Sequential Deformation for Accurate Scene Text Detection

Shanyu Xiao[1[0000−0003−3025−6157]], Liangrui Peng[1[0000−0001−7793−1039]], Ruijie Yan[1[0000−0002−2743−8470]], Keyu An[1[0000−0003−0040−0883]], Gang Yao[1[0000−0001−9311−0337]], and Jaesik Min[2[0000−0002−0007−0637]]

[1] Beijing National Research Center for Information Science and Technology Department of Electronic Engineering, Tsinghua University, Beijing, China
[2] Hyundai Motor Group AIRS Company, Seoul, Korea
xiaosy19@mails.tsinghua.edu.cn, penglr@tsinghua.edu.cn
{yrj17, aky19, yg19}@mails.tsinghua.edu.cn, jaesik.min@hyundai.com

Abstract. Scene text detection has been significantly advanced over recent years, especially after the emergence of deep neural network. However, due to high diversity of scene texts in scale, orientation, shape and aspect ratio, as well as the inherent limitation of convolutional neural network for geometric transformations, to achieve accurate scene text detection is still an open problem. In this paper, we propose a novel sequential deformation method to effectively model the line-shape of scene text. An auxiliary character counting supervision is further introduced to guide the sequential offset prediction. The whole network can be easily optimized through an end-to-end multi-task manner. Extensive experiments are conducted on public scene text detection datasets including ICDAR 2017 MLT, ICDAR 2015, Total-text and SCUT-CTW1500. The experimental results demonstrate that the proposed method has outperformed previous state-of-the-art methods.

Keywords: Scene Text Detection, Deep Neural Network, Sequential Deformation

1 Introduction

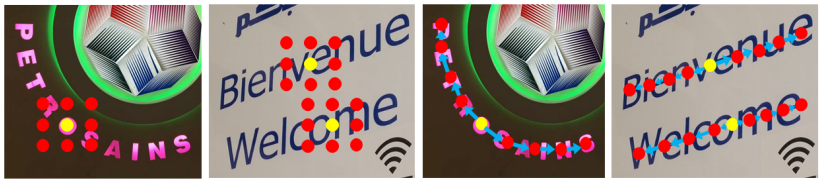
Scene text detection has attracted growing research attention in the computer vision field due to its wide range of real-world applications including automatic driving navigation, instant translation and image retrieval. Scene text's uniqueness of high diversity in geometric transformations including scale, orientation, shape and aspect ratio also makes it distinct from generic objects. It is obvious that scene text detection is a challenging research topic.

Recently, the community has witnessed substantial advancements in scene text detection [1,18,20,26,37–39,50], especially after the emergence of deep neural network. For scene text detection, a straightforward approach is to model text instance (word or text line) as a special kind of object and adopt frameworks of generic object detection, such as SSD [19] and Faster R-CNN [6]. These

methods yield great performance on standard benchmarks. However, their performance and generalization ability are undermined by standard convolution's fixed receptive field and limited capability for large geometric transformations.

Text instance is composed of similar components (e.g. text segments, characters and strokes), where component is spatially smaller and has less geometrical transformations. Consequently, some methods [1, 31] predict text components rather than the whole text instance. As pixel can be regarded as the finest-grained component, many methods [15, 39] localize text based on instance segmentation. These methods are more flexible in modeling, and have a lower requirement for the receptive field, achieving remarkable results on localizing texts with arbitrary shape. Nevertheless, an additional component grouping operation like pixel clustering or segment connecting is always indispensable, where error propagation from wrong component prediction and lack of end-to-end optimization make barriers for the optimal performance.

Component detecting and grouping are also exploited by human visual system [43]. Our eyes first localize one endpoint of a text instance, then sequentially sweeps through the text center line and gazes only a part of the text at one time. Finally, we group different parts into text instance along the sweeping path.



(a) (b) (c) (d)

Fig. 1. Demonstration of the sampling locations for standard convolution and the proposed SDM. For clearer visualization, all sampling locations are mapped on input images. (a)(b) regular sampling locations in standard convolution. The yellow point indicates the center location of convolution. (c)(d) Sequential sampling procedure of our SDM. The yellow point indicates the start location, and each blue arrow indicates the predicted offset of one iteration. Two deformation branches are used in our work.

Inspired by above observations, we propose an end-to-end trainable Sequential Deformation Module (SDM) for accurate scene text detection, which sequentially groups feature-level components to effectively extend the modeling capability for text's geometrical configuration and learn informative instance-level semantic representations along text line. The SDM first samples features iteratively from a start location. SDM runs densely, regarding each integral location on the input feature map as the start location to fit for unique geometric configurations for different instances. As depicted in Fig. 1, by performing sampling in a sequential manner, a much larger effective receptive field than the standard convolutional layer could be achieved. After that, SDM performs

weighted summation on all sampled features to aggregates features and capture the adaptive instance-level representations without complicated grouping post-processing. Besides, we introduce an auxiliary character counting supervision to guide SDM's sequential sampling and learn richer representations. The character counting task is modeled as a sequence-to-sequence problem [33], and the counting network receives all the SDM's sampled features and predict a valid sequence, whose length is expected to equal the character number of corresponding text instance.

The main contributions of this work are three-fold: (1) We propose a novel end-to-end sequential deformation module for accurate detection of arbitrary-shaped scene text, which adaptively enhances the modeling capability for text's geometric configuration and learns informative instance-level semantic representations; (2) We introduce an auxiliary character counting supervision which facilitates the sequential offset predicting and learning of generic features; (3) Integrating the sequential deformation module and auxiliary character counting into Mask R-CNN, the whole network is optimized through an end-to-end multi-task manner without any complicated grouping post-processing. Experiments on benchmarks for multi-lingual, oriented, and curved text detection demonstrate that our method achieves the state-of-the-art performance.

