BTS: A Bi-lingual Benchmark for Text Segmentation in the Wild

Xixi Xu[1], Zhongang Qi[1]*, Jianqi Ma[4], Honglun Zhang[1], Ying Shan[1], Xiaohu Qie[2,3]
[1]ARC Lab, [2]Tencent PCG; [3]Tsinghua University; [4]The Hong Kong Polytechnic University
{axixixu,zhongangqi,honlanzhang,yingsshan,tigerqie}@tencent.com; jianqi.ma@connect.polyu.hk

Abstract

As a prerequisite of many text-related tasks such as text erasing and text style transfer, text segmentation arouses more and more attention recently. Current researches mainly focus on only English characters and digits, while few work studies Chinese characters due to the lack of public large-scale and high-quality Chinese datasets, which limits the practical application scenarios of text segmentation. Different from English which has a limited alphabet of letters, Chinese has much more basic characters with complex structures, making the problem more difficult to deal with. To better analyze this problem, we propose the Bi-lingual Text Segmentation (BTS) dataset, a benchmark that covers various common Chinese scenes including 14,250 *diverse and fine-annotated text images. BTS mainly focuses on Chinese characters, and also contains English words and digits. We also introduce Prior Guided Text Segmentation Network (PGTSNet) , the first baseline to handle bi-lingual and complex-structured text segmentation. A plug-in text region highlighting module and a text perceptual discriminator are proposed in PGTSNet to supervise the model with text prior, and guide for more stable and finer text segmentation. A variation loss is also employed for suppressing background noise under complex scene. Extensive experiments are conducted not only to demonstrate the necessity and superiority of the proposed dataset BTS, but also to show the effectiveness of the proposed PGTSNet compared with a variety of state-of-the-art text segmentation methods.*

1. Introduction

Text segmentation is a fundamental and important task in computer vision. Different from other segmentation tasks such as semantic segmentation and instance segmentation, it is required to parse text instead of objects from complex scenes. With the text mask, it can be applied to various downstream tasks including scene text removal for cover generation and material recreation, font style transfer for AI design, interactive text image editing, etc.

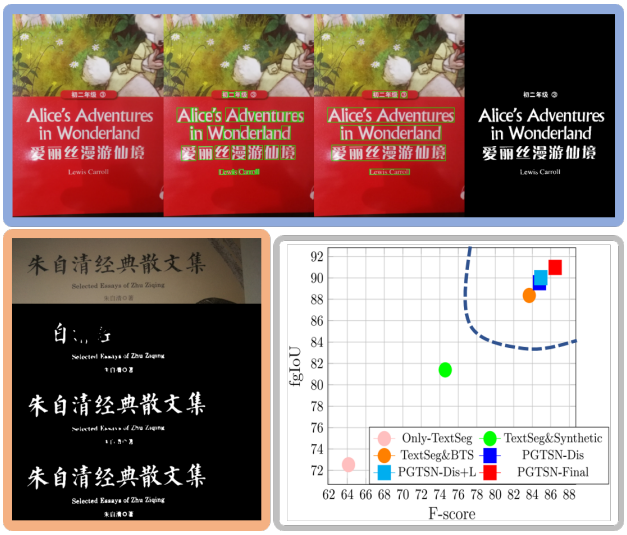

Figure 1. **The top block**: example images and annotations from the proposed BTS dataset. From left to right are images, character-level bounding boxes, word-level bounding boxes, and pixel-level segmentation masks. **The left bottom block**: qualitative results. From top to down are original image, result using the SOTA method (TexRNet [66]) trained on only English dataset (TextSeg [66]), result using TexRNet trained on TextSeg and our bi-lingual dataset BTS, and result using our proposed method PGTSNet trained on TextSeg and BTS. **The right bottom block**: quantitative results on the testing set of the bi-lingual dataset. The circle represents the SOTA method TextRNet; the square represents different variations of our proposed PGTSNet. The points below the dash line are results using only English dataset TextSeg, and using TextSeg with a synthetic bi-lingual dataset, respectively; the points above the dash line are results using TextSeg with our bi-lingual dataset BTS. The results demonstrate that the fine-annotated bi-lingual dataset is necessary, which improves the performance of the existing SOTA method with a large margin, and boosts the overall performance in the area of bi-lingual text segmentation to a new level.

Text segmentation presents two distinct characteristics. First, as the strokes and structure of text are looser and incoherent (unlike object segmentation), it is more challenging to capture fine-grained features of every stroke in a word. For example, some strokes like the lowercase *L* or the dot on

*Corresponding author.

Figure 2. Samples and their segmentation annotations of several mainstream scenes from the proposed BTS dataset, including the street sign, the banner, the couplet, the cover of book, the plaque, the shop sign, and the attraction.

the top of the lowercase *I*, or when the character shares similar features with common background such as the checkered floor or black dot, are easy to be ignored in pixel level segmentation. The situation can be further worse with the Chinese characters. Different from sequentially arranged English words, a Chinese character is formed with stroke combinations in spatial dimension, which thus leaves discontinuous segmentation hollow within the characters. Second, different from the semantic and instance segmentation which usually contain multiple categories, text segmentation is usually treated as a binary classification problem. It treats all different characters the same foreground category, and ignores the semantic variance contained in the characters. Xu *et al*. [66] has proved the significance of using the character prior in enhancing the English text segmentation. However, English has a small-sized alphabet of letters, while Chinese has much more basic characters (*e.g*., over $3,000$ commonly used ones) with complex structure formations, making the problem more difficult to deal with. How to utilize the priors of text to make better segmentation is worth exploring.

