

基于 Hadoop 的服饰图像存储与检索关键技术研究*

郭飞^{1a}, 詹炳宏^{1b,2}, 刘刚³

(1. 北京服装学院 a. 信息工程学院; b. 艺术设计学院, 北京 100029; 2. 北京市数字媒体与交互重点实验室, 北京 100029; 3. 河南工业大学 信息科学与工程学院, 郑州 450001)

摘要: 针对服饰图像都是小文件的特性, 提出了一种 HDFS 和传统关系型数据库相结合的服饰图像及其特征数据的存储结构设计方法, 实现了海量图像信息的快速存储和读取; 改进了多特征点图像特征提取和匹配算法, 并基于 Map/Reduce 框架实现了基于多特征的服饰图像数据分布式检索。实验结果表明, 该方法能够均衡系统负载, 提高资源利用率, 扩展性强, 有效地降低了海量服饰图像检索时间, 是一种高效的服饰图像存储和检索的方法。

关键词: Hadoop 云存储; 小文件; 基于内容的图像检索; 服饰图像

中图分类号: TP391.41 **文献标志码:** A **文章编号:** 1001-3695(2014)04-1086-04

doi: 10.3969/j.issn.1001-3695.2014.04.031

Research on key technology of clothing image storage and retrieval based on Hadoop

GUO Fei^{1a}, ZHAN Bing-hong^{1b,2}, LIU Gang³

(1. a. School of Information Engineering, b. School of Art & Design, Beijing Institute of Fashion Technology, Beijing 100029, China; 2. Beijing Key Laboratory of Digital & Interactive Media, Beijing 100029, China; 3. College of Information Science & Engineering, Henan University of Technology, Zhengzhou 450001, China)

Abstract: According to the clothing image characteristics are small file, this paper proposed a storage structure design method of clothing image and their feature data, which combined HDFS and traditional relational database. It achieved rapidly storage and retrieval for massive image information. Then it improved multi-feature point image feature extraction and matching algorithms, and implemented the clothing image data distributed multi-feature retrieval based on the Map/Reduce platform. The experimental results demonstrate that this method can balance the system load, improve resource utilization and it has strong expansibility, effectively reduces the retrieval time of massive clothing image. It is a efficient storage and retrieval method for clothing image.

Key words: Hadoop cloud storage; small file; content-based image retrieval; clothing image

我国是服饰生产和销售大国, 随着多媒体和网络技术的快速发展, 以及电子商务的迅猛崛起, 服饰图像信息正在以惊人的速度增长。针对服饰的设计、生产、管理, 尤其是 B2C、C2C 销售体系, 怎样有效地组织、管理、存储、检索这些海量服饰图像信息成为国内外相关研究机构和学者关注的热点领域。服饰图像包含丰富的视觉信息(如形状、颜色、纹理等)和语义情感, 通常难以用文字进行客观的描述, 因此基于内容的图像检索(content-based image retrieval, CBIR)技术更适用于对服饰图像信息的检索^[1]。CBIR 通过图像的颜色、形状、纹理、语义等特征对图像进行查询, 在分析图像内容的基础上, 检索出与输入图像相类似的图像^[2,3]。而对服装图像特征数据的高效管理是实现基于内容的服饰图像检索的重要前提。

随着服饰图像的快速增长和图像检索的高并发操作, 对服饰图像特征数据的高效管理和分析成为关键问题, 这主要体现在三个方面: a) 服饰图像数据具有海量特点, 传统的单机存储

受到限制, 在性能和扩展性方面难以满足海量数据要求; b) 服饰图像检索是数据密集型计算^[4], 在处理海量数据时, 传统的单节点系统的实时性和稳定性不能得到保障; c) 服饰图像及其特征数据为非结构化数据, 传统的关系型数据库对数据模式约束过于严格, 难以满足其管理需要。利用计算机集群的分布式算法是解决上述问题行之有效的方法。Hadoop 分布式文件系统(Hadoop distributed file system, HDFS)^[5]和 Map/Reduce^[6,7]分布式编程框架, 提供了分布式数据存储和分析的解决方案, 具有自动负载均衡、较高容错性、易用等优点, 为海量服饰图像数据的并行处理提供了基础。

