# 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，二是采用滑动窗口结合分类器的方式。

### 4.4.1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。

### 4.4.2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。

### 4.4.3基于活动轮廓模型的分割

## 4.5 特征提取

图像在计算机中的表示是以离散的方式存在的，即一系列空间位置上的像素值。如果要对图像进行分类，最简单直接的方式是将图像的像素点拉成一整行特征，然后将这一特征送入分类器中进行分类。但是同一目标的图像像素点容易发生变化，仅仅只有像素点无法表征目标的内在特性，那么就需要找出反应目标内在特性的一些像素点，然后进一步的用数据来表征这些点。特征提取也是识别系统中的关键部分，特征提取的目的是为了能够让相似的目标间相差不大，而让不同目标间的差异较大。针对本章的具体目标提取文本字符的特征，我们是从粗到细提取能表征字符特性的特征。

### 4.5.1 全局特征

在传统的光学字符识别任务里，一般提取特征针对的字符二值化后的图像。字符二值化后精简了信息，减少了噪声，但同时保存了字符的外形和轮廓特征。在实验中全局特征是在二值化后的图像上提取的，本章从粗到细提取了三种全局特征。

1）投影特征

将字符归一化到的大小，对于字符上下左右四个方向进行投影，所用的字符是黑底白字，那么投影就是计算各个方向上前景像素点的分布。这一分布反应字符的外围特性，只能粗略的区分开一些简单的字符。投影特征的提取见图【】，这样字符就可以表示为32维的特征。

2）网格特征

字符的网格特征是先将字符图像归一化为的大小，然后将字符平均的分割出35个大小相等的网格，每个网格中的像素个数为100，最后统计一个网格中前景像素点所占的比例。网格特征的提取见图【】，字符就能够被表示成35 维的特征向量。相对应于投影特征在上下左右四个方向上对前景点进行统计，网格特征在更加细粒度上进行特征提取。

3）轮廓特征

### 4.5.2 局部特征

全局特征描述了整幅图像的一些特性虽然具有一定的区分力，但是如果目标存在着光照不均，遮挡，旋转变换等多种复杂的情况，那么全局特征的效果就会下降很多。在光学字符识别当中，在文档图像较为简单的情况，利用全局特征可以取得较好的效果，而对于遭受污染，文档分辨率低等复杂情况下，全局特征无法很好的表征字符图像。另外一个有挑战性的情况是自然场景中的文本识别，在自然场景中，环境因素如光照，遮挡很难控制，拍摄的角度，远近具有随意性，要能够提取应对这些情况的特征是有一定的难度。幸运的是，目前对于图像局部特征提取算法的研究已经卓有成效，许多鲁棒性的特征提取算法都能够利用在场景文本字符识别当中。局部特征首先找出图像中的关键点，这些关键点一般存在目标的边缘，拐角等梯度或者像素变化明显的地方。确定了图像的关键点之后，需要描述子来描述这些关键点，描述子的计算考虑的是关键点的邻域像素梯度方向和大小的分布，经过不同的统计方式，图像就能表征为具有强分辨的特征描述子。下面介绍在目标识别当中应用广泛的局部特征提取算法

1）SIFT特征

为了能够实现可靠的识别，图像特征要能够在各种变化如图像的尺度，噪声和光照都能够提取到。SIFT 特征提取的流程包括特征点的提取和特征点的表示，其中特征点的提取主要包括构建高斯差分尺度空间，找到高斯尺度空间的极值点，去除不好的特征点和特征点的表示。构建高斯差分尺度空间，构建的方式见式【】