To obtain high-quality segmentation for down-stream tasks, sufficient well-annotated training data is necessary. However, modern text segmentation datasets as well as methods are still left behind. First introduced as a public challenge [27], text segmentation developed slowly in the past few years with few research works and datasets proposed [6, 7, 13]. Among whom, large-scale datasets are with unsatisfying labeling quality [6, 7]. In a smaller scale, TextSeg [66] is proposed to fills the blank of segmentation in the area of artistic design and text effects. However, all these datasets contain only common English characters and numbers, and few work studies Chinese characters without any Chinese large-scale and high-quality datasets released, which limits the practical application scenarios of text segmentation.

To fill the above-mentioned research gaps of limited character types and extend the text segmentation to support more scenes and languages, we propose Bi-lingual Text Segmentation (BTS), a new text segmentation dataset. The diversity of BTS can be described at three levels: (1) scene-level diversity: it covers common life scenes including street signs, shop signs, plaques, attractions, book covers, banners, and couplets; (2) image-level diversity: appearances and geometric variances caused by camera-captured settings and background distractions such as perspective, illumination, resolution, partly blocking, blur and so on, in total including $14,250$ fine-annotated text images; (3) character-level diversity: variances of character categories, up to $3,985$ classes including Chinese characters, English letters, digits, common punctuation with varied fonts and sizes. Image examples and their segmentation annotations in BTS dataset are shown in Fig. 1 and Fig. 2. From Fig. 1 we can see that the fine-annotated bi-lingual dataset can beat the synthetic bi-lingual dataset, improve the performance of the existing SOTA method with a large margin, and boost the overall performance in the area of bi-lingual text segmentation to a new level.

Most of the text segmentation methods inherit semantic or instance segmentation and perform mask level supervision, while unaware of the global structure information of the characters. Therefore we turn to the recognition model for prior guidance to help the model regain the global structure of a character. We propose a novel approach, named as the Prior Guided Text Segmentation Network (PGTSNet) to better deal with bi-lingual text segmentation with text prior guidance. In this study, the main contributions can be summarized in four folds:

- We propose BTS, the first large-scale bi-lingual text segmentation dataset that goes beyond English words and digits including also Chinese characters. BTS provides annotations of text region, transcripts and the text masks, and therefore can be used not only for text segmentation but also for text detection, recognition, and end-to-end text spotting. We prove the superiority of the proposed dataset BTS by comparing and analyzing the methods trained on BTS and a synthetic dataset.
- To better handle text distribution in different scenes, we propose a simple yet effective module to highlight the text region and serve as the prior to boost the text segmentation performance.

- We introduce a plug-in text recognition module as a prior to supervise for more stable and better text segmentation, whose advantage has been verified especially in the segmentation of large-sized text.
- We adopt the total variation loss in the text segmentation task, which exhibits the advantages in suppressing the ambient noises, and is able to supervise the PGTSNet to produce more smooth masks.

2. Related Work

2.1. Semantic and Instance Segmentation

Semantic segmentation aims to assign pixel-level labels in images. Traditional algorithms utilize the handcrafted features. With the development of convolution neural network, Fully Convolutional Networks (FCN) [42] and methods based on it [2, 10, 24, 71] achieve impressive performance. As the predictions of FCN are relatively coarse, several variants of the encoder-decoder structures [2, 11, 36, 48, 52, 71] are devised to improve it by fusing multilevel features. In addition, dilated convolution is introduced to enlarge the receptive field for better context capturing [8–10, 61, 70, 71]. For capturing long range context information, attention based models [58, 62] come into fashion, such as PSANet [72], DANet [18], CCNet [26], etc.

Instance segmentation further predicts distinct pixel labels for each object instance. The major milestone in this literature is Mask R-CNN [22], followed by many studies [38, 51] based on it. Other mainstream top-down methods are also proposed including [21, 29, 33, 64]. Apart from these top-down methods that first locate object bounding boxes and then segment their masks, bottom-up methods [5, 19, 41, 47, 65, 68] are the other branch in this field, where they first locate key points and then find edges as well as affinities to complete the segmentation.

2.2. Text Segmentation

Datasets play a vital role in the development of most computer vision research, especially in deep learning. In the early stages when only some small datasets are available for text segmentation, methods usually utilize handcrafted [1, 49] or low-level features [4, 14, 40], while Markov Random Field (MRF) based methods (e.g., [45]) are regarded as another fashion. Due to the lack of enough real data, weakly supervised methods [7, 46, 59] are proposed, trying to reduce the domain-shift between synthetic and real data, and enhance the model performance in real world with synthetic data.

Recently, models developed with deep learning techniques constantly upgrade the state-of-the-art in text segmentation. A three-stage CNN-based model [57] is introduced to detect, refine, and filter candidate text regions. SMANet adopts encoder-decoder structure from PSPNet [71] and utilizes a multi-scale attention module to assist segmentation. TexRNet [66] combines key features pooling and attention-based similarity checking to boost segmentation performance. The custom trimap loss and glyph discriminator are also introduced to assist the task. Mutually guided network [60] is devised to produce a polygon-level mask in one branch and a pixel-level text mask in the other, which can be trained via a semi-supervised learning strategy. However, most methods are only researched on latin-based benchmarks, while ignoring the segmentation in other widely-used languages, e.g., the Chinese hieroglyph characters. Therefore, a benchmark and baseline for both English and Chinese segmentation is necessary.

2.3. Text Detection and Text Recognition

Text detection aims at localizing text regions by polygon or rectangle boxes. Mainstream methods can be categorized into segmentation-based and regression-based methods. The former ones [16, 32, 50, 67] directly segment text regions and then generate bounding boxes from those regions. PixelLink [16], SSTD [50], PSENet [32], TextField [67], and DBNet [35] are several popular methods in this branch. The latter ones [34, 44, 73] take scene text as general objects, and predict the offsets of anchors or pixels. TextBoxes [34] extends SSD [39] to capture various text shapes by designing customed convolutional kernel and anchor box. RRPN [44] detects arbitrary-oriented scene texts by introducing rotation to anchors as well as RoI-Pooling in Faster R-CNN. Besides, several methods [3, 15, 43, 53, 63, 70] go further to predict character-level boxes.

