doi:10.6041/j.issn.1000-1298.2018.09.020

基于 Hadoop 的蛋鸡设施养殖智能监测管理系统研究

孟超英[1]　张雪彬[1]　陈红茜[2]　李　辉[1]

(1. 中国农业大学信息与电气工程学院, 北京 100083; 2. 中国农业大学网络中心, 北京 100083)

摘要: 为实现对蛋鸡生产过程中长期积累的海量数据进行高效存储和实时查询,利用 Hadoop 生态系统,设计了规模化蛋鸡设施养殖智能监测管理系统。针对环境数据的实时监测及大规模数据查询,用 MySQL 数据库存储近期数据、HBase 存储历史数据,有效提升了检索速度;针对海量异构视频数据的统一管理,设计实现了基于 MapReduce 并行处理框架的分布式转码模块,将 1.5 GB 的视频分割为多个 128 MB 分段后进行转码,转码效率提高了 50%。该系统实现了规模化蛋鸡场生产养殖中对实时信息、历史信息、基础设施信息、生产过程信息的统一管理,并提供了统计分析模块对采集获取的数据进行整合分析,开发了 Web 端网页版本及移动端 APP 版本的智能监测管理系统,便于用户进行实时访问,提高了生产养殖的工作效率。

关键词: 蛋鸡; 数据查询; 智能监测; Hadoop; HBase

中图分类号: S815; TP391　　**文献标识码**: A　　**文章编号**: 1000-1298(2018)09-0166-10

Design of Intelligent Monitoring and Management System Based on Hadoop for Large-scale Layer House

MENG Chaoying[1]　ZHANG Xuebin[1]　CHEN Hongqian[2]　LI Hui[1]

(1. *College of Information and Electrical Engineering*, *China Agricultural University*, *Beijing* 100083, *China*
2. *Network Center*, *China Agricultural University*, *Beijing* 100083, *China*)

Abstract: With the appearance and continuous development of the Internet of things, the monitoring data grows explosively. Accordingly, traditional data storage and processing can not meet the requirements. In order to store data effectively and query data in real time, intelligent monitoring and management system based on Hadoop for large-scale layer house was developed. The HDFS file system and HBase database in the Hadoop ecosystem can store massive data distributed. The environmental monitoring data had the characteristics of once writing and multiple queries. In order to realize the real-time monitoring and large-scale data query for environmental data, MySQL database was used to store recent data and HBase database was used to store historical data. Experiments indicated that the query speed was improved effectively. For the unified management of massive heterogeneous video data, the distributed transcoding of video was designed and implemented. Experimental results showed that the proposed scheme can increase about 50% of the transcoding efficiency when the video size was 1.5 GB and the segment size was 128 MB. The system realized real-time information display, historical information query, infrastructure management, production process management and statistical data analysis, environmental alerts and system management in production and breeding of large-scale layer house, which can be accessed through web pages and mobile APP by users in real time. The actual application showed that the system helped managers to control the production process on all aspects and improve the efficiency of the production personnel.

Key words: laying hens; data query; intelligent monitoring; Hadoop; HBase

0　引言

随着物联网、云计算、互联网技术等的不断发展,信息化水平不断提高,各行业数据爆炸式增长[1]。我国是农业大国,农业大数据对农业现代化具有重要意义[2]。而蛋鸡产业是国内养殖业的主

收稿日期: 2018-03-25　修回日期: 2018-04-28

基金项目: 国家重点研发计划项目(2017YFD0701602、2016YFD0700204)

作者简介: 孟超英(1958—),女,教授,主要从事计算机科学与技术研究,E-mail: mcy@cau.edu.cn

导产业之一，近年来其整合速度、规模化速度正在加速发展[3]。在避免人为干扰的情况下，获取生产养殖、环境监测数据并进行处理分析得到了广泛深入研究。目前蛋鸡养殖企业采用物联网系统，通过传感器实时采集数据并进行展示已经得到应用[4-5]。对采集到的数据进行自动化处理和分析也已经得到小规模示范实施[6]。随着系统的常年运行、数据不断积累，产生的海量数据存储、查询及分析处理问题亟待解决。对于海量环境监测数据，传统数据库的处理能力无法满足实时数据查询的需求。对于海量视频数据，由于格式不同，视频统一管理、实时查看较为困难，后续基于视频的蛋鸡行为分析[7]也将更加复杂。

在前期物联网数据采集系统的基础上，本文设计基于 Hadoop 的规模化蛋鸡设施养殖智能监测管理系统，使用 Hadoop 框架及其生态系统对规模化蛋鸡养殖产生的生产环境数据、生产过程数据及现场监控视频数据进行高效存储、处理和分析，从而进行实时监测、统一管理。

1　Hadoop 及其生态系统

Hadoop 是 Apache 基金会开发的分布式计算框架，可以对海量数据进行分布式存储及并行运算[8]。Hadoop 是当前热门的云平台之一，具有灵活可扩展、经济可靠等优势。目前，Hadoop 已经形成一套包括分布式文件系统（Hadoop distributed file system，HDFS）、MapReduce、HBase、Zoo - keeper、Hive、Pig 等在内的生态系统，并在不断发展扩充[9]。Hadoop2.0 生态系统如图 1 所示。

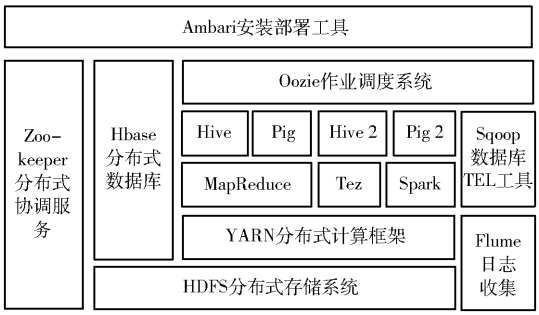


图 1　Hadoop2.0 生态系统

Fig. 1　Hadoop2.0 ecosystem

Hadoop 框架由 2 个核心设计组成：分布式文件系统 HDFS 和分布式并行计算框架 MapReduce[10-12]。HDFS[13-14]作为 Hadoop 的数据存储核心，具有高容错性、高吞吐量等特点。可以高效存储蛋鸡生产养殖中所产生的异构数据。MapReduce[15]用于海量数据的并行运算，为蛋鸡舍监控视频的并行处理提供了支撑。而 Hadoop 生态系统中的 HBase[16-17]是一个基于 HDFS 的分布式非关系型数据库，支持数据的随机查找，能够处理大规模数据，适合海量环境监测数据的实时查询。

