Fourier Contour Embedding for Arbitrary-Shaped Text Detection

Yiqin Zhu[1†], Jianyong Chen[1†], Lingyu Liang[1,3*], Zhanghui Kuang[2*], Lianwen Jin[1,3], Wayne Zhang[2,4,5]
[1]South China University of Technology [2]SenseTime Research [3]Pazhou Lab
[4]Qing Yuan Research Institute, Shanghai Jiao Tong University [5]Shanghai AI Laboratory, Shanghai, China
scut_zhuyiqin@163.com, {theochan666,lianglysky,lianwen.jin}@gmail.com
{kuangzhanghui,wayne.zhang}@sensetime.com

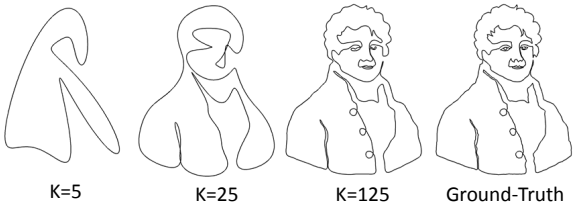
Abstract

One of the main challenges for arbitrary-shaped text detection is to design a good text instance representation that allows networks to learn diverse text geometry variances. Most of existing methods model text instances in image spatial domain via masks or contour point sequences in the Cartesian or the polar coordinate system. However, the mask representation might lead to expensive post-processing, while the point sequence one may have limited capability to model texts with highly-curved shapes. To tackle these problems, we model text instances in the Fourier domain and propose one novel Fourier Contour Embedding (FCE) method to represent arbitrary shaped text contours as compact signatures. We further construct FCENet with a backbone, feature pyramid networks (FPN) and a simple post-processing with the Inverse Fourier Transformation (IFT) and Non-Maximum Suppression (NMS). Different from previous methods, FCENet first predicts compact Fourier signatures of text instances, and then reconstructs text contours via IFT and NMS during test. Extensive experiments demonstrate that FCE is accurate and robust to fit contours of scene texts even with highly-curved shapes, and also validate the effectiveness and the good generalization of FCENet for arbitrary-shaped text detection. Furthermore, experimental results show that our FCENet is superior to the state-of-the-art (SOTA) methods on CTW1500 and Total-Text, especially on challenging highly-curved text subset.

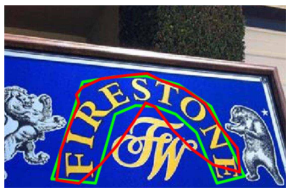
1. Introduction

Benefiting from the development of object detection [3, 9, 10, 16, 23] and instance segmentation [4, 11], text detection has achieved significant progress [1, 7, 8, 14, 22, 24, 25, 27–29, 31, 35–37]. Text detection methods can be roughly divided into segmentation-based approaches [14, 15, 19, 21, 22, 24, 27, 28, 30, 32], and regression-based approaches [12, 25, 33, 35, 36].

Recent research focus has shifted from horizontal or multi-oriented text detection [7, 22, 31, 37] to more challenging arbitrary-shaped text detection [1, 8, 14, 24, 25, 27–29, 35, 36]. Compared to multi-oriented text detection, text instance representations play an indispensable role in arbitrary-shaped text detection. A good representation should be simple and compact with good generalization ability to avoid overfitting. However, designing



(a) Fourier contour fitting with progressive approximation.



(b) TextRay contour [25]

(c) Fourier contour

Figure 1: Comparison with Fourier contour and TextRay contour [25] representations. (a) shows Fourier contour can fit extremely complicated object shapes and get better approximation as the Fourier degree k increases. (b) and (c) compare the TextRay contours and our proposed Fourier contours, where the ground-truth contours are in green and the reconstructed ones are in red. TextRay fails to model highly-curved texts (best viewed in color).

[†]Yiqin Zhu and Jianyong Chen contributed equally to this work.

[*]**Corresponding authors: Lingyu Liang, Zhanghui Kuang.** This research is supported by NSFC (Grant No.61936003), GD-NSF (Grant No.2017A030312006, No.2019A1515011045), SenseTime Research Fund for Young Scholars and CAAI-Huawei MindSpore Open Fund.

a compact text instance representation is not straightforward, because fitting diverse geometry variances of text instances is challenging. Existing arbitrary-shaped text detection approaches represent text instances in the spatial domain of images. They models texts via per-pixel masks [1, 8, 14, 24, 27–29, 32], contour point sequences in the Cartesian system [12, 35, 36] or those in the polar system [25]. Spatial domain based methods have clear drawbacks. Mask representation may lead to intrinsically computationally expensive post-processing, and frequently requires large training data, and contour point sequences may have limited capability to model highly-curved texts.

In this paper, we model text instance contours in the Fourier domain instead of the spatial domain via the Fourier transformation, which can fit any closed contour with progressive approximation in a robust and simple manner. Fig. 1a illustrates that Fourier transformation can accurately fit extremely complicated shapes (*e.g.*, a portrait sketch) with very compact signatures (*e.g.*, $K = 125$ only), and shows that as the Fourier degree k increases, the reconstructed shape approximates the ground truth better. Compared to TextRay [25], a SOTA text contour point sequence in the polar coordinate system, our proposed Fourier contour representation can model high-curved texts better as shown in Fig. 1b-c.

