7

模式识别与机器学习

(Pattern Recognition & Machine Learning)

武汉大学计算机学院　袁志勇

Email: yuanzywhu@163.com

第5章　特征提取和选择

5.1 引言

在模式识别领域，特征的提取与选择是最关键的问题之一，同时也是最困难的问题之一。不同的模式识别应用，需要采用不同的特征提取与选择方法。

对于 实际的模式识别问题，以人脸识别为例，一开始的原始特征可能很多，如在ORL(http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html)人脸数据库中，每幅图像的分辨率为112×92，如果将每个像素作为1维特征，则高达10304维。若把所有的原始特征都作为分类特征送到分类器，不仅使得分类器复杂，分类判别计算量大，而且分类错误概率也不一定小；原始特征的特征空间有很大的冗余，完全可以用很小的空间相当好地近似表示图像，这一点与压缩的思想类似。因此有必要减少特征数目，以获取“少而精”的分类特征，即获取特征数目少且能使分类错误概率小的特征向量。

对特征的要求：

作为识别分类用的特征应具备以下几个条件：

(1)具有很大的识别信息量。即所提供的特征应具有很好的可分性，使分类器容易判别。

(2)具有可靠性。对那些模棱两可，似是而非不易判别的特征应该去掉。

(3)具有尽可能强的独立性。重复的、相关性强的特征只选一个，因为强的相关性并没有增加更多的分类信息，不能要。

(4)数量尽可能少，同时损失的信息尽量小。

模式识别中减少特征数目(或压缩特征空间)的方法有两种：一种是特征提取，另一种是特征选择。

原始特征：通过直接测量得到的特征称为原始特征。比如人体的各种生理指标（描述其健康状况）；数字图像中的各像素点的亮度值（描述图像内容），都是原始特征。

特征提取：通过映射(变换)的方法把高维的特征向量变换为低维的特征向量。

通过特征提取获得的特征是原始特征集的某种组合，即**A:X→Y**，可见新的特征中包含有原有全体特征的信息。

特征选择：从原始特征中挑选出一些最有代表性、分类性能好的特征以达到降低特征空间维数的目的。

也就是说，特征选择就是从已有的**D**个原始特征中挑选出**d**个特征组成一个特征子集，同时将**D-d**个对类别可分离性无贡献的或贡献不大的特征简单地忽略掉。

特征提取与具体问题有很大关系，目前没有理论能给出对任何问题都有效的特征提取方法。由于在许多实际问题中，那些最重要的特征往往不易找到，使得特征选择和特征提取成为构造模式识别系统最困难的任务之一。

如：

- ♦用傅立叶变换或小波变换的系数作为图像的特征；
- ♦指纹的特征；
- ♦统计特征，如矩、灰度共生矩阵[1]等；
- ♦用**PCA**方法作特征压缩；
- ♦用**LDA**方法作特征压缩。

矩的本质是什么

[1]Zhiyong Yuan, Qian Yin et al. Endoscopic Image Classification Based on DWT-CM and Improved BNN for Surgical Tool Appearances. Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, V1, Page(s): 394-397, Hong Kong, August 19-22, 2007.

5.2 基本概念(见教材)

1. 特征的特点

模式识别的主要功能在于利用计算机实现人的类识别能力，它是一个与领域专门知识有关的问题。

研究领域不同，选择的特征也不同，但不论采用什么样的特征，都应该满足如下条件：

(1)特征可以获取

模式识别系统的主要处理设备是计算机，因此作为观察对象的数字化表达，观察对象应该是可以通过数据采集设备输入到计算机的。目前，市场上有各种传感设备和数字化设备，如采集图像信息的图像卡和采集语音信息的声卡等。作为特征，既可以是数字化表达的结果，也可以是在数字化表达基础上形成的参数性质的值，如图像分割后的子目标特征表达等。

(2) 类内稳定

选择的特征对同一类应具有稳定性。 由于模式类是由具有相似特性的若干个模式构成的, 因此它们同属一类模式, 其首要前提是特性相似, 反映在取值上, 就应该有较好的稳定性。

(3) 类间差异

选择的特征对不同的类应该有差异。 若不同类的模式的特征值差异很小，则说明所选择的特征对于不同的类没有什么差异，作为分类的依据时，容易使不同的类产生混淆，使误识率增大。一般来讲，特征的类间差异应该大于类内差异。

2. 特征的类别

特征是用于描述模式性质的一种量，从形式上看可以分为三类：

(1) 物理特征

物理特征是比较直接、人们容易感知的特征，一般在设计模式识别系统时容易被选用。 如为了描述指定班级中的某个学生，可以用以下物理特征：性别、 身高、 胖瘦、肤色等外在特征。 物理特征虽然容易感知，却未必能非常有效地表征分类对象。

(2) 结构特征

结构特征的表达能力一般要高于物理特征，如汉字识别的成功实现离不开结构特征的选择。结构特征的表达是先将观察对象分割成若干个基本构成要素，再确定基本要素间的相互连接关系。

通过要素和相互连接关系表达对象，可以较好地表达复杂的图像信息，在实际中已经有较多的成功应用， 如指纹的识别就是基于结构信息完成的。 结构信息对对象的尺寸往往不太敏感，如汉字识别时，识别系统对汉字大小不敏感，只对笔划结构信息敏感。

结构特征比物理特征要抽象一些，但仍属比较容易感知的特征，如人的指纹特征、 人脸的五官结构信息等，是目前认定人的身份的重要参数。

(3) 数字/数学特征

一般来说，数字特征是为了表征观察对象而设立的特征，如给每个学生设立一个学号，作为标志每个学生的特征。 由于学号是人为设定的，可保证唯一性，但这种特征是抽象的，不容易被人感知。数字特征有时和观察对象的固有特性没有任何联系，有时则是物理特征或结构特征的计算结果。

3. 特征的形成

在设计一个具体的模式识别系统时，往往是先接触一些训练样本，由领域专家和系统工程师联合研究模式类所包含的特征信息，并给出相应的表述方法。 这一阶段的主要目标是获取尽可能多的表述特征。 在这些特征中，有些可能满足类内稳定、 类间离散的要求，有的则可能不满足， 不能作为分类的依据。 根据样例分析得到一组表述观察对象的特征值，而不论特征是否实用，称这一步为特征形成，得到的特征称为原始特征。

在这些原始特征中，有的特征对分类有效，有的则不起什么作用。 若在得到一组原始特征后，不加筛选， 全部用于分类函数确定，则有可能存在无效特征，这既增加了分类决策的复杂度，又不能明显改善分类器的性能。 为此，需要对原始特征集进行处理，去除对分类作用不大的特征，从而可以在保证性能的前提下，通过降低特征空间的维数来减少分类方法的复杂度。

实现上述目的的方法有两种： 特征提取和特征选择。特征提取和特征选择都不考虑针对具体应用需求的原始特征形成过程，而是假设原始特征形成工作已经完成。 然而在实际工作中，原始特征的获得并不容易，因为人具有非常直观的识别能力，有时很难明确描述用于分类的特性依据。 如人脸的判定，人识别脸部特征非常容易，若用计算机来识别人脸，则需要得到多达上千个特征，难度很大。可以说，特征形成是模式识别过程中的重点和难点之一。

4. 特征提取和选择的作用

特征选择是指从一组特征中挑选出对分类最有利的特征，达到降低特征空间维数的目的。

特征提取是指通过映射(或变换)的方法获取最有效的特征，实现特征空间的维数从高维到低维的变换。经过映射后的特征称为二次特征，它们是原始特征的某种组合，最常用的是线性组合。

从定义可以知，实现特征选择的前提是确定特征是否有效的标准，在这种标准下，寻找最有效的特征子集。用于特征选择的特征既可以是原始特征，也可以是经数学变换后得到的二次特征。需要注意，特征提取一定要进行数学变换，但数学变换未必就是特征提取。

特征提取和特征选择的主要目的都是在不降低或很少降低分类结果性能的情况下，降低特征空间的维数，其主要作用在于：

(1) 简化计算。 特征空间的维数越高，需占用的计算机资源越多，设计和计算也就越复杂。

(2) 简化特征空间结构。 由于特征提取和选择是去除类间差别小的特征，保留类间差别大的特征，因此，在特征空间中，每类所占据的子空间结构可分离性更强，从而也简化了类间分界面形状的复杂度。

5.3 类别可分性判据
(特征评判标准)

特征评判标准主要是衡量各类别间的可分性，如使分类器错误概率(误差)最小的那组特征当然是最好的一组特征。从理论上说，这是完全正确的，但在实际应用中存在极大的困难。

因此，希望构造一些更实用、更具有可操作性的评判标准，这些标准应满足以下几点：

(1) 与错误概率(或是错误概率的上、 下界)有单调关系，使判据取极值时对应分类器错误概率较小。

(2)非负性，即：

$$\begin{cases} J_{ij} > 0 & i \neq j \\ J_{ij} = 0 & i = j \end{cases}$$

其中，J_{ij}表示ω_i、ω_j两类间的可分性判据/准则函数。

(3) 对称性, 即：

$$J_{ij} = J_{ji}$$

该特性表明有效性判据对类别号没有方向性，而只强调对区分两类的贡献。

(4) 特征独立时，判据满足可加性，即：

$$J_{ij}(x_1, x_2, \cdots, x_d) = \sum_{k=1}^{d} J_{ij}(x_k)$$

(5) 单调性，当加入新特征时，判据不减少。

$$J_{ij}(x_1, x_2, \cdots, x_d) \leq J_{ij}(x_1, x_2, \cdots, x_d, x_{d+1})$$

下面介绍几种常见的特征评价标准，即类别可分性判据。

1. 基于距离的可分性判据

基于距离的可分性判据直接依靠样本计算，直观简洁，物理概念清晰，因此目前应用较为广泛。基于距离的可分性判据的出发点是：各类样本之间的距离越大、类内离散度越小，则类别的可分性越好。

(1) 两类之间的距离

设两类为 ω_i、 ω_j， 分别有N_i、N_j个样本，即：

$$\omega_i = \{\mathbf{x}_i^i, \mathbf{x}_2^i, \cdots, \mathbf{x}_{N_i}^i\}$$

$$\omega_j = \{\mathbf{x}_1^j, \mathbf{x}_2^j, \cdots, \mathbf{x}_{N_j}^j\}$$

两类之间的平均距离 $D_{\omega_i\omega_j}$ 可由下式定义：

$$D_{\omega_i\omega_j} = \frac{1}{N_i N_j} \sum_{r=1}^{N_i} \sum_{s=1}^{N_j} D(\mathbf{x}_r^i, \mathbf{x}_s^j)$$

