

基于双门限梯度模式的图像文字检测方法

蔡文哲¹ 王斌君¹ 李培岳²

(中国人民公安大学网络安全保卫学院 北京 102623)¹

(中国人民公安大学研究生院 北京 100038)²

摘 要 对复杂自然背景下的图像文字检测技术进行了研究,提出了一种基于双门限梯度模式的图像文字检测方法。首先,在文字粗检测阶段中,该方法抽取了最大极值稳定区域(Maximally Stable Extremal Regions,MSER)作为候选文字区域,避免了对整幅图像进行扫描,极大地提高了检测速度和实时性;其次,在文字精检测阶段的特征提取部分,为了克服文字区域颜色对比反转问题和自然图像的噪声干扰问题,提出了一种双门限梯度模式特征来描述文字区域的纹理特征;最后,在文字精检测的检测器设计中,利用极限学习机构造新的级联型 ELM(Extreme Learning Machine)检测器,极大地缩短了分类器的训练时间。实验结果表明,该方法不仅具有优良的检测性能,而且能极大地缩短分类器训练时间和检测时间。

关键词 图像文字检测,双门限梯度模式,极限学习机,多分类器组合

中图法分类号 TP391.41 文献标识码 A

New Image Text Detection Method Based on Double-threshold Gradient Pattern

CAI Wen-zhe¹ WANG Bin-jun¹ LI Pei-yue²

(College of Network Security, People's Public Security University of China, Beijing 102623, China)¹

(Graduate School, People's Public Security University of China, Beijing 100038, China)²

Abstract This paper studied the traditional image text detection approaches and proposed a new image text detection method based on double-threshold gradient pattern with a pretty fast speed in both classifier training and implementing. Firstly, in the rough detection phase, the maximally stable extremal regions(MSER) was extracted as a candidate text area, to avoid scanning the whole image, greatly improving the detection speed and real-time. Secondly, in the feature extraction part of refine detection phase, in order to overcome the text area color contrast inversion problem and the problem of noise in natural image, this paper creatively presented a dual threshold gradient mode feature to describe the texture of the text area feature. Finally, to design the detector for text fine detection, this paper designed a new Cascade ELM(Extreme Learning Machine) detector by limit learning machine, which greatly shortens the classifier training time. The experimental results show that this method not only has excellent detection performance, but also greatly shortens the classifier training time and testing time.

Keywords Image text detection, Double-threshold gradient pattern, Extreme learning machine, Multiple classifier combination

1 引言

近些年,复杂背景下的文字检测领域已经吸引了大量研究者进行研究。其主要分为两大类:基于区域的方法和基于机器学习的方法。

在基于区域的文字检测方面,研究者提出了许多比较有价值的改进方法^[1-4],总体上来看,该方法对于质量较高、文字类型比较少的图像往往能够以很低的计算代价取得比较满意的结果。但是该方法也存在着许多缺陷,对图像质量要求高、

可扩展性差。因此,单纯基于区域的方法很难适应“大数据”条件下变化多端、对可扩展性要求极高的文字检测任务。

在基于机器学习的文字检测方面, Kim^[5]、Chen^[6]、Pan^[7]、Ye^[8]、Goto^[9]等人提出了一些比较有代表性的改进方法。总体上来看,该文字检测方法准确性较高、鲁棒性较好,对于低质量图像和噪声的容忍程度较高并且可扩展性也比较好。但是此类方法存在两个重要的缺陷:分类器训练时间太长,有时甚至难以接受;检测实时性不佳,计算复杂度偏高。

为了在复杂自然背景下更准确和快速地自动识别与检测

本文受国家高技术研究发展计划 863 项目(2013AA014604),公安部公安理论与软科学基金项目(2013LLYJGADX003),中国人民公安大学基本科研业务费项目(2015JKF01251)资助。

蔡文哲(1989—),男,硕士生,CCF 会员,主要研究方向为图形处理与模式识别、网络安全与执法技术,E-mail:513060612@qq.com;王斌君(1962—),男,博士,教授,主要研究方向为图形处理与模式识别、网络安全与执法、信息安全、数据挖掘;李培岳(1988—),男,硕士生,讲师,主要研究方向为图形处理与模式识别。

文字,本文通过借鉴以上两种文字检测方法的优势,提出一种双阶段图像文字检测方法。该双阶段图像文字检测方法能够在检测性能和检测耗时上得到一个较好的平衡,也代表了当前的文字检测技术的发展方向。

2 双阶段图像文字检测方法

2.1 图像文字粗检测方法

虽然最大稳定极值区域(Maximally Stable Extremal Regions,MSER)提取算法^[10]准确地提取出了图像中的文字信息,但是同时也有大量的非文字MSER被提取出来。在一些背景中包含复杂的自然环境的图像中,这种显而易见的非文字MSER的数量仍然很多,如果直接将其输入到精检测阶段的文字检测器中则会拖慢文字检测的速度。在本文中,检测窗口的大小被设定为 24×32 。基于检测窗口的尺寸大小,进一步引入以下3个几何约束对所得到的MSER进行过滤:

(1)在当前尺度下能够被检测的文字MSER区域的像素总数 N 应该满足 $10 \leq N \leq 250$ 。

(2)在当前尺度下能够被检测的文字MSER区域在竖直方向上的最大高度 H 应该满足 $12 \leq H \leq 36$ 。

(3)在当前尺度下能够被检测的文字MSER区域在水平方向上的最大宽度 W 应该满足 $12 \leq W \leq 48$ 。

这3条假设均略微放宽了检测窗口尺寸的限制,用于粗略提取符合检测窗口尺度的文字。没有对MSER引入更多的限制或者约束,其主要原因是避免因引入过多“数据集相关”的经验性假设而削弱所提文字检测方法的可扩展性和可移植性。这样就可以避免通常的基于连通域的方法带来的可扩展性难题,并且充分发挥基于区域方法的速度优势。

如图1所示,即便是如此简单的3条规则,也足以过滤掉大多数的非文字MSER。得到过滤的MSER后,根据这些MSER的中心位置,在每一个MSER中心位置处放置一个检测窗口,以便于在后面的精检测阶段对候选文字区域进行进一步的验证(如果MSER中心过于贴近图片边缘,便将检测窗口略微向图片内侧移动)。最后,粗检测阶段的输出结果就是一系列待验证的候选文字窗口。



图1 几何过滤后的MSER(左)和粗提取的输出(右)

2.2 图像文字精检测方法

在文字检测中效果较好的经典特征有基于小波分解高频系数wavelet的特征、HOG(Histogram of Oriented Gradient)特征^[11]、LBP(Local Binary Pattern)特征^[12]和LGP(Local Gradient Pattern)特征^[13]。为了解决前景和背景中存在的噪声问题,本文在以上特征基础上提出了一种全新的双门限梯度模式DGP(Double-threshold Gradient Pattern)新特征。DGP模式由LGP模式发展而来,能够在描述文字区域时自适应地“过滤”掉前景和背景中噪声或者灰度波动产生的干扰

模式,同时不会像普通的平滑操作或者eLBP^[14]一样对有用的文字边缘产生影响。DGP特征值的具体求解过程如下。

(1)对于当前的像素点 $P(m,n)$,以其为中心在半径为 r 的圆周上均匀地选取 P 个采样点。这一步与LGP和LBP相同。

(2)使用HOG特征中像素的梯度 $G(m,n)$ 与梯度的方向 $\alpha(m,n)$ 的计算公式或者Sobel梯度算子^[15,16]对图像中的每一个像素的梯度 g_{ij} 进行计算。设立一个全局阈值 $\bar{g}_{global} = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N g_{ij}$,即所有像素的梯度值得平均值。

(3)求取 P 个采样点的局部梯度 g_i 的平均值作为局部门限值,即 $\bar{g}_{local} = \frac{1}{P} \sum_{i=1}^P g_i$ 。 $P(m,n)$ 处的DGP求解的计算方式如下:

$$DGP_p^r = \sum_i s(g_i - T) \times 2^i \quad (1)$$

其中, $T = \max(\bar{g}_{local}, \bar{g}_{global})$, $s(x)$ 定义如下:

