

基于特征脸的人脸检测与识别^{*}

齐怀峰, 接 标, 冯乔生

(云南师范大学计算机科学与技术学院, 云南 昆明 65009)

摘 要: 人脸识别是属于生物特征识别范畴, 然而与其他识别不同的是人脸是一个动态的图像, 有位置与光照的变化, 并且具有相似的外貌特征, 这使得人脸识别成为一个比较困难的问题. 文章阐述了基于特征脸的人脸检测与识别方法. 该方法本质上是主要成分分析方法, 他首先构造特征脸向量, 然后计算新图像和数据库中特征脸的空间距离, 来决定此图像是否是一副人脸图像, 如果是人脸图像, 他是那一副人脸图像.

关 键 词: 人脸识别; 特征脸; 空间距离

中图分类号: TP391. 41 **文献标识码:** A **文章编号:** 1007—9793(2005)06—0022—02

近二十年来, 很多研究者对人脸识别进行了深入的研究, 并提出了很多有效的方法. 其中基于特征脸的人脸识别是一个行之有效的方法. 该方法首先由 Sirovich 和 Kirby 于 80 年代末提出的, 此方法后来被 M. Turk 和 A. Pentland^[1] 改进. 该方法的主要工作原理是, 首先对人脸样本库的每一副人脸图像构造其特征脸向量, 即利用特征脸向量去表示人脸库中的每一个样本. 对于一个新的样本, 计算他和人脸样本库的空间距离, 可以判断他是否是一副人脸图像, 如果是, 他是那一副图像.

该方法的主要过程如下:

1 建立特征向量空间

1.1 算法目标

任何一个图像可以利用一个 1 向量进行表示.

假设 Γ 是一个 $N^2 \times 1$ 的向量, 他对应于某一幅 $N \times N$ 的人脸图 I .

我们的目标是 Γ 可以利用一组特征向量进行表示.

$$\Gamma = \sum_{i=1}^K W_i Y_i + \text{const} \quad (K \ll N^2)$$

1.2 构建特征脸(EigenFaces)空间

(1) 获得一组大小一样并且人脸在图像的中心训练的图像, 用 I_1, I_2, \dots, I_M 表示.

(2) 利用向量 Γ_i 去表示 I_i . ($1 \leq i \leq M$)
($\Gamma_i: N^2 \times 1; I_i: N \times N$)

(3) 计算人脸的平均特征 Ψ : $\Psi = \frac{1}{M} \sum_{i=1}^M \Gamma_i$
($\Psi: N^2 \times 1$) (4) 减去平均脸: $\Phi = \Gamma - \Psi$. ($\Phi: N^2 \times 1$)

(5) 计算协方差矩阵(covariance matrix) C

$$C = \sum_{n=1}^M \Phi_n \Phi_n^T = A A^T \quad (C: N^2 \times N^2)$$

$$A = [\Phi_1, \Phi_2, \dots, \Phi_M] \quad (A: N^2 \times M)$$

(6) 计算协方差矩阵 C 的特征向量 U_i . 我们注意到 C 是 $N^2 \times N^2$ 的矩阵, 一个 500×500 像素的人脸库此时将有 250000 维的特征向量. 计算如此大的特征向量在实时图像处理中是不可接受的, 因此我们采取另外一种方法来计算.

我们来考虑 $A^T A$ ($M \times M$) 矩阵, M 是用来训练的样本数量, 一般情况下有 ($M \ll N^2$).

$$A^T A V_i = \mu_i V_i \Rightarrow A A^T (A V_i)$$

$$= \mu_i (A V_i) \Rightarrow C(A V_i) = \mu_i (A V_i)$$

令: $U_i = A V_i$ 得: $C U_i = \mu_i U_i$. 那么 U_i 就是 C 的特征向量.

^{*} 收稿日期: 2005—05—31

作者简介: 齐怀峰(1979—), 男, 山东省聊城市人, 研究生, 主要研究方向为模式识别, 神经网络.

通讯作者: 冯乔生, 云南省昆明市人, 研究生导师, 副教授, 主要研究方向为机器视觉与虚拟现实.

这样我们就求得了 C 的 M 个特征向量, 虽然不能求出 C 的 N^2 个特征向量, 但是我们在训练时常常不用 M 个特征向量, 而是用 K ($K \ll M$) 个. 在选取 K 时的原则时, 选取特征值比较大的前 K 个^[2].

2 利用特征向量表示人脸

每一个人脸样本可以表示为一个特征脸向量 (K 个) 的线性组合.

$$\mathbb{X}^i = \sum_{j=1}^K W_j U_j$$

(U^j 是 C 的特征向量; $W^j = U^{jT} \Phi^i$, W^j 是一个实数)
此时 Φ 被 Ω^i 线性表示. $\Omega^i = [W_1, W_2, \dots, W_K]^T$, $i = 1, 2, \dots, K$.

3 人脸检测 对于一幅未知图像 Γ , 采用如下方法进行检测

- (1) 计算 $\Phi = \Gamma - \Psi$.
- (2) 把他映射到向量空间 $\mathbb{X} = \sum_{i=1}^K W_i U_i$ (U^i 是 C 的特征向量; $W^i = U^{iT} \Phi$, W^i 是一个实数)
- (3) 计算 $E_d = ||\Phi - \mathbb{X}||$, 如果 $E_d < T_d$, 这幅图像是一幅人脸图像, 否则不是.

4 识别人脸

- (1) 对于一幅人脸图像, 首先进行规范化: Γ

: $\Phi = \Gamma - \Psi$.
(2) 将其映射到特征脸空间中去.
$$\mathbb{X} = \sum_{i=1}^K W_i U_i$$
 (U^i 是 C 的特征向量; $W^i = U^{iT} \Phi$, W^i 是一个实数)
此时 Φ 被 Ω 线性表示. $\Omega = [W_1, W_2, \dots, W_K]^T$, $i = 1, 2, \dots, K$.
寻找最小的误差: $E_r = \text{Min} (||\Omega - \Omega_l||)$, (l 是人脸数据库的样本数).

如果 $E_r < T_r$ (T_r 是所规定的误差界限) 此人脸图像是数据库中的第 r 幅人脸图像.
在其他文献中^[3] 指出误差距离的另外一个有效的计算公式 Mahalanobis Distance.

5 结束语

特征脸的检测与识别方法对于正面的人脸图像有较高的识别率, 但在识别人脸的侧面图像, 存在光照的人脸图像, 识别率不是太高, 因此有必要对该方法进一步的改进.

参 考 文 献:

[1] M. Turk and A. Pentland. Eigenfaces for Recognition[J]. Journal of Cognitive Neuroscience, 1991, 3 (1): 71—86.
[2] Erik Hjelm and Boon Kee Low. Face Detection: A Survey[J]. <http://www.idealibrary.com>.
[3] Matthew A. Turk and Alex P. Pentland. Face Recognition Using Eigenfaces[J]. Vision and Modeling Group, The Media Laboratory, Massachusetts Institute of Technology.

Eigenfaces for Face Detection And Recognition

QI Huai-feng, JIE Biao, Feng Qiao-sheng

(Department of Computer Science and Information Technology, Yunnan Normal University, Kunming 650092 China)

ABSTRACT: Face Recognition is part of biology identification. However face images are dynamic images with pose and illumination variety and all the images have similar features, which make it a difficult problem. This paper presents eigenfaces for face detection and recognition, which in deed is principle component analysis method. This method first construct eigenfaces vectors for every images in face database, second computes space distance with new image for every database face images to decide it is or not a face image, and if it is which image it belong to.

KEY WORDS: Face Recognition; Eigenfaces; space distance