2 Related Work

Scene text detection has been widely studied in the last few years, especially with the popularity of deep learning. In this section, we review related works of two different categories of deep learning based methods according to their modeling granularity, then we look back on relevant works for learning spatial deformation in the convolutional neural network.

Instance-level detection methods [14, 45, 50] follow the routine of generic object detection, viewing the text instance as a specific kind of object. TextBoxes [14] modifies SSD [19] by adding default boxes and filters with larger aspect ratios to handle the text's significant variation of aspect ratio. EAST [50] and Deep Regression [9] directly regress the rotated rectangles or quadrangles of text without the priori of anchors. SPCNET [45] augments Mask R-CNN [7] with the guidance of semantic information and sharing FPN, suppressing false positive detections. These methods achieve excellent performances on standard benchmarks but face problems such as CNN's limited receptive field and incapability for geometric transformation.

Component-level methods [5, 15, 26, 31, 34, 39] decompose instance into components such as characters, text segments or the finest-grained pixels, addressing the problems faced by instance-level modeling. SegLink [31] decomposes text into locally detectable segments and links and combines them into the final oriented detection. TextDragon [5] describes text's shape with a series of local quadrangles to adaptively spot arbitrary-shaped texts. PAN [40] adopts a learnable post-processing implemented by semantic segmentation and embedding to precisely aggregate text pixels. Tian et al. [35] propose to learn shape-aware pixel

embedding to ease separating adjacent instances and detecting large instances. For these methods, a grouping post-processing is required, where the error propagation of wrong component prediction and the lack of end-to-end training could harm the robustness.

Spatial deformation methods [3,12,38] enable the network to adaptively capture the geometric transformations. STN [12] rectifies the image or feature maps via global parametric transformations. Deformable ConvNet [3] augments the spatial sampling locations in convolutional layers with additional predicted offsets. ITN [38] also augments convolution but constrains it as affine transformation to learn the geometry-aware representation for scene text.

Different from existing methods, a sequential deformation module to group feature components is proposed in our paper, adaptively enhancing the instance-level Mask R-CNN without complicated grouping post-processing.

3 Methodology

In this section, we first elaborate the sequential deformation module and auxiliary character counting task. Then we describe the Mask R-CNN equipped with sequential deformation module.

3.1 Sequential Deformation Module

For a standard convolution with weight w and the input feature map x, it first samples features on x using a fixed rectangular sampling grid $\mathcal{R} = \{\mathbf{p}_1, ..., \mathbf{p}_N\}$ (e.g 3×3 grid $\{(-1,-1),(-1,0),...,(0,1),(1,1)\}$ with $N = 9$). Then the weighted summation of sampled features is calculated using weight w. For every location $\mathbf{p}$ on the output feature map y, we have:

$$\mathrm{y}(\mathbf{p}) = \sum_{n=1}^{N} \mathrm{w}(n) \cdot \mathrm{x}(\mathbf{p} + \mathbf{p}_n), \tag{1}$$

As depicted in Fig. 1, the standard convolution is insufficient in scene text detection because of the mismatch of shape and size between the fixed receptive field and text instance. On one hand, to capture the whole instance, the fixed rectangular receptive field is required to completely cover the text's circumscribed rectangle rather than only the text region, while much undesired background information is included. On the other hand, the fixed-sized receptive field is incapable of well-extracting representations for instances with different scale and aspect ratio. This results in imprecise classification and regression, especially for instance-level models that detect text in one or a few stages.

Detecting text component, which is spatially smaller and less geometrically transformed, relieves the above-mentioned problems. Inspired by the insight of detecting text component and spatial deformation learning in CNN, we propose an end-to-end trainable Sequential Deformation Module (SDM). SDM first

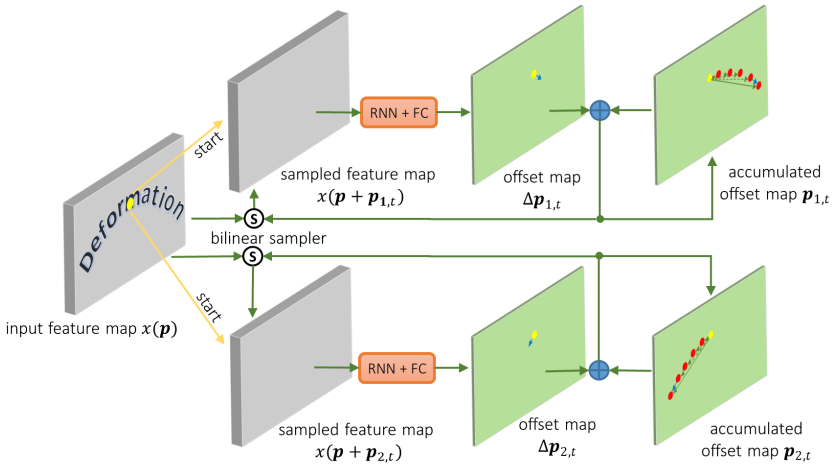


Fig. 2. Illustration of sequential sampling in SDM. Since the SDM runs densely, the notations of all 3D tensor are represented by the corresponding 1D vector at location $\mathbf{p}$. From each integral start location $\mathbf{p}$, SDM samples features along two separate sampling paths.

performs sampling in a sequential manner, and then, like the standard convolution, SDM performs weighted summation on all sampled features to aggregates features and capture the adaptive instance-level representations. The procedure of sequential sampling is illustrated in Fig. 2. Regarding each integral location $\mathbf{p} \in \{(0,0), ..., (H-1, W-1)\}$ on input feature map x (height H and width W) as the start location, the relative sampling locations $\mathcal{S} = \{\mathbf{p}_t | \ t = 1, ..., T\}$ are sequentially generated by offsets accumulation:

$$\mathbf{p}_{t+1} = \mathbf{p}_t + \triangle \mathbf{p}_t, \ t = 0, ..., T-1 \tag{2}$$

where $\mathbf{p}_0 = (0,0)$, $\triangle \mathbf{p}_t$ denotes the current 2D offset and T denotes the predefined iteration number. The relative sampling locations $\mathcal{S}$ form a sampling path.