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (2)$$

(4)求取图像DGP图谱的模式统计直方图作为DGP特征。

从DGP求解过程可以看出,其与LBP及LGP最大的不同是不仅使用了局部门限进行二值化和编码,而且引入了一个全局阈值 \bar{g}_{global} 。这种双门限机制非常简单,并不增加过多的计算复杂度,但十分有效。其原理主要是基于我们对于文字区域的梯度分布规律的一个发现:文字区域的主要组成部分可以分为边缘、前景和背景。如图2所示,在文字区域中,组成有用的重要边缘(在文字区域大部分为文字边缘)的像素平均梯度 \bar{g}_e 要远远大于前景和背景中噪声和灰度波动部分的平均梯度 \bar{g}_g 。同时,有用边缘的像素数目 N_e 要远远小于前景和背景的像素数目 N_g 。由此可得:

$$\bar{g}_g < \bar{g}_{global} = \frac{N_e \bar{g}_e + N_g \bar{g}_g}{N_e + N_g} < \bar{g}_e \quad (3)$$

$$|\bar{g}_g - \bar{g}_{global}| < \|\bar{g}_e - \bar{g}_{global}\| \quad (4)$$

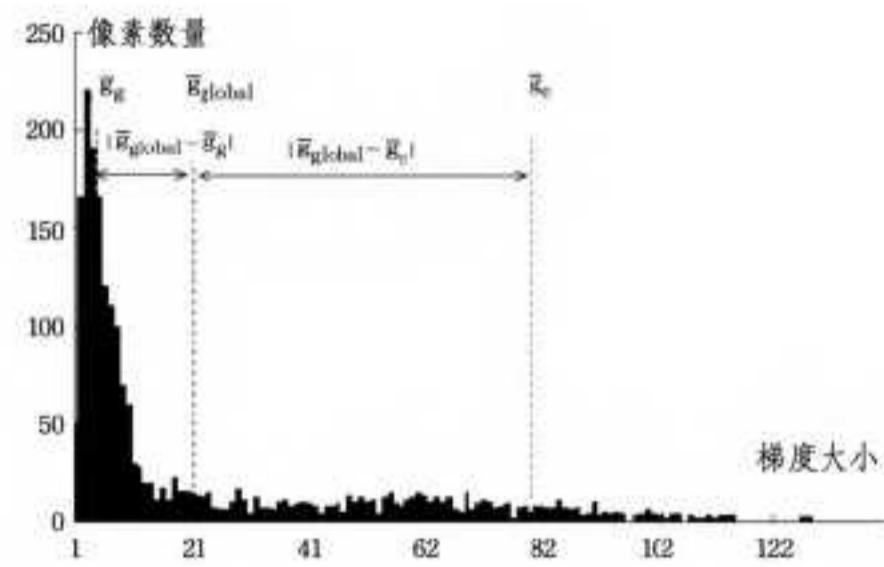


图2 文字区域梯度统计直方图

式(4)成立的原因是有用边缘的像素数目 N_e 要远远小于前景和背景的像素数目 N_g ,而 $\bar{g}_{global} = \frac{N_e \bar{g}_e + N_g \bar{g}_g}{N_e + N_g} = \frac{N_e}{N_e + N_g} \bar{g}_e + \frac{N_g}{N_e + N_g} \bar{g}_g$ 。由于式(3)和式(4)对于绝大多数文字区域均成立,其在求解过程中有如下特点:当使用DGP对前景和背景中灰度较为稳定的区域进行求解时,即局部模式中不存在重要的边缘像素时,有 $\bar{g}_{local} \approx \bar{g}_g < \bar{g}_{global}$,此时有门限 $T = \bar{g}_{global}$,而 \bar{g}_{global} 一般大于局部模式中前景和背景

的像素梯度 g_i , 因此存在噪声和轻微灰度波动的前景和背景像素在求解时会直接被置 0, 这样就不会再产生任何干扰模式。当使用 DGP 对包含重要边缘的区域进行求解时, 由于式 (4) 能够保证 \bar{g}_{global} 更加接近 \bar{g}_g , 因此会以极大概率使 $\bar{g}_{local} > \bar{g}_{global}$ 成立, 此时 $T = \bar{g}_{global}$ 从而使得求解过程与 LGP 基本一致, 不会受到 \bar{g}_{global} 影响, 从而避免有用的边缘在求解时受损。当文字区域的前景和背景的灰度强度变化时, DGP 的全局阈值 \bar{g}_{global} 也会随之变化并始终满足式 (3) 与式 (4)。因此, 从 DGP 的求解过程可以看出, DGP 不但能够产生 0 模式, 还能够在不损害有用边缘的前提下自适应地利用双门限滤除掉前景和背景中的干扰模式, 这一点是传统的平滑操作和简单的门限调整所不能实现的。同时不难看出, DGP 也保留了 LGP 和 LBP 的优势。因此, 与 LGP 和 LBP 等传统特征相比, DGP 对文字区域具有更强的描述能力, 更加适用于文字检测。

如图 3 所示, DGP 可以有效抑制干扰模式, DGP 的背景和前景非常纯净, 重要的文字边缘都几乎完美地保留了下来。甚至对于图 3 中的文字颜色极浅的第三组示例, DGP 也能够很好地描绘出文字的轮廓, 而 eLBP 却不能完成这样的任务。



从左至右依次为: 源图像, LBP, LGP 和 DGP

图 3 不同特征的图谱对比

通过图像文字粗检测方法, 可以极大地缩小文字检测的搜索范围, 只有文字和少量的较难区分的非文字 MSER 被输入到精检测阶段进行进一步的分类和验证。为了对剩余的文字候选区域进行一个精细的分类, 必须提取出最具有分辨能力的特征来描述文字区域的纹理, 同时利用机器学习中的分类器技术设计并组织出一种错误率低、可扩展性能优良的检测器来使用所给的特征进行分类。文字检测方法中最为常用的两种分类器是基于 Adaboost 算法的增强 (boosting) 分类器^[17,18] 与支持向量机 (Support Vector Machine, SVM) 分类器^[19]。在两者的基础上, 为了进一步提高分类器的训练速度, 本文将机器学习领域最新出现的极限学习机 (Extreme Learning Machine, ELM) 引入文字检测领域。

极限学习机 ELM 是由 Huang^[20,21] 等人在 2006 年提出的一种基于单隐层前馈神经网络 (Single-hidden Layer Feed-forward Neural Network, SLFN) 的高效分类器。其最大的优势在于其卓越的训练速度, 相对于其他经典的学习算法能够成百上千倍地提高训练和学习效率。在大数据条件下, 过于冗长甚至难以忍受的训练时间对在线学习、用更大的数据集更新分类器都是极其不利的, 因此在文字检测领域引入极限学习机是很有必要的。

ELM 的基本思想是, 对于基本 ELM (Basic ELM), 在给定隐层节点数目后, 隐层权重和偏置只需随机生成并在后面

的训练过程中保持不变, 不再像传统的后向传播算法 (BP) 那样需要不断地利用误差梯度下降法迭代调整。ELM 中需要进行的是通过求广义逆对输出层的权值和偏置进行调整, 这也是 ELM 能够取得其他分类器难以想象的训练速度的关键所在。同时, Huang^[22] 证明了 ELM 能够求得分类问题的全局最优解, 而 SVM 等分类器一般只能获得局部最优解。因此, ELM 通常能够在实现与传统分类器相近或者更高的泛化性能的前提下成百上千倍地加快学习速度。这也使得 ELM 在大数据背景下成为了一个十分具有前景的机器学习技术, 可以被广泛地应用于在线学习和各种实时机器学习任务。图 4 和图 5 给出了两种最常用的 ELM 的结构示意图, 基于单隐层前馈神经网络的 Basic ELM^[23,24] 和基于核方法的 Kernel-based ELM^[25,26]。

