Robust Text Image Recognition via Adversarial Sequence-to-Sequence Domain Adaptation

Yaping Zhang, Shuai Nie, Shan Liang, and Wenju Liu

***Abstract*— Robust text reading is a very challenging problem, due to the distribution of text images changing significantly in real-world scenarios. One effective solution is to align the distribution between different domains by domain adaptation methods. However, we found that these methods might struggle when dealing sequence-like text images. An important reason is that conventional domain adaptation methods strive to align images as a whole, while text images consist of variable-length fine-grained character information. To address this issue, we propose a novel Adversarial Sequence-to-Sequence Domain Adaptation (ASSDA) method to learn "where to adapt" and "how to align" the sequential image. Our key idea is to mine the local regions that contain characters, and focus on aligning them across domains in an adversarial manner. Extensive text recognition experiments show the ASSDA could efficiently transfer sequence knowledge and validate the promising power towards the various domain shift in the real world applications.**

***Index Terms*— Sequence-to-sequence, domain adaptation, text image recognition.**

I. Introduction

Deep learning methods have achieved remarkable results on text image reading [1]–[6]. While excellent performance has been achieved on the benchmark datasets, robust text image reading in the real world still faces challenges from large variance in viewpoints, appearances and backgrounds, which may cause a considerable domain shift between the training and test data. Several samples are illustrated in Fig. 1, where we can observe a considerable domain shift.

Significant performance drop caused by domain shifts has been observed in many realistic applications. One intuitive and effective solution to this problem is to collect large scale annotated text images, while they are often extremely expensive and cannot cover all diversity. Therefore, it is highly desirable to develop an algorithm to adapt text image recognition models to a new domain that is visually different from the source training domain. An appealing alternative is to take advantage of the unsupervised text images to reduce domain shifts.

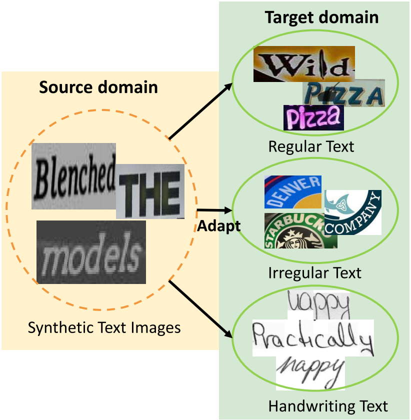


Fig. 1. Illustration of three typical domain shifts in text image recognition scenarios. Here, we can use easily labeled synthetic text images as the source domain (left), and the unsupervised real text images (right) in different scenes as the target domain.

Unsupervised domain adaptation (UDA) has been developed to use unannotated data to reduce the domain shift between the different domains [7], [8]. We could consider the cross-domain text image recognition problem as an unsupervised domain adaptation scenario: full supervision is given in the source domain while no supervision is available in the target domain. However, recent UDA methods generally optimize the global representation to minimize some measure of domain shift, such as maximum mean discrepancy (MMD) [9], [10], correlation alignment distance (CORAL) [11], [12], or adversarial loss [7], [13]–[15], where they typically consider the input images as a whole. While a text image is a combination of different characters, which is a variable-length label sequence instead of an isolation. Thus, the domain shift could occur not only on global image level (*e.g* image background, illumination, *etc.*), but also on local character level (*e.g* character font, content, *etc.*). Consequently, most of popular domain adaptation methods cannot be effectively applied to the sequence prediction, since a global representation neglects important fine-grained information at the local character level, which in turn cannot sufficiently describe the content of sequence-like images.

To address the aforementioned issues, we develop an Adversarial Sequence-to-Sequence Domain Adaptation (ASSDA) method for robust text image recognition. As shown in Fig. 2, the proposed ASSDA incorporates two key components, *global-level alignment* and *local-level alignment*, which respectively address the question of "global image-level

Manuscript received October 18, 2019; revised February 28, 2020 and April 16, 2020; accepted April 16, 2020. Date of publication March 23, 2021; date of current version March 22, 2021. This work was supported in part by the China National Natural Science Foundation under Grant 61971419, Grant 61602089, Grant 61573357, and Grant 61503382 and in part by the Project of Xueersi Online School under Grant Y9D2M10101. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Dong Xu. *(Corresponding author: Wenju Liu.)*

Yaping Zhang is with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing 100190, China, and also with the School of Artificial Intelligence, University of Chinese Academy of Sciences (UCAS), Beijing, China.

Shuai Nie, Shan Liang, and Wenju Liu are with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing 100190, China (e-mail: lwj@nlpr.ia.ac.cn).

Digital Object Identifier 10.1109/TIP.2021.3066903

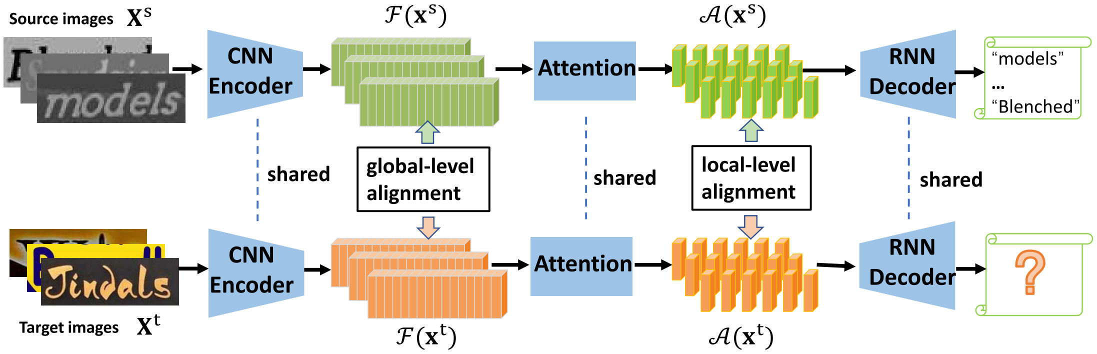

Source images $\mathbf{X}^s$
models
CNN Encoder
$\mathcal{F}(\mathbf{x}^s)$
Attention
$\mathcal{A}(\mathbf{x}^s)$
RNN Decoder
"models"
...
"Blenched"
shared
global-level alignment
shared
local-level alignment
shared
Jindals
Target images $\mathbf{X}^t$
CNN Encoder
$\mathcal{F}(\mathbf{x}^t)$
Attention
$\mathcal{A}(\mathbf{x}^t)$
RNN Decoder
?

Fig. 2. The structure of ASSDA consists of: a CNN encoder to map the input images into a sequence of high-level feature vectors, an attention unit between the encoder and decoder to adaptively focus on the location of character, and an RNN decoder to convert encoded features into output strings recurrently. We tackle the domain shift on two levels, the global-level alignment and the local-level alignment, where two domain classifiers are built on two levels and trained in an adversarial training manner.

domain shift" and "local character-level domain shift". In each component, we train a domain classifier and employ the adversarial training strategy to learn robust features that are domain-invariant. Specifically, the proposed ASSDA is an attention based encoder-decoder model for handling sequences, which is derived from [6]. The key idea is that attention module could learn *"where to adapt"*. *i.e.*, it could automatically concentrate on the most relevant region of one character. Then the local-level alignment could leverage the attended local fine-grained character-level features on both domains to learn *"how to align"* via an adversarial manner. Overall, the cooperation of these components leads to an adaptation process that focuses on the region of interest, thus improving the effectiveness.

We summarize our contributions as follows:

- We introduce an Adversarial Sequence-to-sequence Domain Adaptation dubbed ASSDA, for robust text image recognition, which bridges the sequence-like text image recognition and domain adaptation.
- We design two domain adaptation modules to alleviate the domain shift at both the global-level and local-level, where they collaboratively contribute to guiding model find the domain-invariant representations.
- We introduce a spatial normalization network to the domain adaption process, which makes the model robust and could be generalized to more complex scenes.

The paper surpasses its conference version SSDAN [16] with three major extensions:

- To address the insufficient knowledge transferring problem in the SSDAN, we redesign the alignment module to consider the inevitable cross-domain shifts at different levels, rather than only local-level domain shifts.
- Regarding that various perspective distortions and geometric noises in real scenes, the extended ASSDA incorporates a spatial normalization network to the domain adaption process. It makes our model could be generalized to broader scenes in a unified framework.
- We explore the application of ASSDA in more complex tasks, including irregular text recognition, and more complex cross-domain adaptation tasks.

