1 PCA 与人脸识别及其理论基础

龚 勋（2007-4-29）

1.1 问题描述[1]

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 N^2 的矢量，这样就认为这幅图像是位于 N^2 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。

1.1.1 K-L 变换[1]

PCA 方法是由 Turk 和 Pentlad 提出来的，它的基础就是 Karhunen-Loeve 变换(简称 KL 变换)，是一种常用的正交变换。下面我们首先对 K-L 变换作一个简单介绍:

假设 X 为 n 维的随机变量，X 可以用 n 个基向量的加权和来表示:

$$X = \sum_{i=1}^{n} \alpha_i \phi_i$$

式中: α_i 是加权系数，ϕ_i 是基向量，此式还可以用矩阵的形式表示:

$$X = (\phi_1, \phi_2, \cdots, \phi_n)(\alpha_1, \alpha_2, \cdots, \alpha_n)^T = \Phi \boldsymbol{\alpha}$$

取基向量为正交向量，即

$$\Phi^T \Phi_j = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \Rightarrow \Phi^T \Phi_j = I$$

则系数向量为：

$$\boldsymbol{\alpha} = \Phi^T X$$

综上所述，K-L 展开式的系数可用下列步骤求出:

步骤一 求随即向量 X 的自相关矩阵 $R = E\left[X^T X\right]$*，由于没有类别信息的样本集的* μ *均值向量，常常没有意义，所以也可以把数据的协方差矩阵* $\sum = E\left[(x-\mu)(x-\mu)^T\right]$ *作为 K_L 坐标系的产生矩阵，这里* μ *是总体均值向量。*

步骤二 求出自相关矩阵或协方差矩阵 R 的本征值 λ_i *和本征向量* ϕ_i，$\Phi = (\phi_1, \phi_i, \cdots, \phi_n)$

步骤三 展开式系数即为 $\boldsymbol{\alpha} = \Phi^T X$

K_L 变换的实质是建立了一个新的坐标系，将一个物体主轴沿特征矢量对齐的旋转变换，这个变换解除了原有数据向量的各个分量之间相关性，从而有可能去掉那些带有较少信息的坐标系以达到降低特征空间维数的目的。

1.1.2 利用 PCA 进行人脸识别

完整的 PCA 人脸识别的应用包括几个步骤：人脸图像预处理；读入人脸库，训练形成特征子空间；把训练图像和测试图像投影到上一步骤中得到的子空间上；选择一定的距离函数进行识别。下面详细描述整个过程（源码见’faceRec.m’）。

1. 读入人脸库

归一化人脸库后，将库中的每人选择一定数量的图像构成训练集，其余构成测试集。设归一化后的图像是 n*m,按列相连就构成 N=n*m 维矢量，可视为 N 维空间中的一个点，可以通过 K-L 变换用一个低维子空间描述这个图像。

2. 计算 K- L 变换的生成矩阵

所有训练样本的协方差矩阵为（以下三个等价）：

$$\begin{cases} 1.\ \ C_A = (\sum_{k=1}^{M} x_k \bullet {x_k}^T)/M - m_x \bullet {m_x}^T \\ 2.\ \ C_A = (A \bullet A^T)/M \\ 3.\ \ C_A = \left[\sum_{i=1}^{M}(x_i - m_x)(x_i - m_x)^T\right] \Big/ M \end{cases} \qquad (1)$$

$A = \{\phi_1, \phi_2, ..., \phi_M\}$，$\phi_i = x_i - m_x$, m_x 是平均人脸，M 训练人脸数，协方差矩阵 C_A 是一个N*N的矩阵, N 是 x_i 的维数。

为了方便计算特征值和特征向量，一般选用第2个公式。根据K - L 变换原理，我们所求的新坐标系即由矩阵 $A \bullet A^T$ 的非零特征值所对应的特征向量组成。直接求N*N大小矩阵 C_A 的特征值和正交归一特征向量是很困难的, 根据奇异值分解原理（见段落1.2.5和1.2.6），可以通过求解 $A^T \bullet A$ 的特征值和特征向量来获得 $A^T \bullet A$ 的特征值和特征向量,。

