

基于局部特征聚合的图像检索方法

何云峰¹⁾ 周玲¹⁾ 于俊清^{1),2)} 徐涛¹⁾ 管涛¹⁾¹⁾(华中科技大学计算机科学与技术学院 武汉 430074)²⁾(华中科技大学网络与计算中心 武汉 430074)

摘要 多媒体数据尤其是图像数据的急剧增长,使得基于图像内容的检索成为一个非常重要的研究课题.图像的特征描述以及特征的索引机制是实现基于内容图像检索的关键.针对图像局部聚合描述符(Vectors of Locally Aggregated Descriptors, VLAD)中硬分配难以准确描述局部特征向量与聚类之间隶属关系的问题,采用软分配策略,根据局部特征向量与聚类中心的距离分配不同的隶属权值,生成更具代表性的软分配局部聚合描述符(Soft Assignment-VLAD, SA-VLAD).针对非对称距离计算倒排索引机制(Inverted File with Asymmetric Distance Computation, IVFADC)在查询时为保证结果的查全率而增加候选倒排索引链的数目,导致距离计算和查询时间增加的问题,提出引入简单的散分配方法,将可能落入多条链表中的数据库向量进行多次编码,实现了基于散分配的非对称距离计算倒排索引机制(Dispersed Assignment-IVFADC, DA-IVFADC).实验结果表明:DA-IVFADC机制与SA-VLAD描述符,在很大程度上减少了查询时间,同时有效提高了查询结果的准确率.

关键词 基于内容的图像检索;高维索引;维度灾难;聚合向量
中图法分类号 TP391 DOI号: 10.3724/SP.J.1016.2011.00000

Image Retrieval Based on Locally Features Aggregating

HE Yu-Feng¹⁾ ZHOU Ling¹⁾ YU Jun-Qing^{1),2)} XU Tao¹⁾ GUAN Tao¹⁾¹⁾(School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074)²⁾(Center of Network and Computation, Huazhong University of Science and Technology, Wuhan 430074)

Abstract With the amount of multimedia data, especially image data, increased rapidly, similar image retrieval has become a very important research subject. Features which describe images and feature indexing are important to image retrieval. Vectors of Locally Aggregated Descriptors (VLAD) cannot describe the relationship between local descriptors and clusters. More representative aggregation vector called Soft Assignment-Vectors of Locally Aggregated Descriptors (SA-VLAD) is put forward by using soft-assignment. Soft-assignment is a distributive strategy that the weight assigned to neighboring cells depends on the distance between the local descriptors and the cell centers. Besides, based on Inverted File with Asymmetric Distance Computation (IVFADC), a new indexing scheme named Dispersed Assignment-Inverted File with Asymmetric Distance Computation (DA-IVFADC) is implemented by using dispersed-assignment in indexing stage to resolve the problem of massive distance computation. Experimental results demonstrate that DA-IVFADC and SA-VLAD lessens the query time to a great extent and effectively improves the accuracy rate of results.

Keywords content-based image retrieval; high-dimension indexing; curse of dimension; aggregated descriptors

收稿日期:2011-08-29;最终修改稿收到日期:2011-09-15. 本课题得到国家自然科学基金(61173114, 60903095)、湖北省杰出青年基金(2010CDA084)和中央高校基本科研业务费专项资金(2011QN057, 2011TS094)资助. 何云峰,男,1977年生,博士,讲师,中国计算机学会(CCF)会员,主要研究方向为数字媒体处理与检索. 周玲,女,1987年生,硕士,主要研究方向为数字媒体处理与检索. 于俊清(通信作者),男,1975年生,博士,教授,博士生导师,中国计算机学会(CCF)高级会员,主要研究领域为数字媒体处理与检索、多核处理器编程环境研究. E-mail: yjqing@hust.edu.cn. 徐涛,男,1987年生,硕士,主要研究方向为数字媒体处理与检索. 管涛,男,1978年生,博士,副教授,主要研究方向为数字媒体处理与检索.

1 引 言

随着互联网和多媒体技术的迅速发展,人们可以访问到的多媒体数据,尤其是图像数据,急剧增长.如何实现图像内容的相似性检索成为一个非常重要的研究课题.目前,国内外研究机构已经取得了许多相关成果,开发出一系列商用和原型系统.国内有清华大学的 TV-FI(TV-Find It)系统和浙江大学的 WebscopeCBR,国外有 IBM 研发的 QBIC^[1-2](Query By Image Content)和 Virage^[3]、麻省理工学院多媒体实验室研发的 Photobook、哥伦比亚开发的 VisualSEEK 和 VideoQ 等.

目前,在基于内容的图像检索系统中有两个重要的问题需要解决:

(1) 图像特征. 图像特征包括高层语义特征和低层特征. 由于“语义鸿沟”的存在,高层语义特征难以通过计算机直接获取,以语义关键字检索图像仍然是一项艰巨的任务. 而利用图像低层特征的相似度匹配方法检索图像是另一种重要的图像检索方法,但这种方法依赖于图像特征的有效性. SIFT 特征以其鲁棒性、局部性、辨别能力强的特点,成为图像检索领域最为常用的低层特征之一,但其维度高和数量多的特性又导致了索引和查询的困难. BOF (Bag Of Feature)方法^[4]在局部特征向量基础上,聚合一副图像的数个局部特征向量为一个高维向量,通过对聚合后的向量进行编码并使用倒排索引进行组织,减小了存储空间,降低了查询成本. 但是,减少特征向量的代价是图像的单个特征数据维度非常高,索引时容易引发“维度灾难”^[5]. VLAD(Vectors of Locally Aggregated Descriptors)^[6]是对 BOF 方法的改进,它利用 Fisher kernel^[7]的思想产生查询性能更好、更为简单的聚合描述符. 但是 VLAD 在聚合时采用硬分配策略将局部特征向量分配给聚类类别,从而可能出现两个距离非常近的描述符,被分配到完全不同的聚类中的情况,未能体现向量本身的分布特性,从而降低检索的效率.

(2) 索引结构和方法. 对于大规模的图像数据检索,需要对其特征数据建立索引以提高检索的效率. 而大部分的图像特征数据都是高维数据,不管是基于特征驱动的方法,如 $k-d$ -tree^[8]和 Quad-tree^[9],还是基于数据驱动的方法,如 R-Tree 家族中的 R*-Tree^[10]、R⁺-Tree^[11]、SR-Tree^[12]以及 X-Tree^[13],都难以应对高维特征带来的“维度灾难”.

