



同濟大學

TONGJI UNIVERSITY

硕士学位论文

(专业学位)

基于非侵入式方法的
冷冻机房故障诊断研究

姓名：陈智博

学号：1732702

所在院系：机械与能源工程学院

职业类型：工程

专业领域：动力工程

指导教师：许鹏 教授

副指导教师：

二〇二〇年三月



同濟大學
TONGJI UNIVERSITY

A dissertation submitted to
Tongji University in conformity with the requirements for
the degree of Doctor of Philosophy

**Research on Fault Detection and Diagnosis
of the Chiller Room Using Non-intrusive
Approach**

Candidate: Chen Zhibo

Student Number: 1732702

School/Department: School of Mechanical Engineering

Discipline: Enginerring

Major: Power Engineering

Supervisor: Prof.Xu Peng

May, 2020


基于非侵入式方法的冷冻机房故障诊断研究

陈智博

同济大学

学位论文版权使用授权书

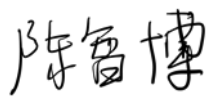
本人完全了解同济大学关于收集、保存、使用学位论文的规定，同意如下各项内容：按照学校要求提交学位论文的印刷本和电子版本；学校有权保留学位论文的印刷本和电子版，并采用影印、缩印、扫描、数字化或其它手段保存论文；学校有权提供目录检索以及提供本学位论文全文或者部分的阅览服务；学校有权按有关规定向国家有关部门或者机构送交论文的复印件和电子版；在不以赢利为目的的前提下，学校可以适当复制论文的部分或全部内容用于学术活动。

学位论文作者签名：

2020 年 3 月 1 日

同济大学学位论文原创性声明

本人郑重声明：所提交的学位论文，是本人在导师指导下，进行研究工作所取得的成果。除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人创作的、已公开发表或者没有公开发表的作品的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位论文原创性声明的法律责任由本人承担。

学位论文作者签名：

2020 年 3 月 1 日

摘要

空调系统的故障诊断是维持空调系统正常运行，降低建筑能耗并提升空调系统设备使用寿命的重要手段。其中，冷冻机房内的设备和系统价格贵，耗能高，因此冷冻机房内设备和系统的故障诊断是空调系统故障诊断中至关重要的一部分。

已有的针对冷冻机房内设备和系统的故障诊断研究大多基于建筑自动系统（BA）采集的数据，这样的故障诊断在实际中会受到传感器和数据通信误差和故障的影响，且实施成本较高。本文利用基于音频和图像的故障诊断算法，以故障状态监测为目的，以冷冻机房比较普遍的日常巡检任务为切入点，以实现利用能够采集音频和图像的巡检机器人对冷冻机房的主要设备进行自动巡检，达到空调系统的智能化管理为愿景，开发多种自动化的故障诊断算法模块。这些诊断模块具有非侵入、成本低和易部署等优点。

本文中基于音频的算法来监测水泵运行情况，算法主要包括信号特征提取和信号时间序列拟合两部分。信号特征提取对信号进行时频变换处理，有限冲激响应滤波，将信号按照频率高低分成不同频段。信号时序拟合对不同频段的信号进行稳态或非稳态的随机时序处理方法，获得信号的时间序列拟合，然后将时间序列拟合的置信区间作为阈值指导水泵运行情况的状态监测。该算法在实践中可以准确地监测水泵运行的异常声音来判断异常和故障。

本文中基于图像的算法来进行温度和压力表盘的示数读取以及水泵温度示数的识别。算法的第一部分是对冷冻机房的图片按照所属设备进行分类，本文利用迁移学习，关闭特征层的梯度更新的方式训练一个 AlexNet 卷积神经网络对表盘、卧式水泵、立式水泵、压缩机 4 种设备进行分类。巡检机器人可以利用分类算法模块识别采集的图像，从而选择对应的图像故障诊断算法模块。

温度和压力表盘示数读取算法利用图像二值化、最大连通域提取、形态学处理和边缘检测等多种图像处理方法获取指针图像，然后基于区域划分利用三角函数计算指针夹角，并换算得到表盘示数。该算法在实践中读数准确率较高，可以监测温度和压力示数来判断异常和故障。

水泵温度识别算法利用图像二值化、缩放等图像处理方法获取水泵红外图像的最高温度示数图像，然后利用光学文字识别方法对温度示数进行识别。该算法识别准确率很高，可以监测水泵的温度来判断异常和故障。

关键词：冷冻机房，故障诊断，音频处理，时间序列分析，图像处理

ABSTRACT

The fault diagnosis of the air-conditioning system is an important means to maintain the normal operation of the air-conditioning system, reduce the energy consumption of the building and increase the service life of the air-conditioning system equipment. Among them, the equipment and systems in the chiller room are expensive and energy-consuming, so the fault diagnosis of the equipment and systems in the chiller room is a vital part of the fault diagnosis of the air conditioning system.

Existing fault diagnosis studies for equipment and systems in the chiller room are mostly based on data collected by building automation systems (BA). Such fault diagnosis will be affected by the deviance and fault of sensors and data communication in practice, and the implementation cost is relatively high. With the purpose of fault status monitoring, and the starting point for the routine inspection tasks that are common in a chiller room, the envision to use an inspection robot capable of collecting audio and images to automatically inspect the main equipment of a chiller room and finally achieve intelligent management of the air conditioning system, this paper uses audio and image-based fault diagnosis algorithms to develop a variety of automated fault diagnosis algorithm modules. These diagnostic algorithm modules are non-intrusive, low cost, and easy to deploy.

In this paper, an audio-based algorithm is used to monitor the operation condition of pumps. The algorithm mainly includes the signal feature extraction and the signal timeseries fitting. The signal feature extraction processes signals using time-frequency domain transform and finite impulse response digital filter, by which signals are divided into different frequency bands according to the frequency. The Signal timeseries fitting performs steady-state or non-steady-state random timeseries analysis on signals in different frequency bands to obtain the timeseries fitting of signals, and then uses the confidence interval of the timeseries fitting as a threshold to guide the status monitoring of the pump operation. This algorithm can accurately monitor the abnormal sound of pumps to detect abnormalities and faults.

in this paper, image-based algorithms are used to read the indications of the temperature and pressure gauges and identify the temperature numbers of the pump images. The first part of algorithms is to classify the equipment images of the chiller

room. This paper uses transfer learning, closes gradient update of the feature layers to train an AlexNet convolutional neural network to classify 4 types of the equipment, including the gauges, the horizontal pumps, the vertical pumps, the compressors. The inspection robot can use the classification algorithm module to identify the collected images, and then select the corresponding image fault diagnosis algorithm module.

The indication reading algorithm of the temperature and pressure gauges uses a variety of image processing methods such as image binarization, maximum connected domain extraction, morphology processing and edge detection to obtain the pointer image, and then uses trigonometric function to calculate the pointer angle based on the area division and finally convert it to the gauge indication. This algorithm has high accuracy in practice, and can monitor the temperature and pressure values to detect abnormalities and faults.

The pump temperature recognition algorithm uses image processing methods such as image binarization and image scaling to obtain the image of the highest temperature value in the infrared image of pump, and then uses the optical character recognition method to identify the temperature value. The algorithm has high recognition accuracy and can monitor the temperature of the pumps to detect abnormalities and faults.

Key Words: chiller room, fault diagnosis, audio signal processing, timeseries analysis, image processing

目录

第 1 章 引言	1
1.1 概述.....	1
1.2 空调系统故障诊断及其研究意义.....	1
1.3 空调系统故障诊断的研究现状.....	2
1.4 非侵入式方法的诊断思想.....	4
1.4.1 基于图像的非侵入式方法	4
1.4.2 基于音频的非侵入式方法	6
1.5 研究路线及技术方法.....	8
第 2 章 基于音频数字信号处理的水泵音频提取.....	11
2.1 声音信号的采样.....	11
2.2 傅里叶变换.....	12
2.2.1 傅里叶变换的基本类型	12
2.2.2 从连续傅里叶变换到离散傅里叶变换	13
2.2.3 工程实用方法——快速傅里叶变换	14
2.2.4 信号幅值域的分析 and 采样频率的确定	14
2.3 信号的滤波.....	17
2.3.1 信号滤波的基本概念	18
2.3.2 有限冲激响应滤波	19
2.3.3 对采集的信号进行滤波处理	20
2.4 信号的滤波实例及结果分析.....	21
2.4.1 中区热水二次泵 1 号滤波实例	21
2.4.2 实例结果分析	27
2.5 本章小结.....	28
第 3 章 基于时间序列分析处理音频的水泵故障分析.....	29
3.1 信号的时间序列分析.....	29
3.1.1 平稳时间序列的分析方法	29
3.1.2 非平稳时间序列的确定性时序分析方法	30

3.1.3	非平稳时间序列的随机时序分析方法	31
3.1.4	时间序列分析信号实例	32
3.2	时间序列分析的平稳性检验	32
3.2.1	平稳性检验的方法	32
3.2.2	对分析信号实例进行平稳性检验	33
3.3	时间序列分析的纯随机性检验	41
3.3.1	平稳时间序列的分析价值	41
3.3.2	平稳时间序列的纯随机性检验	42
3.3.3	对分析信号实例进行纯随机性检验	42
3.4	ARIMA 模型拟合时间序列过程	44
3.4.1	ARIMA 模型建模的基本步骤	44
3.4.2	ARIMA 模型参数的估计	45
3.4.3	ARIMA 模型的评估准则及参数优化	47
3.4.4	ARIMA 模型的有效性校验	48
3.4.5	ARIMA 模型的测试和故障监测指导	54
3.5	本章小结	56
第 4 章	基于深度学习图像处理的冷冻机房关键设备分类	59
4.1	深度学习和张量库 Pytorch 介绍	59
4.1.1	深度学习及其发展	59
4.1.2	深度学习框架 Pytorch	59
4.2	卷积神经网络	60
4.2.1	定义和基本结构	60
4.2.2	经典卷积神经网络 AlexNet	64
4.3	冷冻机房关键设备图片准备	64
4.3.1	分类涉及内容和目的	64
4.3.2	关键设备图像类型和训练预处理	65
4.4	模型的训练准备	70
4.4.1	AlexNet 模型迁移学习	70
4.4.2	模型损失函数	70
4.4.3	模型优化算法	71
4.5	模型训练结果和预测展示	72

4.5.1	模型训练过程	72
4.5.2	模型对新图像的分类验证	73
4.6	本章小结	74
第 5 章 基于图像处理的冷冻机房表盘示数读取		77
5.1	冷冻机房表盘图像示数读取算法思路	77
5.2	表盘图像的图像预处理操作	78
5.2.1	表盘图像的缩放和变换操作	78
5.2.2	表盘图像的二值化操作	79
5.3	表盘图像中指针的提取	83
5.3.1	基于区域标记和区域特征统计的连通域提取方法	83
5.3.2	表盘图像中表盘部分的提取	83
5.3.3	表盘图像中指针部分的提取	84
5.4	指针二值图像的形态学操作	86
5.4.1	形态学操作的腐蚀和膨胀	86
5.4.2	形态学操作的开操作和闭操作	87
5.4.3	指针图像的开运算和闭运算	87
5.5	针尖部分图像的提取	88
5.5.1	依据像素裁剪指针图像	88
5.5.2	指针图像的边缘检测	89
5.5.3	针尖部分的提取	90
5.6	判定指针在表盘中所处的区域	91
5.6.1	表盘 4 个区域的确定	91
5.6.2	依据高宽比初步划分区域 1、4 和区域 2、3	92
5.6.3	依据指针形状特性划分区域 1-4	93
5.7	计算指针的指示角度	93
5.7.1	区域 1 内指针的夹角计算	94
5.7.2	区域 2 内指针的夹角计算	94
5.7.3	区域 3 内指针的夹角计算	94
5.7.4	区域 4 内指针的夹角计算	95
5.7.5	多组点计算角度后求平均	95
5.8	角度与温度或压力示数的对应	96

5.8.1	角度与温度示数的对应	96
5.8.2	角度与压力示数的对应	97
5.9	输出结果验证	99
5.9.1	温度表读数结果展示	99
5.9.2	压力表读数结果展示	99
5.10	本章小结	100
第 6 章	基于红外图像的水泵温度示数提取	101
6.1	红外图像温度读数思路	101
6.1.1	光学文字识别工具 pytesseract	101
6.1.2	红外图像温度识别流程	101
6.2	红外图像中的温度示数提取	101
6.2.1	红外图像的读取	101
6.2.2	红外图像温度示数裁剪	102
6.3	红外图像中温度示数的预处理和识别	102
6.3.1	温度示数图像的二值化处理	102
6.3.2	二值化图像的放大与类型转换操作	103
6.3.3	温度示数的识别及验证	103
6.4	本章小结	106
第 7 章	结论与展望	107
7.1	结论	107
7.2	展望	108
致谢	111
参考文献	113
附录 A 部分代码节选	117
个人简历、在读期间发表的学术论文与研究成果	132

第 1 章 引言

1.1 概述

公共建筑的冷冻机房是公共建筑中承担价值温度和湿度调控的区域，一般拥有冷冻机房的公共建筑中，冷冻机房主要包括冷水机组、冷冻水泵、冷却水泵、集水器和分水器为核心设备以及冷冻水环路和冷却水环路（水冷冷水机组有）等部分管路。

冷冻水管路有供水管路和回水管路之分。冷冻水在冷水机组的蒸发器换热后经由冷冻水供水管路流向建筑不同区域，然后在建筑的不同区域进入该处的空调机组（Air Handling Unit）的冷盘管，与空气进行换热后，经由冷冻水回水管路再次流向蒸发器。对于大型水冷冷水机组，冷却水管路也有供水管路和回水管路之分。冷却水在冷水机组的冷凝器换热后经由冷却水供水管路流向冷却塔，在冷却塔内与空气或水进行换热降温后，经由冷却水回水管路再次流向冷凝器。

冷冻水泵是冷冻水环路的动力来源，就像人体的心脏把动脉血从左心房压向全身一样，冷冻水泵将管路中的冷冻水压送到建筑其他区域。而冷却水泵是冷却水环路的动力来源，负责将冷却水压送到室外的冷区塔。

而冷水机组是消耗外部能源产生冷量制造冷水的部件，可以说是冷冻机房内的最核心部件。

实际中，由于多台冷水机组、多台冷却塔以及多台冷冻水泵和冷却水泵的存在，管路会包括供回水干管和供回水支管，通过分水器 and 集水器进行分配输送和回收。

可见，整个冷冻机房是一个各个设备各司其职，关联紧密的系统。具体到系统的每个设备，如最为重要的冷水机组和水泵，它们各自内部亦有不同的分工。水泵有电机、变频器、叶片等组成，冷水机组由蒸发器、冷凝器、节流阀以及压缩机四大件组成，同时还有供油器等辅助设备。同时，不论是水泵还是冷水机组，都可能装有控制盘，以及压力表、温度表等仪表装置。最终接入公共建筑的建筑自动控制系统，从中可以观测到每个设备的运行当前关键参数的值。

庞大的冷冻机房通过各个设备的协调运行提供着整个建筑的温湿度需求的基本供应，根本目的是保证建筑的各个空间的舒适性。

1.2 空调系统故障诊断及其研究意义

空调系统作为一个有着许多相关设备连接,和严密控制逻辑的系统,是建筑舒适性调节中不可或缺的一部分,它的正常工作对建筑的环境控制有着重要的意义。如果建筑的空调系统运行的稳定性、安全性、可靠性不能得到保证,不仅对相应建筑的正常运作产生不利影响,而且会对在其中的人群带来不舒适的环境体验。如果是办公建筑,则会削弱办公职员的工作效果;如果是酒店建筑,则会影响顾客的居住体验;更甚者,若为重要的需要温度湿度控制严格的医院建筑的手术室,特殊病房等,空调系统的工作正常与否可能影响到重大事态的发生与否。

同时,公共建筑消耗的能源也日益成为能源社会重要的一环,保证空调节能运行,不因为故障等造成浪费能源等情况,也是值得重视的一方面。

因此,监测空调系统的工作状况,并在相关设备发生故障时,进行及时的报警,以及进一步分析产生故障的原因,可以对空调系统进行有效的预防性运维,保证空调系统的正常运行,提高空调系统寿命。同时,建筑的空调系统可能由于种种故障不能满足性能期望,也造成了大量的能源浪费。在如今世界能源消耗规模庞大,而可再生能源的使用又十分有限的形势下,对空调系统的故障检测与诊断方案与技术是提高建筑能效中扮演重要的角色。