2　系统设计

2.1　系统总体架构

基于 Hadoop 的规模化蛋鸡设施养殖智能监测管理系统运行于多地不同蛋鸡养殖场，使用物联网、云存储、异步传输等多项技术构建，向用户提供数据的智能处理、实时展示。系统综合物联网[18]及云平台架构[19]，系统总体架构如图 2 所示。自下而上包括感知层、传输层、基础设施层、中间层及应用层。

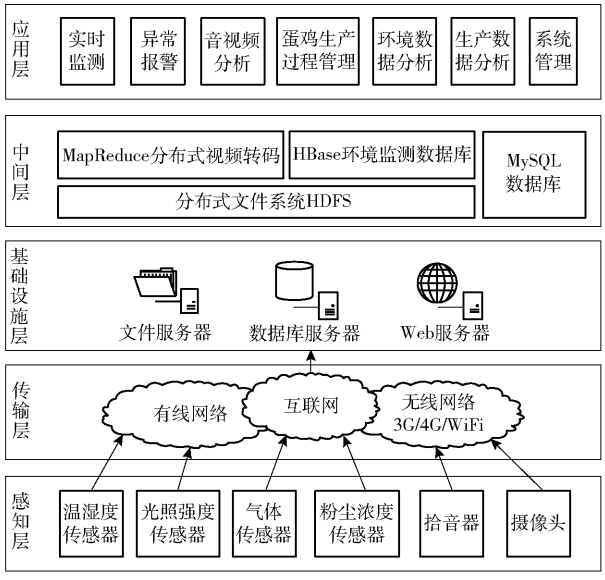


图 2　系统总体架构

Fig. 2　Overall architecture of system

感知层集成了摄像头、拾音器及各种环境传感器，负责进行数据采集。采集到的数据由传输层通过有线网络及无线网络传入基础设施层。基础设施层提供底层的服务器等硬件资源，负责存储数据并向中间层提供数据支撑。中间层负责数据的统一规划，进而分布式存储数据。同时，中间层提供了基于 MapReduce 的分布式转码模块，为异构视频数据的展示下载提供了支持。应用层则向用户提供环境的实时监测、音视频分析、生产过程管理、数据图表分析等相关用户感兴趣的服务。

2.2　系统总体拓扑

系统的数据来源为各地不同养殖场中由生产养殖人员填写的生产流程数据，以及各鸡舍部署的环境传感器采集的环境数据，如温湿度、光照强度、二氧化碳浓度、二氧化硫浓度、氨气浓度等；通过拾音器录制的蛋鸡舍舍内音频数据；使用摄像头拍摄的蛋鸡舍监控图像、视频数据。数据经采集后，暂存于现场服务器中。在现场服务器中部署监控程序监控

数据库及采集到的音视频及图像文件。采集到数据后,Kafka 集群将更改的数据按类型分为环境(Environment)、生产(Production)、音频(Audio)、视频(Video)、图像(Image) 5 个主题发送给数据中心机房的远端服务器集群,数据中心在接收到数据后进行解析,将环境、生产数据存入数据库,将音视频及图像存入 HDFS 分布式文件系统中,并更新其在数据库中的文件路径[20]。用户可通过 Web 网页端及手机 APP 对数据实时访问。系统总体拓扑结构如图 3 所示。

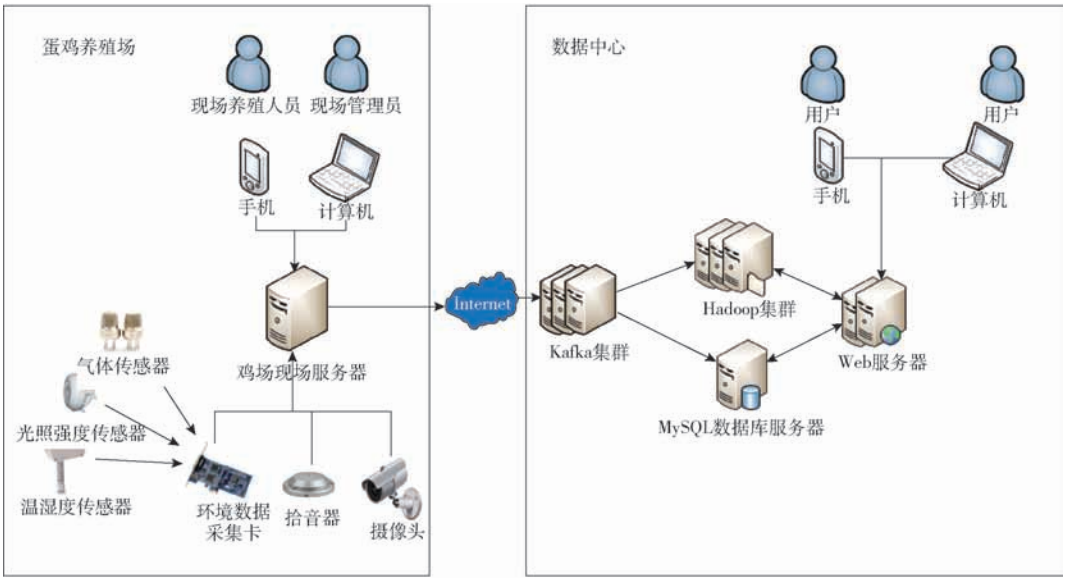


图 3 系统总体拓扑结构

Fig. 3 Overall topology of system

2.3 系统功能设计

根据对不同蛋鸡场的调研、综合分析,设计系统主要功能模块包括实时信息、历史信息、基础设施、生产过程管理、统计分析、环境警报、系统管理 7 个模块。具体功能模块如图 4 所示。