位置敏感哈希^[14](Locality-Sensitive Hashing, LSH)和基于 BOF 的倒排索引是近年来能较好应对“维度灾难”的索引方法. LSH 虽然可以在一定程度上处理维度高的特征数据但却数倍增大了索引的存储空间,难以适应大规模数据库的检索;而建立在 BOF 基础上的词汇树虽然相对局部敏感哈希已经大大节省了索引空间,但仍然不能满足海量图片检索的性能需求. Jegou 等人^[15]在 VLAD 的基础上利用积量化(Product Quantization, PQ)和非对称距离计算(Asymmetric Distance Computation, ADC)方法,实现非对称距离计算倒排索引机制(Inverted File with Asymmetric Distance Computation, IVFADC),对大规模图像库建立索引,但这种方法只有在 20 Byte 的图像编码下才可保证高于 BOF 的查询效率.

针对海量图像数据和“维度灾难”问题,我们提出了一种基于 BOF 的改进方法——结合 VLAD 与软分配,生成抗降维能力更好、识别率更高的软分配局部聚合描述符(Soft Assignment-Vectors of Locally Aggregated Descriptors, SA-VLAD). 针对索引机制 IVFADC 在查询时为保证结果的查全率和准确率而增加候选倒排索引链,导致距离计算和查询时间增加的问题,在索引阶段针对数据库向量进行散分配,减轻了距离计算负担,同时提高了查询结果质量. 实验表明基于软分配的局部聚合描述符与基于散分配的非对称距离计算倒排索引(Dispersed Assignment-Inverted File with Asymmetric Distance Computation, DA-IVFADC)机制在基于内容的大规模图像数据库检索中获得了良好效果.

本文第 2 节介绍 SA-VLAD,包括软分配权值的设置以及聚合描述符的生成过程;第 3 节详细描述 DA-IVFADC 索引机制,包括散分配、积量化、非对称距离计算以及整个索引与检索机制;第 4 节给出 DA-IVFADC 与 SA-VLAD 的实验结果;第 5 节总结全文并提出下一步的工作计划.

2 基于软分配的局部聚合描述符

SA-VLAD 与 VLAD 一样,是一幅图像多个局部描述符的聚合体. SA-VLAD 采用软分配方法代替硬分配,弥补了 VLAD 的不足,生成更为准确的图像描述符.

2.1 隶属权值计算

软分配思想来源于 1965 年 Zadeh 提出的模糊集理论,该理论认为每个元素都是以一定的程度隶

属于某个集合,也可以不同程度地属于几个集合.与传统的“是或者不是”的二值逻辑相比,模糊集理论更能体现出实际应用中的对象隶属关系.软分配方法根据特征向量到聚类中心的距离远近分配不同的权值,称其为特征向量相对聚类的隶属权值.

隶属权值的计算采用模糊 k 均值方法中的隶属度函数计算.模糊 k 均值中要求聚类中的向量与聚类中心误差平方和最小,通常使用准则函数来衡量,准则函数越小,误差越小.准则函数为

$$J_e = \sum_{i=1}^k \sum_{j=1}^n [u_i(\mathbf{S}_j)]^b \|\mathbf{y} - \mathbf{c}_i\|^2 \quad (1)$$

其中, \mathbf{c}_i 是第 i 个聚类的聚类中心; k 是聚类的数目; \mathbf{S}_j 是 n 个局部特征向量中的第 j 个向量; $u_i(\mathbf{S}_j)$ 是特征向量 \mathbf{S}_j 对于第 i 个聚类的隶属度; b 是一个大于 1 的常数,用于控制聚类模糊程度.这里 b 取值为 2.

模糊均值聚类要求每个局部特征向量相对于各个聚类的隶属度之和为 1,即

$$\sum_{i=1}^k u_i(\mathbf{S}_j) = 1, \quad j = 1, 2, \dots, n \quad (2)$$

在隶属度和为 1 的情况下求得 J_e 的最小值,令 J_e 对 \mathbf{S}_j 和 $u_i(\mathbf{S}_j)$ 的偏导数为 0,得必要条件为

$$c_i = \frac{\sum_{j=1}^n [u_i(\mathbf{S}_j)]^b \mathbf{S}_j}{\sum_{j=1}^n [u_i(\mathbf{S}_j)]^b}, \quad i = 1, 2, \dots, k \quad (3)$$

$$u_i(\mathbf{S}_j) = \frac{(1/\|\mathbf{S}_j - \mathbf{c}_i\|^2)^{\frac{1}{(b-1)}}}{\sum_{h=1}^k (1/\|\mathbf{S}_j - \mathbf{c}_h\|^2)^{\frac{1}{(b-1)}}}, \quad j=1, 2, \dots, n, \quad i=1, 2, \dots, k \quad (4)$$

由此得到 \mathbf{S}_j 在第 i 个聚类的隶属度函数 $u_i(\mathbf{S}_j)$.

2.2 SA-VLAD 生成算法

SA-VLAD 使用 k -means 方法获得 k 个聚类中心,然后采用误差最小的分配原则,借鉴模糊均值聚类的软分配方法分配局部特征向量,获得隶属权值并计算聚合描述符.其生成过程如图 1 所示.

基于 k 个聚类进行局部特征向量的聚合生成 SA-VLAD 描述符的详细步骤为

(1) 初始化 SA-VLAD 为维度为 $k \times D$ 的零向量 \mathbf{sv} . k 为聚类中心数目, D 为图像局部特征向量的维度.

(2) 每幅图像的局部特征向量 \mathbf{S} 在所有聚类中心中通过近邻查找得到 t 个与其距离最近的聚类中心:

$$\begin{aligned} c(\mathbf{S})_h = c_i = \\ \{c_i \in \text{codebook} \mid \forall c_j \in \text{codebook} \setminus (c_i \cup c(\mathbf{S})_m), \\ \|\mathbf{p} - \mathbf{c}_i\| \leq \|\mathbf{p} - \mathbf{c}_j\|, 1 \leq i \leq k, 1 \leq j \leq k, h < m < t\} \end{aligned} \quad (5)$$

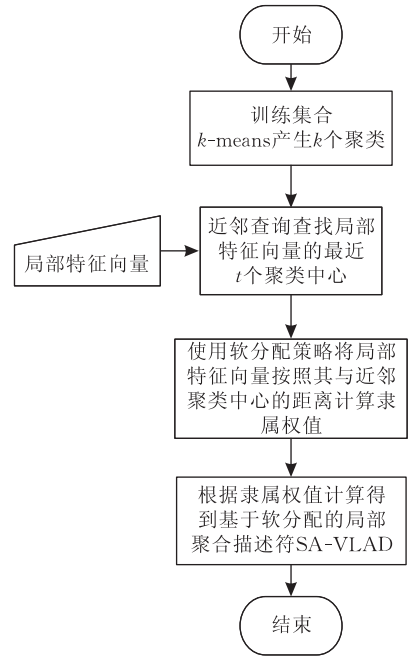


图 1 SA-VLAD 生成示意图

其中, $c(\mathbf{S})_h$ 代表与向量 \mathbf{S} 第 h 近的聚类中心向量. \mathbf{S} 与 $c(\mathbf{S})_h$ 的差值体现了 \mathbf{S} 经过聚类中心映射后的分布情况, **codebook** 是由所有的聚类中心向量集合而成的编码本向量.

