基于分割的自然场景下文本检测方法与应用*

陈小顺，王良君
（江苏大学 计算机科学与通信工程学院，江苏 镇江 212013）

摘 要：自然场景文本检测识别在智能设备中应用广泛，而对文本识别的第一步则是对文本进行精确的定位检测。对于现有像素分割方法 PixelLink 中存在的弯曲文本定位包含过多背景信息、检测图像后处理不足两个主要问题提出改进。引入特征通道注意力机制，关注生成特征图中特征通道间的权重关系，提升检测方法的鲁棒性。接着改变公开数据集标注形式，将坐标点表示为一串带有方向的序列形式，在 LSTM 模型中进行多边形框的学习与框定。最后在公开数据集和自建数据集上进行文本检测测试。实验表明，改进的检测方法在各数据集中表现优于原方法，与当前领先方法精度相近，能够在各个环境中完成对文本的检测功能。

关键词：像素分割；注意力机制；LSTM；自然场景文本检测
中图分类号：TN911.73；TP391.4 **文献标识码**：A DOI：10.16157/j.issn.0258-7998.200316

中文引用格式：陈小顺，王良君. 基于分割的自然场景下文本检测方法与应用[J].电子技术应用，2021，47(2)：54-57.
英文引用格式：Chen Xiaoshun，Wang Liangjun. Text detection and application in natural scene based on segmentation[J]. Application of Electronic Technique，2021，47(2)：54-57.

Text detection and application in natural scene based on segmentation

Chen Xiaoshun，Wang Liangjun
(School of Computer Science and Telecommunication Engineering，Jiangsu University，Zhenjiang 212013，China)

Abstract：Text recognition in nature scene is currently applied in various intelligence equipment. The first step of text recognition is to precisely locate the text. In the Pixel Link text location methods，there are mainly two problems：too much background information is incorporated in the text region，and the test accuracy is insufficient. Aiming at these issues，an improved text location method was proposed to precisely locate the text in the natural scene. At first，an attention mechanism was incorporated into the original network. By focusing on the weight relationship between feature channels in the generated feature map，one can improve the weight coefficient of effective feature channels，and suppress the weight of inefficient or invalid feature channels. In the second，by changing the form of data set annotation，the coordinate points can be expressed as a series of sequence forms，so that the text lines can be framed adaptively in the LSTM model. At last，the located object is rotated according to the angle between a pair of vertexes in the polygon frame，and is subsequently fed to the text recognition interface to obtain the final character. Finally，the text detection test is carried out on the open data set and self-built data set. The experimental results show that the improved detection method is superior to the original method on different dataset，and the accuracy is similar to the current leading method.
Key words：pixel segmentation；attention mechanism；LSTM；natural scene text detection

0 引言

视觉图像是人们获取外界信息的主要来源，文本则是对事物的一种凝练描述，人通过眼睛捕获文本获取信息，机器设备的眼睛则是冰冷的摄像头。如何让机器设备从拍照获取的图像中准确检测识别文本信息逐渐为各界学者关注。

现代文本检测方法多为基于深度学习的方法，主要分为基于候选框和基于像素分割的两种形式。本文选择基于像素分割的深度学习模型作为文本检测识别的主要研究方向，能够同时满足对自然场景文本的精确检测，又能保证后续设备功能(如语义分析等功能)的拓展。

1 基于像素分割的文本检测方法

1.1 PixelLink 算法原理

PixelLink[1]算法训练 FCN[2]预测两种分类：文本与非文本像素、像素间连接关系。数据集中的文本框内像素整体作为文本像素，其他为非文本像素。与九宫格类似，每个像素的周围有 8 个相邻的像素，对应有 8 种连接关系。文本与非文本像素之间的值为负，文本与文本像素之间的值为正，非文本像素之间的值为零。将值为正的像素与相邻 8 个像素之间的连接关系连通成一片分区，