(1)实时信息模块:主要负责展示鸡舍的舍内外环境、音视频等实时信息。监测数据通过环境监测传感器、拾音器、摄像头进行采集,并通过部署于现场的采集程序存入数据库中,生产数据由生产人员填报。实时数据经过分析处理,通过 Web 端与手机 APP 实时向用户展示,便于用户了解现场情况。

(2)历史信息模块:主要包括历史音视频及图像数据的展示。历史数据用于图像分割、视频追踪等,将原始数据与经算法处理后的数据存入系统,用户可下载查看。

(3)基础设施模块:由于各地鸡场情况各不相同,设置基础设施模块。用于存储和展示各地蛋鸡场的鸡场、鸡舍及舍内外采集点的详细信息,由鸡场管理人员填写,便于根据不同鸡场做出不同的分析管理。

(4)生产过程管理模块:主要负责对生产过程中自动采集、用户填写的数据进行统一记录管理。包括日报表、日盈亏表、水电消耗、蛋鸡转群、免疫记录、药品投入以及各项检测。由鸡场生产人员录入,管理人员进行审核,提高了工作效率,同时积累数据为后续分析提供支持。

(5)统计分析模块:主要负责对收集到的生产数据、环境数据进行分析,允许相关人员通过时间、鸡舍号等进行查询,并通过曲线图表进行个性化展示,便于鸡场管理人员直观地了解鸡舍的环境变化

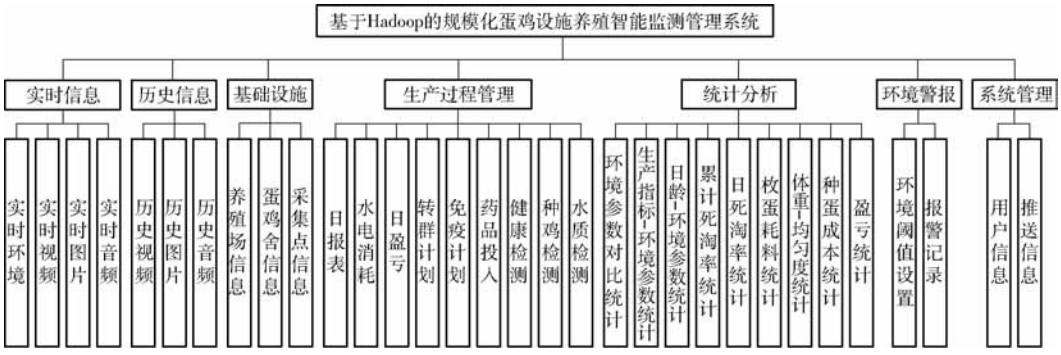


图 4 系统功能模块

Fig. 4 Function module of system

趋势、生产资料消耗以及生产盈亏趋势，指导生产。

(6)环境警报模块：主要负责对环境参数进行警报。管理人员可根据不同鸡舍设置不同环境阈值，采集到的数据经过处理与阈值进行实时对比，超出阈值的系统将通过手机 APP 推送警报信息，便于管理人员及时查看舍内情况，做出应对。

(7)系统管理模块：主要负责对系统用户、系统推送信息查看管理。用户管理即管理员可以添加、删除用户，并对用户权限进行管理，控制不同类别的用户访问不同模块。推送信息管理可以查看预警信息推送情况，同时提供推送消息页面方便通过手机 APP 对指定用户或批量用户进行消息推送。

3　系统数据存储设计与实现

3.1　系统数据存储设计

规模化蛋鸡场的现场数据分别存储于现场服务器与数据中心的集群服务器中。

现场服务器存储近期的数据，数据量较小，日常所产生的结构化数据存储于 MySQL 数据库中。非结构化数据存储于现场服务器的本地文件系统中。

现场服务器通过 Kafka 集群将数据打包发送给数据中心，存储于 MySQL 数据库、Hadoop 集群及搭载在其上的 HBase 数据库中。针对不同数据类型，数据中心对结构化及非结构化数据的存储采用不同设计。

结构化数据指蛋鸡养殖中的生产数据与环境监测数据。由于生产数据的数据量不大，传统的结构化数据库即可满足其存储与检索，因此，生产数据存储在 MySQL 数据库中。而环境监测数据每分钟采集一次，每个采集点每天约采集 1 500 条数据，每日鸡场所产生的环境数据达上万条，日积月累，传统的结构化数据库在查询数据时需要较长时间，无法满足实时查询需求。由于 HBase 适合存储处理大规模数据，并可进行快速随机访问，而 MySQL 在存储、查询数据量较小时实时访问速度优势明显。因此，选择 MySQL + HBase 作为环境数据的存储数据库。

非结构化数据主要包括养殖过程中产生的监控图像数据及音视频数据，每日每间鸡舍所产生的数据超过 100 GB。而 HDFS 文件系统因其具有高效的数据读取及多副本存放模式，适合存储海量数据，为图像数据及音视频数据的存储提供了方便，同时 HDFS 自身具有副本存储及机架感知策略，可以保证数据的安全性。

通过分析，现场服务器的数据存储采用 MySQL 数据库及本地文件系统。数据中心系统选择采用 Hadoop 框架，接收各个数据采集节点所采集的数据，分类后存储于 MySQL、HBase 数据库及 HDFS 文件系统中，最终用户可通过 Web 页面端及手机 APP 对数据中心数据查询分析。

3.2　基于 MySQL + HBase 的环境监测数据存储设计

系统在环境监测数据存储时选择 MySQL + HBase 的混合存储模式。MySQL 存储近一段时期的数据，HBase 存储采集到的所有数据。当 MySQL 存储数据超过阈值后，删除历史记录。

