

基于图谱分解的人脸表情分析

孔 敏^{1,2}, 陈思宝¹, 罗 斌¹

(1. 安徽大学 计算机科学与技术学院, 安徽 合肥 230039;

2. 皖西学院 机械与电子工程系, 安徽 六安 237012)

摘 要:文中探索用人脸的几何结构图谱方法嵌入到模式空间来分析人脸表情,以图的加权邻接矩阵主要特征向量来定义矩阵的特征模。计算谱特征向量-模间邻接矩阵。用两类模式向量在范数下的多维尺度变换方法(MDS)嵌入该向量到一个模式空间,用人脸特征点来表示人脸图,并在模式空间里描述该嵌入方法下的同一人脸的不同表情。

关键词:图谱;人脸识别;表情分析;多维尺度变换

中图分类号:TP391.41

文献标识码:A

文章编号:1005-3751(2006)04-0033-02

Facial Expression Analysis Based on Graph Spectral Decomposition

KONG Min^{1,2}, CHEN Si-bao¹, LUO Bin¹

(1. School of Computer Science and Technology, Anhui University, Hefei 230039, China;

2. Machine and Electronic Engineering Department, West Anhui University, Lu'an 237012, China)

Abstract: In this paper explore how to use spectral methods of geometry structural graph for analyzing facial expression and clustering in the pattern-space. Use the leading eigenvectors of the weighted graph adjacency matrix to define eigenmodes of the adjacency matrix. For each eigenmode, compute vectors of spectral properties. It includes the leading inter-mode adjacency matrices. Embed these vectors in a pattern-space using multidimensional scaling on the norm for pairs of pattern vectors. Illustrate the utility of the embedding methods representing the arrangement of facial expression of dissimilar human face in the pattern-space.

Key words: graph spectra; human face recognition; analysis of facial expression; multidimensional scaling

0 引 言

计算机人脸识别技术是用计算机分析人脸图像,进而从中提取有效的识别信息,用来辨认身份的一门技术。由于人脸表情的丰富变化,年龄的变化以及图像受光照、成像角度、成像距离等因素的影响,目前还没有很好的描述人脸的三维模型。另外,人脸识别还涉及到图像处理、计算机视觉、模式识别以及神经网络等学科,也和人脑的认识程度密切相关。这诸多因素使得人脸识别成为一项极富挑战性的课题^[1]。

早期人脸研究主要有两大方向:一是提取人脸几何特征的方法^[2];二是模板匹配的方法。目前的研究主要集中在两个方向:其一是基于整体的研究方法,它考虑了模式的整体属性,包括特征脸(Eigenface)方法、SVD分解的方法、人脸等密度线分析匹配方法、弹性图匹配方法(Elastic graph matching)、隐马尔可夫模型(Hidden Markov Model)

方法以及神经网络的方法等;其二是基于特性分析的方法,也就是将人脸基准点的相对比率和其他描述人脸脸部特征的形状参数或类别参数等一起构成识别特征向量。笔者希望能建立一个良好的人脸模式,但上述方法难免都会受到以下一个或几个方面的影响:一是图像的质量、大小和视角因素;二是人脸的表情丰富性;三是计算的复杂度(含信息数据的降维问题)。

虽然 Berto^[3]在1993年对模板匹配方法和基于几何特征的方法比较后认为模板匹配方法优于几何特征的方法,但由于几何特征受图像的质量、尺寸和视角的影响较少,故基于人脸的几何特性的结构分析在人脸检测和人脸识别的研究中仍具有很重要的研究价值。文中尝试以结构图的形式采用图的谱分解方法来表示人脸的结构,图谱方法是用固定长度向量来表示图的结构。研究的过程是将原始图像以人脸特征点的形式构成加权邻接矩阵而后建立相应的谱模型,对于每个分解的谱模型,用联合特征向量去计算谱特征,以这种方式解决在图中的点与向量成分之间的对应关系。结果显示特征向量很好地表达了结构模式空间。