其中代表的是构建后的尺度空间，代表的是尺度可变高斯函数， 指尺度，尺度越大表示高斯函数平滑的区域越大，对应的是图像中的概貌特征，反之平滑的区域越小，对应图像的细节特征，表示的是待提取特征的图像。构建好尺度空间后，就要在空间里寻找特征点，特征点的寻找是通过在本层及上下两层尺度和图像域中寻找极值点而得到的。在得到特征点之后需要用一些参数来表示特征点，这些参数包括特征点的坐标，尺度和方向。同时对每个特征点，我们需要用一些特征向量来表征，SIFT是通过对特征点邻域的区块统计梯度直方图而得，最终每个特征点可以表示成128位的特征向量。

2）HOG特征

HOG类似于SIFT也是通过统计像素的梯度直方图而得到。HOG的思想是局部目标的外形和形状可以由亮度梯度或者边缘方向的分布来描述。这些描述子可以通过将图像分割成小的区域，对于每个区域统计区域内像素的梯度方向直方图。为了克服光照和阴影的变化，HOG通过将几个区域合并成一个块，然后以块的大小滑动图片统计块中的梯度直方图。在实验中，所有的图像都归一化的的大小，然后利用式【】的两个模板来获得梯度图像

然后把图像分割成一个个区域，每个区域包含像素，对于每个区域的像素梯度用9个收集器来进行直方图统计，然后将4个区域的直方图组成一个块，块以8的步长上下左右滑动统计，最后得到1296维的特征向量。

3）Shape Context特征

形状上下文（Shape Context）是为了提出一种描述形状的特征，这个特征可以用来计算形状之间的相似性形状上下文的计算方法是对于某个特定点，其余点与之的分布情况，所以其能够提供全局的区分能力较强的特征。计算一幅图像的形状上下文先将图像的灰度分布看成是海拔，计算图像某个高度的等高线，从等高线中取样100个点，对每一点求shape context，即以某点为log\_polar 极坐标的中心，计算其余点到中心的距离对数和与x轴正半轴的夹角，这样就得到99\*2个特征，同样的利用直方图将特征的数量减少，用12个角度bin和5个距离bin。最后的特征数为100\*12\*5。

## 4.6 分类器选择及参数设置

在机器学习中，分类器作用是在标记好类别的训练数据基础上判断一个新的观察样本所属的类别。分类器依据学习的方式可以分为非监督学习和监督学习，非监督学习顾名思义指的是给予分类器学习的样本没有相对应类别标签，主要是解决寻找未标记数据中的隐藏结构，在第三章中的颜色聚类中运用了不同的非监督学习方式。在这一章中主要利用的是监督学习，监督学习通过标记的训练数据推断出分类函数，分类函数可以用来将新样本映射到对应的标签。在监督学习方式中，每个训练样本都是一对包括训练样本的特征和相对应的标签。监督学习的流程包括确定训练样本的类型，收集训练样本集，确定学习函数的输入特征表示，确定学习函数的结构和对应的学习算法，完成整个训练模块设计，评估分类器的正确率。流程的前面几个部分在上面几节已经介绍，这一节的目的是分类器的选取和参数的设置。目前有各种的监督学习方式可以使用，每一种都有对应的优缺点，没有一种学习算法可以在任何监督学习问题上都能够取到好的效果。我们可以依据下面四个要点来选择合适的分类器，

（1）、泛化能力和拟合之间的权衡

过拟合评估的是分类器在训练样本上的性能，如果一个分类器在训练样本上的正确率很高，说明分类器能够很好的拟合训练数据，但是一个很好拟合训练数据的分类器就存在着很大的偏置，所以在测试数据上不一定能够得到好的效果。如果一个分类器在训练数据能够得到很好效果但是在测试数据上效果下降严重，说明分类器过拟合了训练数据。从另一个方面分析，若分类器在测试数据上能够取得好效果，那么说明分类器的泛化能力强。分类器的泛化和拟合是一个此消彼长的过程，泛化能力强的分类器拟合能力一般很弱，另外则反之。所以分类器需要在泛化能力和拟合能力间取得权衡。

