Explore Faster Localization Learning For Scene Text Detection

Yuzhong Zhao
School of Computer Science and Technology
University of Chinese Academy of Sciences
Beijing, China
zhaozuyu20@mails.ucas.ac.cn

Weijia Wu
Key Laboratory for Biomedical Engineering of Ministry
Zhejiang University
Hangzhou, China
weijiawu@zju.edu.cn

Yuanqiang Cai
State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications
Beijing, China
caiyuanqiang@bupt.edu.cn

Weiqiang Wang*
School of Computer Science and Technology
University of Chinese Academy of Sciences
Beijing, China
wqwang@ucas.ac.cn

Abstract—Generally, pre-training and long-time training computation are necessary for obtaining a good-performance text detector based on deep networks. In this paper, we present a new scene text detection network (called FANet) with a Fast convergence speed and Accurate text localization. The proposed FANet is an end-to-end text detector based on transformer feature learning and normalized Fourier descriptor modeling, where the Fourier Descriptor Proposal Network and Iterative Text Decoding Network are designed to efficiently and accurately identify text proposals. Additionally, a Dense Matching Strategy and a well-designed loss function are also proposed for optimizing the network performance. Extensive experiments are carried out to demonstrate that the proposed FANet can achieve the SOTA performance with fewer training epochs and no pre-training. When we introduce additional data for pre-training, the proposed FANet can achieve SOTA performance on MSRA-TD500, CTW1500, and TotalText. The ablation experiments also verify the effectiveness of our contributions. Code is available at https://github.com/callsys/FANet.

Index Terms—Scene text detection, Normalized Fourier descriptor, Dense matching strategy.

I. INTRODUCTION

Scene text detection is an important task of computer vision, and a basis of various text-related applications, so many researchers are concerned about the issue [1]–[6]. Most of the state-of-the-art methods [7]–[11] rely on long-time training to achieve a good performance. Generally researchers use related large datasets for long-time pre-training, and then finetune the network on the target dataset, or directly carry out long-time training on the target dataset. These approaches are not suitable for scenarios that require to rapidly generate models or no large dataset for pre-training.

Inspired by the feature learning ability of the transformer [12] and deformable DETR [13] and the representation ability of Fourier descriptor for arbitrary contours [9], we present a scene text detection network (called FANet) with a Fast convergence speed and Accurate text localization, where the transformer is combined with Fourier descriptor to localize arbitrary-shaped text regions. The DETR based methods [14] can scale-invariantly localize the rectangular target regions. The Fourier descriptor proposed by [9] can model target regions of arbitrary contours, but it cannot be embedded into the DETR detection framework. Thus, we propose a normalization method for Fourier descriptor to enable the DETR based methods to predict the normalized text regions of arbitrary shape. However, we find that the established framework based on the proposed normalization method has slow convergence and low accuracy due to two aspects of reasons, i.e., (1) the change of regression target makes the deformable DETR component become sub-optimal and (2) the Hungarian Matching Strategy hinders the rapid convergence of the network. Correspondingly, we first make some effective
changes to the structure of deformable DETR. Concretely, we propose a Fourier Descriptor Proposal Network (FDPN) to get better candidates for the text decoder. Then, we build an Iterative Text Decoding Network (ITDN) to iteratively refine Fourier proposals. Finally, we propose a well-designed loss function to optimize the descriptor representations and calculate the matching cost. Additionally, we propose a Dense Matching Strategy (DMS) to greatly speed up the convergence and improve the detection accuracy within fewer training epochs. As shown in Fig. 1, the proposed FANet can obtain an F-measure of 83.3% and 84.8% respectively, after training only 100 epochs on datasets CTW1500 [15] and MSRA-TD500 [16] without pre-training, which outperforms current best SOTA methods [7] by 6.5% (83.3 vs. 76.8%) and 16.5% (84.8 vs. 68.3%) respectively. The main contributions of this paper are summarized as follows.

- We present a scene text detection network FANet with fast convergence speed and accurate localization, which uses the transformer to learn text features, and the normalized Fourier descriptor to represent text regions. The proposed FANet has achieved SOTA performances on multiple public benchmarks, e.g., MSRA-TD500, CTW1500 and TotalText.

- We make many changes to the original deformable DETR, including Fourier Descriptor Proposal Network (FDPN), Iterative Text Decoding Network (ITDN), well-designed loss function and Dense Matching Strategy (DMS). These components can effectively improve the convergence speed and accuracy of FANet.