Given an image patch containing a textline, text recognition aims to extract text from it. In general, it can be roughly divided into CTC-based methods [23, 25, 54, 56] and attention-based methods [12, 30, 31, 37, 55]. The former ones employ CNN to extract visual features and RNN to capture features sequence, which are trained end-to-end using the CTC loss [20]. The latter ones replace the CTC with the attention decoding mechanism. In addition, more explicit language modeling methods [17, 69] are proposed to explore internal interaction between vision and language.

3. The BTS Dataset

Compared with semantic and instance segmentation, text segmentation falls behind, one reason for which is lacking of large-scale and fine-annotated dataset. The synthetic labeled data may assist the training of the models. However, there is a gap between the distribution of the real labeled data and that of the synthetic labeled data, which cannot be neglected. Although there exist some weakly supervised methods [7, 46, 59] trying to reduce the distribution-shift, their labeling qualities still need further improvement to meet the requirements of training robust and high-precision

Table 1. The comparisons among a variety of representative datasets.

Dataset	Text Type	Images	Words	Chars	Masks	Char Classes	Language
ICDAR13 FST	Scene	462	1944	6620	Word,Char	36	English
COCO_TS	Scene	14690	139034	-	Word	36	English
MLT_S	Scene	6896	30691	-	Word	36	English
Total-Text	Scene	1555	9330	-	Word	36	English
TextSeg	Scene+Design	4024	15691	73790	Word,Word-Effect,Char	36	English
BTS(Ours)	Scene	14250	44280	209090	Word,Char	3985	Bi-lingual

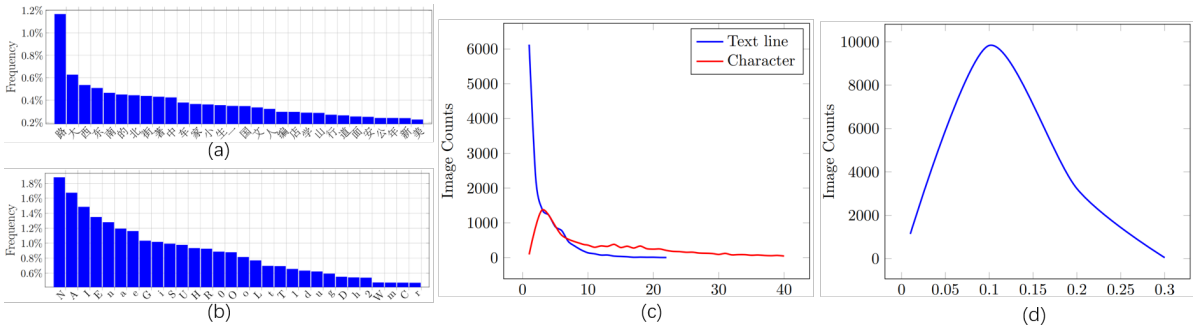

Figure 3. (a) An overview of the top 30 Chinese characters in BTS; (b) an overview of the top 30 English characters and digits in BTS. There are in total $209,090$ characters in BTS. The percentage of Chinese characters and punctuations is 66.4%; the percentage of English characters and digits is 33.6%. (c) an overview of the number of objects in text-level and character-level of BTS. The x-axes represents the number of text lines and characters per image. Most images contain $1-8$ textlines and $3-20$ characters; (d) an overview of the text coverage ratio against the image. The x-axes represents the text coverage ration against image. The curves are smoothed.

text segmentation models. Thus, recent works introduce some high-quality annotated datasets, based on which some novel models are proposed [66]. However, almost all the existing datasets and models focus only on English and digits, and few work studies Chinese text segmentation. Chinese has a much larger-scale alphabet of basic characters with complex structures and a variety of fonts. Thus, extensive and high-quality annotated examples are imperative for the research community on Chinese text vision, which will expand its practical application to more scenarios. In such a condition we introduce the large-scale bi-lingual text segmentation dataset BTS, which mainly focuses on Chinese characters, and also contains English and digits. BTS can be utilized for text detection, text recognition, text segmentation, and character-level detection. In this paper, we focus on its application to text segmentation.

3.1. Data Collection and Annotation

To ensure the representation and generalization of the dataset, we collect images from 7 different scenes, including street sign, shop sign, plaque, attraction, cover of book, banner, and couplet. First, these scenes cover several major scenes where texts appear in daily life from indoors to outdoors, to include diverse backgrounds, perspectives, lighting conditions, etc. Second, these scenes cover texts with diverse characteristics, e.g., the covers of books contain both printed and artistic fonts; the couplets contain both simplified and traditional Chinese characters, most of which are handwriting; the banners contain texts with non-rigid distortions and occlusions. Third, these scenes cover diverse levels of difficulty, e.g., it is easier to do text segmentation on street signs and plaques than on banners and couplets. The number of images: street signs-$3,761$; shop signs-$4,145$; plaques-$2,158$; attractions-$1,024$; covers of books-$2,070$; banners-601; couplets-491. We believe that varieties in these three perspectives can ensure the segmentation model to be well-trained with better generalization.

All images in BTS provide three-level annotations, including pixel-level mask, character-level and textline-level quadrilateral as well as transcription. *To the best of our knowledge, this is the first dataset with comprehensive annotations for text segmentation which contains Chinese characters.* The pixel-level mask annotation is a map shared the same size with the original image, where the pixels of the text regions are treated as the foreground and labeled as 1; other pixels are treated as the background and labeled as 0. The character-level and textline-level quadrilateral annotations are bounding boxes for characters and textlines, respectively, which record the coordinates of four vertices of the quadrilaterals. The transcription annotation records the recognition ground truth for each character and each textline. With these comprehensive annotations, the dataset can be applied to semantic segmentation for texts, instance segmentation for characters, text detection, and text recognition. The details are shown in Fig. 1.