（2）、分类函数的复杂度和训练数据的大小

训练数据的大小对于分类器的选择也是至关重要的，如果是一个简单的分类问题，那么拟合能力强泛化能力弱的分类器就可以通过很小的一部分训练数据来得到。反之，如果是一个复杂的分类问题，那么分类器学习就需要大量的训练数据和泛化能力强的学习算法。一个好的分类器应该能够根据问题的复杂度和训练数据的大小自动的调整拟合能力和泛化能力之间的权衡。

（3）、输入的特征空间的维数

如果输入特征空间的向量维数很高的话，就会造成分类问题变的复杂，即使最后的分类函数仅仅就靠几个特征来决定的。这是因为过高的特征维数会混淆学习算法并且导致分类器的泛化能力过强，而泛化能力过强会使得分类器变化太大，性能下降。因此，一般高维特征向量输入的分类器都需要调节参数使其泛化能力较弱而拟合能力强。另外在实验中，也可以通过从输入数据中去除不相干的特征或者降低特征维数来提高分类器的性能。

（4）输入的特征向量之间的均一性和相互之间的关系

如何特征向量包含多种类型的数据（如离散，连续），许多分类器如SVM，线性回归，逻辑回归就不适用。这些分类器要求输入的特征必须是数字而且要归一化到相似的范围内如之间。而像K最近邻算法和高斯核的SVM这些使用距离函数的分类器对于数据的均一性更加敏感。但是另一种分类器决策树却能够处理这些不均一的数据。如果有多个输入特征向量，每个特征向量之间相互独立即当前特征向量的分类器输出仅仅和当前的特征向量输入有关，那么最好选择那些基于线性函数和距离函数的分类器如线性回归，SVM，朴素贝叶斯。反之，如果特征向量之间存在复杂的相互关系，那么决策树和神经网络更加适合于这类问题。

综上所述，分类器的选择是一个很复杂的问题，涉及到识别的流程的各个模块，下面我们就介绍几种常见的分类器并利用上一节提取到的全局和局部特征来进行测试。

### 4.6.1 K最近邻算法

KNN（k-nearest neighbors）是所有分类器中最为简单的一种，分类器的输出是由样本周围K个最近邻中类别占比最大的那个类决定的，当K=1时，测试样本的标签就由最近的那个类别决定，这里最近的度量一般是采用欧式距离。KNN分类的流程包括两个阶段，在训练阶段，分类器仅仅存储训练样本（包括训练的特征和对应的标签类别）。在测试阶段，可以通过指定K值将测试数据分配给最靠近测试数据的K个样本中出现次数最多的样本标签。KNN需要设置K的值，一般K值越大，分类器抗噪声性能越强，但也不易区分两个类别。在场景文本识别实验中，我们指定K值为1。

### 4.6.2 朴素贝叶斯

朴素贝叶斯是基于贝叶斯定理的简单概率分类器，这一分类器构造是基于输入的特征向量之间是相互独立的前提下。朴素贝叶斯分类器原理的核心是贝叶斯定理。如下式



其中 代表的是输入的 个特征向量，代表的是样本的类别。这一定理是将..：分类器的输出即在输入为某一特征，分给某个类的条件概率转化为求解：类别的先验概率和：确定为某一类的条件下，是某一特征的概率以及：量化常数（仅依赖于特征向量）。依据特征向量之间是相互独立的，可以将式【】转化为求解式【】

