PCA实验人脸库-人脸识别（四）

原创

on2way 于 2015-01-04 13:32:38 发布
分类专栏： 模式识别&机器学习 人脸识别 文章标签： matlab 主成分分析 人脸识别 模式识别

一）：人脸数据库

AR人脸库（包含50位男性和50位女性每人26张人脸共2600张人脸图片 ）：

http://www.datatang.com/data/46195

ORL人脸库（包含40个人的每人10张人脸的共400张人脸）：

http://www.cl.cam.ac.uk/Research/DTG/attarchive:pub/data/att_faces.tar.Z

或者

http://www.cl.cam.ac.uk/Research/DTG/attarchive:pub/data/att_faces.zip

二）：PCA实验

人脸数据库有很多，这里先暂时选取了上面两种来实验。主成分分析PCA算法在前面介绍过，这里主要接上节：

主成分分析-简单人脸识别（二）

这篇里面使用的人脸单一，数据量少，识别率无法观察。这里有了数据库在重新评估下PCA算法的准确率。程序上面略有改动。

2.1）数据库的导入

对上述下载下来的数据库，由于量太大了，没法去一一改文件夹呀整理人脸照片等等，那么如何快速导入并直接使用下载下来的文件夹呢？下载后可以发现AR数据库没有子文件夹，而ORL数据库有二级子文件夹。首先把对应的文件夹复制到matlab程序所在的当前文件夹中，然后编写matlab直接读取文件夹信息：

读取AR文件夹：

```
function filedata = File_Read_AR()
%  用法:    filedata = File_Read_AR();
%  样本数:man:50个,每个样本还有26个子样本,每个子样本大小:165*120
%          woman:50个,每个样本还有26个子样本,每个子样本大小:165*120
% 载入文件夹
pathname = uigetdir(cd, 'C:\Users\Administrator\Documents\MATLAB\人脸识别\PCA+SVM');
filesjpg = ls(strcat(pathname,'/*.pgm'));
files = [cellstr(filesjpg)]; % 得到文件路径
for i = 1:100     %选取前50/100个人的各26张照片(50*26)
    for j = 1:26
        Filesname = strcat(pathname,'/',files((i-1)*26+j));
        filedata{i,j} = imread(cell2mat(Filesname));
    end
end
```

读取ORL文件夹:

```
function filedata = File_Read_ORL()
%  用法:    filedata = File_Read();
%  样本数:40个人,每个样本还有10个子样本,每个子样本大小:119*92
% clc; clear all;
% 载入文件夹
for i = 1:40
    pathname = uigetdir(cd, 'C:\Users\Administrator\Documents\MATLAB\人脸识别\PCA+SVM');
    filesjpg = ls(strcat(pathname,'/*.pgm'));
    files = [cellstr(filesjpg)]; % 得到文件路径
    len = length(files); % 文件个数
    for j = 1:len
        Filesname = strcat(pathname,'/',files(j));
        filedata{i,j} = imread(cell2mat(Filesname));
    end
end
```

这里程序运行时需要选择文件夹,AR只需要选择一次,ORL需要选择40次。

为了避免每次程序实验程序的时候去反复的选择文件夹,在一次运行后保存下来得到filedata文件。如:`save face_orl.mat;`

保存为.mat格式后,以后每次运行的时候只需要点一下就可以导入数据库了。

2.2)将样本分类

有了总样本数据库,现在就是把样本分为训练集于测试集了。定义一个num数表示训练集中的对于每个人,随机选其中的num张脸当成样本,其他的(all-num)当成测试集,最后在转化为PCA可以使用的格式。

```
function [train_face,test_face] = imgdata(filedata,num)
%%      每个人取num个样本脸
k = 2;  %k=1,选用 ORL人脸库  否则选用 AR人脸库
if k == 1
%用法：适用于 ORL人脸库
    [m,n] = size(filedata{1,1});  %取图片大小
    for i = 1:40   %共有40个人
        n_rand = randperm(10);
        for j = 1:num
            train_pic{(i-1)*num+j} = filedata{i,n_rand(j)};%j
        end
        for k = 1:(10-num)
            test_pic{(i-1)*(10-num)+k} = filedata{i,n_rand(k+num)};%k+num
        end
    end
    for i = 1:40*num
        train_face(i,:) = reshape(train_pic{i},1,m*n);
    end
    for i = 1:40*(10-num)
        test_face(i,:) = reshape(test_pic{i},1,m*n);
    end
%%
else
    %用法：适用于 AR人脸库
    [m,n] = size(filedata{1,1});  %取图片大小
    num_people = size(filedata,1); %使用了num_people个人的各26张脸
    for i = 1:num_people   %共有num_people个人
        n_rand = randperm(26);
        for j = 1:num
            train_pic{(i-1)*num+j} = filedata{i,n_rand(j)};%
        end
        for k = 1:(26-num)
            test_pic{(i-1)*(26-num)+k} = filedata{i,n_rand(k+num)};
        end
    end
    for i = 1:num_people*num
        train_face(i,:) = reshape(train_pic{i},1,m*n);
    end
    for i = 1:num_people*(26-num)
        test_face(i,:) = reshape(test_pic{i},1,m*n);
    end
end
```