本文提出一种关系型数据库(relational database management system, RDBMS)和 HDFS 结合的小文件存储方法, 并将基于改进的多特征服饰图像检索技术与 Map/Reduce 并行计算框架相结合, 实现了基于 Hadoop 的分布式服饰图像存储和检索。该方法不局限于对海量服饰图像的分布式管理, 对于其他

收稿日期: 2013-05-30; **修回日期:** 2013-08-20 **基金项目:** 国家科技支撑计划资助项目(2013BAH41F00); 国家自然科学基金资助项目(61174056); 北京市数字与交互重点实验室资助项目(KF2012-15)

作者简介: 郭飞(1978-), 女, 河北邢台人, 讲师, 博士, 主要研究方向为计算机应用、智能识别与控制(feierguo77@163.com); 詹炳宏(1968-), 男, 教授, 博士, 主要研究方向为信息交互设计; 刘刚(1973-), 男, 副教授, 博士, 主要研究方向为智能信息处理与控制、优化算法。

的非结构化的海量小文件数据同样适用。

1 分布式系统整体架构

基于Hadoop的服饰图像存储与检索系统总体上可以分为表现层、应用逻辑层、数据存储与处理层三个层次,系统总体架构如图1所示。各层的主要作用为:

a)表现层。通过良好的交互设计实现用户和系统的交互。用户通过客户端提交待搜索的服饰图像信息或其元数据及语义信息,将服务器搜索的结果以较好的方式呈现给用户。

b)应用逻辑层。针对用户提交的服饰图像信息,执行相应的业务逻辑,完成检索操作。

c)数据存储和处理层。根据用户提交的相关信息及设置的检索类别完成相关的数据分析。如果用户选择基于图像内容检索,则通过Map/Reduce实现待检索图像的特征提取,并与HDFS中存储的服饰图像特征库进行相似度检索,根据匹配结果将HDFS中存储的服饰图像返回给用户。如果用户选择元数据及语义检索,则将用户输入的元数据与语义信息与RDBMS中存储的元数据与语义库相匹配,并根据匹配结果将HDFS中存储的服饰图像返回给用户。如果用户选择两者相结合的检索方式,将元数据及语义检索作为图像内容检索的入口,根据加权算法,综合计算匹配结果。

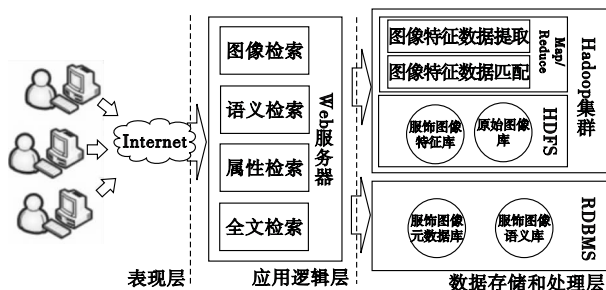


图1 基于Hadoop的服饰图像存储与检索系统整体框架图

2 基于HDFS和RDBMS的服饰图像存储结构设计

HDFS作为分布式文件系统,初始是为了处理大文件(文件大小通常在几百兆以上)而设计,而对于小文件(指尺寸远小于HDFS块大小的文件)会影响其性能及扩展性^[8]。其主要有两个方面的问题:a)NameNode将文件系统的文件、目录信息以对象的形式保存在内存中,大量的小文件必然导致NameNode中对象数量急剧膨胀,大量占用NameNode的内存,最终可能会超出NameNode的物理内存容量,制约了HDFS系统的扩展性;b)Map/Reduce计算是基于文件分片进行的,如果大量的小文件存在,必然导致文件分片过小,在计算时会产生过多的map和reduce任务,这些任务的频繁启停会极大延迟计算完成时间,导致系统性能低下。因此,在使用Hadoop时需要避免基于小文件的计算,而服饰图像的特点是单个文件的体积小,一般为几百KB到几MB的大小,远远小于HDFS的默认块大小(64 MB),而且服饰图像的数量具有海量特性。因此在基于Hadoop进行计算的时候迫切需要解决小文件的存储方式。本文提出一种关系型数据库和HDFS相结合的服饰图像和特征数据小文件存储方法,将图像元数据和图像语义数据存储的关系数据库RDBMS中,将预处理的图像文件和特征数据存储于HDFS中。具体结构如图2所示。