To this end, we propose Fourier Contour Embedding (FCE) method to convert text instance contours from point sequences into Fourier signature vectors. Firstly, we propose a resampling scheme to obtain a fixed number of dense points on each text contour. To maintain the uniqueness of the resulted Fourier signature vector, we set the rightmost intersection between the text contour and the horizontal line through the text center point as the sampling start point, fix the sampling direction as the clockwise direction, and keep the sampling interval along the text contour unchanged. Secondly, the sampled point sequences of contours in the spatial domain are embedded into the Fourier domain via the Fourier transformation (FT).

The advantages of FCE for text instance representation are three-fold:

- **Flexible:** Any closed contour, including extremely complicated shapes, can accurately be fitted;
- **Compactness:** The Fourier signature vectors are compact. In our experiments, our proposed FCE with the degree $K = 5$ can achieve very accurate approximation of texts.
- **Simplicity:** The conversion between a sampled point sequence and a Fourier signature vector of text contours is formulated as FT and Inverse FT. So the FCE method is easy to implement without introducing complex post-processing.

Equipped with the FCE, we further construct FCENet for arbitrary-shaped text detection. Particularly, it consists of a backbone of ResNet50 with deformable convolutional networks (DCN) [38], feature pyramid networks (FPN) [9] and the Fourier prediction header. The header has two individual branches. Namely, the classification branch, and the regression branch. The former predicts text region masks and text center region masks. The latter predicts text Fourier signature vectors in the Fourier domain, which are fed into the Inverse Fourier Transformation (IFT) to reconstruct text contour point sequences. Ground truth text contour point sequences are used as supervision signals. Thanks to the resampling scheme of FCE, our loss in the regression branch is compatible across different datasets, although datasets such as CTW1500 [13] and Total-Text [2] have different numbers of contour points for each text instance.

Experiments validate the effectiveness and good generalization ability of FCENet for arbitrary shaped text detection. Moreover, our FCENet is superior to the state-of-the-art (SOTA) methods on CTW1500 and Total-Text, especially on their highly-curved text subset.

We summarize the contributions of this work as follows:

- We propose Fourier Contour Embedding (FCE) method, which can accurately approximate any closed shapes, including arbitrary shaped text contours, as compact Fourier signature vectors.
- We propose FCENet which first predicts Fourier signature vectors of text instances in the Fourier domain, and then reconstructs text contour point sequences in the image spatial domain via Inverse Fourier Transformation (IFT). It can be learned end-to-end, and be inferred without any complex post processing.
- We extensively evaluate the proposed FCE and FCENet. Experimental results validate the good representation of FCE, especially on highly-curved texts, the generalization ability of FCENet when training on small datasets. Moreover, it has been shown that FCENet achieves the state-of-the-art performance on CTW1500 and Total-Text.

2. Related Work

2.1. Segmentation-Based Methods

These methods mainly draw inspiration from semantic segmentation, which implicitly encodes text instances with per-pixel masks [1, 14, 19, 22, 24, 27, 28, 30, 32, 34]. Most of these methods follow a component-grouping paradigm, which first detect components of scene text instances and then aggregate these components to obtain final mask outputs.

For pixel-based methods, pixel-level score maps are firstly obtained using instance/semantic segmentation framework, and then text pixels are grouped to obtain the output text masks [24, 28, 30, 32]. To further improve the performance, some methods would perform prediction on a transformed space, and then reconstruct the final maps. For example, Tian *et al*. [24] assumed each text instance as a cluster and predicted an embedding map via pixel clustering; TextField [32] generates candidate text parts via linking neighbor pixels with a deep direction field.

For segment-based methods, segments containing parts of words or text lines (fragments) [14, 15, 19, 21, 22, 27], or characters [1, 34] are firstly detected, and then segments are grouped into the whole words/text-line. PSENet [27] detects each text instance with corresponding kernels, and adopts a progressive scale algorithm to gradually expand the predefined kernels and obtain the final detection. SegLink++ [21] achieves dense and arbitrary-shaped scene text detection using instance-aware component grouping with minimum spanning tree. CRAFT [1] obtains character-level detection and estimates the affinity between characters to achieve the final detection.

Some methods train the predictor in a transformed space, and reconstruct the output masks via the predicted features. For example, Tian *et al*. [24] constructed a discriminative representation via embedding pixels into a space where pixels of the same text tend to be in the same clusters and vise versa; Xu *et al*. [32] proposed TextField to learn one direction field to separate adjacent text instances.

2.2. Regression-Based Methods

Regression-based methods are complementary to segmentation-based methods, which explicitly encode text instances with contours (point sequences) of text regions. They aim at adopting the direct shape modeling of text instances to handle complex geometric variances [7, 25, 33, 35–37], and are often simpler and easier to train. However, the constrained representation capability of point sequences for complex text instances may limit the performance of the networks.

