文章编号:1000-0887(2013)11-1209-07

基于 SIFT 特征向量的图像检索优化*

肖曼玉, 卢江虎, 谢公南

(西北工业大学 理学院 应用数学系, 西安 710072)

(我刊编委谢公南来稿)

摘要: 基于 SIFT(scale-invariant feature transform,尺度不变特征转换)向量的图像检索在精度和实时性方面都与使用者的心理预期有较大的偏差,该文在建树(build vocabulary tree)、检索、以及匹配度计算方面做了一些改进,在满足实时性的要求下,提高了检索精度;在建树过程中,重新定义了 SIFT 特征向量聚类机制,将分类和 K 均值聚类法结合起来代替传统的 K 均值聚类法;在进行图像检索时,直接利用已有欧氏距离信息,减少向量之间距离的计算,对 SIFT 向量统一化处理;最后通过改进单位化处理方法,克服 SIFT 大数据造成的误差.数值结果表明,改进后 vocabulary tree 的节点有更强的差异性, 克服了将训练集按数量均分而不是按距离均分和直接决定树的层数的缺陷;使得检索时间很好地满足了实时性的要求;改进的单位化方法消除了 SIFT 大数据的误差,从而极大地提高了检索精度.

关 键 词: SIFT; 图像检索; 倒排文件; K 均值聚类

中图分类号: V19; O343.6 **文献标志码:** A

DOI: 10.3879/j.issn.1000-0887.2013.11.010

引 言

SIFT 特征是一种计算机视觉的算法,用来侦测与描述影像中的局部性特征,它在空间尺度中寻找极值点,并提取出其位置、尺度、旋转不变量.该算法是由 Lowe[1] 在 1999 年提出来的关于将图片的特征提取量化成若干组 128 维向量的方法,其对旋转、尺度缩放、亮度变化保持不变性,对视角变化、仿射变换、噪声也保持一定程度的稳定性,并且具有较强的鲁棒性.在 SIFT 特征向量概念提出来之后,主流图像检索技术都是将图片量化为 SIFT 特征向量或在某些应用中使用该向量的改进版本,然后都是用类似于文本检索的方法基于该向量进行检索,最后由该向量计算被检索图片和图片库的匹配度[2-5].在实际应用中,像 Google 和百度都有上亿张图片,针对海量图片的情况,大部分技术都是将由这些图片提取的特征向量构造基于树结构的数据库,利用树结构的优势来满足在保证精度的条件下进行实时检索的要求[6-8].流程图如图 1 所示.

这主要分为 3 个阶段:

1) 利用训练集将树模型构造出来

* **收稿日期:** 2013-06-08; **修订日期:** 2013-10-14

基金项目: 国家自然科学基金青年科学基金(11302173)

作者简介: 肖曼玉 (1980—),女,湖北人,副教授,博士(通讯作者. E-mail: manyuxiao@nwpu.edu.cn).

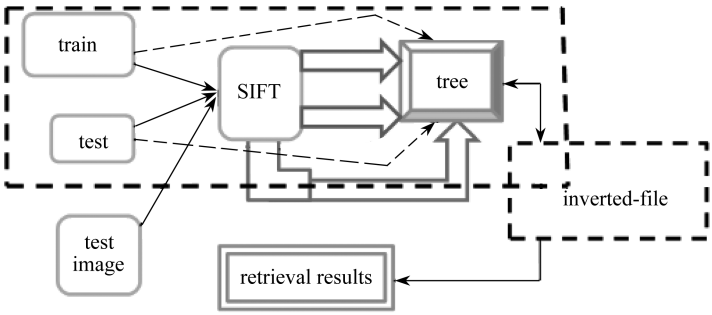

图 1 图像检索流程图

Fig.1 Image retrieval procedure

在建树过程中,先对每个节点指定 K 个分支和指定树的高度 M 层.首先对由训练集提取的特征向量进行聚类得到 K 个聚类中心作为根节点的相应分支节点的中心向量,并分配相应的 SIFT 特征向量,然后以该分支节点为新的根节点,递归进行此操作,直至达到最大层数 M.