图2中上半部分表示图像元数据在RDBMS中的存储结构,

在图像元数据中定义了主键ID、款式、材料等各种属性,图像元数据实际描述了系统中的一个图像文件,因此在HDFS中使用图像元数据的主键ID来唯一标志图像。在HDFS中存储服饰图像文件的时候通过键值对的形式进行存储,键存储图像元数据的主键ID,值存储图像文件的二进制内容。通过这样一种map file的形式,可以将所有原始图像文件存储在一个HDFS文件中,在基于Map/Reduce进行计算的时候由Hadoop自动进行文件的分片和生成适量的Map/Reduce计算任务。同理,对于图像颜色特征数据、图像纹理特征数据等每个内容都存为一个这样的map file文件,并基于这些特征数据文件进行图像检索。通过这样的文件存储结构可以极大地减少文件的数量和Map/Reduce任务的数量,提高了系统性能和扩展性。

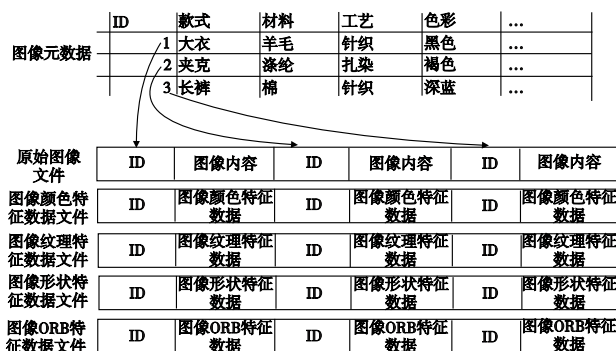


图2 图像数据存储结构图

3 分布式多特征图像检索

3.1 Map/Reduce 设计

Map/Reduce计算模型的核心是map和reduce两个函数,这两个函数的输入和输出如表1所示。为实现服饰图像的检索,需要实现离线图像特征生成和在线图像特征匹配两个Map/Reduce模块,这两个模块根据一定的算法将输入的<key, value>对处理成另一批<key, value>对输出,从而产生需要的最终检索结果。

表1 Map/Reduce 函数输入输出表

函数	输入	输出	说明
map	$\langle K_1, V_1 \rangle$	$\text{list}(\langle K_2, V_2 \rangle)$	将输入文件的一个小数据集解析成一批 $\langle K_1, V_1 \rangle$ 对,输入到map函数中进行处理, $\langle K_2, V_2 \rangle$ 是计算的中间结果
reduce	$\langle K_2, \text{list}(V_2) \rangle$	$\langle K_3, V_3 \rangle$	中间结果 $\langle K_2, \text{list}(V_2) \rangle$ 中 $\text{list}(V_2)$ 表示是属于同一个 K_2 的一批 V_2

3.1.1 离线生成图像特征库

Map任务输入的 K_1 为图像元数据的主键ID, V_1 为图像的二进制内容,在map任务中对每个图像文件提取多特征向量计算,map任务输出的 K_2 也为图像元数据的主键ID, V_2 为图像的多特征数据。Reduce任务输入为map任务的输出,reduce任务输出 $\langle K_3, V_3 \rangle$ 为根据图像元数据主键ID全局排序之后的图像多特征数据,分别以map file的形式存储在一个文件中。

3.1.2 图像特征匹配 Map/Reduce 模块

Map任务输入的 K_1 为图像元数据的主键ID, V_1 为图像特征数据。在map任务中获取待检索示例图像的特征数据,并计算输入的每个图像特征数据与示例图像特征数据的相似度,从而得到符合相似度阈值的图像。经过这样的计算之后,map任务输出的 K_2 为图像元数据的主键ID,输出的 V_2 为空,此

map 任务计算的目的是获取相似性的图像元数据 ID。如果只是基于内容的图像检索则不需要 reduce 计算;如果同时结合元数据和语义进行文字检索,则需要 reduce 计算。Reduce 任务输入的 K_2 为图像元数据主键 ID, V_2 为空,在 reduce 任务中把图像文字检索的结果进行匹配, $\langle K_3, V_3 \rangle$ 为既符合内容检索又符合文字检索的图像 ID,最后将检索结果通过 Web 服务展现给用户。