We eliminate algorithms or out-of-the-box models for the labeling process to prevent some bad labeling cases. The annotation workflow is as follows. **1)** Images cleaning. Unqualified examples such as fuzzy images with unrecognizable characters and strokes will be filtered out. **2)** Manual annotation. All the images in BTS are manually annotated by humans in three levels, including the pixel-level,

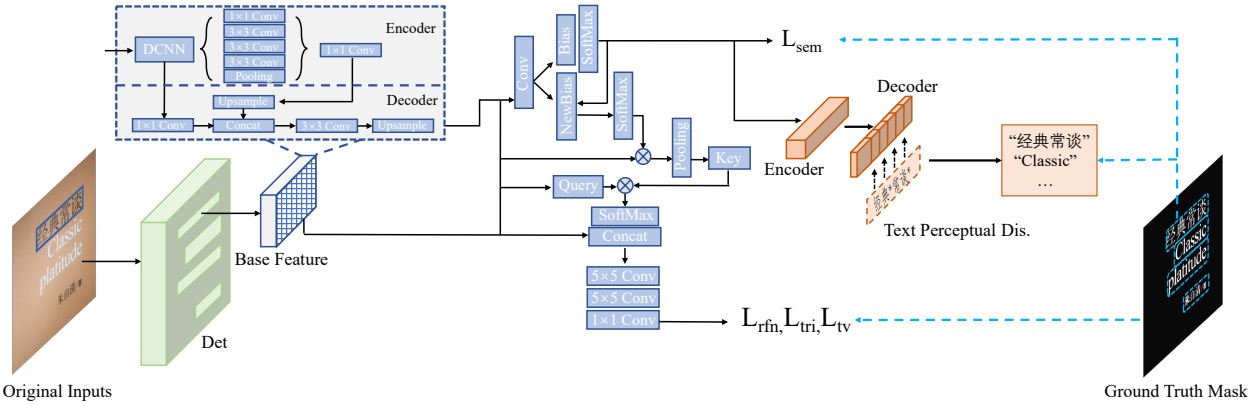

Figure 4. An overview of the proposed PGTSNet. The pipeline consists of a detection module, a feature extraction backbone with a refinement module, and a text perceptual discriminator.

the character-level, and the line-level annotations. **3)** Two rounds of quality checks. During the labeling process, annotators will cross check the annotations from each other; after the labeling process, several professional researchers will double check the annotations. The designed workflow ensures all annotations to be made in relatively high quality and benchmark to be highly-reliable.

3.2. Dataset Statistics

Tab. 1 illustrates the statistical comparisons conducted between BTS and five other representative text segmentation datasets, including ICDAR13 FST [27], MLT_S [7], COCO_TS [6], Total-Text [13], and TextSeg [66]. It shows that BTS contains the Chinese character classes, and thus results in the most class amount of $3,985$. BTS provides more comprehensive annotations than ICDAR13 FST, MLT_S, COCO_TS, and Total-Text. Compared with TextSeg, BTS only lacks of annotations for word-effect, while the size of BTS is much larger than that of TextSeg. The size of COCO_TS is the largest, but the annotations of COCO_TS are machine-generated instead of human-labeled. Therefore, BTS is largest text segmentation dataset of human-labeled images in Tab. 1. Fig. 3 (a)(b) show an overview of the top 30 Chinese characters and the top 30 English characters and digits in BTS, respectively. The percentage of Chinese characters and punctuations is 66.4%, and the percentage of English characters and digits is 33.6% in BTS. Fig. 3 (c) shows an overview of the number of objects in text-level and character-level in BTS. Most images contain $1-8$ textlines and $3-20$ characters.Fig. 3 (d) shows the distribution of the text coverage ratio. The $14,250$ images in BTS are split into training, validation, and testing sets of $10,188$, $2,696$, and 1,366 images, respectively, with a ratio of $7:2:1$.

4. The Prior Guided Text Segmentation Network

We also propose the Prior Guided Text Segmentation Network (PGTSNet) as a novel baseline for bi-lingual text segmentation. Fig. 4 shows the overview pipeline of PGTSNet, which consists of three components: 1) the detection module, e.g., DBNet, for highlighting the regions that may contain text; 2) the base text segmentation module to extract features from the input image and its highlighted regions; 3) a segmentation head with several loss functions including a character discriminative loss, a TV loss, and three items of pixel-level segmentation loss on text and its boundaries to guide the learning of the whole network.

4.1. Design Motivation

Inspired by the design principle of TexRNet which handles the unique challenges of distinguishing text segmentation from semantic segmentation, PGTSNet aims at figuring out the different characteristics between Chinese segmentation and English segmentation to improve the architecture design.

The biggest challenge that distinguishes representative hieroglyphics Chinese from latin English is the complex strokes. Especially for small texts in a large background, the features of hieroglyphics may easily be confused with the background. In this case, a text detection module that serves as a telescope to highlight the interested text region can avoid redundant amplification of irrelevant content, thus leading to better segmentation results.

However, the glyph discriminator of TexRNet has a fatal limitation when dealing the segmentation problem. First, it is a character-level discriminator, which requires extensive fine-grained annotations. As Tab. 1 shows, even for English, most of existing representative datasets cannot meet such character-level annotations, not to mention the more complicate bi-lingual case. Second, in bi-lingual and further multi-lingual scene, the number of characters is much larger, making the classification task far more difficult for a classification model (as a discriminator). Further, TexRNet lacks more fine-grained supervision for the complex seg-

mentation of hieroglyphics. Considering that the confusions between possible strokes and background noise are much more prevalent in bi-lingual scene, we further employed the total variation loss to obtain smoother predictions.