当数据中心集群接收到来自 Kafka 集群的消息后，获取主题为环境(Environment)的消息，将消息解析后获取环境数据，同时插入 MySQL 及 HBase 数据库中，查询不同的数据库均可获得实时数据，确保了数据查询的实时性。

在系统中，采集的环境数据包括采集点信息(场区号、鸡舍号、采集点号)、采集时间及各种环境参数值，对于环境数据的查询多为根据具体采集点及采集时间进行单项或多项参数值的范围查询。根据需求，在 MySQL 数据库中设计并实现二维数据表如表 1 所示。

表 1　MySQL 环境监测数据存储表

Tab. 1　MySQL environment monitoring data storage

字段	数据类型	字段解释
id	int	环境数据编号
houseid	varchar	鸡舍编号
pickpointid	varchar	采集点编号
picktime	datetime	采集时间
intemperature	float	舍内温度
inhumidity	float	舍内湿度
inradiation	float	舍内光照强度
incardioxide	float	舍内二氧化碳浓度
inammonia	float	舍内氨气浓度
inpressure	float	舍内气压
inwindspeed	float	舍内气流
indust	float	舍内粉尘

为保证查询效率，本系统设置 MySQL 数据存储时间为 30 d，每个月数据记录约为 30 万条。30 d 之前的数据则存入 HBase 中。

HBase 提供了基于 RowKey 的单行扫描查询、基于 RowKey 的范围扫描查询和全表扫描查询。通过 RowKey 查询效率较高，而通过列族进行查询则会进行全表扫描，效率低下。因此，设计良好的 RowKey 对查询性能影响极大。

HBase 中的表由 Region 存储，每个 Region 由 StartKey 和 EndKey 表示其范围。而 HBase 初建表默认生成一个 Region，所有数据均写入该 Region 中，直至数据不断增加至一定阈值后进行分裂

(Region-split),Region-split 会消耗较多时间和资源,所以可以对 HBase 表进行预分区,提前创建多个空 Region,确定其 StartKey 和 EndKey,从而减少 Region-split,降低资源消耗。提前预分区处理的 Region 中,仅有一个 Region 无上界,不存在 Endkey。新数据会根据其 RowKey 所在范围被插入不同 Region 中。由于 HBase 中 Rowkey 按字典顺序递增,若设计的 RowKey 同样按照字典顺序增加,新插入的数据会被不断写入无上界 Region 中,造成写热点问题,同时预分区也无法得到有效使用。为解决这一问题,在设计 RowKey 时应避免其按字母顺序递增。

根据 HBase 的特点,若按照 MySQL 数据库中将鸡舍号、采集点编号、采集时间等分别作为列族存储,则在查询一段时间的监控数据时需要进行全表扫描,查询耗时较长,效率较低。因此,选择将“场区号 + 鸡舍号 + 采集点号_采集时间”作为 RowKey,并保证 RowKey 中各个字段长度一定,以提高查询速度。同时,由于场区号采用字母 + 数字的格式,将场区号写在开端也可避免写入数据时 RowKey 的顺序递增问题。环境参数中的温度、湿度、光照强度、二氧化碳浓度、氨气浓度、粉尘浓度、气流、气压等作为不同的列族,查询时即可避免加载无关数据,从而提升速度。在建表前,根据 RowKey 中场区号及鸡舍号作为预分区边界值,调用 HBase 提供的 API 中的 HBaseAdmin. creatTable (tableDescriptor, splitKeys) 即可进行预分区建表。同时,由于 HBase 进行 Region-split 时旧 Region 下线,占用大量 IO 资源,而 HBase 默认的自动 Region-split 会在 Region 数据量到达阈值时进行,无法控制进行 Region-split 的时间。因此,系统选择关闭自动 Region-split,并设置在 IO 资源占用较少的固定时间执行 RegionSplitter 类的滚动拆分,从而降低系统在高 IO 需求时由于 Region-split 占用资源而导致的时间消耗。

查询数据时,如果需要查询场区号为 AH01,鸡舍号为 FD01,采集点号为 01,采集时间为 2017 年 12 月,可以设置起始的 RowKey 为“AH01FD0101_20171201000000”,结束的 RowKey 为“AH01FD0101_20171231246060”,HBase 会根据 RowKey 进行范围扫描查询,从而使查询速度得到保证。

由于 MySQL 在数据量较小时实时查询能力较强,因此,MySQL 负责近 30 d 数据的查询。而查询数据量较大或需要查询历史数据时,则选择 HBase 进行查询。用户查询数据时需要输入查询起止时间,系统判断输入的起止时间是否在近 30 d 内,若属于最近 30 d,则在 MySQL 数据库中进行查询。若起止时间超出最近 30 d,则在 HBase 数据库中查询。通过 Java 语言编程,可实现对 MySQL 及 HBase 的访问控制,对 MySQL 的查询通过对 JDBC 进行控制,对 HBase 的查询通过 HBase 的 API 进行操作。

3.3 实验与结果分析

系统远端数据中心位于中国农业大学网络中心,系统硬件环境为 4 台服务器的 Hadoop 集群,采用该集群进行分析实验。其中一台主节点,3 台从节点。每个节点 CPU 均为 Intel E5 - 2609,主频 1.90 GHz,内存 32 GB,8 TB 硬盘,操作系统为 Ubuntu 16.04,Hadoop 版本为 Hadoop 2.6.5,HBase 版本为 HBase 1.1.12。

系统经过前期运行,已经采集到超过 5 000 万条数据。将数据同时存入 MySQL 及 HBase 数据库中,查询数据结果及分析如下:

(1)在 MySQL 及 HBase 中存入不同存储规模的相同数据,选取不同时间段、不同蛋鸡舍及不同采集点,对 MySQL 及 HBase 同时进行查询数据量为存储规模 20%、40%、60%、80%、100% 的 5 次查询,将查询结果进行平均,得到查询时间如表 2 所示,绘制存储规模为 25 万至 100 万条的折线图,如图 5 所示。