To tackle this problem, many modules are elaborately designed to further improve the flexibility of point sequence representation. LOMO [35] introduces an iterative refinement module (IRM) and a shape expression module (SEM) to progressively refine the text localization of a direct regression. Zhang *et al*. [36] used CNNs to regress the geometry attributes (*e.g*., height, width, and orientation) of a series of small rectangular components divided from text instances, and introduced one Graph Convolutional Network (GCN) to infer the linkages between different text components. TextRay [25] formulates the text contours in the polar system and proposes a single-shot anchor-free framework to learn the geometric parameters. Liu *et al*. [12] introduced

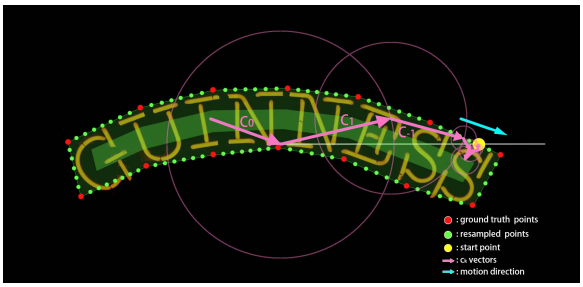


Figure 2: Illustration of FCE. It contains two stages, where **Resampling** obtains dense point sequences (in green) based on ground truth points (in red); **Fourier Transformation** is used to compute Fourier coefficients $\mathbf{c}_k$ with the resampled point sequences. A contour can be reconstructed by the combination of different fixed-frequency circular motions (indicated by pink circles) with the hand direction $\mathbf{c}_k$.

Bezier curves to parameterize curved texts and achieved the SOTA performance in scene text spotting with BezierAlign.

Recent works indicate that effective contour modeling is essential for irregular text instances detection [25, 35, 36] and the downstream recognition [12]. Therefore, it would be significant to design a flexible yet simple representation for arbitrary shaped text detection.

2.3. Explicit vs. Implicit Text Shape Representation

From a perspective of text shape representation, current model can be roughly divided into two categories. Namely, approaches which implicitly model text shapes via per-pixel masks [1, 8, 14, 27–29] or masks reconstructed by transformed features [24, 32], and approaches which explicitly model text shapes using point sequences of contours in the Cartesian system [35, 36] or the polar system [25].

However, per-pixel masks may cause intrinsically high computational complexity of networks (*e.g*., complex post-processing) and require large training data, while point sequences sampled on contours may has limited representation capability and requires deliberately-designed refinement or inference [15, 25, 35, 36].

To tackle this problem, Liu *et al*. [12] introduced Bezier curves to parameterize curved texts, but the control point setting of Bezier curves may limit its representation capability for some cases, as shown in Sec. 4.6. In this paper, text instances are formulated in the Fourier domain, which allows to fit any closed continuous contour with robust and simple manners. In the following section, we would explore the potential of FCE for arbitrary shaped text detection.

3. Approach

In this section, we first introduce the proposed Fourier Contour Embedding (FCE) method, which can approximate arbitrary-shaped text contours as compact Fourier signa-

ture vectors. Then we propose FCENet to detect arbitrary-shaped texts, equipped with FCE.

3.1. Fourier Contour Embedding

We use one complex-value function $f : \mathbb{R} \mapsto \mathbb{C}$ of a real variable $t \in [0, 1]$ to represent any text closed contour as follows:

$$f(t) = x(t) + iy(t), \tag{1}$$

where i represents the imaginary unit. $(x(t), y(t))$ denotes the spatial coordinate at the specific time t. Since f is a closed contour, $f(t) = f(t + 1)$. $f(t)$ can be reformulated by Inverse Fourier Transformation (IFT) as:

$$f(t) = f(t, \mathbf{c}) = \sum_{k=-\infty}^{+\infty} \mathbf{c}_k e^{2\pi i k t}, \tag{2}$$

where $k \in \mathbb{Z}$ represents the frequency, and $\mathbf{c}_k$ is the complex-value Fourier coefficient used to characterize the initial state of the frequency k. Each component $\mathbf{c}_k e^{2\pi i k t}$ in Eq. 2 indicates a circular motion with fixed-frequency k with a given initial hand direction vector $\mathbf{c}_k$, Thus, the contour can be regarded as the combination of different frequent circular motions as the pink circles shown in Fig. 2. From Eq. 2, we observe that the low frequency components are in charge of the rough text contours, while the high are in charge of the details of contours. We empirically find that preserving $K-$lowest ($K = 5$ in our experiments) frequencies only while discarding others can obtain satisfactory approximation of text contours, as shown in Fig. 5.

Since we cannot obtain the analytical form of text contour function f in real applications, we can discretize the continual function f into N points as $\{f(\frac{n}{N})\}$ with $n \in [1, \ldots, N]$. In this case, the $\mathbf{c}_k$ in Eq. 2 can be computed via the Fourier Transformation as:

$$\mathbf{c}_k = \frac{1}{N} \sum_{n=1}^{N} f(\frac{n}{N}) e^{-2\pi i k \frac{n}{N}}, \tag{3}$$

where $\mathbf{c}_k = u_k + iv_k$ with u_k as the real part and v_k as the image part of a complex number. Specially, when $k = 0$, $\mathbf{c}_0 = u_0 + iv_0 = \frac{1}{N} \sum_n f(\frac{n}{N})$ is the center position of the contour. For any text contour f, our proposed Fourier Contour Embedding (FCE) method can represent it in the Fourier domain as a compact $2(2K+1)$ dimensional vector $[u_{-K}, v_{-K}, \cdots, u_0, v_0, \cdots, u_K, v_K]$, dubbed Fourier signature vector.

Our FCE method consists of two stages. Namely, the resampling stage and the Fourier transformation stage. Concretely, in the resampling stage, we sample equidistantly a fixed number N ($N = 400$ in our experiments) points on the text contour, obtaining the resampled point sequence $\{f(\frac{1}{N}), \cdots, f(1)\}$. Note that this resampling is necessary since different datasets have different numbers of ground truth points for text instances, and they are relatively small. *e.g.*, there are 14 in CTW1500 [13] while $4 \sim 8$ in Total-Text [2]. The resampling strategy makes our FCE is compatible to all datasets with the same setting. In the Fourier transformation stages, the resampled point sequence is transformed into its corresponding Fourier signature vector.

