二次式距离上基于 SVD 的高维图像索引方法

崔江涛 孙君顶 付少锋 周利华
(西安电子科技大学计算机学院,西安 710071)

摘 要 向量近似方法(vector approximation file)是解决高维索引中维数灾难问题的一种有效方法,但是它不能直接支持二次式距离上的近邻搜索,为此,提出一种基于奇异值分解(SVD)的二次式距离上的向量近似方法,通过奇异值分解技术将二次式距离变换为欧氏距离形式,对变换后的特征向量进行近似得到近似向量。进行近邻搜索时采用低维过滤算法,先在较高能量的低维子空间内计算近似距离进行过滤,再对过滤结果进行高维距离计算。实验结果表明,低维过滤算法可以过滤掉大部分特征向量,而只有小部分数据需要进行高维距离运算,该方法可以显著提高大型高维图像数据库的近邻搜索性能。

关键词 维数灾难 二次式距离 近邻搜索 奇异值分解 向量近似
中图法分类号: TP311.134.3 **文献标识码**: A **文章编号**: 1006-8961(2006)04-0498-06

Efficient High-Dimensional Image Indexing Based on SVD for Quadratic form Distance

CUI Jiang-tao, SUN Jun-ding, FU Shao-feng, ZHOU Li-hua
(*School of Computer Science and Engineering. Xi'an University, Xi'an* 710071)

Abstract Many traditional indexing methods perform poorly in high dimensional vector space. The Vector Approximation File approach overcomes some of the difficulties of dimensionality curse, but it can't support the quadratic form metric. A novel VA-File approach for quadratic form distance is introduced in this paper. By the SVD of similarity matrix, the quadratic form distance can be converted to the Euclidean distance, and the approximation vector can be obtained. The low-dimensional filter algorithm is also applied during the nearest neighbor search. The vectors are first filtered with the low-dimensional approximate distance measure, and then the candidate results are re-computed with high-dimensional distance measure. The experimental results show that it can save the computational time significantly because only a small set of vectors is computed on the high-dimensional distance measure.

Keywords dimensionality curse, quadratic form distance, nearest neighbor search, singular value decomposition(SVD), vector approximation

1 引 言

基于内容的图像检索(content-based image retrieval,简称 CBIR)已经成为图像数据库中的一项重要应用。当前大多数 CBIR 系统采用向量空间模型,用一定维数的向量来表示图像特征。通过某种距离度量方式计算两个向量之间的距离可以衡量图像之间的相似性,所以图像的相似性查询可以看成是向量空间中的 k 近邻搜索问题。用于表示图像特征的向量往往具有高维特性,高维索引机制也就成为最近几年多媒体领域的一个研究热点[1]。传统的多维检索技术(R^* 树,X 树,SR 树和网格文件等)[2]可以解决向量空间中的 k 近邻搜索问题,但是最近的研究

基金项目:"十五"国防科技(电子)预研项目(413160501)
收稿日期:2005-01-10;**改回日期**:2005-07-25
第一作者简介:崔江涛(1975~),男,讲师,博士。研究方向为图像处理、基于内容的多媒体信息检索、高维索引技术。E-mail: cuijt@mti.xidian.edu.cn

表明这些检索技术在高维情况下(超过几十维)检索效率很低,其性能甚至低于最简单的顺序查找算法[3,4],这种现象又被称为维数灾难(dimensionality curse)。对于采用相关反馈技术的 CBIR 系统而言,多次的人机交互需要进行多轮次的检索过程,检索时间的效率问题也就成为目前 CBIR 系统的一个制约因素。目前,大多数 CBIR 系统的实验原型仅仅处理几百或几千幅图像,即使对所有图像按序遍历进行查询也不会带来性能上的显著下降,因此大部分图像检索系统都没有采用任何检引技术。

向量近似方法(vector approximation file, 简称 VA-File)[4]是目前惟一的在高维情况下优于顺序查找的检索结构,其基本思想是将高维特征向量进行压缩并近似存储,通过对近似向量进行扫描过滤来加快搜索速度。在 VA-File 方法的基础上也发展出了其他相关检索结构,如 VA$^+$-File 方法[5],在进行常规向量近似前采用 K-L 变换来去除各维之间的相关性;在各分量上采用高斯混合模型来拟合数据的边缘分布[6],提高 VA-File 的量化精度,从而提高过滤能力;根据数据点的分布将树型结构与 VA-File 方法相结合,提出的 GC 树[7]方法。

