

DOI:10.3969/j.issn.1001-4551.2020.12.024

基于并行计算的抛光轴表面缺陷检测研究*

姜庆胜,李研彪,计时鸣*

(浙江工业大学 机械工程学院,浙江 杭州 310023)

摘要:针对在生产实践中抛光轴表面缺陷检测的实时性和缺陷分类需求问题,对抛光轴表面缺陷检测系统、缺陷图像采集、图像特征、传统检测方法等方面进行了研究,归纳了轴表面缺陷检测的处理方法。提出了两种基于并行计算的检测系统,即基于云的 Spark 多机并行的分布式计算机集群检测系统和单机多核的 CUDA 并行处理检测系统;通过 Spark 架构多机分布式并行计算机理的分析,以及结合 CUDA 单机多核并行运算原理,从数学角度进行了并行化处理,最后进行了相应的实验。研究结果表明:基于 CUDA 的单机多核的并行处理检测系统适合实时检测,基于 Spark 的多机分布式并行检测系统不适合实时检测系统,但适合大数据处理系统。

关键词:图像处理;Spark;分布式并行计算;图像处理单元;CUDA

中图分类号:TH133.2;TP391.4

文献标识码:A

文章编号:1001-4551(2020)12-1540-07

Surface defect detection of polishing shaft based on parallel computation

JIANG Qing-sheng, LI Yan-biao, JI Shi-ming

(College of Mechanical Engineering, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract: Aiming at the problems of real-time defect detection and defect classification requirements for polishing shaft surface defects in production practice, the surface defect detection system for polishing shafts, defect image acquisition, image features, traditional detection methods, etc. were studied, and the treatment methods of shaft surface defect detection were summarized. Two detection systems based on parallel computing were proposed, i. e. cloud-based Spark multi-machine parallel distributed computer cluster detection system and single-machine multi-core CUDA parallel processing detection system. Through the analysis of Spark architecture multi-machine distributed parallel computer theory, and analysis of the principle of mathematical parallelization combined with CUDA single-machine multi-core parallel operation theory, the experiments were conducted. The results indicate that the single-machine multi-core parallel processing detection system based on CUDA is suitable for real-time detection, and the multi-machine distributed parallel detection system based on Spark is not suitable for real-time detection systems, but is suitable for big data processing systems.

Key words: image processing; Spark; distributed parallel computing; graphic process unit(GPU); CUDA

0 引 言

轴是机械行业普遍使用的零部件,其表面缺陷对其的使用性能和寿命有重要影响。一般情况下,轴的表面缺陷检测依靠工人肉眼观察完成,人工判断缺陷

并剔除次品,检验人员的劳动强度大,长期工作对眼睛有伤害,并且检测结果一致性差,易发生漏检。而利用机器视觉自动检测轴的表面缺陷,可有效提高检测效率、检测质量,保护检测人员健康,是未来技术的发展趋势。

收稿日期:2020-04-14

基金项目:国家自然科学基金资助项目(51575494);浙江省自然科学基金重点资助项目(LZ14E050001)

作者简介:姜庆胜(1968-),男,浙江衢州人,博士研究生,主要从事智能装备和自动化检测设备方面的研究。E-mail:zjjqs@163.com

通信联系人:计时鸣,男,教授,博士生导师。E-mail:jishiming@zjut.edu.cn

在轴的在线检测中,图像处理的速度是提高检测效率的重要因素之一。随着被处理的图像数量的增加,以及图像分辨率的不断提高,处理速度会变得越来越慢。由于计算机硬件系统的限制,会造成对复杂数据的处理速度的限制。根据摩尔定律^[1],CPU 的主频已经发展到了极限,无法仅靠提升硬件性能来提高运算速度。

近年来,超级计算的计算方法已应用到生产实践中。超级计算的最基本方法就是并行计算。并行计算的方法和硬件的结构有着密切的关系,但是其基本的原理和模型就是串行问题并行化。按其硬件形式可分为两类:(1)多机并行集群方式;(2)单机多核,例如 GPU 系统。

