Journal of Computer Applications
计算机应用,2014,34(3):695 -699
ISSN 1001-9081
CODEN JYIIDU
2014-03-10
http://www.joca.cn

文章编号:1001-9081(2014)03-0695-05
doi:10.11772/j.issn.1001-9081.2014.03.0695

基于 Hadoop 云计算平台的大规模图像检索方案

朱为盛[1,2*],王 鹏[3]

(1.中国科学院 成都计算机应用研究所,成都 610041; 2.中国科学院大学,北京 100049;
3.成都信息工程学院 并行计算实验室,成都 610225)

(* 通信作者电子邮箱 zhu-weisheng@163.com)

摘 要: 针对传统图像检索方法在处理海量图像数据时面临困扰的问题,提出了一种基于传统视觉词袋(BoVW)模型和 MapReduce 计算模型的大规模图像检索(MR-BoVW)方案。该方案充分利用了 Hadoop 云计算平台海量存储能力和强大的并行计算能力。为了更好地处理图像数据,首先引入一种改进的 Hadoop 图像数据处理方法,在此基础上分特征向量生成、特征聚类、图片的向量表示与倒排索引构建三个阶段 MapReduce 化。多组实验表明,MR-BoVW 方案具有优良的加速比、扩展率以及数据伸缩率,效率均大于 0.62,扩展率以及数据伸缩率曲线平缓,适于大规模图像检索。

关键词: 云计算; Hadoop; MapReduce; 图像检索; 视觉词袋模型

中图分类号: TP391.3; TP311.1 **文献标志码:** A

Large-scale image retrieval solution based on Hadoop cloud computing platform

ZHU Weisheng[1,2*], WANG Peng[3]

(1. *Chengdu Institute of Computer Application, Chinese Academy of Sciences, Chengdu Sichuan* 610041, *China*;
2. *University of Chinese Academy of Sciences, Beijing* 100049, *China*;
3. *Parallel Computing Laboratory, Chengdu University of Information Technology, Chengdu Sichuan* 610225, *China*)

Abstract: Concerning that the traditional image retrieval methods are confronted with massive image data processing problems, a new solution for large-scale image retrieval, named MR-BoVW, was proposed, which was based on the traditional Bag of Visual Words (BVW) approach and MapReduce model to take advantage of the massive storage capacity and powerful parallel computing ability of Hadoop. To handle image data well, firstly an improved method for Hadoop image processing was introduced, and then, the MapReduce layout was divided into three stages: feature vector generation, feature clustering, image representation and inverted index construction. The experimental results demonstrate that the MR-BoVW solution shows good performance on speedup, scaleup, and sizeup. In fact, the efficiency results are all greater than 0.62, and the curve of scaleup and sizeup is gentle. Thus it is suitable for large-scale image retrieval.

Key words: cloud computing; Hadoop; MapReduce; image retrieval; Bag of Visual Words (BoVW) model

0 引言

随着互联网的高速发展,数据日益呈现出海量化、多媒体化的趋势。如何对海量多媒体信息进行快速的检索已经成为亟待解决的重要问题。在图像检索领域,视觉词袋(Bag of Visual Words, BoVW)模型因其模型简单计算复杂度较低却具有良好的性能,并与文本分析中的词袋(Bag of Words, BoW)模型有很大的相似性,文本检索中很多相对成熟的技术如倒排索引等都可以用在图像检索领域,而受到很多学者的关注。但是传统的串行处理模式无法完成大规模的图像数据处理,探索新的处理模式已经成为该领域的一个研究热点。

近年来云计算发展迅速,其发展过程一直与大规模数据处理密切相关,因此利用云计算平台高效地处理大规模图像数据的检索问题是一个非常有潜力的方向。Hadoop 云计算平台以其优秀的大规模数据处理能力、良好的可扩展性和可靠性以及低成本的优势受到产业界的推崇,涌现出了大量商业应用。但是 Hadoop 最初是针对大规模文本数据处理设计的,内部数据类型有限,不能直接处理多媒体数据。针对这一问题,已有学者进行了一些研究,但相对文本处理来说较少,并且存在一定的局限性。