3.2 多特征图像检索算法

CBIR 方法首先计算待检索示例图像 I 的综合特征向量 $F = (F_1, F_2, \dots, F_n)$, 其中 F_i 是图像 I 的第 i 种特征向量;然后根据 F 检索特征库,得到与 F 距离小于阈值的特征向量组,此组特征向量所对应的图像就是与 I 相似的检索结果^[9,10]。

本文针对服饰图像具有鲜明的颜色、纹理、形状特征的特点,提出一种全局特征和局部特征相结合的特征提取和匹配算法。颜色、纹理、形状特征为全局特征,局部特征采用 ORB (oriented FAST and rotated BRIEF) 描述子,根据用户的选择设置不同特征的权重值,计算加权相似度来进行图像检索。

3.2.1 基于颜色直方图的颜色特征提取

颜色直方图是颜色特征提取中最常用的一种方法^[11]。首先将 RGB 颜色空间变换为 HSV 颜色空间,采用非等间隔量化的方法,将色度 H 划分为八个等级,饱和度 S 和亮度 V 划分为三个等级。按照公式 $L = HQ_s Q_v + SQ_v + V$ 将量化后的三个分量合成一维特征向量,得到 72 柄一维直方图 $l = 9H + 3S + V$ ($Q_s = 3, Q_v = 3$),则直方图向量如式(1)所示。

$$L = \{h[0], h[1], \dots, h[l], \dots, h[71]\} \quad 0 \leq l \leq 71 \quad (1)$$

其中: $h[l]$ 表示第 l 种颜色出现的频次。同时,计算直方图的加权平均颜色 $l_{\text{mean}} = \sum_{l=0}^{71} l \times h[l]$ 和图像主颜色 $l_{\text{max}} = \max(h[l])$ 。

采用欧氏距离进行图像间颜色特征距离的度量,对颜色直方图、平均颜色、图像主颜色分别计算其欧氏距离,然后进行加权平均得到颜色特征的距离 dist_c 。

3.2.2 基于 Gabor 小波变换的纹理特征提取

纹理特征也是服饰图像的重要特征之一,它反映了图像中不依赖于颜色或者亮度的视觉信息。采用 Gabor 小波进行纹理特征提取,被证明是较好的方法^[12,13]。2D 的 Gabor 小波函数为^[14]

$$\psi(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp(2\pi j\omega x) \exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right] \quad (2)$$

对于给定的 $P \times Q$ 大小的图像 $I(x, y)$, 它的小波变换为

$$G_{mn}(x, y) = \sum_s \sum_t I(x-s, y-t) \psi_{mn}^*(s, t) \quad (3)$$

其中, $m=0, 1, \dots, M-1, n=0, 1, \dots, N-1$, 分别是小波的尺度和方向; M 和 N 表示尺度数和方向数; s, t 为滤波器模板变量; ψ_{mn}^* 是 ψ_{mn} 共轭复数。自相似小波函数如式(4)所示。

$$\psi_{mn}(x, y) = a^{-m} \psi(\tilde{x} \tilde{y})$$

$$\tilde{x} = a^{-m}(x \cos \theta + y \sin \theta), \tilde{y} = a^{-m}(-x \cos \theta + y \sin \theta) \quad (4)$$

其中: $a > 1$ 为伸缩因子, $\theta = n\pi/N$ 。

计算图像经尺度和方向 Gabor 变化后系数的均值 μ_{mn} 和方差 σ_{mn} , 作为图像的纹理特征。实验中取尺度为 5, 方向为 6, 则服饰图像纹理特征表示为

$$F = (\mu_{00}, \mu_{01}, \dots, \mu_{45}, \sigma_{00}, \sigma_{01}, \dots, \sigma_{45}) \quad (5)$$

纹理特征的相似度采用欧氏距离进行计算,得到归一化后的纹理特征欧氏距离 dist_t 。

3.2.3 基于不变矩的形状特征提取

边缘不变矩是反映二值图像分布的重要数字特征,它具有旋转、平移、缩放等空间几何不变性,因此被广泛用来描述图像形状特征^[15]。假定 $M \times N$ 的二值图像 $I(x, y)$, 它的 $p+q$ 中心阶矩 μ_{pq} 为

$$\mu_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x-x_c)^p (y-y_c)^q I(x, y) \quad (6)$$