4.2. Network Structure

Given an input image $x \in \mathbb{R}^{H\times W\times 3}$, where H and W represent the height and width of the image, respectively, a plug-in detection module D is first employed to generate n candidate boxes $b_1, b_2, \ldots, b_n$. Image patches are cropped from the original image according to the boxes and combine into a batch $C_1, C_2, \ldots, C_n, x$. For each C_i, $C_i = C(b_i)$, where C is the cropping operation. After fed into a base segmentation module S, the output prediction maps $S(C_1), S(C_2), \ldots, S(C_n), S_x$ will be rearranged to one map x_{output} that shares the same shape with the input x. The rearrangement is conducted according to the position of each candidate patch. We use the DBNet as the detection module here, and more pipeline settings with different detectors will be analyzed in the supplementary material.

Another text prior that most previous methods neglect is the semantic information contained in the text. The segments of text lines should be perceptually recognizable and further recover the semantics of the text. More specifically, during training the ground truth bounding boxes of text lines are added as input to crop patches $p_1, p_2, \ldots, p_k$ from the output feature map, assuming that there are k text lines in x. A frozen recognizer is employed here to serve as the text discriminator. For each patch p_i, it is fed into the discriminator to obtain the discriminator loss $\mathcal{L}_{ctc}$, which indicates the confidences that these patches are recognizable. Here we adopt the ABINet [17] as the discriminator.

4.3. Customized Loss

There are five losses terms employed in PGTSNet. Three among them, i.e., $\mathcal{L}_{sem}$, $\mathcal{L}_{rfn}$ and $\mathcal{L}_{tri}$ are inherited from the base segmentation network [66]. Besides, $\mathcal{L}_{ctc}$ is used for evaluating the text semantics and $\mathcal{L}_{tv}$ is responsible for a smoother prediction.

Similar to most existing text segmentation models, the output map from initial prediction x_{output} can be supervised by ground truth labels x_{gt} by the cross entropy loss,

$$\mathcal{L}_{sem} = -\sum_i x_{gt_i} \log\left(x_{output_i}\right), \tag{1}$$

PGTSNet also adopts the other two loss terms from the base segmentation task, namely $\mathcal{L}_{rfn}$ and $\mathcal{L}_{tri}$. After refinement in the base segmentation, the final output x_{rfn} is supervised by the ground truth x_{gt} with the cross entropy loss and the trimap loss as follows.

$$\mathcal{L}_{rfn} = -\sum_i x_{gt_i} \log\left(x_{rfn_i}\right), \tag{2}$$

$$\mathcal{L}_{tri} = \mathrm{WCE}\left(x_{rfn}, x_{gt}, w_{tri}\right) \tag{3}$$

$$\mathrm{WCE}(x, y, w) = -\frac{\sum_{j=1}^{n} w_j \sum_{i=1}^{c} x_{i,j} \log\left(y_{i,j}\right)}{\sum_{j=1}^{n} w_j} \tag{4}$$

where $WCE(x, y, w)$ is a cross-entropy loss between x and y and is only calculated for pixels at the text boundaries.

In addition, for the text discriminator, we adopt the commonly-used Connectionist Temporal Classification (CTC) loss in recognition task. We briefly explain the effectiveness of it and how it works on the segmentation network.

$$\begin{aligned}\mathcal{L}_{ctc} &= \mathrm{ctc}(O(p_i), t_{gt}) \\ &= \mathrm{ctc}\left(O\left(S\left(x_i\right)\right), t_{gt}\right)\end{aligned} \tag{5}$$

$$\frac{\partial \mathcal{L}_{ctc}}{\partial w} = \frac{\partial \mathcal{L}_{ctc}}{\partial O_w}\frac{\partial O_w}{\partial S_w}\frac{\partial S_w}{\partial w} \tag{6}$$

Here, O and S denote the recognition and segmentation network respectively. By using the chain rule, the gradient of this loss to the network parameters can be expanded. Although the parameters of recognition network are not updated, it is still a auxiliary tool to calculate the text perceptual for the patches from the output of the segmentation network. In this case, the segmentation and recognition networks can cooperate with each other. If the text perceptual of a patch is low, which means the quality and readability of the segmentation output is bad, the loss will become larger and give more punishments to the segmentation network.

Besides, for text segmentation, it is intuitive that the strokes should be relatively coherent and smooth, so a total variation loss is introduced to further suppress segmentation noises.

$$\mathcal{L}_{tv}(x) = \sum_{i,j}\left(\left(x_{i,j+1} - x_{ij}\right)^2 + \left(x_{i+1,j} - x_{ij}\right)^2\right)^{\frac{\beta}{2}} \tag{7}$$

The final loss is weighted combination of the above terms as follows.

$$\mathcal{L}_{\text{final}} = \alpha\mathcal{L}_{\text{sem}} + \beta\mathcal{L}_{rfn} + \gamma\mathcal{L}_{\text{ti}} + k\mathcal{L}_{ctc} + l\mathcal{L}_{tv} \tag{8}$$

where the default weights are $\alpha = 1.0, \beta = 0.5, \gamma = 0.5, k = 0.01, l = 1.0$. In different experiments, they can be tuned according to the training models for better performances and the principle is to balance different terms to a relatively closer magnitude.

5. Experiments

We conduct the experiments from two aspects: one is to analyze the performances of bi-lingual text segmentation on the proposed dataset BTS and on other datasets including

Figure 5. Examples of the synthetic datasets. To better show the texts, some examples are cropped.

TextSeg and a synthetic dataset, whose results demonstrate that a high-quality and fine-annotated dataset is necessary and valuable to help boosting the model's performance and expanding the application scenarios; the other is to evaluate the performance of the proposed approach PGTSNet, whose results show that by introducing the text prior, PGTSNet beats the other state-of-the-art methods on the task of bi-lingual text segmentation with at least 2.67% promotion in fgIoU and 1.74% promotion in F-score.