本文在对 Hadoop 图像数据处理方法和视觉词袋模型两个方面进行深入分析之后,提出了一种基于 Hadoop 的大规模图像检索方案(MapReduce-BoVW, MR-BoVW)。首先,针对现有 Hadoop 图像数据处理方法的不足引入了一种改进的方法。为了避免小文件低效率问题,参考序列文件方法合并文件的思想将大量小图像文件存入一个大的图像库文件中,但是存储的方式不再是序列化的键值对或者 Float 数组,而是原始图片的所有信息。为了方便读写多种类型的图像数据并充分利用现有的 Java 图像处理类库,利用 ImageIO、BufferedImage 等来实现图像库文件的读写,并引入一个索引

收稿日期:2013-08-26;修回日期:2013-10-20。 基金项目:国家自然科学基金资助项目(60702075);四川省青年科学基金前期资助项目(09ZQ026-068);四川省教育厅自然科学重点项目(07ZA014)。

作者简介:朱为盛(1986-),男,浙江温州人,硕士研究生,主要研究方向:云计算、多媒体内容分析; 王鹏(1975-),男,四川乐山人,教授,博士生导师,博士,CCF 高级会员,主要研究方向:云计算、并行计算。

文件来实现随机读取。其次,在综合考虑性能和可扩展性等因素之后提出了一种基于视觉词袋模型和 MapReduce 计算模型的大规模图像检索方案 MR-BoVW,从而解决传统 BoVW 模型面对大规模图像处理时的困扰。多组实验表明 MR-BoVW 方案具有优良加速比、扩展率以及数据伸缩率,可以有效利用 Hadoop 云计算平台海量存储能力和强大的并行计算能力,适于大规模图像检索。

1 相关工作

相关工作主要来自两个方面: Hadoop 图像数据处理方法和视觉词袋模型。

Hadoop 图像数据处理方法方面,文献[1]通过预处理将图像转换成序列化的二进制文件从而利用 Hadoop 实现对天文学图像的分析;但是序列化文件只能顺序读取,无法实现图片的随机读取,对复杂的应用有很大的局限性。文献[2]利用 Hadoop 对遥感图像进行了分析,将图片存入 HDFS(Hadoop Distributed File System)并通过自定义图像接口完成批量图像的读写。文献[3]通过自定义图像接口读写整张图片的方式实现基于 Hadoop 的图像分类;但是这种方法并没有考虑到小文件低效率问题[4-5],即大量小文件的元数据消耗 NameNode 内存,并且处理时需启动大量 Map 任务,而每个 Map 仅处理少量的数据,从而导致低效率甚至资源浪费,因此这种方法处理遥感图像尚可,但并不适合处理互联网图片。文献[6]提出了一种新的方法,通过转化为 Float 数组的方式将大量图片信息存储在一个文件里,并附有一个索引文件,从而解决了小文件低效率问题并且通过索引支持随机读取;但这种方法不能存储原始图片的所有信息,并且数组方式与具体图片类型相关,编解码复杂,目前只支持 RGB 颜色空间和 JPEG、PNG、PPM 格式的图片,应用范围受限。因此,有必要对现有方法做进一步的改进。

近几年来,视觉词袋模型 BoVW 被广泛应用于图像内容分析。该模型通过与文本分析中的词袋模型 BoW 类比而得。与 BoW 类似,BoVW 假设视觉单词之间相互独立,而不考虑它们的空间结构,因此一张图片被表示为视觉单词的集合,通常由视觉单词在视觉词汇表中的分布来描述。典型的 BoVW 模型主要包括以下几个部分:局部特征检测与描述、视觉词汇表构建、图片表示、任务模型选取。根据各个部分具体实现方式的不同,BoVW 有很多变种:

1)局部特征区域检测。文献[7-8]使用了基于兴趣点检测的方法;文献[9]使用了规则网格、随机采样、基于兴趣点等多种方法。

2)局部特征描述。文献[7-9]使用了 SIFT(Scale-Invariant Feature Transform)[10]特征描述符;文献[11-12]同时使用了 SIFT 和 SURF(Speeded Up Robust Features)[13]特征描述符。