其中，$D(x^i_r, x^j_s)$为某种定义下，x^i_r、x^j_s两个模式间的距离。 由点间距离的对称性可知，类间距离也具有对称性。

常用的点间距离有：欧氏距离、马氏距离、绝对距离(城市距离、 **Hamming距离**)、 **Minkowsky**距离等。

Hamming距离定义：

x=(x1,x2,…,xd), **y**=(y1,y2,…,yd),

D(**x**,**y**)=Σ|xi-yi| ；　教材P75 (5-10)

在一个码组集合中，任意两个码字之间对应位上码元取值不同的位的数目定义为这两个码字之间的Hamming距离。即D(**x**,**y**)=∑x[i] ⊕ y[i]，这里i=1,…, d，**x**, **y**都是d位的编码，⊕表示异或。

例如，(00)与(01)的Hamming距离是1，(110)和(101)的Hamming距离是3。

在一个码组集合中，任意两个编码之间Hamming距离的最小值称为这个码组的最小Hamming距离。最小Hamming距离越大，码组抗干扰能力越强。

当取欧氏距离时，两类均方距离为：

$$D^2_{\omega_i\omega_j} = \frac{1}{N_i N_j}\sum_{r=1}^{N_i}\sum_{s=1}^{N_j}(\mathbf{x}_r^i - \mathbf{x}_s^j)^T(\mathbf{x}_r^i - \mathbf{x}_s^j)$$

(2) 各类模式之间的总的平均距离

设N个模式分别属于m类，$\omega_i=\{x^i_k, k=1, 2, \ldots, N_i\}$，$i=1, 2, \ldots, m$，各类模式之间的总的样本平均距离定义为：

$$J(\mathbf{x})=\frac{1}{2}\sum_{i=1}^{m}\tilde{P}(\omega_i)\sum_{j=1}^{m}\tilde{P}(\omega_j)\frac{1}{N_iN_j}\sum_{r=1}^{N_i}\sum_{s=1}^{N_j}D(\mathbf{x}_r^i,\mathbf{x}_s^j) \qquad (5.3.1)$$

其中，$\tilde{P}(\omega_i)$是先验概率$P(\omega_i)$的估计，即：

$$\tilde{P}(\omega_i)=N_i/N,\quad i=1,2,\ldots,m$$

$D(\mathbf{x}_r^i,\mathbf{x}_s^j)$是$\mathbf{x}_r^i$和$\mathbf{x}_s^j$间的距离.

当取欧氏距离时，总的均方距离为

$$J(\mathbf{x}) = D_t^2(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{m}\tilde{P}(\omega_i)\sum_{j=1}^{m}\tilde{P}(\omega_j)\frac{1}{N_i N_j}\sum_{r=1}^{N_i}\sum_{s=1}^{N_j}(\mathbf{x}_r^i - \mathbf{x}_s^j)^T(\mathbf{x}_r^i - \mathbf{x}_s^j)$$

(5.3.2)

(3) 类内离散度矩阵

类内离散度矩阵表示各模式样本在本类的样本均值向量周围散布的情况。设 $\omega_i=\{x^i_k, k=1, 2, \ldots, N_i\}$，$\mu_i$为 ω_i 类的样本均值向量，则类内离散度矩阵为：

$$S_{wi}=\frac{1}{N_i}\sum_{k=1}^{N_i}(\mathbf{x}_k^i-\boldsymbol{\mu}_i)(\mathbf{x}_k^i-\boldsymbol{\mu}_i)^T \qquad (5.3.3)$$

显然有：

$$tr(S_{wi})=\frac{1}{N_i}\sum_{k=1}^{N_i}(\mathbf{x}_k^i-\boldsymbol{\mu}_i)^T(\mathbf{x}_k^i-\boldsymbol{\mu}_i)$$

上式表明，类内离散度矩阵S_{wi}的迹等于该类类内均方欧氏距离.

注：在“矩阵论(Matrix Theory)”中，N×N阶矩阵A的迹记为tr(A)，它定义为A的主对角线的元素之和。

(4) 多类情况下总的类内、类间及总体离散度矩阵

设N个分属m类，$\omega_i=\{x^i_k, k=1, 2, \ldots, N_i\}$, i=1,2,…,m;$S_{wi}$为$\omega_i$类类内离散度矩阵。

总的类内离散度矩阵定义为:

$$S_w=\sum_{i=1}^{m}P(\omega_i)S_{wi}=\sum_{i=1}^{m}P(\omega_i)\frac{1}{N_i}\sum_{k=1}^{N_i}(\mathbf{x}_k^i-\boldsymbol{\mu}_i)(\mathbf{x}_k^i-\boldsymbol{\mu}_i)^T$$

总的类间离散度矩阵定义为:

$$S_b=\sum_{i=1}^{m}P(\omega_i)(\boldsymbol{\mu}_i-\boldsymbol{\mu})(\boldsymbol{\mu}_i-\boldsymbol{\mu})^T=\frac{1}{N}\sum_{i=1}^{m}N_i(\boldsymbol{\mu}_i-\boldsymbol{\mu})(\boldsymbol{\mu}_i-\boldsymbol{\mu})^T$$

总体离散度矩阵定义为:

$$S_t=\frac{1}{N}\sum_{l=1}^{N}(\mathbf{x}_l-\boldsymbol{\mu})(\mathbf{x}_l-\boldsymbol{\mu})^T$$

上面三式中,$P(\omega_i)$为ω_i类的概率,$\boldsymbol{\mu}_i$为ω_i类的样本均值向量,$\boldsymbol{\mu}$为总的样本均值向量,它们分别是如下统计量:

$$P(\omega_i)=\frac{N_i}{N}$$

$$\boldsymbol{\mu}_i=\frac{1}{N_i}\sum_{k=1}^{N_i}\mathbf{x}_k^i$$

$$\boldsymbol{\mu}=\frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i=\sum_{i=1}^{m}P(\omega_i)\boldsymbol{\mu}_i=\frac{1}{N}\sum_{i=1}^{m}\sum_{k=1}^{N_i}\mathbf{x}_k^i$$

可证明:

$$S_t = S_w + S_b$$

$$J(\mathbf{x}) = tr(S_t) = tr(S_w + S_b) \qquad (5.3.4)$$

S_w, S_b和S_t为对称矩阵,而任意对称阵可经正交变换对角化,且对角线上元素为特征值.

由离散度矩阵的定义可知，此时对角线上的元素具有方差、均方距离等含义，且各分量不相关。正交变换为相似变换，变换后矩阵迹不变、行列式值也不变。因此，可以在原特征空间中用S_w、S_b、S_t的迹或行列式构造许多可分性判据。

在概率统计中，μ_i、μ 和S_w、S_b、S_t的定义或公式总结如下(教材P76)：

$$\boldsymbol{\mu}_i = \int \mathbf{x} p(\mathbf{x} \mid \omega_i) d\mathbf{x}$$

$$\boldsymbol{\mu} = E[\mathbf{x}] = \int \mathbf{x} p(\mathbf{x}) d\mathbf{x}$$

$$\mathbf{S}_b = \sum_{i=1}^{m} P(\omega_i)(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T$$

$$\mathbf{S}_w = \sum_{i=1}^{m} P(\omega_i) E_i\left[(\mathbf{x} - \boldsymbol{\mu}_i)(\mathbf{x} - \boldsymbol{\mu}_i)^T\right] = \sum_{i=1}^{m} P(\omega_i) \int (\mathbf{x} - \boldsymbol{\mu}_i)(\mathbf{x} - \boldsymbol{\mu}_i)^T p(\mathbf{x} \mid \omega_i) d\mathbf{x}$$

$$\mathbf{S_t} = \mathbf{S_w} + \mathbf{S_b}$$

为了使所使用的特征能够有效地进行分类，我们希望类间离散度尽量大，同时类内离散度尽量小，从直观上看可以构造下面各种判据：

$$J_1 = \frac{|\mathbf{S}_b|}{|\mathbf{S}_w|}$$

$$J_2 = tr(\mathbf{S}_w^{-1}\mathbf{S}_b)$$

$$J_3 = \ln\left[\frac{|\mathbf{S}_b|}{|\mathbf{S}_w|}\right]$$

$$J_4 = \frac{tr(\mathbf{S}_b)}{tr(\mathbf{S}_w)}$$

$$J_5 = \frac{|\mathbf{S}_w + \mathbf{S}_b|}{|\mathbf{S}_w|}$$

为了有效地分类，它们的值越大越好。

基于距离的可分性判据虽然简单直观，但只是对于类间无重叠的情况效果较好，若类间存在重叠，则效果会受到影响。下面的基于概率的可分性判据能够较好地解决类间有重叠的问题。

2. 基于概率密度函数的可分性判据

基于概率密度函数的可分性判据主要考虑的是两类的概率分布情况。考虑图**5.1**所示两种极端情况，容易看出，图**5.1(a)**中两类是完全可分的，图**5.1(b)**中两类是完全不可分的，两类概率密度函数的重叠程度反映了两类的可分性。因此，可以利用类条件概率密度函数构造可分性判据。

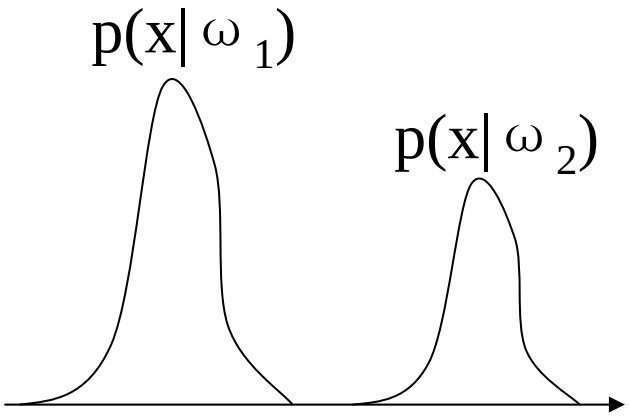


图**5.1(a)**

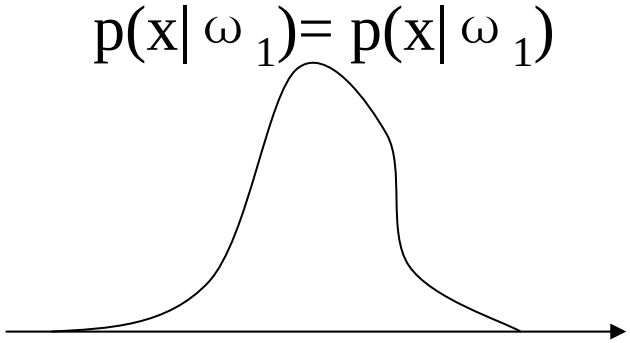


图**5.1(b)**

基于类条件概率密度函数$p(x|\omega_1)$、 $p(x|\omega_2)$的可分性判据J_p满足下面四个条件：

(1) 非负性

$$J_p \geq 0$$

(2) 对称性 ：相对于两个概密具有对称性。

$$J_p[p(\mathbf{x}|\omega_1), p(\mathbf{x}|\omega_2)] = J_p[p(\mathbf{x}|\omega_2), p(\mathbf{x}|\omega_1)]$$