在 CBIR 系统中,常用的距离度量方式包括欧拉距离(L_1 距离和 L_2 距离)、二次式距离(包括马氏距离)等。由于二次式距离的定义方式中考虑了各分量之间的相关性,对于直方图类型特征的特征匹配,二次式距离比使用欧拉距离更有效,但是 VA-File 方法并不直接支持二次式距离。为此提出了一种支持二次式距离的 VA-File 方法,在 VA-File 方法的第 1 阶段过滤过程中采用低维过滤算法,在很少维数上进行距离计算就可以过滤掉大部分的特征向量,仅很少部分特征向量需要进行高维计算,从而在不增加 I/O 复杂度的情况下降低了计算复杂度,提高 VA-File 方法性能。

2　相关工作

VA-File 的基本思想就是对特征向量进行近似表示,在向量空间内对向量进行近似时,对每一维空间 j 分配一定的近似位数 b_j,可以将坐标轴分割成 2^{b_j}个区间,$\sum_{j=1}^{d} b_j = b$,d 是向量空间维数,整个向量空间被分成 2^b 个超立方体形状的胞腔(cell)。一个特征向量可由其所处的超立方体近似表示,称为近似向量,其二进制长度为 b。如图 1 所示,在 2 维空间内对每一维空间各分配 2 位近似位数,则 $\boldsymbol{p}_i$ 所处的胞腔可表示为 1011,所以 $\boldsymbol{p}_i$ 可由近似向量 $\boldsymbol{a}_i$ = 1011 表示。所有特征向量的近似向量顺序排列可以组成一个近似向量文件 VA-File。根据 $\boldsymbol{a}_i$,可以计算查询向量 $\boldsymbol{q}$ 与 $\boldsymbol{p}_i$ 之间的距离上界 u_i 和下界 l_i,如图 1 所示,$l_i \leqslant d_{p_i,q} \leqslant u_i$。$k$ 近邻搜索过程分为两个阶段,第 1 阶段中,顺序扫描整个 VA-File,根据 a_i 计算特征向量和查询向量之间的距离下界和上界,如果某特征向量与查询向量的距离下界大于目前近邻结果集中的第 k 小的距离上界,此向量就可以被过滤掉,因为已经存在至少 k 个更好的候选向量。在第 2 阶段中,对没有过滤掉的候选向量,读取其原始特征向量,按距离下界由小到大的顺序计算 $\boldsymbol{p}_i$ 与查询向量的实际距离,得到最终的 k 个近邻。

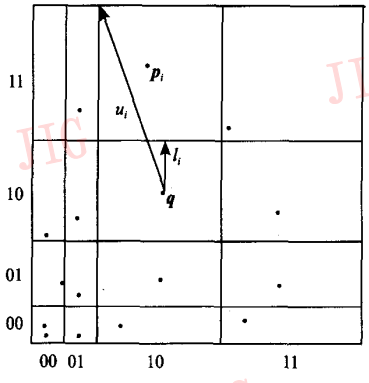


图 1　VA-File 示例

Fig. 1　Example of VA-File

由于近似位串的存储空间远小于原始特征向量,对近似位串的访问可以显著减少系统的 I/O 时间,在过滤阶段过滤掉更多的向量就可以获得更好的性能。VA-File 中的过滤能力主要由两方面决定,首先是距离上下界对原距离的逼近程度,下界距离逼近程度越高,第 2 阶段需要计算的特性向量数目越少;上界距离逼近程度越高,第 1 阶段可以取得更好的过滤能力。其次是数据点分布情况,各个胞腔内数据点数量越均衡,过滤能力越强。

3　二次式距离上 VA-File 方法

3.1　二次式距离 VA-File 构造算法

在 VA-File 方法中,对于欧氏距离(L_2 距离),上下界距离计算公式如下:

$$l_i^2 = \sum_{j=1}^{d} (l_{i,j})^2$$
$$u_i^2 = \sum_{j=1}^{d} (u_{i,j})^2 \quad (1)$$