3)视觉词汇表构建。文献[7-9]使用了 *K*means 聚类方法;文献[14]使用了基于树的方法。

4)图片表示。文献[8]使用了文本检索中的词项频率-逆文档频率(Term Frequency-Inverse Document Frequency, TF-IDF)权重向量表示法;文献[9]使用视觉单词出现次数的直方图描述。

5)任务模型选取。在图片分类任务中文献[7]使用了朴素贝叶斯分类器和支持向量机;在视频对象匹配任务中文献[8]使用了文本检索的方法;在自然场景分类任务中文献[9]使用了 LDA(Linear Discriminant Analysis)模型。

BoVW 模型简单计算复杂度较低却具有良好的性能,但其传统的串行处理模式仍面临海量图像数据处理的困扰,仍需探索新的处理模式。

2 MR-BoVW 方案的设计

出于精确率与计算复杂度以及可扩展性的折中考虑,本文方案采用 SURF 检测并描述局部特征,采用 *K*-means 聚类构建视觉词汇表,采用 TF-IDF 权重向量表示图片,并构建倒排索引来实现高效的检索。

理由如下:文献[11-12]对比了 SIFT 和 SURF,指出 SURF 在保持相近精确率的同时大幅提高了计算效率。*K*-means聚类具有很好的局部性可以方便地并行化,且文献[7-9]的实验结果表明 *K*-means 聚类构建视觉词汇表的方法虽然简单却有很好的性能。BoVW 本身是通过与文本分析中的词袋模型类比而来,与其有很大相似性,而 TF-IDF 权重向量表示法和倒排索引已经在文本检索领域得到成功应用。

MR-BoVW 方案整体设计流程如图 1 所示,方案分为两部分:虚线左边的离线处理部分和虚线右边的在线检索部分。由于 Hadoop 被设计为适于大规模离线数据处理的工具,并不保证在线处理的实时性,因此在线检索部分仍按传统方法进行,离线处理部分基于 Hadoop 设计。

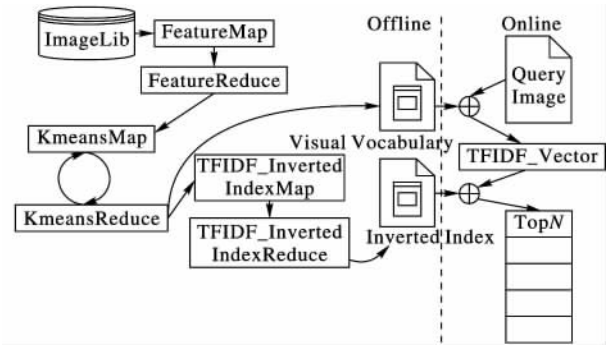

图1 MR-BoVW 方案整体设计

为了更好地处理图像数据,首先引入一种改进的 Hadoop 图像数据处理方法,在此基础上将这部分分为特征向量生成、特征聚类、图片的向量表示与倒排索引构建三个阶段实现。

2.1 改进的 Hadoop 图像数据处理方法

Hadoop 最初是针对大规模文本分析设计的,对图像数据处理支持不足。为了更好地处理图像数据并避免小文件低效率问题,参考序列文件方法合并文件的思想将大量小图像存入一个大的图像库文件中,但是存储的方式不再是序列化的键值对或者 Float 数组,而是原始图片的所有信息。这样不仅有效减小了对 NameNode 的内存需求,也降低了任务管理的开销,可以明显改善处理效率,同时保存的原始图片信息有利于应对复杂的图像处理需求。对图像库的读写需要改写相应的接口,参考已有方法实现的功能自定义了 ImageInpuFormat、

ImageRecordReader、ImageOutputformat、ImageRecordWriter、ImageWritable 几个类。为了方便读写多种类型的图像并充分复用现存的 Java 图像处理类库,上述类均采用 ImageIO、BufferedImage 等来实现。最后,为了实现对图像数据的随机读取,需要一个索引文件,其中保存了图像库文件中所有图片数据的偏移量。通过偏移量可以方便地访问图像库文件中的任意图片。