因此,探索并研究自动的、响应迅速的、精确并可靠的建筑空调系统的故障检测和诊断的方案,以此来确保空调系统的正常运行,具有极大的潜力。不仅于此,建立一套空调系统的故障检测与诊断方案可以帮助建立新建建筑和既有建筑的建造和翻新的标准。

1.3 空调系统故障诊断的研究现状

在已有的对空调系统故障诊断的研究中,对冷水机组和空气处理单元及末端的研究占据主要地位。

Yuebin Yu 等人^[1]对现阶段针对空气处理单元开发的故障诊断方法做了综述。总结了单元空调机组中的典型故障主要有四大类,包括:风机、管道、冷盘管、热盘管、预热盘管等设备的故障;新风、回风和排风阀以及热盘管、冷盘管和预热盘管阀等执行器的故障;温度、湿度、流量以及送风和区域压力等传感器的故障,电机调节、冷热盘管阀序列、流量差、静压、区域温度等控制器的故障。在一定的推理和描述的基础上,将 2005 年之前总结的暖通空调领域的故障诊断方法三大类:基于定量模型法、基于定性模型法和基于历史过程法的分类方法进行了重新总结,重点加入了诸如 PCA(主成分分析)、FDA(Fisher 判别分析)、JAA(联合角度分析)、DWT(离散小波变换)法等数据驱动的方法,将单元空调机

组系统（AHU）的故障诊断方法分为三大类，分别是基于解析的方法，基于知识的方法和数据驱动方法。

Rui Yan 等人^[2]提出了一个基于分类和回归决策树方法的空调机组故障诊断策略，提出的策略可以自动进行特征选择。且能够用测试数据和专家知识进行算法策略的验证。验证也表明，该策略所执行的故障诊断是可解释的，且大部分与专家知识吻合度很高。另有小部分故障并不可靠，旨在某种特定情况下得到验证。

Haitao Wang 等人^[3]针对研究较少的变风量（VAV）末端，开发了一套使用混合方法的故障检测和诊断策略。该策略利用基于残差的累计和（CUSUM）控制图方法来检测 VAV 末端的故障，残差为 VAV 末端设定温度和实际送风温度的温度差与通过自回归时间序列预测产生的温度差的差值。随着不断地记录，如果 CUSUM 值超过限值，则意味着 VAV 末端存在故障或异常。同时该策略利用由专家规则和故障隔离算法组成的故障分类器来诊断识别 15 个故障源。最终在 VAV 末端的实时数据上进行测试和验证，证明该混合策略可以有效地检测故障并诊断故障原因。

Timothy Mulumba 等人^[4]提出了一个混合的空调机组单元故障检测和诊断方法，方法包括引入了外源变量的自回归时间序列模型 ARX 和采用“一对多”分类算法的多类支持向量机（SVM）分类器。文中首先通过特征选择方法选择出一些重要的包括不同的温度、湿度、流量等物理量特征，并从中选出一个决定性特征，其余的则为外源变量。然后利用 ARX 拟合决定性特征与外源变量，并得到一个参数空间。最后通过在故障状态下得到的状态参数空间和正常状态下得到的状态参数空间数据集上训练 SVM 分类器，来进行故障分类和诊断。

Cheng Fan 等人^[5]分析了现阶段数据挖掘技术在建筑系统故障诊断和性能优化方面的应用方向，指出将数据挖掘技术应用在建筑管理的两大挑战分别是：建筑运行数据质量较差，由于传感器、传输系统和数据储存等带来的缺失值、异常值和死值较多；建筑工程和数据挖掘之间的知识结合不紧密，缺少深入将两方面结合的研究人员和工程人员。同时，文章以香港贸易金融中心为例，建立结合时序分析、递归特征消去和集成学习集合方法的能耗预测模型还有基于量化关联性分析的运行策略监测与评估，实现了一个较为综合的数据挖掘技术在建筑系统性能诊断和优化的应用思路与框架。

综上，可以发现在对空调系统故障诊断的研究中，基于建筑控制系统记录的空调系统运行数据被作为发现系统故障的重要数据源。基于空调系统运行数据可以结合相关的冷水机组模型、水泵模型、冷却塔模型等建立空调系统内相关设备的半物理模型，根据这些半物理模型与运行数据的结合，利用多种统计学和机器学习方法可以建立各种数据驱动型的故障检测和诊断模型

1.4 非侵入式方法的诊断思想

本论文基于非侵入式方法对冷冻机房设备进行故障诊断，其主要思想是依靠巡检机器人在机房进行巡视，不采集冷冻机房的运行计量数据，通过音频采集设备采集冷冻机房特定设备的运行音频数据，通过可见光摄像头和红外摄像头采集冷冻机房特定设备的图像。通过分析音频和图像数据，建立信号的数学分析方法和图像处理方法，形成一套完整的冷冻机房特定设备关键故障的故障诊断方法。显然，这一非侵入式的故障诊断方法并不会影响冷冻机房的正常运作，结合巡检机器人的应用，可以实现冷冻机房设备的自动故障诊断。

将声音和图像用于故障诊断的研究也有许多先例。在诸如下水道、电动机(包括定子、转子和轴承等)、齿轮箱等不同的工业场景中也得到了丰富的应用。

1.4.1 基于图像的非侵入式方法

在基于图像的非侵入式故障诊断研究中，红外热成像是一大研究重点对象，因为温度是设备和组件结构健康状况的重要标识，故障的机器、腐蚀的电气接触、损坏的材料组件通常会造成异常的温度分布^[6]。因此红外热成像得到了工程界的广泛关注，在电气设备诸如电机、变压器、输电线路等的故障诊断中^[7-11]还有机械设备和组件诸如风轮机、散热器、定子、转子等的故障诊断中^[12-14]已经获得一定的研究成果并投身于工业应用中。

对于三相电动机的早期故障检测是工业界关心的一大问题，许多的文章也对此有过研究^[15-17]。Indrayudh Bandyopadhyay 等人^[18]开发了一个图像处理和最近邻算法相结合的故障诊断工具，以此来检测电动机换流器的 IGBT（绝缘栅双极型晶体管）中的初期故障，从而防止后续的故障严重化。根据三相电流的 Condordia 模式在不同类型的故障下，会在坐标系上表现出不同的形状这一特性，作者设计了一个基于图像的形状算子来从 Condordia 模式中提取相关信息，然后利用最近邻算法来分类识别故障的特性和位置。

Joshua Myrans 等人^[19]开发了一个利用 CCTV 提取静态照片，实现自动、在线、接近于实时的下水道故障检测方法。故障数据随着 CCTV 的拍摄可以不断更新，并因此而提升预测准确度。作者通过 GIST 算子预测故障的存在，并利用支持向量机和随机森林进行故障分类后，选择精确度更高的随机森林分类器，最终利用隐马尔科夫模型和 order oblivious filter 降低检测的假负率（使得更多的故障情况被发现）。该方法结合了实际图像、特征提取器和机器学习分类算法，最终实现了有实际应用价值的下水道故障检测。

韩国全北国立大学 Alvaro Fuentes 等人^[20]提出了一种基于深度学习的手段来检测番茄植物中的病虫害的方法,首先利用各种分辨率的相机设备实地捕获图像。之后利用基于深度学习的图像识别和图像检测技术对包含疾病和虫害的番茄植物进行训练,实现“图像分类”和“目标检测”两大任务。为了寻找最合适的深度学习框架,作者对比了包括多种特征提取器的3种超深度学习框架,分别是 Faster R-CNN, SSD 和 R-FCN。并发现使用 VGG-16 和 ResNet-50 这两种特征提取器的 Faster R-CNN 和 R-FCN 超框架比其它的平均精确性更好,表现的也更好。

北京林业大学的 Guan Wang 等人^[21]利用从零开始建立一个浅层卷积神经网络的手段和通过微调一个已经训练好的神经网络的顶层来实现迁移学习的手段来判断小样本的细纹理作物疾病严重性。文中利用多种神经网络和参数组合进行训练,最终得出的结论,使用迁移学习的 VGG16 网络预测的最为准确。

来自希腊、德国、西班牙、芬兰四国的研究团队利用目标匹配实现利用红外图像的自动模式识别过程来诊断电机故障^[22]。本文的图像分割阶段利用 SIFT 方法获得“兴趣点”,通过几何和光度学上恒定的描述算子来匹配训练图像和目标图像的兴趣点,之后利用上一步训练的算子匹配训练图片和测试图片,并生成3个区域外缘。计算三个区域的强度来计算其平均温度,并利用这些平均温度组合成不同的温度比例形成故障训练特征,最后选用了2种经典数据挖掘算法 C4.5 决策树和朴素贝叶斯分类器训练生成2种故障分类器对5种情况进行分类(包括1个健康类和转子金属条破损、定子不平衡、风扇故障和轴承故障4个故障类),分类精度均很高。

波兰 AGH 科学技术大学的 Adam Glowacz 等人^[23]利用红外热图像对三相诱导电动机进行故障诊断。本文中,作者利用红外照相机采集热图像后,将其转化成成品红色图像。利用基于二值图像差为主要思路的和图形直方图算法对品红色图像进行特征提取,获得故障特征向量,之后利用反向传播神经网络,最近邻分类器(NN)和 K-均值(K-means)聚类方法进行故障特性训练和识别。作者认为最近邻方法表现最好,这一技术可以有效的对所研究的电机故障图像进行分类,实现较为准确的故障诊断。同时,作者比较了基于红外图像的故障诊断和基于音频信号的故障诊断各自的优势和劣势,并提出在未来,应当分析电流、电压、振动、音频信号和磁场等其他信号,从而开发更可靠的电机故障诊断技术。

综上所述,现阶段利用图像进行故障诊断主要有两大类,一类是对可见光或红外图像进行传统图像处理技术,并结合特征提取和分类方法,进行有效的故障识别。另一类是利用深度学习自动进行特征提取的优势,利用大批量图像进行深度神经网络的训练,再利用训练的神经网络进行故障分类,从而判别故障。

1.4.2 基于音频的非侵入式方法

Vilela R 等人在论文中指出,通常当故障发生时,机器的声学特性会发生变化^[24]。因此,机器的声音携带有关其状况的信息,这些信息有利于我们判断实体机械的故障和不良运行。提取机器的声音特征是故障诊断中的有用工具。

广义的机器声音特征主要有两大来源:机器工作时机体振动所产生的振动信号,以及机器振动引起周围空气介质波动所传播的音频信号。

对于振动信号的研究主要有时间域分析,频率域和倒谱分析,时-频分析,非线性分析等。其中时间域分析主要包括时间特征分析,比如均方根、峰值、方差、峰度、偏度等^[25-28];还有基于时间平均的分析法,如时间同步平均、残差信号以及差分信号等^[29-31];基于时间序列的自回归移动平均 ARMA 模型也是拟合分析振动的有效模型^[31-33]。频率域分析通过信号的频谱特征对信号进行分析,可以通过频谱的多个峰值反映信号的周期性,也可以检测谐波和边带。常见的频谱特征,比如频率的平均值、标准差以及峰值,能量,谱能量的比例,常常被用于水泵、电机和齿轮箱的故障诊断中^[25, 34, 35]。包络分析或者幅值解调也是频谱分析的一大手段,常被用于轴承的故障诊断中,来识别轴承的缺陷特征频率^[34, 36],在齿轮箱中的应用也有相应研究^[37]。时-频分析可以分析非平稳信号,克服了频率域分析只适合分析机器稳定运行时产生的平稳信号的缺点。传统的时-频分析技术主要包括短时傅里叶变换、魏格纳-威尔分布^[38, 39]。近 20 年间,经验模态分解^[40, 41]、希尔伯特-黄变换^[40, 42]、小波变换^[43, 44]等新技术也被用于时-频分析中。

对于声音信号的研究开展相对于振动信号来说较少,可能是因为声音信号会受到其他机械设备和噪声环境等无关声音源的干扰,使得利用声音信号识别机器特征比较困难。因此,利用声音信号更适合对被检测机器设备进行状态监测和异常识别,可以作为前期发现问题的快捷方法。小波变换^[44]以及盲源分割^[45, 46]等降噪方法,被用于音频信号的前处理阶段,用来减少噪声对音频信号的影响。在特征提取方面,用来处理振动信号的技术方法也被用于处理音频信号。

刘滕月^[47]采用振动信号对电机轴承进行故障诊断研究。考虑到轴承发生故障时通常振动信号会变得更复杂,文中对比研究了采用排列熵和样本熵的轴承故障检测方法,证实了两种方法的合理性。在故障特征提取方面,为了解决从复杂振动信号中提取特征困难的问题,基于小波包能量熵还有集合经验模态分解的特征提取方法分别被用来提取轴承故障特征。并利用多分类支持向量机(M-SVM)识别发生在轴承上的故障。文中建立两套电机轴承故障诊断方法,并利用已有的轴承信号数据验证了其合理性和有效性。

李焯^[47]采用电机轴承的振动信号,提出一套轴承故障诊断的策略,以提升轴承故障诊断的精确性。文中主要提出了利用集合经验模态分解(EEMD)降噪预处理原始信号,并利用改进的局部均值分解(ILMD)方法对信号进行故障特征提取,故障的识别使用改进的万有引力搜索算法(IGSA)优化的增量概率神经网络(IPNN)训练识别器。并利用已有的轴承信号数据验证了方法的良好诊断效果。

波兰 AGH 科学技术大学的 Adam Glowacz 等人基于声音信号对单相诱导电动机轴承和定子的故障进行诊断识别^[49]。首先获取健康状态和轴承故障以及定子故障 3 种状况的声音信号,提出的特征提取方法命名为 MSAF-20-MULTIEXPANDED,这一方法利用关注明显频率差值的方法提取显著频率形成特征向量作为训练的故障特征,最后在最近邻分类器、最近均值分类器和高斯混合模型进行故障特征分类识别。最终结果表现良好

波兰 AGH 科学技术大学的 Adam Glowacz 利用音频信号实现单相感应电动机的故障诊断分析^[50]。文中研究了五种工作状态下的电动机:健康电动机,主绕组线圈和辅助绕组线圈短路的电动机,只有辅助绕组线圈短路的电动机,转子断线的电动机,鼠笼环损坏的电动机,以及轴承故障的电动机。开发并实现了一种基于声音频谱差值的音频信号故障特征提取方法 SMOFS-22-MULTIEXPANDED,并使用最近邻(NN)等分类器执行分类步骤。该方法对单相感应电动机的轴承,定子和转子几大主要部件的故障诊断具有良好的效果。

来自墨西哥和西班牙的研究者 Paulo Antonio Delgado-Arredondo 等人^[51]通过分析声学声音和振动信号进行诱导电动机的故障诊断,研究的故障包括有断开两个转子铜条,机械不平衡和轴承缺陷共 3 类。信号获取过程利用麦克风阵列和 3 轴的角速度传感器提取音频和振动信号。提出的方法使用完整集合经验模态分解(CEEMD)来将信号分解为若干本征模函数(IMF)。分解后,通过计算 Gabor 表示的频率边缘以获得频域中的本征模函数(IMF)的频谱内容,根据频谱组分的特征进行故障识别。

Dingcheng Zhang 等人^[52]利用火车轨道边的声音信号来对火车车轮轴承进行适应性故障特征提取。文中作者先利用 ISVD(改进的单值分解法)对路边声音信号进行降噪,再利用 RSSD(基于谐振的信号稀疏分解)对降噪后的信号进行分解,RSSD 可以把车辆的故障轴承声音信号分解为高谐振和低谐振组分。在进行算法的参数调优后,可以获得较为理想的降噪信号和分解组分。之后,利用 Hilbert 变换对包含故障声音信号的低频谐振组分进行解调,通过观察 Hilbert 包裹谱中的重复出现的谐振频率来检测轴承的故障。

杜世斌^[53]从分离故障声音的角度出发,利用独立分量分析(ICA)方法将设备的特征声音信号从混合信号中进行分离,并通过计算声音信号的梅尔倒谱系数

来赋予其相应的特征参数,并借助专家故障诊断库中不同的故障信号来寻找特征参数相似的信号,进而判别是有故障出现。

综上所述,现阶段对音频信号故障诊断的主流研究方法是先获取原始信号,然后对原始信号进行降噪、分解、时域或频域的变换、小波分析、包络谱分析等处理后,利用各种数学方法对变换后的信号进行故障特征提取,作为故障诊断的决策支持数据源。决策支持所用的方法有简单的传统数学特征计算,也有较为复杂的利用机器学习模型训练分类器和识别器等。总体可以概括为特征提取和特征识别两大部分。

