

# 一种多分辨率高维图像特征匹配算法\*

崔江涛 刘卫光 周利华

(西安电子科技大学计算机学院, 西安 710071)

**摘 要** 通过对图像进行特征提取和变换, 图像的相似性匹配可以转换为高维向量空间内的点匹配. 为了解决高维数据的维数灾难问题, 提出一种基于多分辨率数据结构的向量近似方法. 从低分辨率开始计算距离下限, 如果距离下限大于目前结果集中的最大距离, 则不需要在高分辨率上计算其距离而将其排除掉, 从而降低了向量近似方法的运算复杂度. 提出应用此方法的近邻搜索算法并运用到图像数据库的特征匹配中, 实验证明: 新方法搜索效率优于传统的向量近似方法.

**关键词** 高维图像特征; 维数灾难; 多分辨率; 特征匹配

**中图分类号** TP311.134.3

**文献标识码** A

提高图像数据库中近邻搜索的搜索速度.

## 0 引言

图像特征的相似性匹配是图像数据库中的一个重要应用, 常用的图像相似性匹配方法都是基于向量空间模型. 为了增加搜索精度, 用于表征图像的特征向量往往具有高维特性, 其维数数量级可达到  $10^2$ . 在特征向量空间维数较低的情况下, 传统的基于树结构的多维检索技术可以用来解决相似性查询问题, 它们包括 R\* 树, X 树, SR 树和 M 树等<sup>[1,2]</sup>. 但是, 大多数的多维索引结构在空间和时间复杂度上都对维数有着指数依赖关系, 当向量空间维数比较大时, 比如大于 100 时, 这些索引方法性能迅速下降, 其性能甚至不如穷举搜索, 此现象也被称为维数灾难 (curse of dimensionality).

Weber 等人在 1998 年提出了向量近似的方法 (VA-File)<sup>[3]</sup> 来克服图像查询中的维数灾难, 向量近似方法的基本思想是将高维特征数据进行压缩和近似存储, 通过数据点过滤来加快搜索速度. Guang-Ho Cha 等人在此基础上通过增加压缩参数来提高过滤性能<sup>[4]</sup>. 但是在上述方法中, 特征空间中每一维对应的压缩位数是一个系统参数, 作者没有提出选择合适位数的模型和准则, 而系统的运算复杂度和过滤能力又受此参数影响很大, 压缩位数选择少时, 压缩能力强但过滤能力弱, 而位数多时, 过滤能力强但压缩能力弱.

Byung 等提出了一种多分辨率的数据结构用于高维图像数据库的线性穷举搜索<sup>[5]</sup>, 可以显著提高线性穷举搜索的性能, 本文借助此方法思想, 将多分辨率数据结构运用到向量近似方法中, 可以在保证搜索性能的基础上, 尽可能的减少运算复杂度, 从而

## 1 向量近似方法原理

传统的基于树结构的多维检索算法大多采用数据分割和空间分割的方法<sup>[6]</sup>, 其基本思想就是将距离相近的向量存储在同一个叶子节点中, 而根节点中则保存其叶子节点中所有向量所构成的空间区域. 在对向量进行匹配时, 要判断向量是否包含在根节点所代表的空间区域内, 依据此规则在树中遍历根节点和叶子节点. 在进行高维特征向量匹配时, 如果向量维数比较高, 不同的根节点和叶子节点代表的空间区域可能完全重合, 对树结构进行遍历时往往要访问所有的叶子节点, 其性能反而不如线性穷举搜索<sup>[3]</sup>.

既然基于树结构的多维检索方法在高维情况下性能低于线性穷举搜索, 在向量近似方法中搜索方法采用了线性穷举搜索的方法, 但是在搜索时是对近似向量进行匹配. 在进行向量近似时, 将向量空间分割成  $2^b$  个超立方体的子空间, 然后对每个子空间分配一个唯一的数字位串, 每个数字位串的长度都为  $b$ . 因为特征向量可以看作是向量空间中的一个点, 这个点所在的子空间的数字位串就是此特征向量的近似向量. 所有近似向量组成的序列就是一个向量近似文件.