文中的目的是研究如何用谱特征向量^[4]来描述人脸

收稿日期:2005-07-13

基金项目:国家自然科学基金资助项目(60375010);教育部“优秀青年教师资助计划”资助项目(教人司[2003]355)

作者简介:孔 敏(1964-),男,安徽肥西人,副教授,博士研究生,研究方向为数字信号处理、结构模式识别和图像特征提取。

结构。

1 图谱和谱特征

1.1 图谱

文中用 $G_1, G_2, \dots, G_k, \dots, G_N$ 表示图集, $G_k = (V_k, E_k)$, 其中 V_k 是点集, $E_k \subseteq V_k \times V_k$ 是边集, 每个图计算其加权邻接矩阵:

$$A_k(i, j) = \begin{cases} \exp\left(\frac{-d(i, j)^2}{\sigma^2}\right) & (i, j) \in E \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

其中, i 和 j 各表示矩阵的行、列, $d(i, j)$ 为两点之间的距离, $k = 1, \dots, N$ 。每个 A_k 计算出其特征值 λ_k , 其满足 $|A_k - \lambda_k I| = 0$, 相应的特征向量 ϕ_k 满足 $A_k \phi_k = \lambda_k \phi_k$, 这里 ω 是特征模指数, 于是第 k 个图的邻接矩阵可表示为:

$$A_k = \sum_{\omega=1}^{|V_k|} \lambda_k^\omega \phi_k^\omega (\phi_k^\omega)^T \quad (2)$$

研究的目的是探索能否用矩阵的特性去建立图的特征向量, 步骤有二: 一是计算每个模邻接矩阵的特征, 再用相关特征去描绘模邻接矩阵的排列。对每个图仅仅用邻接矩阵的前 n 个特征模, 模特征矩阵是 $\Phi_k = (\phi_k^1 | \phi_k^2 | \dots | \phi_k^n)$ 。

1.2 谱特征

文中从图的邻接矩阵的模得到谱特征, 以特征值的次序来决定特征向量的次序, 研究的模间邻接矩阵谱特征描述如下。

图 k 中子图 S 和 T , 属于集合 $P(S, T) = \{(u, v) | u \in S \wedge v \in T\}$ 的点集合可表示为:

$$U_k(u, v) = \sum_{i \in V_s} \sum_{j \in V_t} \Phi_k(i, u) \Phi_k(j, v) A_k(i, j)$$

图谱特征向量:

$$B_k = (U_k(1, 1), U_k(1, 2), \dots, U_k(1, n), U_k(2, 1), \dots, U_k(2, n), \dots, U_k(n, n))^T$$

2 在模式空间嵌入谱向量

文中用多维尺度变换(MDS)方法来嵌入谱特征到特征空间。

寻找一种数据的分布使得其点间距离与原始数据之间的相似性相一致的一般过程称为多维尺度变换。Shepard 和 Kruskal 研究了一种不同的尺度变换技术称为序数尺度变换^[5], 这里用这一方法嵌入图数据, 在低维空间上重新表示这些数据并加以显示。

此处需要图之间的点间距离, 它是在 L_2 范数下计算图之间的谱模式向量, 对图 i_1 和 i_2 , 此距离是 $d_{i_1, i_2} = \sum_{\alpha=1}^k [B_{i_1}(\alpha) - B_{i_2}(\alpha)]^2$ 。类似地, 一个 $N \times N$ 的相异矩阵 D 可表示如下:

$$D_{i_1, i_2} = \begin{cases} d_{i_1, i_2} & i_1 \neq i_2 \\ 0 & i_1 = i_2 \end{cases} \quad (3)$$

文中用传统 MDS 在欧氏空间按图的差异性 D 嵌入可

视图, 首先计算矩阵 T , 以 r 代表行, c 表示列, 给出式子:

$$T_{rc} = -\frac{1}{2} [d_{rc}^2 - \hat{d}_{r..}^2 - \hat{d}_{.c}^2 + \hat{d}_{..}^2]$$