在计算得到 C_A 的所有非零特征值 $[\lambda_0, \lambda_1, \cdots, \lambda_{r-1}]$（从大到小排序，$1 \le r < M$）及其对应的单位正交特征向量 $[u_0, u_1, \cdots, u_{r-1}]$ 后，可以得到特征空间 $U = [u_0, u_1, \cdots, u_{r-1}] \in \Re^{N*r}$，从而可以计算一张图片X在特征空间上的投影系数（也可以理解为X在空间U中的坐标）：

$$Y = U^T * X \ \in \Re^{r*1} \qquad (2)$$

3. 识别

利用公式（2），首先把所有训练图片进行投影，然后对于测试图片也进行同样的投影，采用判别函数对投影系数进行识别。

1.2 PCA 的理论基础

1.2.1 投影[2]

设 d 维样本 $\mathbf{x}_1$， $\mathbf{x}_2$， $\cdots, \mathbf{x}_n$，以及一个 d 维基 $\mathbf{w}$，那么标量：

$$y_i = \mathbf{w}^T \mathbf{x}_i$$

是相当于 $\mathbf{x}_i$ 在基上的坐标值。如果 $\|\mathbf{w}\|=1$, y_i 就是把 $\mathbf{x}_i$ 向方向为 w 的直线进行投影的结果，可以从图 1 看到。推广之，如果有一组基（m 个）组成的空间 $\mathbf{W}=[\mathbf{w}_1,\mathbf{w}_2,\cdots,\mathbf{w}_m]$，那么可以得到 $\mathbf{x}_i$ 在空间 $\mathbf{W}$ 上的坐标为：$\mathbf{Y}=\mathbf{W}^{\mathbf{T}}\mathbf{x}\in\Re^{m*1}$。

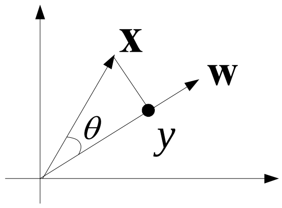

图 1 投影图

证明：$\mathbf{w}^T\mathbf{x}_i=\|\mathbf{w}\|\cdot\|\mathbf{x}\|\cdot\cos\theta$

又$\because$ $\|\mathbf{x}\|\cdot\cos\theta=y$，$\|\mathbf{w}\|=1$

$\Rightarrow \mathbf{w}^T\mathbf{x}_i=y$

进一步，表达式 $\mathbf{w}=\mathbf{m}+a\mathbf{e}$ 表示 $\mathbf{w}$ 是一条通过点 $\mathbf{m}$，方向为 $\mathbf{e}$ 的直线。

1.2.2 PCA 的作用及其统计特性[3]

采用 PCA 对原始数据的处理，通常有三个方面的作用—降维、相关性去除、概率估计。下面分别进行介绍：

- 去除原始数据相关性

从统计学上讲，$E\{[X-E(X)][Y-E(Y)]\}$ 称为随机变量 X 与 Y 协方差，记为 $Cov(X,Y)$。令 $\rho_{XY}=\dfrac{Cov(X,Y)}{\sqrt{D(X)}\sqrt{D(Y)}}$，称为随机变量 X 与 Y 的相关系数。$\rho_{XY}=1$ 则 X 与 Y 是相关的，$\rho_{XY}=0$，则 X 与 Y 是不相关的。