II. RELATED WORKS

A. Transformer modeling

The transformer [12] with both self-attention and cross-attention mechanism has achieved great success in both machine translation and visual recognition. For example, DETR [14] first adopts the transformer architecture for the object detection task. deformable DETR [13] extends DETR with a deformable attention module that reduces the training time significantly. Some previous works also try to explore the potential of transformer on text spotting task. Wu et al. [17], [18] proposed to track and spot text in video with transformer sequence modeling. [8] adopts the transformer architecture in multi-orientation text detection for the first time, but it still suffers from the problems of requiring massive data for pre-training, slow convergence, poor performance and inadequate representation ability. By using better contour representation, feature extraction and network optimization methods, we make the proposed FANet based on transformer surpasses the SOTA text detection algorithm based on CNNs [7], [9], [19].

B. Text Region Representation

Text regions can be modeled via per-pixel masks [7], [20], or modeled by parameters in specified representation spaces. For example, EAST [21] uses a rotate bounding box for text representation. TextRay [22] represents the text contours in the polar system. ABCNet [23] introduced Bezier curves to parameterize curved texts. FCENet [9] represents the text instances in the Fourier domain [24], which allows to represent any closed continuous contour in robust and simple manners. In this paper, we further present a new normalized Fourier descriptor to represent the normalized text regions of arbitrary shapes, which makes it possible to embed the text representation based on Fourier descriptor into the detection framework based on transformers.

III. METHOD DESCRIPTION

A. Overview

As shown in Fig. 2, the proposed FANet mainly consists of three parts: Feature Extraction Network (FEN), Fourier Descriptor Proposal Network (FDPN) and Iterative Text Decoding Network (ITDN). For a given image, it is first encoded as features by the FEN, which consists of a backbone and a transformer encoder. The encoded features are then fed into the FDPN to obtain a set of arbitrary-shaped text contours represented as normalized Fourier descriptor and the object queries transformed from them. Further, the encoded features, the object queries and the Fourier proposals with the highest scores are jointly fed into the ITDN to obtain the refined Fourier proposals. Finally, we get the detection results after applying Inverse Fourier transform (IFT), Inverse Normalization (IN) and Non-Maximum Suppression (NMS) to the refined Fourier proposals.

B. Fourier Descriptor Normalization

Target generation. FCENet uses a complex-value function 
\(z(t) = x(t) + iy(t), t \in [0, 1]\) to represent a text contour, where \((x(t), y(t))\) denotes the spatial coordinate of the text contour. To make the contour representation scale-invariant, we present a normalized form \(\tilde{Z}(t) = \frac{z(t)}{\|z(t)\|} = \lambda(t) + i\gamma(t)\), where \(H, W\) are the height and width of a given image. Since we cannot obtain the analytical form of the real scenario, we can discretize the function \(\tilde{Z}\) into \(N\) points (point sequence in Fig. 2), i.e., \(\{\tilde{Z}(\frac{n}{N})\}, n \in \{0, \cdots, N - 1\}\). Correspondingly, we can calculate the Fourier descriptor of a normalized text contour by Discrete Fourier Transform (DFT), i.e.,

\[
\tilde{c}_k = \frac{1}{N} \sum_{n=0}^{N-1} \tilde{Z}\left(\frac{n}{N}\right)e^{-2\pi ik \frac{n}{N}}, k \in \Psi, \quad (1)
\]

where \(\tilde{c}_k = \tilde{u}_k + i\tilde{v}_k \in \mathbb{C}, \tilde{u}_k\) and \(\tilde{v}_k\) are real part and imaginary part of \(\tilde{c}_k\), \(\Psi \triangleq \{-K, \cdots, 0, \cdots, K\}\), \(K\) is the highest frequency index reserved for the Fourier descriptor of the text contour (\(K=5\) in our experiments). Thus, each ground-truth text contour can be represented as a normalized Fourier descriptor, \(\hat{c} = [\hat{u}_{-K}, \hat{v}_{-K}, \cdots, \hat{u}_0, \hat{v}_0, \cdots, \hat{u}_K, \hat{v}_K] \in \mathbb{R}^{4K+2}\). We take normalized Fourier descriptor as regression targets. As shown in Fig. 3, the values of original Fourier descriptor are generally large and unbounded, while the values of normalized