其中, (x_c, y_c) 为中心点。然后利用图像的第 2 和第 3 阶矩计算归一化的中心矩,再由中心矩求出七个对平移、旋转以及尺度不变的特征 $\varphi_1 \sim \varphi_7$ 作为图像的形状特征向量。

形状特征的相似度采用欧氏距离进行计算,得到归一化后的形状特征欧氏距离 dist_s 。

3.2.4 基于 ORB 算法的局部特征提取

ORB 算法是基于 FAST 特征检测和 BRIEF 描述子的改进算法^[16,17]。与传统的 FAST 算法不同,ORB 算法在提取特征点之前首先构建图像高斯金字塔,在金字塔每层提取特征点,从而具有了尺度特征。ORB 算法给每个特征点都添加了一个方向角,然后基于 BRIEF 算法生成描述子,并在生成描述子前对每个特征点的方向角进行旋转。ORB 算法通过获取特征点邻域的重心确定方向。特征点邻域如式(7)所示。

$$m_{pq} = \sum_{x,y} x^p y^q I(x, y) \quad (7)$$

重心点 P 的坐标为

$$P = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \quad (8)$$

特征点和重心的夹角定义为特征点方向:

$$\theta = \arctan\left(\frac{m_{01}}{m_{10}}\right) \quad (9)$$

x, y 是相对于 FAST 特征点的位置,圆形邻域半径为 $r, x, y \in [-r, r]$ 。

在特征点附近选取若干点对,将这些像素点对的灰度值大小进行比较,组成一个二进制串,即为特征描述子。为了降低对噪声的敏感度,在特征点 31×31 像素邻域内随机选取 5×5 的像素块对,使用积分图对若干对像素块进行比较得到二进制串。由此求角法计算的描述子相关性高,采用贪心算法寻找相关性较低的描述子。一般选取 256 个相关性最低的像素块构成 256 bit 特征描述子,通过汉明距离之间的异或之和来计算两个描述子之间的相似度:

$$\text{dist}_o(K_1, K_2) = \sum_{i=0}^{255} x_i \oplus y_i \quad (10)$$

其中, $K_1 = x_0 x_1 \dots x_{255}, K_2 = y_0 y_1 \dots y_{255}$ 为两个描述子。

3.2.5 多特征融合相似度

分别计算出待检索图像与特征库中指定特征(颜色、纹理、形状、ORB 特征)的相似距离,并进行归一化处理,用 $\text{dist}_c', \text{dist}_t', \text{dist}_s'$ 表示,为单一特征分配相应的权值; W_c, W_t, W_s, W_o 分别为颜色、纹理、形状、ORB 特征的权值。则待检索图像和库中任意图像间的特征融合相似度如式(11)所示,即各相似距离归一化后的加权平均值为

$$D = \frac{W_c \times \text{dist}_c' + W_t \times \text{dist}_t' + W_s \times \text{dist}_s' + W_o \times \text{dist}_o'}{W_c + W_t + W_s + W_o} \quad (11)$$

4 实验分析

4.1 实验环境及设备

通过 20 台服务器构建 Hadoop 分布式系统集群,其中 1 台

为NameNode节点,其余19台为DataNode节点。分布式系统中节点通过千兆以太网连接,使用Java语言实现Map/Reduce框架和服饰图像检索算法。Hadoop版本为0.20.2,JDK版本为jdk1.6.0_35,服务器配置如表2所示。

表2 Hadoop分布式系统的服务器配置

节点	硬件配置(CPU、内存)	操作系统	IP
NameNode	Intel Xeon 3.0 GHz, 8 GB 内存	Linux	192.168.1.100
DataNode1	Intel Core i7, 4 GB 内存	Linux	192.168.1.101
...	...	...	...
DataNode19	Intel Core i7, 4 GB 内存	Linux	192.168.1.119