表 2 不同存储规模 MySQL 及 HBase 查询耗时

Tab. 2 MySQL and HBase query time for different storage scales ms

存储规模/条	MySQL 无索引查询	MySQL 带索引查询	HBase 查询
1.0×10^5	1 927	1 498	2 216
2.5×10^5	2 593	1 936	2 295
5.0×10^5	4 517	2 579	2 308
7.5×10^5	5 433	3 476	2 319
1.0×10^6	6 869	4 423	2 332
5.0×10^6	10 031	4 423	2 358
1.0×10^7	16 800	4 442	2 369
2.5×10^7	30 173	4 483	2 593
5.0×10^7	61 171	4 758	2 910

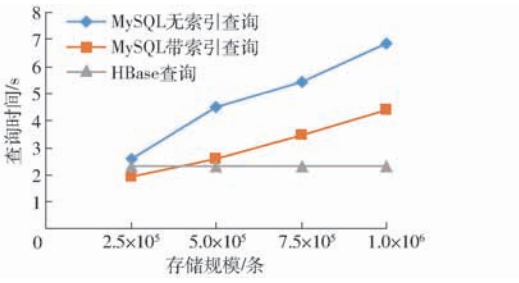


图 5 不同存储规模查询耗时对比

Fig. 5 Comparison of MySQL and HBase query time for different storage scales

由表 2、图 5 可知，随着存储规模的增加，MySQL 数据库查询耗时也不断增加。存储规模在 500 万条时，无索引查询已经超过 10 s，不再适合实时查询。带索引 MySQL 查询在数据规模较小时表现出了良好的查询性能，而当存储数据量接近 50 万条时，其查询速度被 HBase 赶超。HBase 查询时间随存储规模增加也逐渐增加，但其增长较为缓和。因此，在 MySQL 中存储约 30 万条记录较为合理。

（2）选择不同查询数据量，对存储规模为 5 000 万条的 MySQL 及 HBase 分别随机进行查询，每个数据量查询 5 次并将结果取平均，得到查询时间如表 3 所示，将 10 万至 50 万条查询量耗时绘制折线图，如图 6 所示。

表 3　不同查询规模 MySQL 及 HBase 查询耗时

Tab. 3　MySQL and HBase query time for different query scales　　ms

查询数据量/条	MySQL 无索引查询	MySQL 带索引查询	HBase 查询
1×10^3	53 515	307	2 012
5×10^3	54 153	416	2 022
1×10^4	53 783	443	2 035
5×10^4	55 350	1 053	2 359
1×10^5	56 920	2 214	2 693
5×10^5	64 776	10 971	3 602
1×10^6	145 283	24 176	4 096
5×10^6		85 505	4 644

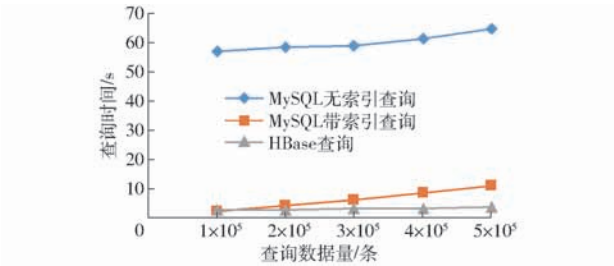


图 6　不同查询量耗时对比

Fig. 6　Comparison of MySQL and HBase query time for different query scales

由图 6、表 3 可知，在存储数据量达到 5 000 万时，无索引 MySQL 即使在小数据量查询时，时间也已超过 50 s，在查询 500 万条数据时发生查询连接超时。在小规模数据查询时，由于 HBase 查询需要将数据加载入内存，总体查询时间比带索引 MySQL 查询长。随着查询量的增加，带索引 MySQL 查询及 HBase 查询耗时均不断增加，但 HBase 查询时间增加较缓。而在查询数据量超过 20 万条时，HBase 查询取得优势，查询性能高于带索引 MySQL。因此，选择使用 MySQL 存储近期数据，HBase 存储历史数据，查询数据时根据不同时间段选择查询不同数据库较合理。

4　视频分布式转码设计与实现

4.1　基于 MapReduce 视频分布式转码设计与实现

系统需要对多个异地鸡场进行统一管理，因此需要一致的数据类型，完成对视频数据的分布式存储。由于各地鸡场采用不同种类摄像头，所采集到的视频类型并不一致，包括 H. 264 及 MP4，而 H. 264 格式视频所占比重较大。H. 264 格式的视频无法通过通用播放器播放，需要先使用专用解码器进行解码，不利于集中展示[21]。另一方面也无法直接实现数据的分析，增加了后续视频算法处理的复杂程度。因此，需要对视频提前转码为 MP4，使格式一致。运行在各地鸡场的系统根据监控需求，视频数据每 30 min 存储一次，每份视频为 1. 5 GB。