2) 将图片库中的图片以 SIFT 向量的形式保存在树的叶子节点中,形成数据库

一棵初步的树建好后,要构造图片数据库,主要是为 vocabulary tree 的叶子节点建立 inverted-file.提取测试集图片库中的 SIFT 向量,对每个特征向量都从树的根节点开始搜寻与其距离最短的子节点,然后以该节点为根节点搜寻新的子节点,直至搜寻到叶子节点,这样每个向量都在树中有唯一的一条基于某种距离的到达叶子节点的路径.在实际应用中,考虑存储空间的限制,保留该向量的 id(indentity)号,即其所属图片的编号.在该叶子节点中有个记录,表示该叶子节点存有该图片的部分信息.对所有的向量都进行相同的操作后,就建立了图片的数据库,对每张图片只存储其向量在叶子节点出现的次数,这是该方法能够应用于海量图片库的根本原因.

3) 进行图像检索,计算图片之间的匹配度

进行检索时,首先要提取被检索图片的 SIFT 特征向量,然后对每个向量也是类似于构造数据库在 vocabulary tree 中寻找一条最短路径.到达叶子节点后取出该叶子节点存储的图片向量信息,表示被检索图片与图片库中的某张图片有若干个相同的向量.根据这些信息为被检索图片和每张图片都构造一个匹配向量,通过匹配向量的匹配度比较来决定检索的结果.

以上研究内容有诸多限制和缺陷:需要根据训练集人工确定 vocabulary tree 高度 M;在建树和检索过程中,每个向量都要与某一节点的 K 个分支节点的中心向量作距离,由于 SIFT 向量为 128 维,故该操作占用了检索过程的大部分时间,不利于实时检索;每个向量到达叶子节点所检索到的 inverted-file 中的向量部分与原向量不匹配,却直接被拿来使用,势必会降低检索精度.考虑到这些问题,本文提出了新的建树模型和检索机制,极大地提高了检索精度并很好地满足了实时精度的要求.

本文的 vocabulary tree 模型结构安排如下:

1) 构造树节点的聚类方法介绍;

2) 按照上述聚类方法利用训练集建树;

3) vocabulary tree 树建好后,提取测试集的 SIFT 特征向量为该树构造倒排文件;

4) 提取一张图片的 SIFT 特征向量然后进行检索;

5) 最后再对结果进行讨论.

注 本文在向量之间作距离时,以欧氏距离为标准(结果证明该距离有最好的检索结果).

1 Vocabulary tree 模型

1.1 聚类方法

传统聚类方法[5,9]的精度之所以一直不高是因为在建立倒排文件和检索过程中,由于“维数灾难”[9]的存在,会导致相似的向量没有分配到一个同一个叶子节点中并可能造成寻找到一条错误的路径.为了减少这些匹配误差,本文采用系统聚类和分类结合,根据类的半径以及聚类中心 $\boldsymbol{R}$ 与被检索向量 $\boldsymbol{P}$ 间的距离来自动进行聚类,在阈值的限制下,自动确定聚类数目.原理如下:

$$\| \boldsymbol{P} - \boldsymbol{Q} \| \leqslant \| \boldsymbol{P} - \boldsymbol{R} \| + \| \boldsymbol{R} - \boldsymbol{Q} \|, \tag{1}$$

其中, $\boldsymbol{Q}$ 为倒排文件下任意一个向量.

向量间的匹配是通过两者之间的距离来衡量的,而在倒排文件建好后,并未保存图片库的 SIFT 特征向量,因此无法直接精确查找与被检索向量相似的项,只能通过上式来确保相似向量一定落在某个范围内,然后在所有的被检索向量查找完之后,存在大量 id 号相同的必然是正确的图片,据此原理来确定类的半径和分支节点的数目.

1.2 建树

构造 vocabulary tree,其实就是利用训练集的所有特征向量,在分类和系统聚类法体系下,寻找一组强大的多维向量坐标.

