

# 场景文字处理

华中科技大学

$$\rho := \frac{1 + \sqrt{-3}}{2}$$

2020-04-3





# 目 录

<b>第一章 文字识别 (Scene Text Recognition)</b>	<b>1</b>
1.1 文字识别方法介绍 . . . . .	1
1.1.1 TextScanner . . . . .	1
1.1.2 SCATTER . . . . .	1
1.1.3 SRN . . . . .	1
<b>第二章 文字检测 (Scene Text Detection)</b>	<b>3</b>
2.1 文字检测是什么? . . . . .	3
2.2 基于回归的方法 . . . . .	3
2.3 基于分割的方法 . . . . .	3
2.4 其他方法 . . . . .	3
<b>第三章 端到端文字识别 (Scene Text Spotting)</b>	<b>4</b>
3.1 任意形状文本端到端识别方法的发展脉络 . . . . .	5
3.2 各种任意形状文本端到端识别方法介绍 . . . . .	5
3.2.1 MaskTextSpotter . . . . .	5
3.2.2 TextDragon . . . . .	7
3.2.3 CharNet . . . . .	8
3.2.4 MaskRoI . . . . .	10
3.2.5 Boundary . . . . .	11
3.2.6 TextPerceptron . . . . .	11
3.2.7 ABCNet . . . . .	13
3.3 任意形状文本端到端识别方法的总结 . . . . .	14
<b>Bibliography</b>	<b>15</b>

# 1

## 文字识别（Scene Text Recognition）

[6, 1, 15]

### 1.1 文字识别方法介绍

#### 1.1.1 TextScanner

vhefbhv

#### 1.1.2 SCATTER

#### 1.1.3 SRN

SRN 的主要出发点是：1) 在文字识别中，由于光照，旋转等因素的影响，仅仅依靠图像的视觉特征极易引起单词中某个字符预测错误。对于这些易错的字符，如果能够利用单词的语义信息，那么将会极大地降低该字符的预测错误率。如图1-1所示，如果仅仅观察每个字符的视觉特征 (b)，某些字符容易预测错误，结合上下文语义信息能够缓解因视觉特征混淆而引起的错误预测。2) 文字识别中基于注意力机制的识别器中 [13]，注意力模块的输出大多是串行的，注意力模块中当前时刻的预测非常依赖于前一时刻的输出，导致模型难以并行处理。为解决模型的效率问题，提出并行的注意力模块。

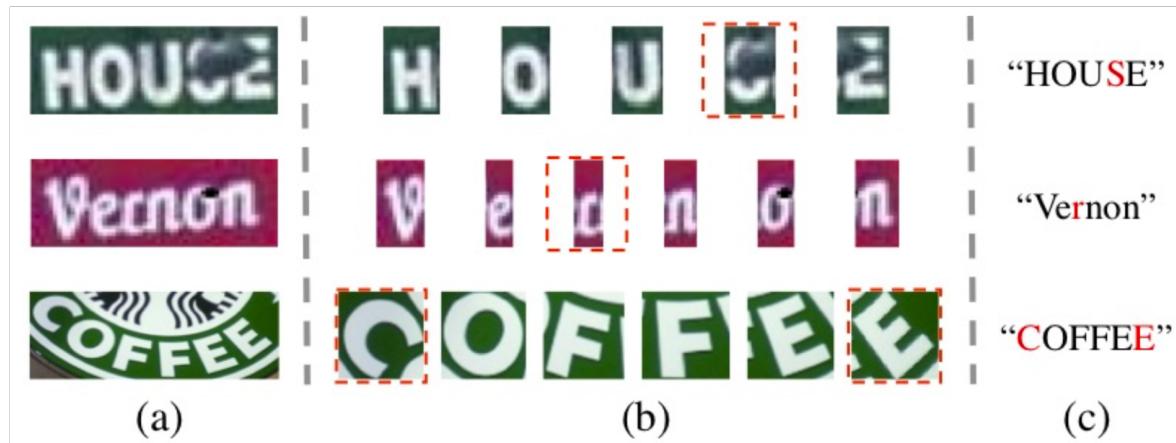


图 1-1 单词中容易预测错误的字符案例：(a) 表示原图；(b) 表示字符，红色标注为易错字符；(c) 为利用上下文语义信息预测结果。

### 1.1.3.1 SRN 的网络结构

SRN 的网络结构如图1-2所示：1) 整体网络的 backbone 为 FPN 网络，用于提取图像的视觉特征；2) 并行的视觉注意力模块 (PAVM) 用于定位每个字符；3) 全局的语义推理模块 (GSRM) 用于利用语义上下文来预测字符。

