ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-4485-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICASSP48485.2024.10446790

PROTOTYPE CALIBRATION WITH SYNTHESIZED SAMPLES FOR ZERO-SHOT CHINESE CHARACTER RECOGNITION

Xiang Ao[1], *Xiao-Hui Li*[1], *Xu-Yao Zhang*[1], *Cheng-Lin Liu*[1,2]*

[1]MAIS, Institute of Automation of Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences

ABSTRACT

Zero-shot Chinese character recognition aims to recognize unseen characters that have never appeared in training. Recently, many methods learn a cross-modal alignment between character samples and auxiliary semantic data like glyph templates in training, and directly employ it to recognize unseen characters by retrieving the class with most similar semantics. However, these approaches suffer from the domain shift problem, which means that the learned alignment shows a deviation on unseen characters. To alleviate this problem, we generate unseen character samples to calibrate the shifted prototypes in the feature space. Specifically, we train a cross-modal prototype classifier and a generator conditioned on glyph templates, then use the generator to synthesize unseen character samples to calibrate the prototypes of the classifier. The calibration process does not require any extra training. Experiments on a handwritten dataset and a nature scene dataset show the superiority of our method and the effectiveness of prototype calibration.

Index Terms— zero-shot, Chinese character recognition, prototype calibration, sample generation

1. INTRODUCTION

To create a recognizer that can identify all possible Chinese characters, we usually need a large number of training samples covering over sixty thousands of character categories, which is often impractical and expensive. Besides, the closed recognizer is incapable of identifying newly found or created characters. Instead, zero-shot Chinese character recognition (ZSCCR) [1] aims to recognize samples of unseen characters never appeared in training with auxiliary semantic information describing these characters, like stroke sequence, radical components or printed glyphs. Thus, the key of ZSCCR is to build a cross-modal alignment between character samples and auxiliary semantic information at the class level after training on seen characters. Afterwards, the alignment is applied on unseen Chinese characters and zero-shot classification can be realized by finding the character with best matching auxiliary information.

According to different forms of auxiliary information, the methods in ZSCCR can be roughly divided into three groups: radical-based, stroke-based, and template-based. Radical-based methods [1–4] analyze radical components and spatial structures among the components to recognize unseen characters. These methods can give an intuitive explanation about the final classification decision but their performance are usually limited. As for stroke-based works [5], they focus on the basic semantic units of a character and extract the stroke sequence of an unseen character. Template-based approaches [6–8] treat the printed character images as the templates for handwritten samples or nature scene instances. These images can be obtained from any font file on the Internet, which is convenient. These approaches can achieve higher classification accuracy due to the smaller modality gap.

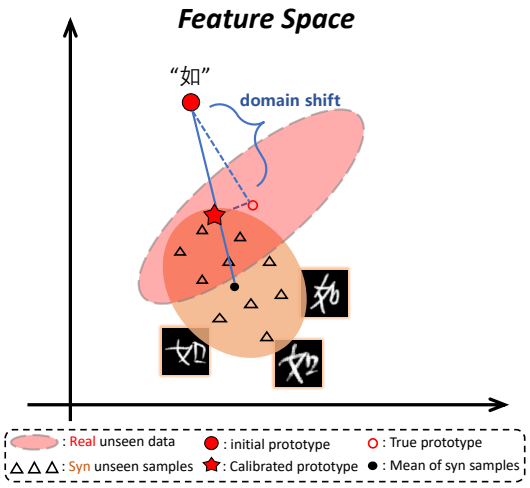


Fig. 1. There is a shift between the initial prototypes and the real distribution of unseen characters in the feature space. We alleviate the problem by prototype calibration with synthesized unseen character samples, then the adjusted prototypes will be closer to the real distribution of unseen categories.

However, all above-mentioned methods suffer from the problem of domain shift [9] which is caused by the fact that the alignment between samples and auxiliary semantic data on seen categories is not always consistent with that on unseen categories. As shown in Fig. 1, a shift issue usually happens if we directly transfer the alignment learned on seen characters to unseen characters since the true prototype is far from the initial one. This domain shift issue fundamentally limits the performance of above methods.

