# 使用Parzen窗法估计概率密度实现手写数字识别

## 实验目的

使用Parzen窗法估计贝叶斯公式中的类条件概率密度，进而实现手写数字的识别。

## 实验方法

### 搭建平台

使用Windows系统自带的画图工具分别构建训练样本集和测试样本集，使用MATLAB GUI编程环境对它们进行处理、识别。

### 特征描述

将手写部分从图像中分割，并转化为灰度图像，而后将其分割为6\*6=36个小图片，用每个小图片中黑色像素占总像素的比重作为该样本的特征，也就是一个36维的向量。

子函数get\_feature(image)代码如下：

function[feature] = get\_feature(image)

%从图像中分割出手写部分并转为灰度

image2 = pretreat(image);

A = mat2cell(image2,[10,10,10,10,10,10],[7,7,7,7,7,7]);

B = reshape(A,1,36);

for i=1:length(B)

temp = cell2mat(B(i));

%feature(i)表示黑像素点占一个小块的比例

feature(i) = sum(temp(:) ~= 255) / numel(temp);

end

%feature

end

### 关键算法

#### 非参数估计的基本原理与直方图方法

直方图（histogram）方法是最简单直观的非参数估计方法，也是日常生活中人们最常用的对数据进行统计分析的方法。

进行直方图估计的做法是：

1. 把样本x的每个分量在其取值范围内分成k个等间隔的小窗。如果x是d维向量，则这种分割就会得到个小体积或者称为小舱，每个小舱的体积记作V。
2. 统计落入每个小舱内的样本数目。
3. 把每个小舱内的概率密度看作常数，并用作为其估计值，其中N为样本总数。

在上面的直方图估计中，我们采用的是把样本空间在样本取值范围内等分的做法。可以设想，小舱的选择是与估计的效果密切相连的：如果小舱选择过大，那么假设p(x)在小舱内为常数的做法就显得粗糙，导致最终估计出的密度函数也非常粗糙；而另一方面，如果小舱过小，则有些小舱就可能没有样本或者很少样本，导致估计出的概率密度函数很不连续。

所以，小舱的选择应该与样本总数相适应。理论上讲，假定样本总数是n，小舱的体积是，在x附近位置上落入小舱的样本数是，那么当样本趋于无穷时趋于的条件是

1. 
2. 
3. 

直观的解释是：随着样本数的增加，小舱的体积应该尽可能小，同时又必须保证小舱内有充分多的样本，但每个小舱内的样本数又必须是总样本数中很小的一部分。

#### Parzen窗法

假定x是d维特征向量，并假设每个小舱是一个超立方体，它在每一维的棱长都为h，则小舱的体积是，要计算每个小舱内落入的样本数目，可以用如下的d维单位方窗函数：



该函数在以原点为中心的d维单位超正方体内取值为1，在其他地方取值都为0。

对于每一个x，要考察某个样本是否在这个x为中心，h为棱长的立方体小舱内，就可以通过计算来完成。于是，对于n个观测样本，落入以x为中心的超立方体内的样本数就可以写成，这样就可以得到对于任意一点x的密度估算的表达式

。

从另外一个角度理解该公式：定义窗函数，它反映了一个观测样本对在x处概率密度的贡献，而概率密度函数就是把每个样本的贡献进行平均，即。

这种用窗函数估计概率密度的方法称为Parzen窗方法。

上述的称为方窗函数，除此之外，还有很多其他窗函数，如高斯窗、超球窗等。

#### 算法实现

本次实验使用的是方窗函数，其MATLAB代码实现如下：

%方形窗口函数

function[res] = window\_function(vec)

if ((vec<0.75) &(vec>-0.75)) == ones(1,36)

res=1;

else

res=0;

end

End

使用该窗口函数估计概率密度的子函数如下：

%用Parzen窗法

function[pXwi] = get\_pXwi(new\_feature)

pXwi = [];

for i=0:9

image\_list = dir(strcat('./样本',num2str(i),'/\*.png'));

p=0;

for j=1:length(image\_list)

feature = get\_feature(imread(strcat('./样本',num2str(i),'/',image\_list(j).name)));

p = p+window\_function(new\_feature-feature);