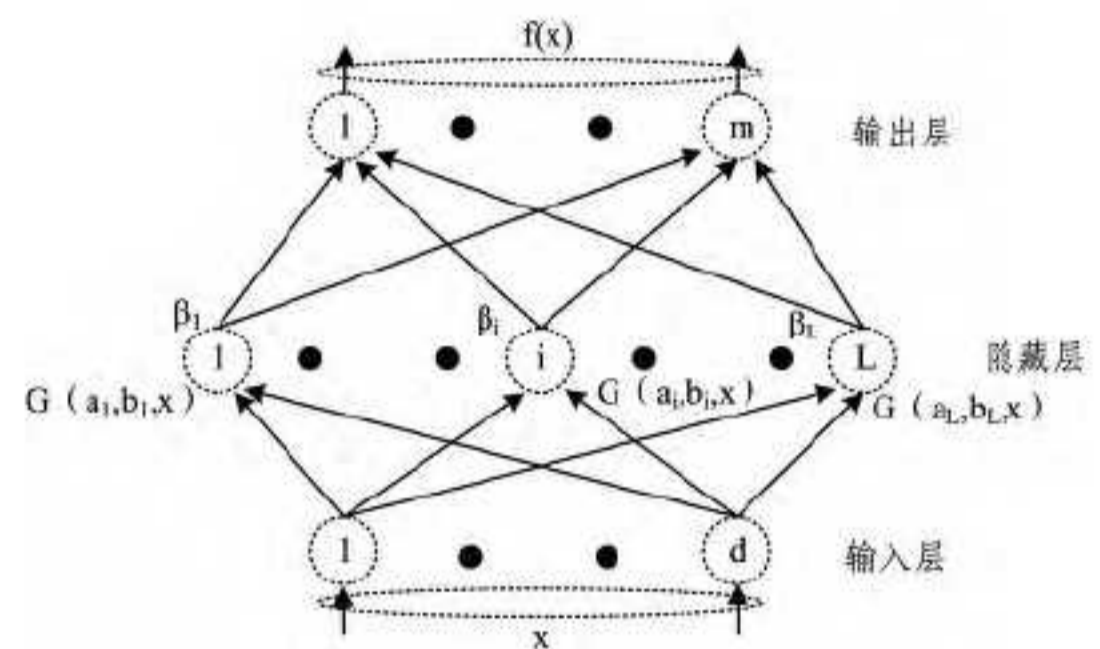


图 4 基于单隐层前馈神经网络的 ELM (Basic ELM)

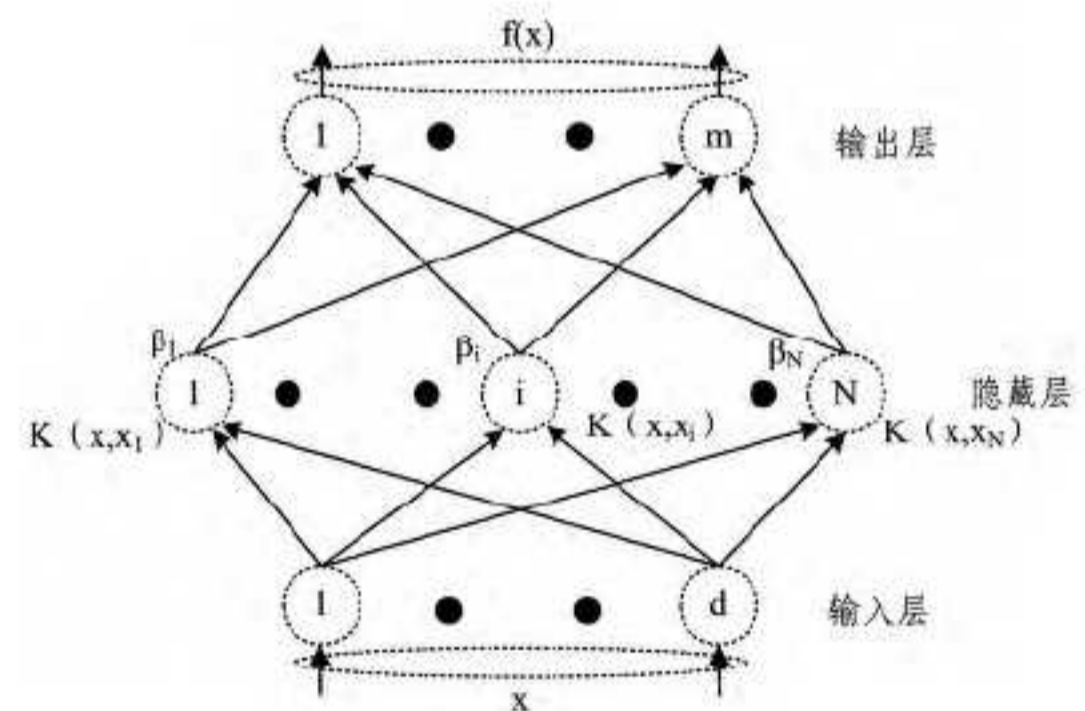


图 5 基于核方法的 ELM (Kernel-based ELM)

给定一个训练集 $X = (\vec{x}_1, \vec{x}_2, \dots, \vec{x}_n)$ 和训练集的标签向量 $\vec{y} = (y_1, y_2, \dots, y_n)$, 其中 \vec{x}_i 为特征向量, 标签 $y_i \in \{-1, 1\}$ 。对于图 4 所示的 Basic ELM, 设给定的隐层节点数目为 L , 其基本的算法基本原理如下。

1. 随机生成每一个隐层节点的权重向量 \vec{a}_i 和偏置 b_i , 其中 $i = 1, 2, \dots, L$ 。
2. 根据训练集 X 和随机生成的隐层参数 (\vec{a}_i, b_i) 计算隐层的输出矩阵 $H_{n \times L}$ 。
3. 根据 H 和给定的标签向量 \vec{y} , 计算输出层的权重向量 $\vec{\beta}$:

$$\vec{\beta} = H^+ \vec{y} \quad (5)$$

其中, H^+ 是 H 的 Moore-Penrose 广义逆 (伪逆), 其求法为:

$$H^+ = H^T (\frac{1}{\lambda} + GHH^T)^{-1} \quad (6)$$

其中, λ 为控制正则化项的系数, 其作用是获得更加稳定的解和提高分类器的泛化性能, I 是逆单位矩阵。Huang 证明了上面的输出层权重求解实际上与下面的优化目标一致:

$$\min \{ \| H\vec{\beta} - \vec{y} \|^2 + \lambda \|\vec{\beta}\|^2 \} \quad (7)$$

其中, $\|\cdot\|$ 代表求范数运算。式(7)表明 ELM 的目标实际上是最小化输出与训练标签的误差向量的范数和正则化项范数之和。最终训练完成的 ELM 对于新来的待分类样本 \vec{x} 的输出表达式为:

$$f(\vec{x}) = h(\vec{x})\vec{\beta} = h(\vec{x})H^T \left(\frac{1}{\lambda} + HH^T \right)^{-1} \vec{y} \quad (8)$$

由此可见, ELM 的训练和求解过程都十分简洁明了, 且需要调整的参数相对于传统的 SLFN 也减少了许多。在此思想上, ELM 发展出了一系列的变化形式, 如使用核方法的 Kernel-based ELM 和用于在线学习的 OS-ELM。图 5 所示的使用核方法的 Kernel-based ELM 主要应用于隐层特征映射未知的情况, 具有训练误差小、泛化能力强的特点, 且不需要调整隐层节点数。在 Kernel-based ELM 中, 核矩阵 Ω_{ELM} 被定义为: $\Omega_{ELM} = HH^T$, 且 Ω_{ELM} 中的每一个元素 $\Omega_{ELM}(i, j) = \vec{h}_i^T \vec{h}_j = K(\vec{x}_i, \vec{x}_j)$ 。与使用内积核的 SVM 类似, Kernel-based ELM 无需显式地求出低维到高维的非线性映射, 输出的求法只需要稍作改动变为:

$$f(\vec{x}) = \begin{bmatrix} K(\vec{x}, \vec{x}_1) \\ \dots \\ K(\vec{x}, \vec{x}_N) \end{bmatrix}^T \left(\frac{1}{\lambda} + \Omega_{ELM} \right)^{-1} \vec{y} \quad (9)$$