To alleviate the problem, besides training a classifier that regards printed images as the character prototype, we also train a generator conditioned on printed images to synthesize unseen character samples, then these samples are employed to calibrate the shifted printed prototype of the classifier, as shown in Fig. 1. And the calibration strategy is simply the interpolation between the initial prototype and the mean of synthesized samples in the feature space. The reason why not directly retraining a classifier with synthesized samples is the gap between the synthesized distribution and the real distribution on unseen characters, as the generator is trained on seen characters, which is shown in Fig. 1. Extensive experiments shows

*Corresponding author.

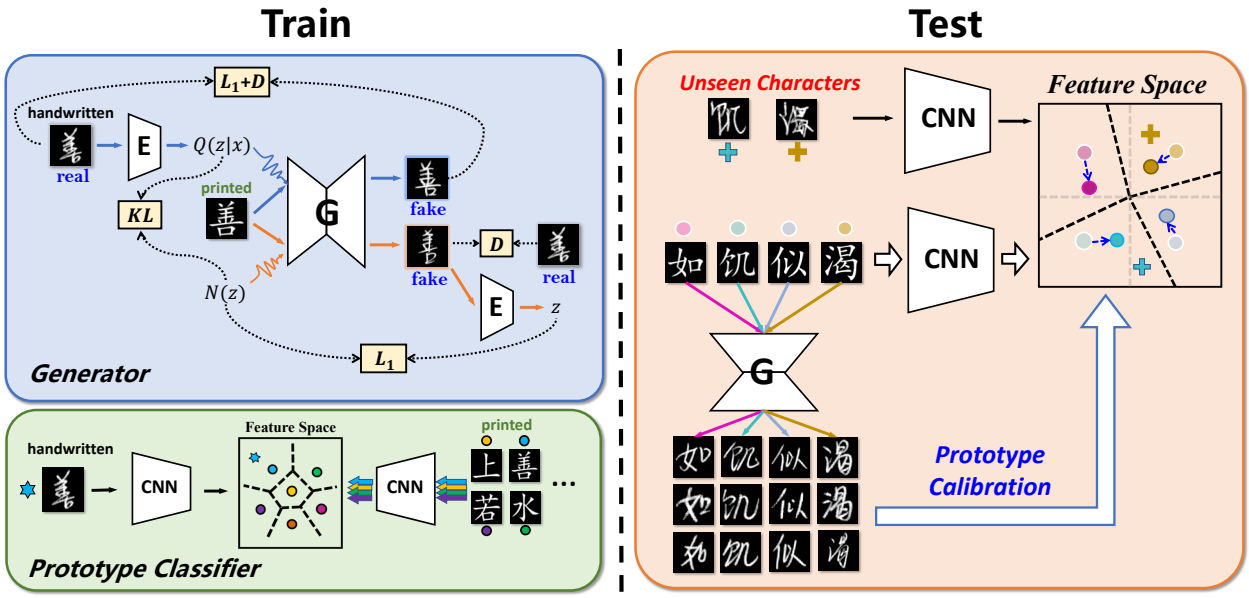


Fig. 2. The framework of our method. It includes a conditional GAN based generator and a prototype classifier. The former learns to create character samples conditioned on printed glyph templates, and the latter learns to map the glyph template to the prototype of samples in the feature space. During test stage, synthesized unseen character samples by the generator are used to calibrate the printed prototype in the feature space.

that our method can achieve the state-of-the-art results and prototype calibration with synthesized samples brings stable performance improvement across different sizes of training class sets. Quantitative analysis in the feature space validates that the degree of domain shift is significantly reduced by prototype calibration.

2. METHOD

The framework of our method in Fig. 2 contains a generator and a prototype classifier, and the generator is used to assist the classifier via prototype calibration during inference.

Problem formulation. We denote the training set as $\mathcal{D}^S = \{(x, y) \mid y \in \mathcal{Y}^S\}$, where x represents a handwritten sample and y is its label from seen character categories $\mathcal{Y}^S$. The auxiliary data are in the form of printed glyph templates $\mathcal{A}^S = \{a_y | y \in \mathcal{Y}^S\}$ with one image per character category. After training on $\mathcal{D}^S$ and $\mathcal{A}^S$ from seen categories $\mathcal{Y}^S$, we aim to get a zero-shot classifier $f : \mathcal{X} \rightarrow \mathcal{Y}^U$, where $\mathcal{Y}^U \cap \mathcal{Y}^S = \emptyset$. Note that the auxiliary data of unseen characters $\mathcal{A}^U = \{a_y | y \in \mathcal{Y}^U\}$ are available in test stage.