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Fourier descriptor \( \hat{c} \) all fall into a small region with fixed boundaries, which can be proved to be:

\[
(\hat{u}_0, \hat{v}_0) \in [0, 1]^2,
\]

\[
(\hat{u}_k, \hat{v}_k) \in \left[-\frac{2}{\pi}, \frac{2}{\pi}\right]^2, k \in \psi/0. \tag{3}
\]

**New activation function.** Considering the above properties of normalized Fourier descriptor, the commonly used activation functions cannot meet our requirements. For example, the identity function is an unbounded function that does not fit the bounded regression target well. The sigmoid function \( \sigma \) cannot output negative values. The range of \( \tanh \) is much larger than our regression target, which results in too many invalid predictions. So we define a new activation function \( f(\hat{c}) \) suitable for the regression of proposed normalized Fourier descriptor as

\[
e_i = f(\hat{c}_i) = \begin{cases} 
\sigma(\hat{c}_i), & i = 2K, 2K+1, \\
\frac{2\tanh(\hat{c}_i)}{\pi}, & i \in \Omega/\{2K, 2K+1\},
\end{cases}
\tag{4}
\]

where \( \hat{c} \) is the stimulus, and \( c \in \mathbb{R}^{4K+2} \) is the prediction of regression branches, where \( \Omega = \{0, \cdots, 4K+1\} \) is the index set aligned with \( \Psi \). As shown in Fig. 3, we use the proposed activation function for the unbounded network output of the regression branch where we need to predict the normalized Fourier descriptor by the network, such as FDPN and ITDN.

**C. Fourier Descriptor Proposal Network**

We build a lightweight Fourier Descriptor Proposal Network (FDPN) following the FEN. As shown in Fig. 2, through a single convolution module, we predict two feature maps, i.e., \( F_{\text{score}} \in \mathbb{R}^{B \times 2 \times h_f \times w_f} \), \( F_{\text{reg}} \in \mathbb{R}^{B \times (4K+2) \times h_f \times w_f} \), where \( B \) is the batch size, \( h_f \) and \( w_f \) are the height and width of the feature map. Each pixel on the feature map is assigned as an object query, which directly predicts a normalized Fourier descriptor and a corresponding score, i.e., Fourier proposal \( c^0 \). The top \( N_q \) Fourier proposals with the highest scores are selected, and we convert them into the initial queries \( q^0 \) of the ITDN through a transformation layer, which consists of a linear layer and a position encoding layer as described in deformable DETR [13]. Finally, the proposals \( c^0 \) and the queries \( q^0 \) are sent to the ITDN.

**D. Iterative Text Decoding Network**

As shown in Fig. 2, Iterative Text Decoding Network (ITDN) is composed of \( D \) duplicate deformable transformer decoder layers. We let each module refine the Fourier descriptor based on the prediction from the previous module and we select the output of the last module as the final predictions. The refinement target of FANet is the normalized Fourier descriptor while the refinement target of deformable DETR [13] is normalized bounding boxes. To adapt to this change of regression target, we make some effective changes for the transformer decoder layer as follows:
(1) **Take the Fourier descriptor as the reference location for the Multi-Scale Deformable Attention Module.**

As shown in Fig. 2, in the calculation process of d-th transformer decoder layer, we first calculate the bounding box \((x, y, w, h) \in [0, 1]^4\) of the Fourier proposal \(c^{d-1}\) through a function \(b\) (boundary function \(b\) in Fig. 2). Then we use \(((px + x)w, (py + y)h)\) instead of \((px, py)\) as the sampling location for the Multi-Scale Deformable Attention (MSDeformAttn) module [13], where \((px, py)\) are the sampling offset predicted by the MSDeformAttn module. Such modifications make the sampled locations of the MSDeformAttn module related to the location of previously predicted text contour.