ELM 具有众多优良性质, 已经被成功地应用于一系列的分类和回归问题。本文引入极限学习后有效地改进了文字检测器的性能, 特别是在可扩展性方面取得了较好的效果。

3 检测器设计

由于文字本身具有高度的变化性, 到目前为止还没有哪一种公认的单独的特征能够最好地描述文字区域的纹理。因此, 在实际的实验中, 为了能够尽可能利用到各个特征, 各种分类器的优势, 取长补短, 以达到最佳的检测效果, 我们需要将不同种类分类器和不同的特征有效地组织在一起成为一个高性能的文字检测器。

由于 Adaboost 和 SVM 是有监督学习中最好的两种算法, 其均具有很好的分辨能力, 因此级联 Adaboost+SVM 的结构一般能够取得非常优异的检测效果^[26]。但是当面对较为庞大的训练数据集或者出现新的训练数据需要重新对检测器进行训练时, 由于每一级的分类器的训练都需要几小时甚至几天, 使用这种结构就会遇到较大的困难, 非常不灵活。

为了进一步增强级联检测器的可扩展性并获得优良的检测性能, 将 ELM 引入到级联分类器的设计之中, 从而将训练时间压缩到几乎最短, 使得当有新的数据来临时对分类器进行重新训练或者在线学习变得非常简单方便。

由于自然环境中存在大量十分容易与文字混淆的虚警, 单一分类器和单一特征难以满足我们对于检测准确性和降低虚警概率的需求。如图 6 所示, 为了满足检测的需求, 设计的级联分类器包含五级 ELM。在级联分类器的前三级, 使用了由 HOG 特征、DGP_s² 和分块提取的 LBP_s¹ 特征训练的 Basic ELM 分类器, 在级联分类器的后两级, 使用了由分块提取的 LGP_s¹ 特征和分块提取的 DGP_s¹ 训练的 Kernel-based ELM 分类器。从粗检测阶段输出的候选文字样本依次通过这五级 ELM, 只有五级 ELM 分类全部为正的样本才被认为是正样本, 否则即被判定为负样本且被直接丢弃。由于级联 ELM 在数据训练的方便性和可扩展性上有显著优势, 本文主要采

用了这种形式的文字检测器。级联 ELM 的算法流程描述如下。

算法 1 cascadeELM (MSEReset, FeatType, FeatPara, Threshold)

1. DetectedText = Null;
2. for i=1 to size(MSEReset) do
3. for in cascadeELM do
4. FeatVec = FeatCalc(FeatType(j), MSEReset(i), FeatPara(j));
5. curClass = ELMTest(FeatVec, Threshold(j))
6. if curClass == nonText then
7. break;
8. elseif curClass == Text and j == length(cascadeELM)
9. addTextBox(DetectedText, MSEReset(i));
10. end
11. end
12. end
13. return DetectedText

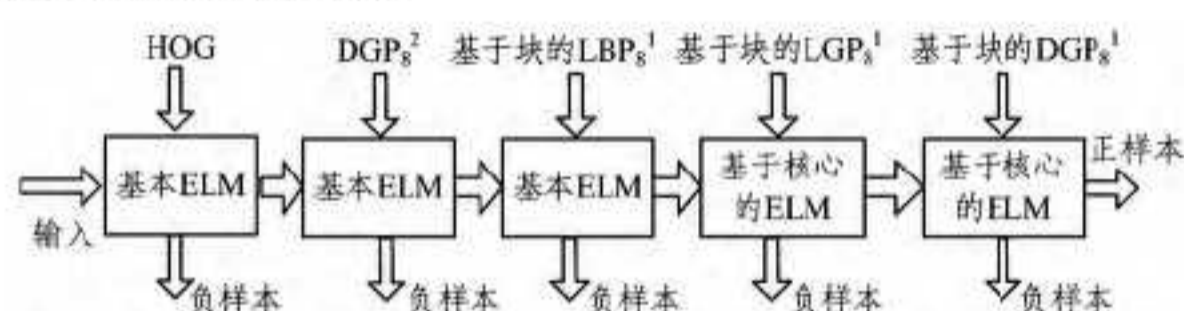


图 6 级联 ELM 分类器

前三级采用 Basic ELM 分类器的主要作用是在较早的阶段就尽可能地滤除大部分相对比较容易分辨的负样本, 所以使用的特征维数都较低且相对更容易计算, 这样就可以避免这些样本进入后面的 Kernel-based ELM 分类器中进行更加费时的特征计算, 减少了检测时间。另外, 由于在测试时 Basic ELM 不需要像 Kernel-based ELM 一样计算核矩阵, 因此其所需的测试时间也更少。相比之下, Kernel-based ELM 虽然需要相对更长的测试时间, 但是由于其采用了核方法, 其训练的错误率更低(在大部分情况下, 泛化性能最优的 Kernel-based ELM 的测试错误率能达到 0), 并且不需要像 Basic ELM 一样调节隐层节点的数量。同时, 我们在实验中发现, 对于分块提取的 LGP_s¹ 特征和分块提取的 DGP_s¹ 使用 Kernel-based ELM 的泛化性能更好。因此, 将 Kernel-based ELM 置于最后两级, 并使用分辨能力最好的但稍微更加费时的 DGP 特征, 但对到达最后两级的文字样本和比较难以分辨的非文字样本进行一个精细的分类。另外一个需要说明的问题是, 由于 Basic ELM 所采用的特征的分辨能力的限制, 为了保证绝大部分(在本文中, 该比例设置为 98% 以上)能够通过前面三级 ELM 而不被错误地抛弃, 我们需要降低前三级 Basic ELM 的判决门限来保证一个较高的检测率。总而言之, 级联的 ELM 能够提高精检测阶段文字检测的速度, 在降低虚警率的同时保证较高的检测率。

4 字框合并、分词与多分辨率分析

通过精检测阶段之后, 将会得到在当前尺度上一系列检测得到的包含文字的检测窗口。为了得到最终的检测结果, 需要将这些检测窗口合并产生文字线(Text Line), 并在这个基础上将文字线按照人类的阅读习惯划分为分开的词语, 以便于将检测结果输入到后端的光学字符识别(Optical Character Recognition, OCR)系统中进行进一步的处理和识别。我们分两步完成这个任务: 首先, 由于在绝大部分场景中文字是

以水平或者近似于水平的方式分布的,按照得到的检测窗口在竖直方向上的重合程度确定出文字线的位置,然后,利用梯度投影分析(Gradient Projection Analysis,GPA)的方法,确定出词语之间的间隔位置,从而对同一文字线中的不同词语进行分割,最后,将不同的分辨率上的检测结果进行合并来得到最终的文字框作为检测结果。

4.1 文字线产生

经基于文字线一般按照水平分布的假设(实际上,竖直的文字线也可以按照类似的方法处理,只需将图像矩阵转置即可),我们认为属于同一文字线的文字窗口在竖直方向(y 方向)的坐标一般比较稳定。具体来讲,对于两个检测到的、前后相邻的文字窗口 W_i 和 W_j ,若其竖直方向上的坐标 y_i 和 y_j 满足 $|y_i - y_j| < r \times Height$ (其中 r 和 $Height$ 分别为重合比例以及检测窗口在竖直方向上的高度),我们就认为两个检测窗口属于同一个文字线。本文中取 $r=0.15$ 。当所有的属于同一个文字行的检测窗口都被找到后,文字线的水平起始位置和终止位置由最左侧和最右侧的两个该文字线的检测窗口确定,同时文字线在竖直方向的坐标由所有窗口的竖直方向坐标的平均值确定,即:

$$y_{textline} = \frac{1}{N_{window}} \sum_{i=1}^{N_{window}} y_i \quad (10)$$

当所有的检测到窗口都被处理之后,能够得到一系列的

文字线,进而能够进行后续处理。

文字线产生的算法流程描述如下所示,文字线产生结果如图 7 所示。

算法 2 TextLineGen(DetectedText, r, Height, Width)

