# 第四章 复杂条件下的文本识别

## 4.1 引言

文本识别的研究是从光学字符识别开始，光学字符识别经过了几十年的发展到目前已经有各种各样的高识别率的产品出现了。目前文本识别的难点在于对复杂条件下的文本进行识别。复杂条件又包括两个方面，一是低质量的光学字符文档，这些文档通常都存在着低分辨率，污染严重，字符粘连和断裂的情况常见。二是自然场景条件下的文本识别，自然场景条件下的文本存在多种不同的形态，文本受到的干扰比较严重如光照、遮挡等。若把文本中的单个字符看成是场景的某目标，那么一些有效的目标检测和识别的算法和框架都可以利用在场景文本识别当中。当然场景文本识别也存在着区别于一般目标识别的特殊之处如文本字符的类别数目众多并且容易相互混肴。居于复杂条件下的文本识别存在的这些难点，本章重点研究了自然场景下的文本识别，从词图像分割、字符特征提取到分类器选择，参数设置等各个环节进行了探讨。在词图像分割阶段本章提出了将定位的中间结果和词图像分割进行结合，摒弃了传统的二值化流程。在特征提取阶段，结合了传统光学字符识别特征提取和现代的目标识别的特征提取，并在输入分类器之前和之后进行结合。本章最后在标准数据集ICDAR2003文本识别数据上进行测试，验证了方法的有效性。

## 4.2 定位和识别的关系探讨

在文本识别之前，我们先探讨一下文本定位和识别的关系，目前的场景文本识别的研究也都是分为这两种框架，一是有明显的定位过程，随后定位的结果送入到光学字符识别模块中，这种框架精力集中在怎么提高定位的结果，而识别是完全依赖于现有的光学字符识别软件。二是没有明显的定位过程，定位和识别同步进行都是属于目标识别的框架中，在判断字符是否是字符时同时也进行分类。两种方法各有优势，但是目前的研究趋向于后面一种框架。下面针对两个问题具体分析一下文本定位和识别的关系。

1）是否需要明显的定位过程

在2010年Kai Wang【】发表端到端的文本识别这一论文后，接下来的几年里很多论文都是建立在这一基础之上，这些论文提出的方法都没有明显的整块文本定位步骤，一般是采用多尺度滑动窗口在原图上进行操作，对于每个窗口提取HOG特征，然后送入SVM分类器进行分类识别，这样明显会引入很多的非字符窗口。随后一般都采用先验信息结合分类器的结果来滤除部分非字符窗口。很明显，多尺度滑动窗口增加了问题的复杂性，场景文本相较于场景目标有相同地方但是还存在着很大区别。场景目标如车，建筑，人大部分情况下都呈现分散的状态，而目标和目标之间的差异都比较大，目标间的排列也没有很大的规律，目标间的距离差异都很大。场景文本中的字符则不同，相互之间的差异不是很大，一个类的部分跟另一个类相似性很高，场景文本中的字符大部分都聚集在一个区块，字符之间的间隔都不是很大。基于字符都聚集在一个文本区块里，我们可以利用简单的整体特性先确定一个大概的文本范围而不是利用滑动窗口在整张图像里进行搜索判断。综上所述，虽然场景文本的字符和场景目标有一定的相似性，但是滑动窗口的框架并不适合于场景文本，一个简单有效的文本定位过程对于精简框架，减少识别过程的复杂度，提高字符识别的正确率都有很大的帮助。

2）文本定位是否完全独立于文本识别

另一种文本定位结合常见的光学字符识别模块的方式，则完全专注于文本定位过程，输入整张场景文本图像后，只要输出文本区域的位置信息就可以，剩下的识别任务都依赖于光学字符识别模块，那么最后的识别结果就会受限于识别模块。分析文本定位的过程，我们会发现定位过程的中间结果如二值化模板包含丰富的信息可以利用在词图像分割和特征提取等模块里。这一前向过程是文本定位提供信息给予后续的识别模块，但同样的识别模块也可以提供反馈给定位模块，在定位过程中可以适当的降低准确率，提高召回率，利用分类器的输出分数来判断区块是否为文本近一步提高正确率。

文本定位和文本识别的关系可以在图【】中体现出来，

二值化模板

文本识别

文本定位

分类器分数

## 4.3 文本识别系统设计

本章提出的文本定位系统如图【】所示，整个系统分为两个阶段，训练和测试阶段。在训练阶段，分别提取训练集的局部和全局特征，采用两种方式来训练SVM模型，一是两种特征直接拼接成一个长特征来训练，二是分别用来种特征来训练模型，在图【】中只展示了第二种。在测试阶段利用定位过程得到的二值化图像来进行分割，分割方式为基于水平投影的方式。在得到词图像分割的位置信息后，分别在单个字符模板上提取全局特征和在初始的彩色的单个分割后的字符上提取局部特征。同测试阶段，采用两种方式来对提取的特征来进行分类识别，最后根据分类器的输出分数来得到分类标签。