At each step, we get a new sampling location $\mathbf{p}+\mathbf{p}_t$ from current accumulated offset $\mathbf{p}_t$. The sampled feature $\mathrm{x}(\mathbf{p}+\mathbf{p}_t)$ represents a feature-level component of text observed at current step, and the whole text instance is gradually grouped naturally through the step-by-step samplings. It's notable that, from any start location within the text instance, we should "sweep" along two opposite directions to capture the whole text instance, wherefore adopting two separate sampling paths $\mathcal{S}_d = \{\mathbf{p}_{d,t} | \ t = 1, ..., T\}$ $(d = 1, 2)$ are more suitable. For two directions $d = 1, 2$, Equ. 2 becomes:

$$\mathbf{p}_{d,t+1} = \mathbf{p}_{d,t} + \triangle \mathbf{p}_{d,t}, \ t = 0, ..., T-1. \tag{3}$$

In this work, the sequential sampling network is realized through a recurrent neural network (RNN) followed by a linear layer, and two separate sampling

paths are generated by two separate sequential sampling networks, so $\triangle \mathbf{p}_{d,t}$ is conditioned on previous sampled features $\{\mathrm{x}(\mathbf{p}+\mathbf{p}_{d,0}), ..., \mathrm{x}(\mathbf{p}+\mathbf{p}_{d,t})\}$:

$$\mathrm{h}_{d,t} = RNN_d(\mathrm{x}(\mathbf{p}+\mathbf{p}_{d,t}),\ \mathrm{h}_{d,t-1}) \tag{4}$$

$$\triangle \mathbf{p}_{d,t} = Linear_d(\mathrm{h}_{d,t}). \tag{5}$$

The RNN stores the historical shape information, and stabilizes the training process and moderately boosts the performance. Based on previous observations, the network will adaptively calibrate magnitude and orientation of $\triangle \mathbf{p}_t$ to ensure the largest possible covering of whole text instance. Sequential sampling network for one direction d is shared across all start locations.

After sequential sampling, the SDM calculates the weighted summation of feature at the start location and all sampled features to generate the output feature map y:

$$\mathrm{y}(\mathbf{p}) = \mathrm{w}_0 \cdot \mathrm{x}(\mathbf{p}) + \sum_{d=1}^{2} \sum_{t=1}^{T} \mathrm{w}(d,t) \cdot \mathrm{x}(\mathbf{p}+\mathbf{p}_{d,t}). \tag{6}$$

In practical implementation, Equ. 6 is equivalently implemented as following:

$$\mathrm{m}(\mathbf{p}) = Concat(\{\mathrm{x}(\mathbf{p}+\mathbf{p}_{d,t}) \mid d=1,2,\ t=1,...,T\} \cup \{\mathrm{x}(\mathbf{p})\}), \tag{7}$$

$$\mathrm{y}(\mathbf{p}) = Conv_{1\times1}(\mathrm{m}(\mathbf{p})), \tag{8}$$

where the intermediate feature map m is the concatenation of all sequentially sampled feature maps and the input feature. C is the channel of input feature map and the intermediate feature map m has a channel of $(2T+1)\cdot C$, corresponding to $2T$ times sampling and the original input. Bilinear interpolation is used to compute $\mathrm{x}(\mathbf{p}+\mathbf{p}_{d,t})$ owing to fractional sampling locations. As bilinear interpolation is differentiable, the gradients can be back-propagated to the sampling locations as well as the predicted offsets, and the training of SDM is conducted via a weakly-supervised end-to-end optimization.

3.2 Auxiliary Character Counting Supervision

The SDM paves a way to adaptively capture the whole text instance, but its sequential sampling is inevitably undermined by error accumulation in offsets. Besides, without any explicit supervision, the model's training stability is unsatisfactory.

In many common datasets, the text transcription labels and naturally the character number labels are provided. Therefore, we further introduce an auxiliary character counting supervision to guide the SDM's precise sequential sampling. This simultaneously enables the model to learn character-level semantic information. Instead of text recognition, we adopt the language-agnostic character counting as the extra supervision, because text recognition task has a large search space and there is a large gap between the convergence rates of scene text detection and recognition. On real datasets where samples for recognition is

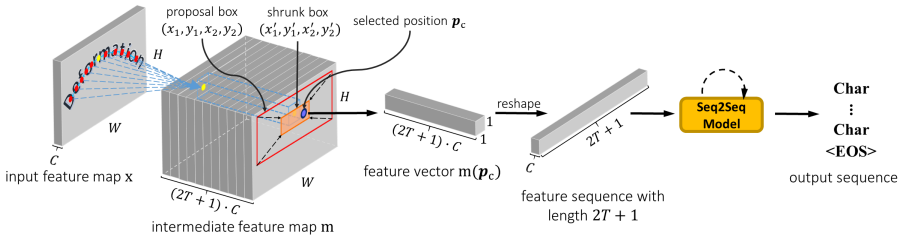


Fig. 3. Sequence-to-sequence based character counting.

very insufficient, jointly training recognition and detection is hard to maximize the performance. By reducing the text recognition to the character counting, the search space is greatly reduced and the optimization is much easier. As a result, we can use character counting to boost our detector on real datasets without additional large synthetic dataset or pre-trained recognition model.