Uniqueness of Fourier Signature Vector. From the above procedure of FCE, it is easy to see that different resampled point sequences can result in different Fourier signature vectors even for the same text contour. To make the signature vector of one specific text unique, and more stable network training, we make constrains on the starting point, the sampling direction, and moving speed of $f(t)$:

- **Starting point:** We set our starting point $f(0)$ (or $f(1)$) to be right most intersection point between the horizontal line through the center point (u_0, v_0) and the text contour.
- **Sampling direction:** We always resample the points along the text contour in the clockwise direction.
- **Uniform speed:** We resample points uniformly on the text contour, and the distance between every two adjacent points keeps unchanged to ensure a uniform speed.

3.2. FCENet

Equipped with the FCE, we further propose the anchor free network FCENet for arbitrary-shaped text detection.

Network Architectures. Our proposed FCENet employs a top-down scheme. As shown in Fig. 3, it contains ResNet50 [5] with DCN [38] as backbone, and FPN [9] as neck to extract multi-scale features, and the Fourier prediction header. We conduct prediction on the feature map P3, P4 and P5 of FPN. The header has two branches, which are responsible for classification and regression respectively. Each branch consists of three 3×3 convolutional layers and one 1×1 convolutional layer, each of which is followed by one ReLU nonlinear activation layer.

In the classification branch, we predict the per-pixel masks of Text Regions (TR). We find that Text Center Region (TCR) prediction can further improve the performance. We believe this is because it can effectively filter out low-quality predictions around text boundaries.

In the regression branch, the Fourier signature vector of one text is regressed for each pixel in the text. To deal with text instances of different scales, the features of P3, P4 and P5 are responsible for small, medium and large text instances, respectively.

The detection results would be reconstructed from the Fourier domain to the spatial domain by IFT and NMS, as shown in Fig. 4.

Ground-Truth Generation. For the classification task, we use the method of [14] to obtain text center region (TCR)

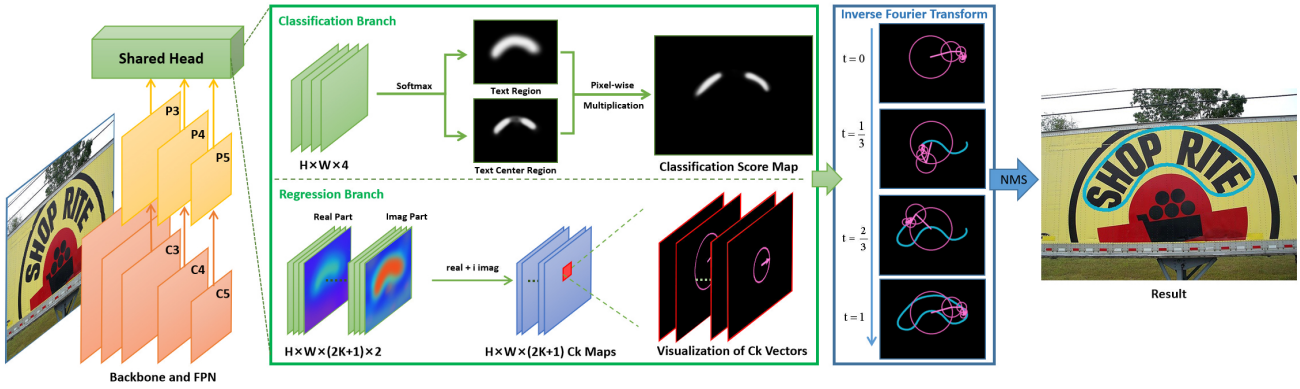
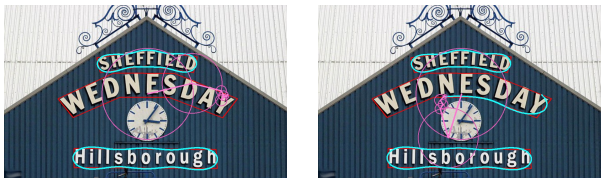


Figure 3: The overall framework of the proposed FCENet. Given an image, its features extracted by the backbone and FPN, are fed into the shared header to detect texts. In the header, the classification branch predicts both the heat maps of text regions and those of text center regions, which are pixel-wise multiplied, resulting in the the classification score map. The regression branch predicts the Fourier signature vectors, which are used to reconstruct text contours via the Inverse Fourier transformation (IFT). Given the reconstructed text contours with corresponding classification scores, the final detected texts are obtained with non-maximum suppression (NMS) [18].



(a) $t = 0$ (b) $t = 1/3$ (c) $t = 2/3$ (d) $t = 1$

Figure 4: Fourier contour reconstruction (blue) via IFT and NMS for arbitrary-shaped texts (red denotes ground-truth) at different time t.

masks via shrinking texts with the shrinking factor being 0.3 (see the green mask in Fig. 2). For the regression task, we compute the Fourier signature vectors $\overline{\mathbf{c}}$ of the ground truth text contours via the proposed FCE method. Note that for all pixels in the mask of one text instance, we predict the text contour, and thus need one Fourier signature vector $\overline{\mathbf{c}}$ with the pixel being the $(0, 0)$ point of the complex coordinate system. Different pixels in the same text instance share the same Fourier signature vector except $\mathbf{c}_0$.

