving that our method does improve the generalisation with acceptable loss of discrimination.

TABLE VII
RESULTS (%) OF ADDING IN ON DIFFERENT LAYERS

Layer	CEDAR		CEDAR-> BHSig-B		BHSig-B		BHSig-B-> CEDAR	
	AUC	EER	AUC	EER	AUC	EER	AUC	EER
1	91.44	15.81	91.91	16.43	96.26	10.25	89.70	17.99
2	94.31	13.28	87.42	21.19	97.74	8.35	**89.71**	**18.25**
3	94.66	12.45	**93.82**	**14.09**	97.73	8.20	87.53	20.39
4	**96.75**	**9.21**	93.31	15.07	**98.78**	**5.72**	88.08	19.66
5	93.04	14.75	82.18	26.05	96.22	11.00	87.53	20.91
1,2	92.06	15.13	90.26	18.14	96.86	9.17	86.98	20.62
1,3	95.59	11.74	73.57	32.91	97.41	8.70	87.26	20.34
1,4	91.51	15.82	92.05	16.18	97.44	8.54	89.48	18.34
2,3	94.29	13.04	86.62	22.25	98.38	6.62	89.68	18.45
2,4	95.10	11.40	89.28	19.42	96.58	10.46	88.75	19.97
3,4	95.13	11.90	91.45	17.09	95.96	10.50	88.26	19.70
1,2,3	90.68	17.22	88.94	19.63	96.87	9.52	85.58	22.25
2,3,4	95.37	10.89	89.03	19.47	98.29	7.33	88.92	19.12

E. Ablation Study

To analyze the effectiveness of individual modules, we conduct experiments in single-domain and cross-domain settings on the BHSig-H and CEDAR datasets. The results are shown in Table VI. By using the IN module, the performance of the target domain is improved compared with that of the source domain, indicating that IN can improve the generalization of the model. When BN or LN modules are used, LN has better results on the single-dataset setting. Fig. 3 shows the change of the loss values of BN and LN when training the model on CEDAR and BHSig-B. It can be seen that the loss value of LN converges faster, suggesting that LN is more able to preserve discriminative features and is suitable for signature verification tasks. Besides, TCI can effectively improve performance on the cross-domain setting, further confirming the effectiveness of our method.

F. Instance Normalization Position Selection

According to previous studies, the introduction of the IN into the shallow layer of the CNN can effectively improve the model generalization, but there is no clear definition for the shallow layer. Hence, we compare the performance when adding IN into different layers for both single domain and cross-domain settings. The results are shown in Table VII. As can be seen, IN can effectively improve model generalization in layers 1-4, which verifies that inserting a shallow layer of IN can improve model generalization. Besides, the performance of using 2-3 layers and 1 layer are comparable, which indicates that using more layers is not necessary. We therefore choose the position of the 4th layer to implement our method.

V. CONCLUSIONS

In this paper, we introduce a generalization feature representation learning framework for offline signature verification. First, the Instance Normalization (IN) is introduced into the CNN backbone to improve the generalization of the model. Second, the Inference Layer Normalization Neck (ILNNeck) module is proposed based on the observation that LN. The LN in this module is more suitable for signature verification than BN, and the trick TCI can effectively improve the generalization of the model. Experimental results on our self-collected dataset and public datasets show that our method performs well in both single-domain and cross-domain settings, indicating that our model does make a good balance between generalization and discrimination.

ACKNOWLEDGEMENTS

This work is partially supported by the National Natural Science Foundation of China (No. 62066042, 62166038, 61971005).

REFERENCES

[1] R. Kumar, J. Sharma, and B. Chanda, "Writer-independent off-line signature verification using surroundedness feature," *Pattern recognition letters*, vol. 33, no. 3, pp. 301–308, 2012.
[2] A. Dutta, U. Pal, and J. Lladós, "Compact correlated features for writer independent signature verification," in *2016 23rd international conference on pattern recognition (ICPR)*. IEEE, 2016, pp. 3422–3427.
[3] S. Dey, A. Dutta, J. I. Toledo, S. K. Ghosh, J. Lladós, and U. Pal, "Signet: Convolutional siamese network for writer independent offline signature verification," *arXiv preprint arXiv:1707.02131*, 2017.
[4] C. Li, F. Lin, Z. Wang, G. Yu, L. Yuan, and H. Wang, "Deephsv: User-independent offline signature verification using two-channel cnn," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2019, pp. 166–171.
[5] P. Wei, H. Li, and P. Hu, "Inverse discriminative networks for handwritten signature verification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5764–5772.
[6] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Meta-learning for fast classifier adaptation to new users of signature verification systems," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1735–1745, 2019.
[7] L. Liu, L. Huang, F. Yin, and Y. Chen, "Offline signature verification using a region based deep metric learning network," *Pattern Recognition*, vol. 118, p. 108009, 2021.
[8] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *arXiv preprint arXiv:1703.07737*, 2017.
[9] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6924–6932.
[10] V. Dumoulin, J. Shlens, and M. Kudlur, "A learned representation for artistic style," in *ICLR*, 2017.
[11] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision*. Springer, 2016, pp. 694–711.
[12] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang, "Learning generalisable omni-scale representations for person re-identification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.
[13] M. K. Kalera, S. Srihari, and A. Xu, "Offline signature verification and identification using distance statistics," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 18, no. 07, pp. 1339–1360, 2004.
[14] J. Ortega-Garcia, J. Fierrez-Aguilar, D. Simon, J. Gonzalez, M. Faundez-Zanuy, V. Espinosa, A. Satue, I. Hernaez, J.-J. Igarza, C. Vivaracho *et al.*, "Mcyt baseline corpus: a bimodal biometric database," *IEE Proceedings-Vision, Image and Signal Processing*, vol. 150, no. 6, pp. 395–401, 2003.
[15] M. A. Ferrer, M. Diaz-Cabrera, and A. Morales, "Synthetic off-line signature image generation," in *2013 international conference on biometrics (ICB)*. IEEE, 2013, pp. 1–7.