2.1. Cross-Modal Prototype Classifier

The classifier follows a simple idea that the glyph templates are assumed as the prototypes of handwritten or scene character samples in the feature space, which is based on the work [6]. And the classification decision is realized by the nearest prototype rule.

The character samples and the printed glyphs are encoded into a shared feature space, and we use the convolutional neural networks (CNN) [10] as the encoder. We denote the encoder of character samples as $\phi(\cdot)$ and the encoder of glyph templates as $\pi(\cdot)$. Then we can obtain the prototype:

$$p_y = \pi(a_y) \tag{1}$$

for the character category y, and the classifier is trained with a cross-entropy loss:

$$L_C = -\log p(y \mid x) = -\log \frac{e^{-\beta \|\phi(x) - p_y\|_2}}{\sum_{i \in \mathcal{Y}^S} e^{-\beta \|\phi(x) - p_i\|_2}}, \tag{2}$$

where β is a learnable parameter and $\|\cdot\|_2$ represents the L2 norm. After training, the alignment between character samples and auxiliary glyph templates has been learned on seen characters.

2.2. Conditional GAN Based Generator

The generator aims to create character samples of accurate contents and diverse styles given one printed image a_y. We formulate the generation process as:

$$\widetilde{x} = G(a_y, z) \tag{3}$$

where $\widetilde{x}$ is the synthesized sample, G stands for the generator, and $z \sim N(0, 1)$ indicates the Gaussian noise. After being trained on seen character data $\mathcal{D}^S$, the generator will indirectly models the conditional distribution $p(x \mid y)$.

Our generator is inspired by BicycleGAN [11], as shown in Fig. 2. It combines a GAN [12] and a variation encoder (VAE) [13] with shared decoder and generator to improve the generation quality.

The final objective loss of the generator is:

$$\begin{aligned} L_G = & L_{GAN}^{VAE}(G, D, E) + \lambda L_1^{VAE}(G, E) + \\ & \lambda_{KL} L_{KL}(E) + L_{GAN}(G, D) + \lambda_{\text{latent}} L_1^{\text{latent}}(G, E), \end{aligned} \tag{4}$$

where G is the generator/decoder, D represents the discriminator, and E indicates the encoder. $L_1^{VAE}(G, E)$ and $L_1^{\text{latent}}(G, E)$ reflect the two cycle constraints to enhance the performance of the generator. Since the design of the generator is not the focus of this work, detailed explanations can be found in [11].

2.3. Prototype Calibration

We aim to use synthesized samples to calibrate the printed prototypes of unseen characters. First, we have initial prototypes mapped from printed glyph templates for each unseen characters:

$$p_i = \pi(a_i), i \in \mathcal{Y}^U. \tag{5}$$

We also have synthesized samples of each unseen categories:

$$\{\widetilde{x}_{i,k} = G(a_i, z_k) \mid i \in \mathcal{Y}^U, k = 1, 2, ..., K\}, \tag{6}$$

where z_k is Gaussian sampling noise.

We further assume that the real distribution lies between the printed prototype and the synthetic distribution, as shown in Fig. 1. Then we choose a simple strategy that the interpolation between the printed prototype and the mean of synthesized data serves as the calibrated prototype:

$$p_i' = (1-\alpha)p_i + \alpha\frac{1}{K}\sum_{k=1}^{K}\phi(\widetilde{x}_{i,k}), \tag{7}$$

Where K and α are two hyperparameters. The adjusted prototype p_i' would be closer to the real distribution of the unseen character.

2.4. Zero-Shot Inference

After prototype calibration, the final decision is obtained by the nearest prototype rule:

$$f(x) = \arg\min_{i \in \mathcal{Y}^U} \|\phi(x) - p_i'\|_2 \tag{8}$$

where $f(x)$ is the class prediction of test sample x and $\|\cdot\|_2$ indicates the Euclidean distance.

3. EXPERIMENTS

3.1. Experimental Setup

Datasets. We conduct experiments on both a handwritten Chinese character dataset HWDB [14, 15] and a natural scene Chinese character dataset CTW [16]. For HWDB, we take five different sizes of seen character categories, i.e. 500, 1000, 1500, 2000, 2755 in experiments, and the number of unseen characters are fixed to 1000. For CTW, the size of seen characters are set to 500, 1000, 1500, 2000 and the number of unseen characters are fixed to 500. As for the auxiliary glyph templates, we choose the SimKai font with a resolution of 64×64 pixels.

