Noisy-Aware Unsupervised Domain Adaptation for Scene Text Recognition

Xiao-Qian Liu, Peng-Fei Zhang, Xin Luo, Zi Huang, *Senior Member, IEEE*, and Xin-Shun Xu, *Senior Member, IEEE*

***Abstract*— Unsupervised Domain Adaptation (UDA) has shown promise in Scene Text Recognition (STR) by facilitating knowledge transfer from labeled synthetic text (source) to more challenging unlabeled real scene text (target). However, existing UDA-based STR methods fully rely on the pseudo-labels of target samples, which ignores the impact of domain gaps (inter-domain noise) and various natural environments (intra-domain noise), resulting in poor pseudo-label quality. In this paper, we propose a novel noisy-aware unsupervised domain adaptation framework tailored for STR, which aims to enhance model robustness against both inter- and intra-domain noise, thereby providing more precise pseudo-labels for target samples. Concretely, we propose a reweighting target pseudo-labels by estimating the entropy of refined probability distributions, which mitigates the impact of domain gaps on pseudo-labels. Additionally, a decoupled triple-P-N consistency matching module is proposed, which leverages data augmentation to increase data diversity, enhancing model robustness in diverse natural environments. Within this module, we design a low-confidence-based character negative learning, which is decoupled from high-confidence-based positive learning, thus improving sample utilization under scarce target samples. Furthermore, we extend our framework to the more challenging *Source-Free* UDA (SFUDA) setting, where only a pre-trained source model is available for adaptation, with no access to source data. Experimental results on benchmark datasets demonstrate the effectiveness of our framework. Under the SFUDA setting, our method exhibits faster convergence and superior performance with less training data than previous UDA-based STR methods. Our method surpasses representative STR methods, establishing new state-of-the-art results across multiple datasets.**

***Index Terms*— Text recognition, domain adaptation, entropy, noisy-aware, consistency regularization.**

I. Introduction

READING text has always been a popular research topic in computer vision. Deep learning-based methods have made remarkable progress in scene text recognition (STR) [1], [2], [3]. It focuses on reading text from scene images and transcribes it into computer-understandable information. The valuable information in scene text images is vital in various downstream tasks, including image-text retrieval, text-based visual question answering, and key information extraction.

Due to the limited availability of annotated real scene text, current STR methods typically rely on large-scale synthetic text for training. These methods are then directly evaluated on real scene text without fine-tuning [8], [9]. Even though these methods have improved accuracy, inherent noise still interferes with model performance. The 'noise' manifests in two perspectives: inter-domain noise, which arises from domain gaps between the synthetic and real text, and intra-domain noise, which stems from various environments, such as lighting variations, occlusion, shadows, and perspective distortions. Both inter- and intra-domain noises may potentially degrade the quality of pseudo-labels generated by pre-trained source models, thus adversely affecting the recognition performance on target real scene text.

Considering the relative ease of collecting unlabeled text, many researchers have gravitated to semi-supervised learning [10], which seeks to boost recognition models by effectively leveraging labeled synthetic text and unlabeled real scene text. Among these semi-supervised STR methods, unsupervised domain adaptation (UDA) and self-supervised learning are two popular ways for their representation effectiveness. Specifically, UDA-based STR methods [5], [6], [7], [11] reduce domain discrepancies by aligning features from labeled source synthetic text with those from unlabeled target real text, aiming to extract domain-invariant representations and eliminate the effects of inter-domain noise. Similarly, self-supervised STR methods [12], [13], [14], [15], [16] use unlabeled real text for self-supervised optimization followed by fine-tuning on the labeled synthetic text for downstream tasks, which attempt to exploit the intrinsic properties of unlabeled real text to improve generalization under intra-domain noise. However, both UDA-based and self-supervised methods tend to fully trust pseudo-labels without assessing their certainty and reliability, leading to potential instability during model training due to domain gaps, as illustrated in Fig. 1. Meanwhile, when the target data is real scene text, UDA-based STR methods often ignore the impact of various real environmental disturbances. Directly utilizing only a limited amount of real text for adaptation hinders the generalization and robustness of the model in various environments. Therefore, thoroughly exploring the intrinsic properties of

Received 13 November 2023; revised 17 August 2024 and 30 September 2024; accepted 28 October 2024. Date of publication 12 November 2024; date of current version 18 November 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62172256, Grant 62202278, and Grant 62202272; and in part by the Natural Science Foundation of Shandong Province under Grant ZR2019ZD06 and Grant ZR2020QF036. The associate editor coordinating the review of this article and approving it for publication was Prof. Yizhou Yu. *(Corresponding author: Xin-Shun Xu.)*

Xiao-Qian Liu, Xin Luo, and Xin-Shun Xu are with the School of Software, Shandong University, Jinan 250101, China (e-mail: jlrxqxq370322@126.com; luoxin.lxin@gmail.com; xuxinshun@sdu.edu.cn).

Peng-Fei Zhang and Zi Huang are with the School of Electrical Engineering and Computer Science, The University of Queensland, Brisbane, QLD 4072, Australia (e-mail: mima.zpf@gmail.com; huang@itee.uq.edu.au).

Digital Object Identifier 10.1109/TIP.2024.3492705

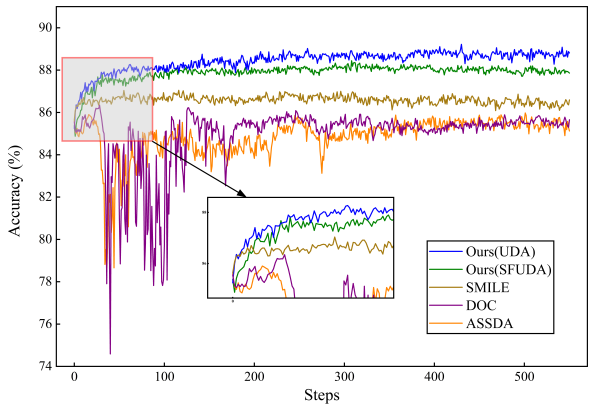


Fig. 1. Training process of UDA-based text recognition methods under the unified TRBA [4] backbone. In the SFUDA, the source data is unavailable, only the pre-trained source model. Compared to the UDA-based methods, SMILE [5], DOC [6], and ASSDA [7], Ours(SFUDA) model is trained with fewer training samples without any supervised information.

the real text remains a critical problem in enhancing STR performance.

To alleviate the above issues, we propose a novel noise-aware unsupervised domain adaptation framework tailored for STR, including reweight pseudo-labels via uncertainty estimation and decoupled triple-P-N consistency matching modules to eliminate the effect of inter- and intra-domain noise. Specifically, by leveraging character representations and initial pseudo-labels generated by the pre-trained source model, the pseudo-labels are reweighted via entropy uncertainty estimation. Since similar semantic characters tend to be close in the feature space, the neighboring relationships can be used to assess the reliability of pseudo-labels. Guided by the neighboring knowledge, the anchor probability distribution is refined. Subsequently, a negative exponential function is adopted to estimate the entropy of the refined probability distribution, where higher entropy indicates greater uncertainty and lower importance of pseudo-labels. Pseudo-labels with high uncertainty are penalized by optimizing the reweighted pseudo-labels with entropy minimization, effectively enhancing their reliability under the inter-domain noise.

Additionally, a decoupled triple-P-N consistency matching module is proposed to enhance the model robustness against intra-domain noise. Technically, due to the unknown real environmental noise, various data augmentations are employed to increase the diversity of target samples. These augmented samples are then subjected to consistency matching to ensure consistent predictions across different noise disturbances. Moreover, we design a character negative learning (NL) strategy to mitigate the impact of noisy pseudo-labels. This strategy decouples the high-confidence-based positive learning (PL) and low-confidence-based negative learning. The character NL is performed across multiple low confidences rather than a single one, thereby improving sample utilization under scarce target samples.

Existing UDA-based STR methods typically require labeled source synthetic text for adaptation. However, accessing source data may not be feasible in some cases involving data privacy concerns or challenges related to large-scale data storage and transmission. To tackle this, we further explore a more demanding UDA setting known as *Source-Free* UDA (SFUDA), where adaptation performs without source data, relying solely on a pre-trained source model. This setting is particularly beneficial when obtaining source data is restricted, but a pre-trained model can be utilized. As depicted in Fig. 1, our SFUDA method *Ours(SFUDA)* uses fewer training data, converges faster, and achieves superior performance compared to traditional UDA-based methods. Moreover, the performance can be further enhanced under the standard UDA setting. Overall, the SFUDA model offers a more flexible and efficient solution for STR in scenarios where accessing source data is challenging. By leveraging the pre-trained source model, our SFUDA model demonstrates promising training efficiency and recognition accuracy results, making it a valuable addition to UDA-based STR methods.

To summarize, the main contributions are as follows:

- We propose to reweight pseudo-labels by assessing the entropy of the refined probability distribution by neighboring knowledge, thus reducing the impact of domain gap noise on pseudo-labels.
- A decoupled triple-P-N consistency matching module is proposed, where the designed character negative learning enables the decoupling of high-confidence-based PL and low-confidence-based NL, thereby improving sample utilization under scarce real text. This is the first effort to introduce NL to character sequence recognition tasks.
- Experiments on seven benchmark datasets conducted under SFUDA and UDA settings demonstrate the superiority of our proposed framework. Our SFUDA model achieves faster convergence and better performance with fewer training data than existing UDA-based methods. Additionally, our method establishes new state-of-the-art (SOTA) results on multiple datasets in the UDA setting.

II. Related Work

A. Deep Learning-Based Text Recognition

Deep learning-based text recognition methods [17], [18] can be categorized into three types based on their decoders: CTC, RNN, and transformer decoders. Representative CTC decoder methods [19], [20], such as CRNN [21], have limitations in recognizing irregular text. In contrast, RNN decoder methods [22], [23] have gained popularity due to their proper localization of characters in images, making them suitable for handling irregular text. For instance, ASTER [24] utilizes a spatial transform network and an RNN decoder for irregular text recognition. Recently, transformer decoder methods [25], [26], [27] have emerged as a promising direction to extract linguistic information. For example, SRN [28] uses a language model to learn the relationship between each character. ABINet [29] explicitly models linguistic rules by a stronger bi-directional language model. More recently, CLIP4STR [30] based on large language models has also been proposed and performs satisfactorily on various text tasks. However, these approaches overlook the domain gaps between synthetic and real text.