1. %根据文字框中心所处的行位置从小到大进行排序,并将文字框按顺序重新
2. %排列,便于之后的文字线产生
3. TextLine= Null;
4. If DetectedText \neq Null then
5. Order= sortRow(DetectedText', ascend');
6. DetectedText= rearrange(Order, DetectedText);
7. curTextLine= Null;
8. for i=1 to size(DetectedText) do
9. if i==1 then
10. addTextBox(curTextLine, DetectedBox(i));
11. preBox= DetectedTextBox(i);
12. %比对当前的文字框和上一个文字框是否符合合并的条件,否则建立一条新的文字线
13. elseif abs(getRow(DetectedBox(i) - getRow(i))) < r * Height and abs(getCol(DetectedBox(i) - getCol(preBox(i)))) < Width then
14. addTextBox(curTextLine, DetectedBox(i));
15. preBox= DetectedTextBox(i);
16. else
17. %将上面得到的 curTextLine 的文字线信息加入到 TextLine 之中
18. %初始化并重新 curTextLine
19. addTextLine(TextLine, getLineInfo(curTextLine));
20. curTextLine= Null;
21. addTextBox(curTextLine, DetectedBox(i));
22. preBox= DetectedTextBox(i);
23. end
24. end
25. %将最后一行文字线信息加入

26. TextLine = addTextLine (TextLine, getLineInfo (curTextLine));
27. end
28. return TextLine;



图 7 精检测输出(左)和文字线产生(右)

4.2 分词与梯度投影分析

在得到文字线之后,使用梯度投影分析的方法将同一文字线的不同文字分开。如图 8 所示,对于同一个文字线,对其所有像素分别计算梯度的绝对值,并按列在竖直方向对同一列的所有像素的梯度进行累加,从而得到梯度在文字行水平方向上的分布直方图(同样地,在竖直方向的梯度投影分析也按照类似的方法进行)。这种操作与投影操作类似,所以称之为梯度投影分析。由于文字行中包含文字的位置通常含有较多的重要的边缘像素,因此其梯度累加和一般较大,而文字行中词语的间隔不包含文字边缘,所以其梯度累计和相对较小。利用这一规律,可以将文字线分割成为单独的词语。

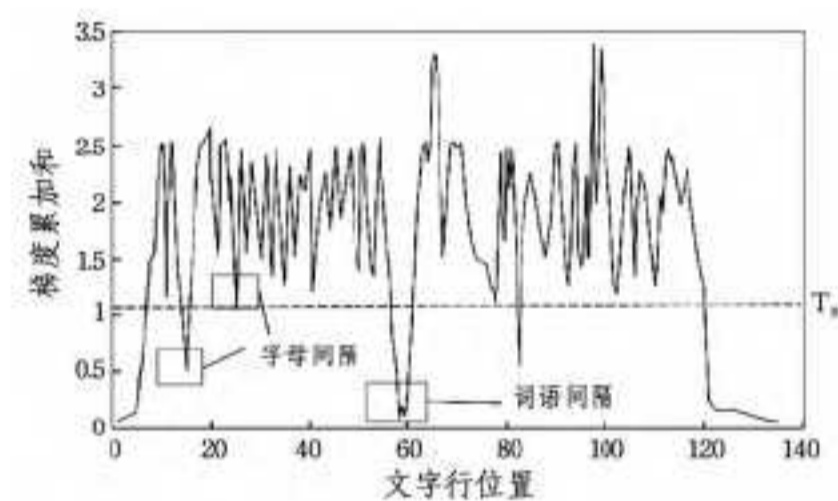


图 8 梯度投影分析

另外,我们注意到,不仅仅词语之间的间隔会在梯度投影分布上产生较小的梯度累加值,一些组成词语的字母之间的间隔也会产生较小的梯度累加值。为了避免字母从词语中被错误地分离出来,根据同一词语的字母间隔一般小于词语之间的间隔的特点,认为在水平方向上间隔小于 5 个像素的前后文字属于同一个词语,再将其重新组合起来。同时,去除掉分词结果中宽度过短的词语片段。分词的结果如图 9 所示。



图 9 分词结果

综上所述,梯度投影分析的算法流程如下。

算法 3 projectAnalysis (TextLine, a, Gradient, Letter-

Thresh, WidthThresh)

1. WordList= Null;
2. if TextLine \neq Null then

```

3. for i=1 to size(TextLine) do
4. % 对梯度在竖直方向上计算投影,计算局部梯度阈值 T
5. curWordRow=getRow(TextLine(i));
6. LocalGrad=getLocalGrad(Gradient,TextLine(i));
7. projGradient=accumulateGrad(LocalGrad',vertical');
8. T=a*mean(projGradient);
9. Flag=False; % 词语生成进行标志 Flag 初始化为 0
10. curWordList=NULL;
11. % 对于当前文字线 TextLine(i) 中的每一个像素进行分析
12. for j=1 to length(TextLine(i)) do
13. if projGradient(j)> then
14. if Flag==False then
15. Flag=True;
16. curWordStart=j;
17. curWordEnd=j
18. else
19. curWordEnd=curWordEnd+1;
20. end
21. elseif Flag==True then
22. % 将词语加入词表中并重新将 Flag 置为 0
23. addWord(curWordList,curWordRow,curWord-
Start,curWordEnd);
24. Flag=False;
25. end
26. end
27. % 将最后一个词加入词表
28. addWord(curWordList,curWordRow,curWordStart,cur-
WordEnd);
29. % 根据 LetterThresh 将相邻的距离很近的字母重新组合
为词语
30. curWordList=regroupWord(curWordList,LetterThresh);
31. % 将其中宽度过短的分词片段去除
32. curWordList=filterWord(curWordList,WidthThresh);
33. addWord(WordList,curWordList);
34. end
35. return WordList;

```

4.3 多分辨率分析

出于检测不同的尺寸大小的文字的需要,对同一幅图片进行多分辨率上的分析,并将在不同的分辨率上检测到的文字进行合并。本文采取了较为简单易行的多分辨率分析策略。

第一步 每个分辨率上使用上文中介绍的文字,线产生方法得到当前情况下的文字,按词的形式进行划分得到文字框: $Box_i^{(R_j)}$, $i=1,2,\dots,N_{R_j}$, $j=1,2,\dots,N_{resolution}$, 其中 N_{R_j} 为在分辨率 R_j 下检测的文字框(词语)数量, $N_{resolution}$ 为多分辨率分析所使用的分辨率的数量。每个文字框 $Box_i^{(R_j)}$ 都记录 4 个值: $Box_i^{(R_j)}(x_i, y_i, height_i, width_i)$, 分别为文字框的中心位置和长宽信息。

第二步 将所有分辨率上的文字框归一化到分辨率为 100% 的原始图片上,其方法为 $Box_{normalized} = Box_i^{(R_j)} / R_j$, 即将原文字框的中心和长宽都按缩放比例 R_j 还原到原来的图片上。这样就得到了不同分辨率下的一系列大小不同的文字框。

第三步 对不同分辨率上的文字框进行合并。如果两个文字框在水平方向上有重合,且在竖直方向上重合的高度超

过较小文字框竖直高度的 50%, 则将两者合并起来。即对于归一化后的 Box_i 和 Box_j , 如果满足:

$$\begin{cases} \text{abs}(y_i - y_j) < \frac{(\text{width}_i + \text{width}_j)}{2} \\ \text{height}_{\text{ave}} = \frac{(\text{height}_i + \text{height}_j)}{2} \\ \text{height}_{\text{ave}} - \text{abs}(x_i - x_j) > r \cdot \min(\text{height}_i, \text{height}_j) \end{cases} \quad (11)$$

其中, $r=0.5$, 那么将 Box_i 和 Box_j 按照图 10 方式合并成为一个新的文字框 Box_{merge} , 并从原来的词表中去除掉已经合并的两个文字框。