2.2　特征向量生成

得益于 Hessian 矩阵的行列式同时表达了位置和尺度的信息,SURF 特征在保持尺度、旋转、照明变化无关特性的同时,使计算过程变得更加高效。其计算过程如下: 首先计算图像中每个像素 $X = (x,y)$ 在尺度 σ 的 Hessian 矩阵:

$$\boldsymbol{H}(X,\sigma) = \begin{bmatrix} L_{xx}(X,\sigma) & L_{xy}(X,\sigma) \\ L_{xy}(X,\sigma) & L_{yy}(X,\sigma) \end{bmatrix}$$

其中: $L_{xx}(X,\sigma)$ 是高斯二阶导数 $\frac{\partial^2}{\partial x^2}g(\sigma)$ 和图像中像素 X 的卷积; $L_{xy}(X,\sigma)$ 和 $L_{yy}(X,\sigma)$ 类似。该矩阵由二阶导数构成,可用不同尺度 σ 下的近似高斯核来计算,因此 Hessian 值变成了 3 个变量的函数: $\boldsymbol{H}(x,y,\sigma)$,然后求其同时在空间域和尺度域上达到局部极大值时的位置和对应的尺度。其特征描述子基于 Haar 小波计算: 对于每一个特征点,计算其在半径为 6σ 的圆形范围内的 Haar 小波在 x 和 y 方向上的响应(记为 dx 和 dy),对覆盖 60° 的窗口内的响应求和,旋转窗口计算得到的最长向量的方向即为主要方向。接下来按该方向构造大小为 20σ 的方形区域,并将其分割成 4×4 的小区域,对每个子区域的 25 个采样点计算 dx 响应和 dy 响应并分别进行求和,对每个子区域提取 4 个描述子的值: $\left[\sum dx \ \sum dy \ \sum |dx| \ \sum |dy|\right]$,共有 16 个子区域就得到一个 64 维的向量,最后将其单位化。

由于图片之间的特征检测与描述相互独立,因此只需简单地将以上计算过程封装进 Map 函数里,并且这一阶段只需要 Map 部分即可完成。以下为这一阶段的 MapReduce 设计:

1) Map。输入为形如〈*image_id*,*image_data*〉的图片。Map 函数对输入的每一张图片执行 SURF 算法提取特征向量,并统计该图片中的特征数 *feature_num*。这个特征数用于后面词项频率的归一化。其输出形式为〈(*image_id*: *feature_num*), *image_feature*〉。

2) Reduce。Reduce 函数的作用类似于恒等式,它仅将每个键值对传递到输出部分。

这一阶段结束后,得到一个每张图片所含特征向量的描述文件。

2.3　特征聚类

K-means 算法的目标是最小化所有向量到其簇中心的距离平方和,即残差平方和(Residual Sum of Squares, RSS):

$$RSS = \sum_{k=1}^{K} \sum_{x \in w_k} |x - \boldsymbol{\mu}(w_k)|^2$$

其计算过程为: 首先随机选取 K 个样本作为初始簇中心,对剩下的每个样本根据其到簇中心的距离分配到各个簇,重新计算 K 个新簇的簇中心; 再将每个样本根据其到簇中心的距离分配到各个新簇。如此迭代直到目标函数收敛或迭代到一个固定步数。

这个迭代过程可以通过重复调用 MapReduce 任务的方式来实现,每启动一次 MapReduce 计算对应一次迭代。以下为这一阶段的 MapReduce 设计:

1) Map。输入为形如〈*line_num*, ((*image_id*: *feature_num*), *image_feature*)〉的待分配样本和上一次迭代(或初始)的簇中心。这里的(*image_id*: *feature_num*)并不参与计算只用于标识特征所属的图片以及图片包含的特征数。Map 函数对输入的每个样本计算出距离最近的簇中心并标记新的簇类别。其输出形式为〈*cluster_id*, ((*image_id*: *feature_num*), *image_feature*)〉。