Losses. The optimization objective of FCE-base network is given by:

$$\mathcal{L} = \mathcal{L}_{cls} + \lambda \mathcal{L}_{reg}, \tag{4}$$

where $\mathcal{L}_{cls}$ and and $\mathcal{L}_{reg}$ are the loss for the classification branch and that for the regression branch, respectively. λ is a parameter to balance $\mathcal{L}_{cls}$ and $\mathcal{L}_{reg}$. We fix $\lambda = 1$ in our experiments. $\mathcal{L}_{cls}$ consists of two parts as:

$$\mathcal{L}_{cls} = \mathcal{L}_{tr} + \mathcal{L}_{tcr}, \tag{5}$$

where $\mathcal{L}_{tr}$ and $\mathcal{L}_{tcr}$ are the cross entropy loss for the text region (TR) and that of the text center region (TCR), respectively. To solve the sample imbalance problem, OHEM [20] is adopted for $\mathcal{L}_{tr}$ with the ratio between negative and positive samples being $3 : 1$.

For $\mathcal{L}_{reg}$, we do not minimize the distances between the predicted Fourier signature vectors and their corresponding ground truth. In contrast, we minimize their reconstructed text contours in the image spatial domain which are more related to the text detection quality. Formally,

$$\mathcal{L}_{reg} = \frac{1}{N'} \sum_{i \in \mathcal{T}} \sum_{n=1}^{N'} w_i l_1(F^{-1}(\frac{n}{N'}, \overline{\mathbf{c}}_i), F^{-1}(\frac{n}{N'}, \hat{\mathbf{c}}_i)), \tag{6}$$

where l_1 is the smooth$-l_1$ loss [17] used for regression, and $F^{-1}(\cdot)$ is the IFT of Eq. 2. $\mathcal{T}$ is the text region pixel index set. $\overline{\mathbf{c}}_i$ and $\hat{\mathbf{c}}_i$ are the text ground truth Fourier signature vector and the predicted one for pixel i. $w_i = 1$ if pixel i in its corresponding text center region while 0.5 if not. N' is the sampling number on the text contours. If N' is too small (typically $N' < 30$), it would probably cause over-fitting. Therefore, we fix $N' = 50$ in our experiments.

The regression loss is extremely important in our FCENet. In ablation studies of Sec. 4.4, results show that it brings absolute 6.9% and 9.3% h-mean improvement on CTW 1500 and Total-text respectively.

4. Experiments

In this section, we first verified the effectiveness of FCE to model text instances, compared with two recent SOTA arbitrary shaped text representation methods, *i.e.*, TextRay [25] and ABCNet [12]. We then evaluated FCENet for text detection. Particularly, we conducted ablation studies for the effectiveness of each component, and the generalization ability by decreasing the training data; we also made extensive comparison with the recent SOTA methods on CTW1500 [13] and Total-Text [2] benchmarks. Since these benchmark datasets also contain a large amount of non-curved texts, we built a much more challenging subset containing highly-curved or highly irregular text for further evaluation.

4.1. Datasets

CTW1500 [13] contains both English and Chinese texts with text-line level annotations, where 1000 images for training, and 500 images for testing.

Total-Text [2] was collected form various scenes, including text-like background clutter and low-contrast texts, with word-level polygon annotations, where 1255 images for training and 300 images for testing.

ICDAR2015 [6] is a multi-orientated and street-viewed dataset which consists of 1000 training and 500 testing images. The annotations are word-level with four vertices.

4.2. Implementation Details

The backbone of FCENet contains ResNet50 with DCN [38] and a FPN [9], as shown in Fig. 3. Each of the regression and classification branch consists of three 3×3 convolutional layers and one 1×1 convolutional layer, whose kernel numbers are set as $[128, 64, 32, 32]$. The text scale ranges of P3, P4 and P5 are set to $[0, 0.4]$, $[0.3, 0.7]$, and $[0.6, 1]$ of the image size respectively, where the overlapping range is to increase the recall rate.

We resize images to 800×800 , and adopt data augmentation strategies, including random crop, random rotations, random horizontal flipping, color jitter and contrast jitter during training. The models are trained using two 2080Ti GPUs with batch size set to 8. Stochastic gradient descent(SGD) is adopted as the optimizer with the weight decay of 0.001, and the momentum of 0.9. The initialized learning rate is 0.001, which is reduced $0.8\times$ every 200 epoches.

During testing, the images are resized as follows: For CTW1500, we first resize the short edge of images to 640, and then resize the long edge of the resulted images to 1280 if it is bigger than 1280. For Total-Text, we first resize the short edge of images to 960, and then resize the long edge of the resulted images to 1280 if it is bigger than 1280. For ICDAR2015, we resize the long edge to 2020 while keeping its original direction.

(a) $K = 3$ (b) $K = 5$ (c) $K = 10$

Figure 5: Fourier contour fitting for arbitrary-shaped texts with increasing Fourier degree K, where green contours denotes ground-truth; red contours denotes FCE fitting results.

(a) TextRay [25] (b) Ours

Figure 6: Comparisons between different text representations in terms of contour fitting ability. Green contours denote ground-truth, while red ones denote the fitting results.

Previous methods in comparisons were implemented with their open source codes, and some of them were tested on MindSpore platform[1].