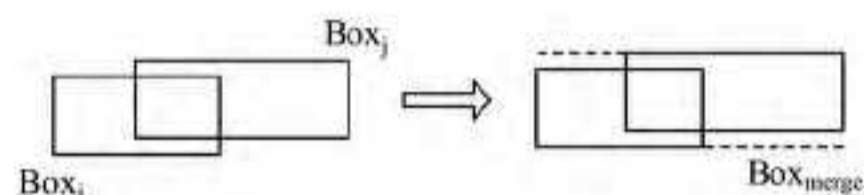


图 10 多分辨率下的文字框合并

在所有的合并完成后,再对合并之后新得到的文字框再次按照类似于上文中的梯度投影分析的方式在竖直和水平方向上分别进行分析从而去除不需要的间隔,对文字框进行进一步的精炼,由此得到最后的检测结果。

综上所述,结合前文的算法 1—算法 3,本文基于多分辨率分析的整个文字检测算法流程描述如下。

算法 4 multiResolutionTextDetection (Image, scalingRatio, mergeThresh)

```

1. overallWord=NULL;
2. for i=1 to size(scalingRatio) do
3. % 在不同的分辨率下进行图像的缩放并求取梯度值
4. curImage=ImgScaling(Img,scalingRatio(i));
5. Gradient=getGrad(curImage);
6. % 当前尺度下的文字粗检测和精检测
7. MAERset=detectMSER(curImage);
8. MSERset=geometricFiltering(MSERset,N,H,W);
9. DetectedText=cascadeELM(MSERset,FeatType,FeatPara,Th-
reshold);
10. % 当前尺度下的文字线生成和梯度投影分析
11. TextLine=TextLineGen(DetectedText,r,Height,Width);
12. WordList=projectAnalysis(TextLine,a,Gradient,Letter-
Thresh,WidthThresh);
13. % 将文字框归一化并加入至
14. WordList=normalizeWord(WordList,scalingRatio(i));
15. addWordList(overallWord,WordList);
16. end
17. % 对不同分辨率下得到的文字框进行合并和精炼
18. mergeFlag=False;
19. Times=0;
20. while(Times==0 or mergeFlag==True)
21. mergeFlag=False;
22. for i=1 to size(overallWord) do
23. For j=1 to size(overallWord) do
24. if i ≠ j then
25. if MergeCondition(overallWord(i),overallWord
(j))==True
26. then
27. newWord=mergeWord(overallWord(i),overallWord(j));
28. addWord(newWord,overallWord);
29. deleteWord(overallWord(i),overallWord);
30. deleteWord(overallWord(j),overallWord);

```

```

31.         mergeFlag = True;
32.         Times = Times + 1;
33.     end
34. end
35. if mergeFlag == True then
36.     break;
37. end
38. end
39. if mergeFlag == True then
40.     break;
41. end
42. end
43. end
44. % 对多分辨率合并后的文字框通过梯度投影分析进行进一步的精炼
45. OverallWord = projAnalysis(overallWord', 'vertical');
46. OverallWord = projAnalysis(overallWord', 'horizatal');
47. return overallWord;

```

根据前文的介绍,本文提出的双阶段图像文字检测方法的总体流程如图 11 所示。



图 11 本文双阶段图像文字检测方法流程

5 实验

利用在文字检测领域最为常用的开放公用数据集 ICDAR 2003 数据集对所提方法进行了训练和测试。ICDAR 2003 数据集包含了一个包含 258 张训练图片的 SceneTrial-Train 图片集和一个包含 251 张测试图片的 SceneTrial-Test 图片集。根据 SceneTrial-Train 提供的图片和 groundtruth(标定好的真实数据),从中提取了 6186 个文字样本(正样本)和 6408 个非文字样本(负样本)用于构成训练数据集。实验程序的运行环境为 MATLAB 2012a, CPU 为 Intel 酷睿 i7 4700HQ,内存为 8GB。

5.1 特征分辨能力对比

本节对图像文字精检验中用到的特征进行了对比。为了能够尽可能公平地反映特征的分辨能力,避免分类器参数调整不当可能带来的影响,使用了一个由 350 个单层决策树弱分类器组成的 Adaboost 分类器。该分类器在每次迭代中挑选出特征向量中使得当前训练样本错误率最低的一维训练出一个单层决策树分类器,在整个训练过程中没有其他需要调整的参数,能够最公平地反映出特征的分辨能力。为了测试特征的泛化能力,将之前提取的样本集随机划分为相等的两个集合,一个集合用于训练 Adaboost 分类器,另一个集合用于测试。

特征对比实验由两个实验组成,第一个实验测试了在不分块条件下的特征分辨能力。 LBP_8^2 、 LBP_8^1 、 $eLBP$ 、 LGP_8^2 、 LGP_8^1 、 DGP_8^1 和 DGP_8^2 被分别从训练样本的整张图片中进行抽取并用于训练分类器,之后在测试集上进行测试;第二个实验测试了在分块的情况下(在图中,分块特征前面加上“bb-”前缀来区分)各个特征的分辨能力。分块抽取的 LBP_8^1 、 LGP_8^1 和 DGP_8^1 被用来和 HOG 以及小波系数特征 wavelet 进

行对比。两个实验的结果分别如图 12、图 13 所示。

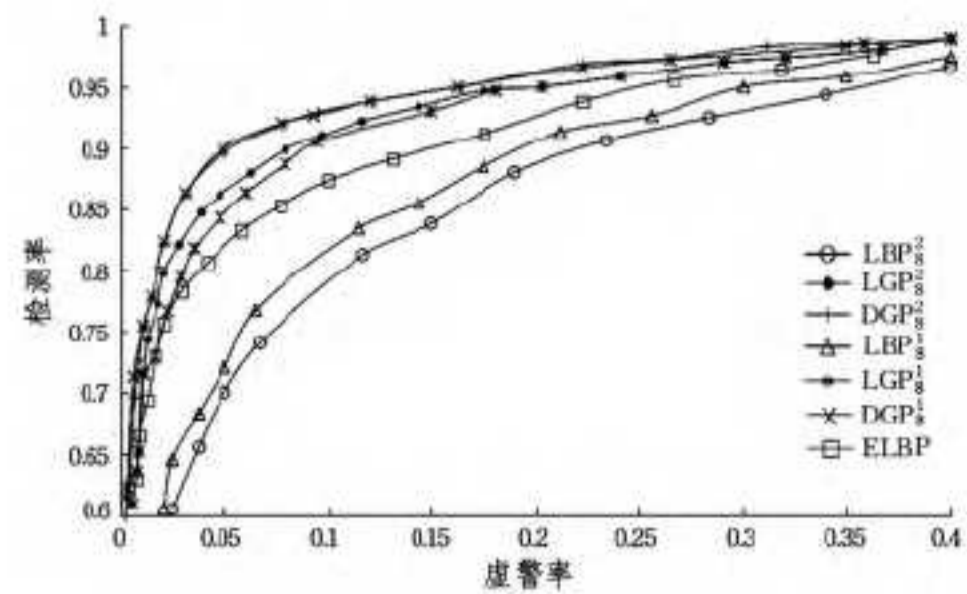


图 12 不分块条件下不同特征的分辨率性能比较

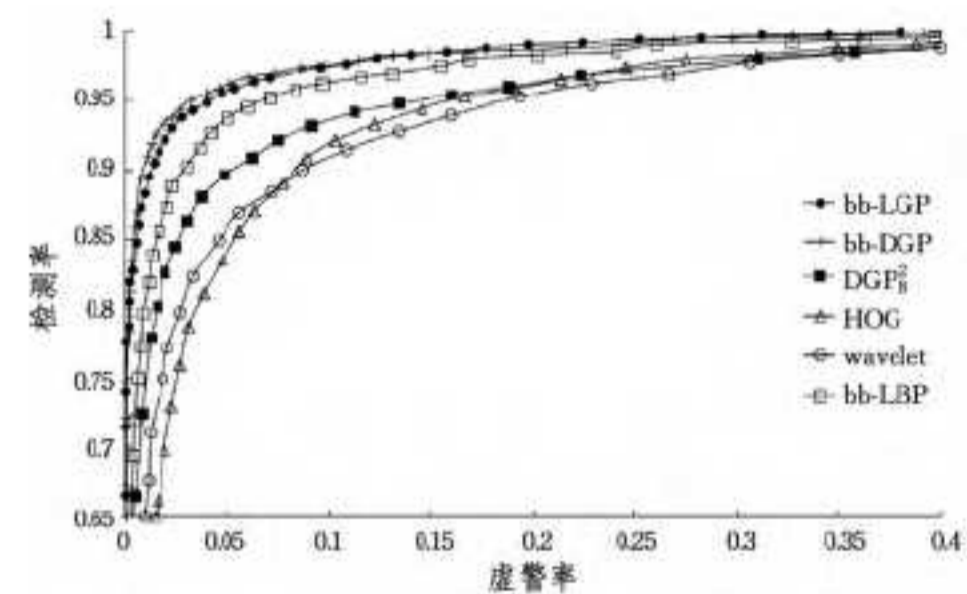


图 13 分块条件下不同特征的分辨率性能比较

在图 12 与图 13 的检测率与虚警率关系曲线中,通过对训练出来的 Adaboost 分类器采取不同的判决门限,从而可以计算出一系列的虚警率和相对应条件下的检测概率。其计算方式如下:

$$\begin{cases} \text{虚警率} = \frac{\text{FalsePositive}}{\text{FalsePositive} + \text{TrueNegative}} \\ \text{检测率} = \frac{\text{TruePositive}}{\text{FalsePositive} + \text{TrueNegative}} \end{cases} \quad (12)$$