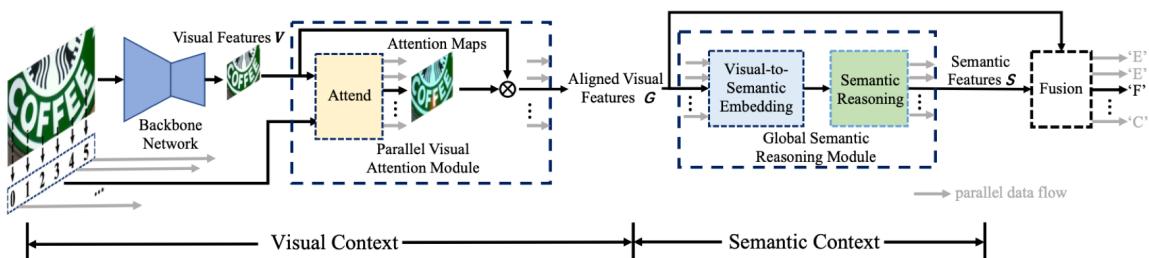


图 1-2 SRN 网络框架图。

# 2

## 文字检测 (Scene Text Detection)

2.1 文字检测是什么？

2.2 基于回归的方法

2.3 基于分割的方法

2.4 其他方法

# 3

## 端到端文字识别 (Scene Text Spotting)

2018 年以前，关于 Scene Text Spotting 的论文 [7]，主要集中在解决旋转文本的端到端识别问题，几乎没有论文解决曲形文本端到端识别的问题。自从发表在 ECCV2018 中的论文，MaskTextSpotter[10]，开始试图解决曲形文本端到端识别的问题以来，大量的工作开始致力于该问题的研究，如 [10, 5, 2, 17, 12, 16, 8, 11]。自此，端到端文本识别方法在理论上能够解决任意形状文本识别的问题。在以下表述中以“任意形状文本端到端识别方法”来统称既能处理多方向又能处理曲形文本的端到端文本识别方法。

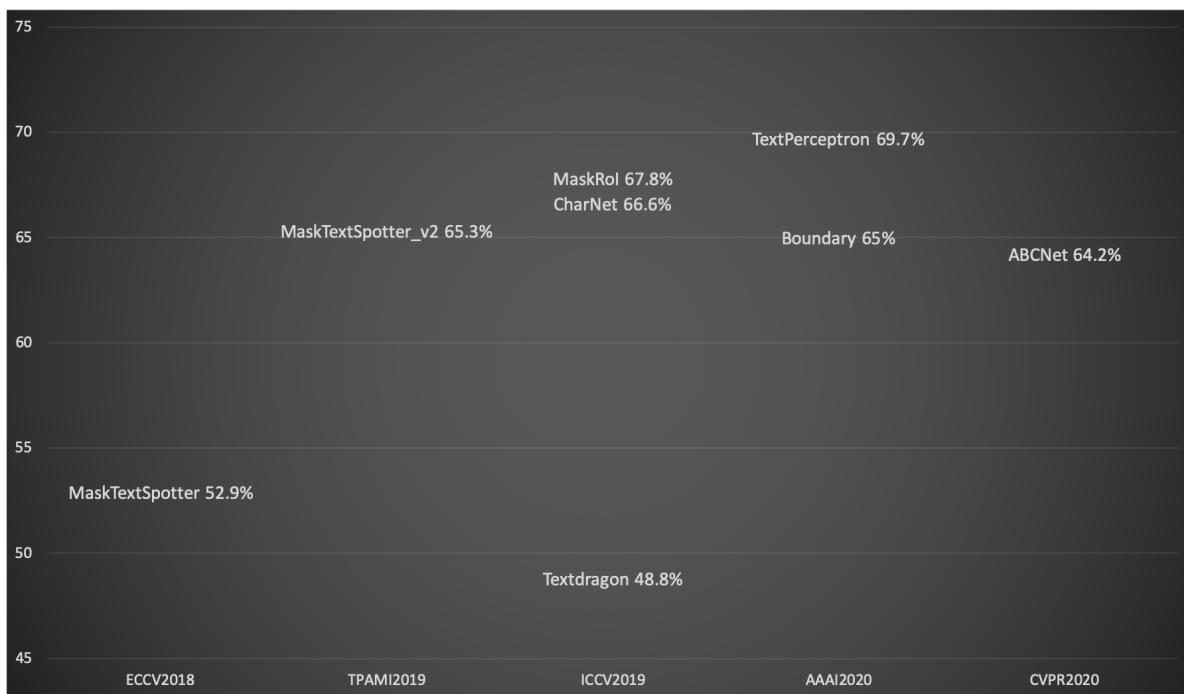


图 3-1 2018 年到 2020 年 3 月期间，曲形文本端到端文本识别方法在 TotalText 上的性能。

首先，我们从问题的角度出发，概述各种任意形状文本端到端识别方法之间的关

系。然后将从方法，实验结果，该方法的优缺点等方面来分别探讨各种方法，这些方法包括：MaskTextSpotter[10, 5]、TextDragon[2]、CharNet[17]、MaskRoI[12]、Boundary[16]、TextPerceptron[11]以及ABCNet[8]。最后，进一步总结归纳以上方法的特点。

### 3.1 任意形状文本端到端识别方法的发展脉络

### 3.2 各种任意形状文本端到端识别方法介绍

#### 3.2.1 MaskTextSpotter

##### 3.2.1.1 MaskTextSpotter 网络结构

MaskTextSpotter 作为首个任意形状文本端到端识别方法，其思路是将每个文字字符作为一个类别进行检测。其网络框架如图3-2所示，网络主体部分与 MaskRCNN[4]一致。由于常规的 MaskRCNN 网络（分割分支进行 1 通道的分割，表示是否为文字两个类别）只能完成文字检测任务，无法进行文字识别，因此作者将 Mask 分支设计为 37 个通道（26 个英文字母加上 10 个数字以及表示是否为字符区域的 1 通道）加上检测的 1 个类别共 38 个类别进行分割，其分割分支如图3-3所示。在测试阶段，检测的 1 通道能够检测任意形状的文本。根据另外 37 个通道的分割信息，按照从左到右的顺序连接每个字符，完成识别任务。