Implementation Details. The input images are resized to 64×64 pixels. The backbone of the classifier is the same as [4]. Moreover, the generator G is resnet-6blocks like [17]. The encoder E and the discriminator D are consistent with [11]. The prototype classifier is trained with learning rate=0.005 and batch size=256 with the Adam optimizer. As for the generator, learning rate=0.0005, batch size=256 for HWDB and batch size=32 for CTW with the same optimizer. In prototype calibration, K is 30 for HWDB and 100 for CTW, and α is set to 0.6 for HWDB and 0.2 for CTW, respectively.

3.2. Zero-Shot Handwritten Character Recognition

We compare to the state-of-the-art in zero-shot handwritten character recognition. The results under different sizes of training characters are reported in Table 1. We list 6 methods for comparison, where DenseRAN [1], Few-shotRAN [3] and HDE [4] are radical-based, and CMPL [6], OSOCR [18], OpenCCD [8] are template-based.

Table 1. Comparing to the state-of-the-art on the HWDB dataset

Method	Accuracy(%)				
	#characters in training set				
	500	1000	1500	2000	2755
DenseRAN [1]	1.7	8.4	14.7	19.5	30.7
Few-shotRAN [3]	33.6	41.5	63.8	70.6	77.2
HDE [4]	33.7	53.9	66.3	73.4	81.0
OSOCR [18]	46.7	72.2	79.8	84.3	-
CMPL [6]	83.6	88.4	90.5	91.9	93.9
OpenCCD [8]	90.9	94.1	94.6	95.6	-
Ours	**94.4**	**95.8**	**96.4**	**96.7**	**97.2**

Table 2. Comparing to the state-of-the-art on the CTW dataset

Method	Accuracy(%)			
	#characters in training set			
	500	1000	1500	2000
DenseRAN [1]	0.1	1.5	5.0	10.1
Few-shotRAN [3]	2.4	10.5	16.6	22.0
HDE [4]	23.5	38.5	44.2	49.8
OSOCR [18]	27.9	48.2	58.6	63.8
OpenCCD [8]	58.2	68.6	74.5	77.2
Ours	**66.4**	**72.1**	**76.5**	**78.3**

From Table 1 we can observe that our method outperforms previous state-of-the-art approaches. Compared with the second-best results, our method achieves gains of 3.5%, 1.7%, 1.8% and 1.1% in terms of accuracy on 500, 1000, 1500 and 2000 seen characters, respectively. The gain in accuracy becomes more significant when the size of seen characters is smaller, which exhibits the advantage of our method with extremely limited data.

3.3. Zero-Shot Scene Character Recognition

We compare our method against several approaches on CTW dataset, as reported in Table 2. The results indicate that our method achieves the best zero-shot recognition performance across 4 different sizes of characters in training set. Specially, it surpasses the previous state-of-the-art model OpenCCD [8] by +8.2% on 500, +3.5% on 1000, +2% on 1500, +1.1% on 2000 in terms of accuracy. Besides, considering that the CTW dataset is much more challenging with extremely class imbalance and many blurry or occluded samples, the experimental results in Table 2 are poorer than that in Table 1.Yet our method shows greater improvements than that on HWDB, which demonstrates the great potential of our approach.

3.4. Visualizations of Synthesized Unseen Character Samples

The generated samples in Fig. 3 show satisfactory performance on HWDB and promising results on CTW in terms of fidelity and diversity. The synthesized handwritten samples of unseen characters look similar to the real ones. Besides exhibiting accurate content, most generated instances show diverse written styles. As for the generated scene samples of unseen characters, many of them accurately reflect the character contents, show diverse shapes and colors, and

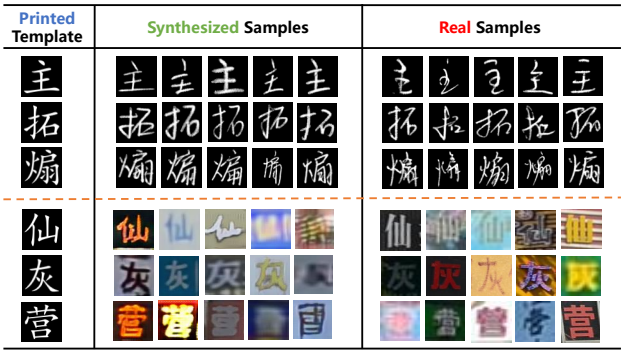


Fig. 3. Comparing the synthesized handwritten/scene samples with the real samples on unseen character categories, after the generator is trained on 1K characters of HWDB dataset and CTW dataset respectively.