(3) 最大值：当两类完全可分时，J_p具有最大值。

(4) 最小值：当两类完全不可分时，J_p具有最小值，即$J_p=0$。

设两类ω1和ω2的概率密度函数分别为$p(x|\omega 1)$、$p(x|\omega 2)$，$\mathbf{x}=(x_1,x_2,\ldots,x_n)^T$，下面构造基于三个基于概率密距离度量函数的可分性判据。

(1)巴氏(Bhattacharyya)判据J_B

Bhattacharyya判据计算式定义：

$$J_B = -\ln \int [p(\mathbf{x}|\omega_1)p(\mathbf{x}|\omega_2)]^{\frac{1}{2}} d\mathbf{x}$$

在最小错误概率判决准则下，最小错误概率**Pe**为：

$$P_e \le [P(\omega_1)P(\omega_2)]^{\frac{1}{2}} \exp(-J_B)$$

(见教材P78，证明此略)

(2) 切诺夫(Chernoff)判据J_C

Chernoff判据定义为：

$$J_c = -\ln \int p^s(\mathbf{x}\,|\,\omega_1)\,p^{1-s}(\mathbf{x}\,|\,\omega_2)d\mathbf{x} \qquad s \in [0,1]$$

由定义式可见，当**s=1/2**时，Chernoff界限距离就是Bhattacharyya距离。

一般情况下J_C的计算比较困难，当ω_1、 ω_2的类条件概率密度函数都是正态分布，即$p(x|\omega 1)\sim N(\mu_i, \Sigma_i)$和$p(x|\omega 2)\sim N(\mu_j, \Sigma_j)$时，可以推导出：

$$J_c = \frac{1}{2}s(1-s)(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T[(1-s)\boldsymbol{\Sigma}_i + s\boldsymbol{\Sigma}_j]^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) + \frac{1}{2}\ln\left|\frac{|(1-s)\boldsymbol{\Sigma}_i + s\boldsymbol{\Sigma}_j|}{|\boldsymbol{\Sigma}_i|^{1-s}|\boldsymbol{\Sigma}_j|^s}\right|$$

当$\Sigma = \Sigma_i = \Sigma_j$时：

$$J_c = \frac{1}{2}s(1-s)(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)$$

(3) 散度J_D

在最小错误率Bayes决策中，对于两类的分类问题，最大后验概率判决准则可以通过似然比$p(x|\omega 1)/p(x|\omega 2)$和阈值$p(\omega 2)/p(\omega 1)$的比较实现，显然似然比对于分类来说是一个重要的度量。 对于给定的阈值$p(\omega 2)/p(\omega 1)$，$p(\omega 1|x)/p(\omega 2|x)$越大， 对类$\omega 1$来讲可分性越好，该比值反映了两类类条件概率密度函数的重叠程度。

为了保证概率密度函数完全重叠时判据为零，应对该比值取对数。 于是，可构造出**D-判据**J_D。

ω**1**类相对于 ω**2**类的平均可分性信息定义为：

$$I_{12} = E\left[\ln \frac{p(\mathbf{x}\mid\omega_1)}{p(\mathbf{x}\mid\omega_2)}\right] = \int p(\mathbf{x}\mid\omega_1)\ln \frac{p(\mathbf{x}\mid\omega_1)}{p(\mathbf{x}\mid\omega_2)}d\mathbf{x}$$

ω_2类相对于 ω_1类的平均可分性信息定义为：

$$I_{21} = E\left[\ln \frac{p(\mathbf{x}\mid\omega_2)}{p(\mathbf{x}\mid\omega_1)}\right] = \int p(\mathbf{x}\mid\omega_2)\ln \frac{p(\mathbf{x}\mid\omega_2)}{p(\mathbf{x}\mid\omega_1)}d\mathbf{x}$$

对于 ω_1和 ω_2两类总的平均可分性信息称为散度，其定义为：

$$J_D = I_{12} + I_{21}$$

$$= \int [p(\mathbf{x}|\omega_1) - p(\mathbf{x}|\omega_2)] \ln \frac{p(\mathbf{x}|\omega_1)}{p(\mathbf{x}|\omega_2)} d\mathbf{x}$$

从数学构造上看，上式是合理的，式中被积函数两概密之差和两概密之比能反映两概密的重叠程度，同时被积函数中两因式总是同号，故其乘积非负。

3. 基于熵函数的可分性判据

由信息论知，对于一组概率分布而言，分布越均匀，平均信息量越大，分类的错误概率越大；分布越接近0-1分布，平均信息量越小，分类的错误概率越小，可分性越好。 因此，可以建立基于熵函数的可分性判据，其中熵函数表征平均信息量。

（具体内容不要求，此略）

5.4 基于可分性判据的特征提取

设有n个原始特征构成的特征向量$\mathbf{x}=(x_1, x_2, \ldots, x_n)^{\mathrm{T}}$，特征提取就是对$\mathbf{x}$作线性变换，产生$d$维向量$\mathbf{y}=(y_1, y_2, \ldots, y_d)^{\mathrm{T}}$，$d \leqslant n$，即：

$$\mathbf{y} = \boldsymbol{W}^{\mathrm{T}}\mathbf{x}$$

式中，$\boldsymbol{W}=\boldsymbol{W}_{n\times d}$称为特征提取矩阵或简称变换矩阵，$\mathbf{y}$称为二次特征。

基于可分性判据的特征提取就是在一定的准则函数(可分性判据)下，如何求最优的变换矩阵$\boldsymbol{W}$。

1. 基于距离可分性判据的特征提取方法

前面研究了基于距离的可分性判据，得到了相应判据，它们都反映了一个基本思想，即类内距离小和类间距离大的要求。下面我们以$J2$准则($J_2 = tr(\mathbf{S}_w^{-1}\mathbf{S}_b)$)为例讨论特征提取的方法。

设S_w和S_b为原始特征空间的总的类内离散度矩阵和类间离散度矩阵，S_w^*和S_b^*为变换后特征空间的总的类内离散度矩阵和总的类间离散度矩阵，W为变换矩阵。则有：

$$\mathbf{S}_w^* = \mathbf{W}^T \mathbf{S}_w \mathbf{W}$$

$$\mathbf{S}_b^* = \mathbf{W}^T \mathbf{S}_b \mathbf{W}$$

为什么不是W^TS_W ?????????

在变换域中，J_2为：

$$J_2(\mathbf{W}) = tr[(\mathbf{S}_w^*)^{-1}\mathbf{S}_b^*] = tr[(\mathbf{W}^T\mathbf{S}_w\mathbf{W})^{-1}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})]$$

若$\mathbf{W}$为非奇异矩阵, 可得$tr[(\mathbf{S}_w^*)^{-1}\mathbf{S}_b^*] = tr[\mathbf{S}_w^{-1}S_b]$, J_2是不变的。

对矩阵作相似变换特征值不变, 其行列式值不变, 其迹不变, 一个方阵的迹等它的所有特征值之和。

设$\mathbf{W}_e$标准正交阵，用W_e对对称阵$\mathbf{S}_w^{-1}S_b$作相似变换使其成为对角阵：

$$\mathbf{W}_e^{-1}\mathbf{S}_w^{-1}S_b\mathbf{W}_e = \mathbf{W}_e^{T}\mathbf{S}_w^{-1}S_b\mathbf{W}_e = \begin{pmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{pmatrix}$$

其中，λ_i(i=1, 2, . . . , n)为$\mathbf{S}_w^{-1}S_b$的特征值，$\mathbf{W}_e$的列向量$\mathbf{w}_i$为$\mathbf{S}_w^{-1}S_b$相应于λ_i的特征向量。

设此处W_e的列向量的排列已作适当调整，使矩阵$\boldsymbol{S}^{-1}{}_w\boldsymbol{S}_b$的特征值$\lambda_1 \geqslant \lambda_2 \geqslant \ldots \geqslant \lambda_n$。

由此可得出，在d给定后，取前d个较大的特征值所对应的特征向量$\mathbf{w}_i(i=1,2,\ldots,d)$构造特征提取矩阵，即：

$$\mathbf{W}=(\mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_d)$$

对$\mathbf{x}$作变换$\mathbf{y}=\mathbf{W}^T\mathbf{x}$，这时对于给定的$d$所得到的$J_2^*$

$$J_2^*(\mathbf{W})=\sum_{i=1}^{d}\lambda_i$$ 达到最大值

此法对于J_4判据也适用。

d的给定有规律可循吗？还是通过猜测或是遍历？

设矩阵$S^{-1}_w S_b$的特征值为 $\lambda_1, \lambda_2, \ldots, \lambda_n$, 按大小顺序排列为：$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$

相应的正交化、归一化的特征向量为：

$$\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_n$$

选前d个特征向量作为变换矩阵：

$$\mathbf{W} = \left[\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_d\right]_{n \times d}$$

2. 基于概率密度函数可分性判据的特征提取方法

基于概率密度函数的可分性判据的方法需要知道各类的概率密度函数的解析形式，难度较大，计算量也较大。一般地，只有当概率密度函数为某些特殊的函数形式时才便于使用，这里只研究多元正态分布的两类问题。