(2) **Use multiplication based refinement instead of the original addition based refinement for the non-do component of Fourier descriptor.**

As shown in Fig. 2, the refinement function maps a proposal \(c^{d-1}\) and an offset prediction \(o^d\) to a new proposal \(c^d\), which can be formulated as \(c^d_i = r(c^{d-1}_i, o^d_i), i \in \Omega\), where \(c^{d-1}\) and \(c^d\) denote the prediction of the regression branch of \((d-1)\)-th and \(d\)-th transformer decoder layer \(c_0\) denotes the proposal output by FDPN), \(o^d\) is the offset prediction of the \(d\)-th transformer decoder layer, which is obtained by inputting \(q^d\) into a linear layer. The addition based refinement function used by deformable DETR is as follows:

\[
e^d_i = r_1(e^{d-1}_i, o^d_i) = f(f^{-1}(e^{d-1}_i) + o^d_i), i \in \Omega, \quad (5)
\]

where \(f\) and \(f^{-1}\) are the activation function and its reverse function. We propose a new multiplication based refinement function for the non-do component of the normalized Fourier descriptor as

\[
e^d_i = r_2(e^{d-1}_i, o^d_i) = f(f^{-1}(e^{d-1}_i)e^{o^d_i}),
\]

\[
i \in \Omega/\{2K, 2K + 1\}. \quad (6)
\]

In the ITDN, the refinement function is defined as

\[
r(c^d_i) = \begin{cases} r_1(c^{d-1}_i, o^d_i), & i = 2K, 2K + 1, \\
r_2(c^{d-1}_i, o^d_i), & i \in \Omega/\{2K, 2K + 1\}. \end{cases} \quad (7)
\]

**E. Optimization strategies**

**Dense Matching Strategy.** The traditional Hungarian Matching Strategy (HMS) used in [13], [14] only matches one query for each ground-truth, which we find is one of the causes of slow convergence of the network. The number of text instances in an image is usually limited, and we denote it as \(N_g\). \(N_g\) is usually much less than the number of queries of the network \(N_q\) (300 in our experiments), i.e., \(N_g \ll N_q\). Using HMS means that only \(N_g\) queries are used as positive samples for the training of regression branch in each iteration, which makes the network easy to overfit and damages the performance of the network due to lack of positive samples. We propose to match multiple queries for each ground-truth text instance. Dense Matching Strategy (DMS) can effectively alleviate the problem of slow convergence, especially in the early stage of training. Since the DMS makes the FANet output overlapping detection results, we have to use the NMS as post-processing. Due to the small number of queries of DETR based methods and small number of \(N_m\), the number of predictions is generally small, so the NMS only brings a low cost.

To use the DMS algorithm, we need to define a Cost function that takes the set of predicted text contours \(\mathcal{A}\) and the set of ground-truth text contours \(\mathcal{A}\) as input and outputs the cost between any two elements in the two sets. We use \(\mathcal{C}_{cls} + \lambda \mathcal{L}_{reg}\) as the matching cost for any \((e, \hat{e})\) pair in the two sets, where \(\mathcal{C}_{cls} = -\log(s)\), where \(s\) is the predicted text confidence score corresponding to proposal \(e\), \(\lambda\) is the hyperparameter that balance the costs. We define \(\mathcal{L}_{reg}\) as follows:

\[
\mathcal{L}_{reg}(e, \hat{e}) = \mathcal{L}_{SD}(e, \hat{e}) + \alpha_1 \mathcal{L}_{FD}(e, \hat{e}) + \alpha_2 \mathcal{L}_{bbox}(e, \hat{e}), \quad (8)
\]

\[
\mathcal{L}_{SD}(e, \hat{e}) = L1[\mathcal{F}^{-1}(e), \mathcal{F}^{-1}(\hat{e})], \quad (9)
\]

\[
\mathcal{L}_{FD}(e, \hat{e}) = L1(e, \hat{e}), \quad (10)
\]

\[
\mathcal{L}_{bbox}(e, \hat{e}) = GIOU(b(e), b(\hat{e})), \quad (11)
\]

Where \(\alpha_1, \alpha_2\) are the hyperparameters that balance the three matching costs. \(L1\) is the L1 loss function, \(b\) is the boundary function, \(GIOU\) is the GIOU loss in [31], \(\mathcal{F}^{-1}\) is the Inverse Fourier Transform. For a given image, we first calculate the cost between \(\mathcal{A}\) and \(\hat{\mathcal{A}}\), then we iteratively perform Hungarian Matching (HM) for \(N_m\) times. Each time we first add the proposals that match any ground-truth text contour to the positive sample set \(\mathcal{P}\) and then remove these proposals from the matching queue by setting the matched rows of \(cost\) to \(+\infty\) to avoid repeated matching. Finally, we collect all the matched proposals as positive samples and the unmatched proposals as negative samples.