其中, TruePositive \ FalsePositive 分别代表正确 \ 错误分类的正样本数, TrueNegative \ FalseNegative 分别代表正确 \ 错误分类的负样本数。

从第一个实验可以清楚地看出,提出的 DGP 在几种特征中拥有最佳的分辨能力。同时,基于梯度进行编码的 DGP 和 LGP 特征要明显优于基于灰度进行编码的 LBP 和 eLBP,在相同的虚警率之下, DGP 和 LGP 能够带来 3%~15% 的检测率提升。另外可以发现,当特征的提取是从整个样本图片中进行而不是分块进行时,半径较大的 LGP_8^2 比半径较小的模式 LGP_8^1 具有更好的分辨能力。通过实验发现,产生这种现象的原因是半径较大的 LGP_8^2 对于文字的整体轮廓具有更好的描述能力(如图 14 所示),其文字边缘的描述模式更不容易与背景的干扰模式相混淆。而 LBP 由于无法很好地解决背景反转和噪声问题, LBP_8^2 并没有展现出和 LGP_8^2 一样优良的轮廓描述能力。同时,我们也发现 DGP_8^1 和 DGP_8^2 几乎具有相同的描述性能,其原因主要在于 DGP 已经很好地消除了背景中的干扰模式问题,所以二者的描述能力非常接近。这也进一步证明了 DGP 的滤除干扰作用和优越性。

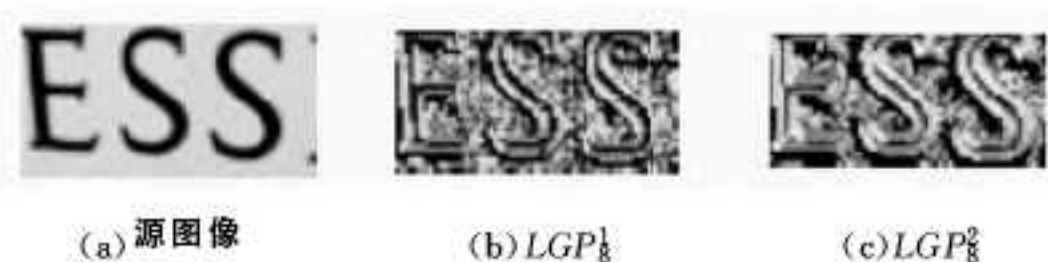


图 14 不同半径模式的描述能力

从第二个实验可以得到如下结论:首先,在分块特征提取

的情况下, DGP 相较于传统的用于文字检测的特征来说仍然具有较大的优势。其次, 对于 LBP_8^1 , LGP_8^1 和 DGP_8^1 , 分块的特征提取能够带来显著的性能提升。在第二个检测率与虚警率关系曲线中给出了在第一个实验中性能最好的 DGP_8^2 作为参照。可以看出虽然不分块的 DGP_8^2 要优于 HOG、小波特征 wavelet 以及分块的 LBP_8^1 特征, 但与分块的 LGP_8^1 , LBP_8^1 和 DGP_8^1 相比仍具有较大的差距。另外, 需要说明的是, 半径较大的 LBP_8^2 , DGP_8^2 和 LGP_8^2 特征在分块的情况下性能有所下降, 主要原因是分块反而不利于这一类特征对于文字的整体轮廓的描述能力。但是利用分块特征提高检测性能也具有一定的代价, 即特征维数的升高和计算特征的时间的增加。因此, 为了在一定程度上缓解这个问题, 在设计检测器时将维数较低或者特征计算相对简单的特征(HOG 和不分块的 DGP_8^2) 放在检测器的前端, 而维数较高或者特征计算相对复杂的特征放在后端, 从而在一定程度上降低检测的时间代价。这也正是级联结构的优势之一。

通过以上的两个实验可以看出, 本文提出的 DGP 确实是一种十分有效的文字区域描述特征。

5.2 分类器性能对比

本节对前面已经介绍的 Adaboost、SVM 以及本文所采用的 ELM 分类器进行对比。在本实验中, 采用分块提取的 LBP_8^1 特征(具有代表性的传统文字检测特征)和分块提取的 DGP_8^1 特征进行分类器的训练和测试。所有分类器的参数都经过了精心的调整以得到最好的性能。在每次实验中, 正样本集和负样本集都被随机划分为数量相等的两部分, 其中一份正样本和负样本被用作训练集训练分类器, 而另外一份作为测试集来得到各个分类器的测试精度。两组实验都重复进行 10 次并将所有结果的平均值作为最后的实验结果。

基于分块提取 LBP_8^1 特征的分类器对比实验结果如表 1 所列。

表 1 分块提取 LBP_8^1 的测试结果

分类器名称	测试正确率(%)	训练时间(s)
Adaboost	94.33	664.31
Basic ELM	93.13	13.02
Kernel-based ELM	92.58	10.78
SVM	95.77	733.61

基于分块提取 DGP_8^1 特征的分类器对比实验结果如表 2 所列。

表 2 分块提取 DGP_8^1 的测试结果

分类器名称	测试正确率(%)	训练时间(s)
Adaboost	95.73	673.27
Basic ELM	95.42	13.90
Kernel-based ELM	95.61	10.53
SVM	96.79	491.54

由表 1 和表 2 可以看出, ELM 将训练分类器所需的时间压缩到传统分类器训练所需时间的几十分之一, 并且这种训练时间上的优势在训练样本的规模进一步地增大时会体现得更加明显。两种 ELM 分类器在分类精度上与 Adaboost 和 SVM 存在略微差距, 但在使用分辨能力较高的特征如 DGP 时, 结果非常接近。如果考虑到 ELM 对于训练时间的大幅压缩和分类器可扩展性的提升, 我们认为这种较小的泛化性能损失是可以接受的。在大数据条件和一些具有较高的实时

性要求的环境下, 这种以微弱的性能损失换取宝贵的实时性和可扩展性、降低训练的难度和复杂度的方法是十分具有实用意义的。另外, 一些分类能力更强的改进型 ELM^[28] 也不断被提出, 未来研究将着眼于在保持 ELM 优秀学习速度的基础上进一步改进 ELM 的性能。

5.3 算法性能比较

为了验证本文提出的基于双门限梯度模式的图像文字检测算法的性能, 在 ICDAR 2003 公开数据集上, 将本文算法与其他文献中的同类算法^[29-32] 进行比较, 结果如表 3 所列。

表 3 不同算法的性能测试结果对比(%)

算法	准确度	召回率	综合性能
本文	81	73	76
文献[29]	77	72	73
文献[30]	74	68	71
文献[31]	64	73	70
文献[32]	72	61	63