2) Reduce。输入为形如〈*cluster_id*, [(*image_id*: *feature_num*), *image_feature*)]〉的样本列表,这里的(*image_id*: *feature_num*)同样不参与计算。所有 *cluster_id* 相同的样本都输送给同一个 Reduce 任务。Reduce 函数累加 *cluster_id* 相同的样本个数与各样本向量分量的和,求各分量的均值得到新的簇中心。其输出形式为〈*cluster_id*, *cluster_mean*〉。

这一阶段结束后得到一个每张图片所含特征及其特征所属视觉单词的描述文件和一个视觉词汇表描述文件,其中 *cluster_id* 为视觉单词编号,簇中心 *cluster_mean* 代表视觉单词。

2.4　图片的向量表示与倒排索引构建

每一张图片被表示为一个向量,其中每个分量对应视觉词汇表中的视觉单词,分量的值为采用 TF-IDF 公式计算出的权重值。当某视觉单词在图片中没有出现时,其对应的分量为 0。

$$TF\text{-}IDF_{t,d} = TF_{t,d} * IDF_{t,d} = \frac{n_{t,d}}{n_d} * \log\frac{N}{DF_t}$$

其中: $n_{t,d}$ 为视觉单词 t 在图片 d 中出现的次数; n_d 为图片 d 总的特征数; $TF_{t,d}$ 即为归一化的词项频率; DF_t 为文档频率,即图片库中出现视觉单词 t 的图片数; N 为图片库中总的图片数。

图片被表示成向量后,它们之间的相似度可以采用余弦相似度计算:

$$sim(d_1,d_2) = \frac{\boldsymbol{V}(d_1) * \boldsymbol{V}(d_2)}{|\boldsymbol{V}(d_1)| \ |\boldsymbol{V}(d_2)|}$$

当向量的维度很高,图片总数很多时,这个计算过程代价很大,需要一个高效的索引结构。由于视觉词汇表中的视觉单词很少同时出现在同一张图片里,因此图片向量是稀疏的,有很多 0 分量。因此对于不含查询图片中视觉单词的图片并不需要参与计算,倒排索引可以实现这种过滤。

由于计算出的 TF-IDF 权重值是浮点数,这会造成空间的浪费,可以将 TF(Term Frequency)值(整数)存储在倒排记录中,而将 IDF(Inverse Document Frequency)值存储在倒排记录表的头部,这样一个向量只需要存储一个浮点数,可以节省存储空间。因此可以把 TF、IDF 的计算和倒排索引的构建放在一起完成。以下 MapReduce 设计实现了 TF、IDF 的计算和倒排索引的构建:

1) Map。输入为形如〈*line_num*,(*cluster_id*,(*image_id*: *feature_num*),*image_feature*))〉的图片所含特征及特征所属视觉单词的描述文件。这里只需要 *cluster_id*、*image_id* 和 *feature_num* 的信息。Map 函数对每个输入提取形如〈*cluster_id*,(*image_id*: *feature_num*)〉的键值对作为输出。

2) Reduce。输入为形如〈*cluster_id*, 【*image_id*: *feature_num*】〉记录列表。所有 *cluster_id* 相同的记录都输送给同一个 Reduce 任务。Reduce 函数对同一 *cluster_id* 记录的值列表【*image_id*: *feature_num*】分别对两个变量 *tc*、*dc* 进行累加,对每个新出现的 *image_id* 同时对 *tc*、*dc* 加 1,对每个已出现过的 *image_id* 只对 *tc* 加 1,然后用 *N* 除以 *dc* 并求对数得到 IDF,对每个 *image_id* 的 *tc* 除以对应的 *feature_num* 得到归一化的 TF。其输出形式为〈(*cluster_id*: *idf*), 【*image_id*: *tf*】〉,其中:(*cluster_id*: *idf*) 为各个视觉单词以及各自的逆文档频率,列表【*image_id*: *tf*】即为该视觉单词对应的倒排记录表。