**Loss function.** The loss function of the proposed network is given by

\[
\mathcal{L} = \mathcal{L}_{cls,0} + \lambda \mathcal{L}_{reg,0} + \sum_{i=1}^{D} w_i (\mathcal{L}_{cls,i} + \lambda \mathcal{L}_{reg,i}), \quad (12)
\]

where \(\mathcal{L}_{cls,0}\) and \(\mathcal{L}_{reg,0}\) denote the classification and regression loss of the FDPN respectively, \(\mathcal{L}_{cls,i}\) and \(\mathcal{L}_{reg,i}\) denote the classification and regression loss of the \(i\)-th decoder layer, \(w_i\) is the hyperparameter that balance the losses of different transformer decoder layers, \(D\) is the number of decoder layers. We use focal loss [32] as our default classification loss for \(\mathcal{L}_{cls,i}\). We define \(\mathcal{L}_{reg,i}\) as follows:

\[
\mathcal{L}_{reg,i} = \frac{1}{|\mathcal{M}|} \sum_{(e, \hat{e}) \in \mathcal{M}_i} (\mathcal{L}_{SD} + \alpha_1 \mathcal{L}_{FD} + \alpha_2 \mathcal{L}_{bbox}), \quad (13)
\]

where \(\mathcal{M}_i, i = 0, 1, \ldots, d\) are the set of matched Fourier descriptor pairs of \(i\)-th decoder layer, i.e., \(\forall (e, \hat{e}) \in \mathcal{M}_i, e \in \mathcal{P}_i\), where \(\mathcal{P}_i\) is the positive sample set obtained by DMS at the \(i\)-th decoding layer. As shown in Fig. 3, we visualize where each regression loss is applied.
TABLE I
COMPARISON WITH RECENT STATE-OF-THE-ART METHODS ON ICDAR 2015, MSRA-TD500, CTW1500 AND TOTALTEXT UNDER THE PROTOCOL OF IOU@0.5, WHERE 'Ext.' DENOTES EXTRA TRAINING DATA. WE USE RESNET50 AS THE DEFAULT BACKBONE FOR ALL ALGORITHMS.

| Methods | Venue | Backbone | Ext. | ICDAR2015 R(%) | P(%) | F(%) | RSRA-TD500 R(%) | P(%) | F(%) | CTW1500 R(%) | P(%) | F(%) | TOTALTEXT R(%) | P(%) | F(%) |
|---------|-------|----------|------|----------------|------|------|----------------|------|------|----------------|------|------|----------------|------|------|
| TextSnake [25] | ECCV 18 | VGG16 [26] | ✓ | 80.4 | 84.9 | 82.6 | 7.9 | 13.2 | 10.8 | 85.3 | 81.9 | 75.6 | 74.5 | 82.7 | 78.4 |
| PAN [7] | ICCV’19 | ✓ | 8.1 | 84.0 | 82.9 | 83.8 | 84.4 | 84.1 | 81.2 | 86.4 | 83.7 | 81.0 | 89.3 | 85.0 |
| DB [27] | AAAI’20 Res50-DCN [28] | ✓ | 83.2 | 91.8 | 87.3 | 79.2 | 91.5 | 84.9 | 80.2 | 86.9 | 83.4 | 82.5 | 87.1 | 84.7 |
| [8] | CVPRW’21 | ✓ | 78.3 | 89.8 | 83.7 | 83.8 | 90.9 | 87.2 | - | - | - | - | - | - |
| PCR [19] | CVPR’21 DLA34 [19] | ✓ | - | - | - | 83.5 | 90.8 | 87.0 | 82.3 | 87.2 | 84.7 | 82.0 | 88.5 | 85.2 |
| LASNet [29] | ICME’22 | ✓ | 84.0 | 91.2 | 87.4 | 80.7 | 87.0 | 83.7 | - | - | - | - | - | - |
| MSNet [30] | ICME’22 | ✓ | 79.6 | 89.7 | 84.4 | - | - | - | 81.0 | 86.9 | 83.8 | 82.8 | 89.1 | 85.8 |
| PSENet [20] | CVPR’19 | ✓ | 79.7 | 81.5 | 80.6 | - | - | - | 75.6 | 80.6 | 78.0 | 75.1 | 81.8 | 78.3 |
| PAN [7] | ICCV’19 | ✓ | 77.8 | 82.9 | 80.3 | 77.3 | 80.7 | 87.9 | 77.7 | 84.6 | 81.0 | 79.4 | 88.0 | 83.5 |
| TextRay [22] | MM’20 | ✓ | - | - | - | - | - | - | 89.4 | 82.8 | 81.6 | 77.9 | 83.5 | 80.6 |
| PCR [19] | CVPR’21 | ✓ | - | - | - | 77.8 | 87.6 | 82.4 | 79.8 | 85.3 | 82.4 | 80.2 | 86.1 | 83.1 |
| FCENet [9] | CVPR’21 | ✓ | 84.2 | 85.1 | 84.6 | - | - | - | 80.7 | 85.7 | 83.1 | 79.8 | 87.4 | 83.4 |
| FANet | ✓ | 87.6 | 85.0 | 86.3 | 84.2 | 82.1 | 88.0 | 84.0 | 87.8 | 85.9 | 84.7 | 87.1 | 85.9 |