## 4.4 词图像分割

词图像分割是整个识别过程中十分重要的模块，分割错误通常是文本识别过程中的瓶颈，尤其对于自然场景的文本而言。如果是对定位过程的结果彩色的词图像进行分割，那么一般要进行二值化，然后在二值化图像上进行分割，因为通常分割不需要太多的细节信息，只需要保持字符间相对位置信息即可。通常的二值化方法对于场景文本图像各种复杂的情况都不能很好的处理，这时我们就要寻找另一种更好的保留位置信息的方法。而在定位阶段，我们采用两种方法来进行候选连通域提取的时候都生成了对应二值化的图像，从图【】可以看出这些二值化图像是十分适合于词分割的。在实验中分别采用两种方式来进行词分割对比，一个简单的基于垂直投影的分割，而是采用滑动窗口结合分类器的方式

1）基于垂直投影的分割

在定位阶段，我们已经看到场景中的大部分文本都是以水平文本行的形式存在的。虽然有一些文本行存在角度倾斜，在经过第一章的倾斜校正预处理之后，文本都能够调整到水平的形式。由于文本的字符之间存在着间隔，垂直投影是最直观，最简单的分割方式。

2）基于滑动窗口的分割

基于投影分割简单但是鲁棒性也较差，如果字符间存在粘连或者断裂的情况，基于投影的分割就会造成将一个字符分割成两个或者两个字符分割成一个。滑动窗口则采用不同的策略，它相当于搜索字符之间所有可能的位置，在实验中我们是以步长为1像素左右移动的，这里不需要上下移动，因为一般文本都是一行一行存在的，而为了处理存在多个不同大小的字符，我们利用多个尺度的窗口来进行滑动。经过滑动处理后，得到一系列的候选窗口，这些候选窗口中存在很大部分的非字符窗口。

## 4.5 特征提取

特征提取

### 4.5.1 全局特征

要实现弱小目标的跟踪，需要对每一幅图像及其之前的图像序列做上述奇异值分解。设当前要处理的图像为，则取前幅图像序列与当前图像构成幅图像序列。

### 4.5.2 局部特征

要实现弱小目标的跟踪，需要对每一幅图像及其之前的图像序列做上述奇异值分解。设当前要处理的图像为，则取前幅图像序列与当前图像构成幅图像序列。

## 4.6 分类器选择及参数设置

要实现弱小目标的跟踪，需要对每一幅图像及其之前的图像序列做上述奇异值分解。设当前要处理的图像为，则取前幅图像序列与当前图像构成幅图像序列。

### 4.6.1 K最近邻算法

要实现弱小目标的跟踪，需要对每一幅图像及其之前的图像序列做上述奇异值分解。设当前要处理的图像为，则取前幅图像序列与当前图像构成幅图像序列。

### 4.6.2 BP神经网络

要实现弱小目标的跟踪，需要对每一幅图像及其之前的图像序列做上述奇异值分解。设当前要处理的图像为，则取前幅图像序列与当前图像构成幅图像序列。

### 4.6.3 支持向量机

要实现弱小目标的跟踪，需要对每一幅图像及其之前的图像序列做上述奇异值分解。设当前要处理的图像为，则取前幅图像序列与当前图像构成幅图像序列。

## 4.7 特征融合

要实现弱小目标的跟踪，需要对每一幅图像及其之前的图像序列做上述奇异值分解。设当前要处理的图像为，则取前幅图像序列与当前图像构成幅图像序列。

### 4.7.1 前期融合

要实现弱小目标的跟踪，需要对每一幅图像及其之前的图像序列做上述奇异值分解。设当前要处理的图像为，则取前幅图像序列与当前图像构成幅图像序列。

### 4.7.2 后期融合

要实现弱小目标的跟踪，需要对每一幅图像及其之前的图像序列做上述奇异值分解。设当前要处理的图像为，则取前幅图像序列与当前图像构成幅图像序列。

## 4.8 实验与结果分析

本章采用第二章所提出的评价准则对算法性能进行分析。

1) 局部信噪比

 (4.6)

其中，、分别是图像的目标平均灰度和目标周围平均灰度。是目标邻域图像与噪声标准差。局部信噪比表现目标与背景噪声的差别的大小，值越大表示目标与噪声的差别越大，目标越明显。图4.6显示了5组序列处理前后平均局部信噪比，虽然有个别帧处理后图像信噪比要低于处理前图像，不过就整体而言，算法确实提高了图像的局部信噪比。