命题 1 对于矩阵 A 来说，如果 AA^T 是一个对角阵，那么 A 中的向量是非相关的。

由 PCA 处理的人脸库数据的非相关性可以从两点进行说明。

（1） 基底的非相关性

特征空间基 $U=[u_0,u_1,\cdots,u_{r-1}]$ 是非相关的，即 $UU^T=I$。

（2） 投影系数的非相关性

由 SVD 可知 $A=\{\phi_1,\phi_2,...,\phi_M\}=U\Lambda V^T$，其中 $\phi_i=x_i-m_x, m_x$ 是平均人脸。根据公式（2）可以把 A 映射到特征空间上，得到：$B=U^T*A$，其中 B 是非相关的，可由下面得到证明：

Y 的协方差矩阵为：$C_B=\dfrac{1}{M}BB^T=\dfrac{1}{M}U^TAA^TU=\dfrac{1}{M}\Lambda^2$ （3）

由命题 1 可知，B 是非相关的。

- 统计参数（均值及方差）

均值即 m_x --平均人脸。

命题 2 随机变量方差越大，包含的信息越多，当一个变量方差为 0 时，该变量为常数，不含任何信息。

用 PCA 计算主分量，就是寻找一组向量，使得原始数据 $A=\{\phi_1,\phi_2,...,\phi_M\}$ 在这组向量上的投影值的方差尽可能大。最大方差对应的向量就是第一主成份，以后递推就是第二主成份，第三主成份……。

用 PCA 计算主分量就是求原始数据 $A=\{\phi_1,\phi_2,...,\phi_M\}$（其中 $\phi_i = x_i - m_x$）协方差矩阵的特征向量 $U=[u_0,u_1,\cdots,u_{r-1}]$，由公式（3）可知，$P=u_i^T A=(p_1,p_2,\cdots,p_m)$ 是 A 在 u_i 上的投影值，其中 P 的方差就是 u_i 对应的特征值 λ_i，可以理解为：

命题 3 所有原始数据在主分量 u_i 上的投影值方差为 λ_i。

- 降维

如果在原始空间表示一幅 n*m 大小的图片 X，那么需要一个 N＝n*m 维矢量，但是当用公式(2)把它映射到特征空间后，只需要一个 r*1 维的向量就可。

另外，由命题 2 可知，可以根据方差的大小来判断特征向量的重要性。由 ORL 图片库的 200 个人脸计算得到的特征值呈图 2 分布，可知特征向量重要性呈指数下降，据此可以只选用前面几个重要的特征向量来构建特征空间。

通过计算，前 71 个特征值占了 90.17%，因此 r 可以取 71 而非 200，从而达到进一步降维的作用。

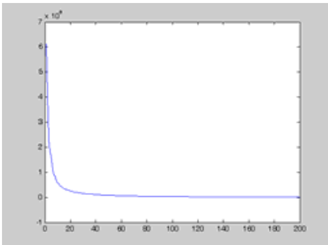

图 2 特征值的分布

1.2.3 特征脸

$U=[u_0,u_1,\cdots,u_{r-1}]$ 中的每一个单位向量都构成一个特征脸，如图 3 所示。由这些特征脸所张成的空间称为特征脸子空间，需要注意对于正交基的选择的不同考虑，对应较大特征值的特征向量(正交基)也称主分量，用于表示人脸的大体形状，而对应于较小特征值的特征向量则用于描述人脸的具体细节，或者从频域来看，

主分量表示了人脸的低频部分，而此分量则描述了人脸的高频部分（源码见’EigenFace.m’）。

图 3 特征脸，分别是第 1，2，10，50，70 分量，最后一张是平均脸。

1.2.4 图片重建

要进行图片 X 的重建，首先对 X 投影到特征空间上，得到系数 $Y=U^T\left(X-m_X\right)$,然后选用一部分系数与特征向量进行原始图片的重建：$X'=m_X+U(1:t)*Y(1:t)$，其中 $1:t$ 表示取前 t 个元素。（见’reconstruct.m’）

在图 4 中，其中前两张图片来自训练样本，第 3 张来自测试样本，可以看到对于训练样本，PCA 系数可以对图片实现很好重建，而对于训练样本以外的图片重建效果很差。