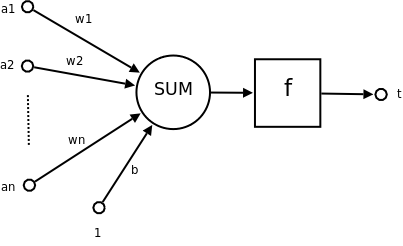


其中是一个值，可以通过训练样本中各个类所占的样本比例来计算。其中通过训练集中样本特征的分布情况来估计可以分为参数估计和非参数估计。在实验中，类似。。。。。

实验待添加

### 4.6.3 BP神经网络

人工神经网络是基于生物神经网络提出的，在生物神经网络中，基本的处理单元神经元相互连接组成一个神经网络。一个基本的人工神经元如下图



可以用数学公式表示为

其中为神经元的输出，为输入向量，为各个输入的权重，为偏置， 为传递函数。由上式可以得到单个神经元的作用是将输入向量和权重内积之后，经过一个传递函数，得到一个标量结果。神经网络则是由很多个这样的神经元组成的，神经元与神经元之间按一定规则连接，并且权重向量是可以根据情况进行更新的。目前有各种类型的神经网络结构被提出，在本章中采用的神经网络是BP神经网络。BP神经网络区别于其他神经网络的特点是信号前向传输，误差反向传输。BP神经网络的训练包含两个过程，第一个阶段是根据输入的特征向量从第一层到最后一层计算神经元的输出。第二个阶段从根据最后一层的误差不断向前调整神经元的权重和阈值。在对BP神经网络训练前，先确定BP神经网络的参数，在场景文本识别实验中，我们创建一个三层的神经网络，输出层为35维的网格特征，隐藏层为25个神经元，输出层为62，训练算法采用的弹性梯度下降法。

### 4.6.4 支持向量机

SVM在最近几年被广泛用于与目标检测和识别当中，SVM是建立在统计学习和结构风险最小的理论上而且被证明比一般的分类器效果要好些。SVM是构建一个模型来模拟现实世界的分类问题，目的是为了使模型和现实世界的差距越小越好。这个模型由已有的一些样本点中学习得来的。所以样本点的好坏和对样本点的处理决定了学习的效果。样本点的好坏需要在数据采集时候的注意的，而对样本点的处理就由模型来决定。SVM能够处理小样本，高维数。模型和现实世界的差距在分类问题中体现为使错误分类的次数最小。

给定一个训练集，，其中是 维的特征向量，的值为1或-1，相对应于的 标签。分类器的目标是寻找最大化几何间隔的超平面，这一平面能够把训练集分成两类，一类的标签为1，另一类的标签为-1。在实际中，SVM解决的是式【】的优化问题



上式受限于



其中为惩罚函数，为松弛变量，为核函数。引入核函数的目的是为了解决线性不可分的问题，核函数能够将特征向量映射到高维，在高维空间里特征向量就能够线性可分。SVM里提供基本的四种核函数

* 线性核：  
* 多项式核： 
* 高斯核： 
* S型核： 

一般来说，最常使用的核函数是高斯核，这一核能够把样本非线性的映射到高维空间，所以其能够处理特征向量和类别标签不是线性关系的情况。线性核是高斯核一种特例，对于特征维数特别大的情况，用线性核的效果要好于高斯核，因为特征维数很大，那么就没有必要把特征映射到更高维空间去。

另外两个参数和是为了解决样本倾斜和离群点的问题。首先是离群点的问题，假设训练样本映射到高维空间后，大部分样本都能够用超平面来进行分类，但是有一两个样本无法分离。那么SVM用的策略是给这些点对应一些松弛变量，这样原始的硬间隔就成了软间隔了，然后用一个惩罚参数来衡量对离群点的重视程度。对于样本倾斜的问题，比如收集样本时某一类的训练样本比另一类大很多，那么数量少的样本赋予更大的惩罚参数，表明我们重视这一部分样本。在实验中，如果是采用高斯核那么需要设置两个参数和 ，这两个参数的设定是通过网格搜索来得到的，先确定一个范围，再确定一定的步长，然后将搜索空间划分成网格，网格上的点对应一个参数对，选择相应的参数对在训练集中用交叉验证，最后选出使训练集交叉验证正确率最高的参数对作为测试的参数。