在利用线性穷举搜索对向量  $q$  进行匹配时, 读取近似文件中的近似向量, 就可以计算出向量  $q$  到此近似向量代表的子空间的距离的下限, 如果距离下限大于目前的已经计算出近邻距离时, 就可以把此向量过滤掉, 从而减少算法的运算复杂度.

## 2 采用多分辨率数据结构的向量近似方法

将多分辨率数据结构运用到近似向量的构建

\*十五国防科技(电子)预研项目(413160501)资助

Tel: 029-88202244 Email: cuijt@mti.xidian.edu.cn

收稿日期: 2004-05-27

中,用于检索具有高维特征的图像,可以进一步降低算法的运算复杂度. 假设  $N$  是特征向量的个数,  $d$  是特征向量的维数,  $L_p$  是向量之间的距离度量方式. 采用  $i \in \{1, \dots, N\}$  来表示特征向量的范围,  $j \in \{1, \dots, d\}$  来表示维数范围, 这样  $p_i$  表示一个独立的特征向量, 而  $p_{i,j}$  则表示  $p_i$  的第  $j$  个分量. 用  $a_i$  来表示  $p_i$  的近似, 采用  $b_j$  表示对第  $j$  维空间进行近似所需要的位数. 第  $j$  维空间中采用  $2^{b_j} + 1$  个刻度来将坐标轴平均划分成  $2^{b_j}$  个区间, 令  $m_j(0)$  和  $m_j(2^{b_j})$  分别对应刻度的最小值和最大值, 坐标轴上的区间标记为  $0 \cdots 2^{b_j} - 1$ . 假设  $p_{i,j}$  坐落的区间标记为  $r_{i,j}$ , 那么  $m_j(r_{i,j}) \leq p_{i,j} < m_j(r_{i,j} + 1)$ . 给定一个需要匹配的特征向量  $q$ , 利用近似向量  $a_i$  可以确定  $q$  到  $p_i$  距离的上限  $u_i$  和下限  $l_i$ , 见图 1, 即  $l_i \leq L_p(q, p_i) \leq u_i$ ,  $u_i$  和  $l_i$  计算公式为

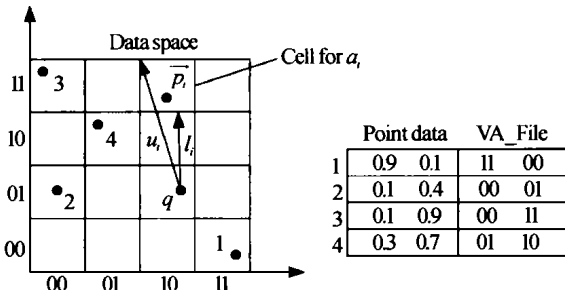


图 1 VA-File 示例  
Fig. 1 An example of VA-File

$$l_i = \left( \sum_{j=1}^d (l_{i,j})^p \right)^{1/p}$$

$$u_i = \left( \sum_{j=1}^d (u_{i,j})^p \right)^{1/p} \quad (1)$$

式中

$$l_{i,j} = \begin{cases} q_j - m_j(r_{i,j} + 1) & (r_{i,j} < r_{q,j}) \\ 0 & (r_{i,j} = r_{q,j}) \\ m_j(r_{i,j}) - q_j & (r_{i,j} > r_{q,j}) \end{cases}$$

$$u_{i,j} = \begin{cases} q_j - m_j(r_{i,j}) & (r_{i,j} < r_{q,j}) \\ \max(q_j - m_j(r_{i,j}), m_j(r_{i,j} + 1) - q_j) & (r_{i,j} = r_{q,j}) \\ m_j(r_{i,j} + 1) - q_j & (r_{i,j} > r_{q,j}) \end{cases}$$