这一阶段结束后得到一个倒排索引文件,其存储着图片库中每张图片的向量表示。在线检索时,同样对查询图片提取 SURF 特征向量,将每个特征分配到与之距离最小的视觉单词;然后计算图片的 TF-IDF 权重向量,再根据图片中出现的视觉单词从倒排索引查询倒排记录表并将其合并;最后计算查询图片向量与得到图片向量的余弦相似度,结果按相似度高低排序,或仅返回前 *N* 张图片。

3 实验与分析

3.1 实验环境和评价标准

实验环境为 5 个节点搭建的 Hadoop 平台,Hadoop 版本为 0.20.205.0,JDK 版本为 1.6。具体配置与角色安排如下:1 个配置为 Intel core 2 duo 2.2 GHz/4 GB 内存的节点作为 Namenode 和 Jobtracker,另 4 个配置为 Intel core i3 3.06 GHz/2 GB内存的节点作为 Datanode 和 Tasktracker,操作系统为 Ubuntu 12.10。

实验采用的图像数据均来自 MIRFlickr25K[15],该数据集收集了 25 000 张 Flickr 网站用户的真实图片,由于实验条件所限,从中选取了前 10 000 张图片作为实验数据集,视觉词汇表大小 *K* 取 500。

实验采用加速比(Speedup)与效率(Efficiency)、扩展率(Scaleup)、数据伸缩率(Sizeup)作为评价指标进行了分析。采用墙上时间(Wall time)即任务从提交的时刻到该任务完成所需的时间作为计量标准,这可以从 Hadoop 的 Web 界面获取。由于 Hadoop 任务调度、负载均衡以及网络通信的影响,同一任务完成所需的时间有一定的差别,而 *K*-means 特征聚类部分有随机初始化簇中心的操作,也会对任务完成时间造成一定的影响,因此对每一组实验重复执行 10 次取平均值来作为最后计量标准。

3.2 加速比与效率

加速比定义为同一任务在单个计算节点运行时间与多个计算节点运行时间之比,效率为加速比与计算节点数之比,它们衡量了 MR-BoVW 方案的整体性能。在这个实验中,保持输入图像数据集大小不变,逐渐增大节点数,对 3 组分别为 6 000,8 000,10 000 张的图像数据集进行了实验。实验结果如图 2 所示。

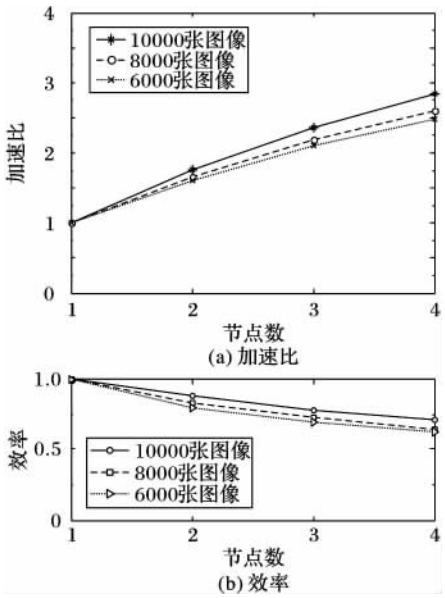

图 2 加速比与效率实验结果

理想情况下,加速比应随着节点数的增加而线性地增长,效率应保持 1 不变。但是由于任务控制与管理开销、通信开销以及负载均衡的影响,实际情况中,加速比并不能达到线性增长,效率也会低于 1。根据文献[16],当效率大于 0.5 时可认为达到很好的性能。从实验结果可以看出,3 组图像数据集的加速比都随着节点数的增加而增长,虽然没有达到线性增长,但 3 组图像数据集的效率都大于 0.5,可以认为达到了很好的性能;并且随着图像数据集的增大加速比的性能越来越好,效率越来越高,这是由于图像数据规模增大时更能发挥每个计算节点的全部计算能力。这说明了 MR-BoVW 方案整体设计以及各个阶段的 Map、Reduce 函数设计较为合理,使整个方案能够快捷地实现并且高效地运行。