SVM只能用来解决二分类的问题，但是在场景文本识别中，需要分类的类别数目为62类，因此我们需要采用不同的方法来使用SVM进行多分类。一般有两种策略可供选择，第一种策略是一对多的策略即将某一个类的样本指定为正类，其余的样本通通都为负类，这样只要训练出62个分类器即可。很明显第一种策略会人为的造成样本倾斜，假设每个类的样本数目都是一样的，那么负样本就比正样本多了62倍了。第二种策略采用的是一对一的方法即对类别中的任意两个类训练出一个分类器，假设有个类需要分类，那么就需要训练出个分类器来分类。在实验中采用的是第二种策略来进行多分类的。

## 4.7 特征融合

在特征提取提取阶段，我们分别提取了两类特征局部和全局特征，每一类又都分别提取了几种特征。这些特征之间的组合是决定最后正确率很关键的一个因素，我们分别实施了两种不同特征组合方式并进行评估。按照特征组合是在分类器之前还是之后，可以分为前期融合和后期融合

1）、前期融合

前期融合是指在进入分类器之前就将不同的特征组合在一起，特征组合的方式可以是直接将特征向量拼接在一起或者是加上权重后连接在一起。

2）、后期融合

后期融合则指在每种特征都分别进入分类器后，将分类器的输出结果进行融合。如将字符的网格特征输入到SVM中，SVM分类器会得到字符属于各个类的分数，同样将HOG特征输入到SVM中，SVM分类器会得到另一个字符属于各个类的分数。将这两个分类器的分数进行相加，最后取字符属于某个类的最大分数的类别作为字符的标签。

## 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。

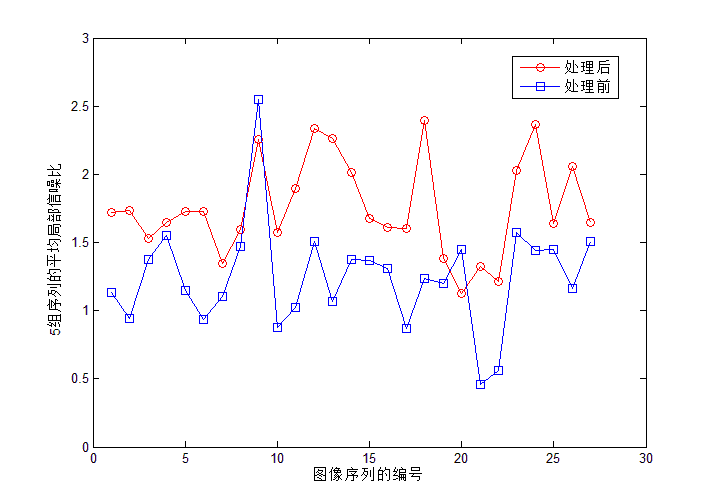


图4.6 图像局部信噪比

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。

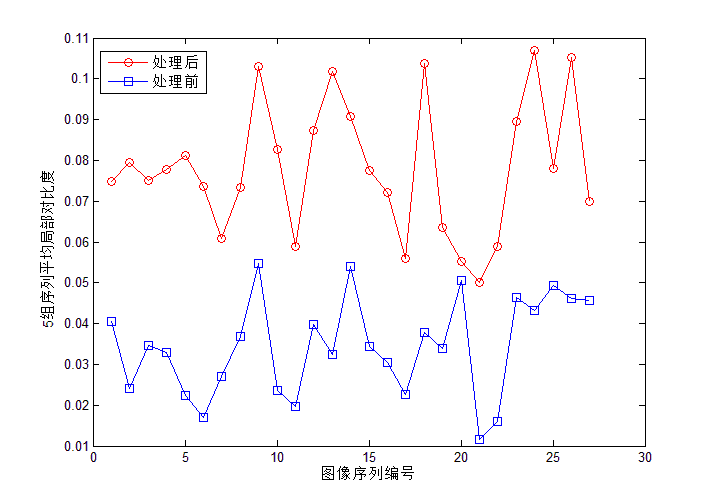


图4.7 图像局部对比度

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。