end

p = p/length(image\_list);%除以样本总数

p = p/2^length(new\_feature);%除以空间体积

pXwi = [pXwi,p];

end

## 实验结果

### 训练样本集构建

使用Windows系统自带的画图软件创建训练样本集，使用10个文件夹分别保存10个类的训练集，如下图。



图 三-1 训练样本集

每一个样本都有一个固定的模板图片，用画图工具在该图片的指定位置手写数字，模板图片如下图。

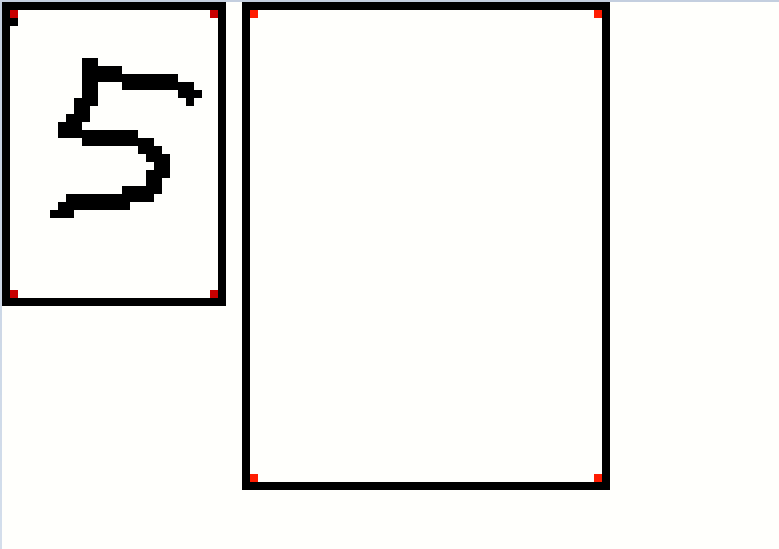


图 三-2 样本构建的模板图片

某一手写完成的样本如下图。