并行计算目的是为了加快运算速度,追寻并行架构的演变轨迹^[2],经历了向量、分布存储^[3]、共享存储、异构几个阶段。与之对应的程序设计语言有:面向向量的 VFortran、面向分布存储的 PVM 和 MPI、面向共享存储的 OpenMP 和面向异构的 OpenCL^[4]。当前应用更广泛的是 Nvidia 公司的使用 CUDA C 语言的显卡和专用计算卡。

Spark 是近几年最流行的也比较成熟的多机并行的分布式集群运算架构^[5-9],随着并行数量的增加,其运算速度明显加快。而对于基于面向异构的 CUDA 的单机多核并行计算也已经有了大量的研究和应用^[10-14],这种方式加速非常明显。另外其他的并行方式有 MPI^[15]、OpenMP^[16]等,但应用不多。

本文研究两种典型的并行计算方式,并提出一种将两种架构结合的方式,实现实时和大数据检测。

1 抛光轴表面缺陷检测

1.1 抛光轴表面缺陷检测系统

抛光轴表面缺陷检测系统不但需要实时识别出产品是否合格,而且也需要缺陷分析。

比较典型的轴表面缺陷检测系统如图 1 所示。

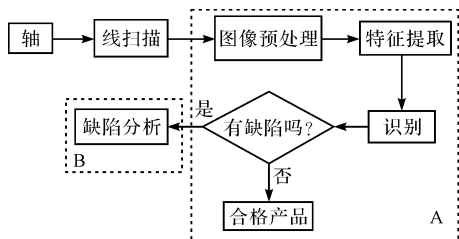


图 1 缺陷检测系统

1.2 图像采集

由于轴是属于有规则型旋转曲面,而且经过抛光,具有高光反射特性,对于这种具有镜面特点的曲面上的缺陷,当前比较流行的图像获取方式是线扫描。在线扫描装置中,相机采用线阵相机,光源采用线光源。为了防止由于镜面的高光反射,而拍摄不到缺陷,必须先确定相机和线光源的角度。同时,轴的转动时间也必须和拍摄时间保持一定的比例关系,否则缺陷会失真。

轴的线扫描装置如图 2 所示。

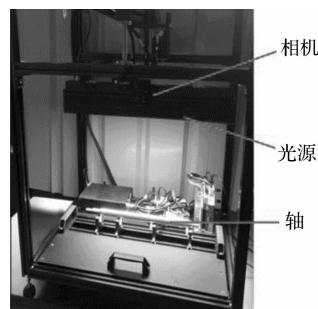


图 2 轴的线扫描装置

线扫描图像采集主要设备型号如表 1 所示。

表 1 图像采集主要设备型号

序号	名称	型号
1	线阵相机	LA-CM-16K05A(16384x1)
2	镜头	Mk-CPN-S 4.0/60
3	线光源	DH-LS600F-HX
4	图像采集卡	OR-Y4C0-XXM00

1.3 图像特征

根据被检测轴缺陷的尺度,某些情况下需要采集较高分辨率的图像。笔者研究的检测对象活塞轴,其长度为 200 mm ~400 mm,直径约 20 mm 左右,表面经抛光处理,表面最小缺陷的直径约 0.1 mm。根据缺陷尺度识别要求,采集图像的分辨率至少要达到 16 384 × 4 096,即 6.7×10^7 像素,这就意味着处理时需要比较长的时间。

活塞轴和部分缺陷图如图 3 所示。

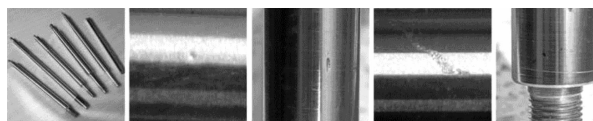


图 3 活塞轴和部分缺陷图

针对高分辨率图像,图像处理的耗时直接影响到

机器视觉自动检测系统效率。因此,要选择合适的图像处理平台和算法。

活塞轴表面图像具有像素多、被检测缺陷点小的特点,一般情况下 6.7×10^7 像素的图像中的缺陷是仅有数十个互相连通的像素团。根据统计分析,16 个像素以下的像素团应作为噪声滤除,而且生产要求在线识别是否合格时间要小于 1 s。