With these extensions, ASSDA outperforms conference paper [16] by a large margin and shows broader applicability. Extensive experiments on benchmark datasets validate the promising power of the proposed model towards various domain shift settings, including *synthetic-to-real* (synthetic text to real scene text), and *cross-domain* (scene text to handwriting text, and handwriting text to scene text).

The remaining parts of this paper are organized as follows. We firstly discuss the related work in Sec.II. Then, we describe the preliminaries and the proposed model ASSDA in Sec.III and Sec.IV, respectively. Furthermore, we present the evaluation and detailed analysis on it. Finally, we draw the concluding remarks.

II. Related Work

In this section, we review the literature of text recognition methods. Then we discuss the recent unsupervised domain adaptation techniques and its trials on text recognition.

A. Text Recognition Methods

Deep learning methods have achieved remarkable results on image text reading [4], [6], [17]. Earlier DNN based methods recognized image text depending on the segmentation of each character [18] or a non-maximum suppression [19], which may be very challenging because of the complicated background and the inadequate distance between consecutive characters. As well, they did not unleash the full potential of word context information in the recognition. Recently, some researchers treated the text recognition task as a sequence learning problem [1], [2], [4], [17]: firstly encoding an entire image text into a sequence of features with CNN, and then decoding character sequence recurrently via recurrent neural network (RNN) with CTC [1], [2] or attention schemes [4], [6]. Nevertheless, CTC methods [1], [2] cannot handle complicated two-dimensional structures, such as irregular curve text. In this case, the attention based encoder-decoder model [4] has shown promising performances as symbol segmentation can be adaptively performed through attention model.

However, the literature is relatively sparse on building a robust text recognizer that can handle varying data in abundance of scenarios effectively. Some methods were designed to handle perspective distortion exhibited in the scene text. For example, [20] and [21] introduce a spatial transformer network [22] to rectify the entire text before recognition.

Furthermore, CharNet [4] tried to introduce a character-level spatial transformer to rectify individual characters, which was capable of handling more complicated forms of distortion that cannot be modeled by a single global transformation easily. However, they were only designed for spatial affine distortions and hard to generalize to the distortion caused by handwriting styles or various backgrounds. In summary, existing text image recognition methods are usually designed for a specific scenario, and the intrinsic domain shift in the text image data is commonly neglected. While our domain adaptation model is designed for cross-domain tasks. Specifically, the ASSDA utilizes the domain adaptation technique to tackle the domain shift problem, which adaptively performs the global-level and character-level adaption in sequence-like text images.

B. Unsupervised Domain Adaptation

Unsupervised domain adaptation has received increasing attention in recent years [7], [8], [13]–[15], [23], [24]. It's often considered as a promising remedy to tackle the domain shift problem [8]. There are many domain adaption methods designed for cross-dataset visual recognition. For example, an exemplar SVM-based domain adaptation method [24] is designed for cross-domain object recognition and action recognition, where the source domain with complex distribution is decomposed into many simpler sub-domains. Instead of learning domain invariant features, this work aims to learn the robust target classifiers directly. The domain adaptation from multi-view to single-view (DAM2S) achieves an interesting multi-view RGB-D data to single-view RGB visual recognition task, which simultaneously reduces the global domain shift between two domains and maximizes the correlation between two global features from different views. This work is also based on SVM classifier.

The majority of recent works use deep convolutional architectures to map the source and target domains into a shared space where the domains are aligned. They generally optimize the global representation via minimizing some measure of domain shift, such as MMD [9], [10], CORAL [11], [12], or adversarial loss [13]–[15], [25]–[27]. For example, collaborative and adversarial network (CAN) [27] uses a collaborative and adversarial training scheme to simultaneously learn the discriminative low-level representation and domain invariant high-level representation. CAN aims to reduce the coarse-grained global level domain shift for two common visual recognition tasks: object recognition and video action recognition.

As stated above, most of recent works concentrate on global visual recognition. However, much less attention has been paid to other computer vision tasks. Recently there are some new concerning tasks such as object detection [28], [29], where they leverage a semantic related but distribution different source domain with sufficient labels of bounding boxes to train a detector on the unlabeled target domain. One of the inspiring work [29] proposes multiple adversarial alignment modules to align the hierarchical feature between two domains, where the available labels of bounding boxes in source domain could provide fine-grained instance-level information. In contrast, our model focuses on variable-length sequence-like text images recognition, where there are no character-level annotations. Essentially, our source domain is weak-supervised sequence-like data. We need to know "where to adapt" in a sequence-like text image. Therefore, the key idea of our model is to address the question of "where to adapt" via attention mechanism.

C. Domain Adaptation for Text Recognition

Some methods have been evaluated on the handwritten character or natural scene digital dataset for recognition tasks and have shown effective performance. However, the majority of recent works simply consider the entire image as a whole, focusing on the design of losses or metrics. Some other methods have been proposed to adapt the different font styles for image-to-image translation via adversarial learning [30]. Similarly, these methods limitedly translate the font in different style of signal characters on a global image, which are still cannot be extended to text-line images. Recently, [31] introduced a geometry-aware domain adaptation network (GA-DAN) to convert a synthetic text image to a real scene text image, and then use the converted text image to train the target recognition model. Therefore, the GA-DAN cannot be trained in an end-to-end unified framework. And the GA-DAN still neglects the fine-grained character level domain shift in a sequence-like text image. To address these problems, we develop a sequence-to-sequence domain adaptation to focus on not only global-level domain shifts but also the fine-grained character-level domain shifts, which could transfer variable-length sequence knowledge successfully in an end-to-end unified framework.

III. Preliminaries

A. Sequence-to-Sequence Domain Adaptation

In this paper, unsupervised sequence-to-sequence domain adaptation is developed for robust text image recognition. Specifically, the source domain text images with well-annotated text labels (a sequence of characters or symbols) are available, while we only have an access to unlabeled text images in target domain, which is in a different distribution. Our task is to train a text image recognition system that can generalize well to the target domain, utilizing unsupervised sequence data. Specifically, we desire to obtain a domain-invariant feature representation that works equally well in both domains. More formally, we assume that there are N^s annotated source domain samples $X^s = \{\mathbf{x}_i^s\}_{i=0}^{N^s}$ with the corresponding labels $\mathcal{Y}^s = \{\mathbf{y}_i^s\}_{i=0}^{N^s}$, and N^t unlabeled target-domain samples $X^t = \{\mathbf{x}_i^t\}_{i=0}^{N^t}$ without any available annotated labels during the training period. For $\mathbf{y} \in \mathcal{Y}^s$, $\mathbf{y} = \{y_1, y_2, \ldots, y_T\}$, where y_k and T denotes a character label and the variable length of text, respectively.

B. Attention Text Recognition

The attentive text recognition can be essentially considered as learning a mapping between a sequence of feature maps encoded from a sequence-like text image $\mathbf{x}$, and a ground truth label sequence $\mathbf{y} = \{y_1, y_2, \ldots, y_T\}$. As shown in Fig. 2,

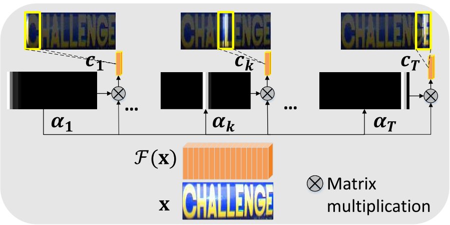


Fig. 3. The illustration of the attention procedure, which is used to localize individual characters in the image. (We use the raw image as intermediate visualization result instead of convolution feature maps for the best view.).

the attentive text recognition pipeline consists of three major components: a CNN encoder, an attention module and a GRU decoder.

1) CNN Encoder: CNN encoder $\mathcal{F}$ takes the raw input image $\mathbf{x}$ from the source or target domain, and produces a feature grid $\mathcal{F}(\mathbf{x})$ of size $H' \times W' \times D$, where D denotes the number of channels, H' and W' are the resulted feature map height and width, respectively. The encoder output is then reshaped as a grid sequence of L elements, $L = H' \times W'$. Each of these elements is a D-dimensional feature vector that corresponds to a local region of the image through its corresponding receptive field. Hence, the whole encoded image $\mathcal{F}(\mathbf{x})$ could be reformatted as,

$$\mathcal{F}(\mathbf{x}) = [\mathbf{f}_1, \ldots, \mathbf{f}_L], \quad \mathbf{f}_i \in R^D, \tag{1}$$

where $\mathbf{f}_i$ corresponds to i-th grid of the encoded image $\mathcal{F}(\mathbf{x})$, which preserves specific spatial information of the input image $\mathbf{x}$.