4.2 实验数据

本实验采用80万张服饰图像作为测试数据。单张图像的每个全局特征数据及其ID平均大约占580 Byte,10万张图像的全局特征数据约占58 MB。单张图像的ORB特征描述子及其ID平均约占12 KB,10万张图像的ORB特征数据约1.2 GB,检索时Map/Reduce自动分块给各个节点运行。本实验分别采用10万、20万、40万、60万、80万张服饰图像作为实验数据。

4.3 实验算法实现

4.3.1 文件存储算法实现

Map文件由两个文件组成:数据文件和索引文件,数据文件采用前面介绍的ID为键,值为文件内容,每个小文件在map文件中都存储为ID和内容的序列,其中ID为定长的8 Byte,内容的长度不定,因此每个小文件的开始位置也不定。为解决小文件定位问题,将小文件的起始位置保存到专门的索引文件中,在索引文件中同样是ID和位置的序列,在此ID和位置都是固定长度的8 Byte。

数据文件结构示例如下:

321	内容	322	内容	323	内容
-----	----	-----	----	-----	----

索引文件结构示例如下:

321	0	322	29170	323	68972
-----	---	-----	-------	-----	-------

上面的图例中有三个小文件,通过上面的算法将三个小文件合并为一个文件(map file),因此只在NameNode中占用一个文件的内存空间,而不是原来的三个文件内存空间。在进行Map/Reduce计算的时候针对这个map file只需要启动一次任务即可完成计算,而不是原来的需要启动三次任务,大幅减少了Map/Reduce任务频繁启停的次数,因此将更多的CPU时间用在了核心计算上,获得了更高的效率。

4.3.2 图像特征提取与检索算法实现

图像特征提取与检索都是基于Map/Reduce来实现的,实现一个基于Map/Reduce的并行计算,关键是map函数和reduce函数的实现。

图像特征提取的map函数如下:

```
public void map (LongWritable key, Text value, OutputCollector<LongWritable, Text> output, Reporter reporter)
```

其中:key为图像的主键ID;value为图像的原始内容;输出的LongWritable为图像的主键ID;Text为图像的特征数据。在此map函数中对输入的图像原始内容进行计算获取图像的综合特征数据并输出。

图像特征提取的reduce函数如下:

```
public void reduce (LongWritable key, Iterator<Text> values, OutputCollector<LongWritable, Text> output, Reporter reporter)
```

其中:key为图像的主键ID;values为图像的特征数据;输出的LongWritable为图像的主键ID;Text为图像的特征数据。在此reduce函数中对输入的图像特征数据进行全局排序,output是经过全局排序后的图像特征数据的集合。

图像检索的map函数如下:

```
public void map (LongWritable key, Text value, OutputCollector<LongWritable, Text> output, Reporter reporter)
```

其中:key为图像的主键ID;value为图像的特征数据;输出的LongWritable为图像的主键ID;Text为空。在map函数中将图像特征数据与待检索图像的特征数据进行匹配获得最终的检索结果,output为检索结果的主键ID集合。

图像检索的reduce函数如下:

```
public void reduce (LongWritable key, Iterator<Text> values, OutputCollector<LongWritable, Text> output, Reporter reporter)
```

其中:key为图像的主键ID;values为空;输出的LongWritable为图像的主键ID;Text为图像的元数据。在需要进行图像属性、文字检索的时候执行此reduce函数,在reduce函数中将map中匹配的图像进行文字、属性的检索,最后获得既符合内容特征又符合文字描述特征的图像,output是检索图像的结果集合。

4.4 实验结果分析

4.4.1 文件存储结构实验性能分析

当管理50万张服饰图像时,本文提出的存储架构和单纯小文件存储方式,NameNode的初始内存对比如图3所示。由图3中可以看出采用本文提出的改进存储结构,初始时NameNode内存占用明显下降,也就是同样的硬件配置,Hadoop可以管理更多的文件。当对10万张图像基于Map/Reduce框架采用10个节点检索图像时,小文件存储方式和本文改进方式平均耗时(30位并发用户平均时间)对比如图4所示。从图4中看出,采用本文存储方式,检索性能有很大提高。