16 384 × 4 096 线扫描图像如图 4 所示。



图 4 16 384 × 4 096 线扫描图像

1.4 表面缺陷检测的处理方法

对于物体表面缺陷的检测,常用的处理方法有^[17]:统计法、频谱法和模型法。(1)统计法又可以分为直方图法^[18]、灰度共生矩阵法^[19]、自相关法^[20]、数学形态学法^[21]; (2)频谱法可以分为傅里叶特征法^[22]、Gabor 特征法^[23]和小波特征法^[24]; (3)模型法可以分为分形体法^[25]、随机场模型法^[26]和新模型法^[27]。

图 3 中,轴表面缺陷点一般是凹坑、压痕、磨损和长条状的划痕。当图像经过均值滤波之后,非缺陷点的像素明显高于某个阈值,所以只要能检测出图片中的阈值小于这个阈值的点数的总数占一定的比例,就可以判断该轴是否存在缺陷。

由于需要检测轴表面非常微小的缺陷,采集的单张图片至少要大于 6.7×10^7 像素,并行计算是最好的选择。并行计算可以分为:任务并行性和数据并行性。对于 Spark 多机分布并行计算适用于任务并行法,而对于单机多核的 GPU 适合使用数据并行法。

1.5 基于传统的像素逐次遍历的缺陷检测

为了便于对比,笔者采纳了基于像素的逐次遍历串行计算方法:采集灰度化图像,通过均值滤波去除 16 点以下的像素团干扰,再将图像的像素值转换为二维数组,对每个数组元素和选取的阈值进行对比,小于某阈值的就进行统计,全部依次对比完成后,得到总的小于该阈值的像数总数,如果该总数达到某个缺陷的值,就认为存在缺陷,一次性完成图像的特征提取和识别。

普通串行算法的缺陷检测系统如图 5 所示。

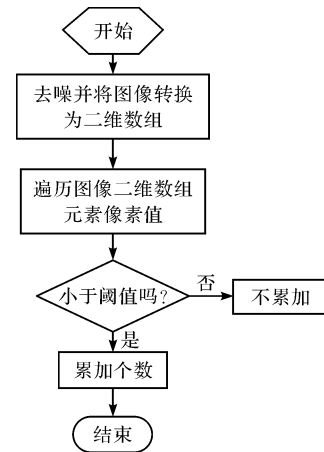


图 5 普通串行算法的缺陷检测系统

2 基于并行计算的检测系统

2.1 基于多机并行的 Spark 分布式并行计算系统

分布式并行处理就是多台具有独立运算功能的计算机并联运行。Spark 是在扩展了 Hadoop + MapReduce 计算模型基础上的通用集群计算平台,其特点是易用性;运算速度快,具有交互功能;通用性,在其基础上可以完成其他各种复杂运算。它可以完成原先需要多种不同的分布式平台的场景的运算。Spark 的 API 接口可以是基于 Python、Java、Scala 和 SQL 的程序,可以和其他大数据工具紧密配合。

Spark 的核心是 Spark core,实现任务调度、内存管理等,但真正的工作是对弹性分布式数据集 RDD 的 API 定义,RDD 主要的编程对象是分布在各个节点上可以并行操作的元素集合^[28]。

Spark 是个多任务的操作模式,就是将一个项目通过管理器分成很多个小任务,然后将这些小任务分到各个计算机节点去运算。

应用于轴表面缺陷的处理,主要是针对图像的预处理和像素的检索,利用了 Spark 的机器学习组件 MLib 中的函数^[29]。

基于 Spark 的缺陷检测系统如图 6 所示。

2.2 基于单机多核的 GPU 并行计算系统

由于其强大的图像并行处理能力,图像处理单元 GPU 被广泛应用于计算密集型计算。CUDA 是 Nvidia 公司在图形处理卡 GPU 上运行的应用软件,它可以让 C 或 C++ 代码在 GPU 上高效运行。