1.5 研究背景及技术路线

1.5.1 研究背景和验证场所

本论文进行的研究均在上海市某物业小区的冷冻机房内进行验证,该冷冻机房大致布置情况为:6台螺杆式热泵机组,2台螺杆式冷水机组;多台水泵,其中立式水泵为低区生活热水一次泵;卧式水泵包括新风水循环泵、天棚水循环泵、冷却水循环泵、地源水循环泵,还有中、高区的生活热水一次泵以及低、中、高区的生活热水二次泵;此外还有板式换热器和热水储蓄罐供热水供应。

针对这一庞大的冷冻机房,论文主要研究内容包括对采集到的机房关键设备和装置的图片进行分类,以及针对水泵和主要管路物理数据(通过指针仪表盘的数据)进行以监测为目的的故障诊断,为的是提早发现异常或故障等情况,为及时进行检查维修提供告警。除了检测水泵的相应故障情况,也对相关系统的一些附属部件进行基于日常巡检需要的“软”故障检测,比如对冷冻机房的水系统管道上的温度表盘和压力表盘进行读数识别。因此,本论文的故障诊断开发倾向于包含以音频和图像为故障诊断数据源的多类型算法的研究,且故障诊断以功能实现和基本状态监测为目的,不对具体的特定故障类型做研究。所以本论文的核心研究内容是以功能实现和通过状态监测来报警为目的的故障诊断算法开发。

整个论文的愿景是希望借助自动巡检机器人,搭载相应的故障检测部件(如读取音频信号的麦克风阵列、拍摄红外照片的红外照相机、拍摄可见光照片的可见光相机),通过嵌入机器人电脑系统的相应故障诊断算法程序来实现机房的自动巡检。因此,本论文的研究核心在于利用必要的技术,以朗诗机房的日常巡检要求为主要参考,来开发多种故障诊断算法的程序,从而实现冷冻机房的自动巡检和简单的故障诊断。

1.5.2 技术路线

整个论文的故障诊断算法开发分为两大核心,以“非侵入”式为主导出发点,从音频信号和图像两个角度来开发故障诊断算法。音频信号故障诊断算法开发基于朗诗机房的不同水泵所采集的音频信号,借助时域和频域分析的傅里叶变换、滤波算法对采集到的特定音频信号进行提取后,利用时间序列拟合方法对提取的音频信号进行建模,并利用基于预测置信度的阈值方法进行故障诊断。图像故障诊断算法开发涵盖利用深度学习图像识别进行的冷冻机房设备分类、基于图像分割、连通域提取、形态学操作等图像处理手段的表盘示数识别,基于光学文字识别手段的红外图像温度识别等多种图像故障诊断算法。整个图像诊断算法开发部分的应用框架主要是借助机器人搭载的红外照相机采集冷冻机房不同设备的图像,然后进行基于图像识别的设备分类判断每次采集的照片具体属于哪一类,之后就针对不同类别的图像采用不同的后续故障诊断算法。本论文中,针对采集到的压力表盘和温度表盘的图像,采用对应的表盘示数识别算法进行示数识别,从而判断管道压力和温度是否正常。针对采集到的水泵的图像,采用光学文字识别手段来识别红外图像的最高温度示数,从而判断水泵温度是否异常。

论文的研究思路框架如图 1 所示:

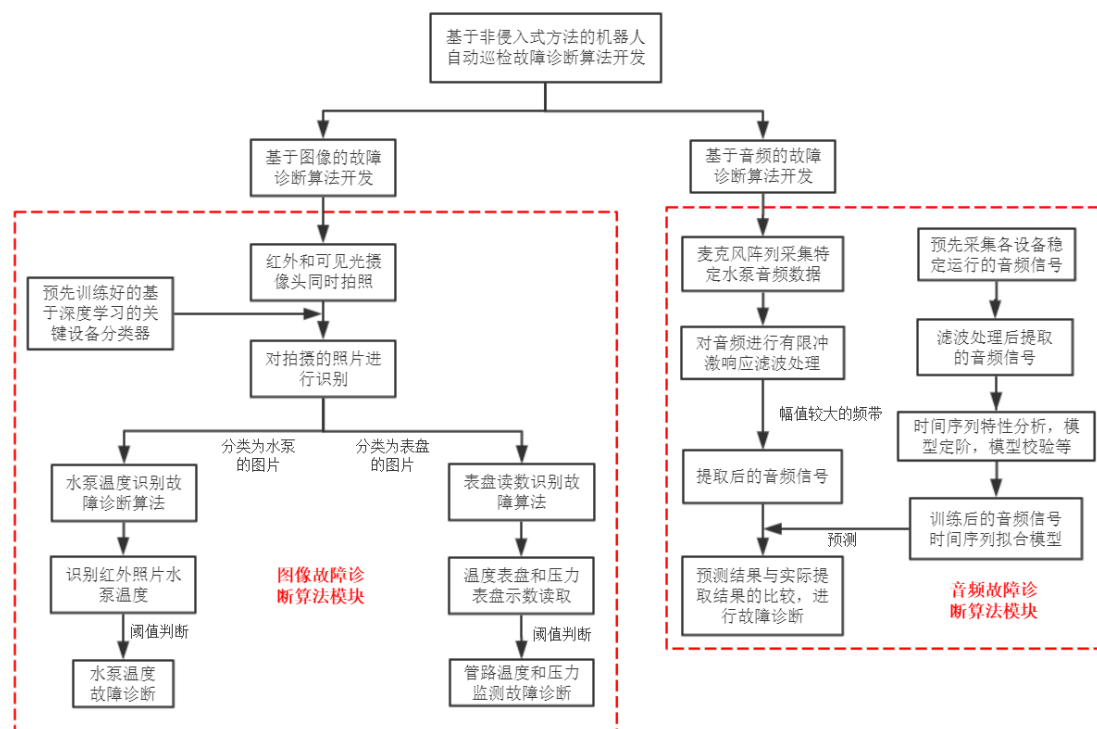


图 1.1 非侵入式故障诊断算法开发思路

第 2 章 基于音频数字信号处理的水泵音频提取

本章内容主要对冷冻机房的水泵音频信号进行采样，并利用时-频变换中的傅里叶变换和滤波方法实现对水泵音频信号的分解和提取。

2.1 声音信号的采样

在工程中记录一段音频时是利用录音设备采样录制，录制的音频信号通常包括一些关键信息，如：采集的音频信号的通道数（单声道或者是双声道信号）；采集的音频信号单位帧的字节宽度，其单位为 Byte；采集音频信号时所选择的采样频率；采集音频信号时所确定的采样点数。其中采样频率和采样点数在音频采样中是相对关键的参数。

除此之外，我们对特定信号进行采样时，应当预先对信号的大致频率域范围进行估计，使得采样过程满足香农采样定理。香农采样定理是数字信号处理领域的重要定理，定理建立了连续信号，即通常所说的模拟信号和离散信号之间的关系。

香农采样定理指的是：定义采样频率 f_s ，信号中含有的最高频率 f_{max} ，当采样频率 f_s 大于信号中最高频率 f_{max} 的 2 倍时，即： $f_s > 2 \times f_{max}$ 时，采样之后的数字信号可以完整地保留原始信号中的信息

利用 ReSpeaker Mic Array v2.0 作为音频信号的采样设备。这是一个远场声音捕捉阵列装置，集成了 4 个 PDM（脉冲密度调制）麦克风来增强设备的声音数字信号处理性能。同时内嵌了很多语音算法。



图 2.1 音频录制设备——麦克风阵列

2.2 傅里叶变换

2.2.1 傅里叶变换的基本类型

傅里叶变换作为一种线性积分变换^[54],常常用来将位于时间域内的信号变换到频率域内;同理,它的反变换用于将位于频率域内的信号变换到时间域。一对傅里叶的正反变换可以将信号灵活地在时间域和频率域内进行变换,从而让我们选择合适的尺度对信号的特征进行观察分析。

傅里叶变换依据时间域信号是连续或离散,是周期性或非周期性,可以分为 4 大类:连续傅里叶变换,将时间域上连续且非周期性的信号变换为频率域上连续且非周期性的信号;傅里叶级数,将时间域上连续且周期性的信号变换为频率域上离散且非周期性的信号;离散时间傅里叶变换,将时间域上离散且非周期性的信号变换为频率域上连续且周期性的信号;离散傅里叶变换,将时间域上离散和周期性的信号,变换为频率域上离散和周期性的信号。

表 2.1 傅里叶变换类型和对应信号特征

傅里叶变换类型	时间域信号特征	频率域信号特征
连续傅里叶变换	连续且呈非周期分布	连续且呈非周期分布
傅里叶级数	连续且呈周期分布	离散且呈非周期分布

离散时间傅里叶变换	离散且呈非周期分布	连续且呈周期分布
离散傅里叶变换	离散且呈周期分布	离散且呈周期分布

分析得出，原信号在时（频）域离散表明其经傅里叶变换后信号在频（时）域呈现周期性，而连续则说明到达域的信号呈现非周期性。即在任意空间连续的信号，经傅里叶变换到另一空间，即为非周期信号；反之，在任一空间离散的信号，经傅里叶变换到另一空间，即为周期信号。

2.2.2 从连续傅里叶变换到离散傅里叶变换

连续傅里叶变换的形式如下，通常所说傅里叶变换也指的是连续傅里叶变换：

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt \quad (2.1)$$

其中 ω 是弧度制单位下的频率， $f(t)$ 是原始信号， $F(\omega)$ 是经过变换后的像信号。连续傅里叶变换是通过傅里叶级数变换成指数形式后，在周期趋向于无穷后求极限推导得出的，适用于分析时间域上连续且呈非周期分布的信号。

工程中音频信号的获取通过采样得到，连续时间信号 $f(t)$ 以及对应的连续傅里叶变换 $F(\omega)$ 都是连续函数。因为数字信号处理设备或系统只能分析长度有限的离散信号，因此必须离散化 $f(t)$ 和 $F(\omega)$ ，并建立相应的傅里叶变换。因此连续信号只是理想的存在，连续时间傅里叶变换和傅里叶级数处理实际信号并不可行。

离散时间傅里叶变换（DTFT, Discrete Time Fourier Transform）也叫序列傅里叶变换，它使得我们可以分析实际工程情形下采样得到的离散时间信号，在频率域内得到信号的频谱。

离散时间傅里叶变换形式如下：

$$X(\omega) = \sum_{n=-\infty}^{+\infty} x[n]e^{-i\omega n} \quad (2.2)$$

其中 ω 是弧度制下的频率变量， $x[n]$ （ n 为所有整数）是一组离散的实数或复数， $X(\omega)$ 是离散时间傅里叶变换后的连续周期函数。所以，离散时间傅里叶变换就是通过傅里叶级数变换一组离散的实数或复数从而产生以频率为变量的连续周期函数。

离散时间傅里叶变换使得分析实际工程条件下的离散采样信号变得可行，但仍然存在两个实际问题：首先，数字信号的频率是一个模拟量，所以为了对信号进行数字的分析和处理，不仅需要在时间域将时间变量进行离散化，同样需要在频率域将频率分量离散化。其次，实际的信号序列大部分是无限长的，为了分析和处理的方便，必须把无限长序列进行截取，把它变成有限长的序列来分析和处理。

为了实质性地解决上述问题，应当选择离散傅里叶变换（DFT, Discrete Fourier Transform）进行实际采样信号的分析处理。离散傅里叶变换把现实中无限长的序列中的一段有限长序列看作是周期序列的一个周期后，对其进行傅里叶变换，与变换后得到连续频谱的离散时间傅里叶变换不同，其变换后的频谱是离散的。因此，不管在时间域还是频率域，它都是有一段有限长序列。在频率域上，DFT 的离散谱就是对 DTFT 连续谱的等间隔采样。

对于一个 N 点序列 $\{x[n]\}_{0 \leq n \leq N}$ ，它的离散傅里叶变换的形式为：

$$X[k] = \sum_{n=0}^{N-1} e^{-i\frac{2\pi}{N}nk} x[n], k = 0, 1, \dots, N-1 \quad (2.3)$$

在运用离散傅里叶变换时，应当注意两点：

(1) 为了离散傅里叶变换的频率域结果可以复现原始时间域的实际信号，必须确保对信号采样时遵循采样定理；(2) 运用离散傅里叶变换处理的信号必须是有时限的，因此首先要有时限的信号进行周期延拓。

2.2.3 工程实用方法——快速傅里叶变换

快速傅里叶变换（Fast Fourier Transform）是对科学有效计算离散傅里叶变换的实用方法的总称。采用快速傅里叶变换可以使得计算过程的乘法次数显著减少，使得利用离散傅里叶变换对工程实际信号进行分析变得更可行。在信号的采样点数目越大时，快速傅里叶变换的方法在节省计算时间上的效果就会越显著。

2.2.4 信号幅值域的分析 and 采样频率的确定

采集朗诗机房运行的中区热水二次泵 1 号的音频信号进行分析，采集过程中采用了 44100Hz、22050Hz 和 11025Hz 三种不同的采样频率。

在采样满足采样定理的条件下，快速傅里叶变换结果的实际展示范围是半奈奎斯特频率域，因为快速傅里叶变换后得到的有效频率域是原信号采样频率的一半。如果原信号中有超过采样频率一半值的信号，则快速傅里叶变换结果无法展示真实的频率域，也即采样信号无法完全复现原始信号。

利用 Python 的傅里叶变换对信号进行快速傅里叶变换，并根据变换结果判断采用哪个采样频率的采样信号进行后续的音频信号分析更合理。

(1) 采样频率为 44100Hz 的信号：

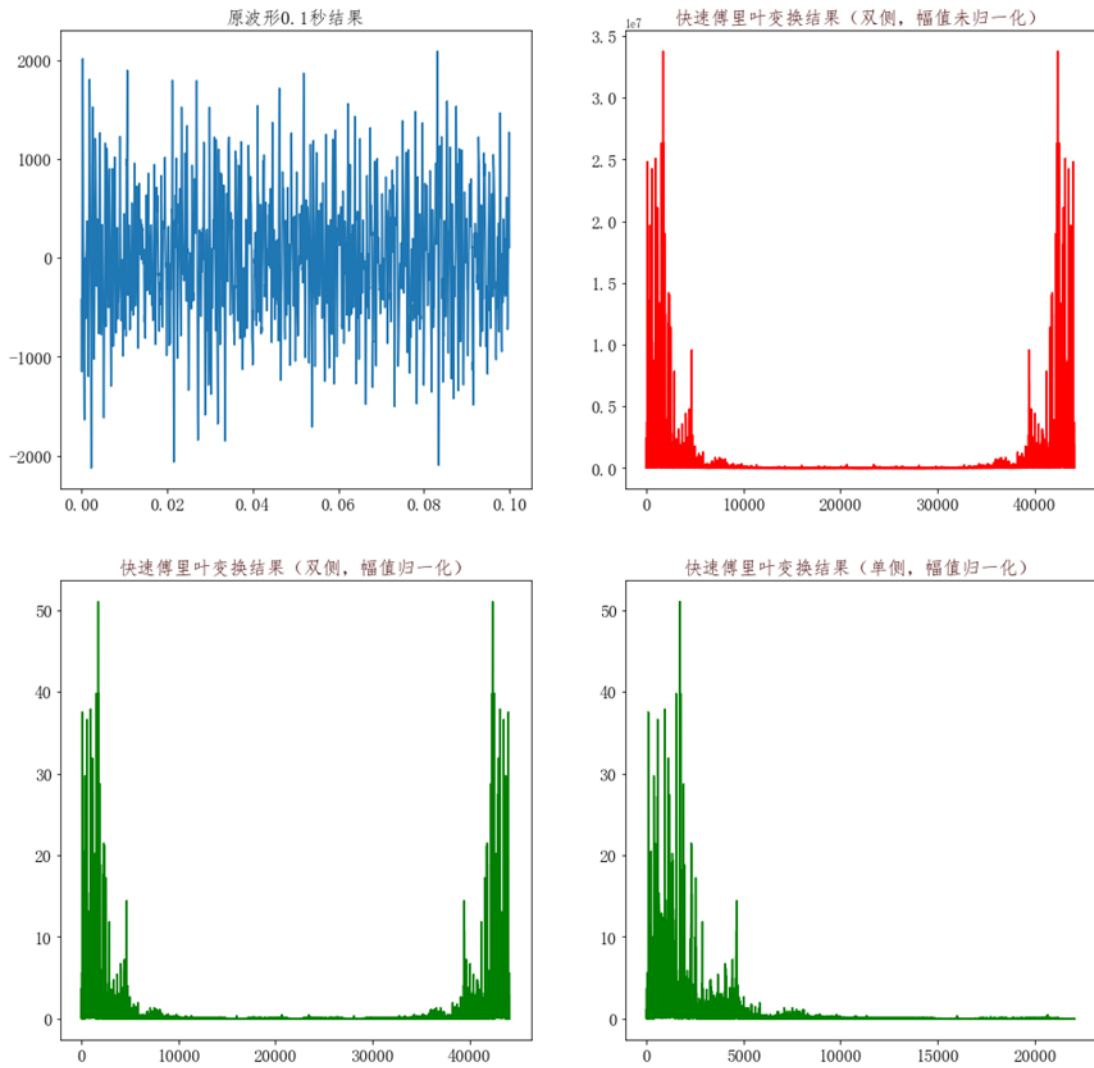


图 2.2 采样频率为 44100Hz 信号的傅里叶变换结果

(2) 采样频率为 22050Hz 的信号:

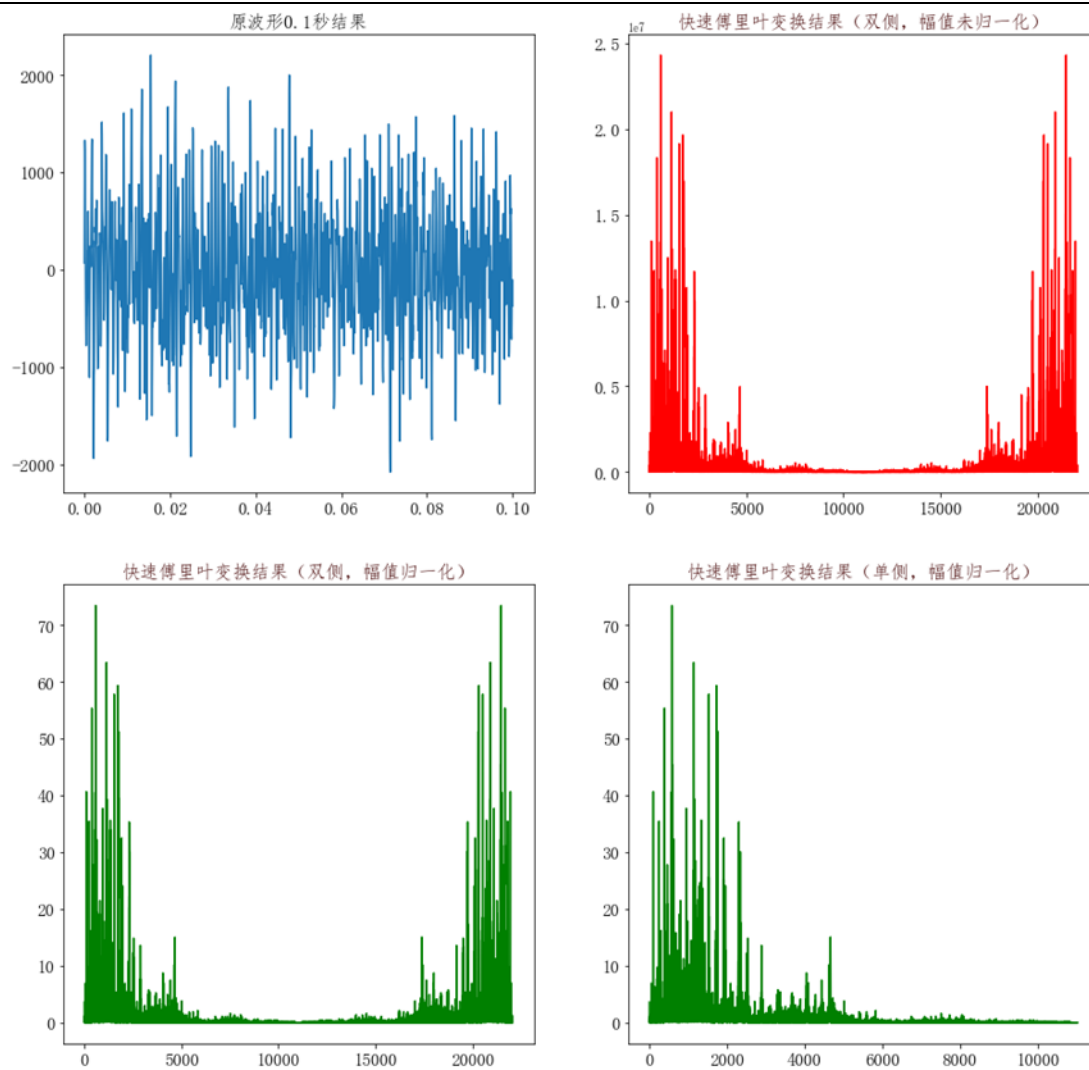


图 2.3 采样频率为 22050Hz 信号的傅里叶变换结果

(3) 采样频率为 11025Hz 的信号:

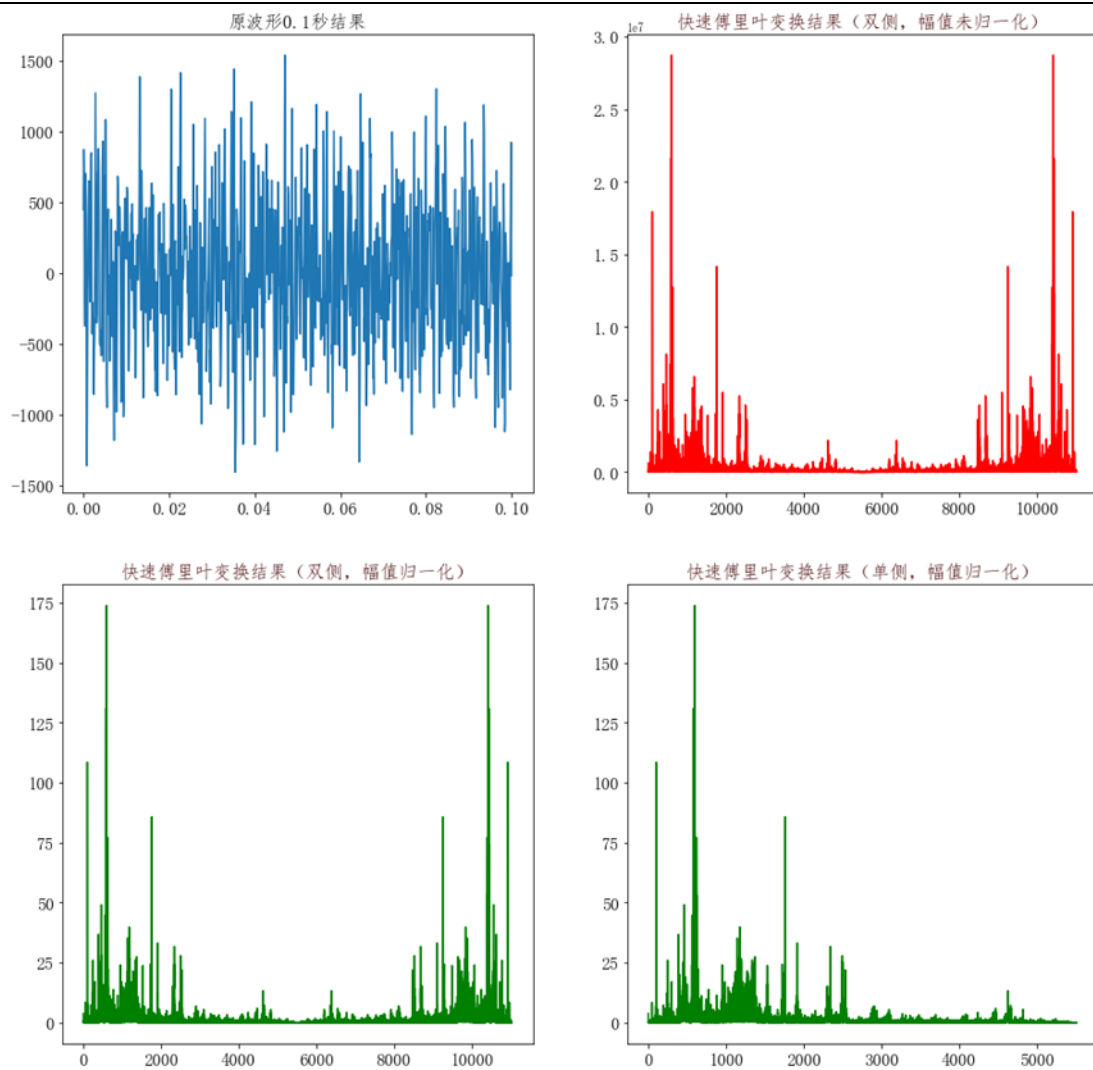


图 2.4 采样频率为 11025Hz 信号的傅里叶变换结果

对三个不同采样频率的同一音频 20 秒时间的采样结果，分析快速傅里叶变换后频率域内的信号结果，可以看出中区热水二次泵的音频信号的主要频谱集中在 3000Hz 以内，所以利用 11025Hz 的采样信号作为基础分析信号是一个相对较好的选择。保证了重要频率域内的信号变换结果可靠，且采样不会冗余浪费。

2.3 信号的滤波

将时域的音频信号变换到频率域后，要利用滤波算法进行频率域滤波，滤波可以使我们获得想要的频率域函数，并将想要的频率域函数还原到原始的时间域，从而实现了信号的分离。

本小节利用 MATLAB 设计了一个可以音频信号滤波的 GUI 应用。这里需要对读入 MATLAB 和 Python 这两种环境下的音频信号的类型进行说明，在 MATLAB 中读入的音频信号默认的数据类型为 double，双精度型，读入后，音频信号的幅值会自动进行正则化处理。MATLAB 中的 sound 函数可以播放音频文件，但必须要求读入的音频信号的数据类型为 double，双精度类型。因此在 MATLAB 中，音频信号的幅值区间为[-1,+1]。在 Python 中展示的音频信号的数据类型均为 int16，16 位整型。但尽管音频信号的幅值表示区间不同，在 MATLAB 和 Python 中读入的同一音频信号均一样，不会存在任何分析差异。

2.3.1 信号滤波的基本概念

根据作用信号和使用设备的不同，滤波器包含模拟滤波器和数字滤波器两大类。模拟滤波器通过组合电子器件如电阻、电容和电感等构成，而数字滤波器是通过数学公式对数字信号进行滤波。

模拟滤波器位于模数转换器 A/D 之前：



图 2.5 模拟滤波器简图

数字滤波器位于模数转换器 A/D 之后：

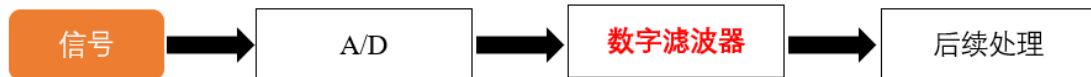


图 2.6 数字滤波器简图

正是由于这样的安放位置不同，模拟滤波器系统相比数字滤波器系统有两大优势。首先，模拟滤波器系统的模数转换器 A/D 在模拟滤波器的后面，若模拟滤波器是一个低通滤波器，即我们希望获得的信号是低频信号的话，模数转换器在被滤波后的信号处只需要进行低频采样就可以。如此相反，为保证原始信号的真实性，数字滤波系统的模数转换器 A/D 必须进行高频采样。因此，模拟滤波器计算负荷相比数字滤波器要小。其次，若模拟滤波器是一个高通滤波器，即我们希望获得的信号是高频信号的话，模拟滤波器系统可以先滤除幅值较高的低频信号，对滤波后的高频信号进行幅值放大，进而可以充分利用模数转换器 A/D 的满量程精度。相反，为了保证模拟信号向数字信号转换过程中，低频信号的量程不会溢出，数字滤波器系统的模数转换器 A/D 的满量程精度不能被有利用，所以其有效位数比较低。

尽管模拟滤波器有上述的优势,但在工程应用和利用计算机进行信号分析的领域上,先对信号进行采样,再进行数字滤波的信号分析方法广泛应用。主要是因为数字滤波器的设计不需要设备成本,只需要基于数学公式就可以设计出满足自己需要的数字滤波器。

所以,在本论文研究的音频信号分析中,利用数字滤波器对快速傅里叶变换后的信号进行数字滤波。

滤波器基本概念涉及到时域和频域间的变换。原始信号经过滤波器后变成滤波信号,在频率域可以解释为滤波信号的频谱等于原始信号的频谱和滤波器频谱的乘积:

$$Y(f) = X(f)H(f) \quad (2.4)$$

在时间域可以理解为滤波信号等于原始信号和滤波器信号的卷积:

$$y(t) = x(t) * h(t) \quad (2.5)$$

这是因为滤波器的直观作用是在频率域内进行乘积运算,获得我们希望得到的原始信号的某一频段的信号成分。

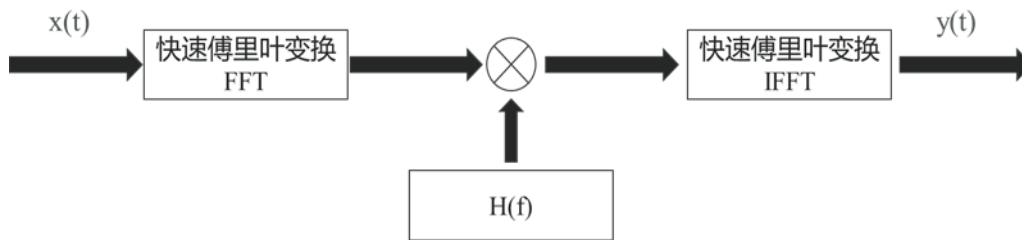


图 2.7 滤波器流程示意图

2.3.2 有限冲激响应滤波

有限冲激响应 (Finite Impulse Response) 滤波器简称 FIR 数字滤波器。如果给有限冲激相应滤波器一个脉冲函数,滤波器的冲激响应函数是一个有限长度的序列,即不是无限振荡的过程,即有限冲激响应滤波器对于脉冲输入信号的响应最终趋向于 0。

有限冲激响应滤波器的传递函数可用 Z 变换形式表述,Z 变换是处理离散时间系统的变换,可将离散时间系统的差分方程形式的时间域数学模型转化为更简单的代数方程形式的频率域数学模型。形式如下, N 为所设计的滤波器阶数:

$$H(z) = \sum_{n=0}^N b_n Z^{-n} \quad (2.6)$$

其中,冲激响应,也就是所设计的滤波器系数为:

$$h(n) = b_n, n = 0, 1, 2, \dots, N \quad (2.7)$$

可见, h(n) 是一个有限长度的序列,有限冲激响应滤波器因而得名。

MATLAB 的 FIR 数字滤波器包含 `fir1` 和 `fir2` 两类创建函数。`fir1` 函数基于窗函数构建 FIR 滤波器，`fir2` 函数基于频率构建 FIR 滤波器。在构建时，`fir2` 函数比 `fir1` 函数增加了过渡点，因此暂态现象较 `fir1` 函数更弱，对于复杂的时间域信号，它的滤波效果也优于 `fir1`。因此，我们选择 `fir2` 函数来设计滤波器，它主要需要输入的参数是：滤波器阶数，滤波器频率—幅值特性对。输出参数是：滤波器的系数，即滤波器的冲激响应。`fir2` 函数的计算过程是利用滤波器频率—幅值特性对进行理想频率响应的插值，然后使用添加 Hamming 窗函数的傅里叶逆变换求取滤波器的系数。

需要注意的是，在利用 `fir2` 函数设计滤波器时，要对求取的滤波器系数进行截断，截断长度即为设计阶数，截断后得到有限冲激响应。而后对其快速傅里叶变换即得到实际的频率—幅值特性。因此。利用 `fir2` 函数设计好的滤波器是理想滤波器，在 MATLAB 中利用 `freqz` 函数可以查看利用 `fir2` 函数设计的滤波器的实际频率—幅值特性。