TABLE II
CONVERGENCE PERFORMANCE COMPARISON WITH RECENT STATE-OF-THE-ART METHODS. WE USE F-MEASURE AS THE EVALUATION PROTOCOL, AND (30, 100, 500)E MEANS TRAINING [30, 100, 500] EPOCHS ON THE DATASET. ‘†’ DENOTE ORIGINAL HUNGARIAN MATCHING STRATEGY (HMS).

| Methods | ICDAR2015 30e | MSRA-TD500 100e 500e | CTW1500 300e 500e | TOTALTEXT 50e |
|---------|-------------|----------------|---------------|-------------|
| PAN [7] | 62.3 73.8 74.9 | 61.4 65.1 70.0 | 72.0 73.4 74.6 | 66.9 73.7 76.0 |
| FANet | 65.6 78.5 83.5 | 47.2 62.2 74.2 | 65.5 75.6 81.0 | 71.3 80.2 83.0 |
| FANet† | 73.9 82.6 81.3 | 82.1 70.6 81.3 | 73.8 91.2 84.6 | 79.0 91.2 84.6 |
| FANet | 77.2 84.1 84.7 | 70.7 84.8 87.3 | 79.1 83.3 84.9 | 81.9 84.1 84.8 |

IV. EXPERIMENT

A. Implementation details

The backbone of FANet is ResNet-50 which is pre-trained on ImageNet. Following FCENet [9], during the target generation stage, we sample equidistantly a fixed number N (N = 400 in our experiments) points on the text contour, obtaining the resampled point sequence, i.e., \{\mathcal{Z}(\frac{c}{n})\}, n \in \{0, ..., N - 1\}. Then, we transform the resampled point sequence into its corresponding normalized Fourier descriptor \(\hat{c}\) with Eq. 1.

The experiments are conducted on the workstation with 8 Tesla V100 GPU. For the experiments without pre-training, we set the match number of the DMS \(N_m\) to 3 by default and MSRA-TD500 to 10 and train FANet for 500 epochs separately on the dataset we report the results. For the experiment with pre-training, we first pre-trained our model for 25 epochs on COCOTextv2 [33], and then finetune our model for 25 epochs on the benchmark datasets respectively, we use HMS as default and DMS \(N_m\) to 5 for MSRA-TD500 by default when FANet is pre-trained use COCOTextv2.

B. Comparison with the state-of-the-art methods

**Straight text.** The proposed FANet achieves SOTA performance with SOTA methods on MSRA-TD500, achieves comparable performance with SOTA methods on ICDAR2015. The proposed FANet outperforms the previous SOTA algorithm [8] by 0.8% (88.0% vs. 87.2%) in F-measure on MSRA-TD500.

**Curve text.** The proposed FANet achieves the SOTA performance on CTW1500 and TotalText. The proposed FANet outperforms the previous SOTA method [19] by 1.2% (85.9% vs. 84.7%) in F-measure on CTW1500, outperforms the previous SOTA method [30] by 0.1% (85.9% vs. 85.8%) in F-measure on TotalText.

**Convergence Performance.** Under the constraint of limited epochs, the proposed FANet can still achieve good results. In particular, with only 30 epochs of training, FANet surpass FCENet by 11.6%, 23.5%, 13.6%, 10.6%, surpass PAN by 14.9%, 9.3%, 7.1%, 15.0% based on F-measure on ICDAR2015, MSRA-TD500, CTW1500 and TotalText respectively, which shows that the proposed FANet can achieve much better performance than the previous SOTA methods with fewer epochs. With only 100 epochs of training, FANet can achieve the detection performance of 84.8% on MSRA-TD500, 83.3% on CTW1500 and 84.1% on TotalText, which already have achieved the SOTA performance if only compared with the methods without pre-training.