对于基于概率密度函数可分性判据的特征提取方法而言，通常选用的变换仍为线性变换，设n维原始特征向量$\mathbf{x}$经线性变换后的二次特征向量为$\mathbf{y}$，即：

$$\mathbf{y} = \mathbf{W}^T \mathbf{x}$$

在映射后的特征空间内建立某种准则函数，使得它为变换矩阵$\mathbf{W}$的函数：

$$J_c = J_c(\mathbf{W})$$

其中，J_c为基于概率密度函数的可分性判据，如前面介绍的**Bhattacharyya距离**和**Chernoff 距离**等可分性判据。通过求解判据的极值点即可得到使映射后的特征组可分性最好的变换矩阵。 在$J_c(\mathbf{W})$可微的情况下，就是求解偏微分方程：

$$\frac{\partial J_c(\mathbf{W})}{\partial \mathbf{W}} = \mathbf{0}$$

这里以**Chernoff**距离为例，分析特征提取方法。 当两类都是正态分布时，两类的分布函数分别为：

$$p(\mathbf{x}\,|\,\omega_1)=\frac{1}{(2\pi)^{n/2}\left|\sum_1\right|^{1/2}}\exp[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_1)^T\sum_1^{-1}(\mathbf{x}-\boldsymbol{\mu}_1)]$$

$$p(\mathbf{x}\,|\,\omega_2)=\frac{1}{(2\pi)^{n/2}\left|\sum_2\right|^{1/2}}\exp[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_2)^T\sum_2^{-1}(\mathbf{x}-\boldsymbol{\mu}_2)]$$

变换后的判据J_c是$\mathbf{W}$的函数，记为$J_c(\mathbf{W})$，根据P78式(5-37)，有：

$$J_C(\mathbf{W}) = \frac{1}{2}s(1-s)tr\{\mathbf{W}^T\mathbf{M}\mathbf{W}[(1-s)\mathbf{W}^T\boldsymbol{\Sigma}_1\mathbf{W} + s\mathbf{W}^T\boldsymbol{\Sigma}_2\mathbf{W}]^{-1}\}$$

$$+\frac{1}{2}\ln\left|(1-s)\mathbf{W}^T\boldsymbol{\Sigma}_1\mathbf{W} + s\mathbf{W}^T\boldsymbol{\Sigma}_2\mathbf{W}\right|$$

$$-\frac{1}{2}(1-s)\ln\left|\mathbf{W}^T\boldsymbol{\Sigma}_1\mathbf{W}\right| - \frac{1}{2}s\ln\left|\mathbf{W}^T\boldsymbol{\Sigma}_2\mathbf{W}\right|$$

式中，$\boldsymbol{M}=(\mu_1-\mu_2)(\mu_1-\mu_2)^{\mathrm{T}}$。

因为$J_c(\mathbf{W})$是标量，可以对$\mathbf{W}$的各个分量求偏导，并令其为零，经简化可得矩阵方程：

$$\mathbf{MW}-[(1-s)\boldsymbol{\Sigma}_1\mathbf{W}+s\boldsymbol{\Sigma}_2\mathbf{W}][(1-s)\mathbf{W}^T\boldsymbol{\Sigma}_1\mathbf{W}+s\mathbf{W}^T\boldsymbol{\Sigma}_2\mathbf{W}]^{-1}\mathbf{W}^T\mathbf{MW}$$
$$+\boldsymbol{\Sigma}_1\mathbf{W}[I-(\mathbf{W}^T\boldsymbol{\Sigma}_1\mathbf{W})^{-1}\mathbf{W}^T\boldsymbol{\Sigma}_2\mathbf{W}]+\boldsymbol{\Sigma}_2\mathbf{W}[\mathbf{I}-(\mathbf{W}^T\boldsymbol{\Sigma}_2\mathbf{W})^{-1}\mathbf{W}^T\boldsymbol{\Sigma}_1\mathbf{W}]=\mathbf{0}$$

上式是关于$\mathbf{W}$的非线性方程，只能采用数值优化的方法得到近似最优解。

在以下两种特殊情况下可以得到最优的解析解。

(1) $\Sigma_1 = \Sigma_2 = \Sigma$, $\mu_1 \neq \mu_2$

在此种情况下，最优特征提取矩阵是由 $\Sigma^{-1}M$矩阵的特征向量构成的。 又因为矩阵M的秩为1，故 $\Sigma^{-1}M$只有一个非零特征值，对应于特征值为零的那些特征向量对$J_c(\mathbf{W})$没有影响，因此可以舍去，所以最优变换$\mathbf{W}$是 $\Sigma^{-1}M$的非零特征值对应的特征向量v，不难得到：

$$\mathbf{W} = v = \Sigma^{-1}(\mu_1 - \mu_2)$$

上面结果与Fisher线性判别式的解相同。

(2) $\Sigma_1 \neq \Sigma_2,\ \mu_1 = \mu_2$

在此种情况下，最优特征矩阵**W***是由 $\Sigma^{-1}_2 \Sigma_1$满足下列关系的前**d**个特征值所对应的特征向量构成的，此时J_c(W)取最大值。

$$(1-s)\lambda_1^s + s\lambda_1^{s-1}$$
$$\geq (1-s)\lambda_2^s + s\lambda_2^{s-1}$$
$$\geq \ldots \geq (1-s)\lambda_n^s + s\lambda_n^{s-1}$$

5.5 主成份分析
(PCA：Principal Components Analysis)

主成份分析是一种有效的特征线性变换方法，也称为*K*－*L*变换/主分量变换/Hotelling变换 (PCA)。*K*－*L*变换是一种基于目标统计特性的最佳正交变换，它的最佳性体现在变换后产生的新的分量正交或不相关。

K-L变换分连续和离散两种情况，这里只讨论离散K-L变换法。

设n维随机向量$\mathbf{x}=(x_1, x_2, \ldots, x_n)^{\mathrm{T}}$，$\mathbf{x}$经标准正交矩阵$\boldsymbol{A}$正交变换后成为向量$\mathbf{y}=(y_1, y_2, \ldots, y_n)^{\mathrm{T}}$，即：

$$\mathbf{y} = \mathbf{A}^T\mathbf{x} \qquad (5.5.1)$$

$\mathbf{y}$的自相关矩阵为：

$$\mathbf{R}_{\mathrm{y}} = E(\mathbf{y}\mathbf{y}^T) = E[\mathbf{A}^T\mathbf{x}\mathbf{x}^T\mathbf{A}] = \mathbf{A}^T\mathbf{R}_{\mathrm{x}}\mathbf{A}$$

其中，$\boldsymbol{R}_{\mathrm{x}}$为$\mathbf{x}$的自相关矩阵，即$\boldsymbol{R}_{\mathrm{x}}=\boldsymbol{E}(\mathbf{x}\mathbf{x}^{\boldsymbol{T}})$，是对称矩阵。

选择矩阵$A=(a_1, a_2, \ldots, a_n)$，且满足：

$$\mathbf{R}_x\mathbf{a}_i = \lambda_i\mathbf{a}_i$$

这里，λ_i为自相关矩阵$\boldsymbol{R}_x$的特征值，并且

$\lambda_1 \geqslant \lambda_2 \geqslant \ldots \geqslant \lambda_n$，$a_i$为$\lambda_i$的正交基向量(特征向量)，即

$a^T_i a_j=1(i=j)$，$a^T_i a_j=0(i\neq j;\ i, j=1, 2, \ldots, n)$。 $\boldsymbol{R}_y$是对角矩阵：

$$\mathbf{R}_y = \mathbf{A}^T\mathbf{R}_x\mathbf{A} = diag(\lambda_1, \lambda_2, \cdots, \lambda_n)$$

若$\boldsymbol{R}_x$是正定的，则它的特征值是正的。此时变换式$\mathbf{y}=\mathbf{A}^T\mathbf{x}$称为**K-L变换**。

由式(5.5.1)可得：

$$\mathbf{x} = (\mathbf{A}^T)^{-1}\mathbf{y} = \mathbf{A}\mathbf{y} = (\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_n)\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \sum_{i=1}^{n} y_i \mathbf{a}_i$$

选择**x**关于$\boldsymbol{a_i}$的展开式的前$\boldsymbol{d}$项在最小均方误差准则下估计**x**，这时估计式表示为：

$$\hat{\mathbf{x}} = \sum_{i=1}^{d} y_i \mathbf{a}_i \ , \quad (1 \le d \le n)$$

估计的均方误差为：

$$\varepsilon^2(d) = E\left[(\mathbf{x}-\hat{\mathbf{x}})^T(\mathbf{x}-\hat{\mathbf{x}})\right]$$

$$= \sum_{i=d+1}^{n} E[y_i^2] = \sum_{i=d+1}^{n} E[y_i\, y_i^T]$$

$$= \sum_{i=d+1}^{n} \mathbf{a}_i^T E[\mathbf{x}\mathbf{x}^T]\mathbf{a}_i = \sum_{i=d+1}^{n} \lambda_i$$

希望选择使估计的均方误差最小的特征向量，因此要选择相关矩阵$\boldsymbol{R}_{\mathbf{x}}$的$d$个最大的特征值对应的特征向量构成变换矩阵$\mathbf{A}$，这样得到的均方误差将会最小，是$n-d$个极小特征值之和。 可以证明，与$d$维向量中的其他$x$逼近值相比，这个结果是最小均方误差解。 这就是K-L变换也称为主分量分析(PCA)的原因。

基于K-L变换/主成份分析的特征提取步骤：

设$\mathbf{x}$是n维模式向量，$\{\mathbf{x}\}$是来自m个模式类的样本集，总样本数为N.