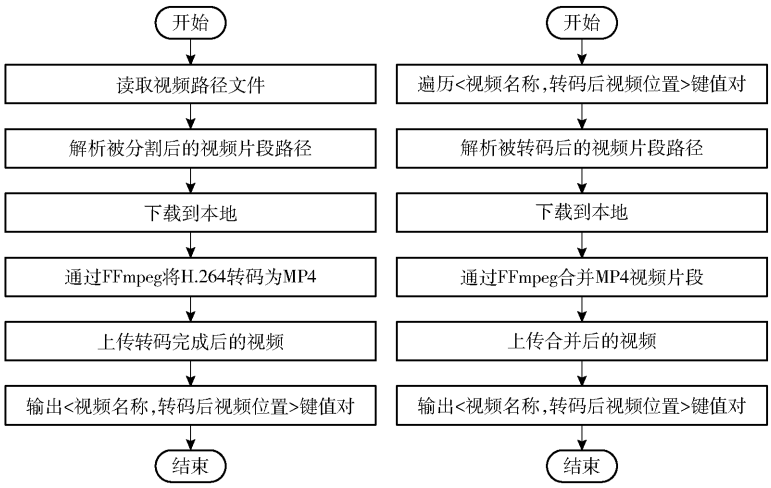
目前视频的转码模式一般分为单点、分布式、面向移动端以及基于云的转码[22]。单点转码使用单个服务器，实现简单，但速度慢，效率较低，大规模转码限制较大。分布式转码采用多台服务器进行转码，转码完成后再进行合并，减少了转码整体时间，但实现复杂。面向移动端的转码降低了视频分辨率及码率，应用性较强，但降低了转码后视频质量，会对之后的视频追踪等算法造成影响，并不适合本系统。基于云的转码利用企业云转码，减少了需要的资源，但稳定性较难保证。因此，需要设计实现简单、高效、不损失视频质量且较为稳定的视频转码模块。

视频数据存储于容错性较高、稳定性良好的 Hadoop 文件系统 HDFS 上。而 Hadoop 的另一核心 MapReduce 适合对数据进行并行处理。MapReduce 使用简单，Map 函数并行处理数据，Reduce 函数规约数据，设计良好的 Map 及 Reduce 函数，可以实现视频的分布式转码。

视频数据在 HDFS 上通过 Block 进行存储，大于 Block 存储量的视频上传后被分为多个块由不同 Block 存储。由于视频由帧组成，Block 上存储的视频可能会出现首尾帧不完整，而帧与帧之间具有关联性，若直接使用 Map 函数通过操作视频帧处理各 Block 上存储的视频块，可能会由于帧的不完整以及与前一帧关联而造成处理错误。因此，为充分利用 MapReduce 的并行处理能力，需要提前对视频数据进行分割。视频分割可以通过视频处理工具 FFmpeg 完成。

FFmpeg[23] 是一个可以运行于 Linux 及 Windows 操作系统上的多媒体处理工具，可以完成音视频的格式转换及分割合并。FFmpeg 支持 mpeg、mpeg4、flv、div 等多种解码编码。

经过 FFmpeg 分割后的视频分别由 HDFS 的不同 Block 存储,同时存储带有原始文件及目标文件路径的文件。存储完成后启动分布式转码程序,执行 Map 及 Reduce 函数。Map 函数的输入为〈文件名,文件路径〉,Map 函数首先解析文件路径并从 HDFS 中下载相应视频数据,然后调用 FFmpeg 进行转码,完成后将视频存入 HDFS 文件系统中。Map 函数的输出为〈转码后视频片段的文件名,转码后视频片段的文件路径〉,被传递给 Reduce 函数进行规约处理。Reduce 函数通过 Map 函数传递的键值对解析出转码完成的视频文件路径并下载视频数据至本地,然后调用 FFmpeg 进行合并,合并后再将视频数据写入 HDFS 中。Map 及 Reduce 函数流程图如图 7 所示。



(a) Map函数 (b) Reduce函数

图 7 Map 函数、Reduce 函数流程图

Fig. 7 Flow chart of Map function and Reduce function

分布式转码模块利用数据中心已搭建的 MySQL 服务器及 Hadoop 集群实现,数据中心存储转码视频具体流程图如图 8 所示。

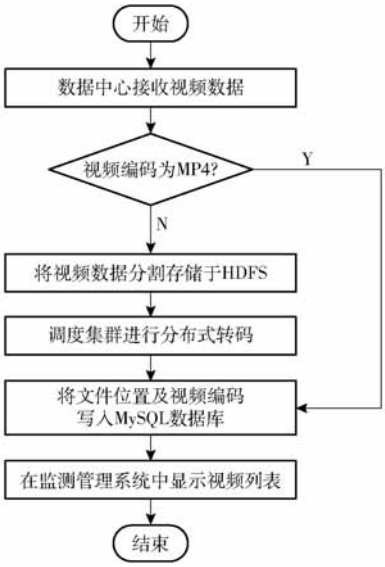


图 8 数据中心视频存储转码流程图

Fig. 8 Flow chart of transcoding

当数据中心接收到来自 Kafka 集群的视频后检查视频格式,若为 MP4 则直接存储于 HDFS 中并将视频编码和存储路径等详细信息写入 MySQL 数据库的视频表中。若格式为 H. 264 则将视频分割为视频段,然后将视频段存储于 HDFS 中。存储完成后开始调用分布式视频转码模块,由 Hadoop 集群中的 NameNode 负责调度,执行 Map 函数转码及 Reduce 函数合并,完成后再将视频上传至 HDFS,并将信息写入 MySQL 数据库。此时系统中历史视频列表即可显示该视频。用户请求视频时,点击历史视频列表中的视频名称,即可获取视频位置,从而播放或下载视频。

4.2 分布式转码性能测试

实验在已搭建好的 Hadoop 集群上进行。单点转码使用与集群中节点相同配置的服务器,转码大小为 1.5 GB、格式为 H. 264 的视频,所需时间为 193.257 s。对不同整体大小、不同分割大小的视频进行转码,分析如下:

(1)将 1.5 GB 的视频分割为不同大小的片段,在集群中进行分布式转码,分割片段大小及转码消耗的时间如图 9 所示。

由图 9 可知,转码消耗时间随分割视频段大小

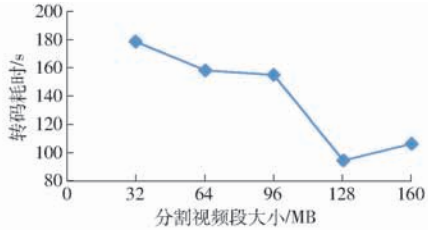


图 9 1.5 GB 视频不同分割大小分布式转码消耗时间

Fig. 9 Distributed transcoding time of 1.5 GB video with different segmenting sizes