图 4 人脸图像重建。第列张图片是输入原始图，其它列图片是重建结果，数字表示 t 的数目。

1.2.5 奇异值分解（SVD）[1]

设 A 是秩为 r 的 m*n（m>>n）维矩阵，则存在两个正交矩阵和一个对角阵：

$$A=[a_1,a_2,\cdots,a_r]=U\Lambda V^T$$

其中 $U=\left[u_0,u_1,\cdots,u_{r-1}\right]$， $V=\left[v_0,v_1,\cdots,v_{r-1}\right]$， $\Lambda=diag(\lambda_0,\lambda_1,\cdots,\lambda_{r-1})$，且 $UU^T=I$， $VV^T=I$，

λ_i 呈降序排列。其中 λ_i^2 为 $AA^T \in \Re^{m*m}$ 和 $A^T A \in \Re^{n*n}$ 的非零特征值，u_i 和 v_i 分别是 AA^T 和 $A^T A$ 对应于 λ_i^2 的特征向量。可得一个推论：

$$U = AV\Lambda^{-1}$$

可以计算 $A^T A$ 的特征值 λ_i^2 及相应的正交归一特征向量 v_i 后，可由推论知 AA^T 的正交归一特征向量

$$u_i = \frac{1}{\lambda_i} Av_i$$

注意，协方差矩阵 $C_A = (A \bullet A^T)/M$ 的特征值为：λ_i^2 / M 。

1.2.6 利用小矩阵计算大矩阵特征向量

高阶矩阵的特征向量可以转化为求低阶矩阵的特征向量：

设：A 是秩为 r 的 m*n（m>>n）维矩阵，$C_X = AA^T \in \Re^{m*m}$，是一个矩阵，现在要求 C_X 的特征值及特征向量，可通过先求小矩阵 $A^T A \in \Re^{n*n}$ 的特征向量 $\left[v_0, v_1, \cdots, v_{r-1}\right]$ 和特征值 $\left[\lambda_0, \lambda_1, \cdots, \lambda_{r-1}\right]$，两者之间有以下关系：

$$A^T A \cdot v_i = \lambda_i \cdot v_i$$

$$\xrightarrow{\text{左乘A}} AA^T \left(A \cdot v_i\right) = \lambda_i \left(A \cdot v_i\right)$$

显然，$C_X = AA^T$ 的特征向量是 $A \cdot v_i$（注意没有单位化），$\left[\lambda_0, \lambda_1, \cdots, \lambda_{r-1}\right]$ 亦为其特征值。

结论：1.2.5 与 1.2.6 的方法计算协方差矩阵的特征向量，特征值的结果是一致的，只是要注意 1.2.5 中的特征值要除以 M，1.2.6 中的特征向量要单位化。

1.2.7 图片归一化

图片标准化通常是一个整体概念，要求把图片归一到均值为 0，方差为 1 下情况下。
这个概念类似于一般正态分布向标准正态分布的转化：

命题 4 若 $X \sim N(\mu, \sigma^2)$，则 $Z = \dfrac{X-\mu}{\sigma} \sim N(0,1)$

所以要对一组图片中的一张 X_i 进行归一化（标准化），只需要减去均值，除以方差就可以了。

均值 $m_X = \sum_i^M X_i \Big/ M$，方差为 $D = E\left[(X-m_X)(X-m_X)^T\right]$

1.3 参考文献

[1] 邓楠, 基于主成份分析的人脸识别, 西北大学硕士学位论文，2006.06
[2] R.O. Duda, P.E. Hart, and D.G. Stork, Pattern Classification, seconded. John Wiley & Sons, 2001.
[3] Sami Romdhani .Face Image Analysis using a Multiple Feature Fitting Strategy. PhD Thesis, University of Basel, Switzerland, January 2005.
[4] Sami Romdhani .FACE RECOGNITION USING PRINCIPAL COMPONENTS ANALYSIS.