[16] S. Pal, A. Alaei, U. Pal, and M. Blumenstein, "Performance of an off-line signature verification method based on texture features on a large indic-script signature dataset," in *2016 12th IAPR workshop on document analysis systems (DAS)*. IEEE, 2016, pp. 72–77.

[17] G. Alvarez, B. Sheffer, and M. Bryant, "Offline signature verification with convolutional neural networks," *Technical report, Stanford University*, 2016.

[18] H. Baltzakis and N. Papamarkos, "A new signature verification technique based on a two-stage neural network classifier," *Engineering applications of Artificial intelligence*, vol. 14, no. 1, pp. 95–103, 2001.

[19] E. J. Justino, A. El Yacoubi, F. Bortolozzi, and R. Sabourin, "An off-line signature verification system using hmm and graphometric features," in *Proc. of the 4th international workshop on document analysis systems*. Citeseer, 2000, pp. 211–222.

[20] A. El-Yacoubi, E. Justino, R. Sabourin, and F. Bortolozzi, "Off-line signature verification using hmms and cross-validation," in *Neural Networks for Signal Processing X. Proceedings of the 2000 IEEE Signal Processing Society Workshop (Cat. No. 00TH8501)*, vol. 2. IEEE, 2000, pp. 859–868.

[21] M. B. Yılmaz and B. Yanıkoğlu, "Score level fusion of classifiers in off-line signature verification," *Information Fusion*, vol. 32, pp. 109–119, 2016.

[22] M. B. Yilmaz, B. Yanikoglu, C. Tirkaz, and A. Kholmatov, "Offline signature verification using classifier combination of hog and lbp features," in *2011 international joint conference on Biometrics (IJCB)*. IEEE, 2011, pp. 1–7.

[23] Y. Serdouk, H. Nemmour, and Y. Chibani, "Combination of oc-lbp and longest run features for off-line signature verification," in *2014 Tenth International Conference on Signal-Image Technology and Internet-Based Systems*. IEEE, 2014, pp. 84–88.

[24] J. Ruiz-del Solar, C. Devia, P. Loncomilla, and F. Concha, "Offline signature verification using local interest points and descriptors," in *Iberoamerican congress on pattern recognition*. Springer, 2008, pp. 22–29.

[25] X. Lu, L. Huang, and F. Yin, "Cut and compare: End-to-end offline signature verification network," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 3589–3596.

[26] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Writer-independent feature learning for offline signature verification using deep convolutional neural networks," in *2016 international joint conference on neural networks (IJCNN)*. IEEE, 2016, pp. 2576–2583.

[27] M. Abuzar Shaikh, T. Duan, M. Chauhan, and S. Srihari, "Attention based writer independent handwriting verification," *arXiv e-prints*, pp. arXiv–2009, 2020.

[28] H. Li, P. Wei, and P. Hu, "Static-dynamic interaction networks for offline signature verification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 3, 2021, pp. 1893–1901.

[29] A. Das, M. A. Ferrer, U. Pal, S. Pal, M. Diaz, and M. Blumenstein, "Multi-script versus single-script scenarios in automatic off-line signature verification," *IET biometrics*, vol. 5, no. 4, pp. 305–313, 2016.

[30] K. Zhou, Y. Yang, T. Hospedales, and T. Xiang, "Learning to generate novel domains for domain generalization," in *European conference on computer vision*. Springer, 2020, pp. 561–578.

[31] M. Ghifary, W. B. Kleijn, M. Zhang, and D. Balduzzi, "Domain generalization for object recognition with multi-task autoencoders," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2551–2559.

[32] H. Nam and H.-E. Kim, "Batch-instance normalization for adaptively style-invariant neural networks," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[33] X. Pan, P. Luo, J. Shi, and X. Tang, "Two at once: Enhancing learning and generalization capacities via ibn-net," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 464–479.

[34] H. Luo, Y. Gu, X. Liao, S. Lai, and W. Jiang, "Bag of tricks and a strong baseline for deep person re-identification," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2019, pp. 0–0.

[35] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.