4.3. Evaluation of FCE

Basic Evaluation. Theoretically, any closed continuous contour can be fitted by Fourier contour with a better approximation via increasing Fourier degree K of FCE. Results of Fig. 5 indicates that only small K can obtain satisfactory fitting for most of arbitrary-shaped texts, which verifies the strong representation ability of FCE.

Comparison. To verify the effectiveness and robustness of FCE for modeling text instances, we conduct comparisons with the recent SOTA arbitrated-shaped text detectors, TextRay [25]. Results of Fig. 6 show that TextRay fails to fit the ground-truth closely for highly-curved texts, while our FCE obtains accurate approximation. Note that our FCE us-

[1]https://github.com/mindspore-ai/mindspore

Table 1: Comparison with related methods on CTW1500, Total-Text and ICDAR2015, where 'Ext.' denotes extra training data and 'FCENet†' denotes FCENet using the ResNet50 without DCN as the backbone.

Methods	Paper	Ext.	CTW1500			Total-Text			ICDAR2015		
			R(%)	P(%)	F(%)	R(%)	P(%)	F(%)	R(%)	P(%)	F(%)
TextSnake [14]	ECCV'18	√	**85.3**	67.9	75.6	74.5	82.7	78.4	80.4	84.9	82.6
SegLink++ [21]	PR'19	√	79.8	82.8	81.3	80.9	82.1	81.5	80.3	83.7	82.0
SAEmbed [24]	CVPR'19	√	77.8	82.7	80.1	-	-	-	85.0	88.3	86.6
CRAFT [1]	CVPR'19	√	81.1	86.0	83.5	79.9	87.6	83.6	84.3	89.8	86.9
PAN [28]	ICCV'19	×	77.7	84.6	81.0	79.4	88.0	83.5	77.8	82.9	80.3
PAN [28]	ICCV'19	√	81.2	86.4	83.7	81.0	**89.3**	85.0	81.9	84.0	82.9
PSENet [27]	CVPR'19	×	75.6	80.6	78.0	75.1	81.8	78.3	79.7	81.5	80.6
PSENet [27]	CVPR'19	√	79.7	84.8	82.2	84.0	78.0	80.9	84.5	86.9	85.7
LOMO [35]	CVPR'19	√	76.5	85.7	80.8	79.3	87.6	83.3	83.5	91.3	87.2
DB [8]	AAAI'20	√	80.2	86.9	83.4	82.5	87.1	84.7	83.2	**91.8**	**87.3**
Boundary [26]	AAAI'20	√	-	-	-	83.5	85.2	84.3	88.1	82.2	85.0
DRRG [36]	CVPR'20	√	83.0	85.9	84.5	**84.9**	86.5	85.7	84.7	88.5	86.6
ContourNet [29]	CVPR'20	×	84.1	83.7	83.9	83.9	86.9	85.4	**86.1**	87.6	86.9
TextRay [25]	MM'20	√	80.4	82.8	81.6	77.9	83.5	80.6	-	-	-
ABCNet [12]	CVPR'20	√	78.5	84.4	81.4	81.3	87.9	84.5	-	-	-
FCENet†	Ours	×	80.7	85.7	83.1	79.8	87.4	83.4	84,2	85.1	84.6
FCENet	Ours	×	83.4	**87.6**	**85.5**	82.5	**89.3**	**85.8**	82.6	90.1	86.2

Table 2: Ablation studies. "TCR" denotes text center region loss, "RL" denotes regression loss in Eq. 6.

TCR	RL	CTW1500			Total-Text		
		R(%)	P(%)	F(%)	R(%)	P(%)	F(%)
-	-	74.2	80.2	77.1	71.2	81.6	76.1
-	√	78.8	83.8	81.3	73.2	84.7	78.6
√	-	74.1	83.6	78.6	72.0	81.6	76.5
√	√	83.4	87.6	85.5	82.5	89.3	85.8

Table 3: Generalization ability evaluation on CTW1500 with different amounts of training data.

Data	Methods	R(%)	P(%)	F(%)
50%	DRRG [36]	61.1	76.6	68.0
	TextRay [25]	75.5	77.8	76.6
	ABCNet [12]	71.1	80.6	75.6
	FCENet†	**76.2**	**84.9**	**80.3**
25%	DRRG [36]	44.5	70.7	54.7
	TextRay [25]	67.9	74.9	71.2
	ABCNet [12]	70.0	75.5	72.7
	FCENet†	**75.7**	**81.9**	**78.7**

es 22 dimensional parameters only while TextRay 44, which is 2 times as big as ours.

4.4. Evaluation of FCENet

Ablation Studies. To evaluate the effectiveness of the components of FCENet, we conducted ablation studies on both CTW1500 and Total-Text dataset, as shown in Table 2. The results indicate that the text center region (TCR) loss of the classification branch and the proposed regression loss (Eq. 6) of the regression branch can dramatically improve the performance of FCENet.

Generalization Ability. Benefiting from the FCE representation, FCENet requires the simple IFT and NMS post-processing only to reconstruct the complex text contours. Moreover, FCE can generate compact text representations, which allows our FCENet has better generalization ability comparing to the SOTA methods. We made comparisons with DRRG [36], TextRay [25], ABCNet [12] and our proposed FCENet on CTW1500 dataset using different amounts of training data, as shown in Table 3.

The results show that the performance of the other methods would drop dramatically when training data is reduced to 50% and 25% of the original. In contrast, our FCENet maintains good accuracy, where all accuracy recall, precision and F-measure are over 73% (precision maintains even over 80%). The results indicate the good generalization ability of our FCENet, and show the wide application potential of our method, especially in the practical scenarios with limited training samples.