常数 t 表示特征向量最多隶属于聚类的个数. t 太大,则隶属的聚类多,获取聚类的时间长,并且与特征向量距离远的聚类因隶属权值太小而没有意义. t 值太小,则不能发现特征向量与聚类之间的实际隶属关系.这里 t 取值为 4.

(3) 计算特征向量 \mathbf{S} 在 t 个最近聚类的隶属权值,并利用隶属权值计算 \mathbf{S} 与 t 个最近聚类中心的差值.一个图像所有局部特征向量与其最近聚类中心的差值的集合即为其 SA-VLAD 描述符. \mathbf{sv}_i 为 d 维的向量,表示图像的 SA-VLAD 描述符在第 i 个聚类中心位置上的差值和,得 \mathbf{sv}_i 为

$$\mathbf{sv}_i = \sum \frac{1}{u_i(\mathbf{S})} (\mathbf{S} - \mathbf{c}_i) \quad (6)$$

$$\mathbf{sv} = (\mathbf{sv}_1, \dots, \mathbf{sv}_i, \dots, \mathbf{sv}_k) \quad (7)$$

\mathbf{sv} 即为图像的 SA-VLAD 特征值.

SA-VLAD 可直接参与顺序检索或者使用索引机制对其进行组织.然而,由于 SA-VLAD 维度很高,对其进行索引之前需要降维,且由图像本身的特性可知,一副图像的局部特征向量如 SIFT,在进行类别分配时,很多会被分配到相同的聚类中,即大量 SIFT 向量的邻近聚类中心相同,且存在没有包含任何特征向量的聚类.经过软分配计算后,在没有特征点落入的聚类中心的位置可能存在大量的连续零

值. 因此可以使用线性降维方法对高维度的 SA-VLAD 进行降维. 第 4 节的实验证明, 相同幅度的降维, SA-VLAD 比 VLAD 损失的精度更小, 查询时平均准确率下降得更小, 说明 SA-VLAD 的抗降维能力更好. 降维后的聚合描述符可顺序扫描查询也可以采用任何适用于特征向量的索引机制对其进行索引和查询.

3 基于散分配的非对称距离计算倒排索引机制

索引机制是基于内容的图像检索的核心, DA-IVFAD 索引机制弥补了 IVFADC 机制的不足, 实现了对特征向量进行精简的编码、有效的距离计算以及高效的倒排链存储.

3.1 散分配

散分配是数据对象在量化过程中的一种分配策略, 借鉴了模糊分类以及 LSH 中多函数映射的思想. 精确分类中对象与聚类之间是一一对应的关系, 一个对象仅隶属于一个聚类, 且完整隶属于一个聚类; 而模糊分类中, 对象与聚类之间是一一对应的关系, 一个对象可能隶属于多个聚类. LSH 分配思想是采用 L 种不同的映射函数, 将每个数据库对象分配给 L 个不同的哈希表, 每个表都完整包含所有的数据库

对象, 同一对象在不同表中产生不同编码.

散分配以模糊分类为基础, 在 LSH 中使用一个对象完整隶属于多个聚类且在每个聚类中为不同编码的策略, 即在 k -means 聚类后获得与对象距离最近的 t 个聚类中心, 当对象与聚类中心距离满足一定阈值时, 将其分别完整地分配到一个或多个聚类中, 并计算相应的剩余向量.

在 IVFADC 索引机制中, 索引阶段数据库向量量化后被唯一的分配到某个链表中, 由于量化方法难免产生误差, 所以为保证结果的查全率, 查询时需加大候选链表的值, 导致计算量与计算时间增加. 因此, 在 IVFADC 中, 高的结果查全率与快的查询速度之间存在一定矛盾. 而散分配可以有效解决这一问题, 在保证查全率的情况下减少查询时间.

3.2 基于散分配的非对称距离计算倒排索引的生成

引入散分配的 DA-IVFADC 索引生成过程如图 2 所示. DA-IVFADC 组织图像库中特征向量, 建立索引的详细步骤如下:

(1) 全局量化数据库向量并进行散分配. 预先使用 k -means 方法对训练集进行聚类, 得到 k 个聚类及其聚类中心向量, 每个聚类认为是最后的存储结构中的一条倒排索引链. 对于输入的数据库向量 y , 获取 n 个与其距离最近的聚类中心.