* 基金项目：国家自然科学基金(61601202)；江苏省自然科学基金(BK20140571)；江苏大学高级专业人才科研启动基金(14JDG038)

每个连通区则代表分割出的文本区。最后通过 OpenCV 中的 minAreaRect 方法直接得到文本区的最小外接矩形边界框。

1.2 文本检测网络模型设计

改进后网络模型如图 1 所示，通过 Mask map 连接。在原有 VGG16[3]网络模型每个池化层后增加图 2 中 SE Block[4]以获取每个特征通道的权重，提升有用特征并抑制低效特征通道。

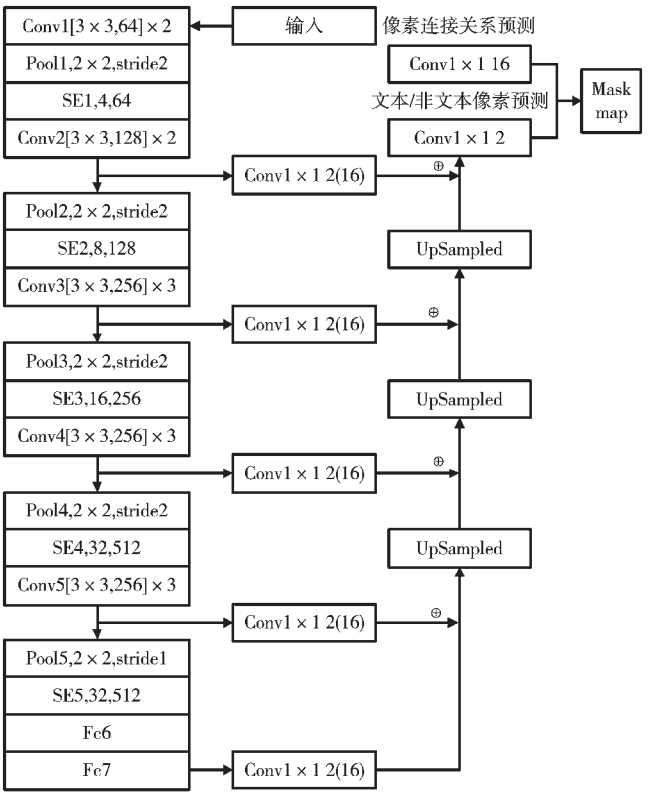

图 1 Mask map 生成

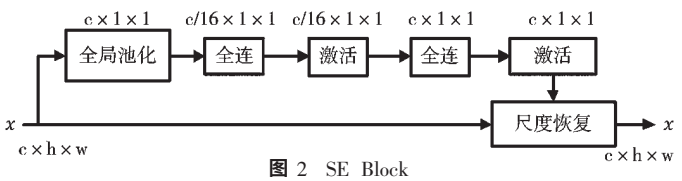

图 2 SE Block

与 FCN 中方法相似，从 Conv3、Conv4、Conv5、Fc7 层进行上采样 UpSampled 与融合⊕，使用双线性插值作为上采样方法，使用加和操作作为融合方法，得到预测特征图 Mask map，过程如图 1 所示。除 Pool5 步长为 1，其余池化层步长都为 2。其中 Fc7 与 Conv5 大小一致，可不经过上采样直接相加。

模型中 1×1 的卷积核共两种，其中 2 个 1×1 的卷积核用于文本和非文本像素预测，16 个 1×1 的卷积核用于像素连接关系预测。

图 2 为插入 PixelLink 方法中的 SE Block，输入特征图与计算后输出尺度不变。

坐标点可以看做是序列问题[5]，对 Mask map 图中生成的矩形框区域进行边界框预测，每次预测一对坐标点，直至矩形框边界。有隐性约束条件，例如第 4 个点必须在第 2 个点的右边，后续对特征图 Mask map 进行基于 RNN 的自适应文本框预测，采用长短期记忆 LSTM[6]模型处理队列顺序问题，最终完成对文本的精确定位。图 3 为文本框预测部分模型结构。