这种上下界距离的计算方式,不能直接用于二次式距离。通过对相似矩阵进行奇异值分解(singular value decomposition,简称 SVD),可以将二次式距离变换为欧氏距离形式。二次式距离定义如下:

$$d_{p,q} = (\boldsymbol{p} - \boldsymbol{q})^{\mathrm{T}} \boldsymbol{A} (\boldsymbol{p} - \boldsymbol{q}) \quad (2)$$

其中,$\boldsymbol{A} = (a_{i,j})_{i,j=1,\cdots,d}$是相似矩阵,在 CBIR 系统中 $\boldsymbol{A}$ 通常是一个正对称矩阵,也是一个正定矩阵。$a_{i,j}$是分量 i 和 j 之间的相似系数。假设,$a_{i,i} = 1$ 并且 $0 \leqslant a_{i,j} < 1 (i \neq j)$。

根据矩阵的 SVD 技术,存在正交矩阵 $\boldsymbol{V}$,使得 $\boldsymbol{A} = \boldsymbol{V}^{\mathrm{T}} \boldsymbol{\Sigma} \boldsymbol{V}$。其中,$\boldsymbol{\Sigma} = diag(\lambda_1, \cdots, \lambda_d)$,并且 $\lambda_1 \geqslant \cdots \geqslant \lambda_k \geqslant \cdots \geqslant \lambda_d \geqslant 0$,$\lambda_i$ 是矩阵 $\boldsymbol{A}$ 的奇异值,也是 $\boldsymbol{A}$ 的特征值。

则

$$d_{p,q} = (\boldsymbol{V}\boldsymbol{p} - \boldsymbol{V}\boldsymbol{q})^{\mathrm{T}} \boldsymbol{\Sigma} (\boldsymbol{V}\boldsymbol{p} - \boldsymbol{V}\boldsymbol{q}) \quad (3)$$

这样二次式距离可以表示成加权欧氏距离形式,而 λ_i 就是各分量的能量加权系数,得上下界距离计算公式为

$$l_i^2 = \sum_{j=1}^{d} \lambda_j (l_{i,j})^2$$
$$u_i^2 = \sum_{j=1}^{d} \lambda_j (u_{i,j})^2 \quad (4)$$

VA-File 中默认各分量中数据分布情况相同,各分量之间分配的位串长度也是相同的。通过 SVD 得到的正交变换方式,不但消除了各分量之间的相关性,而且使变换后的能量主要集中在少数分量上。根据各分量的数据分布情况分配不同位串长度,可以达到更好的近似效果。设变换后第 i 维分量的方差为 σ_i^2,将加权系数考虑在内,式(2)可以表示成

$$d_{p,q} = (\boldsymbol{\Lambda}\boldsymbol{V}\boldsymbol{p} - \boldsymbol{\Lambda}\boldsymbol{V}\boldsymbol{q})^{\mathrm{T}} (\boldsymbol{\Lambda}\boldsymbol{V}\boldsymbol{p} - \boldsymbol{\Lambda}\boldsymbol{V}\boldsymbol{q}) \quad (5)$$

其中,$\boldsymbol{\Lambda} = diag(\sqrt{\lambda_1}, \cdots, \sqrt{\lambda_d})$,所以第 i 维分量的方差为 $\lambda_i \sigma_i^2$。设该分量的近似位串长度为 b_i,b_i 满足 $\sum_i b_i = b$。在量化理论中,为达到更好的量化精度,有一个拇指原则(rule of thumb),即如果 $\sigma_m^2 \geqslant 4^k \sigma_n^2$,则 $b_m \geqslant b_n + k$。此原则在高分辨率均匀量化的前提下成立,但是在非均匀量化情况下,此原则仍具有指导性[5],可以作为位串长度分配的原则。位串长度分配算法如下:

(1)对变换后所有分量,令 $d_i = \lambda_i \sigma_i^2$,$b_i = 0$,$k$ 赋初值为 0;

(2)查找方差最大值,假设 d_i 取得最大值,则 $b_i = b_i + 1$,$d_i = d_i / 4$,$k = k + 1$;