2.3）PCA进行降维

这里的PCA算法后前面的大致相同，但是有一点区别，区别在另一篇讲到过。程序如下：

```
function [img_new,img_mean,V] = PCA(img,k)
%用法：  [img_new,img_mean,V] = PCA(train_face,k)
%reshape函数：改变句矩阵的大小，矩阵的总元素个数不能变
img = double(img);
[m,n] = size(img);  %取大小
img_mean = mean(img); %求每列平均值
img_mean_all = repmat(img_mean,m,1);%复制m行平均值至整个矩阵
Z = img - img_mean_all;
T=Z*Z';  %协方差矩阵（非原始所求矩阵的协方差)
[V,D] = eigs(T,k);%计算T中最大的前k个特征值与特征向量
V=Z'*V;          %协方差矩阵的特征向量
for i=1:k        %特征向量单位化
    l=norm(V(:,i));
    V(:,i)=V(:,i)/l;
end
img_new = Z*V;  %低维度下的各个脸的数据
```

2.4）计算测试集数据库准确率

这里使用到PCA输出的低维下的主成分数据参数对测试集进行测试并计算准确率。

```
%--------------函数说明-------------
%-----人脸匹配----ORL/AR人脸库
%-----输入：训练样本的img_new   img_mean   V
%           k:训练集中每个人脸选取的个数（最大为10 or 26）
%-----输出：准确率accuracy
%-----------------------------------
function accuracy = facefind(img_new,img_mean,V,test_face,k)
%%
num_train = size(img_new,1);  %训练脸总数
num_test = size(test_face,1); %测试脸总数
true_num = 0;
for i = 1:num_test
    pic = double(test_face(i,:));
    pic1 = pic - img_mean;
    pic2 = pic1*V;
    for j = 1:num_train
        error(j) = norm(img_new(j,:) - pic2);  %求取范数距离
    end
    face_find = find(error<=min(error));   %取最小者
    face_find = ceil(face_find/k);  %向上取整  表示识别的是第几个人的脸
    true_face = ceil(i/(26-k));  %注意这个数字：10-ORL人脸库；26-AR人脸库
                                 %真实的是第几个人的脸
     if face_find == true_face
         true_num = true_num+1;  %正确分类的个数
     end
end
accuracy = true_num/num_test;
```

三）实验结果

（3.1）先对ORL进行实验

这里就需要改一下上述涉及到的几个值了。比如imgdata函数中k=1；facefind函数的倒数第六行数字得改到10吧。

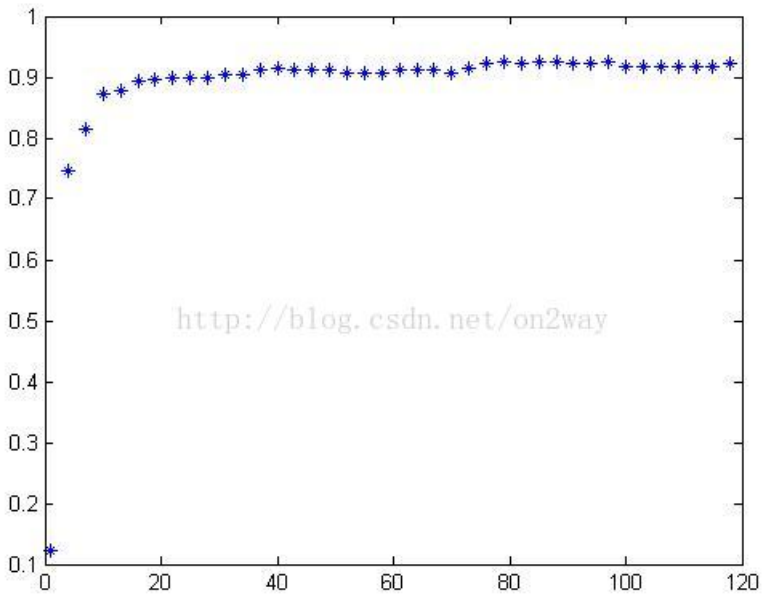
改好后运行一次facefind，选取PCA中降维数为30，每个人中选3个脸作为训练集（其他7个作为测试集）试试。

```
load('face_ORL.mat')
>> [train_face,test_face] = imgdata(filedata,3);
>> [img_new,img_mean,V] = PCA(img,30);
>> accuracy = facefind(img_new,img_mean,V,test_face,3)

accuracy =

    0.9107
```