对所有的特征向量,首先求出平均值向量,如下:

$$\boldsymbol{S} = \frac{1}{M} \sum_{i=1}^{M} \boldsymbol{V}_i . \tag{2}$$

将该向量作为根节点标示符,然后计算所有向量与该中心向量的距离:

$$d_i = \| \boldsymbol{S} - \boldsymbol{V}_i \|, \quad i = 1,2,\cdots,M . \tag{3}$$

对这些距离进行排序,根据式(1)按照距离在阈值的限制下自动对所有向量分类.设聚成了 K 个类,在根节点中保存其分支节点的个数,即 K 个.另外保存每个类与根节点向量的最大距离(在检索的时候可以极大地提高检索效率).这时候依次检查根节点与分支节点之间的距离与该分支节点所代表类的半径之和看其是否满足小于某个阈值.若小于某个阈值,则该分支节点停止对其所代表的类继续进行聚类,转而作为叶子节点,否则继续对其迭代进行聚类,直至满足要求.

这样在摆脱了层数 M 和分支节点数 K 的限制后,由训练集的 SIFT 特征向量就自动形成了一棵高度和分支数目任意的树,这棵树最大限度地避免了“维数灾难”对检索精度的影响.

1.3 倒排文件的建立

在找到一组强大的有层次的多维向量坐标后,就可以利用测试集的所有特征向量为叶子节点构造倒排文件了.

由树的节点构成的向量坐标相当于路标一样,测试集的一个 SIFT 向量到达某一节点后,通过与其计算欧氏距离就可以知道接下来该选择哪个分支节点,直至到达叶子节点.这样每个向量都存在唯一的一条由节点指引的从根节点到叶子节点的路径.该路径指示了该向量如何能“走最少的路”到达叶子节点.对走到某个叶子节点的向量,首先判定与叶子节点中心向量的距离,若大于某个阈值,就舍弃,不将其作为考虑范围,阈值的设定需根据训练集而定.

为了将该方案用于海量图片模型,到达叶子节点的向量本身不能保存,这样会占据太大的空间,代替的是其所属的 id 编号.这样就会在叶子节点中保存一个维数同测试集图片数目相同的数组,每一维代表对应图片的向量的终点为该叶子节点的数目,如图 2 所示.

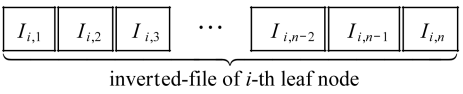

图 2 倒排文件

Fig.2 Inverted-file

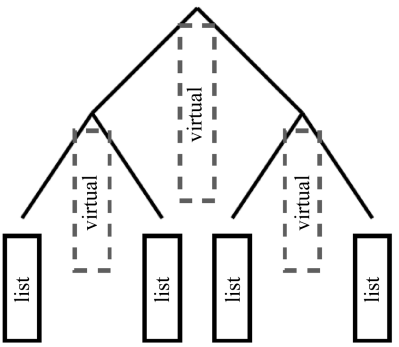

图 3 Vocabulary tree 下的倒排文件[2]

Fig.3 Inverted-file based on the vocabulary tree[2]

图 2 中，$I_{i,j}$ 代表第 i 个叶子节点中存在的图片编号为 j 的特征向量的数目.

最终树的叶子节点的倒排文件宏观图，如图 3 所示.

为了进一步提高检索精度，本文将文本检索中的 TF-IDF（term frequency-inverse document frequency）机制引入图像检索中.由于在将图片量化成若干个 SIFT 特征向量后，图像检索的实质其实就是在海量的向量中寻找与某个向量类似的若干个向量，即用向量来检索向量，这属于文本检索的范畴，引入 TF-IDF 可以明显地改善检索效果.

