**3.基于matlab的人脸识别算法**

**3.1 问题描述**

对于一幅图像可以看作一个由像素值组成的矩阵，也可以扩展开，看成一个矢量，如一幅N\*N 象素的图像可以视为长度为N2 的矢量，这样就认为这幅图像是位于N2 维空间中的一个点，这种图像的矢量表示就是原始的图像空间，但是这个空间仅是可以表示或者检测图像的许多个空间中的一个。不管子空间的具体形式如何，这种方法用于图像识别的基本思想都是一样的，首先选择一个合适的子空间，图像将被投影到这个子空间上，然后利用对图像的这种投影间的某种度量来确定图像间的相似度，最常见的就是各种距离度量。因此，本次试题采用PCA算法并利用GUI实现。

对同一个体进行多项观察时，必定涉及多个随机变量X1，X2，…，Xp，它们都是的相关性, 一时难以综合。这时就需要借助主成分分析来概括诸多信息的主要方面。我们希望有一个或几个较好的综合指标来概括信息，而且希望综合指标互相独立地各代表某一方面的性质。

任何一个度量指标的好坏除了可靠、真实之外，还必须能充分反映个体间的变异。如果有一项指标，不同个体的取值都大同小异，那么该指标不能用来区分不同的个体。由这一点来看，一项指标在个体间的变异越大越好。因此我们把“变异大”作为“好”的标准来寻求综合指标。

**3.1.1 主成分的一般定义**

设有随机变量X1，X2，…，Xp， 其样本均数记为 ， ，…， ，样本标准差记为S1，S2，…，Sp。首先作标准化变换，我们有如下的定义：

(1) 若C1=a11x1+a12x2+ … +a1pxp，…，且使 Var(C1)最大，则称C1为第一主成分；

(2) 若C2=a21x1+a22x2+…+a2pxp， …，(a21，a22，…，a2p)垂直于(a11，a12，…，a1p)，且使Var(C2)最大，则称C2为第二主成分；

(3) 类似地，可有第三、四、五…主成分，至多有p个。

**3.1.2 主成分的性质**

主成分C1，C2，…，Cp具有如下几个性质：

（1) 主成分间互不相关，即对任意i和j，Ci 和Cj的相关系数

Corr(Ci，Cj)=0 i**j

(2) 组合系数(ai1，ai2，…，aip)构成的向量为单位向量，

(3) 各主成分的方差是依次递减的， 即

Var(C1)≥Var(C2)≥…≥Var(Cp)

(4) 总方差不增不减， 即

Var(C1)+Var(C2)+ … +Var(Cp)

=Var(x1)+Var(x2)+ … +Var(xp) =p

这一性质说明，主成分是原变量的线性组合，是对原变量信息的一种改组，主成分不增加总信息量，也不减少总信息量。

(5) 主成分和原变量的相关系数 Corr(Ci，xj)=aij =aij

(6) 令X1，X2，…，Xp的相关矩阵为R, (ai1，ai2，…，aip)则是相关矩阵R的第i个特征向量(eigenvector)。而且，特征值i就是第i主成分的方差， 即

Var(Ci)=*i*

其中i为相关矩阵R的第i个特征值(eigenvalue)

**1≥**2≥…≥*p*≥0

**3.1.3 主成分的数目的选取**

前已指出，设有p个随机变量，便有p个主成分。由于总方差不增不减，C1，C2等前几个综合变量的方差较大,而Cp，Cp-1等后几个综合变量的方差较小, 严格说来，只有前几个综合变量才称得上主(要)成份，后几个综合变量实为“次”(要)成份。实践中总是保留前几个，忽略后几个。

保留多少个主成分取决于保留部分的累积方差在方差总和中所占百分比(即累计贡献率)，它标志着前几个主成分概括信息之多寡。实践中，粗略规定一个百分比便可决定保留几个主成分；如果多留一个主成分，累积方差增加无几，便不再多留。

**3.2 PCA算法的功能实现**

**3.2.1 人脸空间的建立**  
    假设一幅人脸图像包含N个像素点，它可以用一个N维向量Γ表示。这样，训练样本库就可以用Γi（i=1，...，M）表示。    协方差矩阵C的正交特征向量就是组成人脸空间的基向量，即特征脸。  
    将特征值由大到小排列：λ1≥λ2≥...≥λr,其对应的特征向量为μk。这样每一幅人脸图像都可以投影到由u1,u2,...,ur张成的子空间中。因此，每一幅人脸图像对应于子空间中的一点。同样，子空间的任意一点也对应于一幅图像。  
**3.2.2 特征向量的选取**  
  虽然协方差矩阵Ω最多有对应于非零特征值的k（k远小于M）个特征向量，但是通常情况下，k仍然很大，而事实上，根据应用的要求，并非所有的特征向量都有需要保留，而特征空间投影的计算速度是直接与创建子空间所用的特征向量的数目相关，若考虑到计算时间的因素，可以适当的减去一些信息量少的特征向量，而且，去掉这些特征向量之后不一定不利于分类结果，有的情况下反而能够提高识别性能。