常数 n 的值域为 $[2, 5]$, 太大的 n 值会增加距离

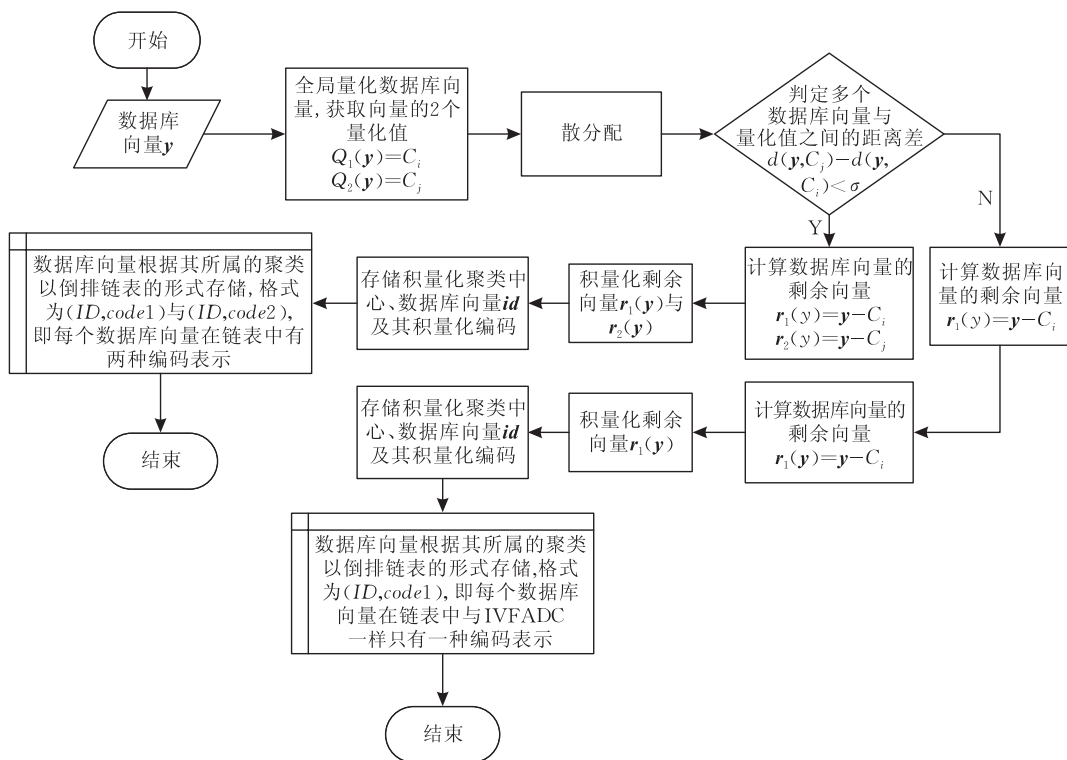


图 2 DA-IVFADC 索引生成示意图

的计算量. 这里, 设置 n 取最小值 2, 即数据库向量散分配到聚类的数目最多为 2 个, 此时获取与向量 y 最近的聚类中心 c_1 与次近的聚类中心 c_2 .

(2) 计算 $d(y, c_1)$ 和 $d(y, c_2)$, 确定数据库向量如何分配到两个聚类中. 如果 $d(y, c_2) - d(y, c_1) \geq \sigma$, 只将数据库向量分配到最近聚类中, 计算剩余向量 $r_1(y)$; 否则, 数据库向量同时分配到两个聚类中, 计算剩余向量 $r_1(y)$ 和 $r_2(y)$;

σ 值是一个阈值, 太大会大量增加存储空间, 过小则使得 DA-IVFADC 退化为 IVFADC. σ 值的目的在于将那些位于聚类之间、距离多个聚类中心距离相差甚小的对象, 同时分配到多个聚类中, 为后续的查询提供便利.

剩余向量 r 是原始向量与其全局量化聚类中心的差值, 向量 y 的剩余向量为

$$r(y) = y - Q(y) \quad (8)$$

(3) 对剩余向量值进行积量化, 得到 $p(r(y))$, 将得到的编码以及图像或者向量 ID 分别添加到对应的倒排索引链中.

积量化编码的思想是“子空间组合编码”. 为避免编码本中码字过多时全局量化带来的时间与空间的高额开销, 积量化将高维度的向量等分为多个子部分, 即划分子空间. 在子空间里进行独立的量化, 然后组合子空间的码字获取整个空间的编码本. r 为一个 D 维向量, $r = \{y_1, y_2, \dots, y_D\}$, 等分为每部分为 d' 维的 m 个子部分, 每个子部分的量化器为 p_j :

$$\underbrace{y_1, \dots, y_{d'}}_{s_1(x)}, \dots, \underbrace{y_{D-d'+1}, \dots, y_D}_{s_m(x)} \rightarrow p_1(s_1(y)), \dots, p_m(s_m(y)) \quad (9)$$

由式(9)可知, 每个子量化器 p_j (如均值聚类) 在 j 个子空间内产生独立的码字 $c_{j,i}$ 和编码本 l_j , 所有子空间产生相同数目维度为 d' 的码字, 整个空间的编码本是所有子空间编码本的笛卡尔积:

$$l = l_1 \times l_2 \times \dots \times l_m \quad (10)$$

由式(10)可知, 整个空间的码字可能是 m 个子空间中任意一个码字的任意组合. 以 k -means 为例, 每个子空间中产生码字 k' 个, 从组合角度出发, 一个 D 维度的向量, 在第 j 个子空间中码字可能为 k' 中的一个, 整个向量的编码是从 $1 \sim m$ 个子部分编码的排列, 从排列组合角度出发, 整个空间的码字数目为

$$k = (k')^m \quad (11)$$

当 $m=D$ 时, 向量的每个维度被看成是一个子部分, 单独进行量化, 转化为实数空间的量化; 当

$m=1$ 时, 即传统的全局量化方式, 即在整个向量空间中进行量化.

DA-IVFADC 与 IVFADC 中的积量化建立在所有原始向量的剩余向量上, 即在整个的框架中, 只存在一个积量化. 实验证明, 单个的积量化仍然可以获得很好的查询效果, 相比为每个聚类的剩余向量设计一个积量化, 节省了存储空间. 如果每一个聚类都使用不同的积量化, 生成不同的积量化聚类中心, 那么 m 个子空间则有 m 个量化器, 有 $m \times k' \times d'$ 个浮点型数据需要存储.

设全局量化为 Q , 积量化为 p , 数据库向量 y 的近似值表示为

$$\ddot{y} = Q(y) + p(y - Q(y)) \quad (12)$$

y 可用全局量化与积量化的二元组表示 $(Q(y), p(r(y)))$, 全局量化后的 $Q(y)$ 决定的是向量所属的链表, 积量化 $p(r(y))$ 表示向量的编码.

索引过程结束后, 每个数据库向量使用 ID 以及编码来标识. ID 是向量在数据库中的唯一标识符, 编码是向量经过全局量化与积量化后得到的聚类中心的联合. ID 与编码是数据库向量在倒排链中的最终表示, 如表 1 所示.

表 1 DA-IVFADC 向量编码

区域	长度/bit
标识符(ID)	8~32
编码	$m \times \lceil \log_2 k' \rceil$

3.3 基于散分配的非对称距离计算倒排索引的查询

DA-IVFADC 的查询过程与索引过程类似, 同样有全局量化的过程, 但是仅对数据库向量进行积量化, 并不对查询向量的剩余向量实施积量化, 量化后的向量与未量化向量之间的距离计算称之为非对称距离计算, 如图 3 所示.