GPU 是一种异构架构^[30],其运行原理是主程序使用串行方式编程,在 CPU 中运行,但遇到程序中的计

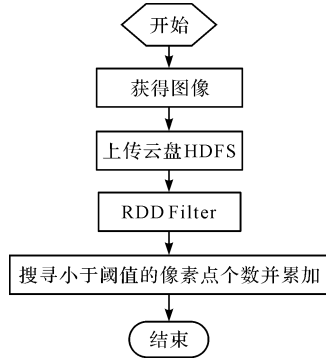


图 6 基于 Spark 的缺陷检测系统

算密集型部分,就把这些数据拷贝到 GPU 中,也就是显卡或计算卡,让 GPU 进行并行计算,当 GPU 计算完成之后,再把数据拷贝到 CPU 中。

为了具有对比性,此处仍然以串行遍历像素点的方法来确认是否有缺陷,不过采用了 CUDA 并行计算,并且使用了线程协助管理和共享内存的特性来优化求和计算^[31]。

并行计算数学原理为:

当 $n \gg m$,且 $i \leq n$ 时,则:

当 $P_i < T$ 时,令 $P_i = 1$;

当 $P_i > T$ 时,令 $P_i = 0$;

由此可得像素和为:

$$S_n = S_1 + S_2 + \dots + S_i + \dots + S_m \quad (1)$$

其中:

$$S_1 = P_1 + P_{m+1} + \dots + P_{n-m+1}$$

$$S_2 = P_2 + P_{m+2} + \dots + P_{n-m+2}$$

.....

$$S_i = P_i + P_{m+i} + \dots + P_{n-m+i}$$

.....

$$S_m = P_m + P_{m+m} + \dots + P_n \quad (2)$$

但是这样并行求和后,仍然会有 m 个数据求和,可以考虑再一次使用取半规约算法的数学方法进行并行求解。

根据式(1)有($S_j \in m$):

$$S_n = (S_1 + S_{\frac{m}{2}+1}) + (S_2 + S_{\frac{m}{2}+2}) + \dots + (S_i + S_{(\frac{m}{2}+j)}) + \dots + (S_{\frac{m}{2}} + S_m) \quad (j = 1, 2, 3, \dots) \quad (3)$$

此式可写为:

$$S_n^{(1)} = (S_1^{(0)} + S_{(\frac{m}{2}+1)}^{(0)}) + (S_2^{(0)} + S_{(\frac{m}{2}+2)}^{(0)}) + \dots + (S_j^{(0)} + S_{(\frac{m}{2}+j)}^{(0)}) + \dots + (S_{\frac{m}{2}}^{(0)} + S_m^{(0)}) \quad (4)$$

令:

$$\begin{aligned} S_1^{(1)} &= S_1^{(0)} + S_{(\frac{m}{2}+1)}^{(0)}, \\ S_2^{(1)} &= S_2^{(0)} + S_{(\frac{m}{2}+2)}^{(0)}, \\ &\dots, \\ S_j^{(1)} &= S_j^{(0)} + S_{(\frac{m}{2}+j)}^{(0)} \end{aligned} \quad (5)$$

可得和为:

$$S_n^{(2)} = (S_1^{(1)} + S_{(\frac{m}{4}+1)}^{(1)}) + (S_2^{(1)} + S_{(\frac{m}{4}+2)}^{(1)}) + \dots + (S_j^{(1)} + S_{(\frac{m}{4}+j)}^{(1)}) + \dots + (S_{\frac{m}{4}}^{(1)} + S_{\frac{m}{2}}^{(1)}) \quad (6)$$