5.1. Datasets

Three datasets are utilized in the experiments, including TextSeg with the most comprehensive annotations for English as shown in Tab. 1, a synthetic bi-lingual dataset, and BTS. The details of BTS are described in the Sec. 3. For the synthetic bi-lingual dataset, it is mainly built for validating the necessity of human labeled data. We try our best to mimic the distributions of the data in real scenarios, by considering as many factors as possible.

The main components to synthesize text segmentation images include background, number of text, text location, text size, font, corpus, color, and noise. We collect $10,000$ images without any text from a variety of video frames as the backgrounds. To align with BTS, the text corpus of the synthetic dataset are formed by random sampling from the textline annotations of BTS. 11 kinds of common fonts with text size ranging from 35 to 60 are applied to the bi-lingual characters. The color of the text is randomly sampled from the RGB color space. Each background image is pasted with three text lines, whose locations are randomly generated. We also add various degradation to the synthetic images, including the Gaussian noise, the salt and pepper noise, Poisson noise, perspective transformation, color reverse, blur, etc. The dataset contains $20,000$ synthetic text segmentation images in total.

Several examples of the synthetic dataset are shown in Fig. 5. We can see that 1) although we can generate synthetic examples with complex background, the text may have no connection with the background. The characteristics of synthetic images are similar to those of the movie frames with subtitles or live comments. 2) The boundaries of the texts have no interaction with the background, which can be distinguished easily. 3) No occlusions or illumination changes caused by the environment of the background occur to the text. Therefore, the synthetic images are much easier than the scene images, and can only handle limited scenarios of the segmentation task.

Figure 6. Qualitative comparison between PGTSNet and TexRNet. From top to bottom, the rows show the input image, the predicted mask of PGTSNet, the inpainting results of PGTSNet with DeepfillV2, the predicted mask of TexRNet, and the inpainting results of TexRNet with the same setting of DeepfillV2, respectively.

Table 2. Comparison experiments of training the same model with different datasets.

Data used	fgIoU	F-score
Only synthetic	34.52	21.93
Only TextSeg	64.13	72.53
TextSeg & synthetic	74.59	81.40
TextSeg & BTS	**83.68**	**88.36**

5.2. Implementation Details

There is no detection module during training in order to accelerate the training process. Instead, local patches randomly cropped from the images (may be randomly scaled) are fed into network. The bounding box annotations and segmentation ground truth also need to be modified accordingly. The base module is initialized by ImageNet pre-trained model. SGD Optimizer is adopted with weight decay of $5e^{-4}$. All methods are trained for $22,000$ iterations or so, until the loss converges. For evaluation, foreground Intersection-over-Union (fgIoU) and F-score measurement on foreground pixels are utilized as in [14, 28].

5.3. Synthetic Data and Real Data Comparisons

This section compares the performances of the same model trained on synthetic data, TextSeg, and BTS, respectively. The results on the test set of BTS are shown in Tab. 2. We compare the effect of different datasets on our base model PGTSNet(base). All the experiments settings are kept unchanged during training except the datasets. It is shown that the improvement brought by the synthetic data is limited and far from requirements.

Table 3. The ablation studies of PGTSNet on BTS. All the training settings of these methods including the training data and the backbone are the same. The column "Dis" denotes whether the text perceptual discriminator is included. The $\mathcal{L}_{tv}$ and the "DET" represent whether the total variation loss and the detection module are activated, respectively.

Method	Dis	$\mathcal{L}_{tv}$	DET	fgIoU	F-score
PGTSNet(base)				83.68	88.36
PGTSNet	✓			84.78	89.55
PGTSNet	✓	✓		84.93	90.02
PGTSNet(final)	✓	✓	✓	**86.48**	**90.98**

Table 4. The comparison experiments of PGTSNet with the state-of-the-art methods.

Method	fgIoU	F-score
DeeplabV3+	71.15	79.60
HRNetV2-W48	81.84	86.17
HRNetV2-W48+OCR	82.76	86.67
TexRNet(DeeplabV3+,no classifier)	83.68	88.36
TexRNet(DeeplabV3+,with classifier)	83.81	89.24
PGTSNet(final)	**86.48**	**90.98**

5.4. Ablation Study

We conduct ablation studies on three key components of the proposed PGTSNet: the detection module, the text perceptual discriminator, and the total variation loss, whose results are shown in Tab. 3. All methods are trained on BTS training set combined with TextSeg training set and validation set. The validation set of BTS is used for evaluation. The backbone of all these methods is Deeplab V3+. The base version of PGTSNet(base) is the TexRNet. With the help of key components, the fgIoU and F-score increase consistently. The complete version of PGTSNet(final) achieves the best performance, with around 2.8% and 2.62% increase in fgIoU and F-score compared with the base version of PGTSNet(base), respectively.

5.5. Comparisons with State-of-the-Art Methods

In this section, PGTSNet is compared against four representative state-of-the-art text and semantic segmentation methods, including Deeplab V3+, HRNetV2-W48, HRNetV2-W48 + Object-Contextual Representations (OCR), and TexRNet. All these methods are retrained on the training set of TextSeg as well as BTS and evaluated on the test set of BTS. As Tab. 4 shows, the proposed PGTSNet outperforms other methods by significant margins.

5.6. Applications and Discussions

With a high-quality text segmentation mask, downstream tasks such as text removal and text style transfer can obtain more beneficial information and achieve much better results. For example, in the text removal task, we feed the segmentation mask along with the original image into an in-painting network to hallucinate a text-free image. As the in-painting network learns to recover the pixels categorized to be foreground in the segmentation mask, any part of strokes wrongly classified as background may be neglected and produce the artifacts. Fig. 6 shows the qualitative comparisons between PGTSNet and TexRNet. The inpainting method we adopt here is DeepfillV2, one of the state-of-the-art method. It is shown that the mask of PGTSNet contains fewer background noises and better captures the complete strokes of characters, and the inpainted image suffers less from halo as well. More cases can be viewed in the supplementary material.