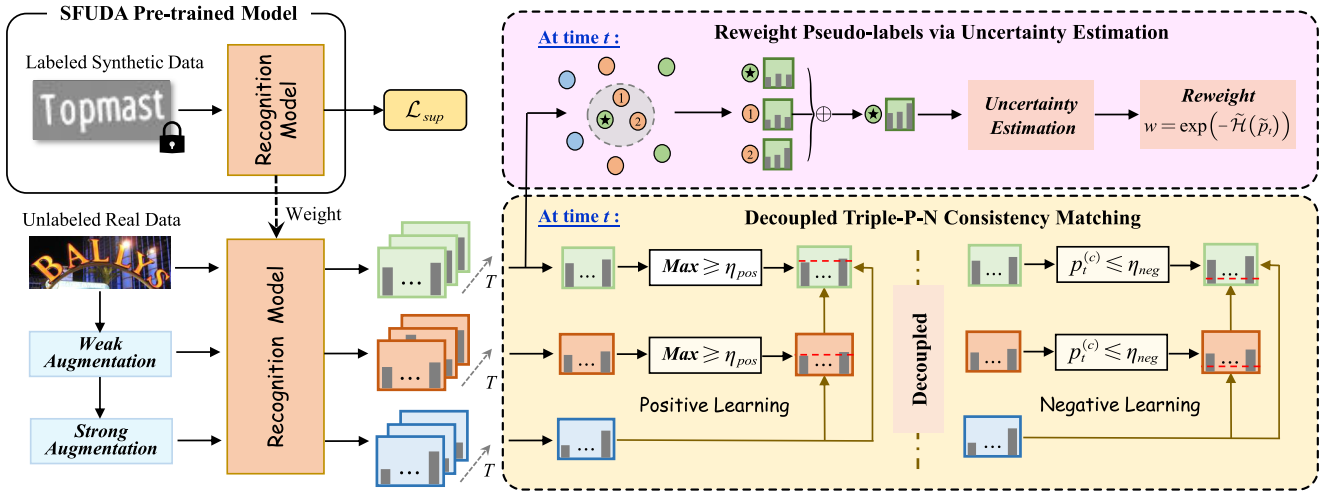


Fig. 2. The pipeline of our proposed framework under the SFUDA setting. Only the pre-trained source model and unlabeled real data are involved in the SFUDA setting. For inter-domain noise, we propose to *Reweight Pseudo-labels via Uncertainty Estimation* of target samples. The anchor probability distribution is refined by neighboring knowledge. Then, the pseudo-labels are reweighted through entropy uncertainty estimation, where the entropy reflects the uncertainty of pseudo-labels. For intra-domain noise, a *Decoupled Triple-P-N Consistency Matching* module is proposed by utilizing various data augmentation to increase data diversity and enhance model robustness. Positive learning is performed on the maximum confidence, while negative learning acts on multiple confidences lower than the pre-defined threshold.

B. Unsupervised Domain Adaptation for STR

UDA [31], [32], [33] aims to reduce domain gaps between labeled source data and unlabeled target data. Maximum mean discrepancy (MMD) [34], [35] and correlation alignment (CORAL) [36], [37] are commonly used divergence measures. Adversarial learning [38] is also used to mitigate inter-domain discrepancies, *e.g.*, ADVENT [39] is based on adversarial learning for entropy minimization optimization. Recently, some source-free adaptation methods [40], [41] have been proposed, adapting to the target domain using only the pre-trained source model and unlabeled target data. U-SFAN [42] proposes quantifying the uncertainty in the source predictions by employing a Laplace approximation. USFDA [43] requires artificially generated negative samples in the source training stage for the model to detect out-of-distribution (OOD) samples. The main limitation is that it fails to process sequential tasks and capture their contextual semantics.

Inspired by these methods, UDA is introduced to exploit unlabeled real scene text for STR performance improvement [5], [7], [44], where each scene is considered a distinct domain. For example, ASSAN [7] and DOC [6] are UDA-based STR methods, which utilize global and local adversarial learning to extract domain-invariant features. SMILE [5] and CADA [11] employ entropy minimization to optimize unlabeled target samples. However, these UDA-based methods overlook the noise in pseudo-labels and treat all target pseudo-labels equally without estimating their uncertainty.

C. Self-Supervised Learning for STR

Pseudo-labeling and consistency regularization (CR) are two mainstream techniques for self-supervised learning. Specifically, pseudo-labeling consists in using pseudo-labels predicted by the pre-trained model as self-supervision. It is introduced in [6], [7], and [11] to utilize real-world unlabeled text images, which adopt a fixed threshold to filter noisy pseudo-labels. Seq-UPS [45] further extends pseudo-labeling to uncertainty-based data selection, but the pseudo-labels are sequence-level. Consistency regularization assumes the model should produce consistent predictions when fed perturbed versions of the same image [46]. Zheng et al. [47] proposed a CR-based framework that addresses character misalignment. SemiMTR [48] is a multimodal text recognizer fine-tuned via a sequential, character-level, and CR between weak and strong augmented views. Nevertheless, these self-supervised methods ignore the effect of inter- or intra-domain noise on the pseudo-labels of unlabeled data.

III. Our Method

This work focuses on addressing the STR task from a UDA perspective. Let $\mathcal{D}^l = \{(x_i^l, y_i^l)\}_{i=1}^{N^l}$ denote the labeled source synthetic text, where x^l is an image containing a word, and $y^l = \{y_1^l, y_t^l, \cdots, y_T^l\}$ represents the corresponding character label. T is the pre-defined maximum decoding length. Let $\mathcal{D}^u = \{(x_i^u)\}_{i=1}^{N^u}$ be the unlabeled target data (default is real scene text). Images x^l and x^u originate from different distributions but share a common label space. Notably, labels for the target data are only obtained during the evaluation phase. In the SFUDA setting, only the pre-trained source model trained on $\mathcal{D}^l$ is accessible, and the source data $\mathcal{D}^l$ cannot be utilized for adaptation. Unlike conventional UDA or SFUDA methods primarily for classification tasks, text recognition involves sequence recognition, where each image represents a word composed of multiple characters with contextual semantics.

Under the SFUDA setting, our proposed framework depicted in Fig. 2 comprises two key modules: reweight pseudo-labels via uncertainty estimation and decoupled triple-P-N consistency matching. The recognition model is initially warmed up using labeled source synthetic text through supervised loss $\mathcal{L}_{sup}$. The pre-trained source model generates

pseudo-labels for unlabeled target images during adaptation. However, due to the presence of both inter- and intra-domain noise, these pseudo-labels may contain inaccuracies. Consequently, our primary objective is to enhance the quality of these noisy pseudo-labels iteratively. This refinement process facilitates knowledge transfer from the source domain to the target domain, ultimately enabling the recognizer to perform effectively on the unlabeled target data $\mathcal{D}^u$.

In the following sections, we first introduce the related preliminaries. Then, the reweight pseudo-labels via uncertainty estimation and decoupled triple-P-N consistency matching modules are described in detail, respectively. Finally, we summarize the overall loss.

A. Preliminaries

1) Baseline Model: Due to the proper text localization capabilities, we select the RNN decoder-based method as our baseline STR model. This model comprises an encoder $\mathcal{F}$, an attentional block, and an RNN decoder $\mathcal{G}$. Given an input image x, we obtain the sequence features $\mathcal{F}(x) = [f_1, f_2, \cdots, f_T] \in \mathbb{R}^{T\times D}$, where D represents the feature dimension. At time t, the decoder $\mathcal{G}$ updates the hidden state s_t based on three factors: (a) previous internal state s_{t-1} of decoder RNN, (b) the character y_{t-1} predicted (or label) at time $t-1$, and (c) a glimpse vector g_t representing the most relevant part of $\mathcal{F}$ for predicting y_t,

$$(o_t, s_t) = \mathcal{G}\left(s_{t-1}; [E(y_{t-1}), g_t]\right), \tag{1}$$

$$g_t = \textstyle\sum_{i=1}^{T} \alpha_{t,i} f_i, \tag{2}$$

$$\alpha_{t,i} = \frac{\exp\left(e_{t,i}\right)}{\sum_{j=1}^{T} \exp\left(e_{t,j}\right)}, \tag{3}$$

$$e_{t,i} = \omega^T \tanh\left(\mathcal{W}_s s_{t-1} + \mathcal{W}_f \mathcal{F} + b\right), \tag{4}$$

where ω, $\mathcal{W}_s$, $\mathcal{W}_f$, and b are trainable parameters, $E(\cdot)$ represents the character embedding layer, and [,] denotes concatenation. Subsequently, the decoder produces the probability $p_t \in \mathbb{R}^C$ for character y_t via a *softmax* function, where C is the character category,

$$p_t = softmax\left(\mathcal{W}_o s_t + b_o\right), \tag{5}$$

where $\mathcal{W}_o$ and b_o are trainable parameters. The baseline model is trained only on source data by a supervised loss,

$$\mathcal{L}_{sup} = \frac{1}{T}\sum_{t=1}^{T} \mathcal{H}(y_t^l, p_t^l) = -\frac{1}{T}\sum_{t=1}^{T} y_t^l \log p_t^l. \tag{6}$$

where $\mathcal{H}$ is a standard cross-entropy loss.

2) Consistency Regularization: Consistency regularization assumes that a model generates consistent predictions when fed perturbed versions of an image. Supposing that φ_w and φ_s denote weak and strong data augmentations, the recognition model $\mathcal{R}$ produces corresponding probabilities $p_w = \mathcal{R}(\varphi_w(x))$ and $p_s = \mathcal{R}(\varphi_s(x))$, respectively. The prediction $\tilde{y}_w = argmax(p_w)$ of the weak augmented sample $\varphi_w(x)$ is employed as the pseudo-label for the strong augmented ones $\varphi_s(x)$. The consistency loss is expressed as,

$$\mathcal{L}_{con}^{P}(p_w, p_s) = \frac{1}{T}\sum_{t=1}^{T} \mathbb{I}\left[max(p_{w,t}) \geq \eta_{pos}\right] \mathcal{H}(\tilde{y}_{w,t}, p_{s,t}), \tag{7}$$

where η_{pos} represents a pre-defined confidence threshold, and the superscript P signifies the default positive consistency loss. $\mathbb{I}(\cdot)$ denotes the indicator function. The $\mathcal{L}_{con}^{P}$ applies solely to instances where the maximum confidence exceeds the threshold η_{pos}.

B. Reweight Pseudo-Labels via Uncertainty Estimation

1) Refine Probability via Neighbours: The pseudo-labels of target samples generated by the pre-trained source model may include incorrect predictions due to the presence of inter-domain noise. Directly optimizing these noisy pseudo-labels can result in unstable training that adversely impacts model performance. Therefore, we introduce a calibration strategy to refine the probability distribution of target samples by leveraging the probabilities of their nearest neighbors. The underlying assumption is that characters with similar semantics tend to be positioned closer within the feature space, resulting in relatively smaller distance measures.