再令:

$$\begin{aligned} S_1^{(2)} &= S_1^{(1)} + S_{(\frac{m}{4}+1)}^{(1)}, \\ S_2^{(2)} &= S_2^{(1)} + S_{(\frac{m}{4}+2)}^{(1)}, \\ &\dots, \\ S_j^{(2)} &= S_j^{(1)} + S_{(\frac{m}{4}+j)}^{(1)} \\ &\dots \end{aligned} \quad (7)$$

以此类推,令通式为:

$$S_j^{(k)} = S_j^{(k-1)} + S_{(\frac{m}{2k}+j)}^{(k-1)} \quad (8)$$

经过 k 次迭代后,可得最后和为:

$$S_n = S_j^{(k)} \quad (j = 1, 2, 3, \dots) \quad (9)$$

式中: i — 像素位置; T — 阈值; P — 像素值; S — 像素的和; n — 总的像素个数; m — 一次并行处理的线程数; k — 迭代次数。

检测系统中,图像噪声滤波和像素点判断统计采用不同的核函数,这里充分利用了 Nvidia 图像卡 GPU 中已经分配的地址,以减少数据拷贝时间。

CUDA 缺陷检测系统如图 7 所示。

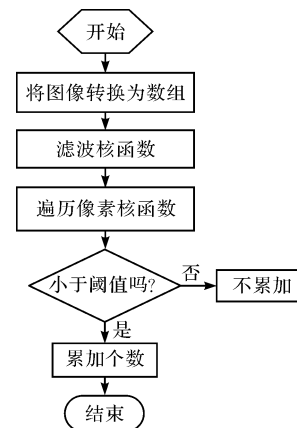


图 7 CUDA 缺陷检测系统

3 实验及结果分析

3.1 实验设备

(1)多计算机集群配置的设备来自阿里服务器,其集群硬件如表 2 所示。

表 2 集群硬件

序号	主机名	Spark 集群计算机配置
1	Master	ECS 云服务器 C5 2 核,8 GB
2	Slave1	ECS 云服务器 C5 2 核,8 GB
3	Slave2	ECS 云服务器 C5 2 核,8 GB

集群软件配置如表 3 所示。

表 3 集群软件配置

操作系统	Hadoop	Java	scala	spark
Ubuntu14.04.5	Hadoop1.x.x	JDK1.6.x	Scala2.9.3	Spark1.0.0

(2) 单机多核运行的实验设备系统配置是: CPU 为 Intel(R) CPU E5-2620 v3 @2.40 GHz(2 core) 内存 32GB, GPU 为 Nvidia 的 GTX1080Ti。操作系统是 Windows Servers 2012 R2 64 位操作系统。使用 OpenCV、C++、CUDA C。1.5 节的串行传统遍历算法也使用该计算机 CPU 运算。

3.2 实验数据

实验中使用图 4 中的被检测图片。由于生产现场被检测的轴是按照流水线逐个检测,每次检测一张图片,并判定是否合格。实验分别按照 1.5、2.1、2.2 节的方法测定运行时间,运行时间按照有数据传输和无数据传输分别测定。

不含通讯和数据拷贝的时间对比如表 4 所示。

表 4 不含通讯和数据拷贝的运算时间对比

名称	传统遍历	Spark	CUDA 加速
运行时间/ms	984	6 000-8 000	19
备注	逐个元素检测	含计算机部署	不含数据拷贝

包含数据通讯和拷贝的时间对比如表 5 所示。

表 5 包含数据通讯和拷贝的时间对比

名称	传统遍历	Spark	CUDA 加速
运行时间 /ms	984	6 000-8 000+网速	320
备注	逐个元素检测	含部署和网络通讯	数据拷贝耗时

3.3 实验结果分析

3.3.1 基于 Spark 的检测

根据 2.1 节采用 Spark 框架检测一张图片的运行时间,包括图片拷贝到 HDFS 上的时间、计算机之间的数据交换通讯时间,以及调度器在不同节点上的工作任务部署时间,至少需要 6 s 以上,这对于实时系统显然是不合适的。