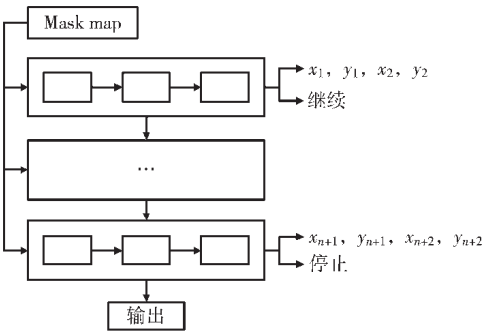

图 3 文本框生成

1.3 文本检测网络模型训练

1.3.1 公开数据集重新标定

自然场景中的文本多用旋转矩形框和四边形框定位，坐标通常以顺时针方向标注。本文将坐标点按照上下一对形式从左到右的顺序排列，通过此方法将公开数据集的坐标数据进行重新编排。

1.3.2 损失函数定义

改进后算法总体损失函数定义如下：

$$L_{sum}=\lambda_1 L_{pixel}+\lambda_2 L_{link}+\lambda_3 \sum_{i\in\{x_1,y_1,x_2,y_2,\cdots,x_n,y_n\}} L_{reg}(u_i,u_i^*)+\lambda_4 \sum_{i\in\{l_1,l_2,\cdots,l_{n/2}\}} L_{cls}(p_i,l_i^*) \quad (1)$$

其中，文本与非文本分类任务上的损失函数 L_{pixel} 和像素连接关系任务的损失函数 L_{link} 与原像素连接 PixelLink 算法保持一致；边界点回归任务损失函数 L_{reg}、停止/继续标签分类任务损失函数 L_{cls} 为框点对预测中的损失函数；λ_1、λ_2、λ_3、λ_4 分别为文本与非文本分类任务、像素连接关系任务、边界点回归任务、停止/继续标签分类任务的权重参数，因像素连接关系预测任务、边界点回归任务、停止/继续标签分类任务都是在第一个文本像素任务基础上进行计算的，所以像素分类任务比这 3 种任务更重要，本实施例中 λ_1=2，λ_2、λ_3、λ_4 默认设置为 1。

边界点回归任务损失函数 L_{reg} 推导如下：设定 v 是像素连接关系预测任务所得特征图 Mask map 的最小外接矩形框，包含矩形的中心点坐标(v_x,v_y)，中心点距离矩形边框的宽 v_w 和高 v_h，$v=(v_x,v_y,v_w,v_h)$。设定 u 是数据集中边界点的真实坐标集，$u=(u_{x_1},u_{y_1},\cdots,u_{x_n},u_{y_n})$，$u^*$是在边界点回归任务中的预测点坐标集，$u^*=(u_{x_1}^*,u_{y_1}^*,\cdots\cdots,u_{x_n}^*,u_{y_n}^*)$。为了能够更好地学习场景文字中各个文字不

同的尺度，对预测的边界点坐标$(u_{x_i}^*, u_{y_i}^*)$作如下处理：

$$\begin{cases} u_{x_i}^* = \dfrac{x_i^* - x_a}{w_a} \\ u_{y_i}^* = \dfrac{y_i^* - y_a}{h_a} \end{cases} \tag{2}$$

其中，x_i^*和y_i^*表示当前文本边界点预测的坐标，x_a和y_a表示当前预测区域对应外接矩形框v_a的中心点坐标，w_a和h_a表示对应的外接矩形框的宽和高。损失函数计算公式如下：

$$L_{reg}(u, u^*) = \text{smooth}_{L1}(u - u^*)$$
$$\text{smooth}_{L1}(t) = \begin{cases} 0.5t^2 & |t|<1 \\ |t|-0.5 & \text{其他} \end{cases} \tag{3}$$