The character counting task is modeled as a sequence-to-sequence (seq2seq) problem [33]. In essence, provided a selected start location, the feature at start location and all the sequentially sampled features could form a input feature sequence, and a seq2seq-based model predicts a valid sequence, whose length is expected to equal the character number of corresponding text instance. The detailed counting process is depicted in Fig. 3.

In SDM, an intermediate feature map m that is the concatenation of all sampled feature maps and the input feature map is obtained first, as described in Sec. 3.1. The map m contains instance-level modeling information for different text instances, and it is capitalized by the consequent character counting.

Then, we select some training samples to optimize the seq2seq-based counting network. In this work, we integrate our SDM into the anchor-based Mask R-CNN [7], where SDM is inserted before the region proposal network (RPN). The effective scheme is adopting the feature vectors around the center area of positive proposal boxes from RPN as training samples. More specifically, we first randomly select K proposals from all positive proposals from RPN. Next, for a selected positive proposal box (x_1, y_1, x_2, y_2), we employ centered shrinkage upon it with shrunk ratio σ. The shrunk box (x_1', y_1', x_2', y_2') (orange area in Fig. 3) lies in the center field of the proposal box, so the features within the shrunk box have higher probability to identify the global existence of text instance. We designate the shrunk box as the distribution region of training sample. We randomly select a position $\mathbf{p}_c = (x_c, y_c)$ from the shrunk box:

$$x_c \sim U(x_1', x_2'),\ \ y_c \sim U(y_1', y_2'), \tag{9}$$

where U denotes the uniform distribution. Given the select position $\mathbf{p}_c$, we get the feature vector $\mathrm{m}(\mathbf{p}_c)$ with channel $(2T+1)\cdot C$ and reshape it to a sequence with length $2T+1$ and channel C, which composes one training sample for character counting. Bilinear interpolation is also used to compute $\mathrm{m}(\mathbf{p}_c)$.

Finally, a one-layer transformer [36] is utilized as our seq2seq model to predict the number of character, and it makes classification for four symbols at each time step, including start-of-sequence symbol "<SOS>", end-of-sequence symbol "<EOS>", padding symbol "<PAD>" and a "Char" symbol. The "Char" symbol represents the existence of one character. Receiving a "<SOS>" symbol and the feature sequence reshaped from feature vector $\mathrm{m}(\mathbf{p}_c)$, the model is expected to maximizing the log probability of the target sequence s. The target sequence s contains consecutive "Char" symbols, whose amount equals the character number of corresponding text instance, and ends with a "<EOS>" symbol. Hence the counting loss is:

$$\mathcal{L}_{cnt} = -\log p(s \mid reshape(\mathrm{m}(\mathbf{p}_c))). \tag{10}$$

The auxiliary counting task can be simply extended to the scene text recognition task by forcing the network to discriminate different characters at each step. The comparison between character counting and recognition is described in Sec. 4.3.

3.3 Mask R-CNN with SDM

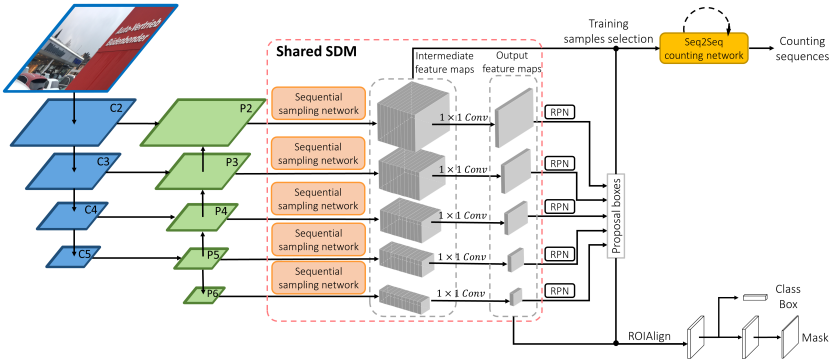


Fig. 4. The architecture of Mask R-CNN equipped with the proposed SDM. C_n $(n = 2, ..., 5)$ and P_n $(n = 2, ..., 6)$ respectively denote the feature maps (stride 2^n) from the backbone network and feature pyramid network (FPN). SDM is inserted before the region proposal network (RPN) and shared between different levels. The seq2seq counting network is only used in the training phase and is also shared between different levels.

In this work, modeling scene text detection as an instance segmentation task, we leverage the powerful Mask R-CNN [7] with feature pyramid network (FPN) [17] as our baseline detector and equip it with the proposed SDM, as shown in Fig. 4. We re-implement the Mask R-CNN tailored for scene text detection in [18]. The main modifications to standard Mask R-CNN in [18]

include: (1) Flipping, resizing and cropping training augmentations; (2) Fine-tuned RPN anchor aspect ratios {0.17, 0.44, 1.13, 2.90, 7.46}; (3) convolution layers with dilation=2 and bilinear upsampling layer in the mask branch; (4) Online hard example mining (OHEM) [32] in bounding box branch. Moreover, we carry out additional color and geometrical augmentations to further enhance the generalization ability. Color augmentations include hue, saturation, brightness and contrast [49] and geometrical augmentation is random rotation in range $[-10°, +10°]$. All these augmentations are performed with a probability of 0.5 independently. Our implemented baseline detector (ResNet-50) achieves a F1-score of 77.07% on MLT2017 dataset.