根据众多文献实验证明,大量图片一次性拷贝到 HDFS 上进行运算,随着计算机节点数量的增加,运算时间明显减少,适合大数据的批量运算。

3.3.2 基于 CUDA 检测

根据 2.2 节,用基于 CUDA 的单计算机多核并行计算方式,实验表明其完整运算时间为 320 ms,可以满足实时要求^[32]。如果采用机器学习,得到预测模型会花很长时间。根据文献[33]实验可知,一张 512 × 512 单张图片的模型检测时间在 2.29 s 以上,这对于 Spark 的分布式计算机集群上是可行的,但不符合实时检测要求。

评估并行加速的指标是加速比,其公式为:

$$S_p = \frac{T_c}{T_p} \quad (10)$$

式中: T_c —串行计算时间; T_p —并行计算时间; S_p —加速比。

由实验可知, GPU 在有通信的时候的加速比是 3; 无通信的时候的加速比是 52, 可见数据的通讯和拷贝时间占用比较多的时间。

3.4 实时检测和大数据量数据分析的混合结构模式

根据以上分析, 对于一个比较完整的轴表面缺陷检测, 既要满足实时在线的快速检测, 又要进行缺陷分类, 宜用人工智能中的基于神经网络的深度学习, 所以笔者提出了一种 CUDA + Spark 的混合模式。

CUDA + Spark 模型如图 8 所示。

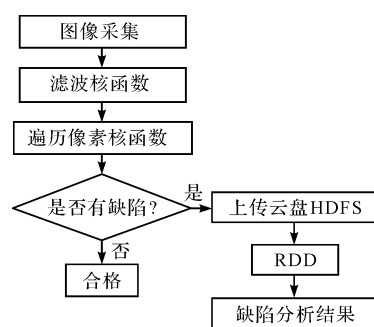


图 8 CUDA + Spark 模型

根据混合模型, 利用 CUDA 实现快速的实时检测, 然后把缺陷图片全部上传到 HDFS 存储, 可以间隔一段时间, 对批量数据进行分析处理。

RDD 最适合对数据集中所有的元素进行相同的操作的批处理类应用, 所以将缺陷分析部分置于 Spark 架构上是可行的。

两种并行方式的结合使用会有更好的空间, 除了

租用云服务器外,也可以将云计算改成就地多计算机系统组成就地云网络,以减少数据传输时间,并用于存储超大数据和长时间的复杂计算,使用 GPU 系统并行计算来作为实时计算,同时利用就地云网络的计算结果来为实时判断提供依据。

4 结束语

当前,基于视觉的缺陷检测已经不仅仅局限于实时检测产品是否合格,也需要分析产生缺陷的原因,而分析产生缺陷的原因需要对大量数据进行存储和处理。

笔者经实验证明了通过数学并行原理优化后的 GPU 片上多核的并行计算方式,适用于高分辨率图像实时环境;而基于云的 Spark 架构分布式并行计算方式,不适用于实时环境,但却适合使用在计算和数据密集、大批量数据处理的场合,而 GPU 却不适合。所以采用 GPU 和 Spark 结合模式可以满足对高分辨率图像的实时检测和大数据量分析场合的需求。

参考文献 (References):