函数$\sum_{i \in \{x_1, y_1, x_2, y_2, \cdots, x_n, y_n\}} L_{reg}(u_i, u_i^*)$为边界点回归损失函数，其中$n$为边界点坐标总数量。而对应的停止/继续标签分类任务损失函数L_{cls}为普通二分类对数损失函数，在边界点回归任务中预测一对坐标后对这一对坐标进行标签分类，定义如下：$L_{cls}(p_i, l_i^*) = -\log p_{i_{l_i^*}}$。其中，$l_i$表示第$i$对边界点标签，$l_i^*$是当前标签的分类，$l_i^*=1$表示这一对坐标点未到文本框尽头，分类为继续，$l_i^*=0$表示这一对坐标点已经将所有文本框定，分类为停止。$p_i$是当前标签经过分类器分类后属于停止/继续标签的概率。因边界框点回归数量为n，则在标签分类任务中分类总次数为$n/2$，该分类的损失函数为$\sum_{i \in \{l_1, l_2, \cdots, l_{n/2}\}} L_{cls}(p_i, l_i^*)$。

1.3.3 训练方法与实验环境

与 Pixellink 方法相似，使用 xavier 参数[7]初始化方法，无需使用 ImageNet 数据集[8]预训练。算法在服务器中用两张 TeslaP4 显卡进行训练，使用 Anaconda+PyCharm 管理，环境及依赖：tensorflow-gpu==1.14，ujson，threadpool，opencv，matplotlib，Pillow，Cython，setproctitle，shapely，Python3.6。初始 100 次迭代中保持学习速率为10^{-3}，后续迭代中保持10^{-2}不变，在 ICDAR2015 数据集上整体迭代约 30 000 次后再将模型训练结果作为预训练值，在其他数据集上进行训练。其中 batch_size 设定为 4，处理器为 Intel Xeon Sliver(2.1 GHz)，机器内存为 40 GB。每次迭代需要 0.8 s 左右，总训练过程约为 15 h。

2 实验结果与分析

2.1 公开数据集测试

本文主要评价方法为 IOU 算法，表 1 中R为召回率，P为精准率，F为综合评价，* 表示算法是基于分割的检测方法，其他为基于候选框的检测方法。测试结果表明本文所改进的方法在各个数据集上均超过原有方法。在对曲向文本的识别方法中领先，并且在水平文本和倾斜文本检测中能够接近基于候选框检测方法的检测精度。

2.2 自建数据集测试

为测试文本检测方法在实际生活应用中的检测效果，使用 OV5648USB 摄像头模块累计拍摄 300 张不同场景下图像作为测试图像，原图分辨率大小为：2 592×1 944。如图 4 所示，为突出显示检测结果，截取主要定位部分，图像中的中文部分以中文词语作为一条文本行，英文以短语作为一条文本行。共计 2 506 条文本，其中 1 964 条

表 1 公开数据集测试结果

数据集	CTW1500[9]			ICDAR2015[10]			MSRA-TD500[11]		
算法	R	P	F	R	P	F	R	P	F
SegLink[12]	40.0	42.3	40.8	76.8	73.1	75.0	70.0	86.0	77.0
CTD+TLOC[9]	69.8	77.4	73.4	–	–	–	–	–	–
*TextSnake[13]	85.3	67.9	75.6	80.4	84.9	82.6	73.9	83.2	78.3
*PixelLink	–	–	–	82.0	85.5	83.7	73.2	83.0	77.8
InceptText[14]	–	–	–	80.6	90.5	85.3	79.0	87.5	83.0
* 本文算法	76.4	80.0	78.2	83.3	86.5	84.9	78.2	84.6	81.3

水平倾斜文本　像素分割图　弯曲文本　像素分割图

(a)测试图 1　(b)测试图 2

图 4 自建数据集检测结果

中文文本(包含数字),542 条英文文本。

深色框为改进前方法的定位结果,浅色框为改进后的方法定位结果,右图为对应的像素分割后的检测结果。从特征图中可以看出,本文方法对长条形的英文检测敏感,能够有效检测出长条形的英文,对曲向的英文有着不错的识别能力。在对图像进行检时,平均每张检测速度为 0.89 s,即 FPS=1.12,R=72.9,P=70.0,F=71.4。