利用K-L变换将$\mathbf{x}$变换为d维$(d < n)$向量$\mathbf{y}$的具体方法如下：

(1)平移坐标系，将模式总体的均值向量作为新坐标系的原点；

(2)求出自相关矩阵$\mathbf{R}_{\mathbf{x}} = E(\mathbf{x}\mathbf{x}^T) \approx \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i\mathbf{x}_i^T$；

(3)求出$\mathbf{R}_{\mathbf{x}}$的特征值$\lambda_1, \lambda_2, ..., \lambda_n$及其对应的特征向量$\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_n$；

(4)将特征值从大到小排序，如：

$$\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_n$$

取前d个大的特征值所对应的特征向量构成变换矩阵

$$\mathbf{A} = (\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_d)$$

(5)将n维的原向量变换成d维的新向量$\mathbf{y} = \mathbf{A}^T\mathbf{x}$

例：已知模式样本数据：

$$\begin{pmatrix}-5\\-5\end{pmatrix},\begin{pmatrix}-5\\-4\end{pmatrix},\begin{pmatrix}-4\\-5\end{pmatrix},\begin{pmatrix}-5\\-6\end{pmatrix},\begin{pmatrix}-6\\-5\end{pmatrix},\begin{pmatrix}5\\5\end{pmatrix},\begin{pmatrix}5\\6\end{pmatrix},\begin{pmatrix}6\\5\end{pmatrix},\begin{pmatrix}5\\4\end{pmatrix},\begin{pmatrix}4\\5\end{pmatrix}$$

试用**K-L**变换作一维数据降维处理。

解：(1)求样本总体均值向量

$$\bar{\mathbf{x}} = \frac{1}{10}\left[\begin{pmatrix}-5\\-5\end{pmatrix}+\begin{pmatrix}-5\\-4\end{pmatrix}+\ldots+\begin{pmatrix}4\\5\end{pmatrix}\right]=\begin{pmatrix}0\\0\end{pmatrix}$$

故无需作坐标系平移。

(2) 求自相关矩阵$\mathbf{R}_x$

$$\mathbf{R}_x = \frac{1}{10}\left[\begin{pmatrix}-5\\-5\end{pmatrix}\begin{pmatrix}-5 & -5\end{pmatrix}+\ldots+\begin{pmatrix}4\\5\end{pmatrix}\begin{pmatrix}4 & 5\end{pmatrix}\right]$$

$$=\begin{pmatrix}25.4 & 25.0\\25.0 & 25.4\end{pmatrix}$$

(3) 求$\mathbf{R}_x$的特征值及其对应的特征向量

$$\left|\lambda\mathbf{I}-\mathbf{R}_x\right|=\begin{vmatrix}25.4-\lambda & 25.0\\ 25.0 & 25.4-\lambda\end{vmatrix}=0$$

即: $(25.4-\lambda)^2-25.0^2=0,$

解得特征值: $\lambda_1=50.4,\qquad \lambda_2=0.4$。

由$\mathbf{R}_x\mathbf{a}_i=\lambda_i\mathbf{a}_i\,(i=1,2)$, 可解出特征向量为

$$\mathbf{a}_1=\frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix}\quad,\mathbf{a}_2=\frac{1}{\sqrt{2}}\begin{pmatrix}1\\-1\end{pmatrix}$$

(4) 取$\mathbf{a}_1$作变换矩阵$\mathbf{A}$

$$\mathbf{A}=\mathbf{a}_1$$

(5) 将原二维样本变为一维样本$\mathbf{y}=A^T\mathbf{x}$:

$$\left(-\frac{10}{\sqrt{2}}\right),\left(-\frac{9}{\sqrt{2}}\right),\left(-\frac{9}{\sqrt{2}}\right),\left(-\frac{11}{\sqrt{2}}\right),\left(-\frac{11}{\sqrt{2}}\right),$$

$$\left(\frac{10}{\sqrt{2}}\right),\left(\frac{11}{\sqrt{2}}\right),\left(\frac{11}{\sqrt{2}}\right),\left(\frac{9}{\sqrt{2}}\right),\left(\frac{9}{\sqrt{2}}\right)$$

本例K-L变换MALAB程序：

```
%Filename:ex7_1.m
X=[-5 -5
   -5 -4
   -4 -5
   -5 -6
   -6 -5
    5  5
    5  6
    6  5
    5  4
    4  5];
[m1 n1]=size(X);
m=mean(X);          %均值向量
```

```
for i=1:m1
   X(i,:)=X(i,:)-m;     %把样本数据平移到m为中心的坐标系下
end
R=zeros(n1);
for i=1:m1
R=X(i,:)'*X(i,:)+R;
end
R=R/10;    %样本X的自相关矩阵R
[A,D]=eig(R) %计算矩阵R的特征值对角阵D及其对应的特征向量矩阵A
%将原二维样本变换成一维样本(这里简化了按特征值排序等操作)
if D(1,1)>D(2,2)   y=A(:,1)'*X'
else y=A(:,2)'*X'
end
```

运行结果：

A =

-0.7071	**0.7071**
0.7071	**0.7071**

D =

0.4000	**0**
0	**50.4000**

y =

Columns 1 through 9

-7.0711 -6.3640 -6.3640 -7.7782 -7.7782 7.0711
7.7782 7.7782 6.3640

Column 10

6.3640

MATLAB中使用函数princomp (Principal Components Analysis)实现了对PCA的封装。

princomp常见调用形式：

[COEFF, SCORE, latent]=princomp(X)

参数说明：

X：原始样本矩阵，其中的1行表示1个样本，其中的1列表示样本特征向量的1维；

COEFF：主成份分量，即样本自相关矩阵的特征向量a_1~a_n；

SCORE：主成份，X的低维表示，即X中的数据在主成份分量上的投影a_1~a_d；

latent：一个包含样本自相关矩阵特征值的向量$[\lambda_1 \sim \lambda_d]^T$。

例：使用princomp作主分量分析。

```
%Filename:ex7_2.m
%使用princomp函数作主成份分析
X=[-5 -5
   -5 -4
   -4 -5
   -5 -6
   -6 -5
    5  5
    5  6
    6  5
    5  4
    4  5];
[A,SCORE,latent]=princomp(X);%主成份分析
A  %A主成份分量, 即A中的各列为样本X自相关矩阵的特征向量a1~a2
SCORE  %SCORE主成份,SCORE(:,1)为X的一维表示, SCORE为X在变换空间的二维表示
latent %X样本自相关矩阵的特征值
```

运行结果：

A =

0.7071 0.7071
0.7071 -0.7071

SCORE =

-7.0711	-0.0000
-6.3640	-0.7071
-6.3640	0.7071
-7.7782	0.7071
-7.7782	-0.7071
7.0711	-0.0000
7.7782	-0.7071
7.7782	0.7071
6.3640	0.7071
6.3640	-0.7071

latent =

56.0000
0.4444

上面主分量分析采用的是样本自相关矩阵，也可以采用样本协方差矩阵作主分量分析(**MATLAB**中使用函数 **pcacov: Principal Components Analysis using a covariance matrix)**。

基于样本协方差矩阵的主分量分析在图像分析中使用较广泛。

设有n×d维样本矩阵**X**(n个样本,每个样本有d维特征).

样本协方差矩阵定义:

$$\mathrm{cov}(\mathbf{X})=\frac{1}{n-1}\sum_{i=1}^{n}(\vec{x}_i-\vec{m})(\vec{x}_i-\vec{m})^T$$

样本散布矩阵/离散度矩阵定义:

$$S=\sum_{i=1}^{n}(\vec{x}_i-\vec{m})(\vec{x}_i-\vec{m})^T$$

例：PCA在人脸识别中的应用。

在人脸识别中，PCA是一种常用的特征提取方法。设一幅$p\times q$大小的人脸图像，可以将它看成是一个矩阵$(f_{ij})_{p\times q}$，f_{ij}为图像在该点的灰度(亮度)。若将该矩阵按列相连构成一个$p\times q$维向量$\mathbf{x}=(f_{11}, f_{21}, \cdots, f_{p1}, f_{21}, f_{22}, \cdots, f_{p2}, \cdots, f_{1q}, f_{2q}, \cdots, f_{pq})^{\mathrm{T}}$。设训练样本集为 $X=\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$，包含N幅图像。

N幅图像的协方差矩阵为：

$$\mathbf{R} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^T$$

其中，$\overline{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$

求出矩阵***R***的前***d***个最大特征值 $\lambda_1, \lambda_2, \ldots, \lambda_d$及其对应的正交化、归一化特征向量 $\alpha_1, \alpha_2, \ldots, \alpha_d$。分别将这***d***个特征向量化为**p×q**矩阵，得到**d**幅图像，称为“特征脸”**(eigenface)**。

下图显示的是对应前30个最大特征值的特征向量的图像(具体实现方法见本章5.7节)。

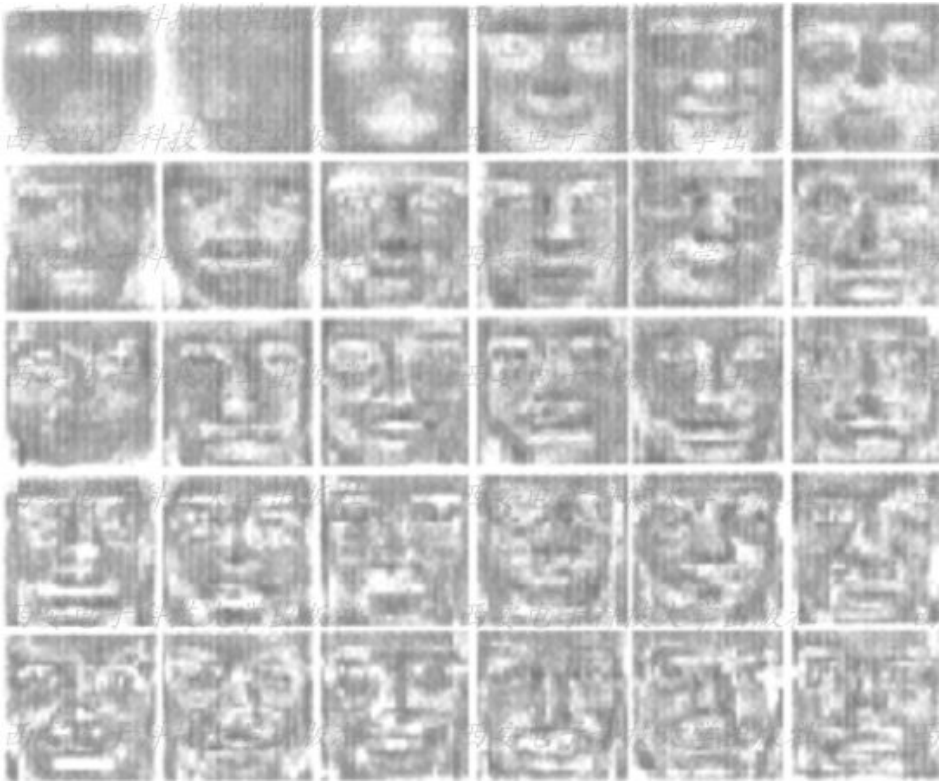


图　“特征脸”图像

将每一幅人脸图像投影到由$a_1, a_2, \ldots, a_d$张成的子空间中，对应于该子空间的一个点，该点的坐标系数对应于图像在子空间的位置，可以作为识别人脸的依据。 对于任意待识别样本$\mathbf{x}$，可通过向“特征脸”子空间投影获得系数向量$\mathbf{y}=(a_1, a_2, \ldots, a_d)^{\mathrm{T}}\mathbf{x}$。