Table 3. Impact of prototype calibration on HWDB dataset and CTW dataset. "PC" is short for prototype calibration

Dataset	PC	Accuracy(%)				
		#characters in training set				
		500	1000	1500	2000	2755
HWDB	No	92.1	93.7	94.6	94.7	94.8
	Yes	94.4	95.8	96.4	96.7	97.2
CTW	No	64.8	70.6	75.4	77.6	-
	Yes	66.4	72.1	76.5	78.1	-

exhibit a certain degree of blurriness like real data. However, there are also some poorly synthesized samples with missed local components, wrongly placed strokes and illegible contents.

3.5. Ablation Study

Prototype calibration can improve the zero-shot recognition performance. The results on HWDB and CTW are shown in Table 3. Note that our method with no prototype calibration equals the basic prototype classifier. The prototype calibration leads to accuracy improvement in both datasets. The gain by "PC" is over 2% on HWDB across different sizes of training classes. We think that prototype calibration reduces the degree of domain shift, then a test sample is more likely to be assigned to the prototype of the correct category.

Prototype calibration can alleviate the degree of domain shift through quantitative analysis. We introduce a quantitative index, namely the score of resistance on domain shift (ScoreRDS) in [19], which is calculated by

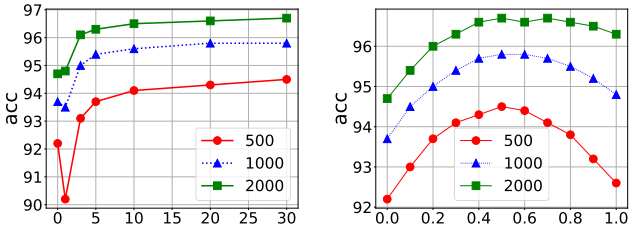
$$ScoreRDS = \frac{\sum_{i \in \mathcal{Y}^U} \|\mu_i - p_i\|_2 / |\mathcal{Y}^U|}{\sum_{j \in \mathcal{Y}^S} \|\mu_j - p_j\|_2 / |\mathcal{Y}^S|}, \quad (9)$$

where μ_i is the mean of real samples of class i in the feature space and p_i is the prototype of class i obtained by us. A smaller ScoreRDS suggests that the degree of domain shift gets lower. In Table 4, with prototype calibration, the value of ScoreRDS decreases from 1.01 to around 0.6 on HWDB and from 1.02 to around 0.9 on CTW, which suggests that the domain shift problem is alleviated by prototype calibration on both datasets.

Hyperparameter analysis of K and α. In Fig. 4(a), we can see that when K is too small, the zero-shot recognition accuracy

Table 4. The degree of domain shift before and after prototype calibration. It is quantified by the metric ScoreRDS. "PC" represents prototype calibration

Dataset	PC	ScoreRDS			
		#characters in training set			
		500	1000	1500	2000
HWDB	No	1.02	1.01	1.01	1.01
	Yes	0.61	0.59	0.6	0.58
CTW	No	1.02	1.02	1.02	1.01
	Yes	0.88	0.89	0.89	0.88



(a) K (b) α

Fig. 4. Zero-shot recognition accuracies with different size K and interpolation coefficient α on HWDB dataset.

drops, since a few samples cannot represent the whole synthesized distribution. As K increases, the performance is improved, which can reflect the benefit of sample generation in zero-shot recognition. From Fig. 4(b), the best performance is achieved when α is set to 0.5 on HWDB, i.e. the calibrated prototype is located between the initial printed prototype and the mean of synthesize samples in the feature space. The results validate our hypothesis that in the feature space, the real distribution of unseen characters lies between the synthesized distribution and the printed prototypes.