Formally, given an image $x \in \mathcal{D}^u$, the model outputs a probability distribution $p \in \mathbb{R}^{T\times C}$, where C is the number of character categories. Following [7] and [13], we denote the feature of the predicted character y_t as g_t. This character feature guides the search for nearest neighbors. To identify which characters' probability distributions are utilized to refine the anchor ones, a character pool $\mathcal{P}$ is used to store the predicted probability distributions of the nearest neighbors, which are measured using the cosine distance in the feature space. Consequently, the refined probability distribution $\tilde{p}_t \in \mathbb{R}^C$ is computed by performing a soft-voting mechanism, averaging the probabilities of the nearest neighbors,

$$\tilde{p}_t = (1-\mu)\, p_t + \mu \frac{1}{|\mathcal{P}|}\sum_{i\in\mathcal{P}} p_{t,i}, \tag{8}$$

where $\mu \in [0, 1]$ is the refined intensity. $\mu = 0$ denotes no refinement, and $\mu = 1$ denotes all determined by the probability distribution of the nearest neighbors.

It is worth noting that the method [40] also employs a pseudo-labels calibration, where the refined probability is entirely determined by the probability distribution of the nearest neighbors. However, its calibration leads to unstable training in STR, while our proposed strategy can optimize the model well. We also analyze this by experiments in Sec. IV-E3.

2) Reweight Pseudo-Labels: Recognizing characters in real text varies in difficulty, with the same character being more challenging to identify in curved text compared to regular text. Treating all characters equally and relying unquestioningly on pseudo-labels can impact effective adaptation. Although there are methods for assessing the uncertainty of pseudo-labels, these methods [49], [50] are not suitable for STR tasks due to assessing the whole image, ignoring fine-grained character properties in STR tasks. Hence, it makes sense to assign distinct weights to individual character pseudo-labels dynamically. This leads us to the question: *How can the weight of each character pseudo-label be determined?*

To address this issue, we draw attention to supervised cross-entropy optimization. A well-performing model typically

predicts a high probability for the correct category and low probabilities for other categories, resulting in a low cross-entropy loss. Such scenarios also exhibit low entropy in the probability distribution. Drawing from this insight, a similar idea is applied to unsupervised optimization. If the entropy of a probability distribution is relatively low, the corresponding pseudo-label is considered reliable, indicating low uncertainty. Conversely, high entropy suggests that the pseudo-label lacks reliability, indicating high uncertainty. Therefore, the entropy of a probability distribution can be applied to measure the uncertainty of pseudo-labels.

More formally, given a refined probability distribution $\tilde{p}_t \in \mathbb{R}^C$ of y_t, the associated entropy $\mathcal{H}(\tilde{p}_t)$ is,

$$\mathcal{H}(\tilde{p}_t) = \mathbb{E}\left[I(\tilde{p}_t)\right] = -\sum_{c=1}^{C} \tilde{p}_t^{(c)} \log \tilde{p}_t^{(c)}, \tag{9}$$

where C is the number of character categories. The entropy is subsequently re-scaled by,

$$\tilde{\mathcal{H}}(\tilde{p}_t) = \frac{\mathcal{H}(\tilde{p}_t)}{\log C}. \tag{10}$$

Then, the weight ω_t of pseudo-label y_t is determined through a negative exponential function,

$$\omega_t = \exp(-\tilde{\mathcal{H}}(\tilde{p}_t)). \tag{11}$$

In this way, high entropy corresponds to low weight, *i.e.*, less importance. Consequently, the reweighted pseudo-labels are optimized using entropy minimization,

$$\mathcal{L}_{wem} = -\frac{1}{T}\sum_{t=1}^{T}\sum_{c=1}^{C} \omega_t \, \tilde{p}_t^{(c)} \log \tilde{p}_t^{(c)}. \tag{12}$$

Note that the entropy is computed on the refined probability distribution rather than those directly predicted by the model used in [50]. The underlying idea is that the refined probability distribution could reflect the character relationships in the feature space. As illustrated in Fig. 3, when the probability distributions of the nearest neighbor characters exhibit consistency, the entropy of the refined distribution remains relatively low, indicating a highly reliable and more important pseudo label. Conversely, if the probability distributions of the nearest neighbors show inconsistency, the entropy of the refined distribution increases, signifying a highly uncertain pseudo label. This lines up with our assumption that low entropy corresponds to low uncertainty.

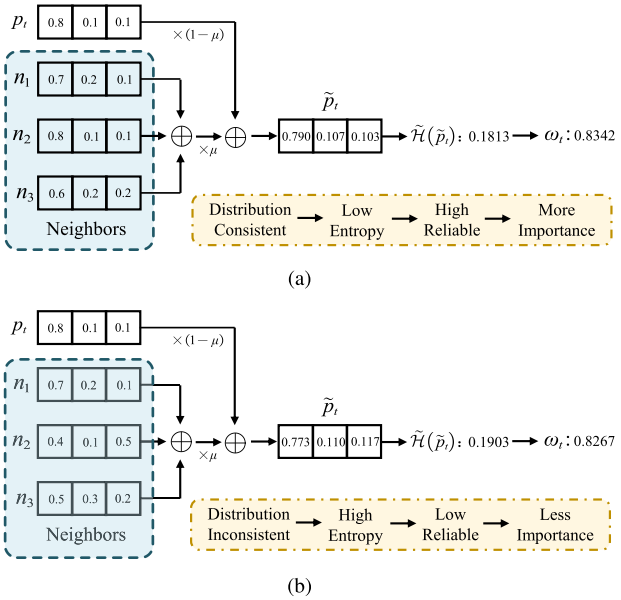


Fig. 3. Refinement and Reweight of different nearest neighbors under the refined intensity $\mu = 0.1$. (a) is the case where the probability distributions of the nearest neighbors are relatively consistent. (b) denotes the case where the probability distributions of the nearest neighbors are inconsistent.

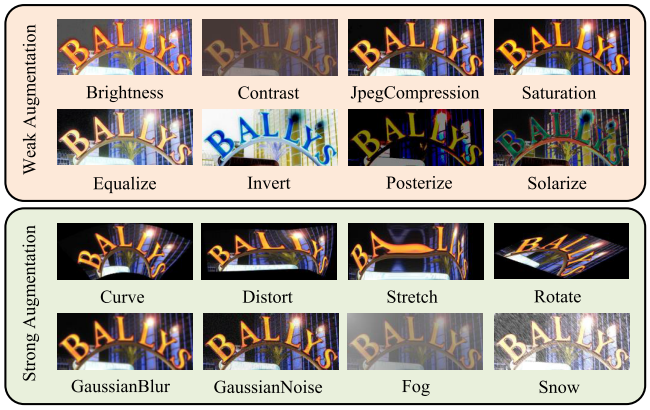


Fig. 4. Examples of various augmentation. The top is weak augmentation, mainly color changes. The bottom is strong augmentation, including geometric transformations, Gaussian blur, Gaussian noise, and weather variations.

C. Decoupled Triple-P-N Consistency Matching

As discussed in Sec. I, the challenges in STR arise not only from the inter-domain noise but also from the inherent complexity of real scene text, such as curvature, lighting variations, occlusion, and shadows. These intra-domain noises significantly impact the robustness of the model in real natural scenes. A straightforward way to address this issue is to introduce real text samples with diverse variations into the target domain to enhance the anti-interference ability of the model. However, acquiring labeled real scene text with various natural transformations is often limited, making this less feasible. To overcome this limitation, we introduce diverse data augmentations to increase data diversity. These data augmentations are classified into weak and strong augmentation for different transform intensities. Specifically, following the definitions outlined in [47], weak augmentation primarily focuses on color variations, including contrast, brightness, saturation, and solarization. Strong augmentation primarily involves geometric transformations such as rotation, curvature, and distortion. In addition, we incorporate more comprehensive perturbations as part of the strong augmentation, including Gaussian blur and weather-related changes like rain, snow, and fog. Visualization examples of these data augmentations are depicted in Fig. 4.

We aim to enable the model to maintain consistent predictions regardless of the noise inputs. Various data-augmented views can be regarded as input images under intra-domain noise. The robustness of the model can be improved by maintaining the consistency of probability distribution.

Consistency regularization aligns perfectly with this goal. Hence, we propose a decoupled triple-P-N consistency matching module, which enables the model to maintain output consistency amidst varying noise disturbances. Firstly, unlike the single matching used in FixMatch [51] and MixMatch [52], which only match strong and weak augmentations, we introduce a triple matching strategy involving the original, weak, and strong views. This matching way enhances the perception of the original view. Secondly, inspired by NLNL [53], we design a character negative learning aimed at learning low-confidence characters. Certain characters may yield low-confidence predictions when confronted with various noises. In cases where traditional high-confidence-based judgment is directly applied, these low-confidence characters could be overlooked. Negative character learning effectively captures the information in these low-confidence characters, thereby improving sample utilization. Unlike positive learning, which focuses on maximum confidence, character negative learning operates on multiple confidences lower than a pre-defined confidence threshold. Lastly, PL and NL cooperate separately, ensuring that high-confidence-based PL and low-confidence-based NL are decoupled.

Formally, given a target image $x \in \mathcal{D}^u$, subjected to both weak and strong augmentations, the model $\mathcal{R}$ produces corresponding probability distributions, $p = \mathcal{R}(x) \in \mathbb{R}^{T\times C}$, $p_w = \mathcal{R}(\varphi_w(x)) \in \mathbb{R}^{T\times C}$, and $p_s = \mathcal{R}(\varphi_s(x)) \in \mathbb{R}^{T\times C}$, respectively. The high-confidence-based triple positive consistency matching loss is formulated as,

$$\mathcal{L}^P_{Tri} = \mathcal{L}^P_{con}(p, p_w) + \mathcal{L}^P_{con}(p, p_s) + \mathcal{L}^P_{con}(p_w, p_s)\,, \tag{13}$$

$$\mathcal{L}^P_{con}(p, p_w) = \frac{1}{T}\sum_{t=1}^{T} \mathbb{I}\left[max(p_t) \geq \eta_{pos}\right] \mathcal{H}(\tilde{y}_t, p_{w,t})\,, \tag{14}$$

$$\mathcal{L}^P_{con}(p, p_s) = \frac{1}{T}\sum_{t=1}^{T} \mathbb{I}\left[max(p_t) \geq \eta_{pos}\right] \mathcal{H}(\tilde{y}_t, p_{s,t})\,, \tag{15}$$

$$\mathcal{L}^P_{con}(p_w, p_s) = \frac{1}{T}\sum_{t=1}^{T} \mathbb{I}\left[max(p_{w,t}) \geq \eta_{pos}\right] \mathcal{H}(\tilde{y}_{w,t}, p_{s,t})\,, \tag{16}$$

where $\tilde{y}_t = argmax(p_t)$, $\tilde{y}_{w,t} = argmax(p_{w,t})$, and η_{pos} is a pre-defined positive confidence threshold. Compared with the conventional positive consistency loss defined in Eq. 7, our triple positive consistency loss strengthens the matching strength by further considering valuable feature information in the raw image. By aligning weak ones with the raw image and strong ones with the raw image, the perception ability in the raw image of the model is boosted.