(3)如果 $k < b$,则继续步骤 2,否则算法结束。

真实图像数据库中,各分量上的数据分布是非均匀分布。本文借鉴文献[6]中的思想,采用混合高斯模型来描述变换后各分量的边缘分布,通过 Expectation-Maximization(EM)算法来估计各种分布参数[9]。假设 $\hat{p}(x)$ 是通过小部分数据训练得出的某分量的概率密度分布函数,使各分割区间内具有相同数量的数据点可以取得更好的过滤性能。假定该分量分配的近似位串长度为 b,那么分割区间共有 2^b 个,用 $m(r)$,$0 \leqslant r \leqslant 2^b$ 表示分割点坐标,那么各分割点应满足下式:

$$\int_{m(r)}^{m(r+1)} \hat{p}(x)\,\mathrm{d}x = \frac{1}{2^b} \int_{m(0)}^{m(2^b)} \hat{p}(x)\,\mathrm{d}x \quad (6)$$

3.2 低维过滤算法

由 SVD 分解过程可知,正交变换后各分量按加权系数 λ_i 由大到小排列。在 VA-File 的第 1 阶段过滤过程中,提出一种低维过滤算法,即只在加权系数较大的少数分量上计算下界距离。如果计算出的下界距离大于目前结果集中的第 k 小的上界距离,就可以把此向量过滤掉,不必访问其他分量。

假设选定的低维维数为 s,$l_i^{(s)}$ 是在 s 维分量上计算出的下界距离,即 $(l_i^{(s)})^2 = \sum_{j=1}^{s} \lambda_j (l_{i,j})^2$,由 l_i 的定义方式可知,$l_i^{(s)} \leqslant l_i$,即低维过滤算法是有效的,不会产生任何错误遗漏。对于没有过滤掉的近似向量,在其所有分量上计算距离上界和距离下界,继续过滤过程。在不考虑特征向量分布,且 s 取值确定的情况下,选择近似位串中的加权系数最大 s 个分量来计算距离,可以达到近似最好的过滤性能,证明过程如下:

假定 $\boldsymbol{B}_s$ 是一个 $s \times d$ 的变换矩阵,与原向量相乘可以得到 s 维分量。在 s 维向量空间上 $\boldsymbol{p}$ 和 $\boldsymbol{q}$ 的距离 $d_{p,q}^{(s)}$ 与实际距离 $d_{p,q}$ 的差值为

$$d_{p,q} - d_{p,q}^{(s)} = (\boldsymbol{p} - \boldsymbol{q})^{\mathrm{T}} (\boldsymbol{A} - \boldsymbol{B}_s^{\mathrm{T}} \boldsymbol{B}_s) (\boldsymbol{p} - \boldsymbol{q}) \quad (7)$$

要使 s 维距离 $d_{p,q}^{(s)}$ 最好地逼近 $d_{p,q}$,需要 $\boldsymbol{B}_s^{\mathrm{T}} \boldsymbol{B}_s$ 满足以下极值条件:

$$\inf_{\boldsymbol{B}_s^{\mathrm{T}} \boldsymbol{B}_s} \left\{ \sup_{\boldsymbol{p}-\boldsymbol{q}} [(\boldsymbol{p} - \boldsymbol{q})^{\mathrm{T}} (\boldsymbol{A} - \boldsymbol{B}_s^{\mathrm{T}} \boldsymbol{B}_s) (\boldsymbol{p} - \boldsymbol{q})] \right\} \quad (8)$$

约束条件是 $\boldsymbol{A} - \boldsymbol{B}_s^{\mathrm{T}} \boldsymbol{B}_s$ 为半正定。取 $\boldsymbol{B}_s^{\mathrm{T}} \boldsymbol{B}_s =$

$\boldsymbol{V}_s^{\mathrm{T}}\boldsymbol{\Sigma}_s\boldsymbol{V}_s$，其中，$\boldsymbol{\Sigma}_s = diag(\lambda_1,\cdots,\lambda_s)$，$\boldsymbol{V}_s$ 是矩阵 $\boldsymbol{V}$ 的前 s 行，可以满足式(8)[10]。即采用加权系数最大得 s 个分量来计算距离，可以获得对原距离最好的近似。