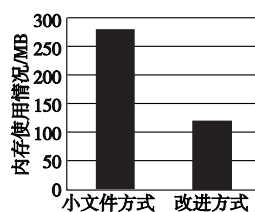


图3 NameNode内存占用情况对比

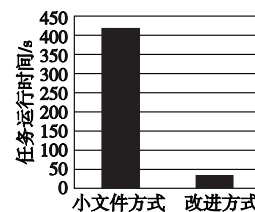


图4 平均耗时对比

4.4.2 不同节点数检索性能对比分析

图像数据分别为10万、20万、40万、60万、80万时,10个DataNode和19个DataNode模式下,检索图像耗时对比如图5所示。

当图像数据特征为10万条时,传统的单服务器B/S模式的检索时间(30位并发用户平均时间)约为277 s,增加到80万数据时检索耗时巨大,将近8倍,因此没在图中体现。采用Hadoop模式,10万条数据时全局特征数据被作为一个map任务的输入进行处理,如果采用10个节点,则ORB数据被分在10个节点上处理,一个节点约处理两个map任务。如果用19个节点,则一个节点处理约一个map任务,性能有很大提高,提高约35.2%。由于任务的初始化、分配以及清除作业的耗时操作,所以虽然节点数增加将近1倍,性能并没有增加1倍。当处理数据达到80万条时,性能提高 (下转第1126页)

表1 不同噪声下的仿真结果比较

噪声方差	算法	均方根误差平均值	估计时间/s
$Q_e = 10$ $R_e = 1$	1	4.796	1.031
	2	3.679	1.026
	3	2.441	1.013
$Q_e = 20$ $R_e = 8$	1	6.340	1.068
	2	4.878	1.054
	3	2.560	1.015
$Q_e = 30$ $R_e = 15$	1	8.427	1.102
	2	6.083	1.061
	3	2.773	1.020

4 结束语

1) 标准 UKF 算法用于非线性系统滤波时具有较好的鲁棒性和收敛速度,但当测量数据中存在野值时,定位精度却降低了。基于方差分量统计量的自适应因子能够自适应地调整不同类或不等精度观测值之间的权比,使权值的分配更具合理性,进而提高了定位精度,但是却不具有抵抗粗差的能力,所以本文提出了抗差 Helmert 方差分量统计自适应因子。

2) 通过实验仿真验证了 ARUKF 算法的可行性。相比方案 1 和 2, ARUKF 算法能很好地提高定位精度,有效地剔除了测量数据中的野值,滤波动态性能得到明显提高。

参考文献:

[1] 万德钧,房建成,王庆. GPS 动态滤波的理论、方法及应用[M]. 南

京:江苏科学技术出版社,2000:73-80.

- [2] 程兰,谢刚. 一种基于扩展 Kalman 滤波的多径估计算法[J]. 太原理工大学学报,2012,43(5):575-579.
- [3] JULIER S J, UHLMANN J K. New extension of the Kalman filter to nonlinear systems[C]//Proc of the 11th International Symposium on Aerospace/Defence Sensing, Simulation and Controls. 1997:182-193.
- [4] 张红梅,邓正隆. UKF 方法在陆地车辆组合导航中的应用[J]. 中国惯性技术学报,2004,12(4):20-23.
- [5] 汪秋婷,胡修林. 基于 UKF 的新型北斗/SINS 组合系统直接法卡尔曼滤波[J]. 系统工程与电子技术,2010,32(2):376-379.
- [6] 李海勇,赵伟,熊剑,等. 强跟踪 UKF 滤波在 SINS/GPS 组合导航中的应用研究[J]. 航空电子技术,2008,39(4):5-10.
- [7] 周露平,王智灵,陈宗海. 基于 M-估计的 UKF 算法及其在运动估计中的应用[J]. 模式识别与人工智能,2010,20(6):849-854.
- [8] 万某峰,赵长胜,王飞. 预测残差判别统计量抗差因子在 UKF 算法中的应用[J]. 测绘科学技术学报,2013,30(1):33-35.
- [9] 杨元喜,任夏,许艳. 自适应抗差滤波理论及应用的主要进展[J]. 导航定位学报,2013,1(1):9-15.
- [10] 闫鑫. 基于抗差估计的 GPS/MIMU 组合导航滤波算法研究[D]. 哈尔滨:哈尔滨工程大学,2010.
- [11] 申文斌. 高精度卡尔曼滤波研究及其在导航系统中的实现[D]. 广州:华南理工大学,2011.
- [12] 唐艳,张晓琳,侯冰,等. COMPASS/GPS 双系统四星定位算法研究[J]. 遥测遥控,2012,33(4):13-19.
- [13] 张延鹏,张赢硕,王尔申,等. BD-2/GPS 组合系统的设计与定位算法[J]. 电子工程设计,2011,19(23):74-77.