根据表 3 测试结果可知, 与其他同类算法相比, 本文提出的基于双门限梯度模式的图像文字检测算法的各方面性能有较大的提高。

结束语 文字检测作为目标检测领域一个十分重要且极具挑战性的问题, 已经引起了越来越多本领域研究者的关注。本文实现了一种具有优良检测性能和较高准确率的双阶段图像文字检测方法, 不仅能以较快速度实现复杂环境下的文字检测, 并借助于 ELM 这一最新的机器学习工具解决了传统的基于机器学习的文字检测方法难以解决的训练时间问题, 拥有极快的学习速度和优秀的分类器扩展性。

参考文献

- [1] Epstein B, Ofek E, Wexler Y. Detecting text in natural scenes with stroke width transform[C]// 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2010:2963-2970
- [2] Li Y, Lu H. Scene text detection via stroke width [C]// 2012 21st International Conference on Pattern Recognition (ICPR). IEEE, 2012:681-684
- [3] Chen X, Yang L, Zhang J, et al. Automatic detection and recognition of signs from natural scenes[J]. IEEE Transactions on Image Processing, 2004, 13(1):87-99
- [4] 陈梓洋, 王宇飞, 钱侃, 等. 自然场景下基于区域检测的文字识别算法[J]. 计算机技术与发展, 2015, 25(7):236-239
- [5] Kim K I, Jung K, Kim J H. Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003, 25(12):1631-1639
- [6] Chen X, Yuille A L. Detecting and reading text in natural scenes [C]// Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2004). IEEE, 2004, 2:366-373
- [7] Pan Y F, Hou X, Liu C L. Robust System to Detect and Localize Texts in Natural Scene Images[C]// The Eighth IAPR International Workshop on Document Analysis Systems (DAS'08). IEEE, 2008:35-42
- [8] E Q, Huang Q, Gao W, et al. Fast and robust text detection in images and video frames [J]. Image and Vision Computing, 2005, 23(6):565-576

- [9] Goto H. Redefining the DCT-based feature for scene text detection[J]. International Journal of Document Analysis and Recognition (IJ DAR), 2008, 11(1): 1-8
- [10] Gomez L, Karatzas D. MSER-based Real-Time Text Detection and Tracking[C]// 2014 22nd International Conference on Pattern Recognition (ICPR). IEEE, 2014: 3110-3115
- [11] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]// IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005). IEEE, 2005, 1: 886-893
- [12] Ojala T, Pietikinen M, Harwood D. A comparative study of texture measures with classification based on featured distributions [J]. Pattern Recognition, 1996, 29(1): 51-59
- [13] Jun B, Kim D. Robust face detection using local gradient patterns and evidence accumulation[J]. Pattern Recognition, 2012, 45(9): 3304-3316
- [14] Cui A C, Asari V K, Braun A D. Robust textural features for real time face recognition[C]// Proc. SPIE9408. Imaging and Multimedia Analytics in a Web and Mobile World 2015. 2015: 940806-940806-13
- [15] 郑英娟, 张有会, 王志巍, 等. 基于八方向 Sobel 算子的边缘检测算法[J]. 计算机科学, 2013, 40(11A): 354-356
- [16] Shen D H, Zhang L C, Xu E. An improved edge detection algorithm based on Sobel operator [J]. Information Technology, 2015, 347-350(4): 3541-3545
- [17] Tian Q C, Zhao X L, Wu X J, et al. Iris Classifier Enhanced Algorithm Based on AdaBoost[M]// Advanced Intelligent Computing Theories and Applications. With Aspects of Contemporary Intelligent Computing Techniques. Springer Berlin Heidelberg, 2007: 1001-1009
- [18] Mori M, Uchida S, Sakano H. Global feature for online character recognition[J]. Pattern Recognition Letters, 2014, 35(1): 142-148
- [19] Sok P, Taing N. Support Vector Machine (SVM) based classifier for Khmer Printed Character-set Recognition[C]// Asia-Pacific Signal and Information Processing Association, 2014 Annual Summit and Conference (APSIPA). IEEE, 2014: 1-9
- [20] Huang G B, Wang D H, Lan Y. Extreme learning machines: a survey[J]. International Journal of Machine Learning and Cybernetics, 2011, 2(2): 107-122
- [21] Iosifidis A, Tefas A, Ioannis Pitas[J]. Pattern Recognition Letters, 2015, 54: 11-17
- [22] Jung K, In Kim K, K Jain A. Text information extraction in images and video: a survey[J]. Pattern Recognition, 2004, 37(5): 977-997
- [23] 何淑琳, 张雪英, 孙颖, 等. 基于极限学习机的语音情感识别[J]. 微电子学与计算机, 2015, 7(7): 50-54
- [24] 卢诚波, 梅颖. 前馈网络的一种高精度鲁棒在线贯序学习算法[J]. 上海交通大学学报, 2015, 49(8): 1137-1143
- [25] Iosifidis A, Tefas A, Pitas I. On the kernel Extreme Learning Machine classifier[J]. Pattern Recognition Letters, 2015, 54: 11-17
- [26] Cao L L, Huang W B, Sun F C. Optimization-Based Extreme Learning Machine with Multi-kernel Learning Approach for Classification[C]// 2014 22nd International Conference on Pattern Recognition (ICPR). IEEE Computer Society, 2014: 3564-3569
- [27] Bandarabadi M, Dourado A, Teixeira C A, et al. Seizure prediction with bipolar spectral power features using Adaboost and SVM classifiers[C]// Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. 2013: 6305-6308
- [28] 李小冬. 核极限学习机的理论与算法及其在图像处理中的应用[D]. 杭州: 浙江大学, 2014
- [29] 杨彬, 夏思宇. 自然场景多方向文本检测方法[J]. 华中科技大学学报(自然科学版), 2015(S1): 228-232
- [30] Yao Cong, Bai Xiang, Liu Wen-yu, et al. Detecting texts of arbitrary orientations in natural images[C]// Proc of IEEE Conference on computer Vision and Pattern Recognition. Rhode Island: Curran Associates, 2012: 1083-1090
- [31] Kang L, Li Y, Doermann D. Orientation Robust Text Line Detection in Natural Images[C]// 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2014: 4034-4041
- [32] Epshtein B, Ofek E, Wexler Y. Detecting text in natural scenes with stroke width transform[C]// 2013 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2010: 2963-2970

(上接第 129 页)

- [7] Bellon O R, Silva L. New improvements to range image segmentation by edge detection[J]. Signal Processing Letters, IEEE, 2002, 9(2): 43-45
- [8] Biswas K, Basu S K. Gesture Recognition using Microsoft Kinect [C]// 2011 5th International Conference on Automation, Robotics and Applications (ICARA). 2011: 100-103
- [9] Raheja J L, Chaudhary A, Singal K. Tracking of fingertips and centers of palm using kinect[C]// 2011 Third International Conference on Computational Intelligence, Modelling and Simulation (CIMSIM). 2011: 248-252
- [10] 汤勇, 顾宏斌, 周末. 交互系统中实时手势分割及指尖检测方法[J]. 光电工程, 2010, 37(7): 145-150
- [11] 何小磊. 基于深度信息和彩色图像的手势识别[D]. 武汉: 华中科技大学, 2012
- [12] 杨全, 彭进业. 采用 SIFT-BoW 和深度图像信息的中国手语识别研究[J]. 计算机科学, 2014, 41(2): 302-307
- [13] 陈子毫. 基于深度信息的手势检测与跟踪[D]. 广东: 华南理工大学, 2012
- [14] 邓瑞, 周玲玲, 应忍冬. 基于 Kinect 深度信息的手势提取与识别研究[J]. 计算机应用研究, 2013, 30(4): 1263-1265
- [15] 王法胜, 鲁明羽, 赵清杰, 等. 粒子滤波算法[J]. 计算机学报, 2014(8): 1679-94
- [16] Swain M J. Interactive Indexing Into Image Databases [J]. Storage & Retrieval for Image & Video Databases, 1993, 1908: 95-103
- [17] Barla A, Odone F, Verri A. Histogram intersection kernel for image classification[C]// 2003 International Conference on Image Processing. 2003, III(2): 513-516
- [18] 胡士强, 敬忠良. 粒子滤波原理及其应用[M]. 北京: 科学出版社, 2010
- [19] 朱志宇. 粒子滤波算法及其应用[M]. 北京: 科学出版社, 2010