DOC: Text Recognition via Dual Adaptation and Clustering

Xue-Ying Ding, Xiao-Qian Liu, Xin Luo, Xin-Shun Xu

***Abstract*—More recently, unsupervised domain adaptation has been introduced to text image recognition tasks for serious domain shift problem, which can transfer knowledge from source domains to target ones. Moreover, in unsupervised domain adaptation for text recognition, there is no label information in the target domain to supervise the domain adaptation, especially at the character. Several existing methods regard a text image as a whole and perform only on global feature adaptation, neglecting local-level feature adaptation, i.e., characters. Others methods only focus their attention on word-level feature alignment while ignoring the categories of local-level characters. To address these issues, we propose a text recognition model via Dual adaptatiOn and Clustering, DOC for short. Regarding word-level, we construct a Global Discriminator for global feature adaptation to reduce text layout bias between source and target domains. Regarding character-level, we propose an Adaptive Feature Clustering (AFC) module, which can extract invariant character features through a local-level discriminator for adaptation. Moreover, it enhances the local-feature adaptation by a clustering scheme, which evaluates the feature adaptation by leveraging the knowledge from the source domain as much as possible. In this way, it can pay more attention to the differences in fine-grained characters. Extensive experiments on benchmark datasets demonstrate that our framework can achieve state-of-the-art performance. We have released the code.** [1]

***Index Terms*—Text Recognition; Unsupervised Domain Adaptation; Domain Shift; Clustering.**

I. Introduction

TEXT image recognition has become one of the most popular fields in computer vision because of its wide range of applications such as document understanding [1], [2], text error correction [3], textVQA [4], [5], multimedia retrieval [6], [7]. Deep learning methods have achieved much progress in the past several years on this task. However, deep learning methods usually require a large amount of labeled data for training. Moreover, it is much time-consuming and expensive to annotate training samples. Therefore, in many real applications, the amount of labeled data is far from enough to train a model with good generalization ability, making this task still more challenging.

This work was supported in part by the National Natural Science Foundation of China under Grant 62172256, 62202278 and 62202272, in part by Natural Science Foundation of Shandong Province under Grant ZR2019ZD06, and in part by the Major Program of the National Natural Science Foundation of China under Grant 61991411. (*Corresponding atuhor: Xin-Shun Xu.*)

X.-Y. Ding, X.-Q Liu, X. Luo are with the School of Software, Shandong University, Jinan, 250101, China. X.-S. Xu is with the School of Software, Shandong University, Jinan China, and Quan Cheng Laboratory, Jinan, China. (e-mail: 202015188@mail.sdu.edu.cn; jlrxqxq370322@126.com; luoxin.lxin@gmail.com; xuxinshun@sdu.edu.cn).

[1]https://github.com/dd0121/DOC

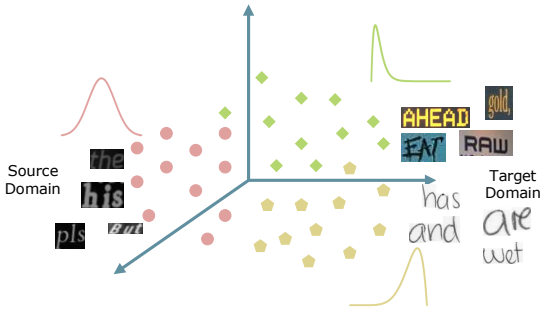


Fig. 1. The illustration of different feature distributions in source and target domains.

An intuitive idea is to transfer knowledge from other domains with labeled data to a new domain lack of training spaces. However, as shown in Figure 1, in some specific application scenarios, text data is much more complex, e.g. different fonts, different distributions and various backgrounds. To address these problems, recently, unsupervised domain adaptation (UDA) has been introduced into the task [8], [9], which aims to transfer knowledge from labeled source domain data to unlabeled target domain data. More importantly, it leverages unlabeled data to reduce the domain shift problem between the source domain and the target domain. Therefore, even the distributions of source domain data and target domain data are different, it can also obtain a distinctive and invariable feature through training, to help a model achieve good performance in a target domain. Generally speaking, most existing unsupervised domain methods minimize the domain shift by optimizing global-level features adaptation [10]. However, the label of a text image is a character sequence with variable length. Consequently, when adopting unsupervised domain adaptation in text image recognition task, the domain shift problem may occur at both global- and local-level. To consider this, more recently, Zhang et al. [8] proposed a method to make domain adaptation at both global- and local-level to void the problem of ignoring local features.

Although several UDA-based text image recognition methods have achieved promising performance, no label guides the target domain data in the training process. Therefore, these methods cannot ensure adapted global/local features are discernible enough to be transferred to characters. How to make use of the knowledge learned from the source domain data to supervise the recognition of the target domain becomes a key problem for UDA-based text image recognition models. To address these issues, we propose a novel UDA-based text recognition model with Dual adaptatiOn and Clustering, DOC

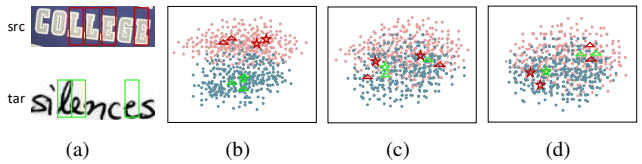


Fig. 2. The illustration of our motivation. Different colors represent features of different domains, and different shapes represent features of different categories.

for short. The motivation of adopting adaptation and clustering scheme could be illustrated in Figure 2. To solve the issue of dataset domain shift shown in subfigure (a) and (b), the dual adversarial adaption schemes perform global- and local-level domain adaptation for extracting domain invariant features, so as to improve the recognition accuracy of the model in the target domain. However, this may still lead to the features of different categories being too similar to distinguish. which is shown in subfigure (c). When adapting local-level features, we consider not only whether the feature comes from the source domain or target domain, but also which character category it belongs to. As shown in subfigure (d), we propose an Adaptive Feature Clustering module. In this module, filtered features from the source domain and target domain are clustered according to filtered source ground truth and target prediction. In this way, the adaptation can be optimized by the clustering result and the model can transcribe the local-level features of the source domain into the corresponding characters in the training process. Consequently, the model is able to make a more accurate prediction for the local-level features of the target domain.

In conclusion, the contributions of our work are as follows:

- We propose a novel UDA-based text image recognition framework, which performs both global- and local-level feature adaptation and further improves the recognition accuracy on the target domain by a clustering module.
- We propose a new Adaptive Feature Clustering (AFC) module at the local-level adaptation. The AFC module not only aligns the local features but also clusters the local features of the source domain and the target domain predicted to be the same characters. The recognition accuracy is further improved in this way.
- We conduct extensive experiments on benchmark datasets and make deep analyses of the experimental results. The results demonstrate that our proposed model generally outperforms some state-of-the-art methods for this task.

II. Related Work

A. Scene Text Recognition

Scene text recognition (STR) as a special field of optical character recognition (OCR), is a computer vision task to convert text images into text sequences [11]. With the gradual maturity of deep learning model, researchers have put forward many effective methods for solving different problems in this field, e.g. the geometry and chromatic distortion [12], [13], low resolution [14], [15], robust recognition [16], [17]. Specifically, for low-resolution scene text recognition, Chen et al. [15] established a text-focused super-resolution framework, which highlights the content of characters looking indistinguishable in low-resolution conditions. To reduce the dependence on low-quality visual information, Fang et al. [18] introduced linguistic knowledge into the task by fusing the text recognition model and language model and making the language model iteratively modify the prediction of the text recognition model. In addition, Li et al. [19] developed a dual relation module consists of a local visual branch and a long-range contextual branch for scene text recognition. To address attention drift problem in recognition, Wang et al. [20] designed a decoupled text decoder that makes the prediction by jointly using the feature map and attention map. In addition, to solve the vocabulary dependence of language-based text recognition methods, Yue et al. [17] introduced a position enhancement branch and a dynamic fusion module to mitigate the issue of misrecognition in contextless scenarios. Generally speaking, these methods rely on complex model structures and a large amount of training data is usually required. Therefore, in some real applications, they often suffer from the limitations of embedded devices on the scale of the model and the amount of available labeled data.