(上接第 1089 页)约 43%。由此可知,Hadoop 在处理海量、数据处理密集型数据时,具有很强的优势。

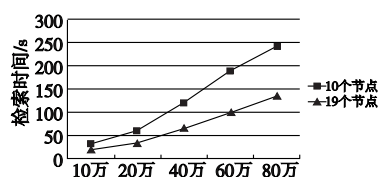


图5 不同DataNode数时检索时间对比

5 结束语

本文在分析小文件存储对 HDFS 和 Map/Reduce 性能影响的基础上,结合关系性数据库实现了服饰图像及其特征数据的 Hadoop 存储。同时,给出了在 Map/Reduce 框架下实现图像分布式检索的具体改进算法描述和实现方案。经实际验证,该方法能有效地解决 Hadoop 处理小文件时 NameNode 内存瓶颈问题,同时具有较好的检索性能和扩展性,能够支撑上层基于特征数据的服饰图像内容检索需求。未来的工作将根据服饰图像检索需求进一步优化多特征点匹配算法及完善系统功能。

参考文献:

- [1] RUI Yong, 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(3):39-62.
- [2] 周明全,耿国华,韦娜. 基于内容图像检索技术[M]. 北京:清华大学出版社,2007.
- [3] 孟繁杰,郭宝龙. CBIR 关键技术研究[J]. 计算机应用研究,2004,21(7):21-24,27.
- [4] 郑湃,崔立真,王海洋,等. 云计算环境下面向数据密集型应用的数据布局策略与方法[J]. 计算机学报,2010,33(S):1472-1480.

- [5] HDFS[EB/OL]. [2009-07-10]. <http://wiki.apache.org/hadoop/ProjectDescription>.
- [6] Map/Reduce[EB/OL]. [2009-07-10]. <http://wiki.apache.org/hadoop-HadoopMapReduce>.
- [7] DEAN J, GHEMAWAT S. MapReduce: simplified data processing on large clusters[J]. Communications of the ACM, 2008, 51(1):107-113.
- [8] LIU Xu-hui, HAN Ji-zhong, ZHONG Yun-qin, et al. Implementing WebGIS on Hadoop: a case study of improving small file I/O performance on HDFS[C]//Proc of IEEE International Conference on Cluster Computing and Workshops. Piscataway: IEEE Press, 2009:1-8.
- [9] 唐立军,段立娟,高文. 基于内容的图像检索系统[J]. 计算机应用研究,2001,18(7):41-45.
- [10] 田卉,覃团发,梁琳. 综合颜色、纹理、形状和相关反馈的图像检索[J]. 计算机应用研究,2007,24(11):292-294,314.
- [11] 金楠,任秀丽. 图像检索中颜色特征的提取与匹配[J]. 计算机辅助设计与图形学学报,2000,12(6):459-462.
- [12] MA W Y, MANJUNATH B S. Texture features and learning similarity[C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. 1996:425-430.
- [13] DUNN D, HIGGINS W E. Optimal Gabor filters for texture segmentation[J]. IEEE Trans on Image Processing, 1995, 4(7):227-237.
- [14] MANJUNATH B S, MA W Y. Texture features for browsing and retrieval of image data[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 1996, 18(8):837-842.
- [15] 黄勇,王崇骏,王亮,等. 基于形状不变矩的图像检索算法研究[J]. 计算机应用研究,2004,21(7):256-260.
- [16] ETHAN R, VINCENT R, KURT K, et al. ORB: an efficient alternative to SIFT or SURF[C]//Proc of International Conference on Computer Vision. 2011:2564-2571.
- [17] 任结,周余,于耀,等. 基于 ORB 自然特征的 AR 实时系统实现[J]. 计算机应用研究,2012,29(9):3594-3596.