**3.2.3 人脸识别**  
   有了这样一个由"特征脸"张成的降维子空间，任何一幅人脸图像都可以向其投影得到一组坐标系数，这组系数表明了该图像在子空间中的位置，从而可以作为人脸识别的依据。换句话说，任何一幅人脸图像都可以表示为这组"特征脸"的线性组合，其加权系数即是K-L变换的展开系数，也可以称为该图像的代数特征。因此，在获得特征脸之后，就可以对每一类别的典型样本进行投影，由此得到每个人脸的投影特征从而构成人脸特征向量，作为下一步识别匹配的搜索空间。  
  输入图像及其在人脸空间上的投影（重构图像），人脸图像在人脸空间中的投影变化不明显而非人脸图像的投影变化明显。因此，检测一幅图像中是否存在人脸的基本思想是，计算该图像中任意位置处的局部图像与人脸空间之间的距离ε。其中，ε是局部图像是否为人脸的度量。因此，计算给定图像任意一点上的ε，就可以得到一映射图ε（x,y）。

**3.3 实验及结果分析**  实验在两个图库上测试，一个是自建人脸库，该库包含10个不同人物，每人有5张不同表情和姿态下的图片，总共50幅。另一个是ORL人脸库，该库包含40个不同人物，每人有10张图片，共400幅。用训练样本进行测试，识别率为100%。而随着训练样本的增加，识别率会有所提升，由于标准人脸库在采集时考虑了多种因素，人脸图像比较标准，所以识别率较自建的人脸库识别率高，另外因为自建人连库的图片太少，即训练样本太少，也会对结果产生影响，效果不是很好。进行直方图均衡化比灰度归一化的识别率高，预处理对识别的效果起着至关重要的作用。而此次实验的预处理还比较粗糙，PCA也只是起到了简单的特征脸降维的作用，要有更好的效果，还必须寻找更好的特征表达，使得可以尽量消除光照、表情、遮掩和姿势的影响。



图2 训练样本在坐标系下的投影矩阵

下图为测试流程截图：

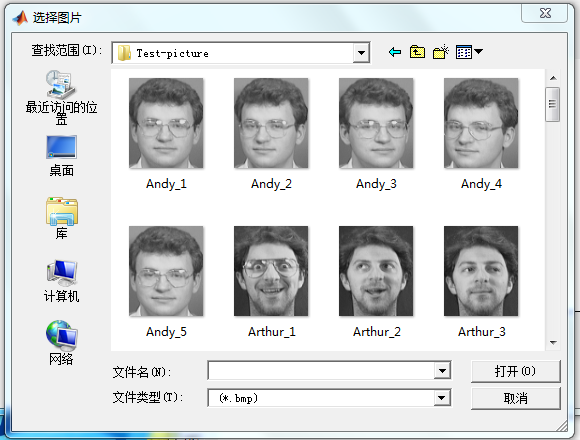
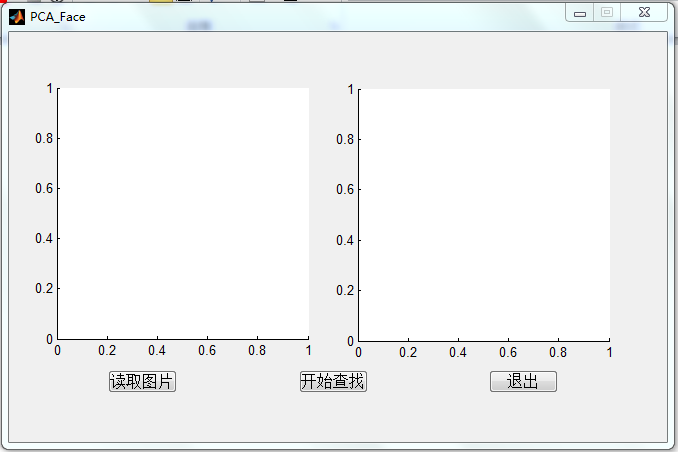