点评：一幅人脸图像往往是由较多的像素构成的，如果以每个像素作为1维特征，将得到一个维数相当高的特征向量，计算将十分困难；而且这些像素之间通常具有相关性。这样，利用PCA技术（或其它特征提取技术）在降价维数的同时在一定程度上去除原始特征各维之间的相关性自然成为一个较理想的解决方案。

补充：人脸图像预处理方法(*)

研究和设计人脸识别算法，通常需要引用国际上的一些标准人脸库，ORL人脸库是其中的一种。

1.ORL人脸库(ORL Database of Faces)简介

ORL人脸数据库由英国剑桥大学的AT&T实验室采集。

(1)ORL数据库共有400幅人脸图像(40人，每人10幅，大小为112像素×92像素，灰度级为256级)。

(2) 该数据库比较规范，大多数图像的光照方向和强度都差不多。但有少许表情、姿势、伸缩的变化，眼睛对得不是很准，尺度差异在10%左右。

(3)不是每个人都有所有的这些变化的图像，即有些人姿势变化多一点，有些人表情变化多一点，有些还戴有眼睛，但这些变化都不大。

正是基于ORL人脸库图像在光照，以及关键点如眼睛、嘴巴的位置等方面比较规范，一些实验可在该图像集上直接展开（可以省略归一化等图像预处理过程，但归一化后的图像识别率会更高一些）。在用ORL进行人脸识别算法研究时，通常采用一部分图像做训练集(如选用每个人的前5张图像作为训练集，这样40个人共有200幅样本图像)，剩下的另外一部分图像做测试集(如选用每个人的后5张图像作为测试集，这样40个人共有200幅待测试图像)。

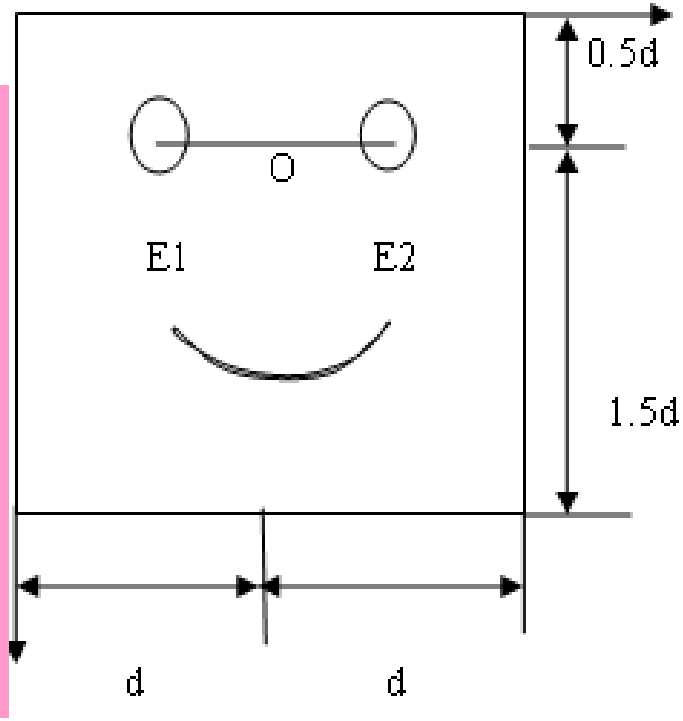
ORL人脸数据库下载网站：

http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

2.人脸图像的预处理

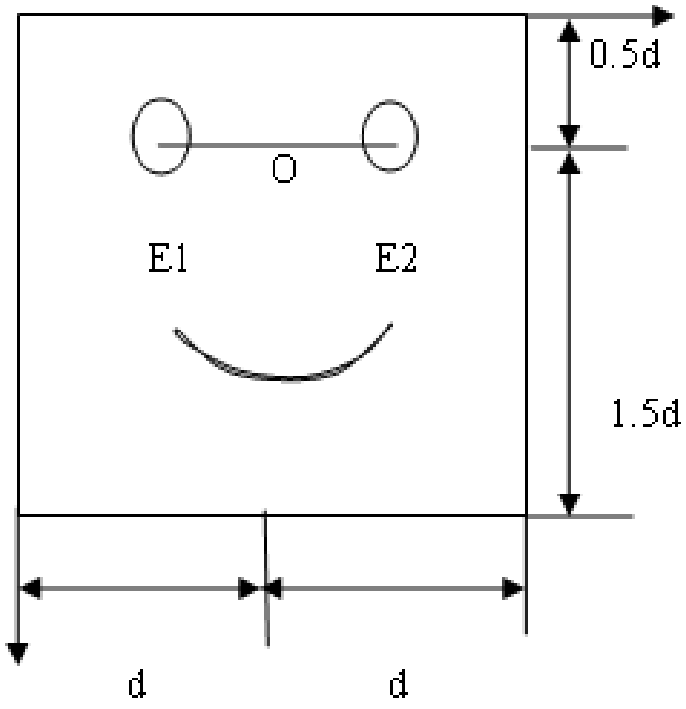
对于不规范的人脸图像，在进行特征提取之前必须对其作预处理(如去噪、几何归一化)。由于两眼之间的距离对大多数人来说都是基本相同的，因此可采用两眼的位置作为几何归一化的依据。

假设人脸图像中两只眼睛的位置分别是**E1**和**E2**(右图所示)。通过如下三个步骤，便能实现人脸的几何归一化。即输入一幅人脸图像，通过旋转、裁剪、缩放，便得到一个双眼水平、间距为**d**，图像比例为**2d×2d**大小，大小为**32×32**的统一规范化图像。



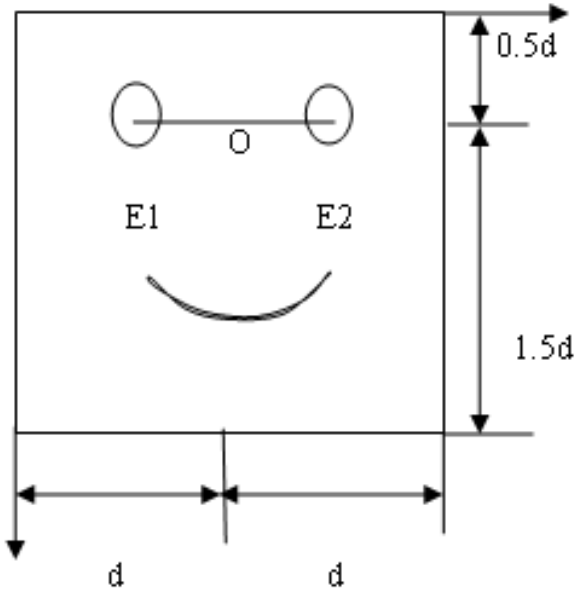
(1)图像旋转

经过图像旋转。使**E1**和**E2**的边线 **E1E2**保持水平，以保证人脸方向的一致性。根据是人脸在平面内的旋转不变性，在这之前所做的工作是双眼定位，可通过人机交互方式用鼠标点击两只眼的位置(或采用算法对人眼进行自动定位，但人眼自动定位难度较复杂，这里采用人机交互方式)。



(2)图像裁减

根据图所示的比例关系，进行图像裁剪，图中**O**为**E1E2**的中点，且假定为**E1E2=d**。经过裁剪后，在**2d×2d** 的图像内，保证**O**点固定于**(d, 0.5d)**处，这样就保证了人脸位置的一致性。根据是人脸在图像平面内的平移不变性。



在第(1)步旋转过程中，两眼位置确定之后，就能够求得**O**点的坐标；但旋转会改变**O**点的位置，因此先将图像的中心移到**O**点，然后进行旋转。这样，旋转之后，**O**点还是图像的中心。

(3)图像缩放

进行图像的缩放变换。得到统一大小的标准图像。统一规定的图像的大小是**32×32**像素点，即使**d=E1E2**为定长（**16**个像素点），缩放倍数β**=2d/32**，这样就保证了人脸图像的大小一致性。根据是人脸在图像平面内的尺度不变性。

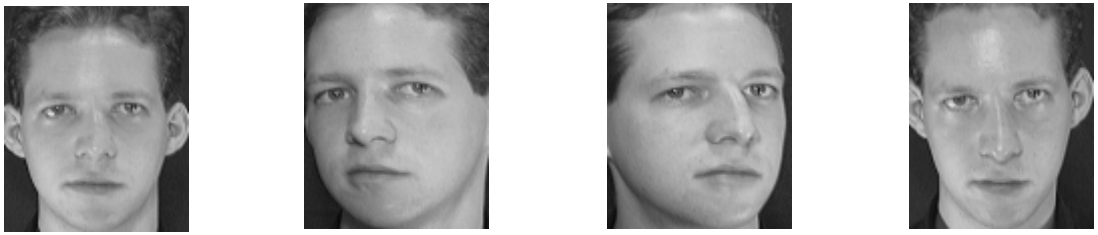


图 a) ORL人脸库中的4幅原始图像

图b) 图a)的几何归一化图像

人脸图像的预处理MATLAB编程举例:

orlface_preprocessing.m程序的作用是对所读入的人脸图像作几何归一化。在本例程序中，读入文件夹‘D:\FACE_PCA\ORL_FACES\s1\’中的x.pgm(x=1~10)人脸原始图像文件，并对其作几何归一化处理，归一化处理后的图像保存在‘D:\FACE_PCA\ORL_FACES\ss1\’文件夹下，文件名为 x.pgm文件名(x=1~10)。