根据上述推导结果，可以建立一种 VA-File 上的二次型距离快速检索算法。d 维向量的 VA-File 建立过程如下：

(1)对相似矩阵 $\boldsymbol{A}$ 进行 SVD 分解，计算加权系数矩阵 $\boldsymbol{\Sigma}$ 和变换矩阵 $\boldsymbol{V}$，对原特征向量进行变换。

(2)采用 EM 算法在每一维空间内拟合出数据点分布的混合高斯模型，计算出近似的概率密度分布函数。

(3)根据各分量方差确定各分量的近似位串长度。

(4)对变换后的向量 $\boldsymbol{p}_i$ 进行近似得到 VA-File。

采用低维过滤算法的 k 近邻查询过程如下：

(1)初始化 k 个最近邻距离，并根据奇异值分布情况选取合适的 s 值作为低维维数。

(2)对每个特征向量 $\boldsymbol{p}_i$，在前 s 维分量上计算 $\boldsymbol{p}_i$ 与查询向量 $\boldsymbol{q}$ 之间的距离下界 $l_i^{(s)}$。

(3)如果 $l_i^{(s)}$ 小于第 k 小的距离上界，则继续在 d 维分量上计算距离下界 l_i 和上界 u_i，否则排除向量 $\boldsymbol{p}_i$。

(4)如果 l_i 小于第 k 小的近邻距离，则保留向量 $\boldsymbol{p}_i$，否则排除向量 $\boldsymbol{p}_i$。

(5)对没有过滤掉的向量，进一步计算得到最终的 k 个近邻。

4　实验结果与分析

采用 MPEG-7 标准中的 256 维的 SCD(scalable color descriptor)作为颜色直方图特征[11]，对二次式距离上基于 SVD 的 VA-File 算法进行模拟实验。测试图像数据库包含10 000幅不同类别自然图像，相似矩阵 $\boldsymbol{A}$ 采用文献[10]中的定义方式：

$$a_{ij} = 1 - d(c_i,c_j)/d_{\max} \tag{9}$$

其中，c_i 和 c_j 是颜色直方图中的第 i 种和第 j 种颜色，a_{ij} 是两种颜色之间的相似系数，$d(c_i,c_j)$ 是两种颜色在 HSV 颜色空间上的欧氏距离，$d_{\max}$ 是任意两种颜色之间的最大距离。实验中 $c_i = (h_i,s_i,v_i)$，$c_j = (h_j,s_j,v_j)$，则

$$d(c_i,c_j) = \sqrt{(h_i - h_j)^2 + (s_i - s_j)^2 + (v_i - v_j)^2} \tag{10}$$

由上述定义可知，$\boldsymbol{A}$ 是一个正对称矩阵。矩阵 $\boldsymbol{A}$ 的奇异值分布在 142.23 ~ 0.03 之间，按降序排列前 8 个奇异值从 162.29 ~ 1.23，其他均小于 1，前 40 个奇异值分布如图 2 所示。图 3 给出了经过变换后各分量的方差，方差越大，表示该分量中数据点分布越分散，需要更多的近似位数，由图可以看出，各分量的方差与其奇异值大小并无明显对应关系，变换后的向量在第 3 分量上取得方差最大值。