3.3 扩展率

扩展率定义为单个计算节点处理较小数据集所需时间与多个计算节点处理更大数据集所需时间之比,它衡量了当问题规模不断增大时 MR-BoVW 方案能否有效地利用可扩展的节点数从而方便地扩展整体计算能力。在这个实验中,以 1 100张图片为基准,每增加 1 100 张图片同时增加一个计算节点,即同比率增大图像数据集的大小和节点数,直到增加到 4 400张图片为止。实验结果如图 3 所示。

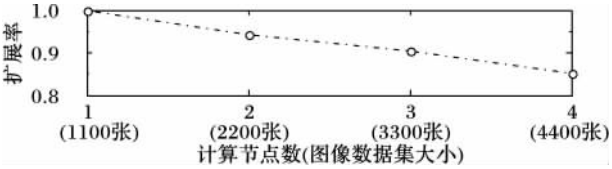

图 3 扩展率实验结果

理想情况下,合理选择问题规模和可利用的节点数可以使扩展率曲线保持水平。但由于任务启动开销以及通信开销随着任务数的增加而增大,实际情况中,扩展率曲线并不能保持水平。分析以上实验结果,由于此时不存在任务排队情况且每一节点处理同样大小的数据排除了负载不均衡的影响,因此可以看出 1 100 张图片即为 1 个计算节点处理能力的瓶

颈,当图像规模更大时将有任务需要等待。可以看到此时扩展率曲线略微向下弯曲,显示了任务管理以及通信开销的影响,但整体来说较接近水平。这说明了 MR-BoVW 方案整体设计的取舍包括 SURF 特征提取、*K*-means 聚类等过程的计算有很好的局部性,可以很好地并行化,从而能够有效地利用可扩展的节点数来实现整体计算能力的扩展。

3.4　数据伸缩率

数据伸缩率定义为处理增大后的数据集所需时间与处理原始数据集所需时间的比值,它衡量了 MR-BoVW 方案处理不同规模图像数据的能力。在这个实验中,保持计算节点数为 4 不变,从 1 000 张图片开始逐步增大数据集大小,每次增加 1 000 张直到 10 000 张为止。实验结果如图 4 所示。

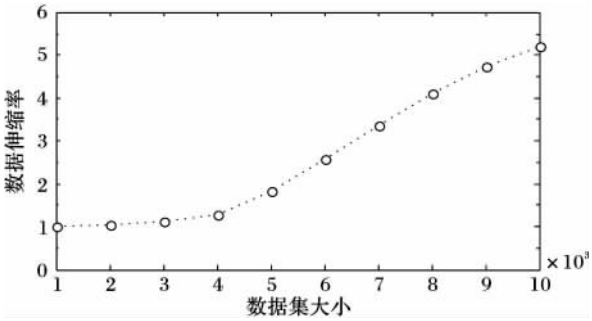

图 4　计算节点为 4 时的数据伸缩率实验结果

从图 4 可以看出,数据伸缩率曲线在 4 000 张附近存在一个拐点。这是由于 4 个计算节点处理能力的瓶颈在 4 400 张左右,当图片数少于 4 400 张时并没有发挥整个平台的全部计算能力,更多计算能力带来的性能提升远大于额外开销,此时数据伸缩率曲线较为平缓; 当图片数多于 4 400 张时启动的任务数增多,额外开销增大,且有任务需要等待同时也将对负载均衡产生影响,此时数据伸缩率曲线较为陡峭,但数据规模越大负载越均衡越能发挥所有节点的计算能力,因此总的来说较为平缓。可以看到图片从 5 000 张增加到 10 000 张时需要 2.83倍的时间,而从 1 000 张增加到 10 000 张时仅需 5.2 倍时间。这说明了 MR-BoVW 方案具有良好的数据伸缩率。

4　结语

本文对基于 Hadoop 云计算平台的大规模图像检索方案设计进行了深入研究。首先在对 Hadoop 图像数据处理方法和视觉词袋模型两个方面进行调研分析之后,对现有 Hadoop 图像数据处理方法做了改进,在此基础上提出了一种基于视觉词袋模型的可并行处理的方案 MR-BoVW。通过多组实验表明,该方案可以有效利用 Hadoop 平台海量存储能力和强大并行计算能力,适于大规模图像检索。