采用颜色直方图  $p$  作为图像特征向量, 直方图  $p$  中的颜色块的个数为  $d$  ( $d = 2^H$ ). 这里  $\sum_{j=1}^d p(j) = 1$ . 将直方图  $p$  中两个相邻的颜色块的值相加, 可以形成一个颜色块数为  $2^{H-1}$  的直方图. 依次类推, 我们得出一个塔式的直方图序列  $\{p^0, \dots, p^h, \dots, p^H\}$ , 其颜色块数分别为  $\{2^0, \dots, 2^h, \dots, 2^H\}$ ,  $p^H = p$ . 这里  $p^h$  是  $p^{h+1}$  的低分辨率直方图向量,  $p^h(j)$  可以通过式 (2) 得到

$$p^h(j) = p^{h+1}(2j-1) + p^{h+1}(2j) \quad (1 \leq j \leq 2^h) \quad (2)$$

在直方图类型的图像特征匹配中,  $L_1$  距离可以达到更好的匹配效果<sup>[7]</sup>. 给定直方图  $p$  和查询直方

图  $q$ , 它们在  $h+1$  分辨率上的  $L_1$  距离标记为  $\|p^{h+1} - q^{h+1}\|$ , 则

$$\|p^{h+1} - q^{h+1}\| = \sum_{j=1}^{2^{h+1}} |p^{h+1}(j) - q^{h+1}(j)| = \sum_{j=1}^{2^h} \{|p^{h+1}(2j-1) - q^{h+1}(2j-1)| + |p^{h+1}(2j) - q^{h+1}(2j)|\} \quad (3)$$

$$\|p^h - q^h\| = \sum_{j=1}^{2^h} |p^h(j) - q^h(j)| = \sum_{j=1}^{2^h} |p^{h+1}(2j-1) - q^{h+1}(2j-1)| + |p^{h+1}(2j) - q^{h+1}(2j)| \quad (4)$$

由式 (3)、(4) 可知

$$\|p^{h+1} - q^{h+1}\| \geq \|p^h - q^h\|$$

所以, 对于数据集中的直方图  $p_i$ , 有不等式

$$\|p_i^H - q^H\| \geq \dots \geq \|p_i^h - q^h\| \geq \dots \geq \|p_i^0 - q^0\| \quad (5)$$

在各个分辨率直方图序列  $p_i^h$  上, 根据向量近似方法, 可以建立一个对应的近似向量  $a^h$ , 得到一个近似向量序列  $\{a^0, \dots, a^h, \dots, a^H\}$ . 相应的可以在不同分辨率上分别计算  $a^h$  所对应的  $l_i^h$ , 此时  $\|p_i^h - q^h\| \geq l_i^h$ . 在进行  $k$ -NN 搜索的过程中, 可以先在低分辨率上计算  $l_i^h$ , 然后用  $l_i^h$  与候选查询集中的最大距离  $knn\_u[k] \cdot dist$  进行比较, 如果  $l_i^h \geq knn\_u[k] \cdot dist$ , 那么由  $l_i^h$  的定义和不等式 (5) 可知

$$\|p - q\| = \|p_i^H - q^H\| \geq \|p_i^h - q^h\| \geq l_i^h \geq knn\_u[k] \cdot dist \quad (6)$$

这样, 可以在只进行低分辨率近似距离计算的情况下, 将点  $p_i$  排除掉. 从而在过滤过程中减少运算复杂度, 节省计算时间. 下面给出进行  $k$ -NN 搜索的算法描述. 数组  $knn$  用来保存查询点  $q$  的  $k$  个近邻的标记  $i$  以及它们到查询点  $q$  的距离, 数组  $knn\_u$  用来保存到目前为止的  $k$  个最近邻向量的标记  $i$  以及它们到到查询点  $q$  的上限距离  $u_i$ , 上述两个数组中的元素按照距离的上升次序来排列. 采用堆  $heap\_l$  来保存  $k$  个最近邻向量的标记  $i$  以及它们到查询点  $q$  的下限距离  $l_i$ .

### $k$ -NN 算法

#### 2.1 近似向量过滤过程

1) 初始化数组  $knn\_u$ , 数组中的距离设为 MAXREAL; 初始化堆  $heap\_l$ ; 将  $i$  和  $h$  设为 1.

2) 如果  $h$  等于  $H$ , 执行 3). 如果  $i$  大于  $N$ , 执行 2. 对每个向量的近似值  $a_i^h$ , 计算它到查询点  $q$  的上限距离  $u_i^h$  和下限距离  $l_i^h$ , 如果  $l_i^h > knn\_u[k] \cdot dist$ , 那么排除此向量, 并且将  $i$  值加 1,  $h$  设为 1; 否则, 将  $h$  值加 1, 继续执行 2).