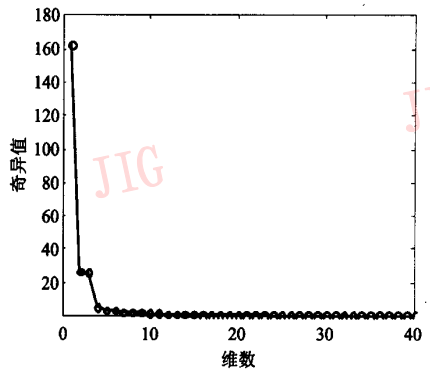


图 2　奇异值分布

Fig. 2　Distribution of the singular value

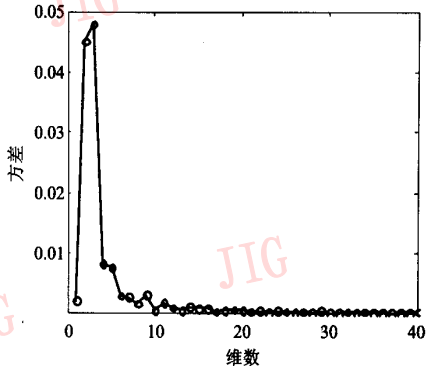
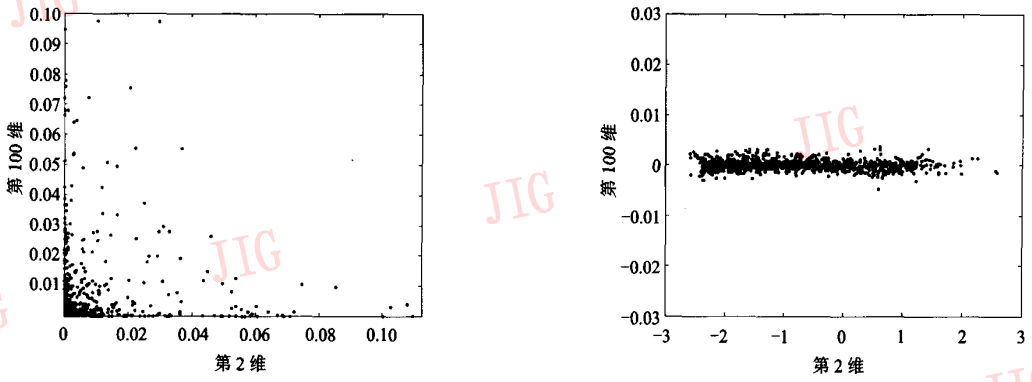


图 3　变换后各分量方差

Fig. 3　Variance of the dimension after translation

随机选择 1 000 个特征向量，选取特征向量中第 2 维分量和第 100 维分量组成一个 2 维空间，图 4 给出了变换前后 2 维空间中特征向量的分布情况，可以看出正交变换后能量主要集中在第 2 维分量上。当近似位串平均长度为 4 位/维时，采用非均匀位串分配算法，变换后第 2 维分量分配的位串长度为 9，第 100 维分量分配的位串长度为 3。

本文实现了 SVD 变换模型下的二次式距离上的 VA-File 方法，并且在第 1 阶段搜索过程中采用了低维过滤算法，低维维数分别选择了 3 维、8 维和 16 维，分别将其标记为 3-SVD，8-SVD 和 16-SVD。



(a) 变换前 (b) 变换后

图 4 正交变换前后数据点分布情况

Fig. 4 Distribution of the data point before transform and after transform

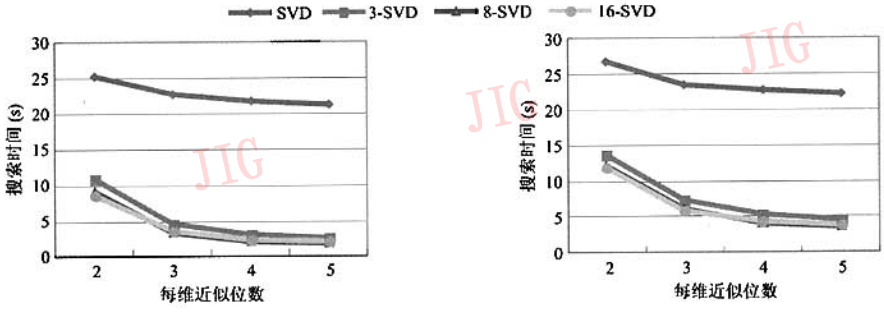
位串长度分别实现了平均 2bit/dim, 3bit/dim, 4bit/dim和5bit/dim 4 种方案。测试结果是随机选取 100 个特征向量进行 10 近邻(10-NN)和 50 近邻(50-NN)搜索的平均结果。实验结果表明,各种算法检索结果完全相同。表 1 是第 1 阶段过滤后剩余的特征向量的比例,其中低维过滤算法是经过低维过滤后剩余的特征向量比例。由表 1 可以看出,在低维过滤阶段已经可以过滤掉大多数特征向量。

表 1 第 1 阶段过滤后剩余向量比例(百分比)