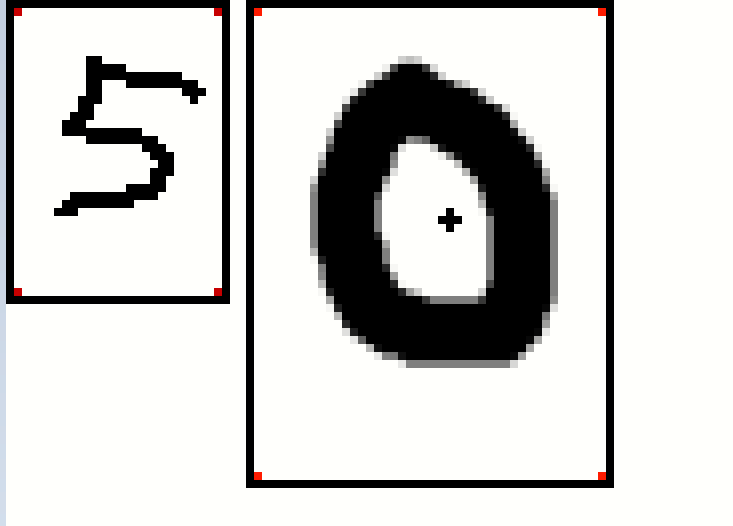


图 三-3 某一手写完成的样本

### 测试集构建

测试集样本的制作与训练集基本相同，不同的是，将所有图片都保存至同一文件夹“测试样本集”中，以供程序读取识别。

本次实验测试样本集如下图。

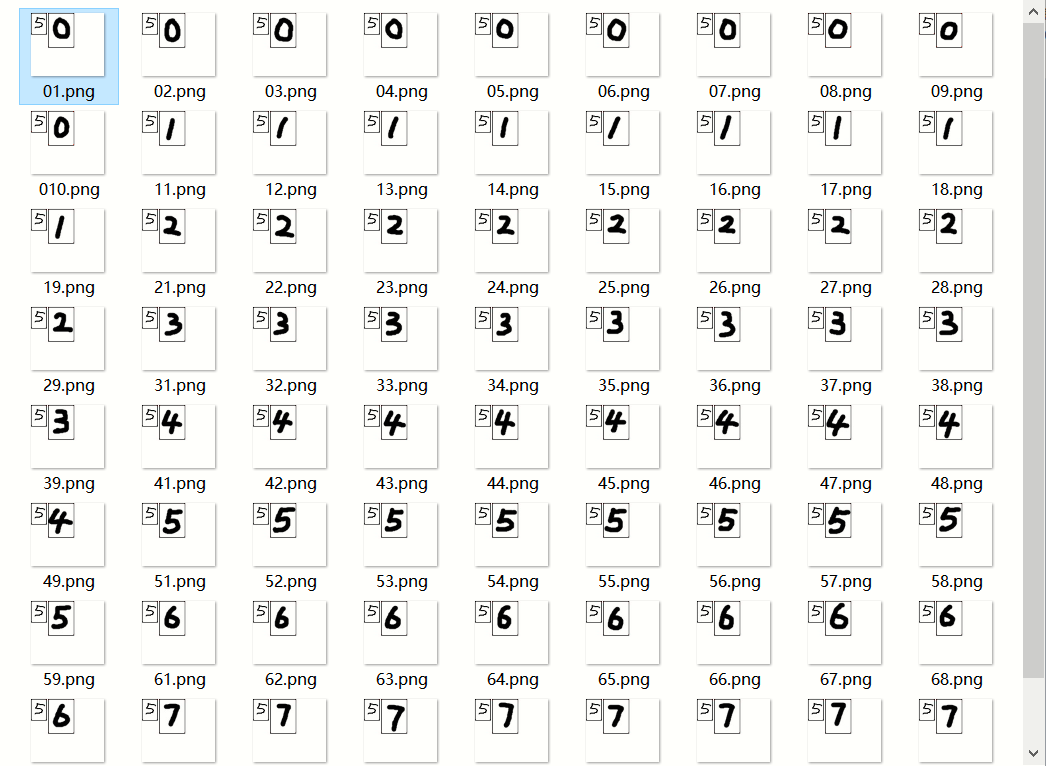


图 三-4 测试样本集

该样本集中有手写数字0-9各10个，它们的命名以各自的数字开头，用于实验中计算正确率。

### 实验结果

实验的运行环境是MATLAB GUI，本次实验各控件布局如下图。

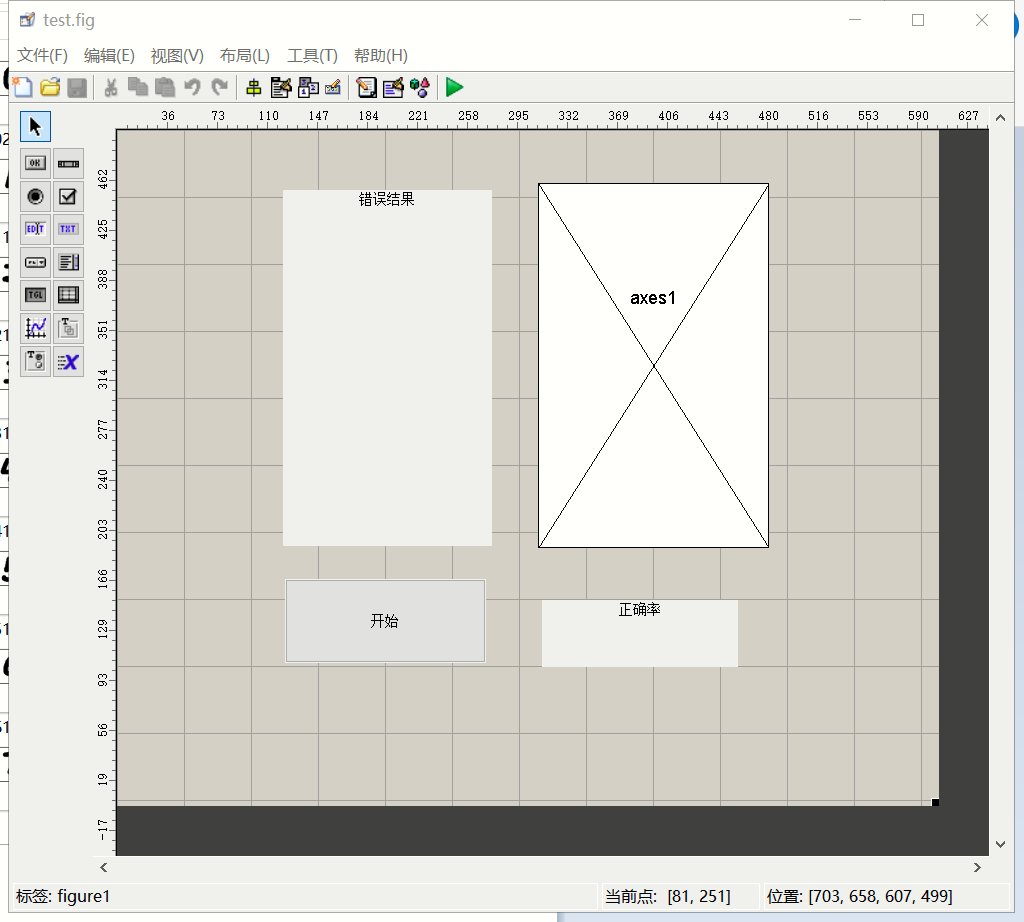


图 三-5 GUI控件的布局