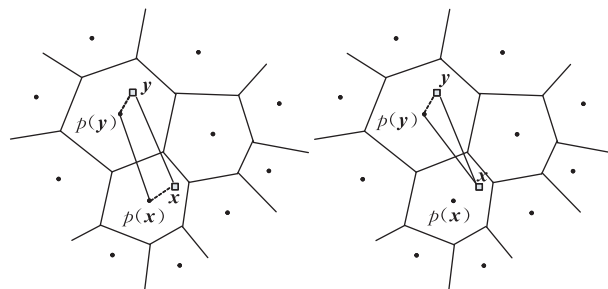


图 3 对称距离计算(左)与非对称距离计算(右)

y 是数据库索引向量, x 是检索向量. $p(x)$ 与 $p(y)$ 分别为 x 和 y 所属聚类的中心点, 是 x 和 y 量化后的代表向量. 在非对称距离计算 ADC 中, 数据库特征向量 y 量化后使用 $p(y)$ 表示, 查询向量 x 不

采取任何量化措施, y 与 x 之间的距离表示为

$$\tilde{d}(x, y) = d(x, p(y)) = \sqrt{\sum_{j=1}^m d(s_j(x), p_j(s_j(y)))^2} \quad (13)$$

图 3 中可看出, 对称距离计算 (Symmetric Distance Computation, SDC) 是计算量化后的查询向

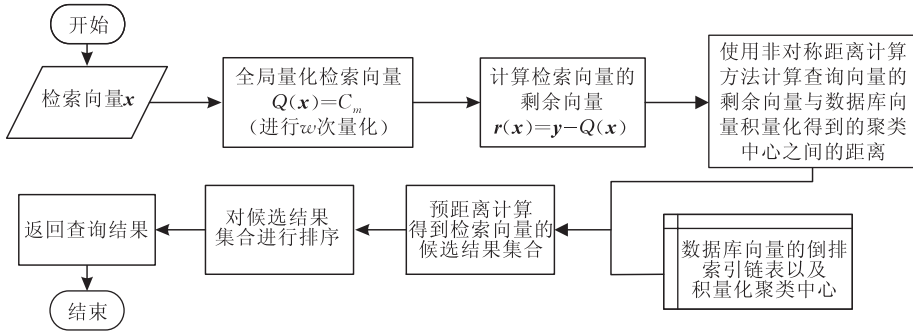


图 4 DA-IVFADC 查询示意图

步骤如下:

1. 根据 k -means 计算查询向量 x , 同时获取 w 个聚类中心, 即 w 条倒排索引链, 以下步骤使用 $r(x)$ 代表 x 与 w 个聚类中心的剩余向量, 全局聚类中心数目即为 k' .
2. 针对每一个子空间和空间中的每一个聚类中心 $c_{j,i}$, 计算平方距离 (非对称距离).
3. 计算 $r(x)$ 与 w 条链表中的对象的距离平方. 步 2 得到了查询向量与链表中向量各个分量的距离, 将 m 个子距离简单相加即可.
4. 选取距离最小的 k 个数据库对象作为最后结果返回.

从查询过程中看出, 大量的计算开销花费在步 2, 计算开销与 w 成正比, w 每增加 1, 计算量增加 $(k')^2$, 而 IVFADC 通过增加 w 的值保证结果的查全率, 增大了距离计算, 进一步证明了 IVFADC 的不足. DA-IVFADC 在索引过程中引入散分配, 可以有效减小 w 值, 加快查询, 保证查全率. 尽管散分配会增加索引时间, 但是整个数据库的索引过程是预先处理的, 不会对查询造成影响.

4 实验结果与分析

4.1 评价指标

实验采用平均准确率 (mean Average Precision, mAP) 作为性能指标来评价检索系统. 设 $p(R)$ 是系统在召回率为 R 时的准确率, mAP 定义为

$$mAP = \int_0^1 p(R) dR \quad (14)$$

在以召回率为横坐标, 准确率为纵坐标的评价曲线上, 理论证明, 最理想的系统, mAP 包含的面积

量与量化后的数据库向量之间的距离. 相比非对称距离计算其优势在于对称距离计算对查询向量进行编码, 节省了空间. 但使用非对称距离计算更能体现对象之间的相似度, 减小量化带来的距离误差.

基于 DA-IVFADC 的查询过程如图 4 所示.

是 1, 即曲线与坐标轴面积越大, 认为系统性能越好.

相似性检索的另一个评价标准是 $\text{recall}@R$. $\text{recall}@R$ 指的是在检索对象的结果集合中, 其最邻近的对象出现在前 R 个位置的概率. $\text{recall}@R$ 常用于近似最近邻检索的测试.

4.2 实验平台

测试使用的软硬件环境为: 硬件环境: Intel(R) Core(TM)2 Quad CPU Q8200@2.33GHz, 4GB 内存. 操作系统: Windows XP. 编程环境: Matlab 7.10.0 (R2010a), Microsoft Visual Studio 2005.

4.3 测试数据集

为验证 DA-IVFADC 与 SA-VLAD, 针对 DA-IVFADC 准备了两个数据库, 每个数据库包含 3 个子集合, 训练集合、索引数据集以及查询数据集. 表 2 是各个数据库向量维度与规模的具体情况.

表 2 DA-IVFADC 测试数据库

数据库名	特征向量的维度	训练集合大小/个	数据库集合大小/个	检索库大小/个
SIFTSMALL	128	25000	10000	100
SIFT	128	100000	1000000	10000
RANDOM	64	25000	10000	100

(1) RANDOM 数据库. 由机器随机生成 10^4 个维度为 64 的浮点数向量, 训练集合大小为 25×10^3 个向量, 数据库集合大小为 10^4 个向量, 查询集合 100 个向量. 目的在于测试 DA-VLAD 对于无规律、各个维度完全独立的非结构化数据的查询性能;

(2) SIFT 数据库, 内容为 128 维度的 SIFT 特征, 训练集合大小为 10^5 个向量, 提取自 Flickr 相册, 索引数据集大小为 10^6 个向量, 查询集合 10^4 个向量,

提取自 INRIA 假日图像库^[16]; 其中 SIFTSMALL 是 SIFT 的子集.

针对 SA-VLAD, 准备了两个数据库, 规模一大一小, SA-VLAD 是聚合描述符, 所以数据库规模与图像数目一致; 小规模图像库采用的是 INRIA 假日图像库, 图像的分辨率为 1024×768 . 此图像库包含 1491 个图像, 分为 500 组, 每组图像分别来自对不同景物的不同视觉的拍摄, 或者经过一定旋转变换. 大规模图像库则包含 INRIA 假日图像库和 Flickr 相册.

4.4 实验结果分析

4.4.1 积量化参数 m 与 k' 对结果的影响

积量化的两个重要参数是子空间数目 m 以及积量化的聚类数目 k' , m 与 k' 决定了向量的编码位数为 $L_{\text{code}} = m \times \log_2 k'$.

ADC 与 SDC 计算是两种不同的计算方式, 他们分别与积量化结合构成两种不同的索引机制: 无全局量化过程, 直接对原始向量进行积量化, 候选结果是整个数据库, 而不是类似 DA-IVFADC 机制中落入候选的 w 条链表中的数据. 在未引入倒排表的查询机制下, 可以清楚看到 m 与 k' 对结果 $\text{recall}@R$ 影响. 图 5 和图 6 分别是基于 SIFTSMALL 数据库, ADC 和 SDC 索引机制在 $m = \{1, 2, 4, 8, 16\}$, $k' = \{24, 26, 28\}$ 时 $\text{recall}@100$ 的实验结果图.