The SDMs are inserted before RPNs for different feature levels, and, following the practice in FPN, the SDMs are shared between different levels. With respect to the auxiliary character counting, a proposal box at level i generates a counting training sample from the corresponding i-th intermediate feature map m_i, and the seq2seq character counting network is also shared across different feature levels. Meanwhile, the RoIAlign layer extracts region features from the output feature map of SDM (i.e. y in Equ. 6). The network is trained in an end-to-end manner using the following objective:

$$\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{box} + \mathcal{L}_{mask} + \gamma \mathcal{L}_{cnt}, \tag{11}$$

where $\mathcal{L}_{cls}$, $\mathcal{L}_{box}$ and $\mathcal{L}_{mask}$ respectively represent the classification, bounding box regression and mask loss in Mask R-CNN, and $\mathcal{L}_{cnt}$ denotes the character counting loss described in Sec. 3.2. The loss weight γ is set to 1.0 empirically.

4 Experiments

We evaluate our method on ICDAR 2017 MLT, ICDAR 2015, Total-Text and CTW1500. Extensive experiments demonstrate that, integrated into the powerful Mask R-CNN framework, our proposed SDM obtains consistent and remarkable performance boost and outperforms state-of-the-art methods.

4.1 Datasets

ICDAR 2017 MLT [30] is a multi-oriented, multi-scripting, and multi-lingual scene text dataset. It consists of 7200 training images, 1800 validation images, and 9000 test images, respectively. The text regions are annotated as quadrangles in word-level or line-level for different languages.

ICDAR 2015 [13] is an incidental multi-oriented text detection dataset for English. It consists of 1000 training images, 500 validation images, and 500 test images, respectively. The text regions are labeled as word-level quadrangles.

Total-Text [2] is a dataset not only contains horizontal and multi-oriented text but also specially features curved-oriented text for English. The dataset is split into training and testing sets with 1255 and 300 images, respectively, and all the text regions are labeled as a polygon in word-level.

CTW1500 [21] is a dataset mainly consisting of curved text with both English and Chinese instances. Each image has at least one curved text when horizontal and multi-oriented texts are also contained in this dataset. The dataset contains 1000 training images and 500 test images. Each text is labeled as a polygon in line-level with 14 vertexes.

4.2 Implementation Details

The main configurations for the re-implementation of Mask R-CNN baseline are described in Sec. 3.3. In the SDM, the iteration number T is set to 5, and the hidden size of RNN in the sequential sampling network is 64. For auxiliary character counting, the proposal box's shrunk ratio σ is empirically set to 0.1 and 0.3 for ResNet-18 and ResNet-50, respectively. The one-layer transformer has one attention head and a model dimension of 256. An additional polygon NMS with threshold 0.2 is applied to suppress redundant polygons. We adopt the SGD optimizer with batch size 32, momentum 0.9 and weight decay 1×10^{-4}. During training stage, on all datasets except ICDAR 2015, image's two sides are resized independently in ranges of $[640, 2560]$, $[640, 1600]$ and $[512, 1024]$, respectively. And for ICDAR 2015, image's long side is resized in range $[640, 2560]$, preserving its aspect ration, and then the height is rescaled from 0.8 to 1.2 while the width keeps unchanged. Horizontal flipping with a probability of 0.5 is applied, and a 640×640 patch is cropped for training. For single scale testing, image's long side is resized to 1600, 1920, 1024 and 768 on four datasets, respectively. For multi-scale testing, the long side is resized to {960, 1600, 2560}, {1280, 1920, 2560}, {640, 1024, 1600} and {512, 768, 1024} on four datasets, respectively.

4.3 Ablation Study

To demonstrate the effectiveness of our approach, extensive ablation studies are conducted on ICDAR 2017 MLT dataset considering its high variety in text and multi-lingual challenge. We evaluate two essential components in our model: Sequential Deformation Module (SDM) and Auxiliary Character Counting (ACC). Results are shown in Table 1.

Baseline The baseline model is built on Mask R-CNN, which is described in Sec. 3.3. It achieves an F-measure of 77.07%.

Sequential Deformation Module The experimental results show that the Sequential Deformation Module brings a gain of 0.68% and 0.55% for ResNet-18 and ResNet-50 backbones. This shows the effectiveness of SDM to handle multi-oriented and multi-lingual text.

Auxiliary Character Counting To verify the effectiveness of auxiliary character counting, we introduce character counting and recognition upon SDM. For recognition, we extended the character counting to recognition by simply replacing the "Char" symbol with symbols of full characters. The counting supervision achieves an improvement of 0.5% on F-measure, while recognition brings no gains. Besides, counting network converges as the training of detector, but the

Table 1. Effectiveness of Sequential Deformation Module (SDM) and Auxiliary Character Counting (ACC) on ICDAR 2017 MLT dataset."P", "R", and "F" refer to precision, recall and F-measure, respectively.

Method	Backbone	P(%)	R(%)	F(%)
Baseline	ResNet-18	80.36	70.04	74.84
Baseline + SDM (w/o ACC)	ResNet-18	81.80	70.31	75.62
Baseline + SDM (w/ ACC)	ResNet-18	82.14	70.72	76.00
Baseline	ResNet-50	82.10	72.62	77.07
Baseline + SDM (w/o ACC)	ResNet-50	83.34	72.64	77.62
Baseline + SDM (w/ ACC)	ResNet-50	84.16	72.82	78.08
Baseline + SDM (w/ recognition)	ResNet-50	82.98	72.95	77.64

recognition network hardly converges. Fig. 5 indicates the guidance of auxiliary character counting for sequential sampling.

Fig. 5. Top: Sequential sampling without and with auxiliary character counting. Bottom: Ablations for iteration number T in sequential sampling on MLT 2017.

Iteration Number for Sequential Sampling As mentioned in Sec. 3.1, the sequential sampling's iteration number T is pre-set and is critical for expanding SDM's sampling range and associated receptive field. We adopt the ResNet-18 backbone and Fig. 5 shows the performances as T changes. The F-measure firstly increases and then saturates for $T \geq 5$. Thus, we use 5 in the remaining experiments. Meanwhile, irrespective of T, all the sampling paths are able to adaptively cover the text regions and avoid going out of the instance. Surprisingly, even when $T = 0$, i.e. using just one feature vector rather sequence to predict character number, a promising improvement of 0.4% is observed, suggesting that the auxiliary character counting is essentially beneficial for detection.