2.3.3 对采集的信号进行滤波处理

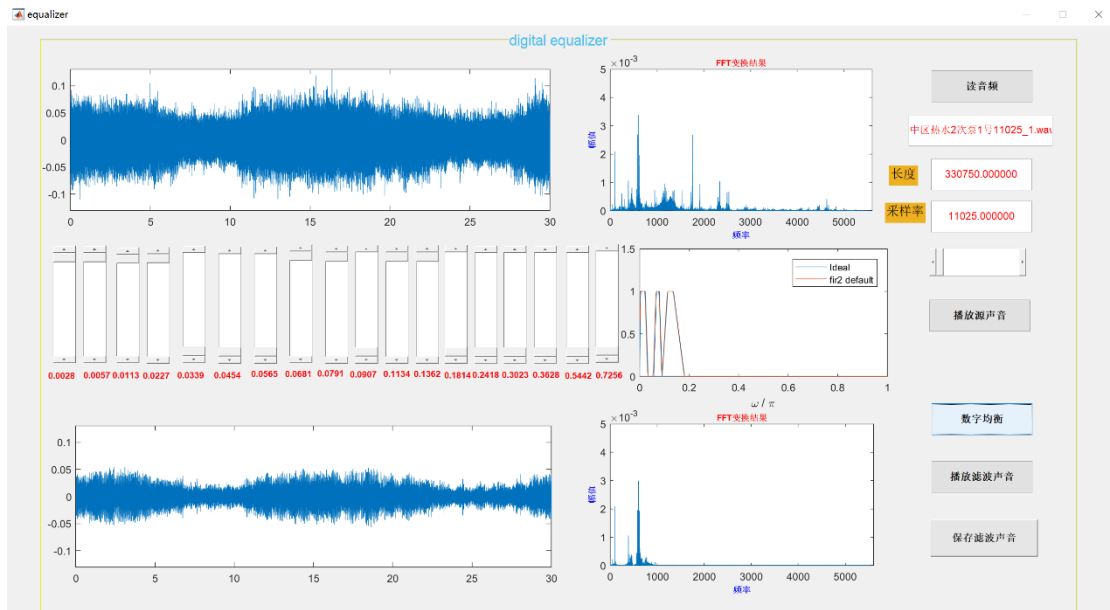


图 2.8 滤波器 GUI 界面

利用 MATLAB 的 GUI 工具设计一个滤波界面，包括导入原始信号，截取长度为 30s 的信号，然后展示信号的 FFT 结果。通过分析原始信号的频谱特征，利用数值滑动条来选择对应频率特征下的幅值特征，从而设计出自己需要的 FIR 数字滤波器。之后，利用滤波器对原始信号进行滤波处理，得到我们希望获得的频率成分的信号。利用这一 GUI，可以灵活的将原始信号按照其频率组分滤波成多

个信号，并保存滤波结果，便于我们分析声音中的不同频率特性，从而对声音信号进行模型建立。

2.4 信号的滤波实例及结果分析

设置 fir2 滤波器的 17 个滑动条对应的频率点，频率点的取值范围为从 0 到 1，是滤波器实际频率域的百分比，对应滤波频率为 0Hz 到采样频率的一半值（快速傅里叶变换能分析的是原信号的采样频率的一半）。在 2.2 节选择的合理的采样频率为 11025Hz，则滤波器的频率间断点对应的滤波频率值是 5512.5Hz 与间断点数值的乘积。

2.4.1 中区热水二次泵 1 号滤波实例

下面对朗诗绿色街区的冷冻机房内的中区热水二次泵 1 号的 11025Hz 采样频率的音频信号为对象进行滤波，并分析相应的滤波结果。通过调节 17 个滑动条的位置，构造不同的 fir2 滤波器，对原始音频信号进行滤波，以获取不同频段下的信号分量。主要滤波思路为根据原始信号经过 FFT 后的频率域结果，按照幅值大小和频率域结果的分布情况，将低频信号、中频信号和高频信号分开，这其中幅值较高的信号为有研究意义的信号，是机器发出的主要声音信息，幅值较低的部分是机器本身带来的各种其他声音信息，还有一部分是环境中其他机器传播来的噪声信号。

低频信号的部分包括了低频段的最高峰值部分，将滤波器 GUI 的 0.097 之前的滑动条全部推到 1，表明滤波的频率域范围约为 0-535Hz，保存滤波后的信号，并分析滤波后信号的时间域结果和经过 FFT 后的频率域结果，如下：

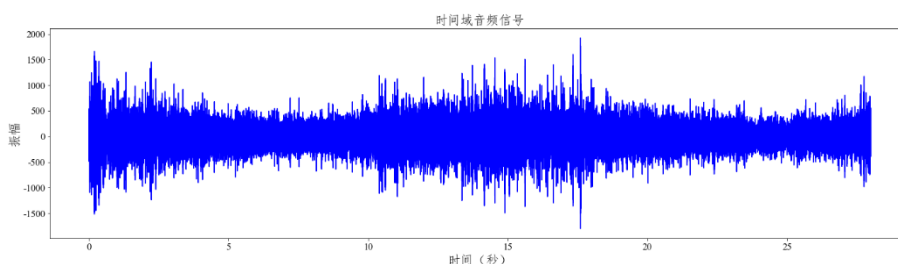


图 2.9 低频信号的时间域结果

截取时间域上 0.1 秒的结果和单双侧 FFT 后的结果，可以看出低频段的信号相对振荡稀疏一些。

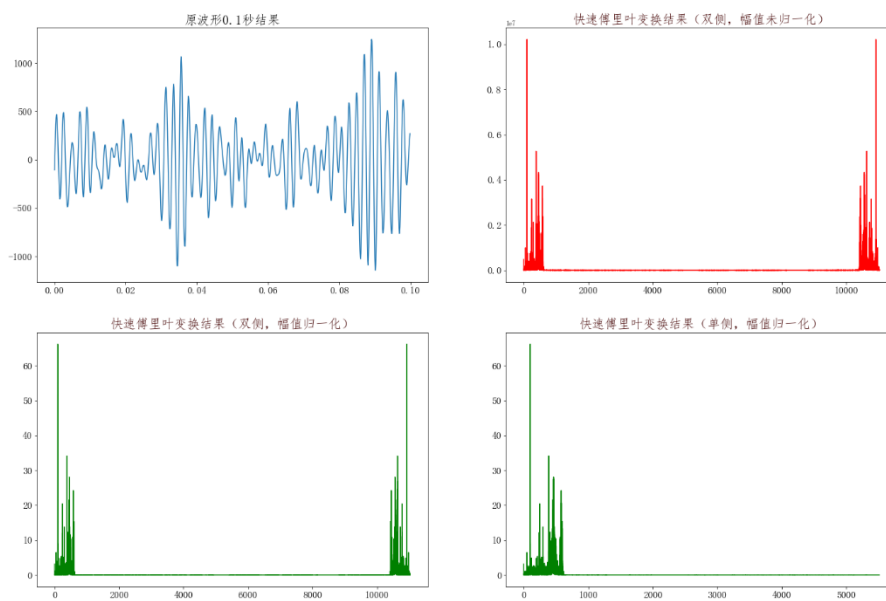


图 2.10 低频信号综合展示

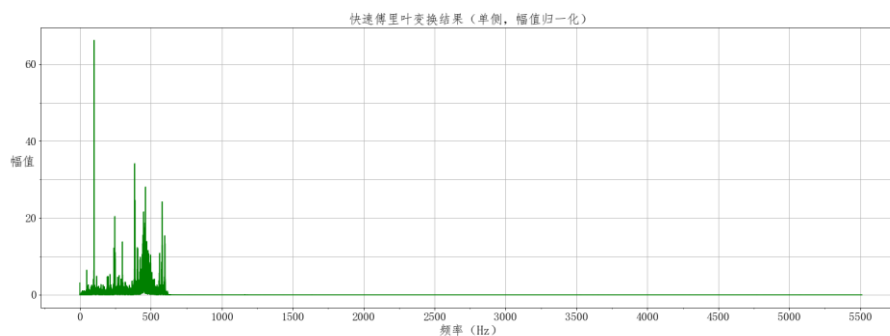


图 2.11 低频信号 FFT 单侧区间且幅值归一后结果

中频信号的部分包括了中频段的最高峰值部分，将滤波器 GUI 的 0.1134 到 0.2418 之间的滑动条全部推到 1，表明滤波的频率域范围约为 625Hz 到 1333Hz，保存滤波后的信号，并分析滤波后信号的时间域结果和经过 FFT 后的频率域结果，如下：

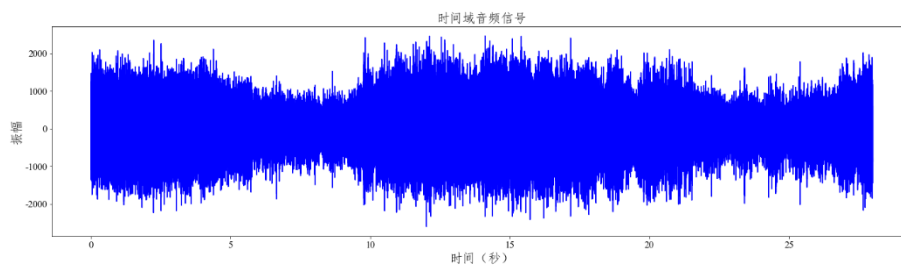


图 2.12 中频信号时间域结果

截取时间域上 0.1 秒的结果和单双侧 FFT 后的结果，可以看出中频段的信号相对振荡密集一些。

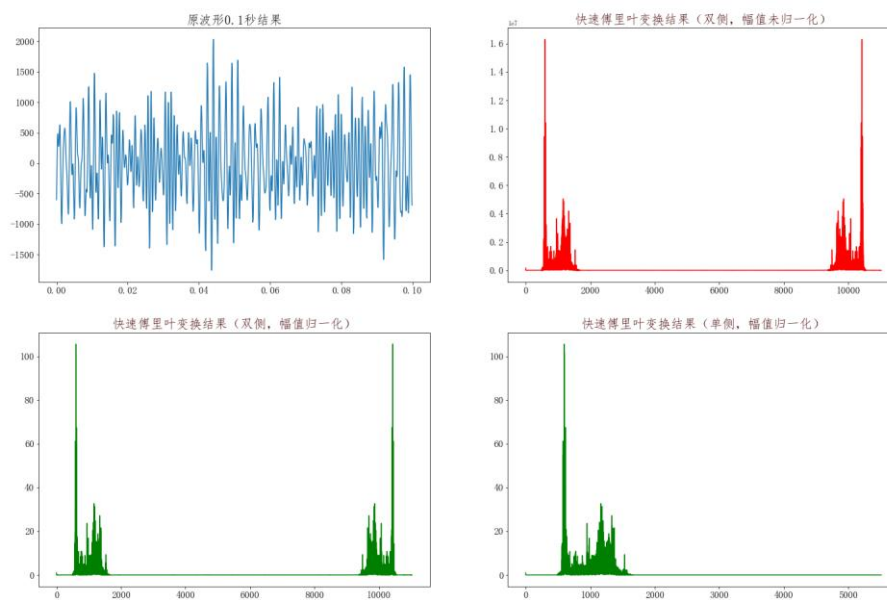


图 2.13 中频信号综合展示

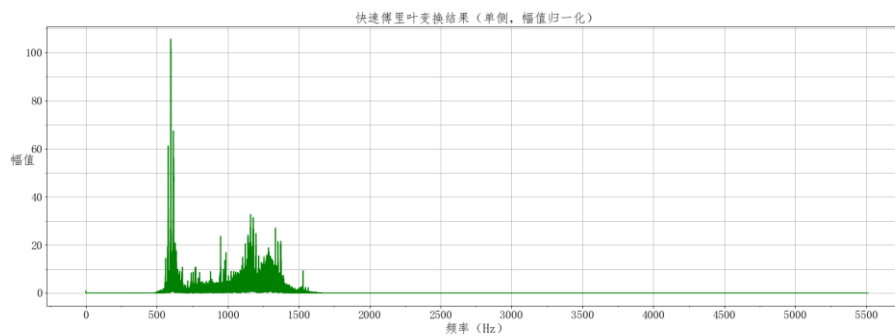


图 2.14 中频信号 FFT 单侧区间且幅值归一后结果

高频信号的部分包括了高频段的最高峰值部分，将滤波器 GUI 的 0.3023 到 0.7256 之间的滑动条全部推到 1，表明滤波的频率域范围约为 1666Hz 到 4000Hz，保存滤波后的信号，并分析滤波后信号的时间域结果和经过 FFT 后的频率域结果，如下：

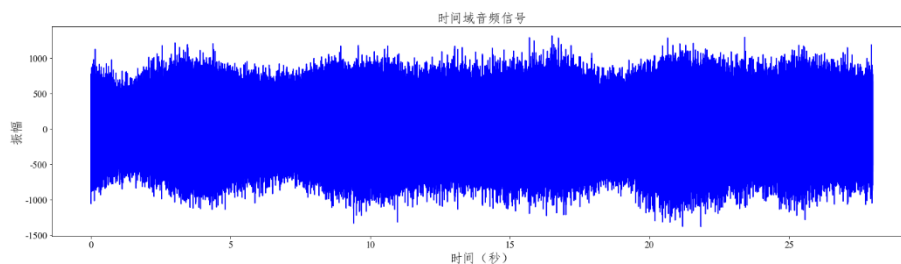


图 2.15 高频信号时间域结果

截取时间域上 0.1 秒的结果和单双侧 FFT 后的结果，可以看出高频段的信号振荡更加密集一些。

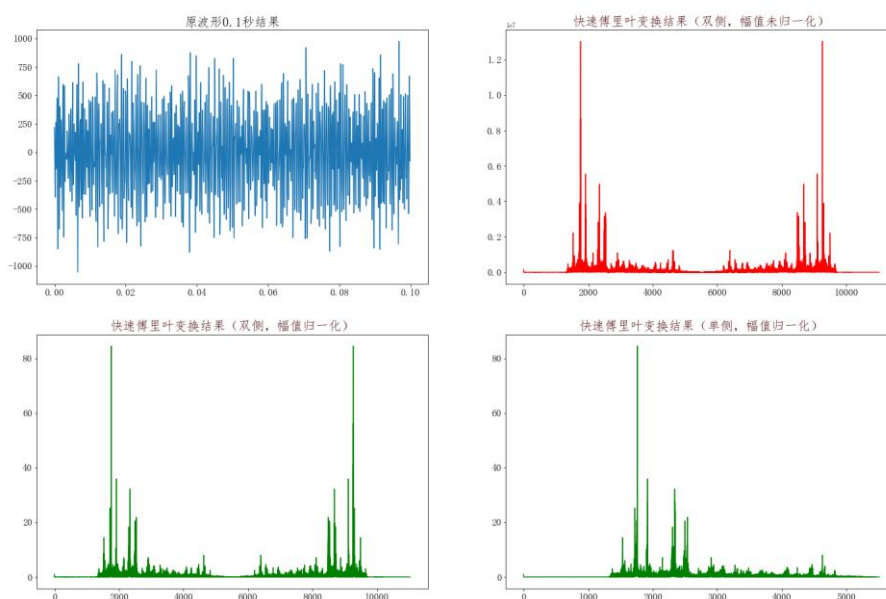


图 2.16 高频信号综合展示

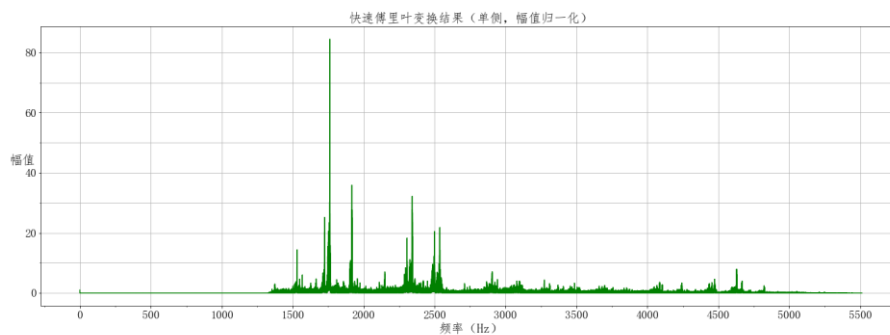


图 2.17 高频信号 FFT 单侧区间且幅值归一后结果

除了将原始音频信号经过 FFT 后的结果分成低频、中频和高频信号外，在特定的幅值凸处也进行特定频率区间的滤波，分别是低频信号的峰值、中频信号的幅值和高频信号的峰值频率处。这里选择低频峰值信号和低频峰值信号进行展示