这里 $\hat{d}_{r..} = \frac{1}{N} \sum_{c=1}^N d_{rc}$ 是第 r 行相异均值, $\hat{d}_{.c}$ 是第 c 列相似均值, $\hat{d}_{..} = \frac{1}{N^2} \sum_{r=1}^N \sum_{c=1}^N d_{rc}$ 是相似矩阵 T 的所有行和列的相同均值。如果 T 的范围是 $k (k \leq N)$, 则进一步得到 k 个非零特征向量和特征值, 按序得到 $X = [f_1, f_2, \dots, f_k]$, 其中 $f_i = \sqrt{\lambda_i} e_i$ 是尺度变换特征向量, 以纵向形式嵌入这些向量: $x_i = (X_{i,1}, X_{i,2}, X_{i,3})^T$ 。

3 实验及结果

实验的过程是首先在实验人脸上确定特征点共 37 个, 而后构成相应的 Delaunay 图, 再构成相应的加权邻接矩阵, 最后按上述的方法进行谱分解方式嵌入谱特征。首先的人脸表情实验是提供两组不同表情、姿势的二维随机人脸图像, 其中实验人脸序列 1 是 41 幅图像, 序列 2 是 75 幅图像。

研究的目的是用不同的谱特征和不同的嵌入策略在模式特征空间中获得较好的人脸表情描述, 实验结果显示模间邻接矩阵谱特征在 MDS 嵌入方法下有较好的描述结果。

图 1 和图 2 显示了随机序列 1(41 幅人脸不同表情图像)和随机序列 2(75 幅人脸不同表情图像)模间邻接矩阵

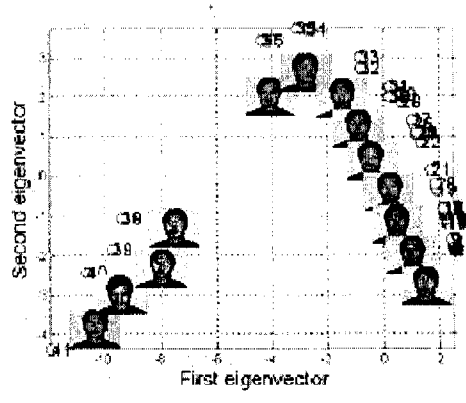


图 1 序列 1:人脸表情图像在二维空间的谱特征显示

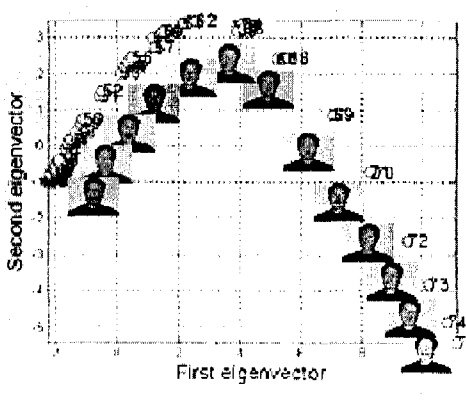


图 2 序列 2:人脸表情图像在二维空间的谱特征显示

(下转第 37 页)

能按正确的顺序执行。

2.4.1 在 VC++ 中

Windows 提供了 4 种同步对象和一组等待函数支持多线程同步。同步对象有信号量对象(Semaphore)、互斥体对象(Mutex)、事件对象(Event)和临界区对象(Critical-Section),MFC 也有相应的封装类。

临界区对象比其他 3 种同步对象更方便更快更有效,因为它不是核心对象,不需要用类似 CreatXXX 的 API 函数获得一个句柄,只需对要保护的共享资源声明一个 CRITICAL_SECTION 类型的变量,调用 InitializeCriticalSection()将其初始化,在 EnterCriticalSection 和 LeaveCriticalSection 函数之间进行共享资源的互斥访问。最后调用 DeleteCriticalSection 函数释放系统资源。

信号量对象维护一个从 0 到最大值之间的一个计数器,用来限制访问共享资源的线程数量。互斥体对象是信号量对象计数器最大值为 1 的特例。事件对象可以用 SetEvent 设置为有信号状态的同步对象,用于通知等待线程已发生一个特殊事件,等待线程可利用该事件对象访问共享资源^[5]。