In contrast to positive learning, which relies solely on maximum confidence, the character negative learning works with multiple confidences with lower levels instead of concentrating solely on minimum confidence. The low-confidence-based triple negative consistency matching loss is defined as,

$$\mathcal{L}^N_{Tri} = \mathcal{L}^N_{con}(p, p_w) + \mathcal{L}^N_{con}(p, p_s) + \mathcal{L}^N_{con}(p_w, p_s)\,, \tag{17}$$

$$\mathcal{L}^N_{con}(p, p_w) = -\frac{1}{T}\sum_{t=1}^{T}\sum_{c=1}^{C} \mathbb{I}\left[p_t^{(c)} \leq \eta_{neg}\right] \log(1 - p_{w,t}^{(c)})\,, \tag{18}$$

$$\mathcal{L}^N_{con}(p, p_s) = -\frac{1}{T}\sum_{t=1}^{T}\sum_{c=1}^{C} \mathbb{I}\left[p_t^{(c)} \leq \eta_{neg}\right] \log(1 - p_{s,t}^{(c)})\,, \tag{19}$$

$$\mathcal{L}^N_{con}(p_w, p_s) = -\frac{1}{T}\sum_{t=1}^{T}\sum_{c=1}^{C} \mathbb{I}\left[p_{w,t}^{(c)} \leq \eta_{neg}\right] \log(1 - p_{s,t}^{(c)})\,, \tag{20}$$

where η_{neg} is a pre-defined negative confidence threshold. The superscript N signifies the negative loss. Then, the decoupled triple-P-N consistency matching loss is,

$$\mathcal{L}_{Tri} = \mathcal{L}^P_{Tri} + \mathcal{L}^N_{Tri}\,. \tag{21}$$

In the decoupled triple-P-N consistency matching module, all character probability distributions are direct outputs without refinement for pseudo-labels. The original output better reflects the recognition ability of the model in the presence of disturbance from intra-domain noise. We have also conducted experiments on consistency matching using refined probability distributions. The results are analyzed in Sec. IV-D2.

D. Joint Training for UDA and SFUDA

For the UDA setting, the overall loss integrates the supervised loss in Eq. 6, the reweighted entropy minimization in Eq. 12, and the decoupled triple-P-N consistency matching loss in Eq. 21. The joint training UDA optimization function is,

$$\mathcal{L}_{UDA} = \mathcal{L}_{sup} + \lambda_{wem}\mathcal{L}_{wem} + \lambda_{Tri}\mathcal{L}_{Tri}\,, \tag{22}$$

where λ_{wem} and λ_{Tri} are trade-off parameters. Under the SFUDA setting, only the pre-trained source model is used to predict pseudo-labels, and the source data is unavailable. Thus, the joint training SFUDA optimization function is,

$$\mathcal{L}_{SFUDA} = \lambda_{wem}\mathcal{L}_{wem} + \lambda_{Tri}\mathcal{L}_{Tri}\,. \tag{23}$$

This comprehensive optimization enables the model to adapt to the target domain by refining probability distribution, incorporating entropy-based reweighting, and maintaining consistency under different noise disturbances. Due to the perception of inter- and intra-domain noise, the model robustness is enhanced, thereby resulting in improved performance.

IV. Experiments and Results

A. Datasets

Four types of 11 datasets are used for text recognition tasks: labeled synthetic text, unlabeled real scene text, handwritten text, and artistic text.

Synthetic Text: Two widely used synthetic datasets are adopted including Synth90k (MJ) [54] and SynthText (ST) [55], which contains 14.5M images in total.

Real Scene Text: Seven benchmarks are tested, including four regular datasets, *i.e.*, IIIT5K [56], SVT [57], IC03 [58], and IC13 [59], and three irregular datasets, *i.e.*, SVTP [60], CUTE80 [61], and IC15 [62]. Details of datasets can be found in the previous work [4].

Handwritten Text: IAM [63] is an English handwritten dataset written by 657 writers. According to standard partition [64], IAM[1] is divided into 53841 training words, 8566 validation words, and 17616 test words.

[1]https://fki.tic.heia-fr.ch/databases/iam-handwriting-database

TABLE I

EVALUATION RESULTS ON THE ADAPTATION FROM SYNTHETIC TO SCENE TEXT COMPARED WITH SOTA METHODS. '*' INDICATES THE REPRODUCED RESULTS UNDER THE SAME TRAINING SET. THE NUMBERS IN PARENTHESES DENOTE THE AMOUNTS, *e.g.*, 100M MEANS 100 MILLION. BOLD AND UNDERLINED INDICATE THE BEST AND SECOND-BEST RESULTS

	Methods	Labeled	Unlabeled	Regular Text				Irregular Text			Avg.
				IIIT5K	SVT	IC03	IC13	SVTP	CUTE80	IC15	
SOTAs	CRNN(TPAMI2017) [21]	MJ	-	82.90	81.60	93.10	91.10	70.00	65.50	69.40	-
	TRBA(ICCV2019) [4]	MJ+ST	-	87.90	87.50	94.90	93.60	79.20	74.00	77.60	-
	Aster(TPAMI2019) [24]	MJ+ST	-	93.40	89.50	94.50	-	78.50	79.50	76.10	-
	Scatter(CVPR2020) [67]	MJ+ST+SA	-	93.70	92.70	-	-	86.90	87.50	-	-
	SRN(CVPR2020) [28]	MJ+ST	-	94.80	91.50	-	95.50	85.10	87.80	82.70	-
	VisionLAN(AAAI2021) [29]	MJ+ST	-	95.80	91.70	-	95.70	86.00	88.50	83.70	
	ABINet(CVPR2021) [26]	MJ+ST	-	96.20	93.50	-	97.40	89.30	89.20	86.00	-
	PerSec(AAAI2022) [12]	MJ+ST	UTI(100M)	88.10	86.80	-	94.20	77.70	72.70	73.60	-
	ConCLR(AAAI2022) [13]	MJ+ST	OutText(1k)	96.50	94.30	-	97.70	89.30	91.30	85.40	-
	DiG(MM2022) [68]	MJ+ST	URD(15.77M)	96.70	94.60	-	96.90	91.00	91.30	87.10	-
	Zheng *et.al.*(CVPR2022)* [47]	MJ+ST	Real(6.9k)	90.70	91.96	96.16	96.15	86.98	84.72	83.77	-
	MATRN(ECCV2022) [69]	MJ+ST	-	96.60	95.00	-	97.90	90.60	93.50	86.60	-
	SIGA(CVPR2023) [14]	MJ+ST	-	96.60	95.10	96.90	97.80	90.50	93.10	86.60	-
	CCD(ICCV2023) [16]	MJ+ST	URD(15.77M)	97.20	94.40	-	97.00	91.80	93.30	87.60	-
	TPS++(IJCAI2023) [70]	MJ+ST	-	96.30	94.30	-	97.80	89.60	89.60	86.50	-
	LPV-Base(IJCAI2023) [71]	MJ+ST	-	**97.30**	94.60	-	97.60	90.90	**94.80**	87.50	-
UDA	SSDAN(CVPR2019) [44]	MJ+ST	Real(6.9k)	87.60	88.10	94.60	93.80	-	73.90	78.70	-
	ASSDA(TIP2021) [7]	MJ+ST	Real(6.9k)	88.30	88.60	95.50	93.70	-	76.30	78.70	-
	SMILE(ICIP2022) [5]	MJ+ST	Real(6.9k)	89.30	87.60	96.00	94.90	-	75.60	78.90	-
	DOC(TMM2023) [6]	MJ+ST	Real(6.9k)	89.00	89.00	95.30	94.30	81.20	77.00	76.00	-
	CADA(TCSVT2023) [11]	MJ+ST	Real(6.9k)	89.30	89.03	95.35	95.10	81.55	78.40	80.12	87.48
Ours	TRBA-Baseline	MJ+ST	-	87.40	87.02	95.12	92.88	80.00	74.22	78.08	85.63
	TRBA-SFUDA	-	Real(6.9k)	90.27	89.18	95.23	94.87	83.26	78.40	82.33	88.44
	TRBA-UDA	MJ+ST	Real(6.9k)	91.27	91.35	95.58	95.68	83.26	80.49	82.88	89.31
	ABINet-Baseline	MJ+ST	-	96.10	93.97	95.93	96.50	89.30	91.67	85.31	92.84
	ABINet-SFUDA	-	Real(6.9k)	96.13	94.28	95.70	97.43	89.46	90.63	87.50	93.49
	ABINet-UDA	MJ+ST	Real(6.9k)	96.43	**95.83**	**97.21**	**98.25**	**91.93**	91.67	**87.82**	**94.12**

Artistic Text: WordArt [65] is an artistic text dataset comprising 6316 artistic text images. Following the splitting rule of TextSeg [66], the dataset is divided into a training set with 4805 images and a test set with 1511 images.

B. Experimental Settings

1) Implementation Details: Two representative STR models, TRBA [4] and ABINet [26], are used to validate the effectiveness of our framework with their default configurations. The *Adadelta* optimizer is used for TRBA during the adaptation, while the *Adam* optimizer is employed for ABINet. The initial learning rates are set to 0.1 for TRBA and 0.0001 for ABINet. Training is conducted for 300,000 iterations with a batch size of 48. The baseline model is warmed by labeled synthetic text and serves as a pre-trained source model for SFUDA. The number of character categories C is 38, including 10 digits, 26 case-insensitive letters, a start symbol ['GO'], and a stop symbol ['S']. The default number of neighbors K is 10, and the refined intensity μ is 0.1. The positive threshold η_{pos} is 0.9, while the negative threshold η_{neg} is 0.1.