将滤波器 GUI 的 0.0227 之前的滑动条全部推到 1，保存滤波信号获得新的音频信号代表低频信号的峰值幅值频率域部分，简记为低频峰值信号。与对低频、中频和低频信号的分析过程类似，低频峰值信号的时间域结果和经过 FFT 后的频率域结果，如下：

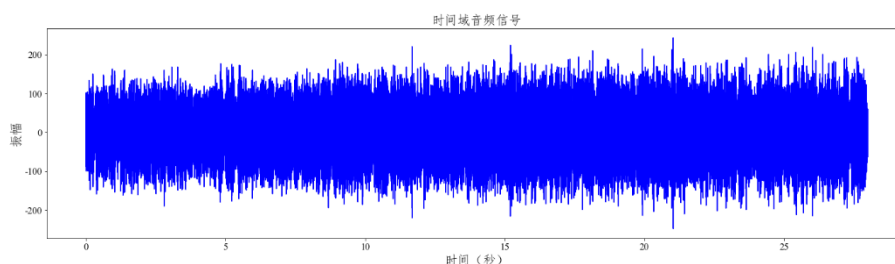


图 2.18 低频峰值信号的时间域结果

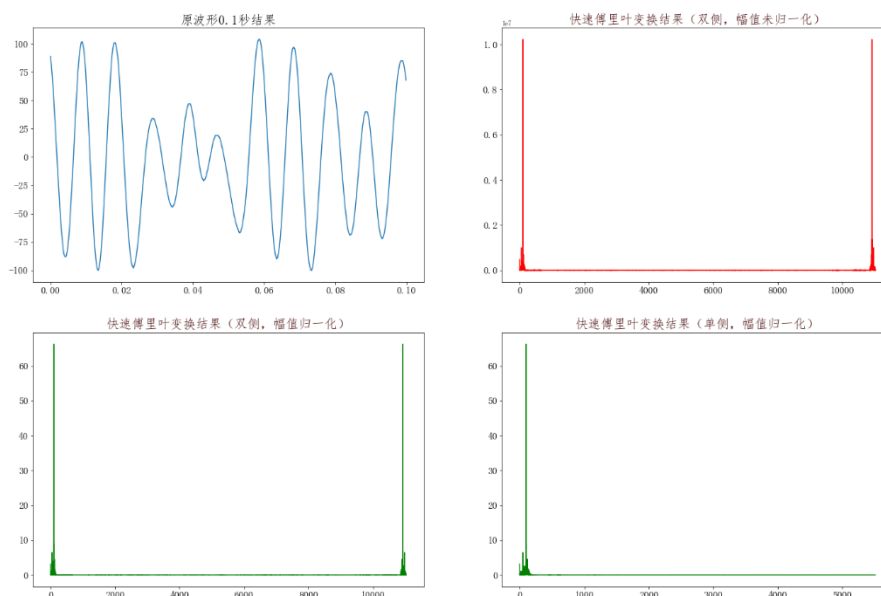


图 2.19 低频峰值信号综合展示

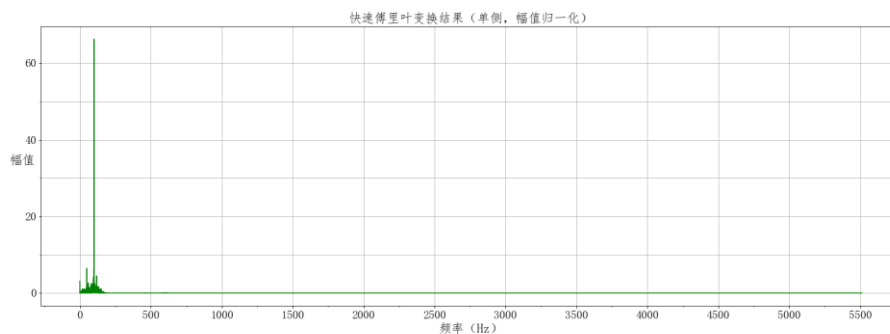


图 2.20 低频峰值信号 FFT 单侧区间且幅值归一后结果

将滤波器 GUI 的 0.3023 和 0.3628 的滑动条推到 1，保存滤波信号获得新的音频信号代表高频信号的峰值部分。高频峰值信号的时间域结果和经过 FFT 后的频率域结果，如下：

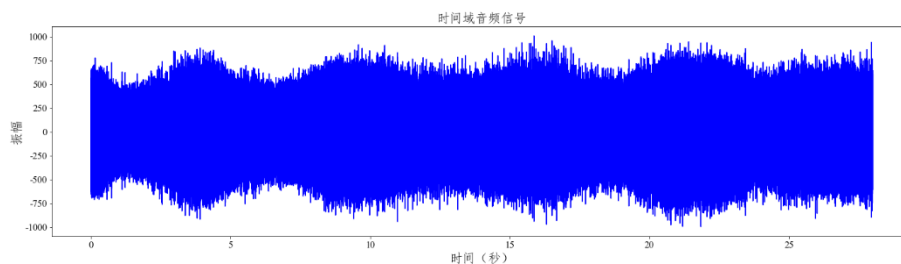


图 2.21 高频峰值信号的时间域结果

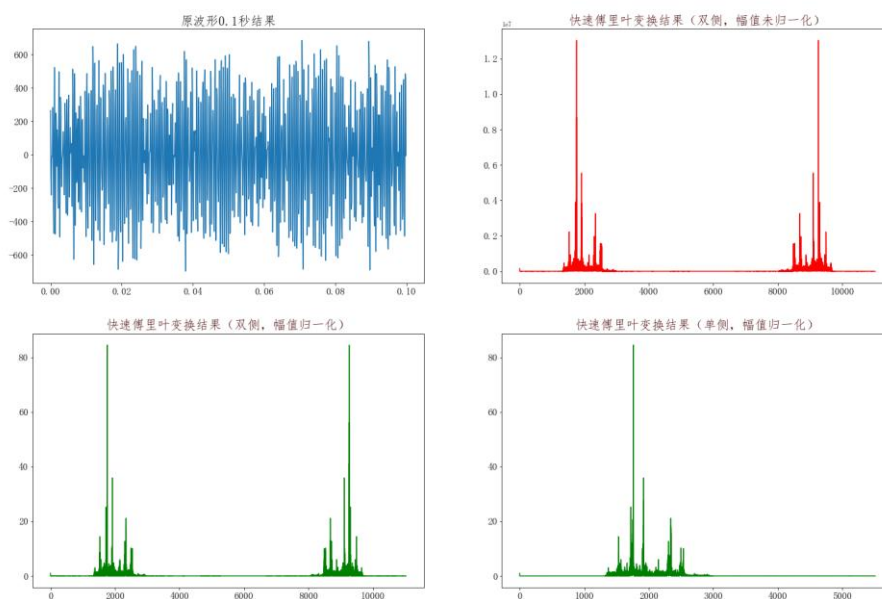


图 2.22 高频峰值信号综合展示

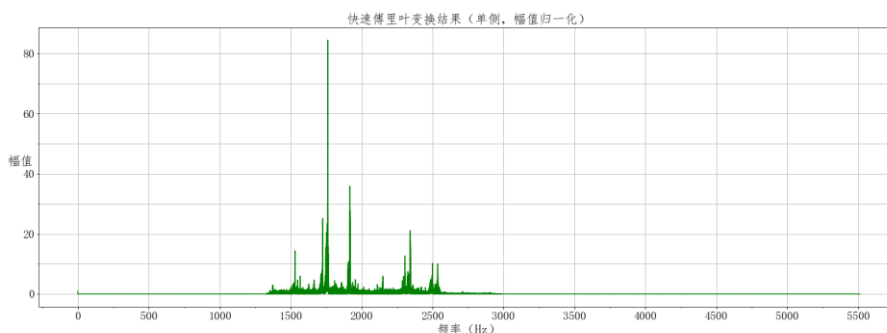


图 2.23 高频峰值信号 FFT 单侧区间且幅值归一后结果

2.4.2 实例结果分析

将原始的水泵音频信号通过滤波分解为低频、中频和高频信号后，分析发现每段信号的频率域除了幅值最大的峰值频域部分，在频域区间内几乎所有频域段都有对应的或大或小的幅值。将信号的时域区间截取 0.1 秒出来，可以发现从低频到高频信号，振荡程度逐渐变强。同时，观察低频、中频和高频信号的时间域，它们时间域上的幅值并不相同，反应了各部分信号在原始信号中的比重不同，中频信号相对比重大一些。

再将低频、中频和高频信号的频率域幅值峰值处周围的信号通过滤波提取出来得到低频峰值、中频峰值和高频峰值信号后，观察截取的低频和高频信号 0.1 秒时间区间段结果，可以看出滤波后信号的频率区间范围变窄，信号含有的频率组分变少，因此在时间域上的信号振荡也变弱，而且更加有规律。

综上所述可以得出结论：

- (1) 原始水泵信号经过 FFT 后的频谱组成成分复杂，在一些频域区间有较高的幅值，在大部分的频域区间的幅值不高。可以推断，较高的幅值一定程度上反映了机器在这一频域区间的音频信号响度大，是反映机器特性的音频信号之一。
- (2) 同时，除了在一些频域区间有较高的幅值外，在其他几乎所有频率域上都有不同大小的幅值，这表明原始水泵信号的频谱成分很复杂，相对来说缺少规律性，是由主要频谱成分和其他附加频谱成分包括机器本体发出的次要声音信号以及环境混杂的声音信号等多方面组成。对于这部分附加频谱成分，由于很难确定其来源，且它的幅值参差不齐，因此不能断定地认为就是无用信号成分。

2.5 本章小结

本章首先介绍了声音信号采样的设备和遵循的采样定理，而后对信号的幅值域分析方法傅里叶变换进行了详细分析，并利用工程实用的快速傅里叶变换对示例信号进行处理确定合适的采样频率作为后续研究的基准。之后介绍了信号有限冲激相应滤波的概念，并制作相应的图像交互界面，来实现滤波功能。最后将信号以低频、中频和高频为三个不同成分进行滤波处理，并展示相应结果。

第3章 基于时间序列分析处理音频的水泵故障分析

上一章中，将采集的水泵音频信号按照频率域分为低频、中频和高频三部分的信号，并分析了相应的结果。由于采集的原始信号是音频信号而不是泵体的振动信号，所以信号频谱成分复杂且具有一定的迷惑性。幅值较高的频域区间有多个，如果断定用特定幅值较高的频域区间代表机器的幅值特性，具有描述难度。同时结合本论文的算法开发最终希望利用机器人巡检进行状态监测式的故障诊断，对特定的故障识别并没有要求。因此采用时间域的信号分析法更适合处理本论文的问题。

因此，本章将分别分析不同频段的水泵音频信号，通过建立时间序列模型的方法，将各个频段的音频信号分为有时间相关性的部分和无明显时间相关性的随机波动部分。然后通过建立阈值故障诊断模型的方法，开发基于时间序列分析音频信号的冷冻机房水泵故障诊断算法。

3.1 信号的时间序列分析

时间序列是依照时间顺序排列的一组数据序列。我们进行时间序列分析，是希望其中的变化规律，并因此而指导对未来变化的预测。利用时间序列进行分析有满足3个要求：发展趋势会持续到将来；分析的数据应该是不规则的；忽略事物发展前后的相互因果关系。

进行时间序列分析之前，应判断该时间序列是否是平稳，平稳时间序列分析通常借助 ARMA 模型；而非平稳时间序列分析包括确定性时序和随机时序分析法两大类，前者对时间序列按照特定的因素进行分解，而后者通常借助运用了差分手段的 ARMA 模型，即 ARIMA 模型。

3.1.1 平稳时间序列的分析方法

ARMA 模型是自回归模型和移动平均模型的组合，数学上可以简记为 ARMA(p,q)模型。经过中心化后，ARMA 模型的形式为：

$$\begin{cases} x_t = \phi_0 + \phi_1 x_{t-1} + \dots + \phi_p x_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \dots - \theta_q \varepsilon_{t-q}, \phi_p \neq 0, \theta_q \neq 0 \\ E(\varepsilon_t) = 0, \text{Var}(\varepsilon_t) = \sigma_\varepsilon^2, E(\varepsilon_t \varepsilon_s) = 0, s \neq t \\ E(x_s \varepsilon_t) = 0, \forall s < t \end{cases} \quad (3.1)$$

ARMA 模型的表述形式中，第一项保证了模型的基本形式，其中 ϕ 和 θ 均为模型的系数，同时限制了模型的阶数；第二项限定了随机扰动序列 $\{\varepsilon_t\}$ 必须为零均值白噪声序列；第三项限定当期的随机扰动应与以往的序列值无关。

ARMA 模型由 AR 模型和 MA 模型两部分组成，AR 模型是前一定期数的时间序列自身拟合，得到自回归形式的时间序列预测模型，MA 模型是前一定期数自回归 AR 模型拟合结果与实际结果的误差作为变量拟合得到的移动平均的时间序列预测模型。因为两种模型在统计上的自相关系数和偏自相关系数呈现出不同的特征，所以结合两种模型对时间序列模型拟合是一个不错的选择。关于根据时间序列的自相关系数图和偏自相关系数图选择模型的具体阶数的内容在后面会进行介绍。

引进作用起到时间延迟作用的延迟算子 B ，把它与时间序列的当前值相乘，就相当于把当前序列值变为过去一个时刻的序列值，如： $x_{t-1} = Bx_t$ 。引入延迟算子后，ARMA(p,q)模型的表述形式简化为：

$$\Phi(B)x_t = \theta(B)\varepsilon_t \quad (3.2)$$

其中，

$$\Phi(B) = 1 - \phi_1 B - \dots - \phi_p B^p \quad (3.3)$$

$$\theta(B) = 1 - \theta_1 B - \dots - \theta_q B^q \quad (3.4)$$

3.1.2 非平稳时间序列的确定性时序分析方法

确定性时序分析中，根据 Cramer 分解定理，任何一个时间序列 $\{x_t\}$ 都可以分解为两部分的叠加和：其中一部分是由多项式决定的确定性趋势成分，另一部分是平稳的零均值误差成分^[55]，可以用公式表示为：

$$x_t = \mu_t + \varepsilon_t = \sum_{j=0}^d \beta_j t^j + \Psi(B)a_t \quad (3.5)$$

其中， μ_t 是均值部分序列， ε_t 是误差部分序列； d 是有限正整数； β_1, \dots, β_d 为多项式的常数系数； $\{a_t\}$ 是零均值的白噪声序列； B 是延迟算子， $\Psi(B)$ 是延迟算子对应的多项式序列。

误差项的期望 $E(\varepsilon_t)$ 等于 0，均值序列反映了时间序列 $\{x_t\}$ 受到的确定性影响，而误差序列反映了时间序列 $\{x_t\}$ 受到的随机性影响

通常利用确定性因素分解方法来进行确定性时间序列分析。根据确定性因素分解方法，可以将原始时间序列分解成四大类子因素，分别代表了时间序列的长期趋势、循环波动、季节性浮动和不稳定随机浮动成分。其中对于一般时间序列，季节性浮动常常被归为循环浮动内，可看成是时间序列的周期因素。实际应用中，可以通过加法模型和乘法模型来体现这几个因素的相互作用关系

加法模型描述为：

$$x_t = T_t + P_t + I_t \quad (3.6)$$

乘法模型描述为:

$$x_t = T_t \cdot P_t \cdot I_t \quad (3.7)$$

其中, T_t 为长期趋势成分, P_t 为周期波动成分, I_t 为随机浮动成分

在常见的确定性因素分解后, 最重要的是要对长期趋势成分 T_t 进行拟合, 一般的拟合方法有趋势拟合法和平滑法, 趋势拟合法包括线性拟合和曲线拟合: 即用一般的多项式公式进行拟合; 平滑法包括移动平均法和指数平滑法: 移动平均法利用一定时间间隔内的平均值作为某一期的估计值, 而指数平滑法考虑到时间间隔对事件发展的影响, 拟合时会通过权重强化近期的序列值而弱化远期序列值。

然而实际应用中, 确定性因素分解方法虽然提取信息便利高效, 但提取的信息比较难以充分反映原始数据。因此, 为了弥补确定性因素分解的信息提取的不充分性, 可以对原始时间序列或原始时间序列的趋势成分 T_t 则采用随机时序分析方法, 即 ARIMA 模型方法进行时间序列拟合更能反映原始数据的变化去吃。