2) Attention: Although the CNN encoder keeps the spatial information, we cannot decide the location of a specific character in a text image. Therefore, an attention model is introduced to learn which part of the text image is the most relevant to a decoding character. As shown in Fig. 3, the attention is a T-step process. At time-step k, the representation of the most relevant part to character y_k of encoding feature map $\mathcal{F}(\mathbf{x})$ is defined as a context vector $\mathbf{c}_k$:

$$\mathbf{c}_k = \sum_{i=1}^{L} \alpha_{k,i} \mathbf{f}_i, \tag{2}$$

where, the attention weights $\alpha_{k,i}$ is calculated by

$$\alpha_{k,i} = \frac{\exp(\mathbf{s}_{k,i})}{\sum_{j=1}^{L} \exp(\mathbf{s}_{k,j})}, \tag{3}$$

where the attention score $\mathbf{s}_{k,i}$ indicates the probability of that the model attends to the i-th sub-region in the encoded map $\mathcal{F}(\mathbf{x})$ when decoding the k-th character of the text image. Following the past empirical work [6], we defined the attention score as

$$\mathbf{s}_{k,i} = \beta^\top \tanh(\mathbf{W}_h \mathbf{h}_{k-1} + \mathbf{W}_f \mathbf{f}_i), \tag{4}$$

where β, $\mathbf{W}_h$ and $\mathbf{W}_f$ are the parameters to be learnt, $\mathbf{h}_{k-1}$ is the previous decoding state in the decoder.

3) RNN Decoder: An RNN decoder is employed to predict the string of an input text image recurrently, where we use gated recurrent unit (GRU) neural network. At decoding time step k, the GRU leverages the context vector $\mathbf{c}_k$, previous state $\mathbf{h}_{k-1}$ and previous predicted character y_{k-1} to generate a new hidden state

$$\mathbf{h}_k = GRU(\mathbf{h}_{k-1}, y_{k-1}, \mathbf{c}_k), \tag{5}$$

where, $\mathbf{c}_k$ is generated by the attention mechanism, which focuses on the most relevant region of current decoding character. Then, the probability of current predicted symbol y_k is computed by :

$$p(y_k|y_{k-1}, \mathbf{c}_k) = g\left(\mathbf{W}_o \tanh(\mathbf{E}\tilde{\mathbf{y}}_{k-1} + \mathbf{W}_d \mathbf{h}_k + \mathbf{W}_c \mathbf{c}_k)\right), \tag{6}$$

where g denotes a softmax activation function, $\mathbf{W}_o$, $\mathbf{W}_d$ and $\mathbf{W}_c$ are the mapping matrices, $\mathbf{E}$ is the embedding matrix, and $\tilde{\mathbf{y}}_{k-1}$ is the one-hot vector of character label y_{k-1}.

The probability of the sequential labels $\mathbf{y}$ is finally given by the product of the probability of each label:

$$P(\mathbf{y}|\mathcal{A}(\mathbf{x})) = \prod_{k=1}^{T} p(y_k|y_{k-1}, \mathbf{c}_k), \tag{7}$$

where $\mathcal{A}(\mathbf{x}) = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_T\}$, which could be regarded as a sequence of attended character-level features from an input text image $\mathbf{x}$.

IV. Proposed Method

A. Framework Overview

In this paper, we propose an adversarial sequence-to-sequence domain adaptation to gradually learn "where to adapt" and "how to align" the sequential feature space between two domains. We introduce two domain adaptation components: global-level alignment and local-level alignment, which are used to align the feature representation distribution on not only global-level but also local-level. During the training, these two modules can guide the learning of features to reduce the gap between domains. After training, the alignment module is no longer needed. Only the backbone of the text image recognition will be used for effective inference, while benefiting from the learned domain-invariant features.

B. Global-Level Alignment

Aligning global image level representations generally helps to reduce the shift caused by the global image difference such as image style, illumination, *etc.* We wish to learn a global representation that is invariant for both appearances (background, illumination, *etc.*), and geometry characteristics (translation, rotation, and affine transformation, *etc.*). However, text images with diverse shapes in real scenes heavily make the model struggling for learning invariant global representations.

To tackle this problem, we introduce a spatial normalization network $\mathcal{N}$, which transforms a raw input image $\mathbf{x}$ into a geometry-normalized image $\hat{\mathbf{x}}$. Motivated by ASTER [32], we apply a learnable Thin-Plate Spline (TPS) transformation network as $\mathcal{N}$ to normalize the irregular text image. This can be mathematically written as

$$\hat{\mathbf{x}} = \mathcal{N}(\mathbf{x}). \tag{8}$$

TPS is a variant of the spatial transformation network (STN) [32], which employs a smooth spline interpolation between a set of fiducial points. Specifically, TPS finds multiple fiducial points at the upper and bottom enveloping points, and normalizes the character region to a predefined rectangle. More detailed information can be referred to ASTER [32].

After the spatial normalization network $\mathcal{N}$, we could further get encoded global-level visual representations $\mathcal{F}(\hat{\mathbf{x}})$ from CNN encoder. To eliminate the domain distribution mismatch on the global level, we introduce a global-level domain classifier D_g. The loss of the global-level domain discriminator D_g as $\mathcal{L}_g$ is denoted as follows,

$$\mathcal{L}_{g_s} = -E_{\mathbf{x}_s \sim \mathbf{X}^s} \left\{ \log(1 - D_g(\mathcal{F}(\hat{\mathbf{x}}_s))) \right\}, \tag{9}$$

$$\mathcal{L}_{g_t} = -E_{\mathbf{x}_t \sim \mathbf{X}^t} \left\{ \log(D_g(\mathcal{F}(\hat{\mathbf{x}}_t))) \right\}, \tag{10}$$

$$\mathcal{L}_g(\mathcal{F}, D_g, \mathbf{X}^s, \mathbf{X}^t) = \frac{1}{2}\left(\mathcal{L}_{g_s} + \mathcal{L}_{g_t}\right). \tag{11}$$

C. Local-Level Alignment

The *global-level alignment* of DA models uses the feature map after the last convolutional layer to align the global feature distribution of different domains. However, such a setting has three limitations. First the model ignores the alignment of fine-grained local character features, making certain domain-sensitive local features weaken the generalization ability of the adaptive model. Second, single adaptation (one domain classifier) is difficult to cancel the data bias between the source domain and the target domain, because the sequential text images are complex combinations of local characters. Third, due to that the target sequence domain is unsupervised, the whole ground truth strings of target domain may suffer the inconsistency with the source domain.

To solve the aforementioned problems, we introduce the idea of local character-level adaptation. We desire to find those regions that cover fine-grained character region, and then align the character-level feature in both the source and target domain. A natural idea is to utilize the attended character regions derived from the attention mechanism. More formally, through attention mechanism, an input image $\mathbf{x}$ can be adaptively decomposed into a series of character-level feature set $\mathcal{A}(\mathbf{x}) = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_T\}$, where $\mathbf{c}_k$ presents the feature of k-th character in the text image $\mathbf{x}$. Specifically, a source text image $\mathbf{x}^s$ and a target text image $\mathbf{x}^t$ are decomposed into a source and target attended character-level feature set $\mathcal{A}(\mathbf{x}^\mathbf{s})$ and $\mathcal{A}(\mathbf{x}^\mathbf{t})$, respectively.