4.5. Evaluation on Benchmark Datasets

We made extensive comparison with most recent SOTA methods on different datasets, as shown in Table 1. The results illustrate that our FCENet obtains the best performance of precision (P) and F-measure (F), and achieves competitive performance of recall (R) on CTW1500 and Total-Text datasets of arbitrary-shaped texts. Note that

(a) TextRay [25] (b) ABCNet [12] (c) Ours (d) GT

Figure 7: Qualitative comparisons with TextRay [25] and ABCNet [12] on selected challenging samples in CTW1500 .

Table 4: Quantitative comparison on the highly-curved text subset of CTW1500. 'Ext.' denotes extra training data.

Methods	Ext.	R(%)	P(%)	F(%)
TextRay [25]	√	71.2	77.0	74.0
ABCNet [12]	√	66.7	79.8	72.6
FCENet	-	**74.7**	**84.3**	**79.2**

most of the previous methods, except ContourNet [29], require extra training data to obtain its best performance, but our FCENet is trained without it. On ICDAR15 dataset of multi-orientated and street-viewed texts, FCENet also achieves competitive results without additional setup.

Moreover, FCENet has simple network architecture, and efficient post-processing (*i.e.*, IFT and NMS), which makes it easy to implement and very practical. Note that even equipping FCENet† with the ResNet50 without DCN as the backbone , which is the same as that of ABCNet [12], it still obtains competitive results in Table 1.

4.6. Evaluation on Highly-curved Subset

Since CTW1500 still contains a large amount of non-curved texts, We selected highly-curved texts from it to build a challenging subset. Comparisons were made among most recent methods with explicit text shape modeling [12, 25], *i.e.*, TextRay [25] and ABCNet [12].

To built the challenging subset, we discard the "simple" non-curved texts but reserve highly-curved texts. We utilize an algorithm to select the subset (total 106 samples), based on the observation that when we remove one of ground truth annotation points except the head and tail, the area of highly-curved text will change greatly. We computed the area of the annotation polygon before A_{bef} and after A_{aft} removing a point in ground truth annotations, and selected samples if $|A_{bef} - A_{aft}|/A_{bef} \geq 0.07$.

Qualitative and quantitative comparisons were shown in Fig. 7 and Table 4, respectively. The results indicate that FCE is complementary to TextRay [25] and ABCNet [12] for explicitly modeling irregular text instances, and also show the effectiveness of FCENet for highly-curved text detection.

5. Conclusion

This paper focuses on the explicit shape modeling for arbitrary-shaped text detection. We propose Fourier contour embedding method, which allows to approximate any closed shapes accurately. Then, we propose FCENet which first predicts Fourier signature vectors of text instances in the Fourier domain, and then reconstructs text contour point sequences in the image spatial domain via the Inverse Fourier Transformation. FCENet can be optimized in an end-to-end manner, and be implemented without any complex post processing. Extensive evaluation were performed for the proposed FCE and FCENet. Experimental results validate the representation capability of FCE, especially on highly-curved texts, and good generalization of FCENet when training with small samples. Moreover, it shows that FCENet achieves the SOTA performance on CTW1500, Total-Text and competitive results on ICDAR2015.

References

[1] Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. Character region awareness for text detection. In *Pro. CVPR*, pages 9365–9374, 2019. 1, 2, 3, 7

[2] Chee Kheng Ch'ng and Chee Seng Chan. Total-text: A comprehensive dataset for scene text detection and recognition. In *Proc. ICDAR*, volume 1, pages 935–942, 2017. 2, 4, 6

[3] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proc. ICCV*, pages 764–773, 2017. 1

[4] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *Proc. ICCV*, pages 2961–2969, 2017. 1

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, June 2016. 4

[6] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu, F. Shafait, S. Uchida, and E. Valveny. ICDAR 2015 competition on robust reading. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1156–1160, 2015. 6

[7] Minghui Liao, Baoguang Shi, and Xiang Bai. Textboxes++: A single-shot oriented scene text detector. *IEEE Transactions on Image Processing*, 27(8):3676–3690, 2018. 1, 3

[8] Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization. In *Proc. AAAI*, pages 11474–11481, 2020. 1, 2, 3, 7

[9] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proc. CVPR*, July 2017. 1, 2, 4, 6

[10] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proc. ICCV*, pages 2980–2988, 2017. 1

[11] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *Proc. CVPR*, pages 8759–8768, 2018. 1

[12] Yuliang Liu, Hao Chen, Chunhua Shen, Tong He, Lianwen Jin, and Liangwei Wang. Abcnet: Real-time scene text spotting with adaptive bezier-curve network. In *Proc. CVPR*, pages 9809–9818, 2020. 1, 2, 3, 6, 7, 8

[13] Yuliang Liu, Lianwen Jin, Shuaitao Zhang, Canjie Luo, and Sheng Zhang. Curved scene text detection via transverse and longitudinal sequence connection. *Pattern Recognition*, 90:337–345, 2019. 2, 4, 6

[14] Shangbang Long, Jiaqiang Ruan, Wenjie Zhang, Xin He, Wenhao Wu, and Cong Yao. Textsnake: A flexible representation for detecting text of arbitrary shapes. In *Proc. ECCV*, 2018. 1, 2, 3, 4, 7