- [1] JACK D. 并行计算综论[M]. 莫则尧译. 北京:电子工业出版社,2005.
- [2] 范植华,廖名学. 并行性分析的数学原理[M]. 合肥:中国科学技术大学出版社,2014.
- [3] SUSAN H H. Efficient data sharing with conditional remote memory transfers [J]. **Computer Architecture News**, 1996,24(5):1-8.
- [4] BENEDICT R Gaster, et al. OpenCL 异构计算[M]. 2 版. 张云泉等译. 北京:清华大学出版社,2012.
- [5] 兰云旭,王俊峰,唐 鹏. 基于 Spark 的并行医学图像处理研究[J]. 四川大学学报:自然科学版,2017,54(1):65-70.
- [6] 王 恒,郑笔耕. 基于 Spark 的 CT 并行图像处理技术研究[J]. 荆楚理工学院学报. 2019,34(5):5-8.
- [7] 曾有灵,陈耿铎,熊 威,等. 基于 Spark 的 CT 图像 FBP 重建算法程序并行设计[J]. 计算机工程与应用,2019,55(10):218-224.
- [8] 王 康. Spark 的并行处理技术在岩石薄片图像的研究与应用[J]. 智能计算机与应用,2019,9(2):196-199.
- [9] 霍帅起. 基于 Spark 的海量遥感图像并行镶嵌处理方法研究[D]. 哈尔滨:东北林业大学信息与计算机工程学院,2018.
- [10] 姜雪茸. 基于 CUDA 的最大互相关算法加速处理研究[J]. 甘肃科技,2019,35(22):18-20.
- [11] 姜雪茸,魏 威. CUDA 技术在数字图像匹配中的应用[J]. 现代信息技,2019,3(18):61-63.
- [12] 费华英. 基于 CUDA 架构图像处理并行算法实验研究[J]. 铜仁学院学报,2017,19(6):58-61.
- [13] 万 兵. 基于 GPU 的图像处理计算方法的相关研究[J]. 智库时代,2019(11):239-240.
- [14] 戴 伟. 射电天文数据实时计算的关键技术研究[D]. 昆明:中国科学院大学(中国科学院云南天文台),2019.
- [15] 余 霞,葛 红,何 俊,等. 基于 MPI 的并行医学图像处理[J]. 计算机工程与科学,2009,31(3):32-34,40.
- [16] 朱 瑾,吴开华. 基于 OpenMP 并行技术的多核环境下轮对图像处理方法研究[J]. 机电工程,2015,32(6):883-886.
- [17] 罗 菁,董婷婷,宋 丹,等. 表面缺陷检测综述[J]. 计算机科学与探索,2014,8(9):1041-1048.
- [18] RAVIKUMAR S, RAMACHANDRAN K I, SUGUMARAN V. Machine learning approach for automated visual inspection of machine components [J]. **Expert Systems with Applications**, 2011,38(4):3260-3266.
- [19] HUANG Jie-xian, LI Di, YE Feng, et al. Detection of surface of solder on flexible circuit[J]. **Optics and Precision Engineering**, 2010,18(11):2443-2453.
- [20] JEONG Y S, KIM S J, JEONG M K. Automatic identification of defect patterns in semiconductor wafer maps using spatial correlogram and dynamic time warping [J]. **IEEE Transactions on Semiconductor Manufacturing**, 2008, 21(4):625-637.
- [21] MAK K L, PENG P, YIU K F C. Fabric defect detection using morphological filters[J]. **Image and Vision Computing**, 2009,27(10):1585-1592.
- [22] TSAI D M, HSIEH C Y. Automated surface inspection for directional textures [J]. **Image and Vision Computing**, 1999,18(1):49-62.
- [23] WANG Qing-xiang, LI Di, ZHANG Wu-jie, et al. Unsupervised defect detection for gold surface of flexible printed board[J]. **Optics and Precision Engineering**, 2010,18(4):981-987.
- [24] LI W C, TSAI D M. Wavelet-based defect detection in solar wafer images with inhomogeneous texture [J]. **Pattern Recognition**, 2012,45(2):742-756.
- [25] MANDELBROT B B. The fractal geometry of nature[M]. [S. I.]: W H Freeman, 1982.
- [26] COHEN F S, FAN Z, ATTALI S. Automated inspection of textile fabrics using textural models[J]. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, 1991,13(8):803-808.