增加不断降低，直至分割为与 HDFS 的 Block 大小相同的 128 MB 时，转码时间最少，为 94.73 s，效率较高，远小于单机转码时间。而在分割视频段大于 128 MB 后，转码时间开始增加。

(2)选择以转码最快的 128 MB 作为视频分割大小，不同大小的视频单机转码耗时及分布式转码耗时如图 10 所示。

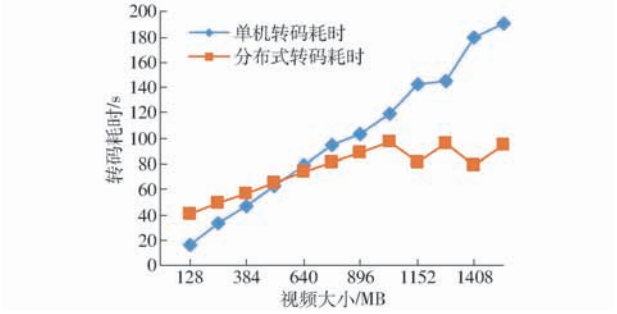


图 10　以 128 MB 分割不同大小视频转码时间

Fig. 10　Transcoding time of different sizes of video with 128 MB segmentation

由图 10 分析可知，单机转码时间随着视频量增加而增加。在视频文件较小时，分布式转码由于需要进行文件的上传、下载及集群间网络通信，时间比单机转码长，不具优势，当视频文件大于 512 MB 后，分布式转码时间开始低于单机转码时间。当视频文件达到 1.4 GB 时，分布式转码耗时达到最低，节约了 56% 的转码时间。在文件大小为 1.5 GB 时，节约时间为 50%。因此，可以考虑在视频存储时适量降低时长以减小视频量，从而达到最佳转码速度。

5　系统应用

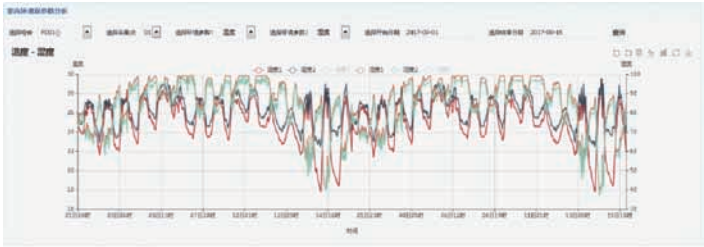
系统 Web 端及手机 APP 客户端已在中国农业大学上庄试验站、德清源延庆生态园、德清源黄山生态园投入使用，中国农业大学网络中心作为数据中心对各养殖场数据进行统一管理。用户通过访问 Web 端访问系统，首先点击首页的标注点选择不同养殖场，如图 11a 所示。各养殖场展示养殖场内各鸡舍信息，包括当日的存栏数、死淘数、产蛋量、耗水耗料及实时温度，如图 11b 所示，点击下方各菜单按钮，可查看舍内外环境数据及实时视频。

(a) Web系统首页　　(b) 养殖场首页

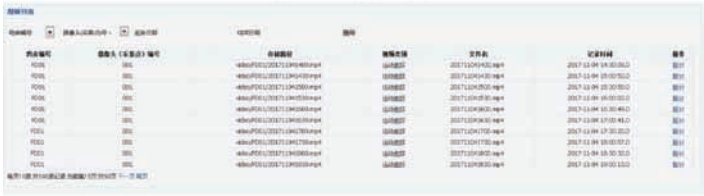
图 11　Web 端系统界面

Fig. 11　System interface of Web terminal

用户进入系统后台后，可对统计分析数据及转换后的视频进行查看，如图 12a、12b 所示。用户通



(a) 室内环境参数分析



(b) 转换后视频列表

(c) 手机APP登录页面　(d) 手机APP实时环境页面　(e) 手机APP管理页面

图 12　系统功能界面

Fig. 12　System application interface

过手机 APP 访问系统,首先输入用户名、密码、类别,系统确认无误后即可登录。登录后可查看实时环境数据,并对生产过程进行管理,如图 12c、12d、12e 所示。

蛋鸡场的养殖人员及管理人员可以通过 Web 端和手机 APP 查看各地蛋鸡舍的实时环境、生产信息、实时音视频数据、统计分析数据,便于及时发现问题、做出决策,从而降低企业风险,提高动物福利。

6 结论

(1)该智能监测管理系统实现了对蛋鸡生产养殖所产生的海量数据的高效存储:采用 MySQL + HBase 数据库的混合模式,保证了数据存储和查询的实时性,满足大规模环境监测数据的查询需求。

(2)设计实现了基于 MapReduce 的分布式监控视频转码模块,将其嵌入系统中,视频传输完成后即可进行转码,用户无需关心各地摄像头的详细信息即可获得统一格式的监控视频。实验表明其效率高于单机转码,可为后续基于视频的算法研究提供良好支持。

(3)该系统对蛋鸡养殖生产过程进行了数据分析,同时提供了多项功能,可以协助生产养殖人员实时、全方位监控生产过程,对规模化蛋鸡场积累数据、分析决策具有重要意义。

参 考 文 献

1 程学旗,靳小龙,王元卓,等. 大数据系统和分析技术综述[J]. 软件学报,2014,25(9):1889 - 1908.
CHENG Xueqi, JIN Xiaolong, WANG Yuanzhuo, et al. Survey on big data system and analytic technology[J]. Journal of Software, 2014,25(9):1889 - 1908. (in Chinese)

2 张浩然,李中良,邹腾飞,等. 农业大数据综述[J]. 计算机科学,2014,41(增刊 2):387 - 392.
ZHANG Haoran, LI Zhongliang, ZOU Tengfei, et al. Overview of agriculture big data research[J]. Computer Science, 2014, 41(Supp. 2):387 - 392. (in Chinese)

3 朱宁,秦富. 蛋鸡产业发展的国际趋势及中国展望[J]. 中国家禽,2016,38(20):1 - 5.