TF-IDF 机制是用来给叶子节点赋予权值.作如下考虑：若某个叶子节点中存在图片库中所有图片的向量，显然这个叶子节点不能很好地作为图片之间特征分辨的对象，因为每个都跟这个叶子节点有关系，检索到这个叶子节点后通过该节点得不到更多的图像之间匹配的信息.考虑另一种极端情况：某个叶子节点中只存在某一张图片或几张相似图片的特征向量，那当某个向量经过最短路径到达该叶子节点，就可以说明该向量所在图片与该叶子节点中倒排文件所代表的的图片有一定的匹配程度，即该叶子节点可以很好地代表某张图片或某几张相似的图片，且在正常情况下只有相似图片才会存在经最短路径到达此节点的向量，该叶子节点就应该可以赋予较大的权值，以便体现该叶子节点的作用.

考虑到图像检索的特殊性，以及针对倒排文件检索的具体要求，利用原本的 TF-IDF 机制显然不能满足图像检索：

$$w = T \times I, \tag{4}$$

其中，w 为叶子节点的权值，T 为某个单词在一篇文章中出现的频率 TF，I 为某个单词在所有文章中出现的情况 IDF.因为找不到合适的参数来应用 TF，因此本文采用

$$w = I \tag{5}$$

来实现叶子节点的权值赋予.

根据叶子节点存在向量所属图片 id 越多，权值越小的原则，令

$$I_i = \lg \frac{N}{\sum_{j=1}^{m} f(I_{i,j})}, \tag{6}$$

其中，N 为测试集图片库中图片的数量，m 为倒排文件向量存在数组的维数，$I_{i,j}$ 表示第 i 个叶子节点中编号为 j 的图片的 SIFT 向量的数目，其中的函数 $f(I_{i,j})$ 为

$$f(I_{i,j}) = \begin{cases} 1, & I_{i,j} > 0, \\ 0, & I_{i,j} = 0, \end{cases} \tag{7}$$

这可以理解为从倒排文件数组中寻找该叶子节点下图片出现的痕迹.由式(5)、(6)、(7)就可得到叶子节点的权值.这样一个叶子节点完整的倒排文件包括该叶子节点的权值和向量存在数组.

1.4 图像检索

对于给定的一张图片,在其量化为 SIFT 特征向量之后,就需要在 vocabulary tree 中寻找最相似的向量来确定图片库图片与其之间的匹配度.

对该图片所有的特征向量,将其与根节点作距离,由于在节点中还保留有各分支节点与其的距离信息,因此可以根据该信息确定最短路径的下一步,即某个分支.这是本模型比前人方法速度快的根本原因.设 vocabulary tree 的高度和平均分支数目分别为 M 和 K,两个向量做欧氏距离所需时间为 t, 则前人一个向量检索大约需时

$$T = M \times (K - 1) \times t. \tag{8}$$

而本模型所需时间仅为 $T = Mt$,即本模型检索所需时间为前人的 $1/(K - 1)$.省出的时间在下面可以继续对检索精度做可能的提高.

在所有的向量寻找到一条由根节点到达叶子节点的最短路径后,取出倒排文件的信息,分别为检索图片和测试集图片构造一个向量,图片向量形式如下:

$$\boldsymbol{S} = (w_1 \times n_1,\ w_2 \times n_2,\ \cdots,\ w_n \times n_n), \tag{9}$$

表示图片向量终点是各个叶子节点的数目与该节点权值的乘积.第 j 张测试集图片向量为

$$\boldsymbol{D}_j = (w_1 \times I_{1,j},\ w_1 \times I_{2,j},\ \cdots,\ w_1 \times I_{n,j}). \tag{10}$$

在前人的方法中,是在将这些向量经过单一化后,计算两者的匹配度:

$$h(\boldsymbol{S},\boldsymbol{D}_j) = \left\| \frac{\boldsymbol{S}}{\| \boldsymbol{S} \|} - \frac{\boldsymbol{D}_j}{\| \boldsymbol{D}_j \|} \right\|. \tag{11}$$

而根据实际结果发现,由于图像包含的信息量不同,提取的 SIFT 特征向量数目也不同,含信息量比较大的图片会多次被检索到,对精度产生了很大的影响,因此本文采用如下的匹配度计算方式:

$$h(\boldsymbol{S},\boldsymbol{D}_j) = \left\| \frac{\boldsymbol{S}}{N} - \frac{\boldsymbol{D}_j}{N_j} \right\|, \tag{12}$$

其中, N 表示被检索图片的 SIFT 特征向量个数.