1.4 附录—matlab 源码

1.4.1 人脸识别

```
% FaceRec.m
% PCA 人脸识别修订版，识别率 88%

% calc xmean,sigma and its eigen decomposition
allsamples=[];%所有训练图像
for i=1:40
    for j=1:5
        a=imread(strcat('e:\ORL\s',num2str(i),'\',num2str(j),'.jpg'));
        % imshow(a);
        b=a(1:112*92);   % b 是行矢量 1×N，其中 N＝10304，提取顺序是先列后行，即从上到下，从左到右
        b=double(b);
        allsamples=[allsamples; b];   % allsamples 是一个 M * N 矩阵，allsamples 中每一行数据代表一张图片，其中 M＝200
  end
end
samplemean=mean(allsamples);   % 平均图片，1 × N
for i=1:200 xmean(i,:)=allsamples(i,:)-samplemean;   % xmean 是一个 M × N 矩阵，xmean 每一行保存的数据是“每个图片数据-平均图片”
end;

% 获取特征值及特征向量
sigma=xmean*xmean';    % M * M 阶矩阵
[v d]=eig(sigma);
d1=diag(d);

% 按特征值大小以降序排列
dsort = flipud(d1);
vsort = fliplr(v);
```

```
%以下选择 90%的能量
dsum = sum(dsort);
        dsum_extract = 0;
        p = 0;
        while( dsum_extract/dsum < 0.9)
                p = p + 1;
                dsum_extract = sum(dsort(1:p));
        end
i=1;

% (训练阶段)计算特征脸形成的坐标系
base = xmean' * vsort(:,1:p) * diag(dsort(1:p).^(-1/2));
% base 是 N×p 阶矩阵，除以 dsort(i)^(1/2)是对人脸图像的标准化（使其方差为 1）
% 详见《基于 PCA 的人脸识别算法研究》p31
% xmean' * vsort(:,i)是小矩阵的特征向量向大矩阵特征向量转换的过程
%while (i<=p && dsort(i)>0)
%     base(:,i) = dsort(i)^(-1/2) * xmean' * vsort(:,i);      % base 是 N×p 阶矩阵，除以 dsort(i)^(1/2)
是对人脸图像的标准化（使其方差为 1）
                                                          % 详见《基于 PCA 的人脸识别算法研究》p31
%     i = i + 1;                                  % xmean' * vsort(:,i)是小矩阵的特征向量向大矩阵特
征向量转换的过程
%end

% 以下两行 add by gongxun 将训练样本对坐标系上进行投影,得到一个 M*p 阶矩阵 allcoor
allcoor = allsamples * base;    % allcoor 里面是每张训练人脸图片在 M*p 子空间中的一个点，
即在子空间中的组合系数，
accu = 0;                    % 下面的人脸识别过程中就是利用这些组合系数来进行识别

% 测试过程
for i=1:40
        for j=6:10 %读入 40 x 5 副测试图像
                a=imread(strcat('e:\ORL\s',num2str(i),'\',num2str(j),'.jpg'));
                b=a(1:10304);
                b=double(b);
                tcoor= b * base; %计算坐标，是 1×p 阶矩阵
                for k=1:200
                                mdist(k)=norm(tcoor-allcoor(k,:));
                        end;
                %三阶近邻
        [dist,index2]=sort(mdist);
                class1=floor( (index2(1)-1)/5 )+1;
                class2=floor((index2(2)-1)/5)+1;
                class3=floor((index2(3)-1)/5)+1;
```

```
            if class1~=class2 && class2~=class3
                class=class1;
            elseif class1==class2
                class=class1;
            elseif class2==class3
                class=class2;
            end;
            if class==i
                accu=accu+1;
            end;
        end;
end;
accuracy=accu/200 %输出识别率
```