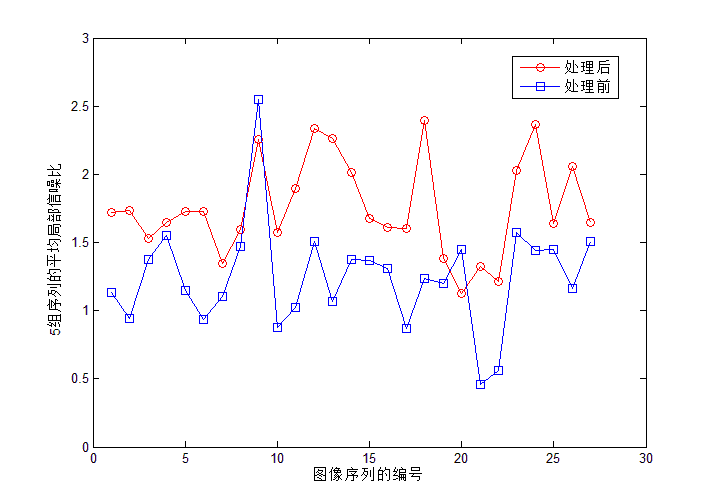


图4.6 图像局部信噪比

2) 局部对比度

 (4.7)

局部对比度表现目标与背景的相对灰度。对比度值越大，则目标越清晰，反之，目标越模糊。图4.7显示了5组序列处理前后平均局对比度，可以看出，处理后每组图像局部目标对比度都要高于处理前的图像，证明算法能够达到增强图像的目的。

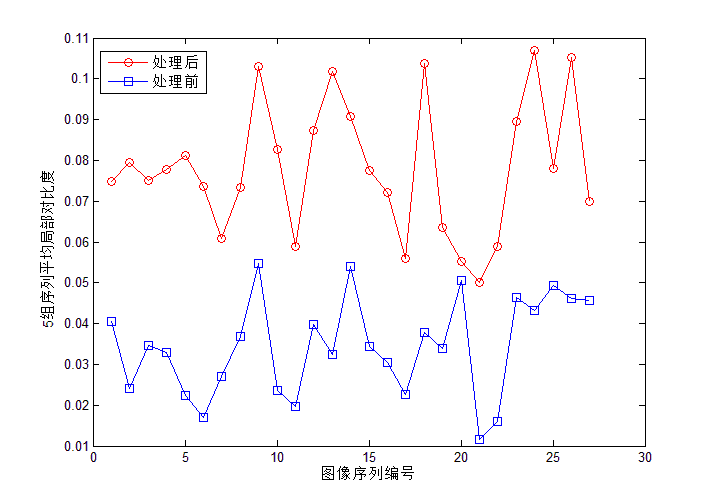


图4.7 图像局部对比度

3) 检测概率及虚警率和平均虚警率

{, #1}检测概率和虚警率和平均虚警率是小目标检测常用评价标准，定义为：

 (4.8)

 (4.9)

 (4.10)

式中：为序列中检测到目标个数；为序列中实际目标个数；为序列中虚假目标个数，为序列帧数。实验用2组图像序列进行分析，分别使用SVD分解、最大中值滤波(MaxMedian)、最大均值滤波(MaxMean)，并对结果进行分析。表4-1和表4-2分别为最大均值与最大中值同文中算法的比较结果：

表4-1 文中算法(SVD)和MaxMean算法的检测结果比较

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | SVD | | | MaxMean | | |
|  |  |  |  |  |  |
| Seq1 | 0.9 | 0.09 | 0.1 | 0.7 | 0.3 | 0.8 |
| Seq2 | 0.95 | 0.04 | 0.05 | 0.5 | 0.32 | 0.95 |

表4-2 文中算法(SVD)和MaxMedian算法的检测结果比较

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | SVD | | | MaxMedian | | |
|  |  |  |  |  |  |
| Seq1 | 0.9 | 0.09 | 0.1 | 0.46 | 0.43 | 1.65 |
| Seq2 | 0.95 | 0.04 | 0.05 | 0.5 | 0.34 | 1.05 |

从实验结果我们可以看出，SVD处理能达到是目标突出的目的，结合帧间与帧内的联合检测能是图像序列的检测率始终保持在90%左右。实验运行软件环境Windows7+Matlab2010，硬件环境CPU主频为2.4G，内存为2G，处理每帧图像平均耗时60ms左右，实验结果显示算法检测率高、鲁棒性高、实时性较好。

## 4.9 本章小结

本文提出的基于奇异值分解的红外弱小目标检测方法，首先将多帧图像进行奇异值分解，再按照合适的特征对图像序列进行重构，即能得到在极微弱信噪比下增强的弱小目标，在采用简单阈值分割就可以自适应的对目标进行检测。同时结合帧间相关与帧内相关的修正算法，实现了对红外弱小目标的准确跟踪。实验效果显示，该算法能够有效在极微弱信噪比下进行目标检测，显示了该算法的有效性。该算法对每帧处理时间很短，显示了该算法的实时性。