**Inference speed and model parameters.** The model parameters of FANet is 41.2M. FANet runs 10 FPS at the test size of 1080 on a single RTX3090.
TABLE IV
ABLATION OF ACTIVATION FUNCTIONS. AF DENOTES ACTIVATION FUNCTION AND NT DENOTES WHETHER TO USE THE NORMALIZED FOURIER DESCRIPTOR AS THE REGRESSION TARGET.

| AF  | NT | CTW1500 R(%) | P(%) | F(%) | MSRA-TD500 R(%) | P(%) | F(%) |
|-----|-----|--------------|------|------|----------------|------|------|
| tanh | ✓   | 82.9         | 85.7 | 84.3 | 75.8           | 86.6 | 80.8 |
| ours | ✓   | 82.5         | 86.9 | 84.6 | 79.2           | 88.3 | 83.5 |

TABLE V
ABLATION OF OUR PROPOSED ITERATIVE TEXT DECODING NETWORK.

| Refinement module | CTW1500 R(%) | P(%) | F(%) | MSRA-TD500 R(%) | P(%) | F(%) |
|-------------------|--------------|------|------|----------------|------|------|
| w/o refinement    | 78.9         | 82.0 | 80.4 | 19.1           | 9.2  | 12.4 |
| w/o reference     | 76.4         | 79.9 | 78.1 | 40.2           | 43.3 | 41.7 |
| add-based         | 77.7         | 85.0 | 81.2 | 36.1           | 47.3 | 40.9 |
| mul-based         | 78.8         | 85.4 | 82.0 | 50.3           | 65.7 | 57.0 |
| ours              | 79.2         | 87.4 | 83.1 | 58.8           | 67.7 | 62.9 |

C. Ablation study

Baseline and core components. Based on deformable DETR, we directly use normalized Fourier descriptor to replace bounding box as the regression target to obtain our baseline algorithm. As shown in Table III, the baseline of the proposed method reaches 80.4% on CTW1500 and 12.4% on MSRA-TD500. Compared with our baseline, ITDN can bring relative improvements of 2.7% (83.1% vs. 80.4%) and 50.5% (62.9% vs. 12.4%) on CTW1500 and MSRA-TD500 respectively. Then, the addition of FDPN can bring relative improvements of 1.5% (84.6% vs. 83.1%) and 20.6% (83.5% vs. 62.9%) on CTW1500 and MSRA-TD500 respectively. Finally, the DMS can bring relative improvements of 0.3% (84.9% vs. 84.6%) and 3.8% (87.3% vs. 83.5%) on CTW1500 and MSRA-TD500 respectively.

Activation function. As shown in line 2 of Table IV, the network will not be trainable if the activation function in the proposed FANet is replaced by identity function. The experimental results show that the proposed activation function achieves better results than tanh and σ.

Iterative Text Decoding Network. As shown in Table V, not taking the Fourier descriptor as the reference location for the Multi-Scale Deformable Attention module (line 2), replacing the refinement function by the refinement function based on addition (line 3) or multiplication (line 4) will all reduce the performance of FANet.

V. CONCLUSION

This paper focuses on fast localization learning and accurate detection for scene text detection. Rooted on deformable DETR, we design a Fourier Descriptor Proposal Network (FDPN), an Iterative Text Decoding Network (ITDN), and a Dense Matching Strategy (DMS) to improve the convergence speed and accuracy significantly. Together with the above components, we propose FANet, a Fast convergence and Accurate scene text detection Network, which can achieve the SOTA performance on MSRA-TD500, CTW1500, and TotalText datasets.

REFERENCES

[1] Q. Ye and D. S. Doermann, “Text detection and recognition in imagery: A survey,” IEEE TPAMI, 2015.

[2] Y. Zhu, C. Yao, and X. Bai, “Scene text detection and recognition: recent advances and future trends,” Frontiers Comput. Sci., 2016.

[3] X. Liu, G. Meng, and C. Pan, “Scene text detection and recognition with advances in deep learning: a survey,” IJIDAR, 2019.

[4] W. Wu, N. Lu, E. Xie, and et al., “Synthetic-to-real unsupervised domain adaptation for scene text detection in the wild.” in ACCV, 2020.