图3 用户使用界面 图4 选择图片

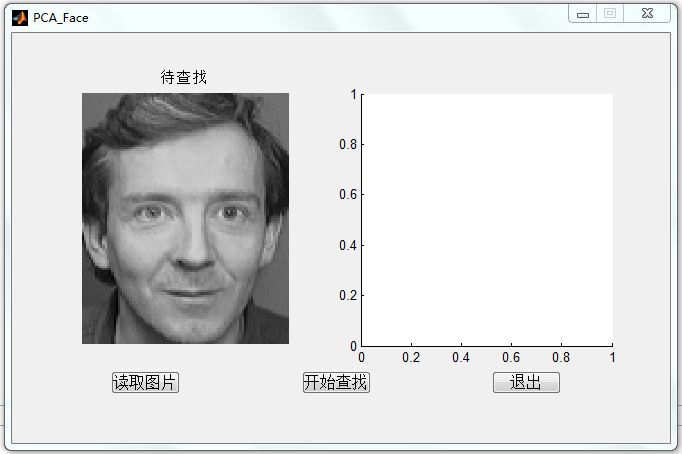


图5 图片选择后 图6 查找后

**3.4 个人总结**

因为我以前自学过一部分GUI方面的知识，所以本次试题我采用了Matlab中的GUI（图形用户界面）为主要表现形式，我认为在这次的题目下GUI比单纯的M文件会有更强的应用、测试及表现效果，但由于之前对人脸识别算法不太了解，所以花了很长时间去分析理解，最终选择了较为简单且实用的PCA算法。从使用的角度考虑，我还为库中的每个人取了一个名字，最终查找到时会在标题处显示其名字，下方显示库中与测试图片最接近的库图片。因为是利用了标准人脸库，并且识别的人数不是很多，所以最终的结果还是非常不错的，识别率可达100%。

**参考文献**

[1] 邓楠, 基于主成份分析的人脸识别, 2006.06

[2] 龚勋，PCA与人脸识别及其理论基础，2007.04

[3] 周涛，基于PCA的人脸识别研究，2004.07

[4] 王宇飞，基于特征脸和支持向量机的人脸识别方法研究与实现，2005.03

[5] 刘保柱，《MATLAB7.0从入门到精通》，人民邮电出版社

**附录—matlab 源码**

**1.“读取图片”按钮**

function pushbutton1\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton1 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

%% 读取待查找图片

global im;%由于要在两个按钮函数中使用，故使用全局变量

[filename, pathname]=...

uigetfile({'\*.bmp'},'选择图片');

str = [pathname, filename];%合成路径+文件名

im = imread(str);%读取图片

axes( handles.axes1);%使用第一个axes

imshow(im);title('待查找')%显示图片

**2.“开始查找”按钮**

function pushbutton2\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton2 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

%% PCA人脸识别

global im;%使用全局变量

imgdata=[];%训练图像矩阵

for i=1:10

for j=1:5

a=imread(strcat('C:\Users\dell\Desktop\ORL\s',num2str(i),'\',num2str(j),'.bmp'));

b=a(1:112\*92); % b是行矢量 1×N，其中N＝10304

b=double(b);

imgdata=[imgdata; b]; % imgdata 是一个M \* N 矩阵，imgdata中每一行数据一张图片，M＝50

end;

end;

imgdata=imgdata'; %每一列为一张图片

imgmean=mean(imgdata,2); % 平均图片，N维列向量

for i=1:50

minus(:,i) = imgdata(:,i)-imgmean; % minus是一个N\*M矩阵，是训练图和平均图之间的差值

end;

covx=minus'\* minus; % M \* M 阶协方差矩阵

[COEFF, latent,explained] = pcacov(covx'); %PCA，用协方差矩阵的转置来计算以减小计算量

% 选择构成95%的能量的特征值

i=1;

proportion=0;

while(proportion < 95)

proportion=proportion+explained(i);

i=i+1;

end;

p=i-1;

% 训练得到特征脸坐标系

i=1;

while (i<=p && latent(i)>0)

base(:,i) = latent(i)^(-1/2)\*minus \* COEFF(:,i); % base是N×p阶矩阵，用来进行投影，除以latent(i)^(1/2)是对人脸图像的标准化

i = i + 1;

end

% 将训练样本对坐标系上进行投影,得到一个 p\*M 阶矩阵为参考

reference = base'\*minus;

%% 测试过程——在测试图片文件夹中选择图片，进行查找测试

a=im;

b=a(1:10304);

b=double(b);

b=b';

object = base'\*(b-imgmean);

distance=100000;

%最小距离法，寻找和待识别图片最为接近的训练图片

for k=1:50

temp= norm(object - reference(:,k));

if (distance > temp)

which = k;

distance = temp;

end;

end;

%找出距离最近的图片所在的位置

num1 = ceil(which/5);%第num1个文件夹

num2 = mod(which,5);%第num2个图片文件

if (num2 == 0)

num2 = 5;

end;

I=imread(strcat('C:\Users\dell\Desktop\Face\s',num2str(num1),'\',num2str(num2),'.bmp'));%读取该图片

axes( handles.axes2);%使用第2个axes

%输出判断所最接近的训练样本的人脸图片并显示其人名

switch num1

case 1

imshow(I);title('此人为 Peter');%显示图片及人名

case 2

imshow(I);title('此人为 Ben');

case 3

imshow(I);title('此人为 Nick');

case 4

imshow(I);title('此人为 Park');

case 5

imshow(I);title('此人为 Linkin');

case 6

imshow(I);title('此人为 Andy');

case 7

imshow(I);title('此人为 Arthur');

case 8

imshow(I);title('此人为 Lisa');

case 9

imshow(I);title('此人为 Dana');

case 10

imshow(I);title('此人为 Sara');

end;

**3.“退出”按钮**

function pushbutton3\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton3 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

%% 退出

close(gcf);