3) 计算  $a_i^H$  到查询点  $q$  的上限距离  $u_i^H$  和下限距离  $l_i^H$ , 如果  $l_i^H > knn\_u[k] \cdot dist$ , 那么排除此向量. 否则, 用  $u_i^H$  和  $i$  替换  $knn\_u[k]$  中的元素, 将数组

$knn\_u$  重新排序, 并且将  $l_i^h$  和  $i$  插入到堆  $heap\_l$  中. 执行 2) 并且将  $i$  值加 1,  $h$  设为 1.

## 2.2 向量运算过程

1) 初始化数组  $knn$ , 数组中的距离设为 MAXREAL.

2) 从堆  $heap\_l$  中弹出堆顶元素  $l_i$  和  $i$ , 如果  $l_i > knn[k]dist$ , 执行 3); 否则用  $\|p_i - q\|$  和  $i$  替换  $knn[k]$  中的元素, 并且将  $knn$  重新排序. 继续执行 2).

3) 算法结束, 数组  $knn$  中保存最终结果.

## 3 实验及算法性能评价

利用图像数据库中的颜色直方图作为高维图像特征, 比较了多分辨率向量近似算法与原算法之间的性能差别. 图像数据库包含 10000 幅不同类别自然图像, 其中部分图像来源于 MPEG-7 内容集和网上一些 FTP 网站, 图像涵盖人物、风景、动植物、建筑等. 试验中近邻搜索中的值为 10. 试验运行环境为微软的 Windows XP 操作系统, 系统配置为 PIV1.7GHz CPU, 384MB 主存. 试验所得结果是进行 100 次 10\_NN 搜索所得的平均结果, 每次进行查询的查询点  $q$  在数据库中随机选择.

针对 256 维颜色直方图, 实验中采用的最高分辨率为 8, 采用多分辨率数据结构的向量近似算法在最高分辨率上计算的颜色块数与原算法相同, 可以保证两种算法得到相同的查询结果. 为了表明多分辨率数据结构的应用为算法性能带来的提升, 我们计算了在不同分辨率上算法访问的数据点的数量. 表 1 为多分辨率的向量近似方法在不同分辨率上的数据点访问数量. 算法中在每一维空间中分别采用了 8 位和 12 位两种近似位数.

表 1 新算法在不同分辨率上的数据访问量

层次	1	2	3	4	5	6	7	8
8 位	10000	3807	1253	516	311	254	250	250
12 位	10000	3791	1232	473	275	202	165	143

由表 1 可以看出, 多分辨率向量近似方法可以在低分辨率上过滤掉大多数候选向量, 从而降低了算法的运算复杂度. 而采用更多近似位数的近似向量具有更好的过滤性能. 进一步的试验中, 我们分别对直方图维数为 32, 64, 128 和 256 的情形下对多分辨率的向量近似方法以及单纯的向量近似方法进行比较. 对上述维数采用的近似位数分别为: 6 位 (维数为 32 和 64), 8 位 (维数为 128 和 256). 我们将近似向量文件一次性装入内存中, 测试两种算法在进行近邻搜索时的运行时间. 图 2 给出了两种算法的 CPU 运算时间比较. 采用多分辨率数据结构的搜索方法运行时间明显小于不采用多分辨率数

据结构的方法.

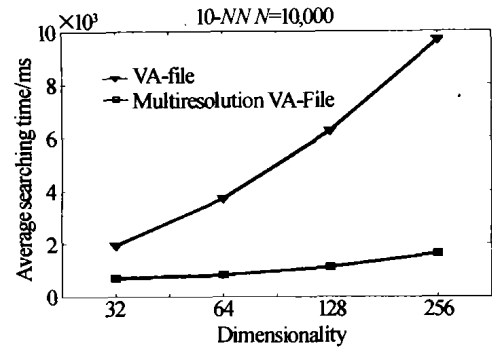


图 2 两种算法近邻搜索平均搜索时间

Fig. 2 Average searching time of two algorithm for K-NN search

采用多分辨率数据结构的向量近似文件大小是原文件的两倍, 可以将低分辨率近似向量与原近似向量分开存储, 由于可以在低分辨率上排除大多数向量, 所以算法不会显著增加二次存储设备的访问时间.