3.1.3 非平稳时间序列的随机时序分析方法

根据 3.1.2 节所述, 方差齐性的非平稳序列可以分解为多项式形式的确定性成分和均值为零的纯随机残差成分。在高等数学中, 离散序列的 d 阶差分过程就相当于连续变量的 d 阶求导过程, 我们可以通过 d 阶差分将原始序列中所蕴含的确定性成分充分提取:

$$\nabla^d \sum_{j=0}^d \beta_j t^j = c, \quad c \text{ 为常数} \quad (3.8)$$

差分阶数 d 的确定, 还要根据序列的不同特点来判断: 如果序列有显著的线性趋势, 只需要 1 阶差分就可以获得平稳序列; 如果序列有曲线变化的趋势, 可以经过 2 阶或 3 阶差分获得平稳序列; 如果序列中存在固有周期信息, 则可以借助步长为周期长度的差分来获取周期信息。

同时, 在实际应用差分方法时, 差分阶数并不是越多越好, 差分后的序列同样是平稳序列的条件下, 差分阶数越高, 差分后序列的方差越大, 实际是导致了有效信息的无谓浪费, 反而降低了估计精度。因此本论文中, 每次差分后, 都通过时间序列的平稳性检验方法来判断差分后的序列是否已经平稳, 从而决定是否有必要继续进行差分。

差分后可以表现出平稳序列性质的原始非平稳时间序列, 可以通过 ARIMA 模型进行拟合建模。ARIMA 模型是包含差分阶数参数的 ARMA 模型, 被称作求和自回归移动平均模型。数学上可以简记为 ARIMA(p,d,q)模型, 形式为:

$$\begin{cases} \Phi(B)\nabla^d x_t = \theta(B)\varepsilon_t \\ E(\varepsilon_t) = 0, \text{Var}(\varepsilon_t) = \sigma_\varepsilon^2, E(\varepsilon_t \varepsilon_s) = 0, s \neq t \\ E x_s \varepsilon_t = 0, \forall s < t \end{cases} \quad (3.9)$$

其中,

$$\Phi(B) = 1 - \phi_1 B - \dots - \phi_p B^p \quad (3.10)$$

$$\theta(B) = 1 - \theta_1 B - \dots - \theta_q B^q \quad (3.11)$$

从上式可以看出 ARIMA 模型相比 ARMA 模型, 唯一不同的地方就是第一项中对时间序列进行了差分。通过差分获得平稳序列, 再通过 ARMA 模型进行时间序列分析, 是 ARIMA 模型的核心思想。ARIMA 模型中的三大参数分别是 p , d 和 q , 其中, p 代表模型中自回归部分的阶数, d 代表模型差分阶数 (模型进行 d 阶差分后由非平稳序列变成了平稳序列若序列不需要差分便是平稳序列, 等价于序列 0 阶差分后平稳, 此时参数 d 为 0, ARIMA 模型也就退化为 ARMA 模型), q 代表模型移动平均部分的阶数。

3.1.4 时间序列分析信号实例

本章时间序列分析的数据来自上一章中区热水二次泵 1 号滤波后得到的低频、中频和高频信号部分, 对每一部分的信号分别建立时间序列模型进行分析, 最后通过设置时域上的幅值阈值的方法, 来监测异常声音判断水泵是否有故障。

同时, 由于确定性时间序列分析的对时间序列的质量和规律性要求高, 因此本章研究内容采用 ARMA 模型对平稳时间序列进行拟合建模, 采用 ARIMA 模型对非平稳时间序列进行拟合建模。

3.2 时间序列分析的平稳性检验

3.2.1 平稳性检验的方法

在分析时间序列前, 首要要对其平稳性进行统计检验, 以确定该序列是否是平稳时间序列。时间序列 $\{X_t\}$ 如果是平稳时间序列, 应该具有两大特性:

(1) 平稳时间序列的均值为常数:

$$EX_t = \mu, \forall t \in T \quad (3.12)$$

(2) 平稳时间序列的自协方差函数和自相关系数只依赖于时间的平移长度而与时间的起止时刻无关:

$$\gamma(t, s) = \gamma(k, k + s - t), \forall t, s, k \in T \quad (3.13)$$

现阶段时间序列的平稳性检验主要通过图检验方法和构造检验统计量进行假设检验,由于构造检验统计量进行检验比起图检验方法更为客观,所以本论文对所处理的信号时间序列采用构造检验统计量的方法进行检验,采用最为常用的单位根检验法。

单位根检验法通过检验时间序列特征方程的特征根是在单位圆内还是单位圆上或单位圆外,来判别时间序列的平稳性。初始单位根检验方法 DF 检验通过对原始时间序列构造一阶自回归过程,因此其适用范围小。得到修正后的增广 DF 检验——ADF 检验可以推广到多阶自回归过程,符合绝大多数实际时间序列的情形。得到一个时间序列,构造时间序列的 p 阶自回归过程,并求其特征方程的特征根 λ_i , 其中 $i = 1, 2, 3, \dots, p$ 。若该方程的全部特征根都落在单位圆内, 即:

$$|\lambda_i| < 1, i = 1, 2, 3, \dots, p \quad (3.14)$$

则时间序列平稳。

若存在一个特征根为单位根, 假设为: $\lambda_1=1$, 则时间序列非平稳。

此时单位根检验的检验假设条件可以表示为:

原假设 H_0 : 存在至少一个特征根是单位根, 此时时间序列非平稳; 备选假设 H_1 : 所有的特征根都在单位圆内, 此时时间序列平稳。所以在一定显著性水平下, 当能够显著地拒绝原假设 H_0 的时候, 该时间序列即为平稳序列。

3.2.2 对分析信号实例进行平稳性检验

首先将滤波得到的信号的开始 1 秒和结束 1 秒的数据删除, 留下采样时间为 28 秒的信号数据, 然后前 25 秒为训练数据, 后 3 秒为测试数据。同时, 由于采样频率较高, 数据点较多, 为了适用于实时地巡检故障诊断, 对原始的训练和测试数据进行降采样操作, 以固定的步长截取数据, 降采样率设置为 2 的倍数, 综合准确性和计算开销考虑, 设置降采样率为 256。

得到训练数据后, 根据 3.1.2 节描述的非平稳时间序列的随机时序分析方法同时对训练数据做一阶差分, 方便原始数据不是平稳时间序列的情况下, 对一阶差分的数据再做平稳性检验。下面是各个信号部分的截取开始 500 个数据点的训练数据和训练数据的一阶差分以及包括全部数据点的测试数据。

低频部分信号如下:

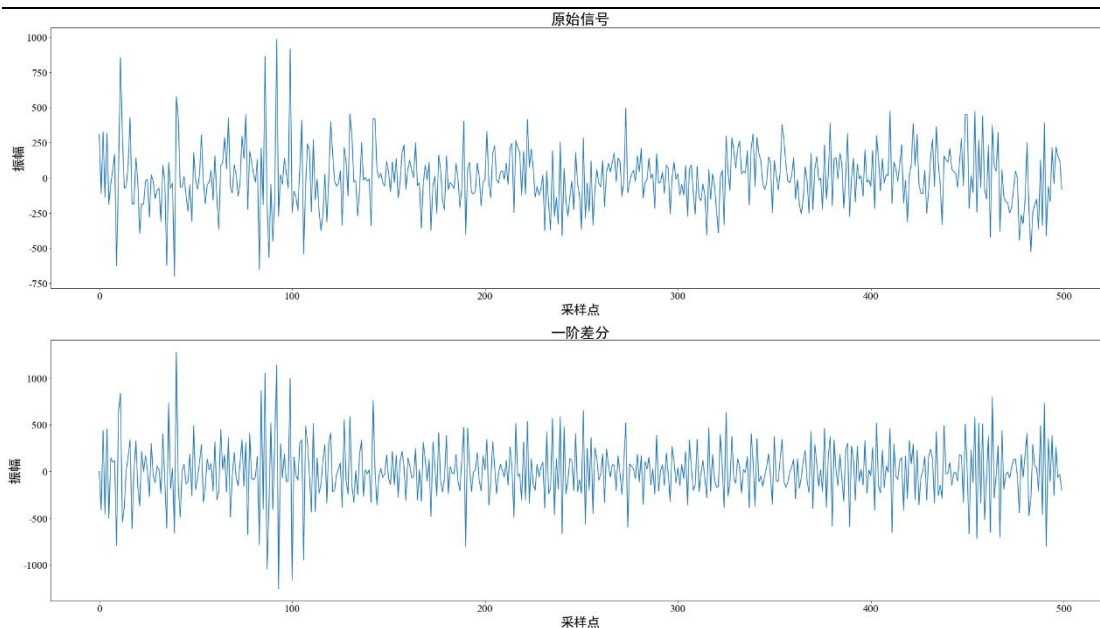


图 3.1 低频信号训练数据和一阶差分

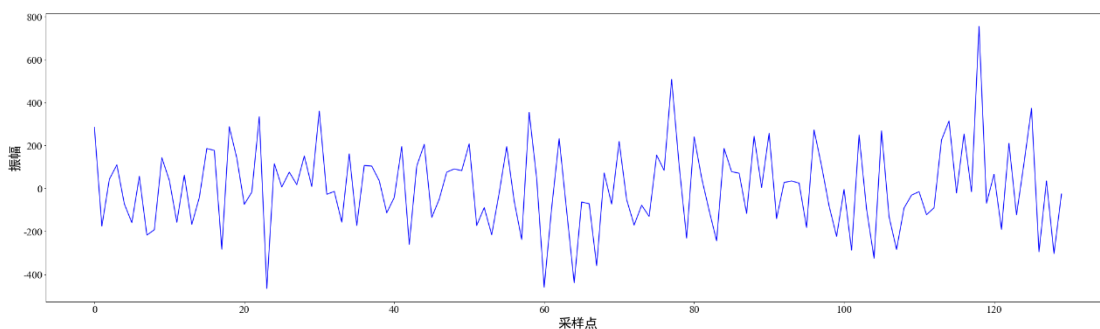


图 3.2 低频信号测试数据

对低频信号的训练数据和测试数据进行平稳性检验，首先绘制自相关系数（ACF）和偏自相关系数（PACF）图，横坐标为延迟阶数。通常 ACF 和 PACF 可以帮助判断时间序列的相关性情况，即确定 ARMA 模型和 ARIMA 模型中的 p 和 q 值大概可以取多少，但可靠性不高。根据平稳时间序列一般具有短期相关性这一特性，观察低频信号的 ACF 和 PACF 图，可以看出不论是训练数据还是测试数据在 1 阶延迟后，数据的相关性并不高，可以推断原始数据更接近于平稳时间序列。

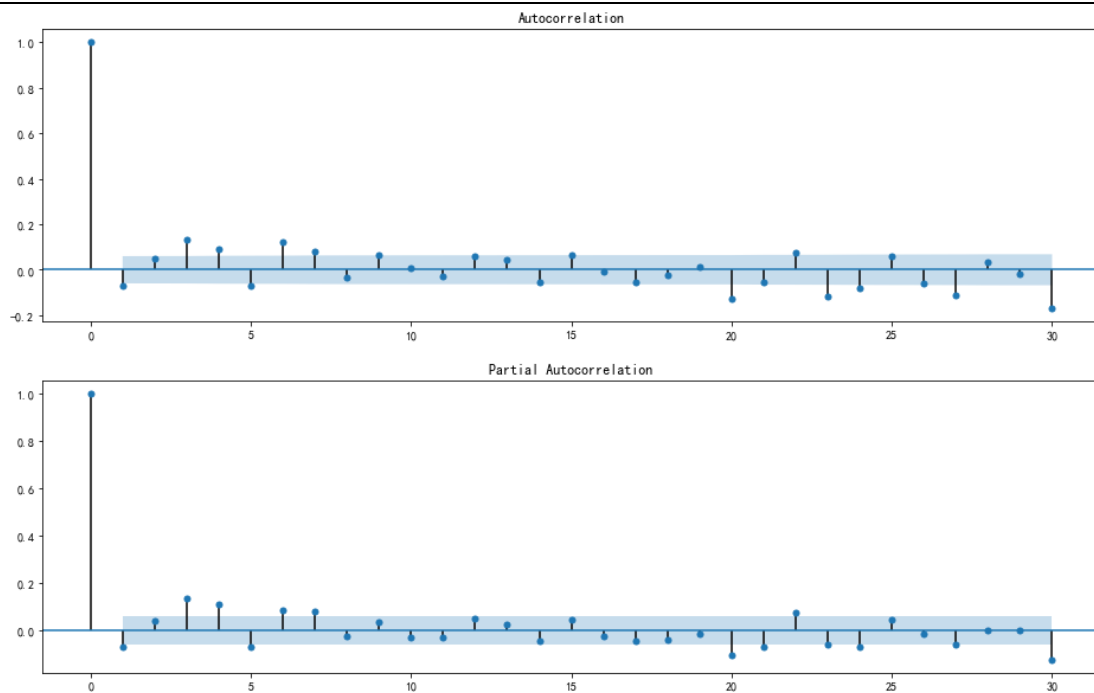


图 3.3 低频信号训练数据的 ACF 和 PACF

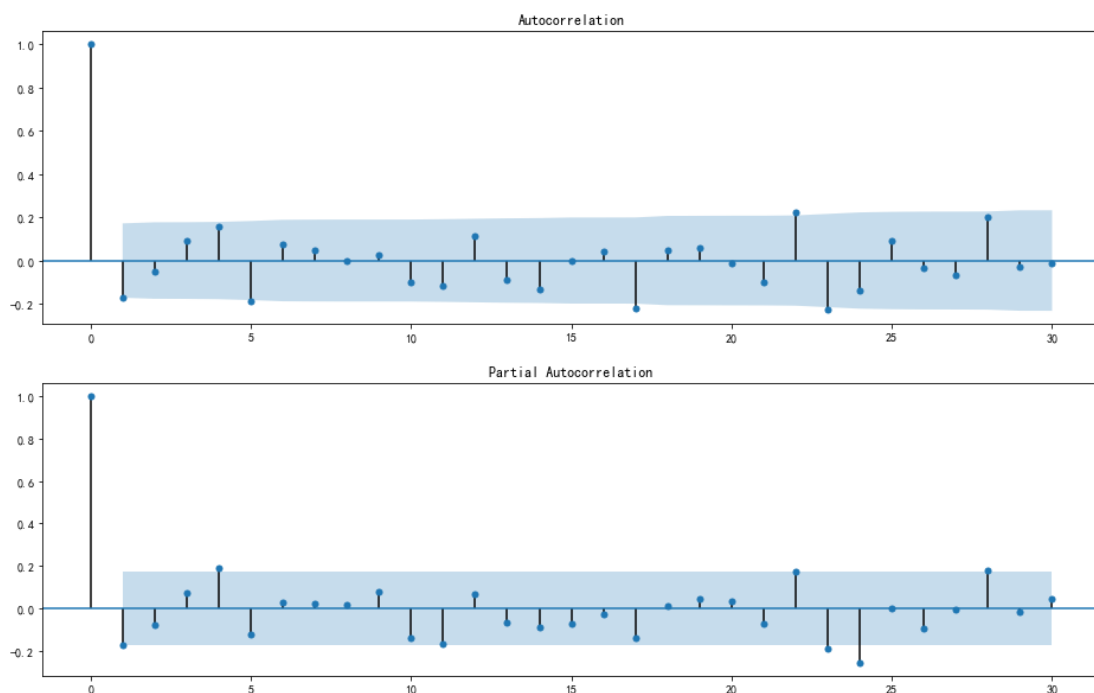


图 3.4 低频信号测试数据的 ACF 和 PACF

低频信号的单位根平稳性检验结果如下，可以看出不论是训练数据还是测试数据的检验统计量的值远小于 1%的关键值，即检验统计量的 p 值远小于 0.01，

可以显著地拒绝原假设 H_0 ,认为训练数据和测试数据是平稳时间序列（是否是纯随机还需要后续检验）:

```
Test Statistic      -7.580559e+00
p-value            2.688974e-11
#Lags Used         2.200000e+01
Number of Observations Used  1.050000e+03
Critical Value (1%)  -3.436593e+00
Critical Value (5%)  -2.864297e+00
Critical Value (10%) -2.568238e+00
dtype: float64
```

图 3.5 低频信号训练数据的单位根平稳性检验法的统计结果

```
Test Statistic      -1.352962e+01
p-value            2.637210e-25
#Lags Used         0.000000e+00
Number of Observations Used  1.290000e+02
Critical Value (1%)  -3.482088e+00
Critical Value (5%)  -2.884219e+00
Critical Value (10%) -2.578864e+00
dtype: float64
```