根据匹配度便可以决定检索到的图片的排列顺序.

2 数值结果与讨论

Ukbench 数据集包含了大约 2 550 组不同的图像.每一组图像由 4 张图片组成,这 4 张图片是在不同的光照、角度、分辨率等及多种背景噪音存在的条件下拍摄同一个事物形成的.可认为是相似图片.根据使用者的心里预期,只要与被检索图片相似的 4 张图片的匹配度比较靠前就可以接受,本文的检索精度定义为:对被检索的一张图片,其检索精度为

$$a_i = \frac{n_i}{4}, \qquad n_i = 0,1,2,3,4, \tag{13}$$

表示在检索到的前 10 张图片中,有 n_i 张是相似图片.

表 1 丢弃 SIFT 后的精度

Table 1 The accuracy after discarding some SIFTs

discard	quantity			
	100	500	1 000	10 000
no	90.6%	82.3%	76.2%	53.9%
yes	94.3%	86.3%	83.25%	70.2%

本文拿 Ukbench 中不同数量的图片作为测试集进行实验,其中图 4 为分别测试 L_1-norm,

L_2-norm 以及本文所提出的归一化标准 unit-norm 对结果的影响.

这里,图 4 的横轴表示图片库中的图片数,纵轴表示检索精度,从图 4 可以看出,采用 unit-norm 计算方式会获得最好的计算检索精度.

表 1 为在建立倒排文件时,实施排除(即将与叶子节点距离大的向量丢弃)后在 unit-norm 标准下,结果如表 1 所示.

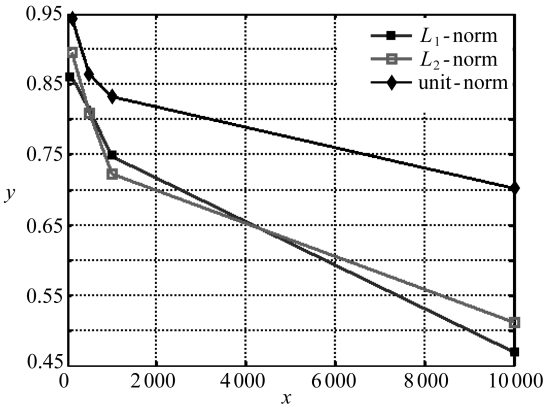

图 4 不同标准的计算结果

Fig.4 Results under different norms

从上述结果中可以发现:SIFT 向量本身存有一定的误差,随着图片数量的增加,这种误差将会被掩盖,同时由于 SIFT 数量的大幅度增加(1 000 张图片的时候有 1 133 199 个 SIFT 向量),SIFT 向量会产生误判的现象,即本应分到这一个分支下,却因为聚类的误差,进入另一个分支下面,而这种误差,本文尽可能在缩小.

分析检索结果,会发现检索结果之所以不是很显著是因为每张图片 SIFT 的多少会影响检索精度.本文用单一化的方法成功消除了 SIFT 量大的情况,但是同时又暴露了少量 SIFT 的弱点:用匹配向量除以较小的 SIFT 值,会增大含有较少 SIFT 向量的图片被命中的机会.截至目前,本文并未能给出合理的解决方案,但是这种误差要远远小于 SIFT 数量大的图片造成的误差.

在海量 SIFT 向量的条件下,一方面是 SIFT 本身的误差,另一方面是 SIFT 量的误差,所以可以适当的丢弃一些 SIFT 向量,正如本文在建立倒排文件时那样,删掉一些距离较大的 SIFT,这样对结果有明显地改进作用.

3 结论与改进

通过对图像检索的改进,本文得出以下结论:

1) 放松对分支数目 K 和树的高度 M 的限制并且让系统根据训练集自动决定分类数目,会对图像检索有明显地改进;

2) 可以根据距离适当地丢弃一些有偏差的 SIFT 特征向量;

3) 采用新的匹配度实现机制,消除 SIFT 大数据的误差;

在本文中,并未完全摆脱 vocabulary tree 分支数目的干扰,需要人工确定.因此可以尝试改进自动确定分支来提高精度,本文建议引入系统聚类法来自动确定聚类数目,即分支数目.