准确率还挺高的0.9107。为了测试降维的维数对准确率的影响，设置PCA中不同的维数值并观察情况，还是以每人3个训练集实验：

```
>> load('face_ORL.mat')
>> num = 3;  %每个样本选取num个脸训练（40个样本，每个人10个脸）
[train_face,test_face] = imgdata(filedata,num);
for k = 1:num:40*num
    [img_new,img_mean,V] = PCA(train_face,k);
    accuracy(k) = facefind(img_new,img_mean,V,test_face,num);
end
    n = size(accuracy,2);
   plot(1:num:n,accuracy(1:num:n),'*');
```

结果如下：

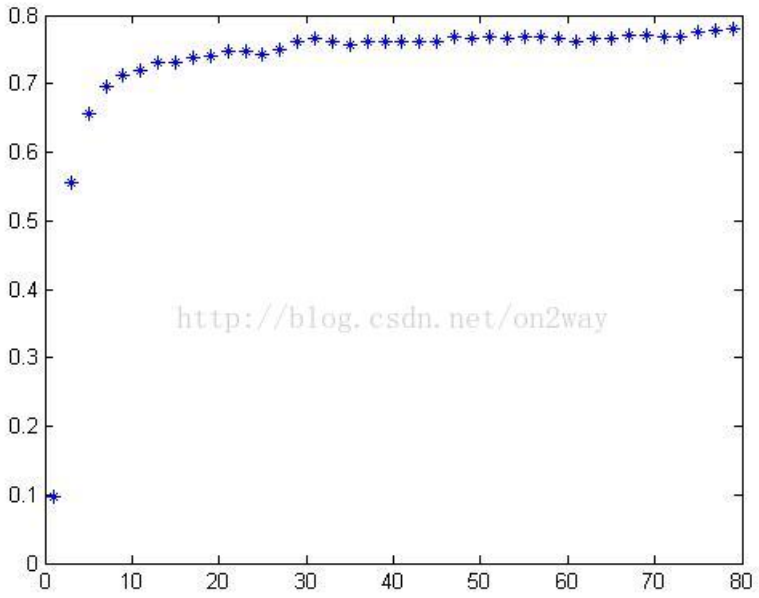


观察accuracy数组可以看到本次的最大识别率在降维度为79维，最大accuracy为0.925；而从图上可以看到基本上到20或者40后就基本上没什么改进了。

下面在实验一次每人2个训练集实验；改num=2；

```
>> load('face_ORL.mat')
>> num = 2;  %每个样本选取num个脸训练（40个样本，每个人10个脸）
[train_face,test_face] = imgdata(filedata,num);
for k = 1:num:40*num
    [img_new,img_mean,V] = PCA(train_face,k);
    accuracy(k) = facefind(img_new,img_mean,V,test_face,num);
end
```

```
    n = size(accuracy,2);
    plot(1:num:n,accuracy(1:num:n),'*');
```



本次accuracy最大值为0.7813；图上也可以看到降维40维后基本上也是没有很大变化，所以这样也就知道对该数据库一般让其降维40就够了。要注意的是每次重新运行一遍实验结果会不一样，原因就是在选取训练集的时候是随机选的，imgdata函数中于这样一个命令randperm吧。