对人工合成数据集中采用了多分辨率向量近似的方法, 人工合成数据集中的数据来源于数据空间  $\Omega = [0, 1]^d$  产生的均匀分布的随机数. 在低分辨率上 ( $h < H$ ) 算法并不具有过滤性能, 经分析可知, 随着分辨率的降低, 在高分辨率上均匀分布的数据点在低分辨率上趋于集中分布, 试验中可以看出集中分布在每一维空间的中间几个区域内, 所以多分辨率的数据结构起不到相应作用.

## 4 结论

将多分辨率数据结构应用于向量近似方法, 通过在低分辨率上过滤掉大多数候选向量, 可以大幅度的降低向量近似方法的计算复杂度. 此方法可以应用于图像数据库各种直方图的查询中, 比如颜色直方图, 亮度直方图以及边缘直方图等, 都能够达到比较好的检索效果和检索速度.

## 参考文献

- Bohm C, Berchtold S, Keim D. Searching in high-dimensional spaces-index structures for improving the performance of multimedia databases. *ACM Computing Surveys*, 2001, **33**(3): 322 ~ 373
- Rui Y, Huang T S, Chang S F. Image retrieval: Current techniques, promising directions, and open issues. *Journal of Visual Communication and Image Representation*, 1999, **10**(4): 36 ~ 92
- Weber R, Schek H J, Blott S. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In: Proc. 24<sup>th</sup> Int. Conf. VLDB, New York, IEEE Press, 1998. 194 ~ 205
- Cha G H, Zhu X, Petkovic D, et al. An efficient indexing method for nearest neighbor searches in high-dimensional

- image databases. *IEEE Trans on Multimedia*, 2002, 5(4): 76 ~ 87
- 5 Song B C, Kim M J, Ra J B. A fast multiresolution feature matching algorithm for exhaustive search in large image databases. *IEEE Trans on Circuits and Systems for Video Technology*, 2001, 11(5): 673 ~ 678
- 6 Lu G. Techniques and data structures for efficient multimedia retrieval based on similarity. *IEEE Trans on Multimedia*, 2002, 11(4): 372 ~ 384
- 7 Manjunath B S, Ohm J R, Vasudevan V V, et al. Color and texture descriptors. *IEEE Trans on Circuits and Systems for Video Technology*, 2001, 11(6): 703 ~ 714

## A Multi-resolution VA-File for High-dimensional Image Feature Matching

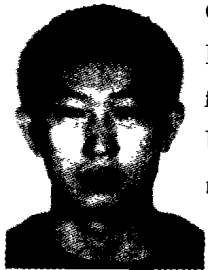
Cui Jiangtao, Liu Weiguang, Zhou Lihua

School of Computer Science and Engineering, Xidian University, Xi'an 710071

Received date: 2004-05-27

**Abstract** The similarity search of images can be transformed into the point matching in the high-dimensional vector space by the feature extraction and transformation. In order to reduce the curse of dimensionality, a new Vector Approximation File approach based on the multi-resolution data structure is proposed. The new approach computes the lower bound of distance from low-resolution level. If it is larger than the latest maximum distance in the result set, the candidate can be removed without calculating the full-resolution distance. The computational time can be dramatically reduced by eliminating improper candidates at lower levels. The algorithm supporting  $k$ -nearest neighbor search is also presented in the new approach and has been applied for feature matching in the large image data sets. The experiment results show that the new approach improves the  $k$ -nearest neighbor search speed and outperforms the Vector Approximation File approach.

**Keywords** High-dimensional image feature; Dimensionality curse; Multi-resolution; Feature matching



**Cui Jiangtao** was born in 1975. He received the B. S. degree in Computer Science and Engineering in 1998 and the M. S. degree in Computer Application of Technology in 2001, both from Xidian University. He is currently for Ph. D. degree in Computer Architecture in Xidian University. His current research interests include image processing, multimedia databases, multimedia application on network.