We notice that if the attention context vector fails to focus on the region of effective character, the adaptation on the attention context vector will not help. To overcome this problem, we introduce a gate mechanism to select effective attention context vectors to perform domain adaptation, as illustrated in Fig. 4. An intuition is that if the current attention context vector $\mathbf{c}_k$ is distinguishable, the probability that $\mathbf{c}_k$ belongs to one specific character y_k will be relatively higher than others. Hence, we further introduce an adaption gate function $\delta(\mathbf{c}_k)$ to judge if a context vector $\mathbf{c}_k$ is attending to a valid character,

$$\delta(\mathbf{c}_k) = \begin{cases} 1 & \text{if } p\left(y_k | y_{k-1}, \mathbf{c}_k\right) > p_c \\ 0 & \text{if } p\left(y_k | y_{k-1}, \mathbf{c}_k\right) < p_c, \end{cases} \tag{12}$$

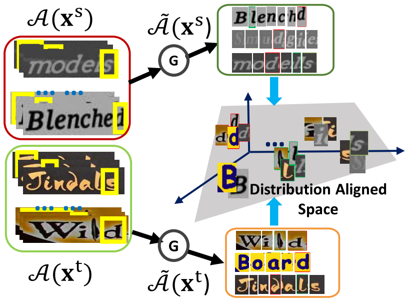

$\mathcal{A}(\mathbf{x}^\mathrm{s})$
$\tilde{\mathcal{A}}(\mathbf{x}^\mathrm{s})$
G
Distribution Aligned Space
G
$\mathcal{A}(\mathbf{x}^\mathrm{t})$
$\tilde{\mathcal{A}}(\mathbf{x}^\mathrm{t})$

Fig. 4. The local-level alignment is to adaptively find valid character-level domain-invariant features between the source and target domain.

where p_c is a confidence threshold. Furthermore, a gate function set $\mathbf{G}$ is adaptively changed according to the specific input image $\mathbf{x}$, which is expressed as:

$$\mathbf{G}(\mathbf{x}) = \{\delta(\mathbf{c}_1), \ldots, \delta(\mathbf{c}_T)\}. \tag{13}$$

Through the gate function, we can update attention context vector set by adaptation gate function set $\mathbf{G}(\mathbf{x})$,

$$\tilde{\mathcal{A}}(\mathbf{x}) = \mathcal{A}(\mathbf{x}) \otimes \mathbf{G}(\mathbf{x}), \tag{14}$$

where $\otimes$ denotes element-wise product operator. Specifically, if $\mathbf{c}_k \times \delta(\mathbf{c}_k) = 0$, then current context vector $\mathbf{c}_k$ will not be added in a new attention context vector set.

By decomposing the text strings into a set of characters, the source and target domain will statistically share the same label space in character-level, and thus the influence of the misalignment problem can be alleviated. Furthermore, we introduce a local-level domain discriminator D_l to align the character-level distribution. The local-level adaptation loss can now be written as

$$\mathcal{L}_{l_s} = -E_{\mathbf{x}_s \sim \mathbf{X}^s} \left\{ E_{\mathbf{c}_k^s \sim \tilde{\mathcal{A}}(\mathbf{x}^\mathbf{s})} \left[\log(1 - D_l(\mathbf{c}_k^s)) \right] \right\}, \tag{15}$$

$$\mathcal{L}_{l_t} = -E_{\mathbf{x}_t \sim \mathbf{X}^t} \left\{ E_{\mathbf{c}_k^t \sim \tilde{\mathcal{A}}(\mathbf{x}^\mathbf{t})} \left[\log(D_l(\mathbf{c}_k^t)) \right] \right\}, \tag{16}$$

$$\mathcal{L}_l(\mathcal{F}, D_g, \mathbf{X}^s, \mathbf{X}^t) = \frac{1}{2}\left(\mathcal{L}_{l_s} + \mathcal{L}_{l_t}\right), \tag{17}$$

where $\mathbf{c}_k^s$ and $\mathbf{c}_k^t$ denote the k-th attended character-level feature vector in a source and a target image, respectively.

D. Overall Objective

With the well-annotated source-domain data, we could learn an optimized source text image recognizer by minimizing a supervised decoding loss, where we can use the negative log likelihood of sequential probability as the decoding loss $\mathcal{L}_{dec}$ to measure the differences between the predicted and the source labeled character sequences:

$$\mathcal{L}_{dec} = E_{(\mathbf{x}^s, \mathbf{y}^s) \sim (\mathcal{X}^s, \mathcal{Y}^s)} \left\{ -\log p(\mathbf{y}^s | \mathbf{x}^s) \right\}. \tag{18}$$

Directly optimizing $\mathcal{L}_{dec}$ may cause overfitting in source domain, and thus fails to perform well for the shifted target domain. The local-level alignment and global-level alignment modules are introduced to guide the model to learn domain-invariant features between the source and target domain. The learnt robust representations should work effectively on the target domain, where they are also required to be discriminative. Therefore, the global-level adaptation loss $\mathcal{L}_g$ in Eq. 11

and the local-level adaptation loss $\mathcal{L}_l$ in Eq. 17 are combined with the discriminative decoder loss L_{dec} in source domain. The overall objective function of our model is defined as:

$$\mathcal{L}_{ASSDA} = \mathcal{L}_{dec} - \lambda_g \mathcal{L}_g - \lambda_l \mathcal{L}_l, \tag{19}$$

where λ_g and λ_l are weights that control the interaction of losses to achieve better trade-off between the global alignment and local alignment.

E. Optimization

During training, we have three different networks: the text image recognition network T, global domain classifier D_g, and local domain classifier D_l. Let us consider their parameters to be θ_T, θ_{D_g} and θ_{D_l}. In one iteration during training, D_g and D_l are optimized to distinguish the global-level and local-level features from the source domain or target domain, respectively. The text image recognition network T is optimized to extract domain-invariant features that can fool D_g and D_l. In other words, D_g, D_l and T play the minimax game with the objective:

$$\min_{T} \max_{D_g, D_l} \mathcal{L}_{ASSDA} = \mathcal{L}_{dec} - \lambda_g \mathcal{L}_g - \lambda_l \mathcal{L}_l. \tag{20}$$

During training, the parameters of the text image recognition network T is optimized to minimize the objective $\mathcal{L}_{ASSDA}$ in Eq. 19. Simultaneously, the optimization goals for D_g and D_l are opposite to the optimization of the text recognition model T. They are trained by an alternative training way in the concurrent sub-processes:

$$\hat{\theta}_{D_g} \leftarrow \theta_{D_g} - \mu_g \frac{\partial L_g}{\partial \theta_{D_g}}, \tag{21}$$

$$\hat{\theta}_{D_l} \leftarrow \theta_{D_l} - \mu_l \frac{\partial L_l}{\partial \theta_{D_l}}, \tag{22}$$

$$\hat{\theta}_T \leftarrow \theta_T - \mu_t \frac{\partial \mathcal{L}_{dec}}{\partial \theta_T} + \mu_t \lambda_g \frac{\partial \mathcal{L}_g}{\partial \theta_T} + \mu_t \lambda_l \frac{\partial \mathcal{L}_l}{\partial \theta_T}, \tag{23}$$

where μ_g, μ_l, μ_t are the learning rate for optimizing the parameters of the domain classifier D_g, D_l and the text image recognition network T, respectively. For the implementation we use a gradient reversal layer (GRL) [13], whereas the ordinary gradient descent is applied for training the domain classifier. The sign of the gradient is reversed when passing through the GRL layer to optimize the base network. The detailed optimization procedure of our proposed ASSDA is depicted in the Algorithm 1.

Algorithm 1 the Proposed ASSDA Algorithm

```
Input: Labeled source data $(\mathbf{x}_s, \mathbf{y}_s) \in (\mathbf{X}_S, \mathbf{Y}_S)$, unlabeled target
    data $\mathbf{x}_t \in \mathbf{X}_T$; a pre-trained source text recognition model $T$.
Output: The optimized global domain classifier $D_g$, the local domain classifier $D_l$,
    and text recognition model $T$ parameterized by $\hat{\theta}_{D_g}$, $\hat{\theta}_{D_l}$, and $\hat{\theta}_T$, respectively.
Randomly initialize model parameters $\theta_{D_g}$ and $\theta_{D_l}$, and initialize
    $T$ with pre-trained source recognition model.
repeat
    // Updating global domain classifier $D_g$
    $\hat{\theta}_{D_g} \leftarrow \theta_{D_g} - \mu_g \frac{\partial L_g}{\partial \theta_{D_g}}$.
    // Updating local domain classifier $D_l$
    $\hat{\theta}_{D_l} \leftarrow \theta_{D_l} - \mu_l \frac{\partial L_l}{\partial \theta_{D_l}}$.
    // Updating target classifier
    $\hat{\theta}_T \leftarrow \theta_T - \mu_t \frac{\partial \mathcal{L}_{dec}}{\partial \theta_T} + \mu_t \lambda_g \frac{\partial \mathcal{L}_g}{\partial \theta_T} + \mu_t \lambda_l \frac{\partial \mathcal{L}_l}{\partial \theta_T}$
until The objective function in Eq. 20 converges.
    Got $\hat{\theta}_{D_g} = \theta_{D_g}$, $\hat{\theta}_{D_l} = \theta_{D_l}$, and $\hat{\theta}_{C_S} = \theta_{C_T}$.
return The optimized model parameters $\hat{\theta}_{D_g}$, $\hat{\theta}_{D_l}$, and $\hat{\theta}_T$.
```

V. Experiments

Datasets: We conduct extensive experiments to validate the proposed ASSDA on following general recognition benchmark datasets, including three different types of text image, *i.e.*, synthetic text, real scene text, and handwritten text.