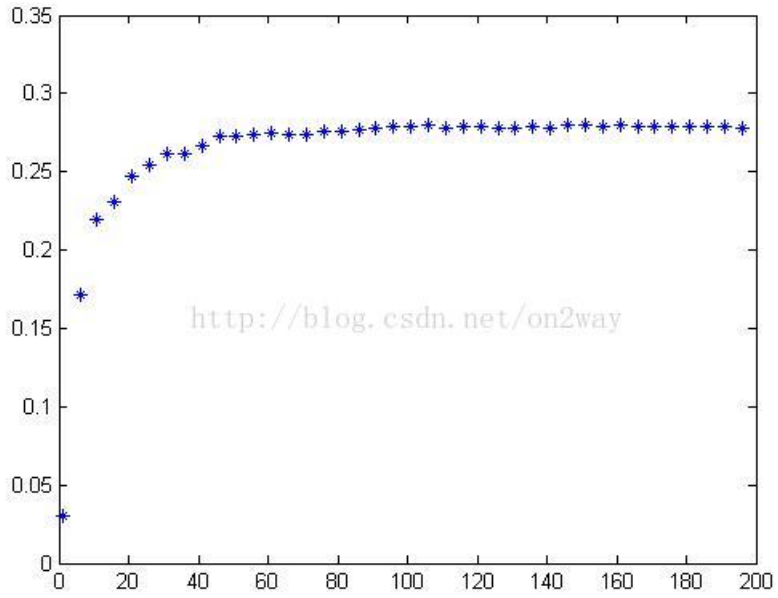
（3.2）再对AR进行实验

改参数imgdata函数中k=2；facefind函数的倒数第六行数字得改到26。

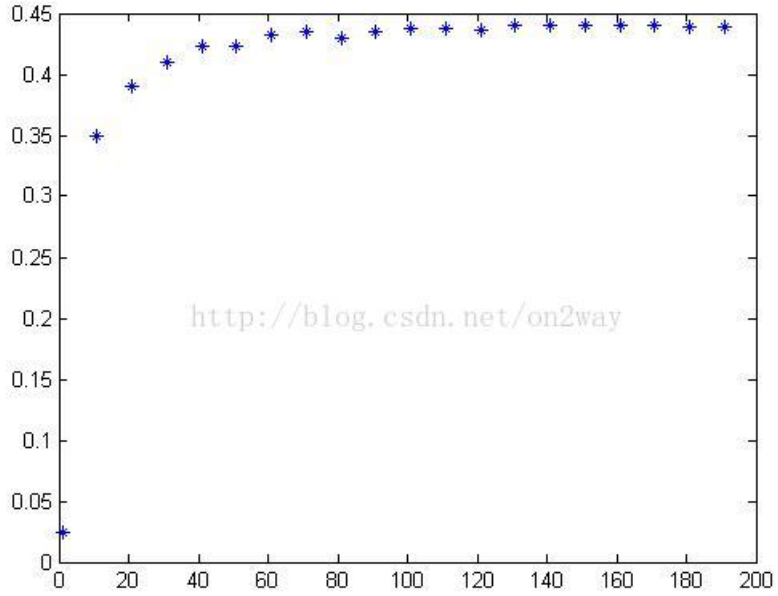
由于这个样本库比较大，我们直选去前50个男士的各26张脸实验，由于每个人有26个样本，先选取每人5个样本实验。

```
>> load('face_AR_50.mat')
>> num = 5;  %每个样本选取num个脸训练（50个样本，每个人26个脸）
[train_face,test_face] = imgdata(filedata,num);
for k = 1:num:200
    [img_new,img_mean,V] = PCA(train_face,k);
    accuracy(k) = facefind(img_new,img_mean,V,test_face,num);
end
    n = size(accuracy,2);
    plot(1:num:n,accuracy(1:num:n),'*');
```