Tab. 1 The percent of remained vectors after the first phase

	10-NN				50-NN			
	2bit/dim	3bit/dim	4bit/dim	5bit/dim	2bit/dim	3bit/dim	4bit/dim	5bit/dim
SVD	16.65	3.90	1.64	1.02	23.33	8.98	4.96	3.66
3-SVD	31.21	15.42	10.75	8.97	37.89	29.72	17.27	15.12
8-SVD	22.39	8.29	4.66	3.53	29.16	14.72	9.89	8.13
16-SVD	19.28	6.03	3.04	2.15	25.78	11.87	7.39	5.86

图 5 给出了 4 种算法进行第 1 阶段过滤的 CPU 运行时间,可以看出,低维过滤算法可以对 VA-File 方法带来较大的性能提升。低维过滤算法中选用维数越高,可以取得更好的过滤效率,但计算距离下界时需要计算更多的维数,如何选择合适的低维维数,需要权衡算法过滤性能和下界距离计算维数。从图 5中可以看出,8-SVD 和 16-SVD 两种算法的运算时间相差不大。



(a) 10-NN $N = 10\,000$ (b) 50-NN $N = 10\,000$

图 5 低维过滤算法平均运行时间

Fig. 5 Average elapsed time of the low-dimension filtering algorithm

5 结 论

针对二次式距离上的高维图像数据库检索,提出了基于 SVD 的 VA-File 方法。二次式距离经过正交变换转换为欧氏距离,各分量之间能量相差较大,此时采用非均匀的位串分配算法,并在能量较大的低维分量上进行低维过滤,可以明显提高 VA-File 方法的第 1 阶段过滤性能。二次式距离变换为欧氏距离还有其他方式,如采用正定矩阵进行变换的方式,变换后各分量之间能量相差不大,其第 1 阶段过滤性能要低于基于 SVD 的变换方式,而且采用低维过滤算法不能带来明显的性能改变。

参 考 文 献(References)

1 Rui Y, Huang T S, Chang S F. Image retrieval: current techniques, promising directions, and open issues [J]. Journal of Visual Communication and Image Representation, 1999, **10**(4):36 ~ 92.

2 Bohm C, Berchtold S, Keim D. Searching in high-dimensional spaces-index structures for improving the performance of multimedia databases[J]. ACM Computing Surveys, 2001, **33**(3):322 ~ 373.

3 Berchtold S, Bohm C, Keim D, *et al.* A cost model for nearest neighbor search in high-dimensional data space[A]. In: Proceedings of ACM Symposium on Principles of Database Systems[C], Tuscon, Arizona, USA, 1997: 78 ~ 86.

4 Weber R, Schek H J, Blott S. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces[A]. In: Proceedings of 24th International Conference on Very Large Database[C], New York, 1998:194 ~ 205.

5 Ferhatosmanoglu H, Tuncel E, Agrawal D. Vector approximation based indexing for non-uniform high dimensional data sets[A]. In: Proceedings of the ACM Int'l Conference on Information and Knowledge Management (CIKM 2000) [C]. New York, 2000: 202 ~ 209.

6 Wu P, Manjunath B, Chandrasekaran S. An adaptive index structure for high-dimensioanl similarity search [A]. In: Proceedings of Advances in Multimedia Information Processing [C], Berlin, Germeny, 2001: 71 ~ 77.

7 Cha G H, Chung C W. The GC-Tree: A high-dimensional index structure for similarity search in image databases [J]. IEEE Transactions on Multimedia, 2002, **4**(2):235 ~ 247.

8 Cui Y. High-dimensional Indexing: Transformational Approaches to High-dimensional Range and Similarity Searches[A]. In: Lecture Notes in Computer Science [M], Heidelberg: Springer-Verlag, 2002, **2341**.

9 Resch B. Mixtures of Gaussians, A Tutorial for Course Computational Interlligence[EB/OL]. http://www.igi.tugraz.at/lehre/CI. 2004-08-14

10 Hafner J, Sawhney H S, Equitz W, *et al.* Efficient color histogram indexing for quadratic form distance functions[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995, **17**(6): 729 ~ 736.

11 Manjunath B S, Ohm J R, Vasudevan V V, *et al.* Color and texture descriptors[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2001, **11**(6):703 ~ 714.