- **Synth90k** [33] is the synthetic text dataset. The dataset contains 9 million images generated from a set of 90k common English words. Words are rendered onto natural images with random transformations and effects. Every image in Synth90k is annotated with a groundtruth word. All of the images in this dataset are taken for training.
- **SynthText** [34] is another widely-used synthetic text dataset. The generation process is similar to that of [33]. But unlike [33], SynthText is targeted for text detection. Therefore, words are rendered onto full images. We crop the words using the groundtruth word bounding boxes.
- **IIIT5K-words (IIIT5K)** [35] contains $2,000$ cropped training scene images and $3,000$ cropped test scene text images from the Internet.
- **Street View Text (SVT)** [19] is obtained from Google Street View, where many images are severely corrupted by noise, blur, and low resolution. It contains 257 images for training and 647 images for test.
- **ICDAR-2003 (IC03)** [36] is a camera-captured scene text dataset. Following the protocol used in [2], we discard words that contain non-alphanumeric characters or have less than three characters. Finally, we got 1156 cropped training images and 860 cropped test images.
- **ICDAR-2013 (IC13)** [37] contains 848 images for training and 857 images for evaluation, following the same protocol used in IC-03.
- **ICDAR-2015 Incidental Text (IC15)** [38] is the dataset from the ICDAR 2015 Robust Reading Competition. It focuses on incidental text images, which are taken by a pair of Google Glasses without careful positioning and focusing. Consequently, the dataset contains a lot of irregular text. Images are obtained by cropping the words using the groundtruth word bounding boxes. Following the protocol used in [6], we get 4468 and 1811 cropped images for training and evaluation, respectively.
- **SVT Perspective (SVTP)** [39] consists of 645 images collected from Google Street View, where perspective projections caused by non-frontal viewpoints exist in most of collected images.
- **CUTE80 (CUTE)** [40] is a dataset focusing on curved text, which contains 288 cropped scene text images.
- **IAM** [41] is a handwritten English text dataset, written by 657 different writers. It is partitioned into writer-independent training, validation and test partitions of 6161, 976 and 2915 lines, respectively. That contains

a total of 46945, 7554 and 20306 correctly segmented words in each partition.

Evaluation Metric: In this paper, we evaluate the text recognition model from following three different evaluation metric:

- **Word Prediction Accuracy** is used to evaluate scene text recognition model, following several benchmark [2], [4].
- **WER and CER** are acronyms of Character Error Rate (CER) and Word Error Rate (WER) [3], [42], respectively. They are used to evaluate the text recognition model from character-level and word level, respectively. CER is defined as the Levenstein distance between the predicted and real character sequence of the word. WER denotes the percentage of words improperly recognized. For CER and WER, small values indicate better performance.

Implementation Details: The architecture of the CNN encoder is derived from the ResNet [43], where the detailed structure is illustrated in Table I. Specifically, *"conv_relu_bn"* denotes the convolutional layer followed by batch normalization layer [44] and rectified linear unit (Relu) activation function [45]. And *"max_pool"* and *"residual_block"* denote the max pooling layer and residual block, respectively. As a result, the resolution of feature maps produced by encoder is $H/16 \times W/4$, where the values of H and W are set according to the specific dataset. After the CNN encoder, we use a bi-directional LSTM to capture more context information for attention, and each LSTM has 256 hidden units. And then, we use a LSTM cell with 512 hidden units for the decoder. The global and local domain discriminators are two layered fully connected neural networks. All of our experiments are implemented with Pytorch. For a fair comparison, our model adopts the same protocols following [6]. The complete model is initially pre-trained to minimize the decoding loss of the source training data, and then is fine-tuned to minimize the overall domain adaptation objective with unsupervised target data. More training details could be referenced from our released code.[1]

A. Domain Adaptation on Public Benchmarks

In this scenario, we explore the domain capability of ASSDA on the public scene text recognition benchmarks. They are usually trained on the synthetic data, while tested on the real scene. Domain shifts often happens due to the existence of different noises. Specifically, we adopt the Synth90k [33] and SynthText [34] as the well annotated source data following the protocol in [6]. The real scene text datasets are used as the *unlabeled target data*. To validate the performance of our ASSDA model, we focus on unconstrained text recognition without any language model or lexicon. We also consider a baseline for ASSDA that omits the local-level and global-level alignment modules to switch off the domain adaption process. Baseline model is only trained on the source data.

TABLE I

THE DETAILED STRUCTURE OF THE CNN ENCODER

Layers	[kernel, stride, channel]	Output size
conv_bn_relu	$[3\times3, 1\times1, 32]$	$H\times W$
conv_bn_relu	$[3\times3, 1\times1, 64]$	$H\times W$
max_pool	$[2\times2, 2\times2]$	$H/2\times W/2$
residual_block	$\begin{bmatrix}3\times3, 1\times1, 128\\3\times3, 1\times1, 128\end{bmatrix}\times2$	$H/2\times W/2$
conv_bn_relu	$[3\times3, 1\times1, 128]$	$H/2\times W/2$
max_pool	$[2\times2, 2\times2]$	$H/4\times W/4$
residual_block	$\begin{bmatrix}3\times3, 1\times1, 256\\3\times3, 1\times1, 256\end{bmatrix}\times2$	$H/4\times W/4$
conv_bn_relu	$[3\times3, 1\times1, 256]$	$H/4\times W/4$
max_pool	$[2\times2, 2\times1]$	$H/8\times W/4$
residual_block	$\begin{bmatrix}3\times3, 1\times1, 512\\3\times3, 1\times1, 512\end{bmatrix}\times5$	$H/8\times W/4$
conv_bn_relu	$[3\times3, 1\times1, 512]$	$H/8\times W/4$
residual_block	$\begin{bmatrix}3\times3, 1\times1, 512\\3\times3, 1\times1, 512\end{bmatrix}\times3$	$H/8\times W/4$
conv_bn_relu	$[3\times3, 2\times1, 512]$	$H/16\times W/4$
conv_bn_relu	$[3\times3, 1\times1, 64]$	$H/16\times W/4$

1) Synthetic Text to Regular Text: Table II presents the test results on regular scene text datasets. As our model could effectively utilize available unsupervised data in a unified framework, we perform two experiments, namely ASSDA-single and ASSDA-all, where the only difference is in the training target data setting. For ASSDA-single, we perform domain adaption separably on each single dataset. While for ASSDA-all, we combine all images from different real scene data as target data. Compared to the baseline model, our ASSDA-single and ASSDA-all both can obtain consistent improvement among different datasets. As current available real scene text images are really small, the ASSDA-single get relatively small gains. However, when we combine unsupervised images from different scenes, we could get more benefits. It's mainly attributed to sequence-to-sequence domain adaptation, which is able to learn more domain-invariant features by exploiting the unsupervised data. Furthermore, we investigate the performance of our model among the recent state-of-the-art approaches [4], [17], [21], [32], [47], which are tailored for scene text recognition. We observe that the ASSDA model can achieve comparable results with them.

2) Synthetic Text to Irregular Text: Table III presents the test results on irregular scene text. The performance of our baseline is at an average level for irregular text, although. We can still observe that the ASSDA model achieves significant improvement compared to the baseline without adaptation. It's notable that RARE [20], STAR-Net [21], Char-Net [4] and ASTER [32] target the irregular scene text recognition, which are designed for spatial distortions. Despite of the brilliant performance on irregular scene text, they would not be easily generalized to the different distortions, such as different background and various handwriting styles. In contrast, our method aims to perform sequence-to-sequence domain adaptation to reduce the domain shift, and correspondingly allows us to relieve different distortions using a general framework in different scenarios.