4 葛文杰,赵春江. 农业物联网研究与应用现状及发展对策研究[J/OL]. 农业机械学报,2014,45(7):222 - 230. http://www.j-csam.org/jcsam/ch/reader/view_abstract.aspx? flag = 1&file_no = 20140735&journal_id = jcsam. DOI:10.6041/j.issn.1000-1298.2014.07.035.
GE Wenjie, ZHAO Chunjiang. State-of-the-art and developing strategies of agricultural internet of things[J/OL]. Transactions of the Chinese Society for Agricultural Machinery,2014,45(7):222 - 230. (in Chinese)

5 孟超英,王佳,陈红茜,等. 基于分布式对象的蛋鸡舍设施养殖数字化智能监测系统[J/OL]. 农业机械学报,2017,48(10):292 - 299. http://www.j-csam.org/jcsam/ch/reader/view_abstract.aspx? flag = 1&file_no = 20171037&journal_id = jcsam. DOI:10.6041/j.issn.1000-1298.2017.10.037.
MENG Chaoying, WANG Jia, CHEN Hongqian, et al. Intelligent monitoring system based on distributed object for layer house[J/OL]. Transactions of the Chinese Society for Agricultural Machinery,2017,48(10):292 - 299. (in Chinese)

6 CHEN Hongqian, XIN Hongwei, TENG Guanghui, et al. Cloud-based data management system for automatic real-time data acquisition from large-scale laying-hen farms[J]. International Journal of Agricultural and Biological Engineering, 2016, 9(4): 106 - 115.

7 WANG Cheng, CHEN Hongqian, ZHANG Xuebin, et al. Evaluation of a laying-hen tracking algorithm based on a hybrid support vector machine[J]. Journal of Animal Science and Biotechnology,2017,8(1): 226 - 235.

8 夏靖波,韦泽鲲,付凯,等. 云计算中 Hadoop 技术研究与应用综述[J]. 计算机科学,2016,43(11):6 - 11,48.
XIA Jingbo, WEI Zekun, FU Kai, et al. Review of research and application on Hadoop in cloud computing[J]. Computer Science, 2016,43(11):6 - 11,48. (in Chinese)

9 陈吉荣,乐嘉锦. 基于 Hadoop 生态系统的大数据解决方案综述[J]. 计算机工程与科学,2013,35(10):25 - 35.
CHEN Jirong, LE Jiajin. Reviewing the big data solution based on Hadoop ecosystem[J]. Computer Engineering and Science, 2013, 35(10):25 - 35. (in Chinese)

10 WHITE T. Hadoop: the definitive guide[M]. Beijing:O'Reilly Media, 2009.

11 LAM C. Hadoop in action[M]. 北京:人民邮电出版社,2011.

12 郝树魁. Hadoop HDFS 和 MapReduce 架构浅析[J]. 邮电设计技术,2012(7):37 - 42.
HAO Shukui. Brief analysis of the architecture of Hadoop HDFS and MapReduce[J]. Designing Techniques of Posts and Telecommunications,2012(7):37 - 42. (in Chinese)

13 王永洲. 基于 HDFS 的存储技术的研究[D]. 南京:南京邮电大学,2013.

14 KONSTANTIN S, KUANG Hairong, RADIA S, et al. The Hadoop distributed file system[C]// Proceedings of the 26th IEEE Symposium on Massive Storage Systems and Technologies (MSST 10). Piscataway: IEEE Press, 2010:1 - 10.

15 DEAN J, GHEMAWAT S. MapReduce: simplified data processing on large clusters[J]. Communications of the ACM, 2008,

51(1):107 - 113.

16 LI Chongxin. Transforming relational database into HBase: a case study[C]// Proceedings 2010 IEEE International Conference on Software Engineering and Service Sciences, 2010.

17 周利珍,陈庆奎. 基于 HBase 的农业无线传感信息存储系统[J]. 计算机系统应用,2012,21(8):6 - 9,26.
ZHOU Lizhen, CHEN Qingkui. HBase-based storage system for wireless sensor information of agriculture[J]. Computer Systems and Applications, 2012, 21(8):6 - 9, 26. (in Chinese)

18 张宇,张可辉,严小青. 农业物联网架构、应用及社会经济效益[J]. 农机化研究,2014,36(10):1 - 5,67.

19 罗军舟,金嘉晖,宋爱波,等. 云计算:体系架构与关键技术[J]. 通信学报,2011,32(7):3 - 21.
LUO Junzhou, JIN Jiahui, SONG Aibo, et al. Cloud computing: architecture and key technologies[J]. Journal on Communications, 2011, 32(7):3 - 21. (in Chinese)

20 陈红茜,滕光辉,邱小彬,等. 基于分布式流式计算的蛋鸡养殖实时监测与预警系统[J/OL]. 农业机械学报,2016,47(1):252 - 259. http://www.j-csam.org/jcsam/ch/reader/view_abstract.aspx? flag = 1&file_no = 20160134&journal_id = jcsam. DOI:10.6041/j.issn.1000-1298.2016.01.034.
CHEN Hongqian, TENG Guanghui, QIU Xiaobin, et al. Real-time monitoring and early warning system based on stream computing for laying hens raise[J/OL]. Transactions of the Chinese Society for Agricultural Machinery,2016,47(1):252 - 259. (in Chinese)

21 周娅. H.264 解码系统设计与关键算法研究[D]. 武汉:华中科技大学,2011.

22 李亚飞. 分布式视频转码系统的设计与实现[D]. 哈尔滨:哈尔滨工业大学,2014.

23 任严,韩臻,刘丽. 基于 FFMPEG 的视频转换与发布系统[J]. 计算机工程与设计,2007,28(20):4962 - 4963,4967.
REN Yan, HAN Zhen, LIU Li. System of convert and distribute of videos based on FFMPEG[J]. Computer Engineering and Design, 2007,28(20):4962 - 4963,4967. (in Chinese)