2.4.2 在 JAVA 中

对于 JAVA 程序员来说,多线程同步的编程相对简单,复杂的工作由虚拟机完成。只需使用同步访问修饰符 Synchronized 并配合等待通知机制,即可控制对象的并发访问。Synchronized 修饰符既可用于方法的声明也可用于代码块。

3 性能与安全

为了进一步提高多线程系统的性能,特别是在开发服务器端系统时,可考虑使用线程池技术。另外一个通用的原则是不要长时间锁定共享资源。时间的长短取决于应用场合。

(上接第 34 页)

谱特征以 MDS 嵌入方法在二维模式空间的分布结果,实验结果表明有较好的结构前部分主要集中了惊讶等眼睛较大时的表情,后部分表现了眼睛较小状态下的表情。此研究表现出对人的嘴部变化不敏感。

4 结束语

文中研究了如何用图谱的向量特征去实现人脸的特征描述。为做到这点,选择人脸的几何特征构建特征向量并选择一定的方法在模式空间嵌入它们,其特征是模间邻接矩阵,嵌入的方法是 MDS,所用的人脸图是以二维视图用特征点构成的 Delaunay 图。文中研究的问题是同一人脸不同表情序列图在模式空间的特征表示,以期表达同一人脸的不同表情、姿势的相似性和相异性,得到了较好的结构变化轨迹,解决这个问题较好的谱特征是模间邻接矩

阵向量且用 MDS 策略来嵌入。文中用的是人脸的正面图像,对侧面人脸图像用本方法还有待研究。

一种可确保不发生死锁的策略就是强迫资源锁定,要么获得所需的所有资源,要么放弃。

要安全地结束一个线程。在 VC++ 中不要使用 TerminateThread()API 终止线程,这会使线程没有机会在结束前清理自己,目标线程不会获得机会,可能会引起内存泄露。在 JAVA 中也不要使用 Thread 的 stop()、suspend()、resume()等已淘汰的方法,可用布尔变量和等待通知机制配合进行替代。

4 结束语

多线程使程序得以将其工作分开,独立运作互不影响。用户交互性好效率高,但使用不当则会适得其反。为了保证数据的一致性,必须采取同步措施对多线程访问共享资源进行正确控制。有时性能和安全是一对矛盾,设计线程安全的类相对于不采用同步控制的类运行速度较慢,需要在具体应用中权衡利弊作出选择。若要开发跨平台的多线程应用,则非 JAVA 莫属。若应用只需在 Windows 平台运行,则用 VC++ 和 JAVA 作开发语言均可,JAVA 相对简单易用,而 VC++ 效率更高。

参考文献:

- [1] Richter J. Windows 核心编程[M]. 王建华等译. 北京:机械工业出版社,2000.
- [2] 骆 斌,费翔林. 多线程技术的研究与应用[J]. 计算机研究与发展,2000,37(4):409-412.
- [3] Kruglinski D J, Wingo S, Shepherd G. Visual C++ 6.0 技术内幕[M]. 希望图书创作室译. 北京:北京希望电子出版社,1999.
- [4] Hyde P. Java 线程编程[M]. 周良忠译. 北京:人民邮电出版社,2003.
- [5] Beveridge J, Wiener R. Win 32 多线程程序设计[M]. 侯 捷译. 武汉:华中科技大学出版社,2002.

参考文献:

- [1] 张翠平,苏光大. 人脸识别技术综述[J]. 中国图象图形学报,2000,5(11):885-894.
- [2] Bledsoe W. Man-machine facial recognition[M]. Palo Alto, CA:Panoramic Research Inc,1966.
- [3] Berto R, Poggio T. Face recognition: Feature versus templates[J]. IEEE Trans on PAMI, 1993, 15(10):1042-1052.
- [4] Luo B, Wilson R C, Hancock E R. Spectral embedding of graphs[J]. Pattern Recognition, 2003,36(10):2213-2223.
- [5] Kruskal J B. Nonmetric multidimensional scaling: a numerical method[J]. Psychometrika,1964,29:115-129.