4. CONCLUSIONS

To alleviate the domain shift problem, we propose a method that synthesizes samples to calibrate the unseen character prototypes of the classifier. Our method trains a conditional GAN-based generator and a prototype classifier simultaneously. Then the synthesized samples of unseen categories are used to adjust the corresponding prototypes via an interpolation strategy. Extensive experiments show the superiority of our method. And ablation studies demonstrate the effectiveness of prototype calibration. In the future, we will use a more powerful generator [20] to create unseen character samples with stronger fidelity and diversity, and explore more reasonable calibration strategies instead of simply interpolation, which we believe can further improve the performance of zero-shot character recognition.

5. ACKNOWLEDGMENTS

This work was supported in part by the National Key Research and Development Program under Grant 2018AAA0100400, in part by the National Natural Science Foundation of China (NSFC) under Grants 62076236 and 62222609.

6. REFERENCES

[1] Wenchao Wang, Jianshu Zhang, Jun Du, Zi-Rui Wang, and Yixing Zhu, "Denseran for offline handwritten chinese character recognition," in *Proceedings of the International Conference on Frontiers in Handwriting Recognition*. IEEE, 2018, pp. 104–109.

[2] Tie-Qiang Wang, Fei Yin, and Cheng-Lin Liu, "Radical-based chinese character recognition via multi-labeled learning of deep residual networks," in *Proceedings of the International Conference on Document Analysis and Recognition*. IEEE, 2017, pp. 579–584.

[3] Tianwei Wang, Zecheng Xie, Zhe Li, Lianwen Jin, and Xiangle Chen, "Radical aggregation network for few-shot offline handwritten chinese character recognition," *Pattern Recognition Letters*, vol. 125, pp. 821–827, 2019.

[4] Zhong Cao, Jiang Lu, Sen Cui, and Changshui Zhang, "Zero-shot handwritten chinese character recognition with hierarchical decomposition embedding," *Pattern Recognition*, p. 107488, 2020.

[5] Jingye Chen, Bin Li, and Xiangyang Xue, "Zero-shot chinese character recognition with stroke-level decomposition," in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, 8 2021, pp. 615–621.

[6] Xiang Ao, Xu-Yao Zhang, and Cheng-Lin Liu, "Cross-modal prototype learning for zero-shot handwritten character recognition," *Pattern Recognition*, vol. 131, pp. 108859, 2022.

[7] Zhiyuan Li, Qi Wu, Yi Xiao, Min Jin, and Huaxiang Lu, "Deep matching network for handwritten chinese character recognition," *Pattern Recognition*, p. 107471, 2020.

[8] Chang Liu, Chun Yang, and Xu-Cheng Yin, "Open-set text recognition via character-context decoupling," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4523–4532.

[9] Yanwei Fu, Timothy M Hospedales, Tao Xiang, and Shaogang Gong, "Transductive multi-view zero-shot learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 11, pp. 2332–2345, 2015.

[10] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[11] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman, "Toward multimodal image-to-image translation," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[13] Diederik P Kingma and Max Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[14] Cheng-Lin Liu, Fei Yin, Da-Han Wang, and Qiu-Feng Wang, "Casia online and offline chinese handwriting databases," in *Proceedings of the International Conference on Document Analysis and Recognition*. IEEE, 2011, pp. 37–41.

[15] Fei Yin, Qiu-Feng Wang, Xu-Yao Zhang, and Cheng-Lin Liu, "Icdar 2013 chinese handwriting recognition competition," in *Proceedings of the International Conference on Document Analysis and Recognition*. IEEE, 2013, pp. 1464–1470.

[16] Tai-Ling Yuan, Zhe Zhu, Kun Xu, Cheng-Jun Li, Tai-Jiang Mu, and Shi-Min Hu, "A large chinese text dataset in the wild," *Journal of Computer Science and Technology*, vol. 34, no. 3, pp. 509–521, 2019.

[17] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1125–1134.

[18] Chang Liu, Chun Yang, Hai-Bo Qin, Xiaobin Zhu, Cheng-Lin Liu, and Xu-Cheng Yin, "Towards open-set text recognition via label-to-prototype learning," *Pattern Recognition*, vol. 134, pp. 109109, 2023.

[19] Zhen Jia, Zhang Zhang, Liang Wang, Caifeng Shan, and Tieniu Tan, "Deep unbiased embedding transfer for zero-shot learning," *IEEE Transactions on Image Processing*, vol. 29, pp. 1958–1971, 2019.

[20] Jonathan Ho, Ajay Jain, and Pieter Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.