2) Evaluation Metric: We assess the performance of STR models using word-level accuracy as the primary metric. To provide a comprehensive evaluation, we introduce an average metric *Avg.*, which computes the mean results across all samples from seven real scene datasets. In cases where the target domain is handwritten text, we follow standard practice by reporting the word error rate (WER) and character error rate (CER) for handwritten text recognition (HTR).

C. Comparison With SOTAs

We comprehensively compare our proposed framework with several SOTA STR methods, particularly those focused on UDA. Specifically, we choose TRBA and ABINet as baseline models with the default parameter configurations. The reason for selecting these two STR methods is that, on the one hand, they represent different decoding ways. TRBA employs an RNN decoder, while ABINet uses a transformer decoder. On the other hand, the TRBA baseline allows us to make fair comparisons with existing UDA-based STR methods that are also based on TRBA architecture. Compared to the simple architecture of TRBA, ABINet is a more powerful baseline that can be appropriately compared with the SOTA methods. Technically, the proposed framework is deployed directly to the output of TRBA, while it is deployed on the last iteration output of the fusion phase for ABINet due to it being a visual-language-fusion architecture.

For the fairness of the comparison, in addition to presenting the results in the original papers, we reproduce TRBA and ABINet as baseline models, denoted as TRBA-Baseline and ABINet-Baseline, respectively. Some of our reproduced results even outperform those reported in the original papers. Additionally, we extend the evaluation to a special case, SFUDA. From the results in Table I, we can observe:

- Our methods consistently outperform the baselines under both the SFUDA and UDA settings. Compared to TRBA-Baseline, our methods achieve an average improvement to 2.81% (85.63%→88.44%) and 3.68% (85.63%→89.31%) in the SFUDA and UDA settings. Similarly, compared to

TABLE II

ABLATION STUDIES OF SUB-COMPONENTS OF THE PROPOSED METHOD UNDER SFUDA AND UDA SETTING. EM: ENTROPY MINIMIZATION LOSS

SourceData	EM	Refine &Reweight EM	PL Consistency	Decoupled Triple-P-N Consistency	Avg.
✓	✗	✗	✗	✗	85.63
✓	✓	✗	✗	✗	87.41
✓	✗	✓	✗	✗	87.68
✓	✗	✗	✓	✗	86.97
✓	✗	✗	✗	✓	89.02
✓	✗	✓	✗	✓	89.31
✗	✓	✗	✗	✗	86.02
✗	✗	✓	✗	✗	86.17
✗	✗	✗	✓	✗	86.70
✗	✗	✗	✗	✓	87.70
✗	✗	✓	✗	✓	88.44

ABINet-Baseline, the average results of ABINet-SFUDA and ABINet-UDA are improved by 0.65% and 1.28%, respectively. This highlights the generality of our method in enhancing the performance of off-the-shelf STR models.

- Compared to UDA-based methods employing the same TRBA baseline, our TRBA-SFUDA model achieves superior results with fewer training samples. Furthermore, our TRBA-UDA model surpasses these UDA-based methods under the same UDA settings, showcasing faster convergence and more stable training, as depicted in Fig. 1.
- Under the more robust ABINet baseline, our ABINet-UDA achieves competitive performance with SOTA methods, particularly the self-supervised methods utilizing a substantial amount of real scene data. Although the LPV-Base method outperforms our model on the IIIT5K and CUTE80 datasets, it performs poorly on the remaining five datasets under globally optimized conditions.

In general, the results on real scene text demonstrate that our proposed framework can perceive inter-domain and intra-domain noise, effectively mitigating the domain discrepancies in synthetic and real text and enhancing the model's robustness to real environmental noise.

D. Ablation Study

Due to the simplicity of the TRBA, it is selected to perform ablation experiments to analyze the proposed framework.

1) Effect of Each Component: We conduct experiments to validate the effectiveness of each proposed module in both the SFUDA and UDA settings. The results are summarized in Table II. We establish the baseline accuracy, yielding a relatively low performance of 85.63%. By incorporating the reweight pseudo-labels via the uncertainty estimation module alone, the accuracy is increased by 2.05% (85.63%→87.68%) in the UDA setting and by 0.54% (85.63%→86.17%) in the SFUDA setting. Similarly, when solely utilizing the decoupled triple-P-N consistency matching module, we observe a substantial improvement of 3.39% (85.63%→89.02%) and 2.07% (85.63%→87.70%) in the UDA and SFUDA settings, respectively. Finally, the accuracy is further increased when both proposed modules are jointly optimized.

TABLE III

EVALUATION RESULTS OF REFINE AND REWEIGHT PSEUDO-LABELS UNDER UDA SETTING. EM: ENTROPY MINIMIZATION LOSS

Model	Avg.
Baseline	85.63
EM	87.41
Refine EM	87.55
Reweight EM	87.42
Refine&Rewight EM	87.68
Refine Decoupled Triple-P-N Consistency	88.76
Decoupled Triple-P-N Consistency	89.02

TABLE IV

EVALUATION RESULTS OF DIFFERENT CONSISTENCY LOSSED UNDER UDA SETTING

Model	Avg.
Baseline	85.63
Consistency PL	86.97
Consistency Triple-PL	88.56
Consistency NL	86.01
Consistency Triple-NL	86.70
Ours(Decoupled Triple-P-N Learning	89.02

Additionally, we further explore the effects of directly optimizing the model using the entropy minimization loss (*EM*) and the consistency regularization loss (*PL Consistency*). The results reveal that the performance of our proposed modules surpasses that of models directly optimized with these two losses, demonstrating the effectiveness of our method in both UDA and SFUDA settings.

2) Effect of Refine and Reweight: A deeper investigation into the reweight pseudo-labels via uncertainty estimation module could provide a nuanced understanding of the impacts of probability refinement and pseudo-labels reweighting. The results detailed in Table III demonstrate that models optimized solely with refinement (*Refine EM*) and reweighting (*Reweight EM*) experience marginal enhancements compared to the one optimized directly using entropy minimization (*EM*). Notably, a slight performance boost is observed when combining both probability refinement and reweighting (*Refine&Reweight EM*). Furthermore, applying probability refinement to the decoupled triple-P-N consistency matching module decreases performance (89.02%→88.76%). This aligns with the principle of consistency regularization, wherein the direct output of weak augmented images serves as pseudo-labels for strong augmented ones rather than the refined outputs.

3) Effect of Consistency Matching: In the decoupled triple-P-N consistency matching module, the high-confidence-based PL and low-confidence-based NL jointly ensure the prediction consistency under varying noise conditions. To understand the effectiveness of PL and NL, we conduct an in-depth analysis as presented in Table IV. In comparison to the *Baseline* model, single PL or NL improvements elevate the accuracy by 1.34% (85.63%→86.97%) and 0.65% (85.63%→86.01%), respectively. This single PL or NL primarily aligns the output of strong augmented images with their weak augmented counterparts. However, such a matching way may overlook some critical original image information. To address this limitation, we devise triple-PL and triple-NL matching ways,

TABLE V

COMPARISON WITH SOME METHODS THAT FOCUS ON NOISE LABELS. NER-TR AND GUIDING-TR DENOTE THE RESULTS OF DEPLOYING THEIR CORE METHODS TO THE STR TASK

Model	Labeled	Unlabeled	IIIT5K	SVT	IC03	IC13	SVTP	CUTE80	IC15	Avg.
NEL-TR(WACV2022) [72]	MJ+ST	Real(6.9k)	89.53	88.56	95.35	94.28	81.09	78.05	79.02	87.15
CADA(TCSVT2023) [11]	MJ+ST	Real(6.9k)	89.30	89.03	95.35	95.10	81.55	78.40	80.12	87.48
Guiding-TR(CVPR2023) [40]	MJ+ST	Real(6.9k)	90.77	90.42	95.63	95.80	83.10	80.14	82.88	89.07
Ours	MJ+ST	Real(6.9k)	91.27	91.35	95.58	95.68	83.26	80.49	82.88	89.31

TABLE VI

EVALUATION RESULTS FOR DIFFERENT AMOUNTS OF TRAINING DATA. THE *-SFUDA MODELS ARE NOT ABLE TO OBTAIN THE CORRESPONDING AMOUNT OF SOURCE DATA, ONLY THE SOURCE PRE-TRAINED MODEL

Index	Labeled	Unlabeled	Methods	IIIT5K	SVT	IC03	IC13	SVTP	CUTE80	IC15	Avg.
1	20%	20%	CADA	88.83	87.94	94.77	93.93	80.78	77.70	78.96	86.69
			TRBA-SFUDA	88.27	87.79	95.70	94.17	80.78	77.35	79.51	86.70
			TRBA-UDA	89.00	88.87	95.70	95.57	81.40	77.70	80.18	87.42
2	20%	80%	CADA	88.27	89.03	95.35	94.75	80.93	78.40	79.63	86.90
			TRBA-SFUDA	89.60	89.34	95.23	94.75	82.95	79.44	82.33	88.21
			TRBA-UDA	89.77	90.11	95.12	94.87	83.88	80.84	83.10	88.63
3	50%	50%	CADA	88.57	88.72	95.47	94.75	80.78	77.70	78.74	86.77
			TRBA-SFUDA	89.10	88.72	95.70	95.33	81.40	75.61	81.56	87.65
			TRBA-UDA	89.90	89.34	95.81	96.38	83.10	77.70	82.33	88.50
4	80%	20%	CADA	89.07	88.87	95.23	95.22	81.24	79.10	79.57	87.26
			TRBA-SFUDA	88.27	87.79	95.70	94.17	80.78	77.35	79.51	86.70
			TRBA-UDA	88.90	89.49	95.47	95.57	81.86	78.05	81.67	87.79
5	80%	80%	CADA	88.87	89.03	95.58	94.63	80.93	77.00	79.46	87.05
			TRBA-SFUDA	89.60	89.34	95.23	94.75	82.95	79.44	82.33	88.21
			TRBA-UDA	90.40	89.80	95.35	95.68	83.88	80.49	83.21	88.96
6	100%	100%	CADA	89.30	89.03	95.35	95.10	81.55	78.40	80.12	87.48
			TRBA-SFUDA	90.27	89.18	95.23	94.87	83.26	78.40	82.33	88.44
			TRBA-UDA	91.27	91.35	95.58	95.68	83.26	80.49	82.88	89.31

which improve the accuracy further, enhancing it by 2.93% (85.63%→88.56%) and 1.07% (85.63%→86.70%) respectively. By applying both triple-PL and triple-NL, the model gains a more comprehensive improvement, resulting in an overall accuracy boost of 3.39% (85.63%→89.02%). This collective matching way jointly captures richer information from the unlabeled real scene text.