Figure 7. Examples of the text segmentation on the application of expression recreation.

Another application is the expression recreation as shown in Fig. 7. Dynamic expression Gifs are widely used in daily chatting scenarios and there are abundant materials available that may be recreated to convey new semantics. PGTSNet can extract the precise segmentation masks for various and even tiny characters in those Gifs, which is of great benefits to subsequent recreation procedures.

6. Conclusion

In this paper, we construct a large-scale bi-lingual text segmentation dataset named BTS, which contains $14,250$ images with $44,280$ textlines and $209,090$ characters. With the comprehensive annotations, it can be used for training and evaluation of textline-level and character-level detection, recognition and segmentation. To the best of our knowledge, it is the first bi-lingual dataset for text segmentation. All the data will be released for further academic researches following the predefined protocol. Further, we propose a prior guided text segmentation network with a detection module, a text perceptual discriminator, and a smooth loss, to reveal the unique challenges that distinguish bi-lingual segmentation from universal text segmentation. The experimental results show the effectiveness of the proposed approach compared with the state-of-the-art methods.

References

[1] Binarization of historical document images using the local maximum and minimum. In *IAPR workshop on document analysis systems;DAS 2010*, 2011. 3

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis Machine Intelligence*, pages 1–1, 2017. 3

[3] Y. Baek, B. Lee, D. Han, S. Yun, and H Lee. Character region awareness for text detection. In *IEEE*, 2019. 3

[4] B. Bai, F. Yin, and C. L. Liu. [ieee 2014 11th iapr international workshop on document analysis systems (das) - tours, france (2014.4.7-2014.4.10)] 2014 11th iapr international workshop on document analysis systems - a seed-based segmentation method for scene text extraction. pages 262–266, 2014. 3

[5] M. Bai and R. Urtasun. Deep watershed transform for instance segmentation. *arXiv*, 2016. 3

[6] S. Bonechi, P. Andreini, M. Bianchini, and F. Scarselli. *COCO-TS Dataset: Pixel–Level Annotations Based on Weak Supervision for Scene Text Segmentation*. Artificial Neural Networks and Machine Learning – ICANN 2019: Image Processing, 2019. 2, 5

[7] S. Bonechi, M. Bianchini, F. Scarselli, and P. Andreini. Weak supervision for generating pixel–level annotations in scene text segmentation. *Pattern Recognition Letters*, 138, 2020. 2, 3, 5

[8] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *Computer Science*, (4):357–361, 2014. 3

[9] L. C. Chen, G Papandreou, I Kokkinos, K Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2018. 3

[10] L. C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. 2017. 3

[11] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. *Springer, Cham*, 2018. 3

[12] Z. Cheng, B. Fan, Y. Xu, Z. Gang, and S. Zhou. Focusing attention: Towards accurate text recognition in natural images. In *IEEE International Conference on Computer Vision*, 2017. 3

[13] C. K. Ch'Ng and C. S. Chan. Total-text: A comprehensive dataset for scene text detection and recognition. *IEEE*, 2018. 2, 5

[14] A. Clavelli, D. Karatzas, and J. Llados. A framework for the assessment of text extraction algorithms on complex colour images. In *IAPR workshop on document analysis systems;DAS 2010*, 2011. 3, 7

[15] Y. Cong, B. Xiang, S. Nong, X. Zhou, and Z. Cao. Scene text detection via holistic, multi-channel prediction. 2016. 3

[16] D. Dan, H. Liu, X. Li, and C. Deng. Pixellink: Detecting scene text via instance segmentation. 2018. 3

[17] S. Fang, H. Xie, Y. Wang, Z. Mao, and Y. Zhang. Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition. 2021. 3, 6

[18] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu. Dual attention network for scene segmentation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[19] N. Gao, Y. Shan, Y. Wang, X. Zhao, Y. Yu, M. Yang, and K. Huang. Ssap: Single-shot instance segmentation with affinity pyramid. 2019. 3

[20] Alex Graves, Santiago Fernández, and Faustino Gomez. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *International Conference on Machine Learning*, 2006. 3

[21] B. Hariharan, P Arbeláez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In *European Conference on Computer Vision*, 2014. 3

[22] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn, 2018. 3

[23] Pan He, Weilin Huang, Yu Qiao, Chen Change Loy, and Xiaoou Tang. Reading scene text in deep convolutional sequences. *AAAI Press*, 2015. 3

[24] P. Hu, F. C. Heilbron, O. Wang, Z. Lin, and F. Perazzi. Temporally distributed networks for fast video semantic segmentation. 2020. 3

[25] W. Hu, X. Cai, J. Hou, S. Yi, and Z. Lin. Gtc: Guided training of ctc towards efficient and accurate scene text recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(7):11005–11012, 2020. 3

[26] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu. Ccnet: Criss-cross attention for semantic segmentation. In *International Conference on Computer Vision*. 3

[27] D. Karatzas, P. P. Roy, and L D. Icdar 2011 robust reading competition. In , 2011. 2, 5

[28] Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, and Lluis Pere De Las Heras. Icdar 2013 robust reading competition. In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, 2013. 7

[29] A. Kirillov, R. Girshick, K. He, and P Dollár. Panoptic feature pyramid networks. 2019. 3

[30] C. Y. Lee and S. Osindero. Recursive recurrent nets with attention modeling for ocr in the wild. In *IEEE Conference on Computer Vision Pattern Recognition*, 2016. 3

[31] Hui Li, Peng Wang, Chunhua Shen, and Guyu Zhang. Show, attend and read: A simple and strong baseline for irregular text recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:8610–8617, 2019. 3

[32] X. Li, W. Wang, W. Hou, R. Z. Liu, T. Lu, and J. Yang. Shape robust text detection with progressive scale expansion network. 2018. 3