程序清单:

```
%文件名:orlface_preprocessing2.m
%功能:人脸图像的预处理
%第0步:提取ORL人脸库上每幅人脸图像眼睛位置坐标，保存在eyelocs中
%两眼坐标位置是通过人机交互方式点击鼠标得到的
```

```
clear all;
close all;
clc;
sstring='D:\FACE_PCA\ORL_FACES\s1\';

directory_list=dir([sstring '*.pgm']);
nfiles=length(directory_list);
axis ij;
I=imread([sstring directory_list(1).name]);

Images=zeros(size(I,1),size(I,2),nfiles);
```

```
for i=1:nfiles
   Images(:,:,i)=imread([sstring directory_list(i).name]);
end

eyelocs=zeros(nfiles,4);

for i=1:nfiles
   clf;  % Clear current figure
   imagesc(Images(:,:,i));
   hold on
   [x1,y1]=ginput(1);
   plot(x1,y1,'rx');
   [x2,y2]=ginput(1);
   plot(x2,y2,'rx');
   drawnow;
   eyelocs(i,:)=[x1,y1,x2,y2];
   hold on
   pause;
end
```

```
eyedistances=sqrt((eyelocs(:,1)-eyelocs(:,3)).^2+...
   (eyelocs(:,2)-eyelocs(:,4)).^2);
%save -ascii eyedistances.dat eyedistances

%第1步:图像旋转, 使两眼之间的连线保持水平
dx=eyelocs(:,3)-eyelocs(:,1);
dy=eyelocs(:,4)-eyelocs(:,2);
rotation=atan(dy./dx)*180/pi;
eyex=round((eyelocs(:,1)+eyelocs(:,3))./2);
eyey=round((eyelocs(:,2)+eyelocs(:,4))./2);
S=size(Images);
center=round(S(1:2)./2);
In1=round(eyedistances);
In2=round(eyedistances./2);

nsize=32;
Io=zeros(nsize,nsize,nfiles);
```

```
for i=1:nfiles
    if(eyex(i)<center(2))
        Ib=[zeros(S(1),2*(center(2)-eyex(i))),Images(:,:,i)];
    else
        Ib=[Images(:,:,i),zeros(S(1),2*(eyex(i)-center(2)))];
    end
    if(eyey(i)<center(1))
        Ib=[zeros(2*(center(1)-eyey(i)),S(2)+2*abs(center(2)-eyex(i)));Ib];
    else
        Ib=[Ib;zeros(2*(eyey(i)-center(1)),S(2)+2*abs(center(2)-eyex(i)))];
    end

    Ib=imrotate(Ib,rotation(i),'nearest');
```

```
%第2步:图像裁减(对每幅图像进行裁剪)
  %[c1,c2]=round(size(Ib)./2);
  c1=round(size(Ib,1)/2);
  c2=round(size(Ib,2)/2);

  x1=c2-In1(i);
  y1=c1-In2(i);
  x2=c2+In1(i);
  y2=c1+In1(i)+In2(i);

  Iob=Ib(y1:y2,x1:x2);
  %第3步:图像缩放，得到32*32的几何归一化图像
  Io(:,:,i)=imresize(Iob,[nsize,nsize],'nearest');
end
```

```
Io=mat2gray(Io);

for i=1:nfiles
   imwrite(Io(:,:,i),[sstring 'n' directory_list(i).name]);
end
mkdir('D:\FACE_PCA\ORL_FACES\','ss1');
%建立第i个人的文件夹ssi,用于保存第i个人的归一化图像(这里假设为
第1个人)
sstring2='D:\FACE_PCA\ORL_FACES\ss1\';
%存储第i个人的归一化图像的文件路径字符串(这里假设为第1个人)
for i=1:nfiles
   imwrite(Io(:,:,i),[sstring2 directory_list(i).name]);
end
```

修改程序中的下面三句:

(1)sstring='D:\FACE_PCA\ORL_FACES\s1\';

(2)mkdir('D:\FACE_PCA\ORL_FACES\','ss1');

(3)sstring2='D:\FACE_PCA\ORL_FACES\ss1\';

将数字**1**换成其他的数字**i(i=2~40)**，并重新运行程序，**40**次运行程序后即可完成所有人脸图像的预处理。

K-L变换小结：

K-L变换适用于任何概率分布，它是在最小均方误差为准则进行数据压缩，是在最小均方误差意义下的最优正交解。**K-L**变换是一种常用的特征提取方法，适用于任意概率密度函数，能对应地保留原样本中方差最大的数据分量，在消除模式特征之间相关性、突出差异性方面有最优的效果，因此**K-L**变换也称为主分量分析或主成份分析。

5.6 快速PCA及其实现 (fast PCA)

PCA的计算中最主要的工作量是计算样本协方差矩阵的特征值和特征向量。设样本矩阵X大小为n×d(n个d维样本特征向量)，则样本协方差矩阵S (离散度矩阵/散布矩阵是样本协方差矩阵的n-1倍，scatter matrix) 将是一个d ×d的方阵，故当维数d较大时计算复杂度会极高。如，当维数d=10000，S是一个10000 ×10000维的矩阵，此时如果采用前面的princomp函数计算主成份，MATLAB通常会出现内存耗尽的错误，即使有足够的内存，要得到S的全部特征值也可能要花费数小时的时间。

1.快速PCA的基础理论

当样本散布矩阵**S**的维数**d**较大时，计算复杂度会极高。有一个非常好的技巧可以用来计算矩阵**S**非零特征值所对应的特征向量。

设$Z_{n\times d}$为样本矩阵X中的每个样本减去样本均值$\vec{m}$后得到的矩阵，则散布矩阵S为$(Z^TZ)_{d\times d}$. 现在考虑矩阵$R=(ZZ^T)_{n\times n}$，很多情况下,由于样本数目n远远小于样本维数d(如人脸图像)，R的大小也远远小于散布矩阵S，然而，它与S有着相同的非零特征值.

设n维列向量$\vec{v}$是R的特征向量，则有：

$$(ZZ^T)\vec{v}=\lambda\vec{v} \qquad (5.6.1)$$

将(5.6.1)式两边同时左乘Z^T，并应用矩阵乘法的结合律得：

$$(Z^TZ)(Z^T\vec{v})=\lambda(Z^T\vec{v}) \qquad (5.6.2)$$

式(5.6.2)说明$(Z^T\vec{v})$为散布矩阵$S=(Z^TZ)_{d\times d}$的特征向量. 这说明可以先计算小矩阵$R=(ZZ^T)_{n\times n}$的特征向量$\vec{v}$, 然后通过左乘Z^T得到散布矩阵$S=(Z^TZ)_{d\times d}$的特征向量$Z^T\vec{v}$.

2. 快速PCA的MATLAB实现

下面编写**fastPCA**函数用来对样本矩阵**A**进行快速主成份分析和降维（降至**k**维），其输出**pacA**为降维后的**k**维样本特征向量组成的矩阵，每行一个样本，列数**k**为降维后的样本特征维数，相当于**princomp**函数中的输出**SCORE**，而输出**V**为主成份分量，即**princomp**函数中的**COEFF**。**fastPCA**函数代码清单如下：

```
function [pcaA V] = fastPCA( A, k )
% Filename: fastPCA.m
% 快速PCA
%
```

```
% 输入：A - 样本矩阵，每行为一个样本
%       k - 降维至 k 维
%
% 输出：pcaA - 降维后的 k 维样本特征向量组成的矩阵，每行一个样本，列数 k 为降维后的样本特征维数
%       V - 主成份向量

[r c] = size(A);

% 样本均值
meanVec = mean(A);

% 计算协方差矩阵的转置 covMatT
Z = (A-repmat(meanVec, r, 1));
covMatT = Z * Z';
```

```
% 计算 covMatT 的前 k 个特征值和特征向量
[V D] = eigs(covMatT, k);

% 得到协方差矩阵 (covMatT)' 的特征向量
V = Z' * V;

% 特征向量归一化为单位特征向量
for i=1:k
   V(:,i)=V(:,i)/norm(V(:,i));
end

% 线性变换（投影）降维至 k 维
pcaA = Z * V;

% 保存变换矩阵 V 和变换原点 meanVec
save('d:\face_Pca\Mat/PCA.mat', 'V', 'meanVec');
```

fastPCA函数实现中调用了MATLAB库函数eigs来计算矩阵$\mathbf{R}=(\mathbf{ZZ}^{\mathrm{T}})_{n\times n}$的前$k$个特征向量，即对应于最大的$k$个特征值的特征向量，其调用形式为：

[V, D]=eigs(R, k)

入口参数：

-**R**为要计算特征值和特征向量；

-**k**为要计算的特征向量数目；

返回值：

-输出矩阵$\mathbf{V}_{n\times k}$的每列对应1个特征向量，k个特征向量从左到右排列；

-对角矩阵$\mathbf{D}_{k\times k}$对角线上的每个元素对应一个特征值。

5.7 基于PCA的人脸特征提取

本节使用PCA技术进行人脸特征提取。一幅人脸照片往往由比较多的像互构成，如果以每个像素作为1维特征，将得到一个维数非常高的特征向量，计算将十分困难；而且这些像素之间通常具有相关性。因此，利用PCA技术在降低维数的同时在一定程序上去除原始特征各维之间的相关性自然成为一种特征提取的解决方案。

1. 人脸数据集及人脸图像预处理

本案例使用的人脸数据集为ORL人脸库(40人，每人10幅图像，共400幅112×90大小的图像)，人脸图像预处理及实现方法见5.5节中的介绍。经过人脸图像预处理后，生成400幅32×32大小的图像。

采用5.5节的人脸图像预处理程序所生成的40个人的图像文件保存于‘D:\FACE_PCA\ORL_FACES\ssi‘文件夹下(i=1~40， ssi表示存放第1个人到第40个人的文件目录)，如‘D:\FACE_PCA\ORL_FACES\ss1‘文件夹下保存的是第1人的10幅32×32图像(文件名为1.pgm~10.pgm)。

经过人脸图像预处理后，我们选用每个人的前5幅图像作为实验的数据集，这样40个人共有200幅样本图像。

2. 生成样本矩阵

生成样本矩阵所要做的工作是将这200幅32×32人脸图像转换成向量的形式，从而组成样本矩阵。可编写一个函数ReadFaces实现此功能。

ReadFaces依次读入样本图像(这里假设40个的样本图像位于’D:\FACE_PCA\ORL_FACES\’路径下，如第18个人的10幅图像位于’D:\FACE_PCA\ORL_FACES\ss18’中)，然后将32×32像素的图像按列存储为一个1024的行向量作为样本矩阵FaceContainer中的一体样本(一行)，最后将样本矩阵保存至Mat目录下的FaceMat.mat文件。