4) Comparison With UDA Methods: We also compare other UDA methods that aim to address noisy pseudo labels. Only CADA filters out pseudo labels with low confidence in STR tasks using a simple high threshold. Therefore, we select two representative methods for handling noisy pseudo labels, NEL [72] and Guiding [40], for comparison. The former [72] mitigates the effect of noisy pseudo labels through negative ensemble learning, while the latter [40] uses neighborhood knowledge entirely for pseudo label refinement. We incorporate the core innovations of these two methods into the corresponding modules of our method, denoted as NEL-TR and Guiding-TR, respectively. Our method yields optimal results from Table V. This is partly due to the appropriate calibration of the pseudo labels and partly to the low-confidence character-based negative learning instead of all matches except pseudo labels as in [72]. Furthermore, while the Guiding-TR [40] achieves comparable results, the pseudo label calibration it employs leads to oscillations in the later stage of optimization, as detailed in subsection IV-E3.

5) Training Data Volume Analysis: We conduct a series of experiments to evaluate the performance with varying amounts of training data. Specifically, we vary the proportion of labeled and unlabeled data under the UDA and SFUDA settings, ranging from {20%, 50%, 80%, 100%}. As seen in Table VI:

- Reducing the labeled and unlabeled data in the same proportion (Indexes 1, 3, 5, and 6) degrades the performance of our method and CADA. Still, our method consistently outperforms CADA in both cases.
- With the same amount of labeled data (Indexes 1, 2 and Indexes 4, 5), the UDA method with 80% unlabeled data outperforms the 20% setting by approximately 1.2% (87.42%→88.63% and 87.79%→88.96%). Conversely, with the same amount of unlabeled data (Indexes 1, 4 and Indexs 2, 5), the UDA method with 80% labeled data outperforms the 20% setting by only about 0.3% (87.42%→87.79% and 88.63%→88.96%). This demonstrates that unlabeled target data is crucial to the adaptation task. In scenarios with limited access to labeled data, comparable results can still be achieved using a larger proportion of unlabeled target data.

6) Generalization to Handwritten Text: We further test the generalization of our framework by adapting it to handwritten text. In this case, the source domain is labeled synthetic text, and the target domain is unlabeled handwritten text IAM. Adapting to handwritten text is notably more challenging due to the pronounced domain discrepancies between handwritten text and synthetic text, such as unique stroke characteristics. To accommodate this, we reduce the trade-off parameter

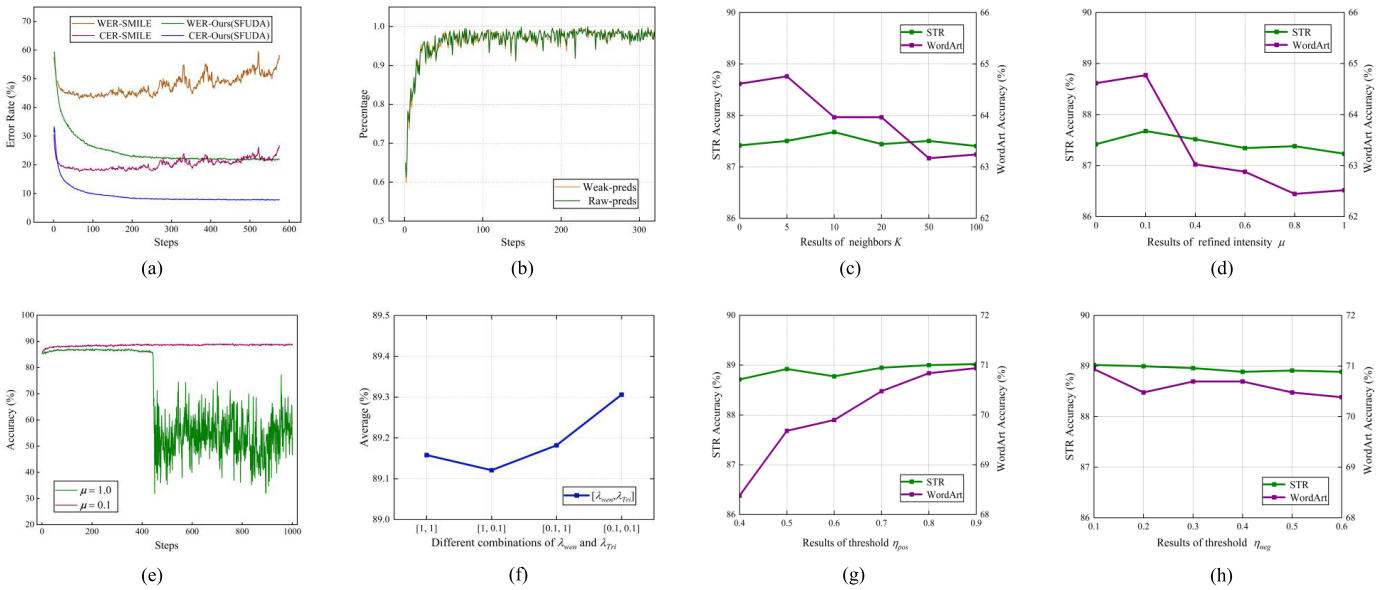


Fig. 5. (a) Comparison of SMILE [5] and our methods on WER and CER during training. SMILE shows oscillations in the later optimization stages, whereas our method is more stable. (b) Percentage of pseudo-labels participating in a consistency matching in each minibatch. (c)-(d) Evaluation results of various neighbors and refined intensity. (e) Evaluation results for refined intensity of 0.1 and 1. (f) Evaluation results of various trade-off parameter combinations. (g)-(h) Evaluation results of various positive and negative thresholds.

TABLE VII

EVALUATION RESULTS ON THE TASK FROM SYNTHETIC TEXT TO HANDWRITTEN TEXT IAM. '*' DENOTES THE REPRODUCED RESULTS. BOLD INDICATES THE BEST RESULTS. THE HYPERPARAMETERS ARE SET TO DEFAULT VALUES CONSISTENT WITH STR

	Methods	Labeled	Unlabeled	WER↓	CER↓
SOTAs	Base(TIP2021) [7]	MJ+ST	-	54.30	28.41
	SSDAN(CVPR2019) [44]	MJ+ST	IAM	53.65	27.26
	ASSDA(TIP2021) [7]	MJ+ST	IAM	43.78	19.96
	SMILE(ICIP2022)* [5]	MJ+ST	IAM	45.57	19.35
	CADA(TCSVT2023) [11]	MJ+ST	IAM	45.70	19.67
	DOC(TMM2023) [6]	MJ+ST	IAM	37.44	16.52
Ours	TRBA-Baseline	MJ+ST	-	57.07	30.90
	TRBA-SFUDA	-	IAM	**21.47**$^{\downarrow 35.6}$	**7.60**$^{\downarrow 23.3}$
	TRBA-UDA	MJ+ST	IAM	23.46	8.59

TABLE VIII

EVALUATION RESULTS ON THE TASK FROM SYNTHETIC TEXT TO ARTISTIC TEXT. BOLD INDICATES THE BEST RESULTS. THE HYPERPARAMETERS ARE SET TO DEFAULT VALUES CONSISTENT WITH STR

	Methods	Labeled	Unlabeled	Avg.
SOTAs	ASSDA(TIP2021) [7]	MJ+ST	WordArt	62.08
	SMILE(ICIP2022) [5]	MJ+ST	WordArt	63.03
	CADA(TCSVT2023) [11]	MJ+ST	WordArt	63.82
	DOC(TMM2023) [6]	MJ+ST	WordArt	63.03
Ours	TRBA-Baseline	MJ+ST	-	58.68
	TRBA-SFUDA	-	WordArt	**78.19**$^{\uparrow 19.51}$
	TRBA-UDA	MJ+ST	WordArt	70.94

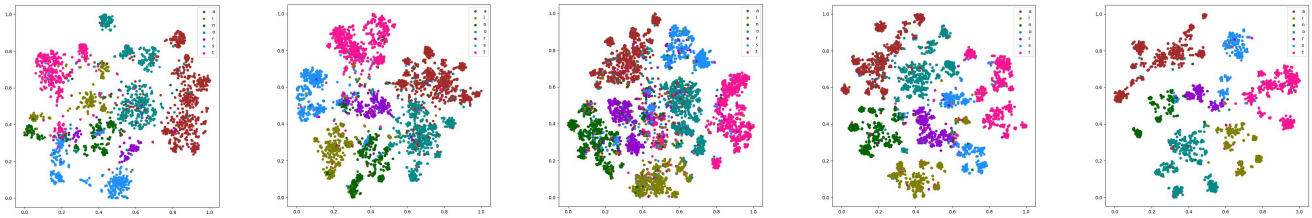
corresponding to inter-domain noise, setting $\lambda_{wem} = 0.001$. The results from Table VII reveal that:

- Compared to the TRBA-Baseline, our SFUDA and UDA models exhibit significant improvements. The TRBA-SFUDA achieves a remarkable reduction in WER and CER by 35.6% (57.07%→21.47%) and 23.3% (30.90%→7.60%), respectively. Surprisingly, the SFUDA model even outperforms the UDA one despite the latter having access to more source data. This phenomenon can be attributed to the initial capacity of the pre-trained source model for representation learning. Including the source data during adaptation introduces interference due to more pronounced domain discrepancies.
- Compared to SOTA methods using the same baseline, TRBA-SFUDA achieves superior results with fewer training samples. Specifically, compared to DOC, TRBA-SFUDA reduces WER and CER by 15.97% (37.44%→21.47%) and 8.92% (16.52%→7.60%). This improvement underscores the generalizability and effectiveness of our proposed framework for adapting to handwritten text. Overall, by leveraging a model originally tailored for STR, our method exhibits the potential to enhance HTR performance.

Moreover, for a more insightful understanding of our model, we visualize the optimization process of WER and CER on the IAM dataset. As depicted in Fig. 5(a), our method exhibits a more stable training process, indicative of its ability to capture distinctive features of handwritten text adequately.

7) Generalization to Artistic Text: To verify the necessity and effectiveness of domain adaptation, we conduct experiments on the WordArt dataset, which contains artistic text with more pronounced discrepancies from synthetic text. We reproduce four representative UDA-based text recognition methods experimented on WordArt. From the results in Table VIII, we can see that:

- Compared to TRBA-Baseline, our method shows an improvement of 19.51% and 11.87% in the SFUDA and UDA settings, respectively.
- Compared to our UDA method, the SFUDA method achieves better performance despite using less training data (78.19% vs. 70.94%). This phenomenon is also observed with handwritten text, suggesting that when the source and



(a) Baseline (b) ASSDA (c) SMILE (d) Ours(SFUDA) (e) Ours(UDA)

Fig. 6. Visualization of target domain character features on the adaptation of synthetic text to handwritten text IAM.

target domains differ significantly, the inclusion of source data may introduce interference with the target data.