3 结论

本文提出改进的文本检测方法在数据集表现上均超过原有方法,接近当前领先的算法精度,能够提高已有文本识别系统对自然场景下曲向文本与模糊文本的识别精度。后续结合自然语言处理和语义分割任务,又可以将所识别的文本内容、文本背景内容组合生成关于一张图片中文本的具体描述内容,使得使用者获取更多的文本信息。

参考文献

[1] Deng Dan, Liu Haifeng, Li Xuelong, et al. Pixellink: detecting scene text via instance segmentation[C]. Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[2] LONG J, SHELHAMER E, DARRELL T. Fully convolutional networks for semantic segmentation[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015: 3434-3440.

[3] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[C]. International Conference on Learning Representations, 2015.

[4] Hu Jie, Shen Li, Sun Gang. Squeeze-and-excitation networks[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2018: 7132-7141.

[5] Wang Xiaobing, Jiang Yingying, Luo Zhenbo, et al. Arbitrary shape scene text detection with adaptive text region representation[C]. 2019 IEEE/CVF Conference on CVPR, 2019.

[6] HOCHREITER S, SCHMIDHUBER J. Long short-term memory[J]. Neural Computation, 1997, 9(8): 1735-1780.

[7] GLOROT X, BENGIO Y. Understanding the difficulty of training deep feedforward neural networks[C]. Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, 2010: 249-256.

[8] Deng Jia, Dong Wei, SOCHER R, et al. ImageNet: a large-scale hierarchical image database[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2009: 248-255.

[9] Liu Yuliang, Jin Lianwen, Zhang Shuaitao, et al. Curved scene text detection via transverse and longitudinal sequence connection[J]. Pattern Recognition, 2018, 90(6): 337-345.

[10] KARATZAS D, GOMEZ-BIGORDA L, NICOLAOU A, et al. ICDAR 2015 competition on robust reading[C]. Document Analysis and Recognition (ICDAR), 13th International Conference, IEEE, 2015: 1156-1160.

[11] Yao Cong, Bai Xiang, Liu Wenyu, et al. Detecting texts of arbitrary orientations in natural images[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012: 1083-1090.

[12] Shi Baoguang, Bai Xiang, BELONGIE S. Detecting oriented text in natural images by linking segments[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017: 3482-3490.

[13] LONG S, RUAN J, ZHANG W, et al. TextSnake: a flexible representation for detecting text of arbitrary shapes[C]. European Conference on Computer Vision (ECCV), 2018: 19-35.

[14] Yang Qiangpeng, Cheng Mengli, Zhou Wenmeng, et al. IncepText: a new inception-text module with deformable PSROI pooling for multi-oriented scene text detection[C]. Twenty-Seventh International Joint Conference on Artificial Intelligence (LJCAI-18), 2018.

(收稿日期:2020-04-17)

作者简介:

陈小顺(1996-),男,硕士研究生,主要研究方向:人工智能。

王良君(1982-),通信作者,男,博士研究生,讲师,主要研究方向:图像编码、压缩感知,E-mail:ljwang0819@ujs.edu.cn。

(上接第 53 页)

[7] IMOKHAI I. OkHttp interceptors with Retrofit[EB/OL]. (2019-10-23)[2020-03-24]. https://medium.com/swlh/okhttp-interceptors-with-retrofit-2dcc322cc3f3.

[8] 扔物线.给 Android 开发者的 RxJava 详解[EB/OL].(2017-09-16).[2020-03-24].http://gank.io/post/560e15be2dca-930e00da1083.

(收稿日期:2020-03-24)

作者简介:

李想(1992-),男,硕士,助理研究员,主要研究方向:移动端研发、遥感大数据。

特日根(1987-),通信作者,男,博士,副高级研究员,主要研究方向:遥感大数据,E-mail:terigen@charmingglobe.com。

版权声明