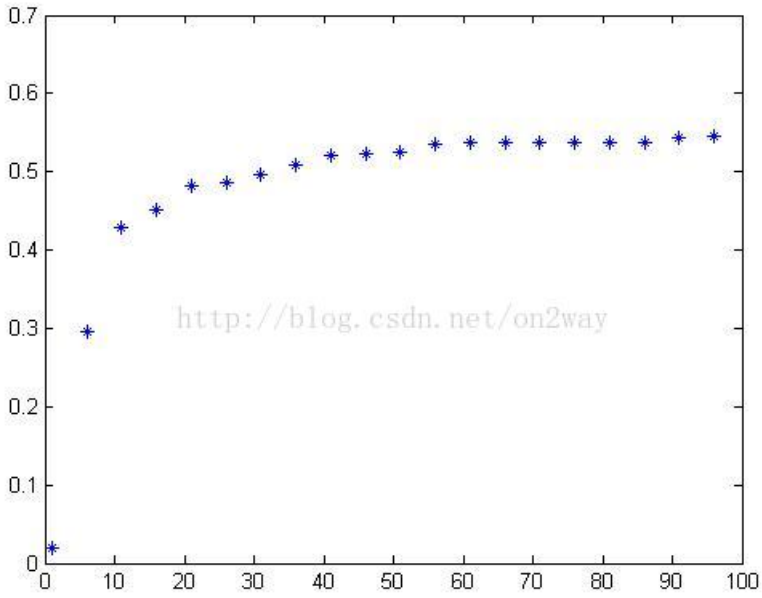
擦，这个数据库运行2分钟多，结果还不好，最大accuracy才0.28；



把训练集加多一点，看看准确率高不高点，选num=10,重复上述程序，发现最大accuracy为0.44；



似乎100维后结果就不变了，降低一点让程序快一点，同时再改num=15实验，发现最大accuracy为0.54；



想想每人总共才26个样本，用了一大半（15个），准确率才0.54，这是为何？说明这个数据库的人脸之间差异性很大，单纯的PCA已经无法很好的识别出来了。相比之下ORL数据库似乎差异性小些，同时也反映一个现实就是想要验证你的算法识别率好不好，拿AR数据库来实验吧，准确率一定让你大吃一惊（哈哈）。

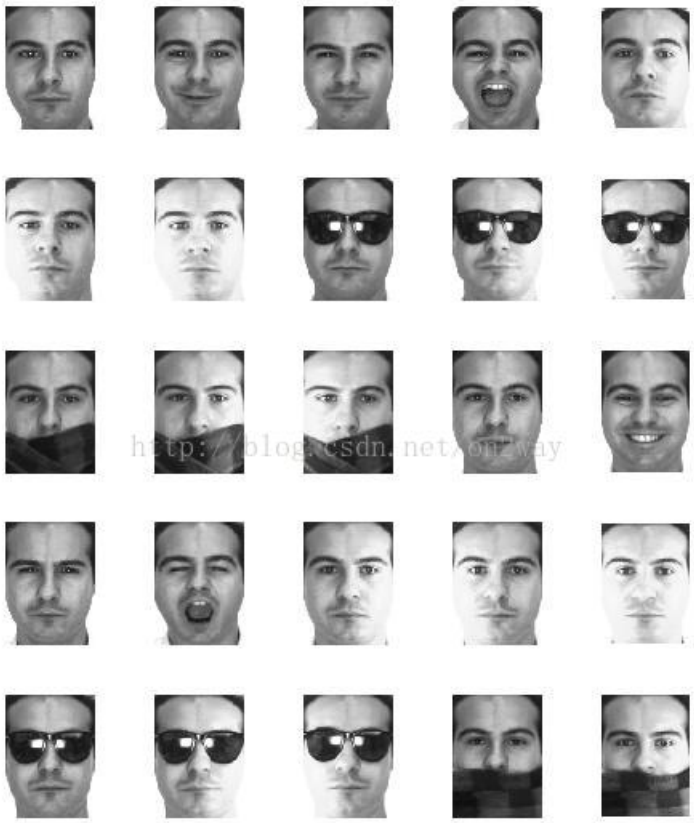
现在来看看为什么AR数据库会如此呢？现在进行史上最优秀的识别算法--人眼识别（-_-!）,先显示ORL中某个样本的10副图：

```
>> load('face_ORL.mat')
>> for i = 1:10
     pic = filedata{1,i};
     subplot(2,5,i),imshow(pic);
 End
```

人眼识别好简单，样本间差异性确实不大。

再来看看AR数据库吧，由于每个人26张，为了视图方便这里显示25张吧，

```
>> load('face_AR_50.mat')
>> for i = 1:25
     pic = filedata{1,i};
     subplot(5,5,i),imshow(pic);
 End
```



再擦，亮度差异大，戴墨镜，戴围巾，各种表情，我现在知道了为什么单纯的PCA扛不住了，我现在就让你去人眼识别第10张与第25张，如果不和你说这是同一个人，你能识别出来不（-_-!!!!）？

ok归根到底，PCA还是有一点用的，适合于全局特征，像这种局部特征确实无能为力，现在只能再闭关修行寻找更好的算法，虽然不可能达到100%识别率，但是提高准确率还是极有可能的。