- Compared to other UDA-based methods, our method achieves optimal performance in both SFUDA and UDA settings. This is attributed to the joint perception of inter- and intra-domain noise.

E. Algorithm Analysis

1) Analysis of Triple Consistency Matching: In the decoupled triple-P-N consistency matching module, only pseudo-labels with confidence exceeding the threshold η_{pos} are considered in triple PL matching. We track the proportion of each minibatch that participates in PL matching to gain insights into the training dynamics. The results, depicted in Fig. 5(b), show increasing pseudo-labels surpassing the threshold η_{pos} as training progresses. Remarkably, the trends in the percentages of raw images and corresponding weak augmented images serving as pseudo-labels remain consistent. This observation demonstrates the effectiveness of our proposed framework in enhancing the quality of pseudo-labels.

2) Analysis of Neighbors: We further assess the effect of the number of neighbor characters on probability refinement. Experiments are performed on synthetic text to real scene text and artistic text to analyze parameter sensitivities under different inter-domain discrepancies. As illustrated by the green line in Fig. 5(c), the results are relatively better when the number of nearest neighbors is set to 10. The results on STR are not significantly affected by K, indicating a wide range of stability intervals. In contrast, the results on the artistic text (purple line in Fig. 5(c)) show that better performance is achieved when K is set to 5. As the number of nearest neighbors increases, the accuracy tends to decrease. This phenomenon may be attributed to the fact that when the domain gap is obvious, selecting more neighbors could result in the inclusion of characters from other classes, potentially affecting the stability of the anchor probability distribution.

3) Analysis of Refined Intensity: We investigate the influence of different refined intensities on the stability of the target probability distribution by varying μ within the range of $\{0, 0.1, 0.4, 0.6, 0.8, 1\}$. The green line in Fig. 5(d) indicates that the effect of refined intensity on real scene text is slight. In contrast, the purple line indicates that the effect of refined intensity on artistic text is more pronounced, with relatively good results achieved at a refined intensity of 0.1. As the refined intensity increases, the accuracy decreases. In addition, to gain a more intuitive understanding of the model optimization process, we visualize the training curves of real scene text when the refined intensity is set to 0.1 (our default setting) and 1. In [40], the target probability distribution is obtained by averaging the probabilities of the neighbors, *i.e.*, the refined intensity is 1. Fig. 5(e) illustrates that when the refined intensity is 1, the adaptation becomes unstable in the later stages of training. This suggests that overly refining the probability distribution in text recognition tasks can consequently negatively impact model stability and recognition performance.

4) Parameter Sensitive Analysis: We first explore the impact of different trade-off parameters in Fig. 5(f). Specifically, we vary λ_{wem} and λ_{Tri} within the range of $\{0.1, 1.0\}$. The results demonstrate that different combinations of trade-off parameters affect the performance. Notably, when the combination is $\{0.1, 0.1\}$, the proposed two modules effectively perceive inter- and intra-domain noise, thereby enhancing the model performance. In addition, we explore the effects of positive and negative probability thresholds. As shown by the green line in Fig. 5(g) and Fig. 5(h), the effect of η_{pos} and η_{neg} is weak on the synthetic text to real scene text adaptation task. Due to the more pronounced domain gaps with the synthetic text, these two probability thresholds are more influential for the artistic text (purple line), but there is still a wide range of stability intervals. It can also produce comparable results if the default parameters of real scene text are used directly without fine-tuning to perform artistic text adaptation.

5) Visualization: We initially employ the t-SNE tool to visualize the distribution of character categories in the feature space on the handwritten text adaptation. Given many character categories, we randomly select a subset of categories for visualization. Fig. 6 clearly illustrates that the category boundaries of our two models are more distinct when compared to the Baseline, ASSDA, and SMILE. Furthermore, our SFUDA and UDA models exhibit more aggregated intra-class features and further away from inter-class features. Subsequently, we conduct visualizations to analyze the attention and recognition results, including real scene text, handwritten text, and artistic text. The results, as shown in Fig. 7, indicate that our proposed model demonstrates increased robustness, outperforming the baseline model, which sometimes makes incorrect predictions. However, as depicted on the right of the green dashed line, our model gives error results. For example, the real scene text 'ballys' is too curved, leading to a failure to localize 's'. Handwritten text is recognized as similar characters due to the continuous strokes of the handwriting, *e.g.*, the 'h' in 'they' is incorrectly recognized as 'u' or 'l'. When large and small words co-exist, the model

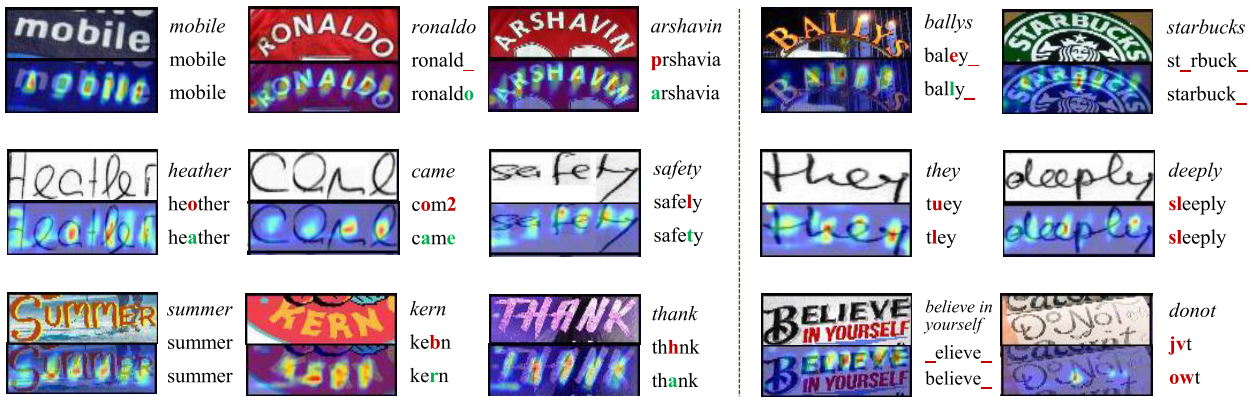


Fig. 7. Visualization of attention and prediction results. Left: raw images and attention results. Right: the three strings near each image represent the *ground truth*, the prediction of baseline, and the prediction of our method, respectively. Green and red colors indicate correct and incorrect prediction results. The left of the green dotted line is the correct cases, and the right is the failed cases.

can only locate larger words, *e.g.*, 'in yourself' is ignored in an artistic text. This reflects the limitations of our model for representation learning on curved text and stroke-ambiguous text, which will be investigated in our future work.

V. Conclusion

This paper presents an effective UDA framework to improve STR performance. The framework comprises two key components: a reweight pseudo-labels via uncertainty estimation module and a decoupled triple-P-N consistency matching module. These modules are strategically designed to address inter-domain and intra-domain noise, respectively. By assigning varying weights to target pseudo-labels based on entropy uncertainty, the method effectively mitigates the impact of domain gaps on target samples. Additionally, the quality of target pseudo-labels is improved using consistency regularization based on various data augmentations, enhancing model robustness against real noise. Extensive experiments conducted in both UDA and SFUDA settings demonstrate the superior performance of our proposed method.

References

[1] P. Wang et al., "PGNet: Real-time arbitrarily-shaped text spotting with point gathering network," in *Proc. Conf. Artif. Intell. (AAAI)*, 2021, pp. 2782–2790.

[2] Y. He et al., "Visual semantics allow for textual reasoning better in scene text recognition," in *Proc. 36th AAAI Conf. Artif. Intell.*, vol. 36, no. 1, Jun. 2022, pp. 888–896.

[3] P. Dai, H. Zhang, and X. Cao, "SLOAN: Scale-adaptive orientation attention network for scene text recognition," *IEEE Trans. Image Process.*, vol. 30, pp. 1687–1701, 2021.

[4] 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. 4714–4722.

[5] Y.-C. Chang, Y.-C. Chen, Y.-C. Chang, and Y.-R. Yeh, "Smile: Sequence-to-sequence domain adaptation with minimizing latent entropy for text image recognition," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2022, pp. 431–435.

[6] X.-Y. Ding, X.-Q. Liu, X. Luo, and X.-S. Xu, "DOC: Text recognition via dual adaptation and clustering," *IEEE Trans. Multimedia*, vol. 25, pp. 9071–9081, 2023.

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

[8] J. Baek, Y. Matsui, and K. Aizawa, "What if we only use real datasets for scene text recognition? Toward scene text recognition with fewer labels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 3112–3121.

[9] J. Lin, Z. Cheng, F. Bai, Y. Niu, S. Pu, and S. Zhou, "Text recognition in real scenarios with a few labeled samples," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 370–377.

[10] Y. Gao, Y. Chen, J. Wang, and H. Lu, "Semi-supervised scene text recognition," *IEEE Trans. Image Process.*, vol. 30, pp. 3005–3016, 2021.

[11] X.-Q. Liu, X.-Y. Ding, X. Luo, and X.-S. Xu, "Unsupervised domain adaptation via class aggregation for text recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 10, pp. 5617–5630, Mar. 2023.

[12] H. Liu et al., "Perceiving stroke-semantic context: Hierarchical contrastive learning for robust scene text recognition," in *Proc. AAAI Conf. Artif. Intell.*, 2022, pp. 1702–1710.

[13] X. Zhang, B. Zhu, X. Yao, Q. Sun, R. Li, and B. Yu, "Context-based contrastive learning for scene text recognition," in *Proc. AAAI Conf. Artif. Intell.*, 2022, pp. 3353–3361.

[14] T. Guan et al., "Self-supervised implicit glyph attention for text recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 15285–15294.

[15] C. Luo, L. Jin, and J. Chen, "SimAN: Exploring self-supervised representation learning of scene text via similarity-aware normalization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 1029–1038.

[16] T. Guan, W. Shen, X. Yang, Q. Feng, Z. Jiang, and X. Yang, "Self-supervised character-to-character distillation for text recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 19416–19427.

[17] C. Liu, C. Yang, and X.-C. Yin, "Open-set text recognition via character-context decoupling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 4513–4522.