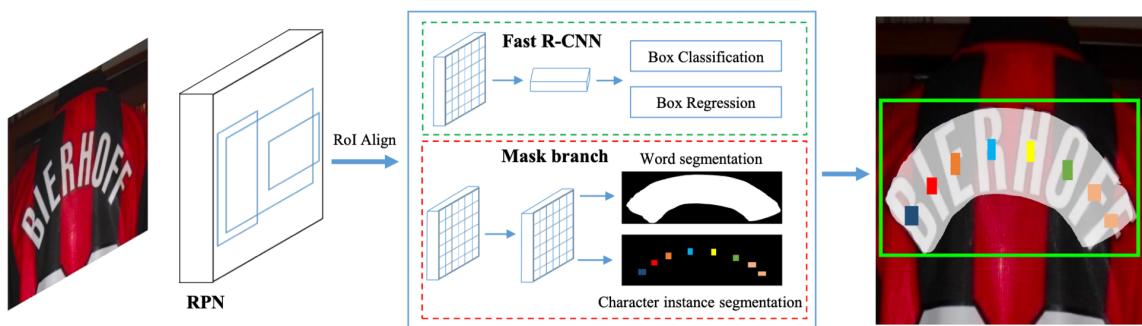


图 3-2 MaskTextSpotter 框架图。

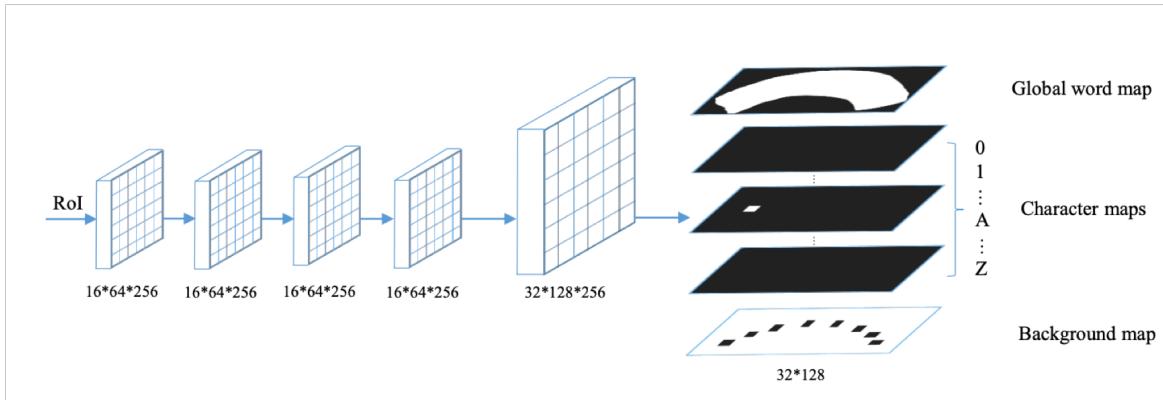


图 3–3 MaskTextSpotter 的分割分支结构图。

### 3.2.1.2 MaskTextSpotter 实验细节

MaskTextSpotter 的训练过程分为两部分：合成数据集预训练和真实数据集微调阶段。合成数据集使用的是 SynthText[3]，batch size 为 8，输入图像以短边为 800，保持长宽比进行训练。在真实数据微调阶段，batch size 为 8，采用多尺度训练的策略，短边分为 (600, 800, 1000) 三个尺度进行训练，使用的数据集有 SynthText, ICDAR2013, ICDAR2015, TotalText 以及来自 [19] 的 1162 张图像。

### 3.2.1.3 MaskTextSpotter\_v2 网络结构

由于 MaskTextSpotter 中将通过将每个字符作为单独的类别分割出来作为识别结果，这样会导致识别过程中忽略文字的语义特征。基于此，MaskTextSpotter\_v2 的主要出发点是将文字的语义信息融入到识别分支中，最终其网络结构如图3–4所示。可以看出，该网络结构和 MaskTextspotter 相比，在识别分支加入了序列识别分支。其识别分支具体结构如图3–5所示。

识别分支由两部分组成：基于字符分割的识别分支和基于序列识别的识别分支。每个识别分支在输出识别结果的同时对识别结果的置信度进行打分，最终的识别结果取置信度较高的分支的识别结果。

### 3.2.1.4 MaskTextSpotter\_v2 实验细节

MaskTextSpotter 的训练过程分为两部分：合成数据集预训练和真实数据集微调阶段。合成数据集使用的是 SynthText[3]，batch size 为 8，输入图像以短边为 800，保持长宽比，学习率从 0.001 开始，第 100k, 200k 次下降 0.1，训练 270k 步。在真实数据微调阶段，batch size 为 8，采用多尺度训练的策略，短边分为 (600, 800, 1000, 1200, 1400)

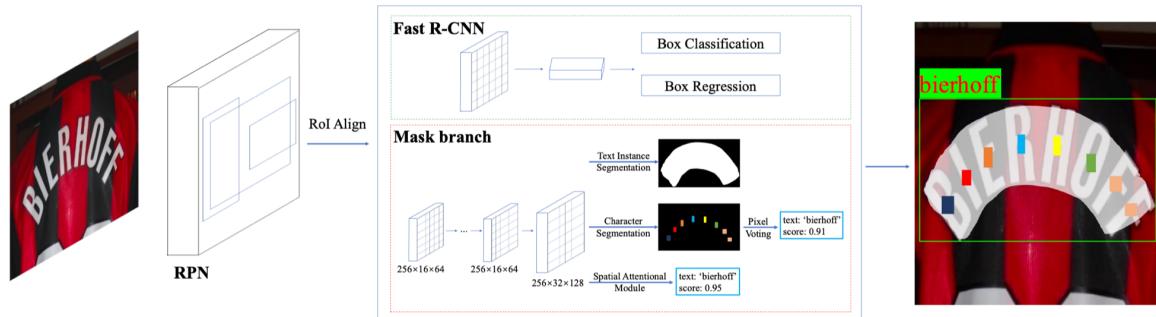


图 3-4 MaskTextSpotter\_v2 框架图。

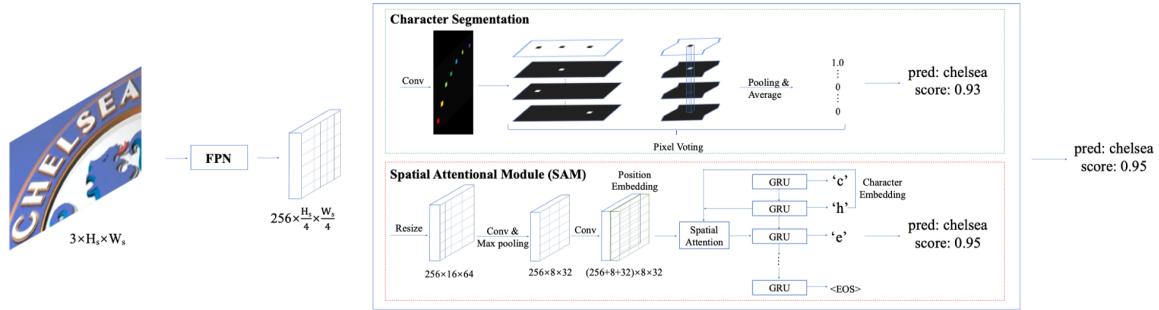


图 3-5 MaskTextSpotter\_v2 识别分支结构图。

三个尺度进行训练，使用的数据集有 SynthText, ICDAR2013, ICDAR2015, TotalText 以及来自 [19] 的 1162 张图像。学习率从 0.001 开始，第 100k 下降 0.1，训练 150k 步。

### 3.2.2 TextDragon

虽然 MaskTextSpotter 能够进行任意形状文本端到端的识别，但是数据集字符级别的标注的要求使得网络的训练比较昂贵。TextDragon 意在只使用单词级别的标注来设计任意形状文本端到端识别系统。

#### 3.2.2.1 TextDragon 网络结构

TextDragon 对任意形状文本的表示方式主要来自于文字检测方法 TextSnake[9]，也就是将任意形状的文本表示为一系列的带方向正方形。然后从属于同一文本实例的带方向正方形中聚合，采样一个子集形成文本区域。基于该子集的带方向正方形，则有：1) 从这些带方向正方形中提取文字边界点来表示检测结果，2) 对每个正方形区域的特征进行字符分类，通过 CTC 解码成文本字符串。TextDragon 的网络结构如图3-6所示。

具体地，文字的表示方法如图3-7所示。文本实例由文本的中心线、正方形边长以及正方形旋转方向构成。在测试阶段，推理过程如下：1) 通过文本中心线来获取每个文本实例大致区域；2) 在每个文本实例的中心线上获得所有的预测的带方向正方形，并保留这些 IOU 大于 0.5，旋转角度差小于 45 度的正方形区域；3) 根据其位置将这些保留的正方形进行排序，形成表示该文本区域的正方形子集。4) 最终通过 RoISlide 从这些矫正后的文本区域中获取特征，进行字符串识别，而文本边界由正方形子集形成的边界点构成。

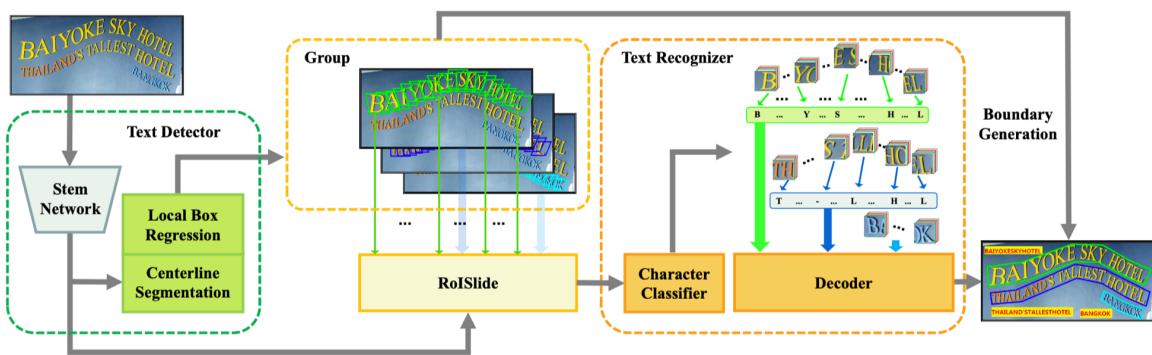


图 3-6 TextDragon 框架图。

### 3.2.2.2 TextDragon 实验细节

TextDragon 的训练过程分为两部分：合成数据集预训练和真实数据集微调阶段。合成数据集使用的是 SynthText，输入图像大小为 512\*512，学习率为 0.01，训练 600k 步。在真实数据微调阶段，输入图像大小为 512\*512，数据集为相对应数据集的训练集，学习率为 0.001，训练 120k 步。

### 3.2.3 CharNet

CharNet 与 TextDragon 一样是在任意形状文本端到端识别算法只有 MaskTextSpotter 的背景下出现的方法。他的主要出发点是：当前端到端识别的方法中，都是 two stage 的（这里 two stage 是指检测网络得到检测结果，再根据检测结果利用 ROI 操作获取特征进行识别），作者认为 two stage 中 ROI 提取很难提取准确（主要是检测存在误差），并且 two stage 过程繁琐，不便使用。因此，作者想设计一个 single stage 的网络，同时输出文本实例的检测和识别结果。

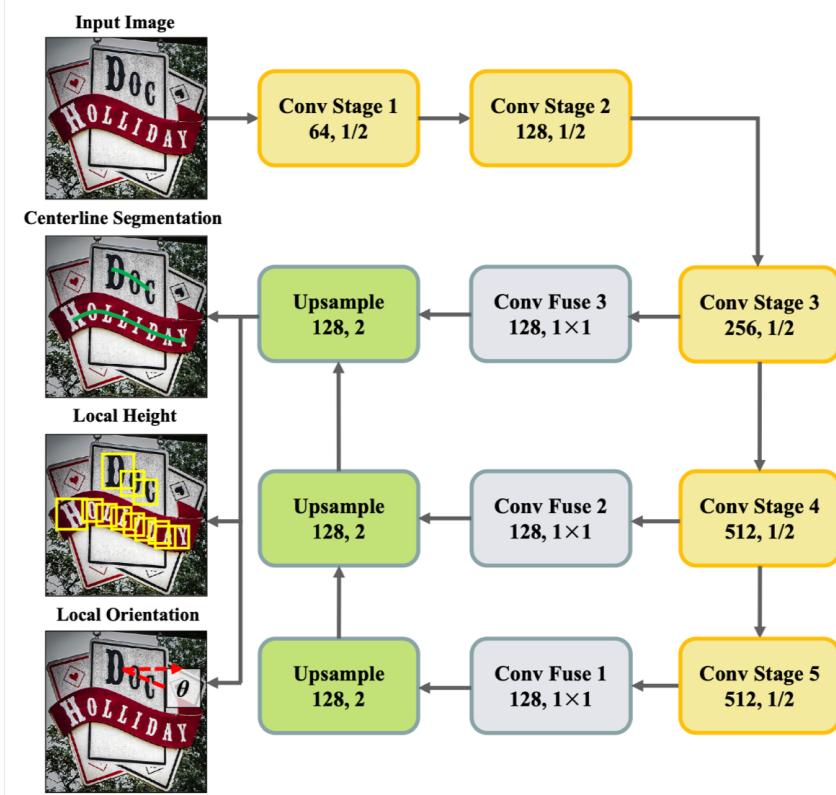


图 3-7 TextDragon 文本实例表示方法。

### 3.2.3.1 CharNet 网络结构

CharNet 在核心问题上和 MaskTextSpotter 一致，都是通过字符级别的分割解决任意形状文本的识别问题。MaskTextSpotter 是在 ROI 内进行分割，而 CharNet 是在全图进行字符级别的分割。那么，CharNet 就剩下最后一个需要解决的问题：如何将分割出的字符 group 成为一个文本区域？如 CharNet 网络结构图3-8所示，其 Detection Branch 的作用便是预测一些信息来将分割出的字符聚合成字符串。对于多方向文本，其 Detection Branch 的表示方法为 EAST[20] 的表示方式，利用预测的四边形框的 IoU 聚合每个字符为字符串。对于曲形文本，其 Detection Branch 的表示方法为 TextField[18] 的表示方法。

### 3.2.3.2 CharNet 实验细节

CharNet 的训练过程分为两部分：合成数据集预训练和真实数据集微调阶段。合成数据集使用的是 SynthText，batch size 为 32，学习率为 0.0002，数据集迭代 5 epochs。在真实数据微调阶段，数据集为相对应数据集的训练集，学习率为 0.002，分三步进行迭代训练，三步训练回合分别为 100, 400, 800 epochs。这里每步迭代训练是指：利用先

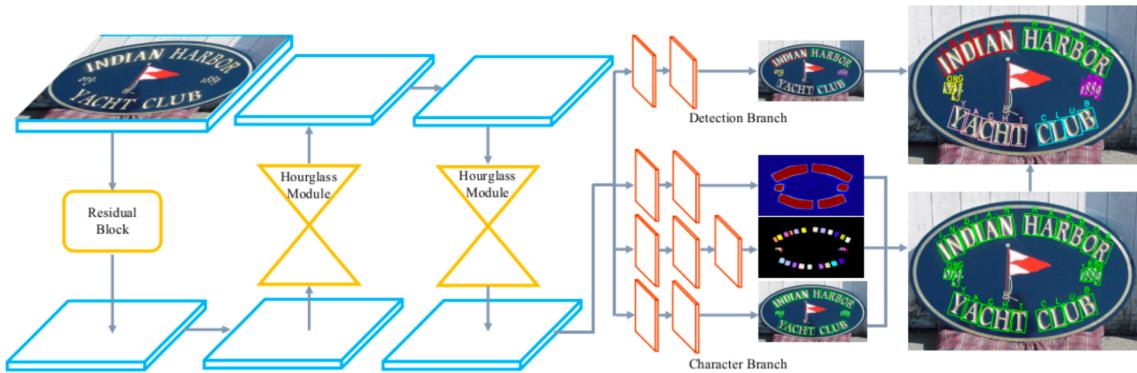


图 3-8 CharNet 框架图。

前的模型获得训练集的字符级别标注（检测框），过滤出正确的字符级别标注来训练模型，迭代 n (100, 400 或 800) 个 epochs。

### 3.2.4 MaskRoI

MaskRoI 与 CharNet 以及 TextDragon 是同时期文章，该方法不需要字符级别的标注，同时不需要将任意形状文本矫正为水平文本进行识别。

#### 3.2.4.1 MaskRoI 网络结构

如图3-9所示，MaskRoI 和 MaskTextSpotter 一样，也是基于 MaskRCNN 框架进行改进的。识别分支采用基于 attention 的序列识别方案。为了解决任意形状文本的 RoI 容易采样到背景或者相邻文本特征的问题，在进行序列识别之前，进行了特征过滤操作。该操作就是将文本实例分割图和 RoI 的特征进行相乘。

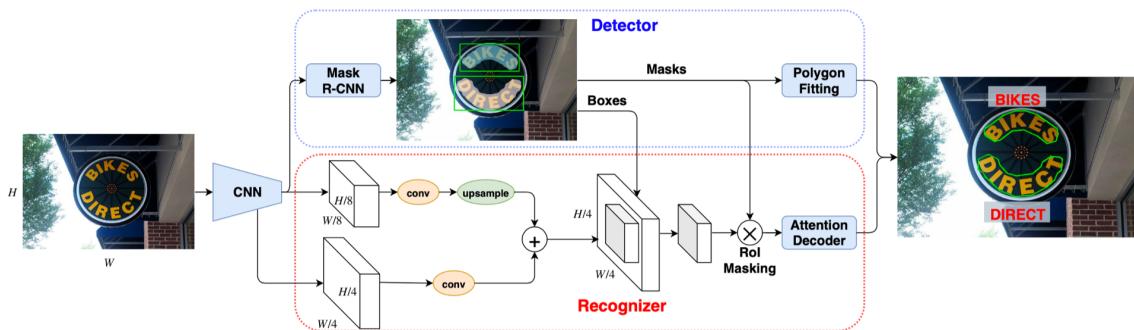


图 3-9 MaskRoI 框架图。

### 3.2.4.2 MaskRoI 实验细节

MaskRoI 采用一步训练的方式，数据集包括 SynthText, ICDAR2015, COCOText, ICDAR-MLT, TotalText 以及网络收集的通过 Google OCR API 标注的 30k 张图像。采用多尺度训练的策略，短边为 480 到 800 之间。

### 3.2.5 Boundary

### 3.2.6 TextPerceptron

TextPerceptron 的主体思路也是文本实例边界点检测 + TPS+ 序列识别。和 Boundary 不同之处在于文本边检点检测过程，具体地说，是通过文本实例的几何属性和后处理得到边界点。

#### 3.2.6.1 TextPerceptron 网络结构

TextPerceptron 的网络结构如图3-10所示。边界点的检测是基于分割的方法，预测的文本实例的几何属性包括：1) 文本上下边界；2) 文本实例的开端；3) 文本实例的结尾；4) 文本实例的中间区域；5) 开端以及结尾处的角点回归；6) 文本中心区域的边界点回归。其中 5) 和 6) 的标签定义如图3-11所示。

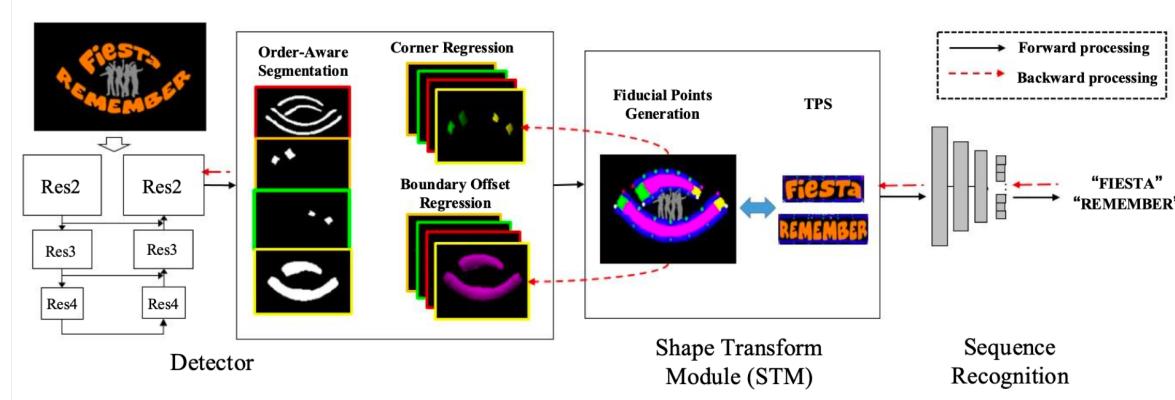


图 3-10 TextPerceptron 框架图。

#### 3.2.6.2 TextPerceptron 边界点获取过程

根据预测的文本实例的几何属性，后处理得到边界点的过程如下：1) 根据中心区域和开端以及结尾的匹配程度可以获得开端、结尾匹配对，上下边界可以用于区分相邻的文本实例；2) 在开端、结尾分割图处获取文本实例的 4 个角点；3) 如图3-12所示，

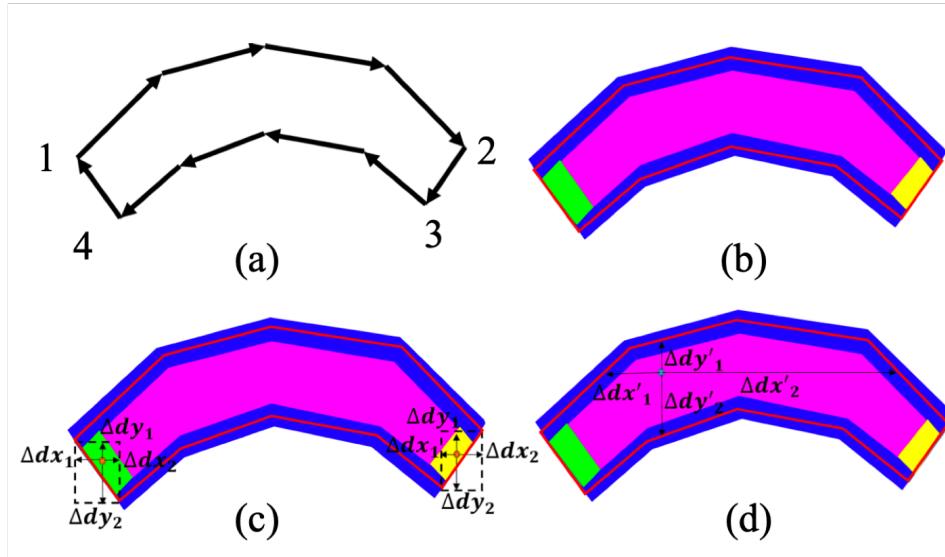


图 3-11 TextPerceptron 角点和边界点回归的定义。

获得较长边的角点对的中心点，作垂线，获得该点对所处边界的交点作为一个边界点，以此类推，获得所有边界点。

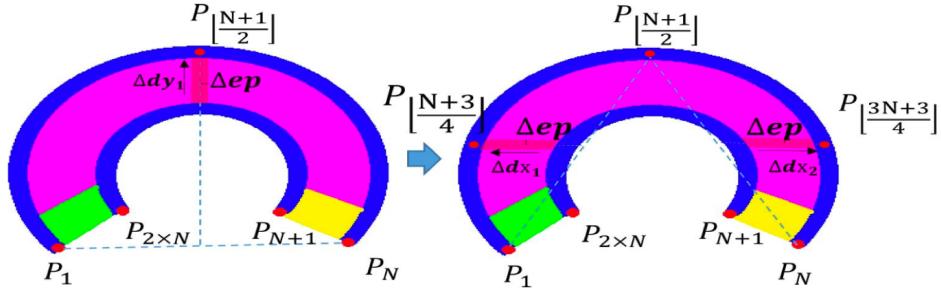


图 3-12 TextPerceptron 后处理过程。

### 3.2.6.3 TextPerceptron 实验细节

TextPerceptron 分为 3 个阶段，1) 训练检测分支，在 SynthText 上以学习率 0.002 训练 5 epochs；2) 训练识别分支，在 SynthText 上以学习率 0.002 训练 5 epochs；3) 检测识别联合训练，在各自训练集上以学习率 0.001 训练 80 epochs，每 20 epochs 学习率乘以 0.1。

### 3.2.7 ABCNet

ABCNet 的主体思路也是文本实例边界点检测 +TPS+ 序列识别。边界点的检测采样回归的方法，和 Boundary 不同的是，检测部分采用 anchor-free 的方法。论文的框架主要基于 FCOS[14] 上进行改进。

#### 3.2.7.1 ABCNet 网络结构

ABCNet 采用贝塞尔曲线来表示文本实例的边界，贝塞尔曲线描述效果如图3-14所示。网络框架如图3-13所示，FCOS 采用密集预测的方式。从代码中可以看出，检测部分的预测信息包括：1) 每个 bbox 的得分；2) 中心区域预测；3) bbox 回归；4) 贝塞尔曲线控制点预测。获得贝塞尔曲线后，

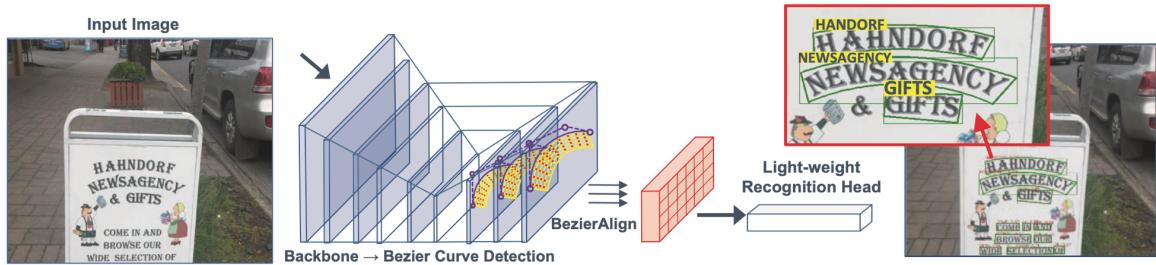


图 3-13 ABCNet 框架图。

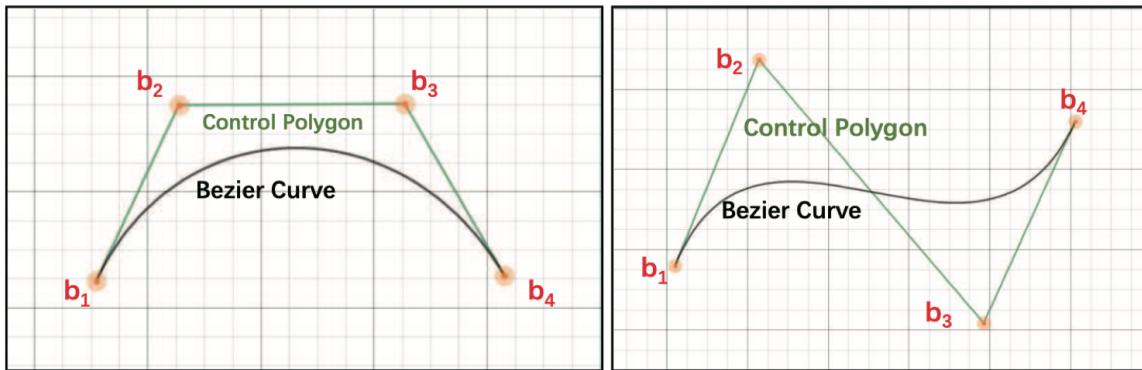


图 3-14 贝塞尔曲线。

贝塞尔曲线获得的边界如图3-15所示。

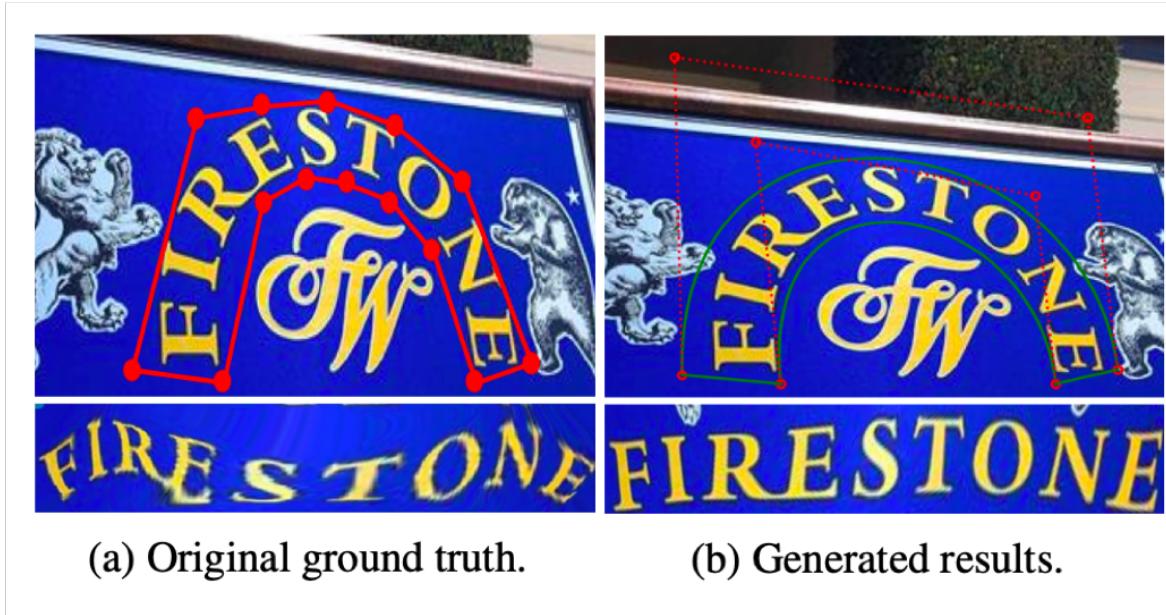


图 3-15 贝塞尔曲线。

### 3.2.7.2 ABCNet 实验细节

TextPerceptron 分为 2 个阶段：1) 合成的 150k 合成数据集，15k 的 COCOText，7k 的 ICDAR-MLT 预训练；2) 相应的训练集训练。

## 3.3 任意形状文本端到端识别方法的总结

# Bibliography

- [1] Yu Deli et al. “Towards Accurate Scene Text Recognition with Semantic Reasoning Networks”. In: (2020).
- [2] Wei Feng et al. “TextDragon: An End-to-End Framework for Arbitrary Shaped Text Spotting”. In: *The IEEE International Conference on Computer Vision (ICCV)*. Oct. 2019.
- [3] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. “Synthetic data for text localisation in natural images”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2315–2324.
- [4] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. “Mask r-cnn”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2961–2969.
- [5] Minghui Liao et al. “Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes”. In: *IEEE transactions on pattern analysis and machine intelligence* (2019).
- [6] Ron Litman et al. “SCATTER: Selective Context Attentional Scene Text Recognizer”. In: (Mar. 2020).
- [7] Xuebo Liu et al. “Fots: Fast oriented text spotting with a unified network”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 5676–5685.
- [8] Yuliang Liu et al. “ABCNet: Real-time Scene Text Spotting with Adaptive Bezier-Curve Network”. In: *arXiv preprint arXiv:2002.10200* (2020).
- [9] Shangbang Long et al. “Textsnake: A flexible representation for detecting text of arbitrary shapes”. In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 20–36.
- [10] Pengyuan Lyu et al. “Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 67–83.

- [11] Liang Qiao et al. “Text Perceptron: Towards End-to-End Arbitrary-Shaped Text Spotting”. In: *arXiv* (2020), arXiv–2002.
- [12] Siyang Qin et al. “Towards unconstrained end-to-end text spotting”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2019, pp. 4704–4714.
- [13] Baoguang Shi et al. “Aster: An attentional scene text recognizer with flexible rectification”. In: *IEEE transactions on pattern analysis and machine intelligence* 41.9 (2018), pp. 2035–2048.
- [14] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. “Fcos: Fully convolutional one-stage object detection”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2019, pp. 9627–9636.
- [15] Zhaoyi Wan et al. “TextScanner: Reading Characters in Order for Robust Scene Text Recognition”. In: *arXiv preprint arXiv:1912.12422* (2019).
- [16] Hao Wang et al. “All You Need Is Boundary: Toward Arbitrary-Shaped Text Spotting”. In: *arXiv preprint arXiv:1911.09550* (2019).
- [17] Linjie Xing, Zhi Tian, Weilin Huang, and Matthew R. Scott. “Convolutional Character Networks”. In: *The IEEE International Conference on Computer Vision (ICCV)*. Oct. 2019.
- [18] Yongchao Xu et al. “TextField: learning a deep direction field for irregular scene text detection”. In: *IEEE Transactions on Image Processing* 28.11 (2019), pp. 5566–5579.
- [19] Zhuoyao Zhong, Lianwen Jin, Shuye Zhang, and Ziyong Feng. “Deeptext: A unified framework for text proposal generation and text detection in natural images”. In: *arXiv preprint arXiv:1605.07314* (2016).
- [20] Xinyu Zhou et al. “EAST: an efficient and accurate scene text detector”. In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2017, pp. 5551–5560.