To further investigate the adaptability of the proposed ASSDA, we also compare our model with the finetuning

[1] https://github.com/AprilYapingZhang/Seq2SeqAdapt

TABLE II

SCENE TEXT RECOGNITION ACCURACIES ON REGULAR SCENE TEXT RECOGNITION BENCHMARKS

Model	Reference	IIIT5K	SVT	IC03	IC13
RARE [20]	CVPR 2016	86.2	85.8	93.9	92.6
STAR-Net [21]	CVPR 2016	87.0	86.9	94.4	92.8
R2AM [17]	CVPR 2016	83.4	82.4	92.2	90.2
CRNN [2]	TPAMI 2017	82.9	81.6	93.1	91.1
GRCNN [46]	NIPS 2017	84.2	83.7	93.5	90.9
Char-Net [4]	AAAI 2018	83.6	84.4	91.5	90.8
SSDAN [16]	CVPR 2019	87.6	**88.1**	94.6	**93.8**
STR2019 [6]	ICCV 2019	**87.9**	87.5	**94.9**	93.6
Baseline	ours	87.5	86.7	95.1	92.9
Finetuning	ours	**89.7**	87.3	94.4	**94.3**
ASSDA-single	ours	87.6	87.8	95.5	93.8
ASSDA-all	ours	88.3	**88.6**	**95.5**	93.7

TABLE III

SCENE TEXT RECOGNITION ACCURACIES ON IRREGULAR SCENE TEXT RECOGNITION BENCHMARKS

Model	Reference	IC15	CUTE
RARE [20]	CVPR 2016	74.5	70.4
STAR-Net [21]	CVPR 2016	76.1	71.7
Char-Net [4]	AAAI 2018	60.0	–
ASTER [32]	TPAMI 2019	76.1	**79.5**
SSDAN [16]	CVPR 2019	78.7	73.9
STR2019 [6]	ICCV 2019	77.6	74.0
Baseline	ours	78.1	74.2
Finetuning	ours	**79.7**	74.9
ASSDA	ours	**78.7**	**76.3**

TABLE IV

EVALUATION ON THE SCENE TEXT *v.s.* HANDWRITTEN TEXT TASKS. FOR CER AND WER, SMALL VALUES INDICATE BETTER PERFORMANCE

Methods	ST →HT		HT→ST	
	WER	CER	WER	CER
Baseline	54.30	28.41	89.67	71.04
SSDAN [16]	53.65	27.26	86.57	67.25
ASSDA	**43.78**	**19.96**	**84.94**	**62.48**

method. The ASSDA focuses on leveraging the *unsupervised target data* to reduce the domain shift. In contrast, the fine-tuning method must use the labeled training splits of the target dataset. However, there may be no access to the labeled training splits in some target domain, especially in real applications. To address this issue, we combine the available training images from different real scene data as target data. For a fair comparison, we adopt the same combined training target data as used in ASSDA. As shown in Table II and Table III, the fine-tuning model (Finetuning) not always performs better than the unsupervised domain adaptation model. It implies that the proposed ASSDA with unsupervised domain adaptation may be able to learn more robust features than the supervised fine-tuning in some specific scenes.

B. Domain Adaptation on Specific Cross-Domain Tasks

We further evaluate the ASSDA on two different cross-domain tasks: *scene text v.s. handwritten text* and *real regular text to real irregular text*, to explore the model generalization.

1) Scene Text v.s. Handwritten Text: We evaluate our algorithm on the 2 cross domain adaptation experiments: Scene Text → Handwritten Text (ST → HT), Handwritten Text → Scene Text (HT → ST), using the training set only during training process and evaluating on the standard test sets. The token "→" means the direction from the source domain to the target. As shown in Fig. 1, the scene text has many differences with handwritten text. Specifically, the baseline model is firstly trained by supervised source data. And then we employ ASSDA to take advantage of some unsupervised target to finetune the model. For the analysis, we evaluate the model on both CER and WER. As shown Tabel IV, the baseline model has a poor performance on cross-domain target data. When we employ ASSDA to learn domain invariant features, we could get some improvement in both character level and word level. Compared to SSDAN, the extended ASSDA made a big progress. Especially on the ST →HT setting, we could get nearly 10% improvement. Although the improvement is relatively small on the more difficult setting HT→ST, it could also validate the generalization of our model.

2) Real Regular Text to Real Irregular Text: We explore the adaptability of the ASSDA towards perspective distortions when the available source dataset is small. Following the protocol used in [31], we use the real regular scene text and real irregular scene text as the source domain and target domain, respectively. Specifically, the real regular scene text dataset is composed of the training data from IC13, IIIT5K, and SVT, whose total number of data is small. We denote the combined source data as "COMB". While the curved dataset CUTE and perspective dataset SVTP are used as two target datasets. The Table V shows that non-sequential domain adaptation methods ADDA [14] and CyCADA [26] are not sufficiently robust to reduce the domain shift in sequence-like text images. However, our model could get improvement even when the available source data is small. We observe that GA-DAN [31] shows better performance in Table V. One possible reason may be that GA-DAN could be well-tuned in two-stage frameworks. While our ASSDA is trained in a unified framework via an end-to-end way.

C. Ablation Study

In this part, we design several variants of our model to validate the contributions of different components.

TABLE V

EVALUATION ON REAL REGULAR TEXT TO REAL IRREGULAR TEXT TASKS

Model	COMB→CUTE	COMB→SVTP
ADDA [14]	32.1	45.6
CyCADA [26]	32.2	43.6
GA-DAN [31]	43.1	51.7
SSDAN [16]	33.8	45.9
Baseline	32.4	45.7
ASSDA	**38.3**	**47.1**

Two variants are provided as baselines: the baseline model without spatial normalization network (Baseline *w/o* $\mathcal{N}$), and the baseline model with spatial normalization network (Baseline). Both baselines are trained only on source data without any adaptation. Furthermore, more variants based on the baseline models are adapted with unsupervised target data via Global-level Alignment (GA, only with D_g), Local-level Alignment (LA, only with D_l), and multiple-level alignment (ASSDA, with both D_g and D_l), respectively.

To sufficiently investigate the effect of different components, we conduct the experiments on two different cross-domain tasks: from Synthetic Scene Text to Real Scene text (Syn→Real) and from Synthetic Scene Text to Handwritten Text (ST→HT). Regarding that the real scene text includes regular text and irregular text, we denote the adaptation from synthetic text to regular text and irregular text as ST→RT and ST→IT, respectively. The experimental results are reported in Table VII and Table VI.

1) The Effect of Spatial Normalization Network: The model *Baseline w/o* $\mathcal{N}$ and model *Baseline* demonstrate that the baseline text recognition model with and without spatial normalization network $\mathcal{N}$, respectively. It can be observed that the text recognition model can be benefited from introducing $\mathcal{N}$ in Table VII and Table VI. But we notice that the gain is limited when the $\mathcal{N}$ is applied to non-geometric distortions such as the setting in Table VII. Furthermore, we investigate how the spatial normalization network $\mathcal{N}$ influences the domain adaptation. As shown in the Table VI, the comparison pairs (*Baseline w/o* $\mathcal{N}$, *GA w/o* $\mathcal{N}$) and (*GA w/o* $\mathcal{N}$, *GA*) show that the $\mathcal{N}$ plays an import role in global-level alignment, while the comparison between *LA w/o* $\mathcal{N}$ and *LA* shows that the local-level alignment get very limited benefits from $\mathcal{N}$. However, the comparison pair (*ASSDA w/o* $\mathcal{N}$, ASSDA) validates that the $\mathcal{N}$ does boost the joint global-level and local-level alignment.

2) Global-Level Alignment v.s. Local-Level Alignment: We observe that the global-level alignment can help the model to get more robust performance, after we diminish the geometric shift by $\mathcal{N}$. As shown in Table VII and Table VI, the adaptation model with both global-level and local-level alignment (ASSDA), *i.e.* with D_g and D_l, performs better than the adaptation model only with local-level alignment (*LA*). It validates that the global-level alignment can facilitate the sequence-to-sequence domain adaptation. Furthermore, we also observe that the local-level alignment plays a vital role in sequence-to-sequence domain adaption from the comparison pairs (*Baseline w/o* $\mathcal{N}$, *LA w/o* $\mathcal{N}$) and (*Baseline*, *LA*) in Table VII and VI And the comparison pairs between the local-level alignment and global-level alignment, *i.e.* (*LA w/o* $\mathcal{N}$, *GA w/o* $\mathcal{N}$) and (*LA*, *GA*), validate that the fine-grained character-level knowledge transfer between the source and target sequence data is more effective and robust than the global-level alignment. The comparison between (*LA w/o* $\mathcal{N}$, *LA*) also shows that $\mathcal{N}$ the has a negligible effect on local-level alignment.