- [27] CHAO S M, TSAI D M. Anisotropic diffusion with generalized diffusion coefficient function for defect detection in low-contrast surface images [J]. **Pattern Recognition**, 2010, 43(5):1917-1931.
- [28] HOLDEN K(美). Spark 快速大数据分析[M]. 王道远译. 北京:人民邮电出版社,2015.
- [29] 王晓华. SparkMLlib 机器学习实践[M]. 北京:清华大学出版社,2015.
- [30] JOHN CHENG(美)等. CUDA C 编程权威指南[M]. 颜成钢译. 北京:机械工业出版社,2017.
- [31] JASON SANDERS, EDWARD KANDROT(美). CUDA by example: an introduction to general-purpose GPU programming[M]. 聂学军译. 北京:机械工业出版社,2011.
- [32] 姜庆胜,李研彪,计时鸣. 活塞杆抛光表面微小缺陷的快速筛查技术[J]. 计算机集成制造系统:1-19[2020-04-05]. <http://kns.cnki.net/kcms/detail/11.5946.TP.20200312.1422.014.html>.
- [33] JIANG Qing-sheng, TAN Da-peng, LI Yan-biao, et al. Object detection and classification of metal polishing shaft surface defects based on convolutional neural network deep learning[J]. **Applied Sciences**, 2020, 10(1):87.

[编辑:冯黎雅]

本文引用格式:

姜庆胜,李研彪,计时鸣. 基于并行计算的抛光轴表面缺陷检测研究[J]. 机电工程,2020,37(12):1540-1546.

JIANG Qing-sheng, LI Yan-biao, LI Shi-ming. Surface defect detection of polishing shaft based on parallel computation[J]. *Journal of Mechanical & Electrical Engineering*, 2020, 37(12):1540-1546.《机电工程》杂志:<http://www.meem.com.cn>

(上接第 1514 页)

定性变差,响应速度达到极限,罐笼振动严重,双闭环比例系数比对系统动态特性呈非单调影响,过大或过小都会使系统动态特性变差,甚至导致钢丝绳打滑,需要综合寻求最优解。

参考文献(References):

- [1] WÓJCIK M. Simulation testing of emergency braking of the mining shaft hoist [J]. **Journal of Konbin**, 2010, 23(14, 15): 401-408.
- [2] RAZDOLSKY A. G. Propagation of longitudinal deformation wave along a hoisting rope carrying an intermediate concentrated load[J]. **Journal of Dynamic Systems, Measurement, and Control**, 2014, 136(6):061002.
- [3] S. KACZMARCZYK, W. OSTACHOWICZ. Transient vibration phenomena in deep mine hoisting cables. part 1: mathematical model [J]. **Journal of Sound and Vibration**, 2003, 262(2):219-244.
- [4] WANG G Y, XIAO X M, MA C. et al. Nonlinear dynamic behavior of winding hoisting rope under head sheave axial wobbles[J]. **Shock and Vibration**, 2019(3): 1-11.
- [5] 肖林京,吕淑芳,孙孟杰. 安全制动时提升系统振动分析[J]. 山东矿业学院学报,1995,14(3):77-80.
- [6] 曹国华. 矿井提升钢丝绳装载冲击动力学行为研究[D]. 北京:中国矿业大学机电工程学院,2009.
- [7] 黄家海,贺亚彬,于培,等. 落地式摩擦提升机建模和振动特性分析[J]. 机械工程学报,2019,55(12):205-214.
- [8] CAO Guo-hua, WANG Jin-jie, ZHU Zhen-cai. Coupled vibrations of rope-guided hoisting system with tension difference between two guiding ropes[J]. **Proceedings of the Institution of Mechanical Engineers Part C Journal of Mechanical Engineering Science**, 2016, 232(2):1989-1996.
- [9] 麻慧君,张国新,权龙,等. 基于PID自整定控制恒减速制动系统的研究[J]. 煤矿机械,2013,34(7):155-156.
- [10] 郝建功,张耀成. 矿井提升机动力制动过程中的模糊控制[J]. 煤炭学报,2003,28(1):96-100.
- [11] 康喜富,李斌,权龙,等. 矿用提升机恒减速制动特性研究[J]. 液压与气动,2017(5):79-83.
- [12] 司昌练. 基于电液位置伺服系统机床滑台模糊自适应控制研究[J]. 液压气动与密封,2019(5):37-42.
- [13] 康喜富. 大型矿用提升机恒减速制动电液控制系统性能研究[D]. 太原:太原理工大学机械学院,2017.

[编辑:冯黎雅]