致谢 该工作特别感谢西北工业大学基础研究基金的资助(NPU-FFR-JC20120241)和西北工业大学高性能中心对算例测试计算的支持.

参考文献(References):

[1] Lowe D G. Object recognition from local scale-invariant features[C]//*International Conference on Computers Vision*. Corfu, 1999: 1150-1157.

[2] Nistér D, Stewénius H. Scalable recognition with a vocabulary tree[C]//*IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (*CVPR*). Vol **2**. New York, USA, 2006: 2161-2168.

[3] Indyk P, Motwani R. Approximate nearest neighbors: towards removing the curse of dimensionality[C]//30*th Annual ACM Symposium on Theory of Computing*. New York, USA, 1998: 604-613.

[4] Mikolajczyk K, Schmid C. A performance evaluation of local descriptors[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (*PAMI*), 2005, **27**(10): 1615-1630.

[5] Obdržálek Š, Matas J. Sub-linear indexing for large scale object recognition[C]//*British Machine Vision Conference*(*BMVC*). Oxford, UK, 2005.

[6] 王群伟. 基于 SIFT 特征点提取的图像检索研究[D]. 硕士学位论文. 武汉: 华中科技大学, 2010. (WANG Qun-wei. Image retrieval based on SIFT feature point detection[D]. Master Thesis. Wuhan: Huazhong University of Science and Technology, 2010.(in Chinese))

[7] Sivic J, Zisserman A. Video google: a text retrieval approach to object matching in videos [C]//9*th IEEE International Conference on Computer Vision* (*ICCV*). Vol **2**. Nice, France, 2003: 1470-1477.

[8] Berg A C, Berg T L, Malik J. Shape matching and object recognition using low distortion correspondence[C]//*IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (*CVPR*). Vol **1**. San Diego, USA, 2005: 26-33.

[9] 陈作平, 叶正麟, 郑红婵, 赵红星. 基于 K 均值聚类的快速分形编码方法[J]. 中国图象图形学报, 2007, **12**(4): 586-591.(CHEN Zuo-ping, YE Zheng-lin, ZHENG Hong-chan, ZHAO Hong-xing. Fast fractal coding technique based on *K*-mean clustering[J]. *Journal of Image and Graphics*, 2007, **12**(4): 586-591.(in Chinese))

[10] Beis J S, Lowe D G. Shape indexing using approximate nearest-neighbor search in high-dimensional spaces[C]//*IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (*CVPR*). San Juan, Puerto Rico, 1997:1000-1006.

Optimization of SIFT-Based Image Retrieval

XIAO Man-yu, LU Jiang-hu, XIE Gong-nan

(*Department of Applied Mathematics*, *School of Science*, *Northwestern Polytechnical University*, *Xi' an* 710072, *P.R.China*)

(Contributed by XIE Gong-nan, M. AMM Editorial Board)

Abstract: In order to deal with the great discrepancy between the expectations of users and the real performance in image retrieval, some improvement on building tree, retrieval and matching methods were made with great success both in accuracy and in efficiency. More precisely, a new clustering strategy was firstly redefined during the building of vocabulary tree, which combined the classification and the conventional *K*-means method. Then a new matching method to eliminate the error caused by large-scale SIFT was introduced. What was more, a new unit mechanism was adopted to shorten the cost of indexing time. Finally, the numerical results show that an excellent performance is obtained after these improvements. A vocabulary tree with more distinguished nodes is achieved, of which the height is defined automatically and the index accuracy is enhanced greatly. Furthermore, a faster indexing procedure is realized, of which the indexing time is much less than 1 s.

Key words: SIFT; image retrieval; inverted-file; *K*-means clustering

Foundation item: The National Science Fund for Young Scholars of China(11302173)