4.4 Comparative Results on Public Benchmarks

Detecting Multi-lingual Text On ICDAR 2017 MLT, we train the network on 9000 training and validation images for 140 epochs with the weight pre-trained on

ImageNet [4]. The learning rate is initialized as 4×10^{-2} and reduced by a factor of 10 at epoch 80 and 125. For single-scale testing, our models with ResNet-18 and ResNet-50 achieve F-measures of 76.00% and 78.08%. For multi-scale testing, the model with ResNet-50 achieves 80.61% F-measure, outperforming all the state-of-the-art methods. Even though a weak backbone (ResNet-18) is adopted, our model is also very competitive compared with the best PMTD [18] (79.13% vs 80.13%). The results are listed in Table 2. Some qualitative results are shown in Fig. 6(a), showcasing the SDM's great robustness for multi-lingual text, long text and complicated background.

Table 2. Comparative Results on ICDAR 2017 MLT and ICDAR 2015 datasets. * denotes the results based on multi-scale testing. "P", "R", and "F" refer to precision, recall and F-measure, respectively.

Datasets	**ICDAR 2017 MLT**			**ICDAR 2015**		
Method	**P(%)**	**R(%)**	**F(%)**	**P(%)**	**R(%)**	**F(%)**
EAST [50]	-	-	-	83.27	78.33	80.72
TextSnake [26]	-	-	-	84.90	80.40	82.60
RRD* [16]	-	-	-	88.00	80.00	83.80
Lyu et al.* [28]	74.30	70.60	72.40	89.50	79.70	84.30
LOMO* [48]	79.10	60.20	68.40	87.80	87.60	87.70
PSENet [39]	77.01	68.40	72.45	89.30	85.22	87.21
SPCNET* [45]	80.60	68.60	74.10	-	-	-
FOTS* [20]	81.86	62.30	70.75	91.85	87.92	89.84
PMTD [18]	85.15	72.77	78.48	91.30	87.43	89.33
PMTD* [18]	84.42	**76.25**	80.13	-	-	-
Ours (ResNet-18)	82.14	70.72	76.00	91.14	84.69	87.80
Ours* (ResNet-18)	85.44	73.68	79.13	90.15	88.16	89.14
Ours (ResNet-50)	84.16	72.82	78.08	88.70	88.44	88.57
Ours* (ResNet-50)	**86.79**	75.26	**80.61**	**91.96**	**89.22**	**90.57**

Detecting Oriented English Text On ICDAR2015, the weights trained on ICDAR 2017 MLT are used to initialize the models. We fine-tune the network for 80 epochs with learning rate 4×10^{-3} in the first 40 epoch and 4×10^{-4} in the remaining 40 epoch. As shown in Table 2, our model with ResNet-50 even surpasses FOTS [20], which is trained with both detection and recognition supervision. The visualization in Fig. 6(b) shows our SDM can effectively tackle challenging situations including skewed viewpoint and low resolution. Notably, our model could accurately locate different text instances under the crowded scene.

Detecting Curved Text We evaluate our method on the Total-Text and CTW1500 to validate SDM's ability to detect curved text. For Total-Text, the network is also initialized with weights pre-trained on ICDAR 2017 MLT. Considering there's no text transcription in CTW1500, we initialize the network

Table 3. Comparative Results on Total-Text and CTW1500 datasets. * denotes the results based on multi-scale testing. "P", "R", and "F" refer to precision, recall and F-measure, respectively.

Datasets	Total-Text			CTW1500		
Method	P(%)	R(%)	F(%)	P(%)	R(%)	F(%)
CTD + TLOC [21]	74.30	69.80	73.40	-	-	-
TextSnake [26]	82.70	74.50	78.40	85.30	67.90	75.60
Mask TextSpotter [27]	69.00	55.00	61.30	-	-	-
PSENet [39]	84.02	77.96	80.87	84.84	79.73	82.20
CRAFT [1]	87.60	79.90	83.60	86.00	81.10	83.50
DB-ResNet-50 [15]	87.10	82.50	84.70	86.90	80.20	83.4
PAN [40]	89.30	81.00	85.00	86.40	81.20	83.70
PAN Mask R-CNN [11]	-	-	-	86.80	83.20	85.00
CharNet* [46]	88.00	85.00	86.50	-	-	-
Baseline (ResNet-50)	87.44	84.93	86.16	84.16	81.99	83.06
Ours (ResNet-50)	89.24	84.70	86.91	85.82	82.27	84.01
Ours* (ResNet-50)	**90.85**	**86.03**	**88.37**	**88.40**	**84.42**	**86.36**

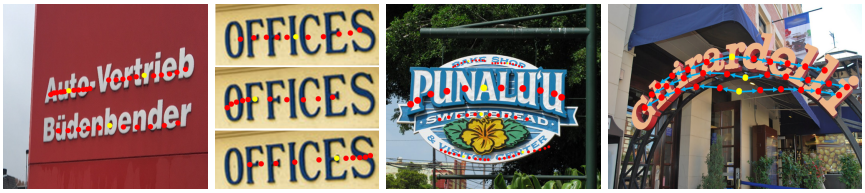
with weights trained on Total-Text and disable the character counting task. All models are fine-tuned for 140 epochs with learning rate 4×10^{-3} in the first 80 epochs and 4×10^{-4} in the remaining epochs.