随着云计算和多媒体技术的飞速发展,基于 Hadoop 的大规模多媒体分析将逐渐成为一个新的研究热点。下一步的研究方向包括: 1) 充分利用多媒体数据社会化的特性,引入标签分析以改善 MR-BoVW 方案的效果; 2) 为 MR-BoVW 方案探索新的索引结构,例如局部敏感哈希(Locality-Sensitive Hashing, LSH) 并将之与倒排索引的性能进行对比。

参考文献:

[1] WILEY K, CONNOLLY A, KRUGHOFF S, *et al*. Astronomical image processing with Hadoop [C]// Proceedings of the 20th Conference on Astronomical Data Analysis Software and Systems. San Francisco: Astronomical Society of the Pacific, 2011: 93 – 96.

[2] ALMEER M H. Cloud Hadoop map reduce for remote sensing image analysis [J]. Journal of Emerging Trends in Computing and Information Sciences, 2012, 3(4): 637 – 644.

[3] ZHU Y. Image Classification based on Hadoop platform [J]. Journal of Southwest University of Science and Technology, 2011, 26(2): 70 – 73. (朱义明. 基于 Hadoop 平台的图像分类[J]. 西南科技大学学报, 2011, 26(2): 70 – 73.)

[4] WHITE T. Hadoop: the definitive guide [M]. 2nd ed. Sebastopol, CA: O'Reilly, 2010: 203.

[5] DONG B, QIU J, ZHENG Q, *et al*. A novel approach to improving the efficiency of storing and accessing small files on Hadoop: a case study by PowerPoint files [C]// SCC 2010: Proceedings of the 2010 IEEE International Conference on Services Computing. Washington, DC: IEEE Computer Society, 2010: 65 – 72.

[6] SWEENEY C, LIU L, ARIETTA S, *et al*. HIPI: a Hadoop image processing interface for image-based mapreduce tasks [D]. Charlattesville: University of Virginia, 2011.

[7] CSURKA G, DANCE C, FAN L, *et al*. Visual categorization with bags of keypoints [C]// ECCV 2004: Proceedings of the 8th European Conference on Computer Vision, LNCS 3023. Berlin: Springer, 2004: 22.

[8] SIVIC J, ZISSERMAN A. Video Google: a text retrieval approach to object matching in videos [C]// ICCV'03: Proceedings of the Ninth IEEE International Conference on Computer Vision. Washington, DC: IEEE Computer Society, 2003: 1470 – 1477.

[9] LI F F, PERONA P. A Bayesian hierarchical model for learning natural scene categories [C]// CVPR'05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2005: 524 – 531.

[10] LOWE D G. Distinctive image features from scale-invariant keypoints [J]. International Journal of Computer Vision, 2004, 60(2): 91 – 110.

[11] VALGREN C, LILIENTHAL A J. SIFT, SURF & seasons: appearance-based long-term localization in outdoor environments [J]. Robotics and Autonomous Systems, 2010, 58(2): 149 – 156.

[12] DREUW P, STEINGRUBE P, HANSELMANN H, *et al*. SURFface: face recognition under viewpoint consistency constraints [C]// Proceedings of the 2009 British Machine Vision Conference. London: BMVA Press, 2009: 1 – 11.

[13] BAY H, ESS A, TUYTELAARS T, *et al*. Speeded-Up Robust Features (SURF) [J]. Computer Vision and Image Understanding, 2008, 110(3): 346 – 359.

[14] MOOSMANN F, NOWAK E, JURIE F. Randomized clustering forests for image classification [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008, 30(9): 1632 – 1646.

[15] HUISKES M J, LEW M S. The MIR flickr retrieval evaluation [C]// Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval. New York: ACM Press, 2008: 39 – 43.

[16] GOLLER A, GLENDINNING I, BACHMANN D, *et al*. Parallel and distributed processing [M]// Digital Image Analysis. Berlin: Springer-Verlag, 2001: 135 – 153.