[18] C. Liu, C. Yang, H.-B. Qin, X. Zhu, C.-L. Liu, and X.-C. Yin, "Towards open-set text recognition via label-to-prototype learning," *Pattern Recognit.*, vol. 134, Feb. 2023, Art. no. 109109.

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

[20] B. Su and S. Lu, "Accurate recognition of words in scenes without character segmentation using recurrent neural network," *Pattern Recognit.*, vol. 63, pp. 397–405, Mar. 2017.

[21] 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.

[22] 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.*, Jun. 2016, pp. 4168–4176.

[23] A. K. Bhunia, A. Sain, A. Kumar, S. Ghose, P. N. Chowdhury, and Y.-Z. Song, "Joint visual semantic reasoning: Multi-stage decoder for text recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 14920–14929.

[24] 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.

[25] N. Lu et al., "MASTER: Multi-aspect non-local network for scene text recognition," *Pattern Recognit.*, vol. 117, Sep. 2021, Art. no. 107980.

[26] 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 *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 7094–7103.

[27] Y. Wang et al., "PETR: Rethinking the capability of transformer-based language model in scene text recognition," *IEEE Trans. Image Process.*, vol. 31, pp. 5585–5598, 2022.

[28] D. Yu et al., "Towards accurate scene text recognition with semantic reasoning networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12110–12119.

[29] Y. Wang, H. Xie, S. Fang, J. Wang, S. Zhu, and Y. Zhang, "From two to one: A new scene text recognizer with visual language modeling network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 14174–14183.

[30] S. Zhao, R. Quan, L. Zhu, and Y. Yang, "CLIP4STR: A simple baseline for scene text recognition with pre-trained vision-language model," 2023, *arXiv:2305.14014*.

[31] R. Wang, Z. Wu, Z. Weng, J. Chen, G.-J. Qi, and Y.-G. Jiang, "Cross-domain contrastive learning for unsupervised domain adaptation," *IEEE Trans. Multimedia*, vol. 25, pp. 1665–1673, 2023.

[32] W. Deng et al., "Informative feature disentanglement for unsupervised domain adaptation," *IEEE Trans. Multimedia*, vol. 24, pp. 2407–2421, 2022.

[33] G. Li, G. Kang, W. Liu, Y. Wei, and Y. Yang, "Content-consistent matching for domain adaptive semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 440–456.

[34] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. J. Smola, "A kernel method for the two-sample-problem," in *Proc. Neural Inf. Process. Syst.*, 2006, pp. 513–520.

[35] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," 2014, *arXiv:1412.3474*.

[36] B. Sun, J. Feng, and K. Saenko, "Return of frustratingly easy domain adaptation," in *Proc. AAAI Conf. Artif. Intell.*, 2016, vol. 30, no. 1, pp. 2058–2065.

[37] B. Sun and K. Saenko, "Deep CORAL: Correlation alignment for deep domain adaptation," in *Proc. Eur. Conf. Comput. Vis. Workshops*, vol. 9915, 2016, pp. 443–450.

[38] F. Pan, I. Shin, F. Rameau, S. Lee, and I. S. Kweon, "Unsupervised intra-domain adaptation for semantic segmentation through self-supervision," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3763–3772.

[39] T.-H. Vu, H. Jain, M. Bucher, M. Cord, and P. Pérez, "ADVENT: Adversarial entropy minimization for domain adaptation in semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 2517–2526.

[40] M. Litrico, A. Del Bue, and P. Morerio, "Guiding pseudo-labels with uncertainty estimation for source-free unsupervised domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 7640–7650.

[41] J. Pei, Z. Jiang, A. Men, L. Chen, Y. Liu, and Q. Chen, "Uncertainty-induced transferability representation for source-free unsupervised domain adaptation," *IEEE Trans. Image Process.*, vol. 32, pp. 2033–2048, 2023.

[42] S. Roy et al., "Uncertainty-guided source-free domain adaptation," in *Proc. 17th Eur. Conf. Comput. Vis. (ECCV)*, vol. 13685, 2022, pp. 537–555.

[43] J. N. Kundu, N. Venkat, M. V. Rahul, and R. V. Babu, "Universal source-free domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4543–4552.

[44] 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. 2735–2744.

[45] G. Patel, J. Allebach, and Q. Qiu, "Seq-UPS: Sequential uncertainty-aware pseudo-label selection for semi-supervised text recognition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2023, pp. 6169–6179.

[46] M. Sajjadi, M. Javanmardi, and T. Tasdizen, "Regularization with stochastic transformations and perturbations for deep semi-supervised learning," in *Proc. Neural Inf. Process. Syst.*, 2016, pp. 1163–1171.

[47] C. Zheng, H. Li, S. Rhee, S. Han, J. Han, and P. Wang, "Pushing the performance limit of scene text recognizer without human annotation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 14096–14105.

[48] A. Aberdam, R. Ganz, S. Mazor, and R. Litman, "Multimodal semi-supervised learning for text recognition," 2022, *arXiv:2205.03873*.

[49] Z. Zheng and Y. Yang, "Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation," *Int. J. Comput. Vis.*, vol. 129, no. 4, pp. 1106–1120, Apr. 2021.

[50] B. Xie, L. Yuan, S. Li, C. H. Liu, and X. Cheng, "Towards fewer annotations: Active learning via region impurity and prediction uncertainty for domain adaptive semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 8058–8068.

[51] K. Sohn et al., "FixMatch: Simplifying semi-supervised learning with consistency and confidence," in *Proc. Neural Inf. Process. Syst.*, 2020, pp. 596–608.

[52] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "MixMatch: A holistic approach to semi-supervised learning," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 5050–5060.

[53] Y. Kim, J. Yim, J. Yun, and J. Kim, "NLNL: Negative learning for noisy labels," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2019, pp. 101–110.

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

[55] 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.

[56] A. Mishra, K. Alahari, and C. V. Jawahar, "Scene text recognition using higher order language priors," in *Proc. Brit. Mach. Vis. Conf.*, 2012, pp. 1–11.

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

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

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

[60] 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.

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

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

[63] 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.

[64] J. Sueiras, "Continuous offline handwriting recognition using deep learning models," 2021, *arXiv:2112.13328*.

[65] X. Xie, L. Fu, Z. Zhang, Z. Wang, and X. Bai, "Toward understanding WordArt: Corner-guided transformer for scene text recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 303–321.

[66] X. Xu, Z. Zhang, Z. Wang, B. Price, Z. Wang, and H. Shi, "Rethinking text segmentation: A novel dataset and a text-specific refinement approach," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 12045–12055.

[67] R. Litman, O. Anschel, S. Tsiper, R. Litman, S. Mazor, and R. Manmatha, "SCATTER: Selective context attentional scene text recognizer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11959–11969.

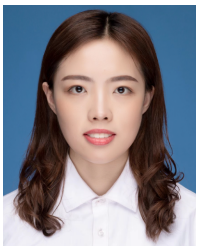
[68] M. Yang et al., "Reading and writing: Discriminative and generative modeling for self-supervised text recognition," in *Proc. 30th ACM Int. Conf. Multimedia*, Oct. 2022, pp. 4214–4223.

[69] B. Na, Y. Kim, and S. Park, "Multi-modal text recognition networks: Interactive enhancements between visual and semantic features," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 446–463.

[70] T. Zheng, Z. Chen, J. Bai, H. Xie, and Y.-G. Jiang, "TPS++: Attention-enhanced thin-plate spline for scene text recognition," in *Proc. 32nd Int. Joint Conf. Artif. Intell.*, Aug. 2023, pp. 1777–1785.

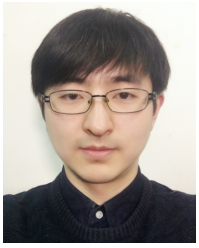
[71] B. Zhang, H. Xie, Y. Wang, J. Xu, and Y. Zhang, "Linguistic more: Taking a further step toward efficient and accurate scene text recognition," 2023, *arXiv:2305.05140*.

[72] W. Ahmed, P. Morerio, and V. Murino, "Cleaning noisy labels by negative ensemble learning for source-free unsupervised domain adaptation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 356–365.



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

Peng-Fei Zhang received the B.Sc. and M.S. degrees from Shandong University, Jinan, China, in 2015 and 2018, respectively, and the Ph.D. degree in computer science from The University of Queensland, Brisbane, QLD, Australia. He is currently a Postdoctoral Research Fellow with the School of Electrical Engineering and Computer Science, The University of Queensland. His research interests include machine learning, information retrieval, privacy protection, and multimedia analysis and search.



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. He has published over 20 papers on IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, ACM MM, SIGIR, WWW, and IJCAI. His research interests include machine learning, multimedia retrieval, and computer vision. He serves as a reviewer for the ACM International Conference on Multimedia, the International Joint Conference on Artificial Intelligence, the AAAI Conference on Artificial Intelligence, IEEE TRANSACTIONS ON CYBERNETICS, *Pattern Recognition*, and other prestigious conferences and journals.

Zi Huang (Senior Member, IEEE) received the B.Sc. degree from Tsinghua University, Beijing, China, and the Ph.D. degree in computer science from The University of Queensland, Brisbane, QLD, Australia. She is currently a Professor with the School of Electrical Engineering and Computer Science, The University of Queensland. Most of her publications have been published in leading conferences and journals, including ACM Multimedia, ACM SIGMOD, IEEE ICDE, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, *ACM Transactions on Information Systems*, and *ACM Computing Surveys*. Her current research interests include multimedia search, social media analysis, database, and information retrieval.

Xin-Shun Xu (Senior Member, IEEE) received the M.S. degree in computer science from Shandong University, China, in 2002, and the Ph.D. degree in computer science from the University of Toyama, Japan, in 2005. He joined the School of Computer Science and Technology, Shandong University, as an Associate Professor, in 2005, and the LAMDA Group, Nanjing University, China, as a Postdoctoral Fellow, in 2009. From 2010 to 2017, he was a Professor with the School of Computer Science and Technology, Shandong University. He is currently a Professor with the School of Software, Shandong University. He is the Founder and the Leader of the Machine Intelligence and Media Analysis (MIMA) Group, Shandong University. He has published in IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, AAAI, CIKM, IJCAI, MM, SIGIR, WWW, and other venues. His research interests include machine learning, information retrieval, data mining, and image/video analysis and retrieval. He also serves as a PC/SPC Member or a reviewer for various international conferences and journals, such as AAAI, CIKM, CVPR, ICCV, IJCAI, MM, IEEE TCSVT, IEEE TIP, IEEE TKDE, IEEE TMM, and IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE.