计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2020,29(12):257−262 [doi: 10.15888/j.cnki.csa.007707]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于分割的任意形状场景文本检测①

蔡鑫鑫, 王　敏

(河海大学 计算机与信息学院, 南京 211100)
通讯作者: 蔡鑫鑫, E-mail: 2360866893@qq.com

摘　要: 随着深度学习技术的发展, 自然场景文本检测的性能获得了显著的提升. 但目前仍然存在两个主要的挑战: 一是速度和准确度之间的权衡, 二是对任意形状的文本实例的检测. 本文采用基于分割的方法高效准确的检测任意形状场景文本. 具体来说, 使用具有低计算成本的分割头和简洁高效的后处理, 分割头由特征金字塔增强模块和特征融合模块组成, 前者可以引入多层次的信息来指导更好的分割, 后者可以将前者给出的不同深度的特征集合成最终的特征进行分割. 本文采用可微二值化模块, 自适应地设置二值化阈值, 将分割方法产生的概率图转换为文本区域, 从而提高文本检测的性能. 在标准数据集 ICDAR2015 和 Total-Text 上, 本文提出的方法使用轻量级主干网络如 ResNet18 在速度和准确度方面都达到了可比较的结果.

关键词: 自然场景文本检测; 分割; 特征金字塔增强模块; 特征融合模块; 可微二值化模块

引用格式: 蔡鑫鑫,王敏.基于分割的任意形状场景文本检测.计算机系统应用,2020,29(12):257–262. http://www.c-s-a.org.cn/1003-3254/7707.html

Arbitrary Shape Scene Text Detection Based on Segmentation

CAI Xin-Xin, WANG Min

(College of Computer and Information, Hohai University, Nanjing 211100, China)

Abstract: With the development of deep learning technology, the performance of natural scene text detection has been significantly improved. Nonetheless, two main challenges still exist: the first problem is the trade-off between speed and accuracy, and the second one is to model the arbitrary-shaped text instance. In this study, we propose a segmentation-based method to tackle arbitrary-shaped text detection efficiently and accurately. Specifically, we use a low computational-cost segmentation head and efficient post-processing. The segmentation head is made up of Feature Pyramid Enhancement Module (FPEM) and Feature Fusion Module (FFM). FPEM can introduce multi-level information to guide the better segmentation. FFM can integrate the features given by the FPEMs of different depths into a final feature for segmentation. We use a Differentiable Binarization (DB) module, which can perform the binarization process in a segmentation network. Optimized along with a DB module, a segmentation network can adaptively set the thresholds for binarization, which not only simplifies the post-processing but also enhances the performance of text detection. On the standard datasets ICDAR2015 and Total-Text, the method proposed in this study uses a lightweight backbone network such as ResNet18 to achieve comparable results in terms of speed and accuracy.

Key words: natural scene text detection; segmentation; feature pyramid enhancement module; feature fusion module; differentiable binarization module

① 收稿时间: 2020-05-01; 修改时间: 2020-05-27; 采用时间: 2020-06-05; csa 在线出版时间: 2020-11-30

1 引言

文本是传递语义信息的最基本媒介, 它在日常生活中无处不在: 路牌、商店招牌、产品包装、餐馆菜单等, 这种自然环境中的文本被称为场景文本[1]. 自然场景中的文本检测及识别技术可以广泛地应用到场景分析与理解、视觉障碍导航、智能交通管理、无人驾驶等领域. 由于场景文本具有不同的尺度和形状, 包括水平文本、多方向文本和曲线文本, 因此对每个文本实例的边界框或区域进行定位仍然是一项具有挑战性的任务. 基于分割的场景文本检测方法能够预测像素级的结果来描述各种形状的文本, 近年来受到了广泛的关注. 然而, 大多数基于分割的方法推理速度慢, 模型和后处理步骤复杂, 限制了它们在真实环境中的部署. 同时已有高效的文本检测器多数是针对四边形文本实例设计的, 在检测曲线文本时存在不足.

为了解决这些问题, 本文提出了一种可以检测任意形状文本的方法, 同时可以在速度和准确度之间达到很好的平衡. 本文方法主要有 3 个步骤: (1) 使用轻量级的分割网络提取特征, 预测文本的概率图和阈值图; (2) 根据可微二值化模块将概率图和阈值图结合得到近似的二值图, 自适应地预测图像中每个位置的阈值, 从而很好的区分前景和背景中的像素; (3) 对近似二值图进行简单后处理, 得到文本区域.

2 相关工作

目前基于深度学习的自然场景文本检测方法可以分为 3 类: 基于区域建议的方法、基于分割的方法和混合方法[2].

基于区域建议方法的主要思想是先对自然场景文本图像提取候选框, 然后对每个区域进行分类和回归, 最后得到文本检测结果. Textboxes[3] 方法可以快速地计算文本在每个区域存在的可能性, 将常用的卷积核修改成1×5, 使其更适合自然场景文本检测. Textboxes++ 方法[4] 将 Textboxes 水平排列文本检测器扩展为任意方向排列文本检测器. 基于区域建议的方法通常使用简单的后处理算法 (如非极大值抑制算法 NMS), 但是大多数方法不能精确的表示不规则文本 (如曲线文本) 的边界框.

基于分割的方法主要借鉴了全卷积神经网络 (FCN) 的思想, 针对图像中每一个像素点进行分类判断, 以达到语义级别分割的目的. EAST 方法[5] 首先通过 FCN 输出文本区域像素级检测结果, 然后将上述结果通过 NMS 算法获得文本区域. TextSnake 方法[6] 将文本实例描述为一个以对称轴为中心的重叠圆盘序列, 每个圆盘都与潜在的可变半径和方向相关, 这种几何属性是通过 FCN 模型来估计的. PSENet[7] 为每个文本实例生成不同比例的内核, 并逐渐将最小比例内核扩展为具有完整形状的文本实例. 基于分割的方法通常需要复杂的后处理, 会降低推理速度.

混合方法是将上述的两种方法相结合来进行场景文本检测. LOMO 方法[8] 可以多次逐步定位文本, 通过迭代细化逐步感知整个长文本, 考虑文本实例的几何特性对不规则文本进行精确再现. FTSN 模型[9] 从实例感知语义分割的角度, 利用语义分割任务和基于区域建议的目标检测任务的优点, 对文本实例进行联合检测和分割.

本文方法侧重于在不损失推理速度的情况下, 将二值化过程包含到训练周期中来改进分割结果.

3 基于分割的任意形状场景文本检测

图 1 为本文提出方法的总体框架. 首先, 为了提高效率, 本文采用一种计算效率高的分割头来细化特征. 分割头包括两个关键模块: 特征金字塔增强模块 (FPEM) 和特征融合模块 (FFM)[10]. 如图 1 所示, FPEM 是可级联的, 可以附加在主干之后, 使不同尺度的特征更深入和更具表现力, 然后利用特征融合模块 (FFM) 将不同深度的 FPEM 产生的特征融合得到最终的分割特征 F. 其次, 利用特征 F 对概率图 (P) 和阈值图 (T) 进行预测, 根据可微二值化模块 (DB) 将概率图和阈值图结合得到近似的二值图 (B'), 自适应地预测图像中每个位置的阈值[11]. 最后, 在推理阶段, 通过边界框形成从近似二值图中获得文本区域的边界框.

3.1 特征金字塔增强模块 FPEM

FPEM 是一个 U 型模块, 如图 2 所示, 它由两个阶段组成: Up-Scale 增强和 Down-Scale 增强. Up-Scale 增强作用于输入的特征, 它以步长 32, 16, 8, 4 像素在特征图上迭代增强. 在 Down-Scale 阶段, 输入的是由 Up-Scale 增强生成的特征, 增强的步长从 4 到 32, 同时该阶段输出的特征就是最终 FPEM 的输出. 本文使用分离卷积代替常规卷积来构建 FPEM 的连接部分 (见图 2 虚线部分), 因此 FPEM 能够以较小的计算开销扩大感受野和加深网络.

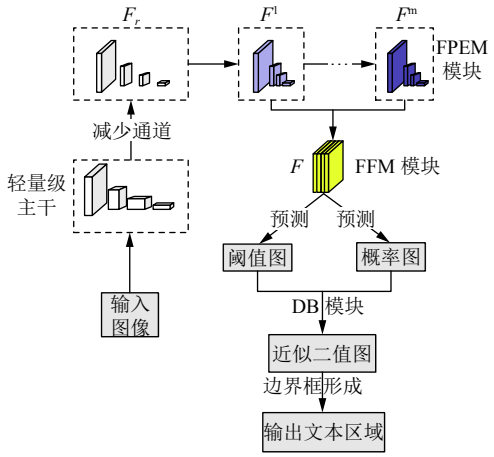

图 1 总体框架

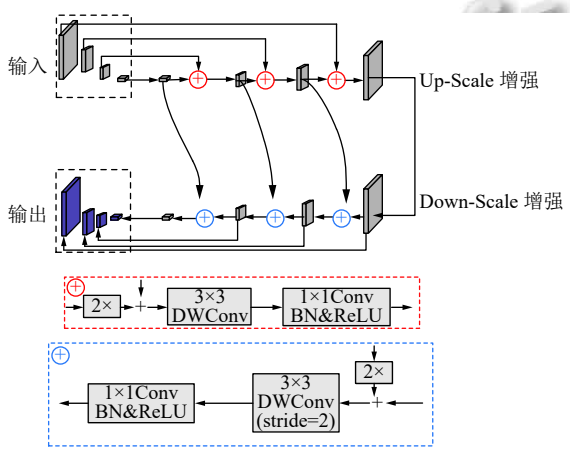

图 2 FPEM 模块

与 FPN 类似, FPEM 能够通过融合低层和高层信息来增强不同尺度的特征, 同时它还有两个优点: (1) FPEM 是一个级联模块, 随着级联数目 m 的增加, 不同尺度的特征图融合更充分, 特征的感受野变得更大; (2) FPEM 的计算开销很小, 它建立在分离卷积的基础上, 计算量大约为 FPN 的 1/5.

3.2 特征融合模块 FFM

采用特征融合模块 FFM 对不同深度的特征 $F^1,F^2,\cdots,F^m$ 进行融合, 对于语义分割来说, 低层和高层的语义信息都很重要, 组合这些特征直接有效的方法是对它们进行上采样和级联. 然而, 此方法给出的融合特征图具有较大的通道数 (4×128×m), 这会降低最终的预测速度. 因此, 本文采用另一种融合方法, 如图 3 所示, 首先通过逐元素增加的方法组合相应的尺度特征图, 然后对添加后的特征图进行上采样并连接成仅具有 4×8 个通道的最终特征图.

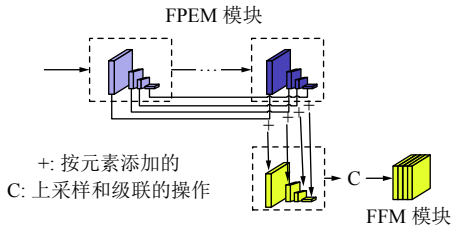

图 3 FFM 模块

3.3 可微二值化模块 DB

根据分割网络生成的概率图 $P\in R^{H\times W}$, 其中 H 和 W 表示图的高度和宽度, 需要将其转换为二值图 $P\in R^{H\times W}$, 其中值为 1 的像素被认为是有效的文本区域. 通常, 这种二值化过程可以描述如下:

$$B_{i,j}=\begin{cases}1, & P_{i,j}\geq t\\ 0, & \text{其他}\end{cases} \tag{1}$$

其中, t 为预定义的阈值, (i,j)表示图中的坐标点.

式 (1) 中描述的标准二值化是不可微的, 在训练过程中不能随着分割网络进行优化. 为了解决这一问题, 本文使用步长函数进行二值化:

$$B'_{i,j}=\frac{1}{1+e^{-k(P_{i,j}-T_{i,j})}} \tag{2}$$

其中, B'是近似二值图, T 是从网络中学习的自适应阈值图, k 是放大系数, 本文设置为 k=50. 该近似二值化函数与标准二值化函数相似, 但具有可微性, 因此可以在训练期间随分割网络进行优化. 自适应阈值的可微二值化方法不仅可以区分文本区域和背景, 而且可以分离出连接紧密的文本实例.

DB 提高性能的原因可以用梯度的反向传播来解释, 以二元交叉熵损失为例, 定义$f(x)=1/(1+e^{-kx})$作为本文的 DB 函数, 其中$x=P_{i,j}-T_{i,j}$, 那么正标签的损失l_+和负标签的损失l_-为:

$$l_+=-\log\frac{1}{1+e^{-kx}} \tag{3}$$

$$l_-=-\log\left(1-\frac{1}{1+e^{-kx}}\right) \tag{4}$$

可以用链式法则很容易地计算出损失的微分:

$$\frac{\partial l_+}{\partial x} = -kf(x)e^{-kx} \tag{5}$$

$$\frac{\partial l_-}{\partial x} = kf(x) \tag{6}$$

由微分可知: (1) 梯度被放大系数 k 增大; (2) 梯度的放大对大多数错误预测区域都是显著的, 从而有利于优化和帮助产生更显著的预测. 此外, 当$x = P_{i,j} - T_{i,j}$时, P 的梯度受到 T 的影响, 并在前景和背景之间重新缩放.

3.4 标签生成

概率图的标签生成是受到 PSENet[7] 的启发, 如图 4 所示, 给定一个文本图像, 其文本区域的每个多边形由一组线段来描述:

$$G = \{S_k\}_{k=1}^{n} \tag{7}$$

式中, n 是顶点的数量, 在不同的数据集中可能不同, 如 ICDAR2015 数据集的 4 个顶点和 Total-Text 数据集的 16 个顶点. 然后, 使用 Vatti 裁剪算法将多边形 G 缩小到G_s, 收缩的偏移量 D 是由原多边形的周长 L 和面积 A 计算出来的:

$$D = \frac{A(1-r^2)}{L} \tag{8}$$

其中, r 是收缩比, 设置为 0.4.

通过类似的过程, 可以为阈值图生成标签. 首先将文本多边形 G 以相同的偏移量 D 展开至G_d, 将G_s和G_d之间的间隙作为文本区域的边界, 通过计算到 G 中最接近的线段的距离来生成阈值图的标签.

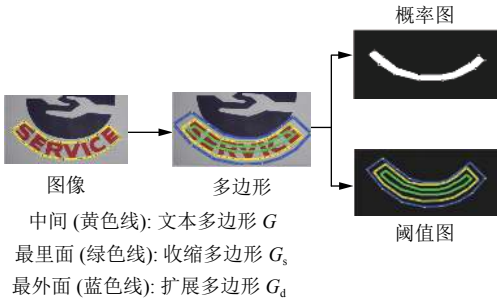

图 4 标签生成

3.5 损失函数

损失函数 L 可以表示为概率图L_s的损失、二值图L_b的损失和阈值图L_t的损失的加权和:

$$L = L_s + \alpha \times L_b + \beta \times L_t \tag{9}$$

根据损失的数值, α和β分别设置为 1 和 10.

本文对L_s和L_b都应用了二元交叉熵损失 (BCE). 为了克服正负数的不平衡, 在 BCE 损失中采用了 hard negative mining 的方法.

$$L_s = L_b = \sum_{i \in S_l} y_i \log x_i + (1-y_i)\log(1-x_i) \tag{10}$$

S_l是正负比为1 : 3的采样集.

L_t为扩展多边形G_d内预测和标签之间的 L_1 距离之和:

$$L_t = \sum_{i \in R_d} |y_i^* - x_i^*| \tag{11}$$

其中, R_d是扩展多边形G_d内像素的一组索引, y^*是阈值图的标签.

在推理阶段, 使用概率图来生成文本边界框, 框的形成过程包括 3 个步骤: (1) 将概率图二值化为常数阈值 (0.2), 得到二值图; (2) 从二值图中得到连通区域 (缩小后的文本区域); (3) 使用 Vatti 裁剪算法中的偏移量 D'对缩小的区域进行扩展. D'计算为:

$$D' = \frac{A' \times r'}{L'} \tag{12}$$

其中, A'是收缩多边形的面积, L'是收缩多边形的周长, r'设置为 1.5.

4 实验与分析

4.1 数据集

SynthText 是一个包含 800 k 图像的合成数据集, 此数据集仅用于对本文的模型进行预训练.

ICDAR2015 是多方向文本检测常用的数据集, 它由 1000 张训练图像和 500 张测试图像组成, 文本实例以单词级标注.

Total-Text 是包含各种形状文本的数据集, 包括水平、多方向和曲线文本实例, 它由 1255 张训练图像和 300 张测试图像组成, 文本实例以单词级标注.

4.2 实验设计

对于所有模型, 首先使用 SynthText 数据集对它们进行 100 k 次迭代预训练, 然后使用 1200 epochs 对真实数据集上的模型进行微调, 训练批大小设置为 8. 本文遵循多学习率策略, 当前迭代的学习率$L_r = Ir \times (1 - i/\max_i)^p$, 其中初始学习率$Ir$设置为 0.001, p为 0.9, 权重衰减为 0.0001, 动量为 0.9, $\max_i$表示最大迭代次数, 这取决于最大的 epoch.

训练数据的数据扩充包括: (1) 在$(-10°, 10°)$范围内随机旋转角度; (2) 随机裁剪; (3) 随机翻转. 所有处

理后的图像都重新调整为 640×640, 以提高训练效率.

在训练阶段, 忽略所有数据集中标记为“DO NOT CARE”的模糊文本区域. 在推理阶段, 保持测试图像的高宽比, 并通过为每个数据集设置合适的高度来重新调整输入图像的大小. 推理速度测试批大小为 1, 在单个线程中使用单个 1080ti GPU, 推理时间包括模型前向传播时间和后处理时间.

本文检测性能评测方法主要考虑 3 个性能参数: 准确率 *Precision*、召回率 *Recall* 和综合评价指标 *F-measure*, 其中综合评价指标是准确率与召回率的调和平均值, 该值是评价文本检测方法性能的综合指标. 定义如下:

$$Precision = \frac{|TP|}{|E|} \tag{13}$$

$$Recall = \frac{|TP|}{|T|} \tag{14}$$

$$F-measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{15}$$

其中, TP, E, T 分别表示正确的检测结果集合, 检测结果集合以及 Ground-truth 集合.

4.3 实验结果与分析

为了验证特征金字塔增强模块 FPEM 和特征融合模块 FFM 的有效性, 在数据集 ICDAR2015 实验中, 与特征金字塔 FPN 进行了实验对比, 如表 1 所示, 不管主干网络是 ResNet-18 还是 ResNet-50, FPEM-FFM 都具有较高的性能 (73.72% vs 79.14%, 75.59% vs 79.96%) 和速度 (4.26 vs 21.31, 2.84 vs 14.22).

表 1 不同设置的实验结果

Backbone	FPN	FPEM-FFM	DB	ICDAR2015			
				Precision	*Recall*	*F-measure*	FPS
ResNet-18	√	×	×	71.82	75.73	73.72	4.26
ResNet-18	×	√	×	84.93	74.09	79.14	21.31
ResNet-18	×	√	√	**86.41**	**76.36**	**81.07**	**43**
ResNet-50	√	×	×	76.50	74.70	75.59	2.84
ResNet-50	×	√	×	84.23	76.12	79.96	14.22
ResNet-50	×	√	√	**89.27**	**75.56**	**82.42**	**27**

为了验证可微二值化模块 DB 的有效性, 在数据集 ICDAR2015 实验中进行了有无 DB 模块的实验对比, 从表 1 中可以看到 DB 显著地提高了数据集上 ResNet-18 和 ResNet-50 的性能. 对于 ResNet-18 主干网, DB 在 ICDAR2015 数据集上实现了 1.93% 的性能提升. 对于 ResNet-50 主干网, DB 在 ICDAR2015 数据集上实现了 2.46% 的性能提升. 此外, 在两个主干网中, 有 DB 模块比没有 DB 模块的速度都提高了约 2 倍.

从表 1 中可以看到 ResNet-50 主干网模型比 ResNet-18 模型性能更好, 但运行速度更慢. 具体来说, ResNet-50 模型比 ResNet-18 模型的性能高 1.35%, 但时间成本约为 ResNet-18 的 1.6 倍.

本文将所提出的方法与之前的方法在两个标准数据集上进行了比较, 包括多方向文本数据集 ICDAR2015 和曲线文本数据集 Total-Text.

ICDAR2015 数据集是一个面向多方向的文本数据集, 它包含许多小的和低分辨率的文本实例. 在表 2 中可以看到, 与之前最快的方法 EAST 相比, “FPEM-FFM-DB(ResNet-50)”的性能比它高 4.22%, 运行速度快了两倍. 当使用 ResNet-18 主干时, “FPEM-FFM-DB(ResNet-18)”速度可以达到 43 fps, *F-measure* 为 81.07%.

表 2 ICDAR2015 数据集上的检测结果

方法	*Precision*	*Recall*	*F-measure*	FPS
CTPN(2016)[12]	74.2	51.6	60.9	7.1
EAST(2017)[5]	83.6	73.5	78.2	13.2
SSTD(2017a)[13]	80.2	73.9	76.9	7.7
Textboxes++(2018)[4]	87.2	76.7	81.7	11.6
Textsnake(2018)[6]	84.9	80.4	82.6	1.1
PSENet-1s(2019a)[7]	86.9	84.5	85.7	1.6
FPEM-FFM-DB(ResNet-18)	**86.41**	**76.36**	**81.07**	**43**
FPEM-FFM-DB(ResNet-50)	**89.27**	**76.56**	**82.42**	**27**

本文在曲线文本数据集 Total-Text 上验证所提出方法的鲁棒性, 如表 3 所示, 本文的方法在性能和速度上都达到了较好的结果. 具体来说, “FPEM-FFM-DB(ResNet-50)”在性能上比之前的最新方法高出 1.15%, 同时运行速度比之前的方法都快. 使用 ResNet-18 主干网可以进一步提高速度, 但性能会有所下降. 与最近的基于分割的方法 PSENet 相比, “FPEM-FFM-DB(ResNet-50)”速度快 10.5 倍, “FPEM-FFM-DB (ResNet-18)”速度快 17.9 倍.

表 3 Total-Text 数据集上的检测结果

方法	*Precision*	*Recall*	*F-measure*	FPS
Textsnake(2018)[6]	82.7	74.5	78.4	–
ATRR (2019b)[1]	80.9	76.2	78.5	–
Make TextSpotter(2018)[14]	82.5	75.6	78.6	–
TextField(2019)[15]	81.2	79.9	80.6	–
CSE(2019b)[16]	81.4	79.1	80.2	–
PSENet-1s(2019a)[7]	84.0	78.0	80.9	3.9
FPEM-FFM-DB(ResNet-18)	**86.62**	**73.19**	**79.34**	**69.88**
FPEM-FFM-DB(ResNet-50)	**89.35**	**75.86**	**82.05**	**40.86**

5 结论与展望

本文提出了一种可以高效准确的检测任意形状文本的框架, 该框架包括由特征金字塔增强模块和特征融合模块组成的轻量级分割头和分割网络中的可微二值化过程, 该分割头既有利于特征提取, 又能带来较小的额外计算量, 可微二值化模块可以显著地提高文本检测的性能. 即使使用轻量级主干网络 (ResNet-18), 本文的方法也能以较快推理速度在测试数据集上实现可比较的性能. 虽然本文的方法呈现出了一个可以与其他先进方法相媲美的效果, 但是实现实时精确的自然场景文本检测任务还面临诸多挑战, 未来仍有很多工作有待去解决.

参考文献

1 Wang XB, Jiang YY, Luo ZB, *et al*. Arbitrary shape scene text detection with adaptive text region representation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, CA, USA. 2019. 6442–6451.

2 Liu XY, Meng GF, Pan CH. Scene text detection and recognition with advances in deep learning: A survey. International Journal on Document Analysis and Recognition, 2019, 22(2): 143–162. [doi: 10.1007/s10032-019-00320-5]

3 Liao MH, Shi BG, Bai X, *et al*. Textboxes: A fast text detector with a single deep neural network. Proceedings of the 31st AAAI Conference on Artificial Intelligence. San Francisco, CA, USA. 2017. 4161–4167.

4 Liao MH, Shi BG, Bai X. Textboxes++: A single-shot oriented scene text detector. IEEE Transactions on Image Processing, 2018, 27(8): 3676–3690.

5 Zhou XY, Yao C, Wen H, *et al*. East: An efficient and accurate scene text detector. Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, USA. 2017. 2642–2651.

6 Long SB, Ruan JQ, Zhang WJ, *et al*. Textsnake: A flexible representation for detecting text of arbitrary shapes. Proceedings of the 15th European Conference on Computer Vision. Munich, Germany. 2018. 19–35.

7 Wang WH, Xie EZ, Li X, *et al*. Shape robust text detection with progressive scale expansion network. Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, CA, USA. 2019. 9328–9337.

8 Zhang CQ, Liang BR, Huang ZM, *et al*. Look more than once: An accurate detector for text of arbitrary shapes. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, CA, USA. 2019. 10544–10553.

9 Dai YC, Huang Z, Gao YT, *et al*. Fused text segmentation networks for multi-oriented scene text detection. Proceedings of 2018 24th International Conference on Pattern Recognition. Beijing, China. 2018. 3604–3609.

10 Wang WH, Xie EZ, Song XG, *et al*. Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. Proceedings of 2019 IEEE/CVF International Conference on Computer Vision. Seoul, Republic of South Korea. 2019. 8439–8448.

11 Liao MH, Wan Y, Yao C, *et al*. Real-time scene text detection with differentiable binarization. arXiv preprint arXiv: 1911.08947, 2019.

12 Tian Z, Huang WL, He T, *et al*. Detecting text in natural image with connectionist text proposal network. Proceedings of the 14th European Conference on Computer Vision. Amsterdam, the Netherlands. 2016. 56–72.

13 He P, Huang WL, He T, *et al*. Single shot text detector with regional attention. Proceedings of 2017 IEEE International Conference on Computer Vision. Venice, Italy. 2017. 3066–3074.

14 Lyu PY, Liao MH, Yao C, *et al*. Mask TextSpotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. Proceedings of the 15th European Conference on Computer Vision. Munich, Germany. 2018. 71–88.

15 Xu YC, Wang YK, Zhou W, *et al*. TextField: Learning a deep direction field for irregular scene text detection. IEEE Transactions on Image Processing, 2019, 28(11): 5566–5579. [doi: 10.1109/TIP.2019.2900589]

16 Liu ZC, Lin GS, Yang S, *et al*. Towards robust curve text detection with conditional spatial expansion. Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, CA, USA. 2019. 7261–7270.