As visualized in Fig. 7, our SDM is capable of capturing various shapes, which leverages the model's weakness on geometric transformation. The quantitative results for the curved datasets are shown in Table 3. Our model respectively brings an absolute improvement of 0.75% and 0.95% on Total-Text and CTW1500 Datasets, and also surpasses all previous methods. Especially, our model outperforms the state-of-the-art on Total-Text by a large margin of 1.87%. This verifies the SDM's generalization ability on arbitrary-shaped scene text detection.

(a) MLT2017 (b) ICDAR 2015 (c) Total-Text (d) CTW1500

Fig. 6. Qualitative results of the proposed method on four public datasets.

4.5 Discussion for SDM's Adaptability



(a) (b) (c) (d)

Fig. 7. Examples of sequential sampling locations (red points) for different start locations (yellow points). Sampling locations for (a) different instances, (b) different start locations within the same instance, (c) different curved text instances at multi-level feature maps (larger point indicates higher feature map) and (d) start locations inside and outside the text region are visualized. In (d), the sequential relations are also visualized through blue arrows to distinguish different sampling paths.

Owing to the iteration number T in SDM is pre-set, we expect the SDM has sufficient adaptability for geometrical variance and start location. From the visualization in Fig. 7, it shows that: 1) The sampling range will automatically expand and shrink to fit for different instances and different start locations within the same instance, and avoid going out of the instance; 2) For the more challenging curved texts, the SDM still performs competently to capture the curved shape; 3) For a start location outside the text, the SDM will try to follow the text center line. These imply that the SDM has high adaptability and dynamically calibrates itself to capture the text instance as far as possible, enriching CNN's capability to model geometrical transformations.

5 Conclusion

In this paper, we propose a novel end-to-end sequential deformation module for accurate scene text detection, which adaptively enhances the modeling capability for text's geometric configuration without any post-processing. We also introduce an auxiliary character counting supervision to facilitate the sequential sampling and features learning. The effectiveness of our method has been demonstrated on several public benchmarks for multi-lingual, multi-oriented and curved text.

Acknowledgement

The authors would like to thank the reviewers for their valuable comments to improve the quality of the paper. This research is supported by a joint research project between Hyundai Motor Group AIRS Company and Tsinghua University. The second author is partially supported by National Key R&D Program of China and a grant from the Institute for Guo Qiang, Tsinghua University.

References

1. Baek, Y., Lee, B., Han, D., Yun, S., Lee, H.: Character Region Awareness for Text Detection. In: Proc. CVPR. pp. 9365–9374 (2019)
2. Ch'ng, C.K., Chan, C.S.: Total-text: A comprehensive dataset for scene text detection and recognition. In: Proc. ICDAR. pp. 935–942 (2017)
3. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: Proc. ICCV. pp. 764–773 (2017)
4. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Li, F.F.: Imagenet: A large-scale hierarchical image database. In: Proc. CVPR. pp. 248–255 (2009)
5. Feng, W., He, W., Yin, F., Zhang, X.Y., Liu, C.L.: TextDragon: An end-to-end framework for Arbitrary shaped text spotting. In: Proc. ICCV. pp. 9076–9085 (2019)
6. Girshick, R.: FastR-CNN. In: Proc. CVPR. pp. 1440–1448 (2015)
7. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: Proc. ICCV. pp. 2961–2969 (2017)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc. CVPR. pp. 770–778 (2016)
9. He, W., Zhang, X.Y., Yin, F., Liu, C.L.: Deep direct regression for multi-oriented scene text detection. In: Proc. ICCV. pp. 745–753 (2017)
10. Huang, L., Yang, Y., Deng, Y., Yu, Y.: Densebox: Unifying landmark localization with end to end object detection. arXiv preprint arXiv:1509.04874 (2015)
11. Huang, Z., Zhong, Z., Sun, L., Huo, Q.: Mask R-CNN with pyramid attention network for scene text detection. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 764–772 (2019)
12. Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. In: Proc. NIPS. pp. 2017–2025 (2015)
13. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., Matas, J., Neumann, L., Chandrasekhar, V.R., Lu, S., et al.: Icdar 2015 competition on robust reading. In: Proc. ICDAR. pp. 1156–1160 (2015)
14. Liao, M., Shi, B., Bai, X., Wang, X., Liu, W.: Textboxes: A fast text detector with a single deep neural network. In: Proc. AAAI. pp. 4161–4167 (2017)
15. Liao, M., Wan, Z., Yao, C., Chen, K., Bai, X.: Real-time Scene Text Detection with Differentiable Binarization. arXiv preprint arXiv:1911.08947 (2019)
16. Liao, M., Zhu, Z., Shi, B., Xia, G.s., Bai, X.: Rotation-sensitive regression for oriented scene text detection. In: Proc. CVPR. pp. 5909–5918 (2018)
17. Lin, T.Y., Dollar, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature Pyramid Networks for Object Detection. In: Proc. CVPR. pp. 2117–2125 (2017)
18. Liu, J., Liu, X., Sheng, J., Liang, D., Li, X., Liu, Q.: Pyramid Mask Text Detector. arXiv preprint arXiv:1903.11800 (2019)
19. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: SSD: Single shot multibox detector. In: Proc. ECCV. pp. 21–37 (2016)
20. Liu, X., Liang, D., Yan, S., Chen, D., Qiao, Y., Yan, J.: FOTS: Fast Oriented Text Spotting With a Unified Network. In: Proc. CVPR. pp. 5676–5685 (2018)
21. Liu, Y., Jin, L., Zhang, S., Zhang, S.: Detecting curve text in the wild: New dataset and new solution. arXiv preprint arXiv:1712.02170 (2017)
22. Liu, Y., Zhang, S., Jin, L., Xie, L., Wu, Y., Wang, Z.: Omnidirectional Scene Text Detection with Sequential-free Box Discretization. arXiv preprint arXiv:1906.02371 (2019)