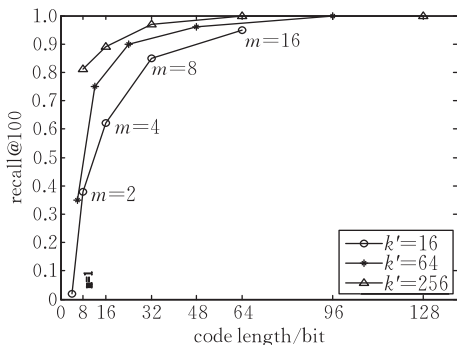


图 5 ADC 索引机制中参数设置对 $\text{recall}@100$ 的影响

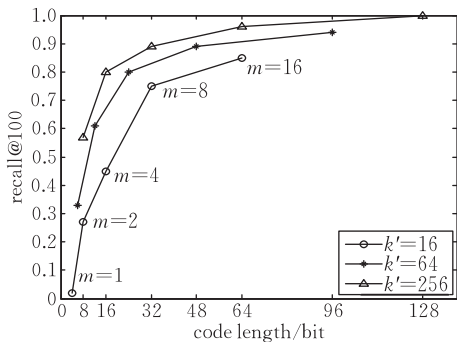


图 6 SDC 索引机制中参数设置对 $\text{recall}@100$ 的影响

由图 5 和图 6 可以看出, 在相同编码长度即 L_{code} 相同的情况下, 子空间聚类中心 k' 多而子空间

数目 m 少相比子空间数目多聚类中心少所取得的效果要好. 编码位数为 64, $m=8, k'=256$ 时 $\text{recall}@100$ (三角形线) 都比 $m=16, k'=16$ 的 $\text{recall}@100$ (圆形线) 要高.

参数相同时, ADC 比 SDC 的 $\text{recall}@100$ 值要高. 在图 5 中, $m=8, k'=64$ 时 $\text{recall}@100$ 接近 1. 图 6 中, $m=8, k'=256$ 时 $\text{recall}@100$ 才接近 1, 此时 SDC 存储积量化聚类中心的空间远比 ADC 要大. ADC 的优越性证明 IVFADC 以及 SA-IVFADC 索引机制中仅对数据库数据进行量化, 对查询数据不进行量化是明智的选择.

积量化的理论证明, 针对半结构化的特征数据, 按照向量本身维度之间的独立性对高维向量进行有规律的子空间划分可以保证较好的检索性能. 从 SIFT 特征的提取方法得知, 当 $m=8$ 时, 积量化的每个子空间为 16 维度, SIFT 特征的每 16 个维度正好是一组, 符合各个子空间独立性高、能量平衡的原则. 实验结果证明, $m=8, k'=256, L_{\text{code}}=64$ bit 时 SIFT 特征在 ADC 与 SDC 中 $\text{recall}@100$ 都接近 1. 所以在 DA-IVFADC 中设定 $m=8, k'=256$.

4.4.2 DA-IVFADC 中 k^* 、 m 及 k' 、 w 对结果的影响

w 是查询时候选链表的数目, m 是积量化子空间数目, k' 是每个子空间中的聚类数目, k^* 是全局量化时的聚类数目, 即链表数目.

图 7 是 SIFT 数据库在 DA-IVFADC 积量化中子空间聚类中心为 $k'=256, m = \{1, 2, 4, 8, 16\}$, $k^* = \{1024, 4096\}$, $w = \{1, 4, 8, 64\}$, 散分配参数 $n=2$, 距离阈值 $\sigma=1000$ 时记录的 $\text{recall}@100$ 数据.

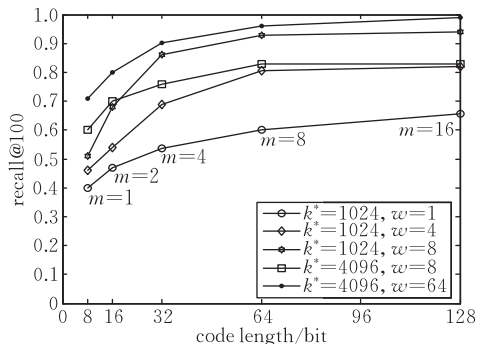


图 7 DA-IVFADC 索引机制中参数对 $\text{recall}@R$ 的影响

从图 7 中曲线变化看出, 当 k^* 与 w 确定时, 编码位数的增长并不能明显提高查询效率, 曲线在 $m=8$, 编码长度为 64 bit 后趋近平缓. 原因是当 k^* 确定后, 每条链表中的数据库对象已经确定, w 确定后, 候选结果对象确定. m 增加到一定程度, 所有落入 w 条链表中的相似对象已经返回, 数据库中其余

的近邻对象因为不落在 w 条链表中的任何一条, 在 w 确定后已经被过滤掉, 所以曲线逐渐平缓. k^* 值确定, w 越大, 查全率越高, $\text{recall}@100$ 相应增加, 但 w 不是越大越好, w 越大, 计算检索对象子空间剩余向量与候选链表子空间聚类中心之间距离计算的时间代价越大, 影响查询效率. k^* 值越大, 数据库向量分的越细, 要保证结果质量, 必须增加 w 的值, k^* 值过小, 退化为 ADC 查询, 不能充分发挥全局量化分类更细以及倒排链加速查询的作用.

因此, DA-IVFADC 的各参数值需根据数据库实际情况进行设计.

4.4.3 DA-IVFADC 与 IVFADC 的比较

图 8 是 RANDOM 测试库在 $m=8, k'=256, L_{\text{code}}=64 \text{ bit}, k^*=64, w=\{1, 2, 10, 16\}$ 时, IVFADC 与 DA-IVFADC 的 $\text{recall}@100$ 对比图, 因为 $R>20$ 以后曲线基本没有变化, 所以只画出 $R<20$ 的部分. 图 9 是相应的查询时间对比图, 结合两图可以看出, w 相同时, 查询时间 time 相差甚小, 但 DA-IVFADC 的准确率明显高于 IVFADC. IVFADC 在 $w=16, \text{time}=12.9$ 时 $\text{recall}@20=0.98$, 而 DA-IVFADC 在 $w=10, \text{time}=8.2$ 时 $\text{recall}@20>0.98$.