1.4.2 特征人脸

```
% eigface.m
function [] = eigface()



% calc xmean,sigma and its eigen decomposition
allsamples=[];%所有训练图像
for i=1:40
        for j=1:5
            a=imread(strcat('e:\ORL\s',num2str(i),'\',num2str(j),'.jpg'));
            % imshow(a);
            b=a(1:112*92);   % b 是行矢量 1×N，其中 N＝10304，提取顺序是先列后行，即从上到下，从左到右
            b=double(b);
            allsamples=[allsamples; b];   % allsamples 是一个 M * N 矩阵，allsamples 中每一行数据代表一张图片，其中 M＝200
    end
end
samplemean=mean(allsamples);   % 平均图片，1 × N
for i=1:200 xmean(i,:)=allsamples(i,:)-samplemean;   % xmean 是一个 M × N 矩阵，xmean 每一行保存的数据是“每个图片数据-平均图片”
end;

% 获取特征值及特征向量
sigma=xmean*xmean';     % M * M 阶矩阵
[v d]=eig(sigma);
d1=diag(d);

% 按特征值大小以降序排列
```

```
dsort = flipud(d1);
vsort = fliplr(v);


%以下选择 90%的能量
dsum = sum(dsort);
        dsum_extract = 0;
        p = 0;
        while( dsum_extract/dsum < 0.9)
                p = p + 1;
                dsum_extract = sum(dsort(1:p));
        end

p = 199;

% (训练阶段)计算特征脸形成的坐标系
%while (i<=p && dsort(i)>0)
%       base(:,i) = dsort(i)^(-1/2) * xmean' * vsort(:,i);    % base 是 N×p 阶矩阵，除以 dsort(i)^(1/2)是对人脸图像的标准化，详见《基于 PCA 的人脸识别算法研究》p31
%       i = i + 1;                                % xmean' * vsort(:,i)是小矩阵的特征向量向大矩阵特征向量转换的过程
%end
base = xmean' * vsort(:,1:p) * diag(dsort(1:p).^(-1/2));

% 生成特征脸
for (k=1:p),
            temp = reshape(base(:,k), 112,92);
            newpath = ['d:\test\' int2str(k) '.jpg'];
            imwrite(mat2gray(temp), newpath);
end

avg = reshape(samplemean, 112,92);
imwrite(mat2gray(avg), 'd:\test\average.jpg');

% 将模型保存
save('e:\ORL\model.mat', 'base', 'samplemean');
```

1.4.3 人脸重建

```
% Reconstruct.m
function [] = reconstruct()



load e:\ORL\model.mat;
```

```
% 计算新图片在特征子空间中的系数
img = 'D:\test2\10.jpg'
a=imread(img);


b=a(1:112*92);     % b 是行矢量 1×N，其中 N＝10304，提取顺序是先列后行，即从上到下，从左到右
b=double(b);
b=b-samplemean;

c = b * base;    % c  是图片 a 在子空间中的系数, 是 1*p 行矢量

% 根据特征系数及特征脸重建图

% 前 15 个
t = 15;
temp = base(:,1:t) * c(1:t)';
temp = temp + samplemean';
imwrite(mat2gray(reshape(temp, 112,92)),'d:\test2\t1.jpg');

% 前 50 个
t = 50;
temp = base(:,1:t) * c(1:t)';
temp = temp + samplemean';
imwrite(mat2gray(reshape(temp, 112,92)),'d:\test2\t2.jpg');

% 前 100 个
t = 100;
temp = base(:,1:t) * c(1:t)';
temp = temp + samplemean';
imwrite(mat2gray(reshape(temp, 112,92)),'d:\test2\t3.jpg');

% 前 150 个
t = 150;
temp = base(:,1:t) * c(1:t)';
temp = temp + samplemean';
imwrite(mat2gray(reshape(temp, 112,92)),'d:\test2\t4.jpg');

% 前 199 个
t = 199;
temp = base(:,1:t) * c(1:t)';
temp = temp + samplemean';
imwrite(mat2gray(reshape(temp, 112,92)),'d:\test2\t5.jpg');
```