图中已表明了各控件的作用，其中axes1用于运行时显示正在识别的样本。

程序运行时效果如下图：

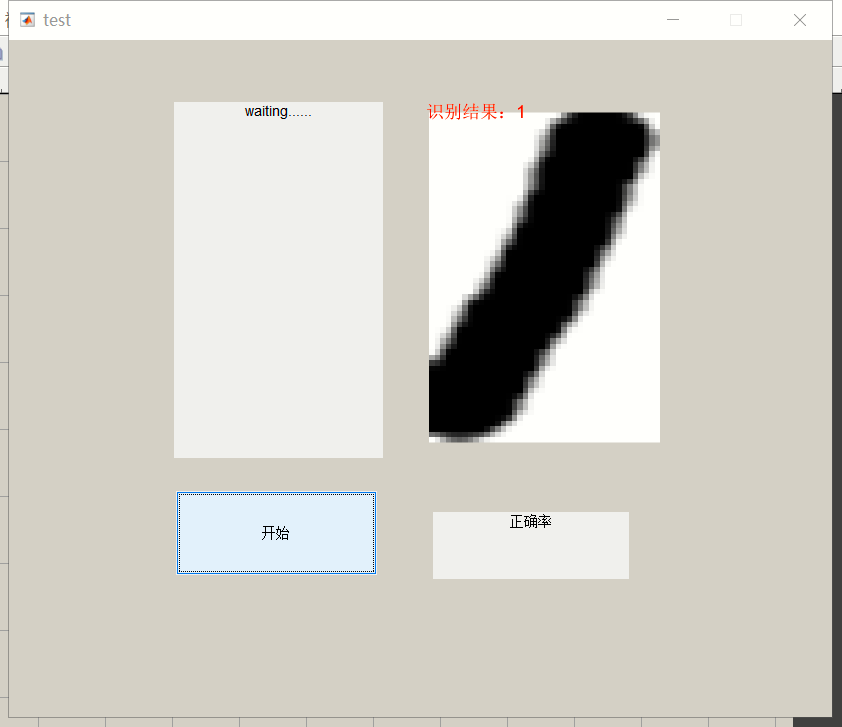


图 三-6 程序正在运行

程序运行结束时如下图：

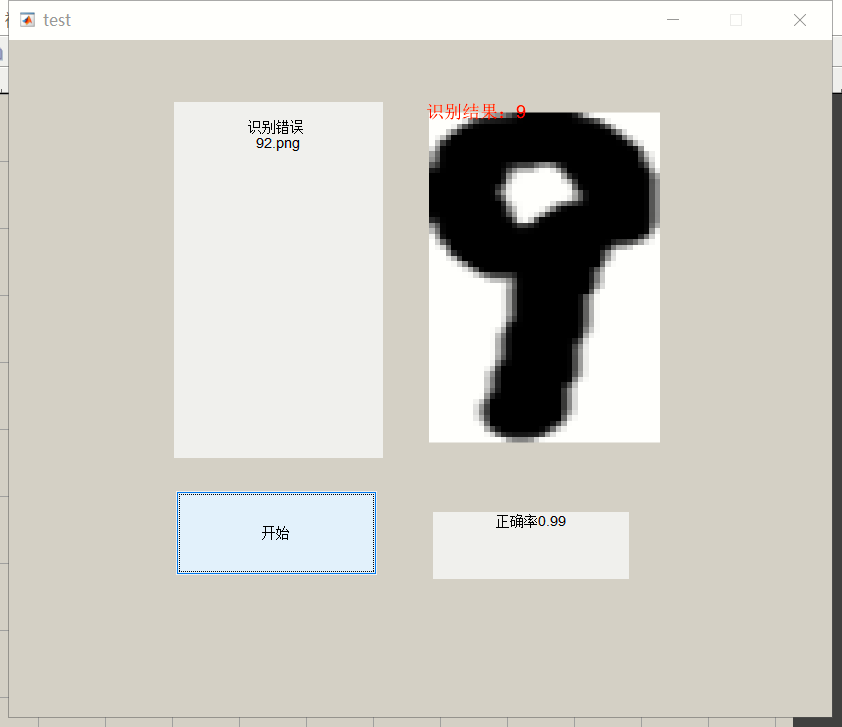


图 三-7 程序运行结束

### 性能量化评估

本次实验待识别样本共100个，运行结果显示，本次实验中识别的正确率为99%。

## 结果讨论

运行结果显示，对于数字9的样本“92.png”识别错误，该样本如下图。

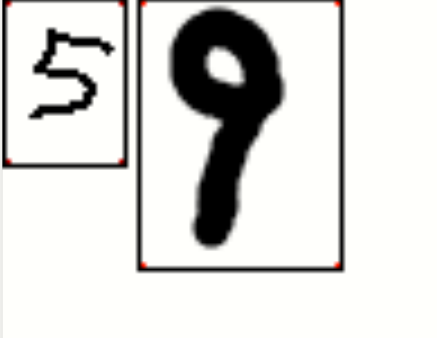


图 四-1 识别错误的样本