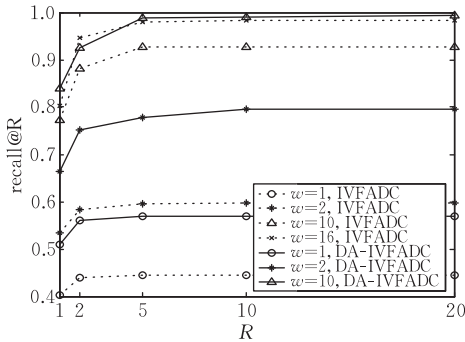


图 8 RANDOM 数据库中的 $\text{recall}@R$ 值

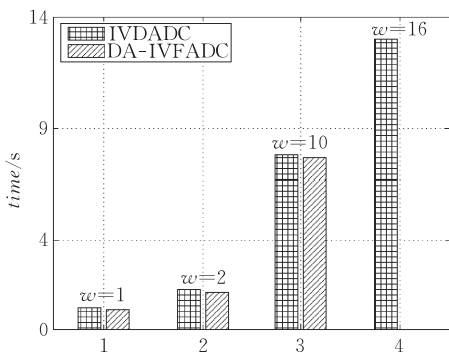


图 9 RANDOM 500 个检索对象查询时间图

图 10 的测试数据库是 1 MB 的 SIFT 特征向量在 $m=8, k'=256, L_{\text{code}}=64 \text{ bit}, k^*=1024, w=\{1, 2, 8, 16\}$ 时, IVFADC 与 DA-IVFADC 的 $\text{recall}@20$

对比图, 图 11 是相应的查询时间对比图. 图 9 统计的是 500 个检索对象的查询时间, 图 11 统计的是 10000 个检索对象平均每个对象的查询时间. 结合图 10 与图 11 可以看出, DA-IVFADC 在大数据量情况下性能仍然优于 IVFADC.

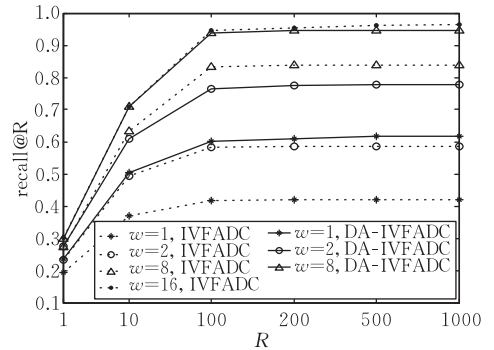


图 10 SIFT 数据库中的 $\text{recall}@R$ 值

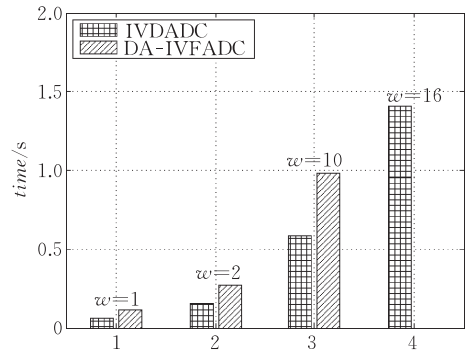


图 11 SIFT 数据库的平均查询时间图

4.4.4 基于软分配的局部聚合描述符实验结果

(1) SA-VLAD 与 VLAD 在没有索引机制时的查询情况.

表 3 是 SA-VLAD 与 VLAD 是在 INRIA 假日数据库上, 没有任何索引机制下的 mAP 值. 表中列出了聚类数目 $k=\{16, 64\}$, 降维后维度 $D'=\{256, 128, 64, 32\}$ 的情况. 从表中数据可以看出, $k=64$ 比 $k=16$ 时得到的聚合向量性能好. k 值越大, 生成聚合向量的过程花费越大, 聚合向量的维度越高, 降维带来的负担与误差越大; k 值越小, 虽然降维负担小, 但是聚合描述符的查询性能会下降. 下面的实验将针对 $k=64$ 时的情况讨论.

从表 3 中的数据可知, 在没有索引机制情况下 $D'=256$ 时 mAP 较高. 但是, 海量数据库的检索不能离开索引机制, 高维度的数据会增大索引与检索的负担, 针对文章中的索引机制, 有必要观察在不同 D' 值时 ADC 与 SA-IVFADC 情况下的 mAP 值, 从而获取一个最佳 D' .

表 3 无索引机制下 SA-VLAD 与 VLAD 的平均准确率

	k	D	平均准确率			
			$D'=256$	$D'=128$	$D'=64$	$D'=32$
VLAD	16	0.449	0.447	0.445	0.442	0.384
	64	0.516	0.505	0.500	0.470	0.416
SA-VLAD	16	0.446	0.445	0.442	0.441	0.392
	64	0.524	0.529	0.510	0.490	0.440

(2) 索引机制下 D' 的选择

表 4 是原始维度为 8192 的 SA-VLAD 降维到不同维度后,在无索引机制、ADC 以及 SA-IVFADC 索引机制下的 mAP 值.从表中数据可以看出,不管是无索引机制还是在 ADC 或者 DA-IVFADC 索引机制下, $D'=64$ 时获得的 mAP 值总和最高.

表 4 不同降维幅度下 SA-VLAD 的平均准确率

	平均准确率			
	$D'=256$	$D'=128$	$D'=64$	$D'=32$
None index	0.529	0.510	0.490	0.440
ADC	0.409	0.435	0.464	0.435
DA-IVFADC	0.411	0.423	0.446	0.409

(3) 大规模数据库检索中 SA-VLAD 与 VLAD 性能对比

图 12 是在 1 MB 图片数据库下记录的 VLAD 与 SA-VLAD 在 3 种不同索引机制下的 mAP 值.分别是降维后 $D'=64$ 时,无索引机制、ADC 索引机制以及 DA-IVFADC 机制下的 mAP 值. DA-IVFADC 中, $m=16$, $k'=256$, $L_{code}=16$ Byte. 由图 12 可以看出,聚合向量从 8192 维降到 64 维度, SA-VLAD 比 VLAD 的 mAP 值下降幅度慢,说明采用 PCA 降维时,前者比后者带来的误差更小,数据量增大, mAP 值都降低,同样的索引机制时, SA-VLAD 的 mAP 都优于 VLAD 的 mAP ,差距在 0.05 以内,说明 SA-VLAD 仍然有改进的空间.

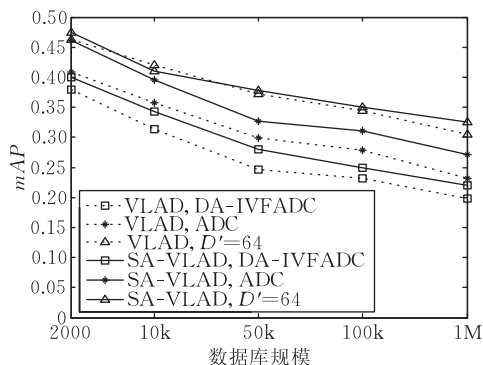


图 12 SA-VLAD 与 VLAD 在不同索引机制下的平均准确率

SA-VLAD 与 VLAD 查询时间的差距就是检索对象生成聚合向量的时间差.经统计,平均每个对象的查询时间差距为 0.09 ms.由此可见, SA-VLAD 在不明显增加查询时间、存储空间与 VLAD

保持一致的情况下,提高了查询结果的准确率.

而且, $D'=64$ 时,在 DA-IVFADC 中索引 SA-VLAD 向量,每幅图像仅需占用 20 Byte (16 Byte 的 code + 4 Byte 的 ID).当图像库规模上亿时, DA-IVFADC+SA-VLAD 仍然可以实现数据库基于内存的索引.

(4) 查询结果对比

图 13 是 VLAD 与 DA-IVFADC 结合的查询结果示例图,图 14 是 SA-VLAD 与 DA-IVFADC 结合的查询结果示例图.两图中每列的第一个图像为检索图像,每幅检索图像返回四个查询结果.



图 13 VLAD+DA-IVFADC 查询结果示例



图 14 SA-VLAD+DA-IVFADC 查询结果示例

5 结束语

SA-VLAD 聚合向量算法以及散分配的 DA-IVFADC 索引机制,一定程度上提高了查询性能.下一步的研究主要包括以下几个方面:

(1) 对多种局部描述符生成聚合向量进行研究,使聚合向量算法适用更广泛;

(2) 生成 SA-VLAD 聚合向量时,针对不同图像的特征,设计与实际分布更为贴切的软分配算法;

(3) 索引机制与聚合向量都依赖训练集合产生的编码本,大量的增删操作会导致中心点的偏移,重新训练编码本代价巨大,如何让基于训练产生编码本的索引机制适应频繁的插入删除操作,也是下一步要解决的主要问题.

参 考 文 献

- [1] Chang S K, Hsu A. Image information system: Where do we go from here? *IEEE Transactions on Knowledge and Data Engineering*, 1992, 4(5): 431-442
- [2] Niblack W, Barber R, Equitz W, et al. The QBIC project: Querying images by content, using color, texture and shape//*Proceedings of the SPIE Storage and Retrieval for Image and Video Databases*. San Jose, USA, 1993: 173-187
- [3] Bach J, Fuller C, Gupta A. Virage image search engine: An open framework for image management//*Proceedings of the SPIE Conference on Storage and Retrieval for Image and Video Databases IV*. San Jose, USA, 1996: 76-87
- [4] Panigrahy Rina. Entropy based nearest neighbor search in high dimensions//*Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms*. Miami, USA, 2006: 1186-1195
- [5] Berchtold S, Ertl K B, Kriegel H P. The pyramid-technique: Towards breaking the curse of dimensionality//*Proceedings of the 1998 ACM SIGMOD International Conference on Management of data*. Washington, USA, 1998: 142-153
- [6] Jégou Hervé, Douze Matthijs, Schmid Cordelia. Aggregating local descriptors into a compact image representation//*Proceedings of the IEEE 23rd Conference on Computer Vision and Pattern Recognition*. San Francisco, USA, 2010: 3304-3311
- [7] Perronnin F, Dance C R. Fisher kernels on visual vocabularies for image categorization//*Proceedings of the IEEE 10th Conference on Computer Vision and Pattern Recognition*. Minneapolis, USA, 2007: 1-8
- [8] Bentley J L. Multidimensional binary search trees used for associative Searching. *Communications of the ACM*, 1975, 18(9): 509-517
- [9] Raphael Finkel, Bentley J L. Quad trees: A data structure for retrieval of composite keys. *Acta Informatica*, 1974, 4(1): 1-9
- [10] Beckman N, Kriegel H, Schneider R. The R^* -tree: An efficient and robust access method for points and rectangles//*Proceedings of the 1990 ACM SIGMOD International Conference on Management of Data*. New Jersey, USA, 1990: 322-331
- [11] Sellis T, Roussopoulos N, Faloutsos C. The R^+ -tree: A dynamic index for multi-Dimensional objects//*Proceedings of the 13th International Conference on Very Large Data Bases*. Brighton, UK, 1987: 507-518
- [12] Katayama N, Satoh S. The SR-tree: An index structure for high dimensional nearest neighbor queries//*Proceeding of the ACM SIGMOD Conference on Management of Data*. Tucson, USA, 1997: 369-380
- [13] Berchtold S, Keim D A, Kriegel H P. The X-Tree: an index structure for high-dimensional data//*Proceedings of the 22nd VLDB Conference*. Mumbai, India, 1996: 28-39
- [14] Gionis A, Indyk P. Similarity search in high dimensions via hashing//*Proceedings of the 25th International Conference on Very Large Data Bases*. Edinburgh, UK, 1999: 518-529
- [15] Hervé Jégou, Douze Matthijs, Schmid Cordelia. Product quantization or nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010, 33(1): 117-128
- [16] Jégou Hervé, Douze Matthijs, Schmid Cordelia. Hamming embedding and weak geometric consistency for large scale image search//*Proceedings of the 10th European Conference on Computer Vision*. Marseille, France, 2008: 304-317



HE Yun-Feng, born in 1977, Ph.D.. His research interest is digital media processing and retrieval.

ZHOU Ling, born in 1987, M. S.. His research interest is digital media retrieval.

YU Jun-Qing, born in 1975, Ph. D. , professor. His research interests include digital media processing and retrieval, multi-core programming environment.

XU TAO, born in 1987, M. S.. His research interest is digital media retrieval.

GUAN Tao, born in 1978, Ph. D. , associate professor. His research interest is digital media retrieval.

Background

With the development of Internet and multimedia technology, the amount of multimedia data, especially image data, are increased rapidly. How to search image data based on image content, called CBIR (Content based image Retrieval), has become a very important research subject. Generally, similarity between image objects can be measured by distance between feature vectors. Due to the characteristic of high dimension, high-dimension indexing mechanism becomes the key technology for content-based retrieval in large-scale image database. To resolve the problem of ‘curse of di-

mension’, organizing a large number of high-dimension feature vectors and designing efficient high-dimension indexing methods have become a great challenge for all the researchers. The paper is focus on the feature description of image and the high-dimension feature indexing, and is part of the project which is supported by the National Natural Science Foundation of China, Funds for Distinguished Young Scientists of Hubei Province and the Fundamental Research Funds for the Central Universities.