D. Algorithm Analysis

In this scenario, we firstly investigate the effect of different alignment strategy, and then analyze the parameter sensitiveness. Furthermore, we visualize the attention results, and explore the effect of domain adaptation via feature visualization. Finally, we make some discussions on the limitation of ASSDA.

1) The Effect of Alignment Strategy: We conduct experiments for investigating the effect of the domain alignment strategy. In the ASSDA, the source and target distributions are aligned in an adversarial manner, where λ_g and λ_l control the alignment process. The alignment process is crucial to obtain the effective information to exploit the unlabeled target data. In our experiment, we set the value of λ_g equal to the value of λ_l. We explore two different dynamic alignment strategy to ensure the reliability of learning progress. Specifically, the defined two alignment schemes are shown in Fig. 5, and they denote the changing value of λ_g and λ_l during the training process, where $i = iteration/total\_iterations$.

Since our designed alignment strategy aims to dynamically mine the character which are more likely to be a valid character, to verify its effectiveness, we design a fixed alignment strategy, which learns the alignment with the fixed value of λ_g and λ_l, as the competitor. The results are reported in Table VIII. It can be observed that under the same setting of experiment, The progressive strategy achieves better performance compared to the fixed alignment strategy. Specifically, the Progressive-2 alignment strategy yields an improvement of 10.52% compared with the fixed alignment strategy.

2) Parameter Sensitive Analysis: We evaluate the sensitiveness of the hyper-parameter p_c in the Eq. 12. Here, we conduct the experiments on the ST → HT task. Specifically, we explore the different p_c from $\{0, 0.1, 0.2, 0.4, 0.8\}$, respectively. The evaluation is conducted by changing one parameter while keeping the other hyper-parameters fixed. The p_c in the gate function of Eq. 12 decides whether an attended feature performs domain adaptation or not. Specifically, if the probability that the current feature vector belongs to a valid character is larger than p_c, the vector will be performed domain adaptation, otherwise, it will be neglected as a noise. From other perspective, if $p_c = 0$, the gate function will not work, which means performing sequence domain adaptation on character-level feature without any guidance. While p_c is too large, the gate function will be too strict to select enough valid features. Fig. 6 shows different gains of p_c values. The results experimentally prove that the gate function is important to the overall performance.

TABLE VI

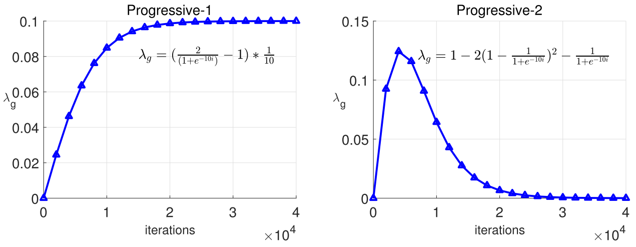
COMPONENT ANALYSIS ON THE SYNTH→REAL SETTING. FOR ACCURACY, BIG VALUES INDICATE BETTER PERFORMANCE

Model	Component			ST →RT				ST→IT	
	$\mathcal{N}$	D_l	D_g	IIIT5K	SVT	IC03	IC13	IC15	CUTE
Baseline *w/o* $\mathcal{N}$	✗	✗	✗	86.66	85.62	94.76	91.13	**74.82**	75.26
GA *w/o* $\mathcal{N}$	✗	✗	✓	86.43	84.85	95.00	92.18	73.94	**76.66**
LA *w/o* $\mathcal{N}$	✗	✓	✗	**86.77**	85.47	**95.00**	92.07	74.43	76.31
ASSDA *w/o* $\mathcal{N}$	✗	✓	✓	86.57	**85.94**	94.54	**92.53**	74.43	75.61
Baseline	✓	✗	✗	87.40	87.02	95.12	92.88	78.07	74.22
GA	✓	✗	✓	87.83	87.02	95.12	93.58	77.86	75.61
LA	✓	✓	✗	87.83	87.48	95.23	93.58	**78.91**	76.31
ASSDA	✓	✓	✓	**88.26**	**88.56**	**95.46**	**93.70**	78.69	**76.31**

TABLE VII

COMPONENT ANALYSIS ON THE ST →HT SETTING. FOR CER AND WER, SMALL VALUES INDICATE BETTER PERFORMANCE

Model	$\mathcal{N}$	D_l	D_g	WER	CER
Baseline*w/o* $\mathcal{N}$	✗	✗	✗	56.29	30.08
GA *w/o* $\mathcal{N}$	✗	✗	✓	56.59	28.89
LA *w/o* $\mathcal{N}$	✗	✓	✗	48.97	23.83
ASSDA *w/o* $\mathcal{N}$	✗	✓	✓	**44.67**	**20.60**
Baseline	✓	✗	✗	54.30	28.41
GA	✓	✗	✓	52.41	25.73
LA	✓	✓	✗	48.58	22.99
ASSDA	✓	✓	✓	$\mathbf{43.78}^{\downarrow \mathbf{12.50}}$	$\mathbf{19.96}^{\downarrow \mathbf{10.12}}$



(a) Progressive-1

(b) Progressive-2

Fig. 5. The scheme of alignment strategy.

TABLE VIII

ABLATION STUDIES FOR PROGRESSIVE ADAPTATION. FOR CER AND WER, SMALL VALUES INDICATE BETTER PERFORMANCE

Method	WER	CER
Source-only	54.30	28.41
Fixed	50.69	30.53
Progressive-1	45.25	21.44
Progressive-2	$\mathbf{43.78}^{\downarrow \mathbf{10.52}}$	$\mathbf{19.96}^{\downarrow \mathbf{8.45}}$

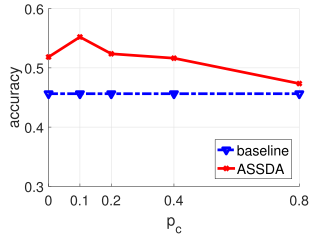


Fig. 6. The effect of model parameters p_c.

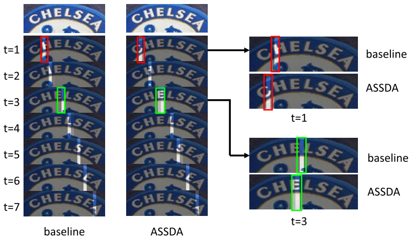


Fig. 7. Attention visualization on one irregular scene text from CUTE. The first column and the second column denote the attention results without and with domain adaptation, respectively. The last column is to shown the difference of the attention maps between the baseline model and ASSDA.

3) Visualization on the Attention Result: In this scenario, we visualized the attention result at each time step. As shown in Fig. 7, we randomly choose one irregular text image from CUTE80. It can be seen that the model could focus on the most relevant areas of one character at one specific time. Consequently, we could get the fine-grained character-level information, and then perform the character-level domain adaption. More interestingly, we find the ASSDA model can learn more precise alignment, according to the two cases of the last column in Fig. 7. These results again validate the effectiveness of ASSDA.

4) Visualization on the Feature Distribution: To demonstrate the domain adaptation effectiveness on different feature level, we use the t-SNE tool to visualize the feature distribution of different domains in the task ST→HT. Specifically, we visualize the domain distribution on global-level features from the Baseline, global-level features from the ASSDA, local-level features from the Baseline, and local-level features from the ASSDA, respectively. As shown in the left part of Fig. 8, we observe clear domain shifts between the source and target domain, when the features are extracted from the baseline model without any adaptation. While we can see that the adapted features from ASSDA model are confused at both global-level and local-level in the right part of Fig. 8. It reveals that the proposed ASSDA has the ability to reduce the domain shift at the different feature level. Here, the global-level features are extracted from the whole sequence-like word image, and the local-level features denote the fine-grained

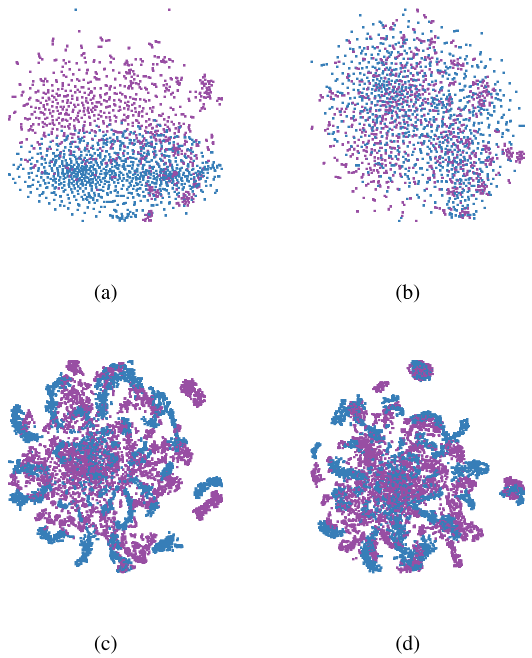


Fig. 8. The t-SNE visualizations of domain distribution on (a) global-level features from the Baseline, (b) global-level features from the ASSDA, (c) local-level features from the Baseline, and (d) local-level features from the ASSDA. Specifically, the feature distribution in the left and right are from the Baseline and ASSDA, respectively. While, the top and bottom distribution indicate the coarse global-level and local-level distributions, respectively. In each image, the purple and blue dots denote the features from the source domain and target domain, respectively.

character-level features. We notice that the coarse global-level features (Fig. 8(a)(b)) cannot be separated according to character class information, but character-level features are class-separable. We think the phenomenon is reasonable, as the coarse global-level features are the combination of variable-length sequence-like images rather than one specific character. As a result, we infer that the fine-grained local character-level adaptation plays a vital role in boosting the adaptability of the recognition model.