在MATLAB命令窗口中对其进行测试如下：

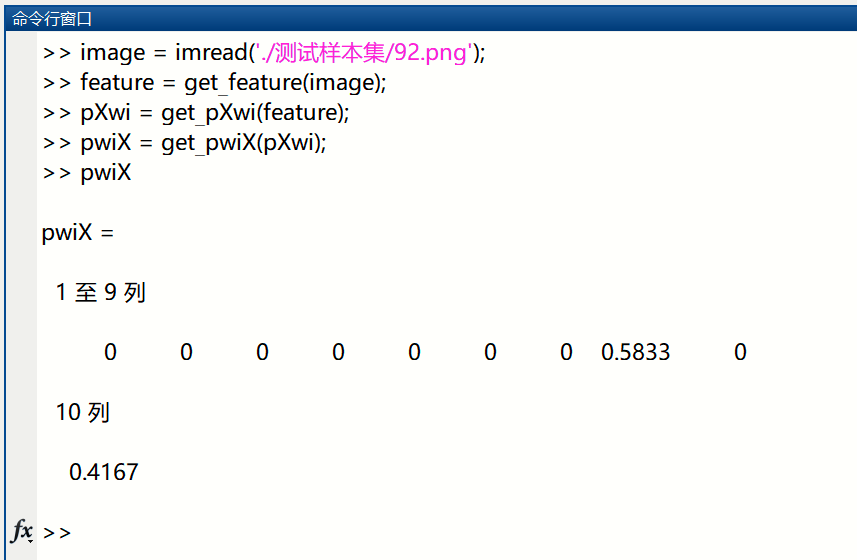


图 四-2 对识别错误的样本进行测试

结果显示，程序认为该样本是7的概率为0.5833，是9的概率为0.4167。直观看来，这两个数字确实有相似之处。

从测试结果看，程序认为是其他数字的概率均为0，原因在于本次试验使用的是最简单的方窗函数，如果是用高斯窗或超球体窗结果会更“精确”。

## 结论

本次实验使用MATLAB编程环境，完成了Parzen窗函数估计概率密度的算法实现，进而通过贝叶斯公式实现了手写数字识别，通过对测试样本集（100个样本）的识别，结果显示正确率达99%，表示实验过程中各个样本集的建立以及所用算法较为可靠。