[15] Chixiang Ma, Lei Sun, Zhuoyao Zhong, and Qiang Huo. Relatext: Exploiting visual relationships for arbitrary-shaped scene text detection with graph convolutional networks. *arXiv preprint arXiv:2003.06999*, 2020. 1, 3

[16] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proc. CVPR*, pages 779–788, 2016. 1

[17] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Proc. NIPS*, pages 91–99. Curran Associates, Inc., 2015. 5

[18] A. Rosenfeld and M. Thurston. Edge and curve detection for visual scene analysis. *IEEE Transactions on Computers*, C-20(5):562–569, 1971. 5

[19] Baoguang Shi, Xiang Bai, and Serge Belongie. Detecting oriented text in natural images by linking segments. In *Proc. CVPR*, pages 2550–2558, 2017. 1, 2, 3

[20] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proc. CVPR*, June 2016. 5

[21] Jun Tang, Zhibo Yang, Yongpan Wang, Qi Zheng, Yongchao Xu, and Xiang Bai. Seglink++: Detecting dense and arbitrary-shaped scene text by instance-aware component grouping. *Pattern Recognition*, 96:106954, 2019. 1, 3, 7

[22] Zhi Tian, Weilin Huang, Tong He, Pan He, and Yu Qiao. Detecting text in natural image with connectionist text proposal network. In *Proc. ECCV*, pages 56–72. Springer, 2016. 1, 2, 3

[23] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proc. ICCV*, pages 9627–9636, 2019. 1

[24] Zhuotao Tian, Michelle Shu, Pengyuan Lyu, Ruiyu Li, Chao Zhou, Xiaoyong Shen, and Jiaya Jia. Learning shape-aware embedding for scene text detection. In *Proc. CVPR*, pages 4234–4243, 2019. 1, 2, 3, 7

[25] Fangfang Wang, Yifeng Chen, Fei Wu, and Xi Li. Textray: Contour-based geometric modeling for arbitrary-shaped scene text detection. In *Proc. ACM MM*, pages 111–119, 2020. 1, 2, 3, 6, 7, 8

[26] Hao Wang, Pu Lu, Hui Zhang, Mingkun Yang, Xiang Bai, Yongchao Xu, Mengchao He, Yongpan Wang, and Wenyu Liu. All you need is boundary: Toward arbitrary-shaped text spotting. In *Proc. AAAI*, pages 12160–12167, 2020. 7

[27] Wenhai Wang, Enze Xie, Xiang Li, Wenbo Hou, Tong Lu, Gang Yu, and Shuai Shao. Shape robust text detection with progressive scale expansion network. In *Proc. CVPR*, pages 9336–9345, 2019. 1, 2, 3, 7

[28] Wenhai Wang, Enze Xie, Xiaoge Song, Yuhang Zang, Wenjia Wang, Tong Lu, Gang Yu, and Chunhua Shen. Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In *Proc. ICCV*, pages 8440–8449, 2019. 1, 2, 3, 7

[29] Yuxin Wang, Hongtao Xie, Zheng-Jun Zha, Mengting Xing, Zilong Fu, and Yongdong Zhang. Contournet: Taking a further step toward accurate arbitrary-shaped scene text detection. In *Proc. CVPR*, pages 11753–11762, 2020. 1, 2, 3, 7, 8

[30] Enze Xie, Yuhang Zang, Shuai Shao, Gang Yu, Cong Yao, and Guangyao Li. Scene text detection with supervised pyramid context network. In *Proc. AAAI*, volume 33, pages 9038–9045, 2019. 1, 2, 3

[31] Youjiang Xu, Jiaqi Duan, Zhanghui Kuang, Xiaoyu Yue, Hongbin Sun, Yue Guan, and Wayne Zhang. Geometry normalization networks for accurate scene text detection. In *Proc. ICCV*, pages 9137–9146, 2019. 1

[32] Yongchao Xu, Yukang Wang, Wei Zhou, Yongpan Wang, Zhibo Yang, and Xiang Bai. Textfield: Learning a deep direction field for irregular scene text detection. *IEEE Transactions on Image Processing*, 28(11):5566–5579, 2019. 1, 2, 3

[33] Chuhui Xue, Shijian Lu, and Wei Zhang. MSR: Multi-scale shape regression for scene text detection. *Proc. IJCAI*, 2019. 1, 3

[34] Cong Yao, Xiang Bai, Nong Sang, Xinyu Zhou, Shuchang Zhou, and Zhimin Cao. Scene text detection via holistic, multi-channel prediction. *arXiv preprint arXiv:1606.09002*, 2016. 2, 3

[35] Chengquan Zhang, Borong Liang, Zuming Huang, Mengyi En, Junyu Han, Errui Ding, and Xinghao Ding. Look more than once: An accurate detector for text of arbitrary shapes. In *Proc. CVPR*, pages 10552–10561, 2019. 1, 2, 3, 7

[36] Shi-Xue Zhang, Xiaobin Zhu, Jie-Bo Hou, Chang Liu, Chun Yang, Hongfa Wang, and Xu-Cheng Yin. Deep relational reasoning graph network for arbitrary shape text detection. In *Proc. CVPR*, pages 9699–9708, 2020. 1, 2, 3, 7

[37] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. EAST: An efficient and accurate scene text detector. In *Proc. CVPR*, pages 5551–5560, 2017. 1, 3

[38] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proc. CVPR*, pages 9308–9316, 2019. 2, 4, 6