[5] W. Wu, J. Xing, C. Yang, and et al., “Texts as lines: text detection with weak supervision,” Mathematical Problems in Engineering, 2020.

[6] S. Xing, C. Gong, H. Guo, and et al., “Unsupervised domain adaptation GAN inversion for image editing,” CoRR, 2022.

[7] W. Wang, E. Xie, and et al., “Efficient and accurate arbitrary-shaped text detection with pixel aggregation network,” in IEEE ICCV, 2019.

[8] Z. Raisi, M. A. Naiel, G. Younes, S. Wardell, and J. S. Zelek, “Transformer-based text detection in the wild,” in IEEE CVPRW, 2021.

[9] Y. Zhu, J. Chen, L. Liang, and et al., “Fourier contour embedding for arbitrary-shaped text detection,” in IEEE CVPR, 2021.

[10] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, “Mask R-CNN,” in IEEE ICCV, 2017.

[11] W. Wu, J. Xing, and H. Zhou, “Textcohesion: Detecting text for arbitrary shapes,” arXiv preprint arXiv:1904.12640, 2019.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in NIPS, 2017.

[13] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, “Deformable DETR: deformable transformers for end-to-end object detection,” in ICLR, 2021.

[14] N. Carion, F. Massa, G. Synnaeve, and et al., “End-to-end object detection with transformers,” in ECCV, 2020.

[15] Y. Liu, L. Jin, S. Zhang, and et al., “Curved scene text detection via transverse and longitudinal sequence connection,” PR, 2019.

[16] C. Yao, X. Bai, W. Liu, Y. Ma, and et al., “ZText: Deep learning for unstructured text with arbitrary orientations in natural images,” in IEEE CVPR, 2012.

[17] W. Wu, Y. Cai, D. Zhang, and et al., “A bilingual, openworld video text detection model and end-to-end video text spotting with transformer,” NIPS, 2021.

[18] W. Wu, D. Zhang, Y. Fu, and et al., “End-to-end video text spotting with transformer,” arXiv preprint arXiv:2203.10539, 2022.

[19] P. Dai, S. Zhang, H. Zhang, and X. Cao, “Progressive contour regression for arbitrary-shape scene text detection,” in IEEE CVPR, 2021.

[20] W. Wang, E. Xie, X. Li, and et al., “Shape robust text detection with progressive scale expansion network,” in IEEE CVPR, 2019.

[21] X. Zhou, C. Yao, H. Wen, W. Yuan, S. Zhou, W. He, and J. Liang, “East: an efficient and accurate scene text detector,” in IEEE CVPR, 2017.

[22] F. Wang, Y. Chen, F. Wu, and X. Li, “Textray: Contour-based geometric modeling for arbitrary-shaped scene text detection,” in ACM MM, 2020.

[23] Y. Liu, H. Chen, C. Shen, and et al., “Abnet: Real-time scene text spotting with adaptive bezier-curve network,” in IEEE CVPR, 2020.

[24] C. Gong, X. Zhang, and Y. Niu, “Identification of epilepsy from intracranial eeg signals by using different neural network models,” in Computational Biology and Chemistry, vol. 87, 2020.

[25] S. Long, J. Ruan, W. Zhang, and et al., “Textsnake: A flexible representation for detecting text of arbitrary shapes,” in ECCV, 2018.

[26] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in ICLR, 2015.

[27] M. Liao, Z. Wan, C. Yao, K. Chen, and X. Bai, “Real-time scene text detection with differentiable binarization,” in AAAI, 2020.

[28] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, “Deformable convolutional networks,” in IEEE ICCV, 2017.

[29] Z. Guo, P. Fang, H. Li, and et al., “Location-aware feature selection network for multi-oriented scene text detection,” in IEEE ICME, 2022.

[30] X. Liu, X. Liu, B. Wang, X. Luo, and X. Xu, “Arbitrary-shaped scene text detection with scoring mask quality,” in IEEE ICME, 2022.

[31] H. Rezatofighi, N. Tsoi, and et al., “Generalized intersection over union: A metric and a loss for bounding box regression,” in IEEE CVPR, 2019.

[32] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in IEEE ICCV, 2017.

[33] A. Veit, T. Matera, L. Neumann, J. Matas, and S. Belongie, “Coco-text: Dataset and benchmark for text detection and recognition in natural images,” in arXiv preprint arXiv:1601.07140, 2016.