[33] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei. Fully convolutional instance-aware semantic segmentation. In *Computer Vision Pattern Recognition*, 2017. 3

[34] Minghui Liao, Baoguang Shi, Xiang Bai, Xinggang Wang, and Wenyu Liu. Textboxes: A fast text detector with a single deep neural network. *CoRR*, abs/1611.06779, 2016. 3

[35] M. Liao, Z. Wan, C. Yao, K. Chen, and X Bai. Real-time scene text detection with differentiable binarization. 2019. 3

[36] T. Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[37] R Litman, O. Anschel, S. Tsiper, R Litman, S. Mazor, and R Manmatha. Scatter: Selective context attentional scene text recognizer. 2020. 3

[38] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. Path aggregation network for instance segmentation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[39] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*, 2016. 3

[40] X. Liu and J. Samarabandu. Multiscale edge-based text extraction from complex images. In *2006 IEEE International Conference on Multimedia and Expo*, 2013. 3

[41] Y. Liu, S. Yang, B. Li, W. Zhou, J. Xu, H. Li, and Y. Lu. Affinity derivation and graph merge for instance segmentation. *Springer, Cham*, 2018. 3

[42] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):640–651, 2015. 3

[43] P. Lyu, M. Liao, C. Yao, W. Wu, and X. Bai. Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. 2018. 3

[44] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, and X. Xue. Arbitrary-oriented scene text detection via rotation proposals. *IEEE Transactions on Multimedia*, 2018. 3

[45] Anand Mishra, Karteek Alahari, and C. V. Jawahar. An mrf model for binarization of natural scene text. In *International Conference on Document Analysis Recognition*, 2011. 3

[46] N. Nayef, Y. Fei, I. Bizid, H. Choi, and J. M. Ogier. Icdar2017 robust reading challenge on multi-lingual scene text detection and script identification - rrc-mlt. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, 2017. 3

[47] D. Neven, B De Brabandere, M. Proesmans, and L Van Gool. Instance segmentation by jointly optimizing spatial embeddings and clustering bandwidth. *IEEE*, 2019. 3

[48] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. *arXiv preprint arXiv:1505.04366*, 2015. 3

[49] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems Man Cybernetics*, 9(1):62–66, 2007. 3

[50] H. Pan, W. Huang, H. Tong, Q. Zhu, and X. Li. Single shot text detector with regional attention. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017. 3

[51] S. Qiao, L. C. Chen, and A. Yuille. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. *arXiv*, 2020. 3

[52] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *Springer International Publishing*, 2015. 3

[53] B. Shi, X. Bai, and S. Belongie. Detecting oriented text in natural images by linking segments. In *IEEE Computer Society*, 2017. 3

[54] B. Shi, B. Xiang, and Y. Cong. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 39(11):2298–2304, 2016. 3

[55] B. Shi, M. Yang, X. Wang, L. Pengyuan, Y. Cong, and B. Xiang. Aster: An attentional scene text recognizer with flexible rectification. *IEEE Transactions on Pattern Analysis Machine Intelligence*, PP:1–1, 2018. 3

[56] B. Su and S. Lu. Accurate recognition of words in scenes without character segmentation using recurrent neural network. *Pattern Recognition*, 63:397–405, 2017. 3

[57] Y. Tang and X. Wu. Scene text detection and segmentation based on cascaded convolution neural networks. *IEEE Transactions on Image Processing*, 2017. 3

[58] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv*, 2017. 3

[59] A. Veit, T. Matera, L. Neumann, J. Matas, and S. Belongie. Coco-text: Dataset and benchmark for text detection and recognition in natural images. 2016. 3

[60] Chuan Wang, Shan Zhao, Li Zhu, Kunming Luo, Yanwen Guo, Jue Wang, and Shuaicheng Liu. Semi-supervised pixel-level scene text segmentation by mutually guided network. *IEEE Transactions on Image Processing*, 30:8212–8221, 2021. 3

[61] J. Wang, K. Sun, T. Cheng, B. Jiang, and B. Xiao. Deep high-resolution representation learning for visual recognition. 2019. 3

[62] X Wang, R. Girshick, A. Gupta, and K. He. Non-local neural networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[63] L. Xing, Z. Tian, W. Huang, and M. R. Scott. Convolutional character networks. *IEEE*, 2019. 3

[64] Y. Xiong, R. Liao, H. Zhao, R. Hu, M. Bai, E. Yumer, and R. Urtasun. Upsnet: A unified panoptic segmentation network. 2019. 3

[65] X. Xu, M. T. Chiu, T. S. Huang, and H. Shi. Deep affinity net: Instance segmentation via affinity. 2020. 3

[66] X. Xu, Z. Zhang, Z. Wang, B. Price, and H. Shi. Rethinking text segmentation: A novel dataset and a text-specific refinement approach. 2020. 1, 2, 3, 4, 5, 6

[67] Yongchao Xu, Yukang Wang, Wei Zhou, Yongpan Wang, and Xiang Bai. Textfield: Learning a deep direction field for irregular scene text detection. *IEEE Transactions on Image Processing*, PP(99):1–1, 2019. 3

[68] T. J. Yang, Maxwell D Collins, Y. Zhu, J. J. Hwang, T. Liu, X. Zhang, V Sze, G. Papandreou, and L. C. Chen. Deeperlab: Single-shot image parser. 2019. 3

[69] D. Yu, X. Li, C. Zhang, T. Liu, and E. Ding. Towards accurate scene text recognition with semantic reasoning networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[70] F Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. 2016. 3

[71] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *IEEE Computer Society*, 2016. 3

[72] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and J. Jia. Psanet: Point-wise spatial attention network for scene parsing. In *European Conference on Computer Vision*, 2018. 3

[73] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang. East: An efficient and accurate scene text detector. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3