B. Unsupervised Domain Adaptation

Deep learning models usually require a large amount of training data. In addition, they also assume that training data and testing data have identical distributions [21]. However, in real applications, they may suffer from two problems that limit their performance. (1) There is no large amount of data for training. In some scenarios, we even have no labeled data. Some methods have been proposed to address this problem, which exploit existing datasets in other domains (source domains) and transfer the knowledge to the target domain. However, these methods inevitably face the second problem: (2) The distributions of the source and target domains may be different. Unsupervised domain adaption methods have recently been proposed to reduce the domain discrepancy problem. They usually contain a domain discriminator to distinguish the source and target domain and a feature extractor to learn domain-invariant representations by adversarial learning. For example, Yaros et al. [22] constructed a minimax loss, and made the discriminator promote the recognition in the source and target domain by using a gradient reversal layer. Eric et al. [23] designed a domain discriminator, and introduced a GAN-loss to migrate the feature mapping and the classifier from source domain to target domain. Hsu et al. [24] bridged the domain gap with an intermediate domain by feature alignment and image-level adaptation. To learn a model on target domain at a finer category level, Zhang et al. [25] introduced two-level domain confusion losses to learn the classifier for the target domain. Jing et al. [26] proposed an adversarial domain adaptation method that uses mixup to generate multiple features with different mixup ratios. Moreover, Deng et al. [27] developed a component called Informative Feature Disentanglement that enable informative feature refinement before the adaptation.

This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2023.3245404

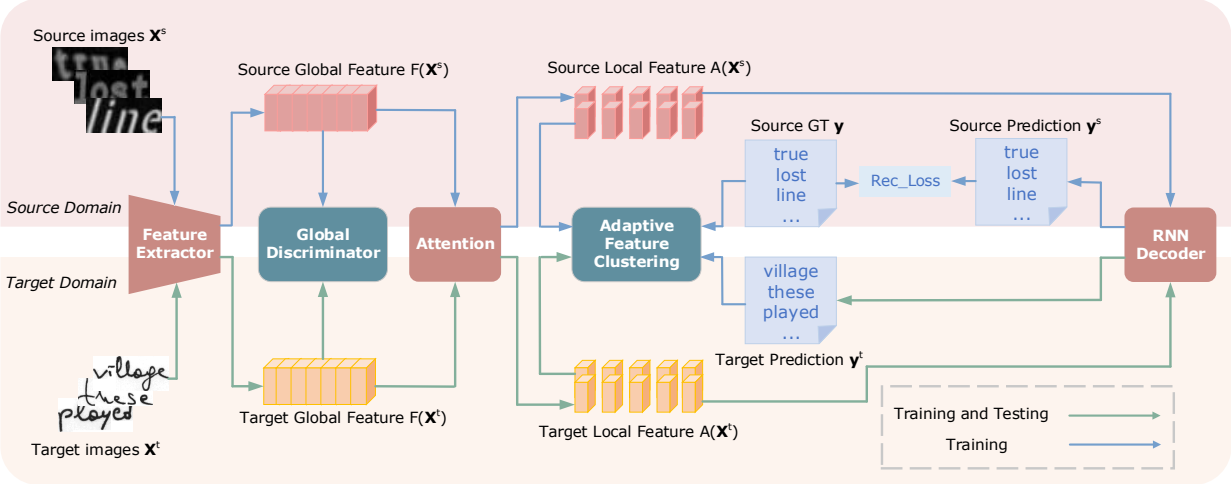


Fig. 3. The structure of the proposed model.

C. UDA-based Text Recognition

More recently, unsupervised domain adaptation has been introduced into the text image recognition task. For example, Zhan et al. [10] proposed a Geometry-Aware Domain Adaptation Network (GA-DAN) that reduces the domain shift in appearance and geometry spaces by generating adaptive text images from multiple spatial perspectives. However, it takes the text image as a whole for domain adaptation, ignoring the fact that the text is a variable-length sequence composed of fine-grained characters. For this purpose, Kang et al. [28] proposed an adaptable handwritten word recognizer with a discriminator incorporating a temporal pooling step to adapt to variable-length sequences. Zhang et al. [29] proposed a Sequence-to-Sequence Domain Adaptation Network (SSDAN) that focuses on aligning the distribution of the character-level feature space rather than a global coarse-grained alignment. However, only focusing on local-level domain adaptation may lead to insufficient knowledge transfer. More recently, Zhang et al. [8] proposed an Adversarial Sequence-to-Sequence Domain Adaptation framework (ASSDA), which uses the domain adaptation scheme at both global- and local-level. Relying on this mechanism, the model can not only align the word feature at the global-level, but also align the character feature at the local-level.

As mentioned above, although these UDA-based methods have achieved much promising results, they focus most of their attention on feature alignment and neglect the category of features themselves, which is also an important factor for the text image recognition task. We believe that the key for the model to learn local-level feature alignment is that the features can be translated into correct characters. Motivated by these issues, our model performs domain adaptation at both global- and local-levels. Moreover, it takes the source domain features as the guidance, aligns the local features, and ensures that the features are effective enough.

III. Our Method

In this section, we introduce the details of our proposed model. As shown in Figure 3, the framework consists of five modules, i.e., Feature Extractor, Global Discriminator, Attention, Adaptive Feature Clustering (AFC) and Decoder. Thereinto, Feature Extractor extracts global features of data from both source and target domains. Thereafter, Global Discriminator performs global-level visual feature adaptation between the source domain and target domain. The attention module extracts the local features of the positions related to the characters. The Adaptive Feature Clustering module performs local-level feature adaptation between the source domain and target domain by introducing a clustering scheme on the local features from the source domain and target domain. In this way, it can generate effective local features for prediction. Finally, the decoder with the RNN structure decodes the local features into character sequences. Our main contributions lie in the design of Global Discriminator, Adaptive Feature Clustering, and the design of loss functions in different modules. It is worth noting that, as mentioned previously, our model performs both global- and local-level feature adaptation. The global-level feature adaptation is performed in the Global Discriminator module; the local-level feature adaptation is performed in the Local Discriminator module which is incorporated in the AFC. The main notations used in the paper are summarized in Table I. Due to space limitations, we only elaborate on key modules and the designed loss functions in the following subsections. The implementation of other modules is briefly introduced in the experiments.

TABLE I
Main notations used in the paper

	Source domain	Target domain
Image	$\mathbf{X}^s$	$\mathbf{X}^t$
Global feature	$F(\mathbf{X}^s)$	$F(\mathbf{X}^t)$
Local feature	$A(\mathbf{X}^s)$	$A(\mathbf{X}^t)$
Prediction	$\mathbf{y}^s$	$\mathbf{y}^t$
Ground truth	$\mathbf{y}$	-

This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2023.3245404

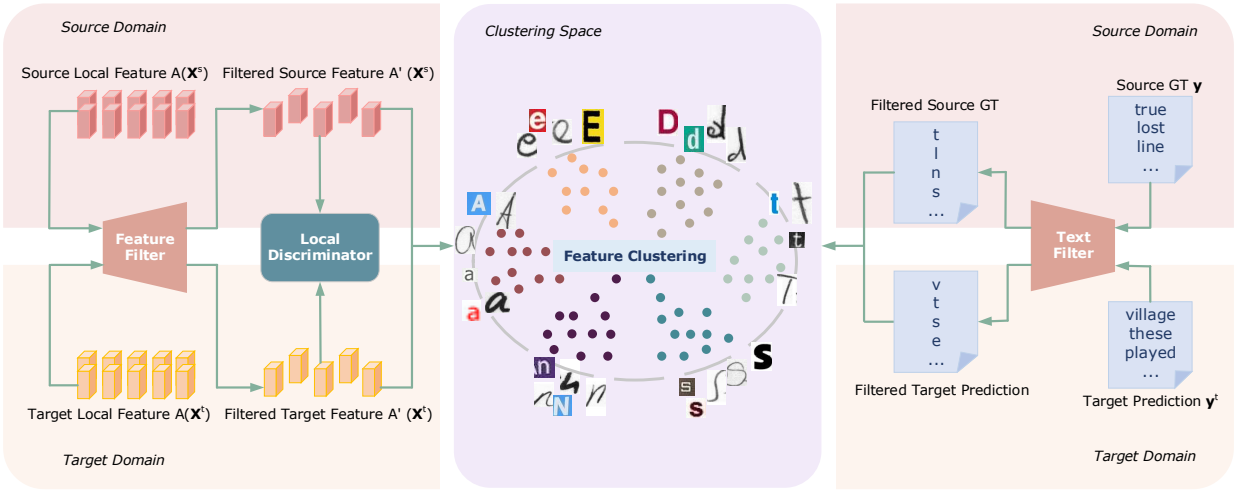


Fig. 4. Illustration of the Adaptive Feature Clustering module.

A. Global Discriminator

To extract the global visual features that are invariant to style, background, and geometric distortion, we construct a Global Discriminator D_g after the Feature Extractor. Specifically, the global-level features from the source domain and target domain are extracted and reshaped as follows:

$$\begin{aligned} F\left(\mathbf{X}^s\right) &= \left\{\mathbf{f}_1^s, \mathbf{f}_2^s, \ldots, \mathbf{f}_L^s\right\} \in (H \times W) \times C_F, \\ F\left(\mathbf{X}^t\right) &= \left\{\mathbf{f}_1^t, \mathbf{f}_2^t, \ldots, \mathbf{f}_L^t\right\} \in (H \times W) \times C_F, \end{aligned} \tag{1}$$

where H and W are the height and width of the feature map, respectively. The sequence length is $L = H \times W$ and C_F is the dimension of feature $\mathbf{f}_i$.

For the discriminator, we hope it can accurately distinguish the features from the source domain and the target domain. For text recognition, we hope that the extracted visual features can fool the discriminator so that it cannot distinguish whether the features come from the source domain or the target domain. For this purpose, we perform the invariable global visual features adaptation through adversarial learning. The adaptation loss $\mathcal{L}_{D_g}$ of the Global Discriminator D_g is defined as follows:

$$\mathcal{L}_{D_g} = -\frac{1}{N_s}\sum_{i=1}^{N_s}\log\left(1 - D_g\left(F\left(\mathbf{X}_i^s\right)\right)\right) - \frac{1}{N_t}\sum_{j=1}^{N_t}\log\left(D_g\left(F\left(\mathbf{X}_j^t\right)\right)\right), \tag{2}$$

where N_s and N_t are the number of samples in the source domain and the target domain, respectively.

B. Adaptive Feature Clustering

To decode visual features into character sequences, an attention module is further introduced between feature extractor and decoder. In specific, the attention module takes the global-level features $F\left(\mathbf{X}^s\right)$ and $F\left(\mathbf{X}^t\right)$ as input and outputs the local-level features $A\left(\mathbf{X}^s\right)$ and $A\left(\mathbf{X}^t\right)$, which are defined as follows:

$$\begin{aligned} A\left(\mathbf{X}^s\right) &= \left\{\mathbf{g}_1^s, \mathbf{g}_2^s, \ldots, \mathbf{g}_T^s\right\} \in T \times C_A, \\ A\left(\mathbf{X}^t\right) &= \left\{\mathbf{g}_1^t, \mathbf{g}_2^t, \ldots, \mathbf{g}_T^t\right\} \in T \times C_A, \end{aligned} \tag{3}$$

where $\mathbf{g}_i$ represents the glimpse vector generated in step i for maximizing the probability of predicting the ground truth character $\mathbf{y}_i$ from the attention weighted feature, T is the decoding time step, and C_A is the dimension of glimpse vector.

When domain adaptation is introduced into local-level features, we should guarantee that (1) The adapted local-level features can be transcribed into characters; (2) The adapted local-level features can be correctly classified. For this purpose, we design the Adaptive Feature Clustering module as shown in Figure 4, which is composed of four sub-modules or schemes, i.e., Feature Filter, Text Filter, Local Discriminator and Feature Clustering. Thereinto, local features are first filtered by the Feature Filter, which filters the features with low confidence being predicted as characters. Local Discriminator performs the local-level features adaptation. Feature Clustering determines the character categories to which the local-level adapted features belong. It is worth noting that we design a Text Filter in AFC. Its purpose is to filter the source ground truth and target prediction. The details of each component including its motivation are elaborated in the following subsections.

1) Feature Filter: As mentioned previously, Feature Filter filters out the features with low confidence being predicted as characters. if a local feature $\mathbf{g}_i$ predicts the character y_i with high confidence, it will be regarded as an effective local feature; otherwise, the feature will be filtered out. More specifically, inspired by ASSDA [8], the Feature Filter calculates mask $G\left(\mathbf{X}\right)$ by setting a confidence threshold p_c to obtain the filtered local features $A'\left(\mathbf{X}^s\right)$ and $A'\left(\mathbf{X}^t\right)$. The corresponding formulas are defined as follows:

$$\delta\left(\mathbf{g}_i\right) = \begin{cases} 1 & \text{if } p\left(y_i \mid y_{i-1}, \mathbf{g}_i\right) > p_c, \\ 0 & \text{if } p\left(y_i \mid y_{i-1}, \mathbf{g}_i\right) < p_c, \end{cases} \tag{4}$$

$$G\left(\mathbf{X}\right) = \left\{\delta\left(\mathbf{g}_1\right), \ldots, \delta\left(\mathbf{g}_T\right)\right\}, \tag{5}$$

$$\begin{aligned} A'\left(\mathbf{X}^s\right) &= A\left(\mathbf{X}^s\right) \otimes G\left(\mathbf{X}^s\right), \\ A'\left(\mathbf{X}^t\right) &= A\left(\mathbf{X}^t\right) \otimes G\left(\mathbf{X}^t\right), \end{aligned} \tag{6}$$

where $\otimes$ represents element-wise product operator.

2) Local Discriminator: To generate effective local-level features, we further incorporate a Local Discriminator, i.e. D_l, to adapt the local features. Similar to that of Global Discriminator, the adaptation loss $\mathcal{L}_{D_l}$ of D_l is defined as:

$$\mathcal{L}_{D_g} = -\frac{1}{N_s}\sum_{i=1}^{N_s}\log\left(1 - D_g\left(F\left(\mathbf{X}_i^s\right)\right)\right) - \frac{1}{N_t}\sum_{j=1}^{N_t}\log\left(D_g\left(F\left(\mathbf{X}_j^t\right)\right)\right), \tag{7}$$

where N_s' and N_t' are the number of filtered samples in the source domain and the target domain, respectively.

3) Text Filter: Under the condition that the target domain dataset has no label, we can only supervise the performance of the model in the target domain by leveraging the knowledge learned from the source domain as much as possible. For his purpose, we take the local features and labels of the source domain as supervised information. In specific, we further construct the Text Filter to filter the source ground truth $\mathbf{y}$ and target prediction $\mathbf{y}^t$ according to $G(\mathbf{X})$ to get a set of source character labels $\tilde{\mathbf{y}}^s$ and a set of target character predictions $\tilde{\mathbf{y}}^t$ corresponding to $G(\mathbf{X})$. Then, we take them as the categories of feature clustering, which further supervise the local feature adaptation. The corresponding representations are defined as follows:

$$\begin{aligned}\tilde{\mathbf{y}}^s &= \mathbf{y} \otimes G\left(\mathbf{X}^s\right),\\ \tilde{\mathbf{y}}^t &= \mathbf{y}^t \otimes G\left(\mathbf{X}^t\right),\end{aligned} \tag{8}$$

where $\otimes$ is the element-wise product operator.

4) Feature clustering: After we obtain the local-level features, we need to further assign them to correct character categories. However, as mentioned previously, the data in the target domain has no label. Therefore, we can only evaluate the feature adaptation by leveraging the knowledge from the source domain as much as possible. Ideally, the features with the same label should be close enough and the features with different labels to be far enough, whether from the source or target domain. Therefore, the features from the source domain should be correctly assigned to their categories; simultaneously, the features from the target domain closer to the features of the source domain should be assigned to the corresponding categories. Apparently, this task can be treated as a clustering task. In specific, we concatenate $A'\left(\mathbf{X}^s\right)$ and $A'\left(\mathbf{X}^t\right)$ as clustering features, and concatenate the filtered ground truth and predictions, i.e. $\tilde{\mathbf{y}}^s$ and $\tilde{\mathbf{y}}^t$ as clustering categories. In addition, we adopt a center loss [30] to evaluate the clustering. Correspondingly, the formulas are defined below:

$$\begin{aligned}\mathcal{X} &= A'\left(\mathbf{X}^s\right) \oplus A'\left(\mathbf{X}^t\right),\\ \mathcal{Y} &= \tilde{\mathbf{y}}^s \oplus \tilde{\mathbf{y}}^t,\end{aligned} \tag{9}$$

$$\mathcal{L}_{center} = \frac{1}{2}\sum_{i=1}^{m}\left\|\mathcal{X}_i - \boldsymbol{C}_{\mathcal{Y}_i}\right\|_2^2, \tag{10}$$

where $\oplus$ represents concatenation operation, m is the number of features to be clustered, $\mathcal{X}$ and $\mathcal{Y}$ are the concatenated clustering features and categories, respectively. $\mathcal{X}_i$ is the i-th feature belonging to the $\mathcal{Y}_i$-th class; $\boldsymbol{C}_{\mathcal{Y}_i}$ is the feature center of the $\mathcal{Y}_i$-th class which is updated during learning.

C. Training and Testing

During training, the local features $A\left(\mathbf{X}^s\right)$ and $A\left(\mathbf{X}^t\right)$ from the attention module are fed into the decoder to obtain the source prediction $\mathbf{y}^s$ and the target prediction $\mathbf{y}^t$. And a negative log-likelihood function is used to minimize the loss between $\mathbf{y}^s$ and source domain ground truth $\mathbf{y}$. In addition, $\mathbf{y}^t$ is used as the result of recognition. The recognition loss $\mathcal{L}_{\text{rec}}$ is formulated as:

$$\mathcal{L}_{rec} = -\frac{1}{N_s}\sum_{i=1}^{N_s}\log p\left(\mathbf{y}^s \mid \mathbf{y}\right). \tag{11}$$

To summarize, our framework adopts the adversarial-based unsupervised domain adaption to help the task of text recognition. It performs both global- and local-level feature adaption. Moreover, it incorporates a clustering scheme to supervise the local domain adaption. Therefore, jointly considering the modules in the framework, the overall optimization loss $\mathcal{L}$ is defined as follows:

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda_c\mathcal{L}_{center} - \lambda_g\mathcal{L}_{D_g} - \lambda_l\mathcal{L}_{D_l}, \tag{12}$$

where $\mathcal{L}_{rec}$, $\mathcal{L}_{center}$, $\mathcal{L}_{D_g}$ and $\mathcal{L}_{D_l}$ are the recognition loss, the clustering center loss, the global adaptation loss and the local adaptation loss, which are defined in Equation 11, Equation 10, Equation 2 and Equation 7, respectively. λ_c, λ_g and λ_l are all hyper-parameters.

During testing, a text image is fed into the Feature Extractor to get its global feature, which is input to the Attention module generating the local feature. Finally, the RNN-based Decoder generates the character sequence from the local feature.

IV. Experiments

In the experiments, we used two synthetic text datasets as source domain datasets, and test on four regular datasets, three irregular datasets, and a handwritten text dataset. The details of each dataset are introduced below.

SynthText (ST) [31] is a synthetic dataset with 800 thousand images and 6 million synthetic text instances cropped from Synth90k, which contains a large number of images with irregular text.

MJSynth (MJ) [32] is a synthetic dataset for STR, which contains 8.9 million images, and each image has a character level annotation.

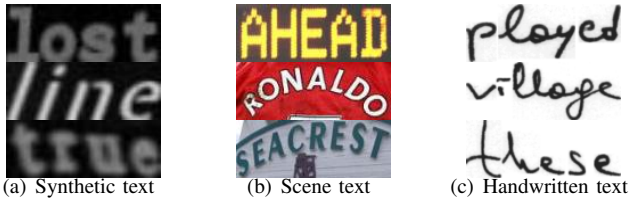
IIIT5K (IIIT) [33] is a regular dataset cropped from Google Image Search. It includes 2000 training images and 3000 test images.

Street View Text (SVT) [34] is a regular dataset collected from Google Street View images which contain 257 for training and 647 for testing.

ICDAR2003 (IC03) [35] is a regular dataset created for ICDAR2003 competition, which contains 867 pictures. We adopt the dataset of 860 images in which the images with labels of less than three characters are excluded.

ICDAR2013 (IC13) [36] is a regular dataset created for ICDAR2013 competition, which contains 848 images for training and 857 images for testing.

ICDAR2015 (IC15) [37] is an irregular dataset created for ICDAR2015 competition, which contains 4468 images for training and 1811 images for testing.



(a) Synthetic text (b) Scene text (c) Handwritten text

Fig. 5. The examples of synthetic text datasets, scene text datasets and handwritten text dataset.

SVT Perspective (SVTP) [38] is an irregular datasets which contains 645 images from Google Street View. It is composed of images with the same address but different perspectives collected from the SVT dataset, so it is specially used to evaluate the angle distortion of text recognition.

CUTE80 (CUTE) [39] is an irregular dataset collected from natural scenes which contains 288 clipped images. Most of the texts in the image are curved text images, therefore it is often used to test the performance of model recognition curved text.

IAM [40] is a handwritten texts dataset from 657 writers, including 115320 words. In our experiment, we use its test set as the target dataset which contains 20306 words.

The performance of our model and compared baselines are evaluated in terms of word accuracy, Word Error Rate (WER), and Character Error Rate (CER).

In order to more intuitively show the bias between synthetic text datastes, scene text datasets and handwritten datasets, we give several examples of diffierent types of datasets in Figure 5. From this figure, we find the these images all contain characters. However, the extracted features may be much different, which may result in domain shift.

A. Implementation Details

We adopted ResNet[41] as the backbone and a LSTM with hidden size of 256 as the RNN Decoder. We initialized the recognition model with the pre-trained model in [42], and then fine-tuned the whole model with source domain data and target domain data. All images are resized to 32×100 and then fed into Thin-Plate-Spline (TPS) network [43] for rectification before extracting features. 36 symbols are recognized, including 26 letters and 10 digits. We trained our model with AdaDelta [44] optimizer for 300K iterations with a batch size of 192. Both discriminator D_g and D_l contain two layers of fully connected networks with 100 and 2 dimensions (source or target), respectively. λ_c in Equation 12 is set to 0.0001. λ_g and λ_l are updated according to the following scheme as used in ASSDA [8] for fair comparison:

$$\lambda_g, \lambda_l = 1 - 2\left(1 - \frac{1}{1+e^{-10\sigma}}\right)^2 - \frac{1}{1+e^{-10\sigma}}, \qquad (13)$$

where $\sigma = \frac{n}{N}$, n is the current iterations, N represents the total iterations. This scheme can make the model focus on training the classification ability of the global and local domain discriminators in the early stage of the training process. Moreover, it also reduces the weight of the loss of the domain discriminator in the late stage of the training process, so as to focus on the training of the feature extractor and obtain the domain invariant feature that can fool the domain discriminators.

TABLE II
WER AND CER ON HANDWRITTEN DATASET IAM. [9]* MEANS THE UNSUPERVISED DOMAIN ADAPTATION RESULTS IN [9].

Method	Reference	WER	CER
[42]	ICCV 2019	54.30	28.41
SSDAN [29]	CVPR 2019	53.65	27.26
[9]*	ICCV 2021	**41.3**	-
ASSDA [8]	TIP 2021	43.78	**19.96**
DOC	-	**37.44**	**16.52**

TABLE III
ACCURACY ON REGULAR SCENE TEXT DATASETS. [9]* MEANS THE UNSUPERVISED DOMAIN ADAPTATION RESULTS IN [9].

Method	Reference	IIIT	SVT	IC03	IC13
RARE [45]	CVPR 2016	86.2	85.8	93.9	92.6
STAR-Net [46]	CVPR 2016	87.0	86.9	94.4	92.8
R2AM [47]	CVPR 2016	83.4	82.4	92.2	90.2
CRNN [48]	TPAMI 2017	82.9	81.6	93.1	91.1
GRCNN [49]	NIPS 2017	84.2	83.7	93.5	90.9
Char-Net [50]	AAAI 2018	83.6	84.4	91.5	90.8
[42]	ICCV 2019	87.9	87.5	94.4	92.3
SSDAN [29]	CVPR 2019	83.8	84.5	92.1	91.8
ASSDA [8]	TIP 2021	88.3	88.6	**95.5**	93.7
[9]*	ICCV 2021	82.6	80.1	-	84.2
DOC	-	**89.0**	**89.0**	95.3	**94.3**
DOC*	-	**87.2**	**86.7**	**95.6**	**93.6**

B. Performance on Handwritten Dataset

In the experiment on the handwritten dataset, i.e. IAM, we used the synthetic text datasets MJ and ST as the source domain data and IAM as the target domain data. We compared our model with the best unsupervised domain adaption based text recognition models in recent years including the models in [9], SSDAN [29] and ASSDA [8]. The results are summarized in Table II and we have the following observations.

- DOC achieves the best results on this dataset in terms of both WER and CER.
- The method in [42] does not adopt an unsupervised domain adaption scheme. All UDA-based methods perform much better than it, demonstrating the effectiveness of the unsupervised domain adaption mechanism.
- It is worth noting that ASSDA also performs global- and local-level domain adaption. However, DOC still achieves much better results than ASSDA. For example, WER and CER of DOC are 6.34% and 3.44% lower than those of ASSDA, demonstrating the effectiveness of the domain adaption modules in DOC.

C. Performance on Scene Text Datasets

We also tested our proposed method on regular and irregular scene text datasets. Specifically, we used the synthetic text datasets MJ and ST as the source domain, the union of the training sets of IIIT, SVT, IC03 and IC13 as the target

TABLE IV
ACCURACY ON IRREGULAR SCENE TEXT DATASETS. [9]* MEANS THE UNSUPERVISED DOMAIN ADAPTATION RESULTS IN [9].

Method	Reference	IC15	SVTP	CUTE
RARE [45]	CVPR 2016	74.5	-	70.4
STAR-Net [46]	CVPR 2016	76.1	-	71.7
[42]	ICCV 2019	71.8	79.2	74.0
SSDAN [29]	CVPR 2019	**78.7**	-	73.9
ASSDA [8]	TIP 2021	**78.7**	-	76.3
[9]*	ICCV 2021	66.8	74.2	75.8
DOC	-	76.0	**81.2**	**77.0**
DOC*	-	**75.5**	**79.1**	74.2

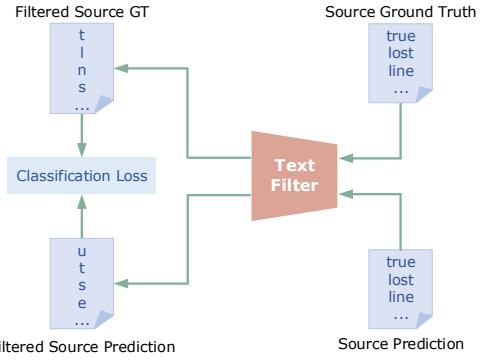


Fig. 6. The illustration of classification loss compared with clustering.

domain to train the model, and then tested it on the corresponding testing sets of benchmark datasets, respectively. The recognition accuracy results on regular and irregular datasets are summarized in Table III and Table IV, respectively. In addition, the model of [9]* uses the same source domain as DOC, but uses IAM, the handwritten text dataset, as the target domain, which is different from DOC. In order to further evaluate the robustness and generalization of DOC, we changed the target domain to IAM and conducted experiments. The results of DOC with this setting are denoted as DOC*. From these tables, we can observe that

- DOC achieves the best results on three regular datasets, i.e. IIIT, SVT, and IC13, two irregular datasets, i.e. SVTP and CUTE; and achieves the second-best result on IC03. Especially, it outperforms unsupervised domain adaptation-based methods on three regular datasets, and two irregular datasets. Moreover, compared with [9]*, DOC* achieves better results on three regular datasets, i.e. IIIT, SVT, IC13, and two irregular datasets, i.e. IC15 and SVTP. It further demonstrates the effectiveness of DOC.
- Both ASSDA [8] and DOC incorporate global- and local-level adaptation. Both of them outperform SSDAN[29] which only uses local-level adaptation. This verifies the effectiveness of global-level adaptation.
- DOC outperforms ASSDA [8] on most regular and irregular datasets. The main reasons include it adopts different adaptation schemes, especially the clustering-based learning scheme.

D. Ablation Evaluation

In our framework, several key components contribute to the performance, e.g. the global adaptation scheme, the local domain adaptation mechanism, and the clustering scheme. To evaluate their contribution to the improvement of performance, we further conducted ablation experiments on four regular scene datasets, three irregular scene datasets, and one handwritten dataset.

It is worth noting that we incorporate a clustering scheme in local-level feature adaptation. Actually, we can also adopt a classification scheme instead of a clustering scheme. For example, as shown in Figure 6, we get the filtered source ground truth $\tilde{\mathbf{y}}^s$ and the filtered source prediction $\widehat{\mathbf{y}^s}$ after the source prediction $\mathbf{y}^s$ and source ground truth $\mathbf{y}$ pass through the text filter in AFC module. Thereafter, we can minimize a classification loss between $\tilde{\mathbf{y}}^s$ and $\widehat{\mathbf{y}^s}$. In other words, we treat it as a classification task, and only leverage the information for the source domain, while ignoring that in the target domain. Here, we adopt a cross-entropy loss as defined as follows,

$$\mathcal{L}_{local\_cls} = cross\_entropy\left(\widehat{\mathbf{y}^s}, \tilde{\mathbf{y}}^s\right). \tag{14}$$

In order to evaluate which scheme (classification or clustering) is effective on the improvement of performance, we conducted experiments on 8 datasets. The results are shown in Table V. Thereinto, the baselines are defined as follows.

Baseline: A variant of DOC, which does not incorporate the clustering loss.

Baseline+cls: A variant of Baseline, which adopts a classification loss $\mathcal{L}_{local\_cls}$ instead of the clustering center loss $\mathcal{L}_{center}$.

Baseline+cls+c: A variant of DOC, which sues both the classification loss $\mathcal{L}_{local\_cls}$ and the clustering center loss $\mathcal{L}_{center}$.

From Table V, we have the following observations,

- DOC performs the best on 6 datasets in all baselines. Baseline+cls and Baseline+cls+c outperform Baseline on 3 and 7 datasets, respectively.
- DOC also outperforms Baseline+cls on 6 datasets, which adopts the classification loss. It demonstrates that the clustering loss is more effective than the classification loss on local-level domain adaptation. The main reason is that in the case of unsupervised domain adaptation, only using classification loss on the source domain data may waste the information contained in the target data, resulting in poor robustness of the learned model.
- DOC outperforms Baseline+cls+c on 7 datasets. One of the main reasons is that classification may disrupt the learning of clustering and lead to performance deterioration.

In addition, we also conducted experiments on five datasets to evaluate the effectiveness of Local Discriminator and Feature Clustering in AFC. The results are shown in Table VI. From this table, we have the observations as follows:

- DOC outperforms all models with incomplete components on all datasets, demonstrating the effectiveness of Local Discriminator and the clustering scheme. The lack of anyone will lead to a significant decline in performance.

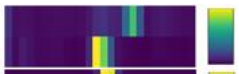
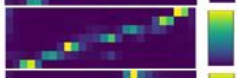
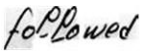
TABLE V
EVALUATION OF CLUSTERING OR CLASSIFICATION SCHEME DURING LOCAL FEATURE ADAPTATION. THE PERFORMANCE IS MEASURED IN TERMS OF WORD ACCURACY.

Method	$\mathcal{L}_{local\_cls}$	$\mathcal{L}_{center}$	IIIT	SVT	IC03	IC13	IC15	SVTP	CUTE	IAM
Baseline			88.0	87.2	95.1	93.3	77.7	80.0	74.9	55.4
Baseline+cls	✓		88.0	87.2	95.2	**95.0**	77.4	79.4	74.9	60.9
Baseline+cls+c	✓	✓	88.5	88.2	95.2	94.0	**77.9**	80.9	74.6	61.2
DOC		✓	**89.0**	**89.0**	**95.3**	94.3	76.0	**81.2**	**77.0**	**62.6**

TABLE VI
ABLATION EVALUATION OF LOCAL DISCRIMINATOR AND FEATURE CLUSTERING. THE PERFORMANCE IS MEASURED IN TERMS OF WORD ACCURACY. LD MEANS LOCAL DISCRIMINATOR

	$\mathcal{L}_{D_l}$	$\mathcal{L}_{center}$	IIIT	SVT	SVTP	CUTE	IAM
DOC w/o AFC			88.0	88.7	80.6	76.0	51.1
DOC w/o LD		✓	88.2	**89.0**	80.0	76.7	51.6
DOC	✓	✓	**89.0**	**89.0**	**81.2**	**77.0**	**62.6**

TABLE VII
THE COMPARISON OF THE ATTENTION MAP BETWEEN DOC AND THE MODEL IN [42]. WE MARK THE ERROR PREDICTED CHARACTERS IN RED AND THE CORRECT CHARACTERS IN GREEN.

Images	Model	Attention map	Prediction
	DOC		to
	[42]		4_
	DOC		difficult
	[42]		diff__ull
	DOC		off
	[42]		of_
	DOC		followed
	[42]		f__lowed

- DOC without Local Discriminator outperforms DOC without AFC on four datasets, which verifies the effectiveness of the clustering scheme.

E. Visualization

To gain deep insights into our model, we also give visualizations of some results, including some attention maps, recognition results, and local-level features adaptation, which are elaborated in the following subsections.

1) Visualization of Attention Map: We visualize the attention maps of some samples from handwritten datasets and the results are illustrated in Table VII. In this experiment, the attention maps of the model in [42] are used for comparison, which adopts no clustering and adaptation. From the results, we can find that when the characters of handwritten text are obviously inclined (images of the 1st and 3rd lines) or the distance between characters is very small (images of the 1st, 2nd and 3rd lines), our model can better distinguish each different characters and recognize them correctly than the model in [42].

TABLE VIII
RECOGNITION RESULTS OF DOC COMPARED WITH THE MODEL IN [42]. WE MARK THE ERROR CHARACTERS IN RED AND THE CORRECT CHARACTERS IN GREEN.

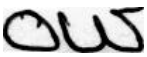
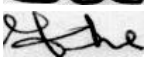
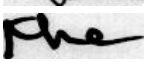
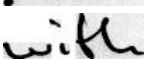
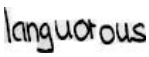
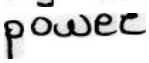
Images	Model in [42]	DOC
	whon	when
	tho	the
	gu	ow
	4he	the
	pho	phe
	will	with

TABLE IX
FAILURE CASES RECOGNIZED BY DOC. WE MARK THE ERROR CHARACTERS IN RED AND THE CORRECT CHARACTERS IN GREEN.

Error samples	Ground truth	Prediction of DOC
	down	dowm
	languorous	languotous
	power	powee
	something	somethims

2) Visualization of Recognition: Moreover, in Table VIII, we give some handwritten recognition results. The model in [42] is also compared with our approach. From the results, we can observe that, due to the large font gap between handwritten text and synthetic text, some characters with similar appearance ('e' in the image of the 2nd line and so on.) cannot be distinguished by the model in [42]. However, DOC can well capture the fine-grained differences between characters and distinguish them better, which also demonstrate the effectiveness of Feature Filter and Text Filter in the AFC module. In addition, we also provide some failure cases recognized by DOC in Table IX, we think the failure comes from the following reasons:

- As shown in Table IX, some texts are scrawled, which

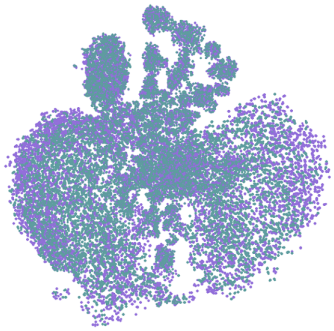


Fig. 7. The visualization of local-level features from source domain and target domain in our model.

are hard to recognize. In some samples, some letters are very similar to others in appearance. The AFC module clusters features at the character level, neglecting the use of word context information to some extent, which leads to recognition errors when viewing a character alone without observing the whole word.

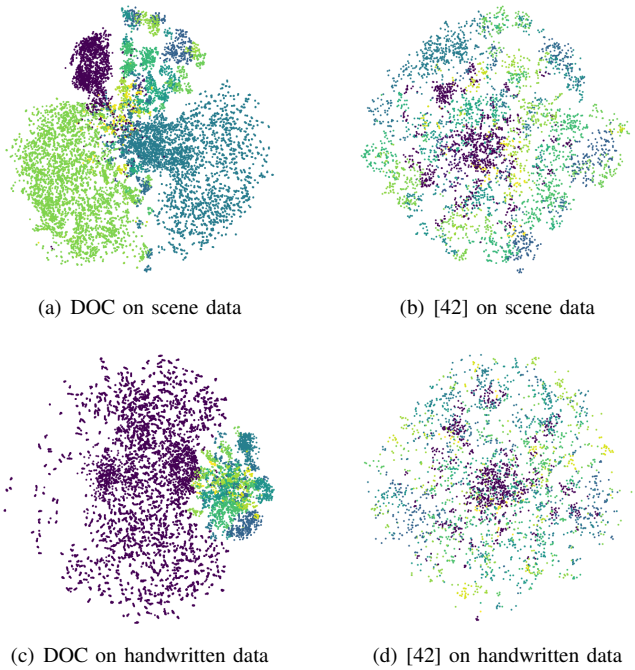
- The number of images in the target domain datasets is much smaller than that in the source domain datasets, which will result in limited adaptive knowledge learned by DOC.

3) Visualization of Local-level Feature: To demonstrate the effect of adaptation and clustering, we visualize the local-level features from our model's source and target domain. We used the visualization tool t-SNE to map the local-level features from the source domain and the target domain in our model with scene text as the target domain. The results are shown in Figure 7, in which purple dots and green dots represent the local features from the source domain and target domain, respectively. From this figure, we can observe that the local-level features DOC generates can make the text recognition model unable to distinguish whether they come from the source domain or the target domain. Therefore, DOC can not only better cluster the features from the two domains in the AFC module but also migrate the knowledge learned from the source domain to the target domain, which is also the key to improving the performance of DOC.

Moreover, we also visualize the filtered local-level features from the target domain on different target domain datasets. As shown in Figure 8, we compare the results of DOC with that of [42]. Comparing Figure 8 (a), (b) and Figure 8 (c), (d), we clearly observe that the feature representations generated by DOC is more aggregated; however those of [42] is a little dispersed. It more strongly explains that under the unsupervised condition of the target domain, the feature clustering in our AFC module can cluster effective local-level features better, to improve the effect of text recognition.

V. Conclusion

In this paper, we present a novel text image recognition framework. It addresses two important problems in this field: (1) There is a domain shift problem when a model tries to leverage data from other domains; (2) Under the unsupervised condition, there is no label in the target domain data in the training process. We propose an unsupervised domain adaptive-based text recognition model named DOC, which incorporates both global- and local-level feature adaptations for extracting domain invariant features. Moreover, we propose an Adaptive Feature Clustering (AFC) module, which can further filter, adapt and cluster the local-level features extracted by the attention mechanism, to separate local features of different categories as much as possible, so as to improve the recognition effect on the target domain. We test our proposed model through extensive experiments. The results demonstrate that DOC can achieve state-of-the-art results on benchmark datasets. We also illustrate the effectiveness of some modules/schemes, e.g. AFC, local feature adaptation, by some visualization results.



(a) DOC on scene data (b) [42] on scene data

(c) DOC on handwritten data (d) [42] on handwritten data

Fig. 8. The visualization of local-level features of target domain. Dots of different colors represent local-level features from different categories.

VI. Acknowledgement

The authors would like to thank the Associate Editor and the anonymous reviewers for their deep and careful work, which is much helpful in improving this paper.

References

[1] A. R. Katti, C. Reisswig, C. Guder, S. Brarda, S. Bickel, J. Höhne, and J. B. Faddoul, "Chargrid: Towards understanding 2d documents," *arXiv preprint arXiv:1809.08799*, 2018.

[2] J. Wang, L. Jin, and K. Ding, "Lilt: A simple yet effective language-independent layout transformer for structured document understanding," *CoRR*, vol. abs/2202.13669, 2022.

[3] Y. Li, J. Du, J. Zhang, and C. Wu, "A tree-structure analysis network on handwritten chinese character error correction," *IEEE Transactions on Multimedia*, 2022.

[4] A. Singh, V. Natarajan, M. Shah, Y. Jiang, X. Chen, D. Batra, D. Parikh, and M. Rohrbach, "Towards VQA models that can read," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8317–8326.

[5] A. F. Biten, R. Tito, A. Mafla, L. Gomez, M. Rusinol, E. Valveny, C. Jawahar, and D. Karatzas, "Scene text visual question answering," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 4291–4301.

[6] Y. Zhong, H. Zhang, and A. K. Jain, "Automatic caption localization in compressed video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 4, pp. 385–392, 2000.

[7] Q. Ye, Q. Huang, W. Gao, and D. Zhao, "Fast and robust text detection in images and video frames," *Image and Vision Computing*, vol. 23, no. 6, pp. 565–576, 2005.

[8] Y. Zhang, S. Nie, S. Liang, and W. Liu, "Robust text image recognition via adversarial sequence-to-sequence domain adaptation," *IEEE Transactions on Image Processing*, vol. 30, pp. 3922–3933, 2021.

[9] A. K. Bhunia, A. Sain, P. N. Chowdhury, and Y.-Z. Song, "Text is text, no matter what: Unifying text recognition using knowledge distillation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 983–992.

[10] F. Zhan, C. Xue, and S. Lu, "GA-DAN: geometry-aware domain adaptation network for scene text detection and recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9104–9114.

[11] X. Chen, L. Jin, Y. Zhu, C. Luo, and T. Wang, "Text recognition in the wild: A survey," *ACM Computing Surveys (CSUR)*, vol. 54, no. 2, pp. 1–35, 2021.

[12] M. Li, B. Fu, Z. Zhang, and Y. Qiao, "Character-aware sampling and rectification for scene text recognition," *IEEE Transactions on Multimedia*, 2021.

[13] C. Zhang, Y. Xu, Z. Cheng, S. Pu, Y. Niu, F. Wu, and F. Zou, "SPIN: structure-preserving inner offset network for scene text recognition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, pp. 3305–3314.

[14] Y. Mou, L. Tan, H. Yang, J. Chen, L. Liu, R. Yan, and Y. Huang, "Plugnet: Degradation aware scene text recognition supervised by a pluggable super-resolution unit," in *Proceedings of European Conference on Computer Vision*, 2020, pp. 158–174.

[15] J. Chen, B. Li, and X. Xue, "Scene text telescope: Text-focused scene image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 026–12 035.

[16] Z. Wan, J. Zhang, L. Zhang, J. Luo, and C. Yao, "On vocabulary reliance in scene text recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 422–11 431.

[17] X. Yue, Z. Kuang, C. Lin, H. Sun, and W. Zhang, "Robustscanner: Dynamically enhancing positional clues for robust text recognition," in *Proceedings of European Conference on Computer Vision*, 2020, pp. 135–151.

[18] S. Fang, H. Xie, Y. Wang, Z. Mao, and Y. Zhang, "Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 7098–7107.

[19] M. Li, B. Fu, H. Chen, J. He, and Y. Qiao, "Dual relation network for scene text recognition," *IEEE Transactions on Multimedia*, 2022.

[20] T. Wang, Y. Zhu, L. Jin, C. Luo, X. Chen, Y. Wu, Q. Wang, and M. Cai, "Decoupled attention network for text recognition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, pp. 12 216–12 224.

[21] Y. Zhang, "A survey of unsupervised domain adaptation for visual recognition," *CoRR*, vol. abs/2112.06745, 2021.

[22] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[23] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2962–2971.

[24] H.-K. Hsu, C.-H. Yao, Y.-H. Tsai, W.-C. Hung, H.-Y. Tseng, M. Singh, and M.-H. Yang, "Progressive domain adaptation for object detection," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, 2020, pp. 749–757.

[25] Y. Zhang, H. Tang, K. Jia, and M. Tan, "Domain-symmetric networks for adversarial domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5031–5040.

[26] M. Jing, L. Meng, J. Li, L. Zhu, and H. T. Shen, "Adversarial mixup ratio confusion for unsupervised domain adaptation," *IEEE Transactions on Multimedia*, 2022.

[27] W. Deng, L. Zhao, Q. Liao, D. Guo, G. Kuang, D. Hu, M. Pietikäinen, and L. Liu, "Informative feature disentanglement for unsupervised domain adaptation," *IEEE Transactions on Multimedia*, vol. 24, pp. 2407–2421, 2021.

[28] L. Kang, M. Rusiñol, A. Fornés, P. Riba, and M. Villegas, "Unsupervised adaptation for synthetic-to-real handwritten word recognition," in *IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 3491–3500.

[29] Y. Zhang, S. Nie, W. Liu, X. Xu, D. Zhang, and H. T. Shen, "Sequence-to-sequence domain adaptation network for robust text image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2740–2749.

[30] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *Proceedings of European Conference on Computer Vision*, 2016, pp. 499–515.

[31] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2315–2324.

[32] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," *arXiv preprint arXiv:1406.2227*, 2014.

[33] A. Mishra, K. Alahari, and C. Jawahar, "Scene text recognition using higher order language priors," in *Proceedings of British Machine Vision Conference*, 2012, pp. 1–11.

[34] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, 2011, pp. 1457–1464.

[35] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, R. Young, K. Ashida, H. Nagai, M. Okamoto, H. Yamamoto *et al.*, "ICDAR 2003 robust reading competitions: entries, results, and future directions," *International Journal of Document Analysis and Recognition*, vol. 7, no. 2, pp. 105–122, 2005.

[36] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. i Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazan, and L. P. De Las Heras, "ICDAR 2013 robust reading competition," in *International Journal of Document Analysis and Recognition*, 2013, pp. 1484–1493.

[37] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu *et al.*, "ICDAR 2015 competition on robust reading," in *International Journal of Document Analysis and Recognition*, 2015, pp. 1156–1160.

[38] T. Q. Phan, P. Shivakumara, S. Tian, and C. L. Tan, "Recognizing text with perspective distortion in natural scenes," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 569–576.

[39] A. Risnumawan, P. Shivakumara, C. S. Chan, and C. L. Tan, "A robust arbitrary text detection system for natural scene images," *Expert Systems with Applications*, vol. 41, no. 18, pp. 8027–8048, 2014.

[40] U.-V. Marti and H. Bunke, "The iam-database: An english sentence database for offline handwriting recognition," *International Journal on Document Analysis and Recognition*, vol. 5, no. 1, pp. 39–46, 2002.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, 2016, pp. 770–778.

[42] J. Baek, G. Kim, J. Lee, S. Park, D. Han, S. Yun, S. J. Oh, and H. Lee, "What is wrong with scene text recognition model comparisons? dataset and model analysis," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 4714–4722.

[43] F. L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 6, pp. 567–585, 1989.

[44] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *CoRR*, vol. abs/1212.5701, 2012.

[45] B. Shi, X. Wang, P. Lyu, C. Yao, and X. Bai, "Robust scene text recognition with automatic rectification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4168–4176.

[46] W. Liu, C. Chen, K.-Y. K. Wong, Z. Su, and J. Han, "Star-net: A spatial attention residue network for scene text recognition," in *Proceedings of British Machine Vision Conference*, 2016, p. 7.

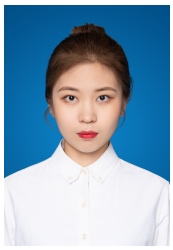
[47] C.-Y. Lee and S. Osindero, "Recursive recurrent nets with attention modeling for OCR in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2231–2239.

[48] 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 Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 11, pp. 2298–2304, 2016.

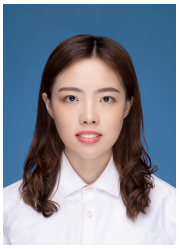
[49] J. Wang and X. Hu, "Gated recurrent convolution neural network for OCR," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[50] W. Liu, C. Chen, and K.-Y. Wong, "Char-net: A character-aware neural network for distorted scene text recognition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018, pp. 7154–7161.

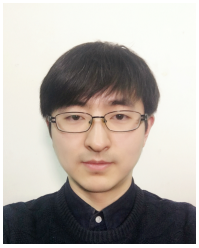
This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2023.3245404



Xue-Ying Ding received the bachelor's degree in software engineering from Shandong University, China, in 2020. Currently, she is a graduate student at the School of Software, Shandong University, Jinan, China. Her current research interests include machine learning, computer vision, and scene text recognition.



Xiao-Qian Liu received the M.S. degree in Control engineering in 2020 from Shandong University, China. She is currently pursuing the Ph.D. degree in artificial intelligence at the School of Software, Shandong University, Jinan, China. Her research interests include deep learning, computer vision, domain adaption, and OCR.



Xin Luo received the Ph.D. degree in computer science from Shandong University, Jinan, China, in 2019. He is currently an assistant professor with the School of Software, Shandong University, Jinan, China. His research interests mainly include machine learning, multimedia retrieval and computer vision. He has published over 20 papers on TIP, TKDE, ACM MM, SIGIR, WWW, IJCAI, et al. He serves as a Reviewer for ACM International Conference on Multimedia, International Joint Conference on Artificial Intelligence, AAAI Conference on Artificial Intelligence, the IEEE Transactions on Cybernetics, Pattern Recognition, and other prestigious conferences and journals.

Xin-Shun Xu is currently a professor with the School of Software, Shandong University. He received his M.S. and Ph.D. degrees in computer science from Shandong University, China, in 2002, and Toyama University, Japan, in 2005, respectively. He joined the School of Computer Science and Technology at Shandong University as an associate professor in 2005, and joined the LAMDA group of Nanjing University, China, as a postdoctoral fellow in 2009. From 2010 to 2017, he was a professor at the School of Computer Science and Technology, Shandong University. He is the founder and the leader of MIMA (Machine Intelligence and Media Analysis) group of Shandong University. His research interests include machine learning, information retrieval, data mining and image/video analysis and retrieval. He has published in TIP, TKDE, TMM, TCSVT, AAAI, CIKM, IJCAI, MM, SIGIR, WWW and other venues. He also serves as a SPC/PC member or a reviewer for various international conferences and journals, *e.g.* AAAI, CIKM, CVPR, ICCV, IJCAI, MM, TCSVT, TIP, TKDE, TMM and TPAMI.