ReadFaces函数实现代码：

```
function [imgRow,imgCol,FaceContainer,faceLabel]=ReadFaces(nFacesPerPerson, nPerson, bTest)
% Filename: ReadFaces.m
% 读入ORL_FACES预处理后的32*32人脸库中指定数目的人脸前五张(训练)
%
% 输入：nFacesPerPerson - 每个人需要读入的样本数，默认值为 5
%                  nPerson - 需要读入的人数，默认为全部 40 个人
%                    bTest - bool型的参数。默认为0，表示读入训练样本（前5张）；如果为1，表示读入测试样本（后5张）
%
% 输出：FaceContainer - 向量化人脸容器，nPerson * 1024 的2维矩阵，每行对应一个人脸向量
```

```
if nargin==0 %default value
    nFacesPerPerson=5; %前5张用于训练
    nPerson=40;
    %要读入的人数(每人共10张，前5张用于训练)
    bTest = 0;
elseif nargin < 3
    bTest = 0;
end

img=imread('ORL_FACES/ss1/1.pgm');
%为计算图像大小先读入一张
[imgRow,imgCol]=size(img);
FaceContainer = zeros(nFacesPerPerson*nPerson, imgRow*imgCol);
faceLabel = zeros(nFacesPerPerson*nPerson, 1);
```

```
% 读入训练数据
for i=1:nPerson
    i1=mod(i,10); % 个位
    i0=char(i/10);
    strPath='ORL_FACES/ss';
    if( i0~=0 )
        strPath=strcat(strPath,'0'+i0);
    end
    strPath=strcat(strPath,'0'+i1);
    strPath=strcat(strPath,'/');
    tempStrPath=strPath;
```

```
for j=1:nFacesPerPerson
        strPath=tempStrPath;

        if bTest == 0 % 读入训练数据
            strPath = strcat(strPath, '0'+j);
        else
            strPath = strcat(strPath, num2str(5+j));
        end

        strPath=strcat(strPath,'.pgm');
        img=imread(strPath);

 %把读入的图像按列存储为行向量存到向量化人脸容器faceContainer的
对应行中
        FaceContainer((i-1)*nFacesPerPerson+j, :) = img(:)';
        faceLabel((i-1)*nFacesPerPerson+j) = i;
    end % j
end % i

% 保存预处理的人脸样本矩阵
save('d:\face_Pca\Mat/FaceMat.mat', 'FaceContainer')
```

3. 主成份分析

经过上面的处理后，矩阵FaceContainer每一行就成为一个代表某个人脸样本的特征向量。通过主成份分析的方法可将这些1024维的样本特征降维至低维(这里降维为20维)。这样数据集中每个人脸样本都可以由一个20维的特征向量来表示，以作为后续分类所采用的特征，如作为神经网络的输入。在本节结果的基础上还可采用支持向量机(SVM)对这些20维的人脸样本进行分类，从而设计并实现一种人脸识别原型系统。

为什么是20维？？

对样本矩阵FaceContainer进行主成份分析的全部过程封装在Main_FacePCA函数中，其参数k是主分量的数目，即降维至k维。 Main_FacePCA函数首先调用ReadFaces函数获得人脸样本矩阵FaceContainer，然后利用上一节中的fastPCA算法计算出样本矩阵的低维表示LowDimFaces和主成份分量矩阵W，并将LowDimFaces保存至Mat目录下的LowDimFaces.mat文件中。

Main_FacePCA函数实现代码：

```
function Main_FacePCA(k)
% main_facepca.m
% 基于ORL人脸数据集的主成份分析
%
% 输入：k - 降至 k 维
% 定义图像高、宽的全局变量 imgRow 和 imgCol，它们在 ReadFaces 中被赋值
global imgRow;
global imgCol;

% 读入每个人的前5副图像
nPerson=40; %最大40（即40个人）
nFacesPerPerson = 5;
display('读入人脸数据...');
[imgRow,imgCol,FaceContainer,faceLabel]=ReadFaces(nFacesPerPerson,nPerson);
display('.............................');
```

```
nFaces=size(FaceContainer,1);%样本（人脸）数目
display('PCA降维...');
% LowDimFaces是200*20的矩阵, 每一行代表一张主成份脸/特征脸(共40人，每人5张)，每个脸20个维特征
% W是分离变换矩阵, 1024*20 的矩阵
[LowDimFaces, W] = fastPCA(FaceContainer, 20);
% 主成份分析PCA
visualize_pc(W);%显示主成份脸
save('d:/Face_Pca/Mat/LowDimFaces.mat', 'LowDimFaces');
display('计算结束。');
```

通过下面命令可完成对Main_FacePCA函数的调用，将人脸样本向量降至20维。

```
%将工程所在文件夹FACE_PCA添加到系统路径表
>>addpath(genpath('d:\face_pca'));
>>Main_FacePCA(20); %提取前20个主成份，即降维至20维
```

上述命令运行后，会在MAT目录下生成LowDimFaces.mat文件，其中的200×20维矩阵LowDimFaces是经过PCA降维后，原样本FaceContainer的低维表示。200个人脸样本所对应的每一个特征向量由原来的1024维变成了20维，这就将后续的分类问题变成了一个在20维空间中的划分问题，过程大大简化。

4. 主成份脸/特征脸可视化

fastPCA函数的另一个输出是为主分量矩阵W，它是一个1024×20的矩阵，每列是一个1024维的主分量(样本协方差矩阵的特征向量)，在人脸分析中，习惯上称之为主成份脸或特征脸。事实上，可以将这些列向量以32×32(或更大)的分辨率来显示，此工作由visualize_pc函数实现。

visualize_pc函数实现代码：

```
function visualize_pc(E)
% Filename: visualize_pc.m
% 显示主成份分量（主成份脸/特征脸，即变换空间中的基向量）
%
% 输入：E - 矩阵，每一列是一个主成份分量


[size1 size2] = size(E);
global imgRow;
global imgCol;
row = imgRow;
col = imgCol;
```

```
if size2 ~= 20
   error('只用于显示 20 个主成份');
end;

figure
img = zeros(row, col);
for ii = 1:20
    img(:) = E(:, ii);
    subplot(4, 5, ii);
    imshow(img, []);
end
```

上面函数调用运行后，20个主成份脸如下图所示。从图中不难理解为什么主分量会被称为主成份。

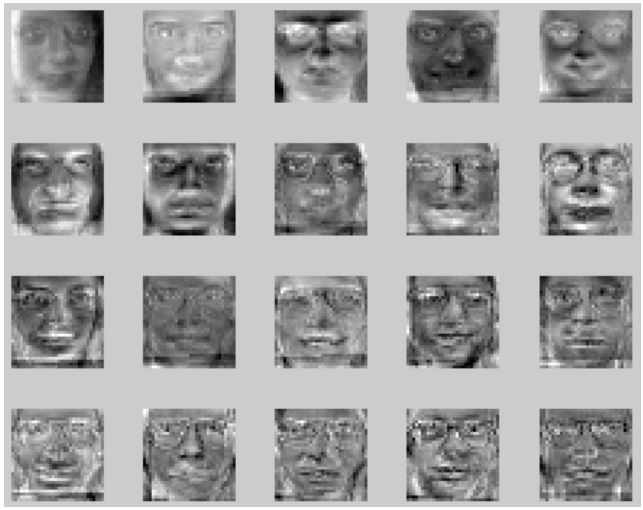


图 20个主成份脸/特征脸

特征选择与特征提取课堂思考题：

1. 简述特征选择和特征提取的异同。

2. 简述基于距离可分性判据的特征提取方法。

3. 简述基于概率密度函数可分性判据的特征提取方法。

4. 已知一组数据的协方差矩阵为 $\begin{bmatrix} 1 & 1/2 \\ 1/2 & 1 \end{bmatrix}$

试问：

(1) 协方差矩阵中各元素的含义是什么？

(2) K-L变换的最佳准则是什么？

(3) 为什么说经K-L变换后消除了各分量之间的相关性？

5. 若有下列两类样本集：

ω **1: x1=(0,0,0)^T, x2=(1,0,0)^T,x3=(1,0,1)^T,x4=(1,1,0)^T**

ω **2: x5=(0,0,1)^T,x6=(0,1,0)^T,x7=(0,1,1)^T,x8=(1,1,1)^T**

要求用**K-L**变换法，分别把特征空间维数降到**d=2**和**d=1**。试编写满足要求的**MATLAB**程序。

参考答案：

1.特征选择是指,从L个度量值$(x_1, x_2, ..., x_L)$中按一定的准则选择出供分类用的子集,作为降维(m维,$m < L$)的分类特征。

特征提取是指, 使一组度量值$(x_1, x_2, ..., x_L)$通过某种变换$T_i(.)$产生新的m个特征$(y_1, y_2, ..., y_m)$作为降维的分类特征,这里$i = 1, 2, ..., m; m < L.$

注意，特征选择是“挑选”出较少的特征用于分类，特征提取是通过“数学变换”产生较少的特征。它们都是为了在尽可能保留识别信息的前提下，降低特征空间的维数，以实现有效的分类。

特征选择和特征提取有时并不是截然分开的，如，可以先进行特征选择，从原始测量数据中去掉那些明显没有分类信息的特征，然后再进行特征提取，进一步降低维数。

2.假设有n个原始特征：$X=[x_1,x_2,...,x_n]^T$，希望通过线性映射压缩为d个特征$Y=[y_1,y_2,...,y_d]^T$，其变换关系为$Y=W^TX$，W为$n\times d$矩阵。

令S_w，S_b为原空间(即X的)离散度矩阵，S_w^*,S_b^*为映射后(即Y的)离散度矩阵：$S_b^*=W^TS_bW$，$S_w^*=W^TS_wW$，经变换后的J_2变为

$$J_2(W)=tr[(S_w^*)^{-1}S_b^*]=tr[(W^TS_wW)^{-1}(W^TS_bW)]$$

将上式对W的各个分量求偏导数并令其为零即可以确定一个W值。

3.假设有n个原始特征：$X=[x_1,x_2,...,x_n]^T$，希望通过线性映射压缩为d个特征$Y=[y_1,y_2,...,y_d]^T$，其变换关系为$Y=W^TX$，W为$n\times d$矩阵。

求出变换后的概率距离判别函数J_p，将此函数对W的各个分量求偏导数并令其为零可以确定一个W值。

4. 答：已知协方差矩阵 $\begin{bmatrix} 1 & 1/2 \\ 1/2 & 1 \end{bmatrix}$ 则：

(1) 其对角元素是各分量的方差，非对角元素是各分量之间的协方差。
(2) K-L变换的最佳准则为：对一组数据按一组正交基进行分解，在只取相同数量分量的条件下，以均方误差计算截尾误差最小。
(3) 在经K-L变换后，协方差矩阵成为对角矩阵，因而各主分量间的相关消除。

5. 参考本章的MATLAB编程例题，答案此略。