5) Limitation of the ASSDA: In this scenario, we discuss the limitation of the proposed model. (1) The proposed ASSDA model fails to consider the open-set problem. It's noted that the proposed ASSDA has one assumption that the source and target domain share the same label space, and the domain shifts are only from visual differences. (2) The proposed model tries to align the local character-level distributions between two domains based on the character information that is captured by attention mechanism automatically. Therefore, the performance of ASSDA may be limited due to the inaccurate character region awareness. (3) Although, we introduce a spatial normalization network, the ASSDA will fail when the curve angle is too large, due to the complicated distortions disturbed the capture of character-level information. Those observations imply that robust scene text recognition still remains a challenging problem waiting for solutions.

VI. Conclusion

In this paper, we present a novel ASSDA model for robust text image recognition, which bridges the sequence-like text image recognition and domain adaptation. It's capable of aligning the cross-domain distribution on both global-level and local-level. It could be generalized to reduce different types of domain shifts, which include appearances and handwriting style, *etc.* Comprehensive experimental results on several datasets and extensive analyses have demonstrated the effectiveness of our algorithm. An interesting open issue for future research is to further adjust ASSDA framework to better deal with various sequence domain shift.

References

[1] P. He, W. Huang, Y. Qiao, C. C. Loy, and X. Tang, "Reading scene text in deep convolutional sequences," in *Proc. AAAI*, vol. 16, 2016, pp. 3501–3508.

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

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

[4] W. Liu, C. Chen, and K.-Y. K. Wong, "Char-Net: A character-aware neural network for distorted scene text recognition," in *Proc. AAAI Conf. Artif. Intell.*, New Orleans, LA, USA, 2018, pp. 7154–7161.

[5] Z. Cheng, Y. Xu, F. Bai, Y. Niu, S. Pu, and S. Zhou, "AON: Towards arbitrarily-oriented text recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 5571–5579.

[6] J. Baek *et al.*, "What is wrong with scene text recognition model comparisons? Dataset and model analysis," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4715–4723.

[7] V. K. Kurmi, S. Kumar, and V. P. Namboodiri, "Attending to discriminative certainty for domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 491–500.

[8] J. Zhang, W. Li, P. Ogunbona, and D. Xu, "Recent advances in transfer learning for cross-dataset visual recognition: A problem-oriented perspective," *ACM Comput. Surveys*, vol. 52, no. 1, pp. 1–38, Feb. 2019.

[9] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," 2015, *arXiv:1502.02791*. [Online]. Available: http://arxiv.org/abs/1502.02791

[10] B. Yang, A. J. Ma, and P. C. Yuen, "Domain-shared group-sparse dictionary learning for unsupervised domain adaptation," in *Proc. AAAI*, 2018, pp. 7453–7460.

[11] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 443–450.

[12] J. Zhuo, S. Wang, W. Zhang, and Q. Huang, "Deep unsupervised convolutional domain adaptation," in *Proc. 25th ACM Int. Conf. Multimedia*, Oct. 2017, pp. 261–269.

[13] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, pp. 1180–1189.

[14] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, vol. 1, no. 2, p. 4.

[15] C. Chen *et al.*, "Progressive feature alignment for unsupervised domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 627–636.

[16] 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 *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2740–2749.

[17] C.-Y. Lee and S. Osindero, "Recursive recurrent nets with attention modeling for OCR in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2231–2239.

[18] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Deep features for text spotting," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 512–528.

[19] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 1457–1464.

[20] B. Shi, X. Wang, P. Lyu, C. Yao, and X. Bai, "Robust scene text recognition with automatic rectification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4168–4176.

[21] W. Liu, C. Chen, K.-Y. Wong, Z. Su, and J. Han, "STAR-Net: A SpaTial attention residue network for scene text recognition," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, vol. 2, 2016, p. 7.

[22] M. Jaderberg *et al.*, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.

[23] W. Li, L. Chen, D. Xu, and L. Van Gool, "Visual recognition in RGB images and videos by learning from RGB-D data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 2030–2036, Aug. 2018.

[24] W. Li, Z. Xu, D. Xu, D. Dai, and L. Van Gool, "Domain generalization and adaptation using low rank exemplar SVMs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 5, pp. 1114–1127, May 2018.

[25] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, vol. 1, no. 2, p. 7.

[26] J. Hoffman *et al.*, "CyCADA: Cycle-consistent adversarial domain adaptation," in *Proc. ICML*, 2018, pp. 1989–1998.

[27] W. Zhang, D. Xu, W. Ouyang, and W. Li, "Self-paced collaborative and adversarial network for unsupervised domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Dec. 25, 2019, doi: 10.1109/TPAMI.2019.2962476.

[28] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. Van Gool, "Domain adaptive faster R-CNN for object detection in the wild," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3339–3348.

[29] Z. He and L. Zhang, "Multi-adversarial faster-RCNN for unrestricted object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6668–6677.

[30] S. Azadi, M. Fisher, V. Kim, Z. Wang, E. Shechtman, and T. Darrell, "Multi-content GAN for few-shot font style transfer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, p. 13.

[31] F. Zhan, C. Xue, and S. Lu, "GA-DAN: Geometry-aware domain adaptation network for scene text detection and recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9105–9115.

[32] B. Shi, M. Yang, X. Wang, P. Lyu, C. Yao, and X. Bai, "ASTER: An attentional scene text recognizer with flexible rectification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 9, pp. 2035–2048, Sep. 2019.

[33] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," 2014, *arXiv:1406.2227*. [Online]. Available: http://arxiv.org/abs/1406.2227

[34] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2315–2324.

[35] A. Mishra, K. Alahari, and C. Jawahar, "Scene text recognition using higher order language priors," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2012, pp. 1–12.

[36] S. M. Lucas *et al.*, "ICDAR 2003 robust reading competitions: Entries, results, and future directions," *Int. J. Document Anal. Recognit. (IJDAR)*, vol. 7, nos. 2–3, pp. 105–122, Jul. 2005.

[37] D. Karatzas *et al.*, "ICDAR 2013 robust reading competition," in *Proc. 12th Int. Conf. Document Anal. Recognit. (ICDAR)*, Aug. 2013, pp. 1484–1493.

[38] D. Karatzas *et al.*, "ICDAR 2015 competition on robust reading," in *Proc. 13th Int. Conf. Document Anal. Recognit. (ICDAR)*, Aug. 2015, pp. 1156–1160.

[39] T. Q. Phan, P. Shivakumara, S. Tian, and C. L. Tan, "Recognizing text with perspective distortion in natural scenes," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 569–576.

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

[41] U.-V. Marti and H. Bunke, "The IAM-database: An English sentence database for offline handwriting recognition," *Int. J. Document Anal. Recognit.*, vol. 5, no. 1, pp. 39–46, Nov. 2002.

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

[43] Z. Cheng, F. Bai, Y. Xu, G. Zheng, S. Pu, and S. Zhou, "Focusing attention: Towards accurate text recognition in natural images," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5086–5094.

[44] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Int. Conf. Mach. Learn.*, vol. 37, 2015, pp. 448–456.

[45] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 807–814.

[46] J. Wang and X. Hu, "Gated recurrent convolution neural network for OCR," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 335–344.

[47] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep structured output learning for unconstrained text recognition," 2014, *arXiv:1412.5903*. [Online]. Available: http://arxiv.org/abs/1412.5903