23. Liu, Z., Lin, G., Yang, S., Liu, F., Lin, W., Goh, W.L.: Towards robust curve text detection with conditional spatial expansion. In: Proc. CVPR. pp. 7269–7278 (2019)
24. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proc. CVPR. pp. 3431–3440 (2015)
25. Long, S., He, X., Yao, C.: Scene text detection and recognition: The deep learning era. arXiv preprint arXiv:1811.04256 (2018)
26. Long, S., Ruan, J., Zhang, W., He, X., Wu, W., Yao, C.: TextSnake: A Flexible Representation for Detecting Text of Arbitrary Shapes. In: Proc. ECCV (2018)
27. Lyu, P., Liao, M., Yao, C., Wu, W., Bai, X.: Mask TextSpotter: An End-to-End Trainable Neural Network for Spotting Text with Arbitrary Shapes. In: Proc. ECCV. pp. 67–83 (2018)
28. Lyu, P., Yao, C., Wu, W., Yan, S., Bai, X.: Multi-oriented scene text detection via corner localization and region segmentation. In: Proc. CVPR. pp. 7553–7563 (2018)
29. Ma, J., Shao, W., Ye, H., Wang, L., Wang, H., Zheng, Y., Xue, X.: Arbitrary-oriented scene text detection via rotation proposals. IEEE Transactions on Multimedia pp. 3111–3122 (2018)
30. Nayef, N., Yin, F., Bizid, I., Choi, H., Feng, Y., Karatzas, D., Luo, Z., Pal, U., Rigaud, C., Chazalon, J., et al.: ICDAR2017 robust reading challenge on multi-lingual scene text detection and script identification-RRC-MLT. In: Proc. ICDAR. pp. 1454–1459 (2017)
31. Shi, B., Bai, X., Belongie, S.: Detecting Oriented Text in Natural Images by Linking Segments. In: Proc. CVPR. pp. 2550–2558 (2017)
32. Shrivastava, A., Gupta, A., Girshick, R.: Training region-based object detectors with online hard example mining. In: Proc. CVPR. pp. 761–769 (2016)
33. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Proc. NIPS. pp. 3104–3112 (2014)
34. Tian, Z., Huang, W., He, T., He, P., Qiao, Y.: Detecting text in natural image with connectionist text proposal network. In: Proc. ECCV. pp. 56–72 (2016)
35. Tian, Z., Shu, M., Lyu, P., Li, R., Zhou, C., Shen, X., Jia, J.: Learning Shape-Aware Embedding for Scene Text Detection. In: Proc. CVPR. pp. 4234–4243 (2019)
36. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Proc. NIPS. pp. 5998–6008 (2017)
37. Wang, F., Zhao, L., Li, X., Wang, X., Tao, D.: Geometry-aware Scene Text Detection With Instance Transformation Network. In: Proc. CVPR. pp. 1381–1389 (2018)
38. Wang, F., Zhao, L., Li, X., Wang, X., Tao, D.: Geometry-aware scene text detection with instance transformation network. In: Proc. CVPR. pp. 1381–1389 (2018)
39. Wang, W., Xie, E., Li, X., Hou, W., Lu, T., Yu, G., Shao, S.: Shape Robust Text Detection With Progressive Scale Expansion Network. In: Proc. CVPR. pp. 9336–9345 (2019)
40. Wang, W., Xie, E., Song, X., Zang, Y., Wang, W., Lu, T., Yu, G., Shen, C.: Efficient and Accurate Arbitrary-Shaped Text Detection with Pixel Aggregation Network. In: Proc. CVPR. pp. 8440–8449 (2019)
41. Wang, X., Jiang, Y., Luo, Z., Liu, C.L., Choi, H., Kim, S.: Arbitrary shape scene text detection with adaptive text region representation. In: Proc. CVPR. pp. 6449–6458 (2019)
42. Wigington, C., Tensmeyer, C., Davis, B., Barrett, W., Price, B., Cohen, S.: Start, follow, read: End-to-end full-page handwriting recognition. In: Proc. ECCV. pp. 367–383 (2018)

43. Wikipedia: Eye movement in reading. https://en.wikipedia.org/wiki/Eye_movement_in_reading
44. Wu, W., Xing, J., Zhou, H.: TextCohesion: Detecting Text for Arbitrary Shapes. arXiv preprint arXiv:1904.12640 (2019)
45. Xie, E., Zang, Y., Shao, S., Yu, G., Yao, C., Li, G.: Scene text detection with supervised pyramid context network. In: Proc. AAAI. pp. 9038–9045 (2019)
46. Xing, L., Tian, Z., Huang, W., Scott, M.R.: Convolutional Character Networks. In: Proc. ICCV. pp. 9126–9136 (2019)
47. Xue, C., Lu, S., Zhan, F.: Accurate Scene Text Detection through Border Semantics Awareness and Bootstrapping. In: Proc. ECCV. pp. 355–372 (2018)
48. Zhang, C., Liang, B., Huang, Z., En, M., Han, J., Ding, E., Ding, X.: Look More Than Once: An Accurate Detector for Text of Arbitrary Shapes. In: Proc. CVPR. pp. 10552–10561 (2019)
49. Zhang, Z., He, T., Zhang, H., Zhang, Z., Xie, J., Li, M.: Bag of freebies for training object detection neural networks. arXiv preprint arXiv:1902.04103 (2019)
50. Zhou, X., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W., Liang, J.: EAST: an efficient and accurate scene text detector. In: Proc. CVPR. pp. 5551–5560 (2017)
51. Zhu, X., Hu, H., Lin, S., Dai, J.: Deformable convnets v2: More deformable, better results. In: Proc. CVPR. pp. 9308–9316 (2019)