图 3.6 低频信号测试数据的单位根平稳性检验法的统计结果

中频信号部分如下:

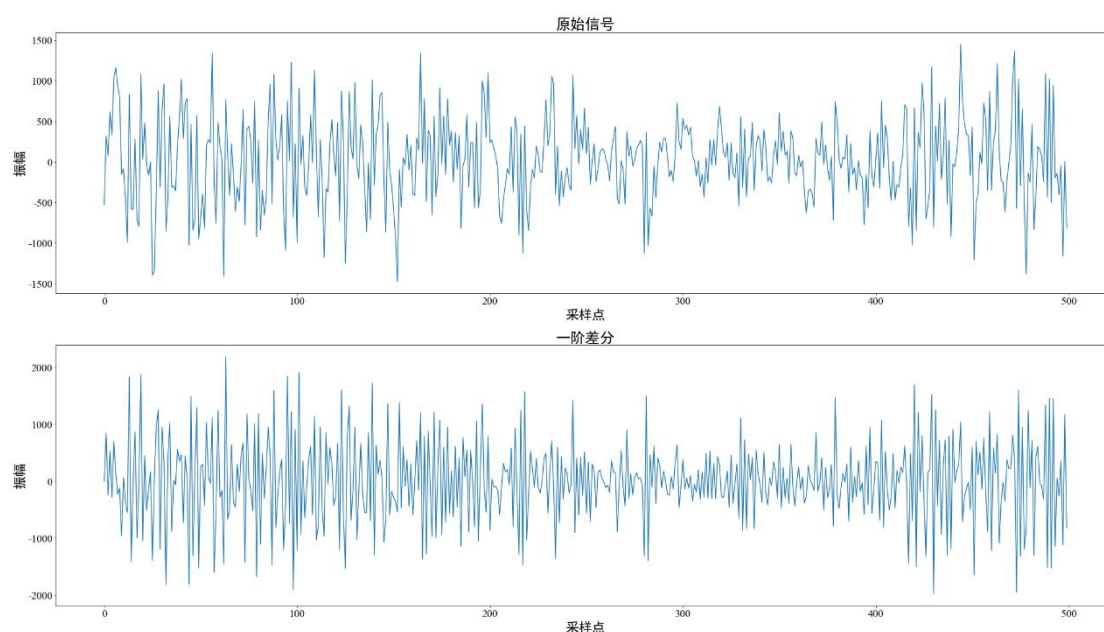


图 3.7 中频信号训练数据和一阶差分

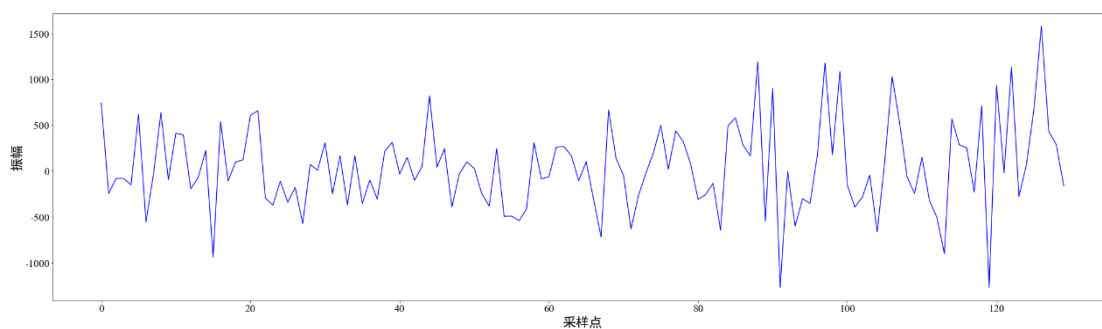


图 3.8 中频信号测试数据

对中频信号的训练数据进行平稳性检验，首先绘制自相关系数（ACF）和偏自相关系数（PACF）图。分析方法同低频信号，可以看出不论是训练数据还是测试数据在 1 阶延迟后，数据的相关性并不高，可以推断原始数据更接近于平稳时间序列。

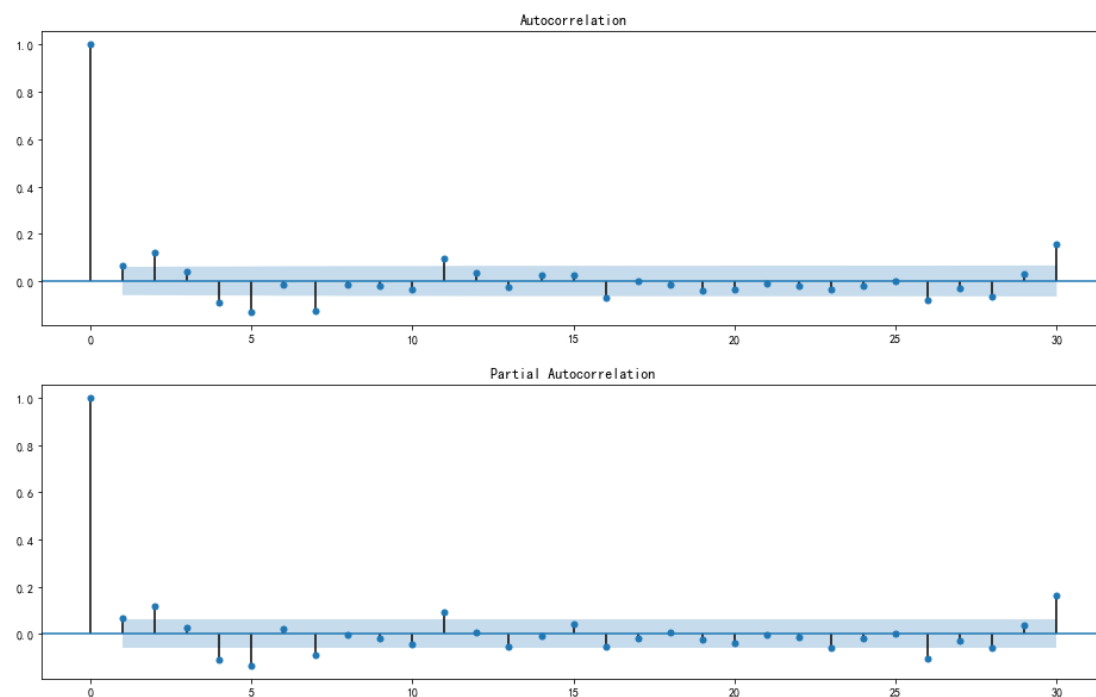


图 3.9 中频信号训练数据的 ACF 和 PACF

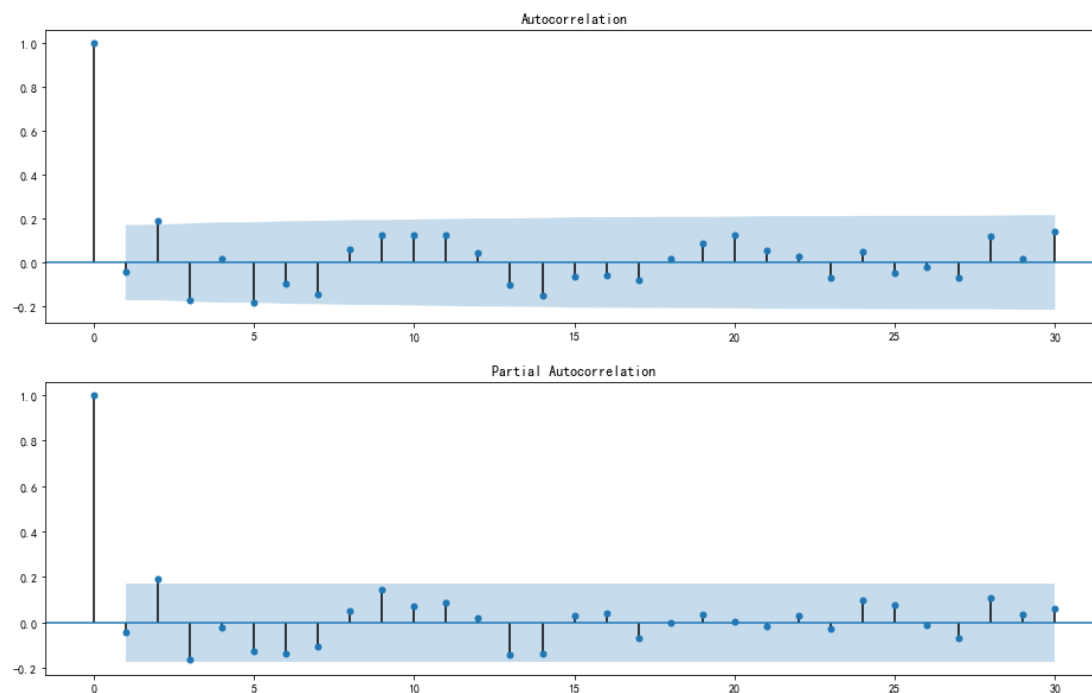


图 3.10 中频信号测试数据的 ACF 和 PACF

中频信号的单位根平稳性检验结果如下，可以看出不论是训练数据还是测试数据的检验统计量的值远小于 1%的关键值，即检验统计量的 p 值远小于 0.01，可以显著地拒绝原假设 H_0 ，认为训练数据和测试数据是平稳时间序列（是否是纯随机还需要后续检验）：

Test Statistic	-1.060868e+01
p-value	5.911066e-19
#Lags Used	1.000000e+01
Number of Observations Used	1.062000e+03
Critical Value (1%)	-3.436522e+00
Critical Value (5%)	-2.864265e+00
Critical Value (10%)	-2.568221e+00
dtype: float64	

图 3.11 中频信号训练数据的单位根平稳性检验法的统计结果

Test Statistic	-5.937094e+00
p-value	2.307159e-07
#Lags Used	5.000000e+00
Number of Observations Used	1.240000e+02
Critical Value (1%)	-3.484220e+00
Critical Value (5%)	-2.885145e+00
Critical Value (10%)	-2.579359e+00
dtype:	float64

图 3.12 中频信号测试数据的单位根平稳性检验法的统计结果

高频信号部分如下：

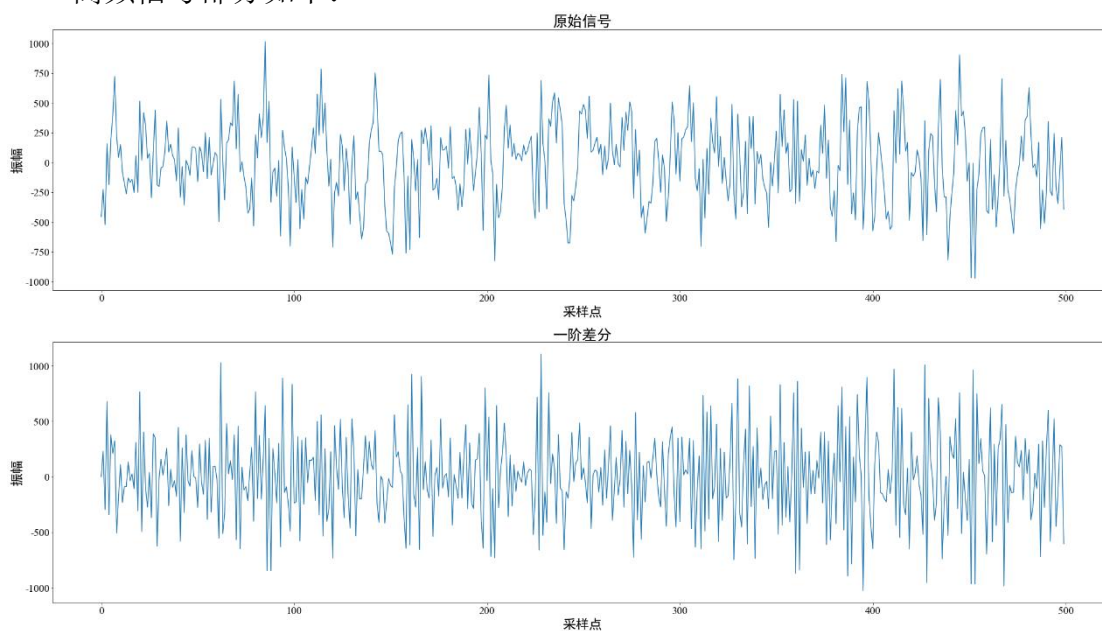


图 3.13 高频信号训练数据和一阶差分

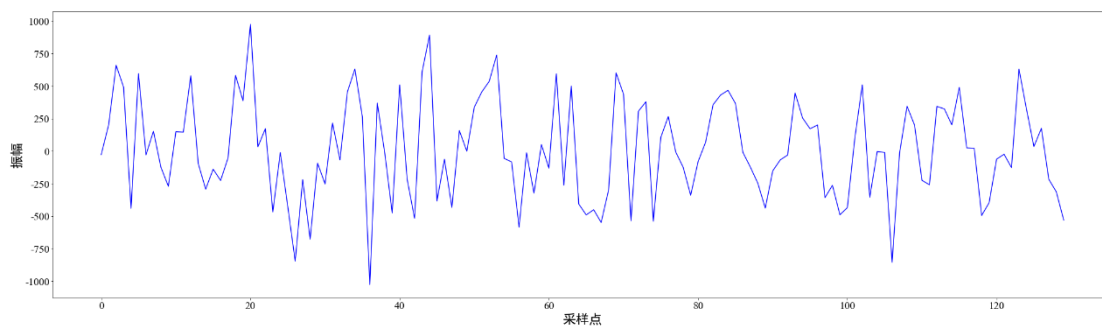


图 3.14 高频信号测试数据

对高频信号的训练数据进行平稳性检验，首先绘制自相关系数（ACF）和偏自相关系数（PACF）图。分析方法同低频和中频信号，可以看出不论是训练数据还是测试数据在 1 阶延迟后，数据的相关性并不高，可以推断原始数据更接近于平稳时间序列。

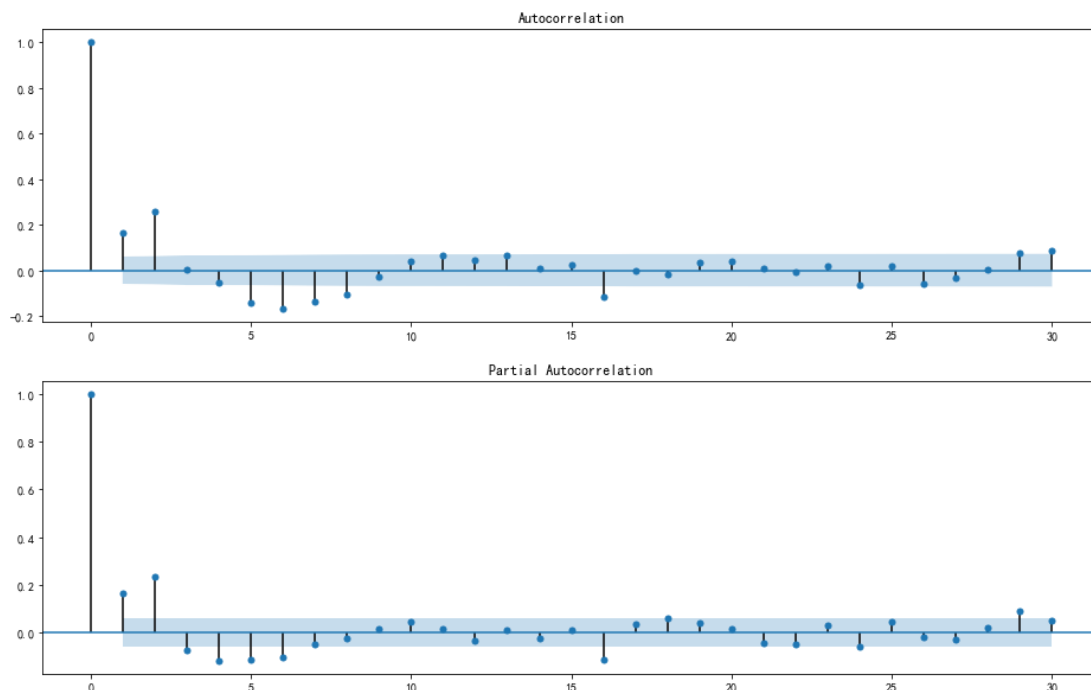


图 3.15 高频信号训练数据的 ACF 和 PACF

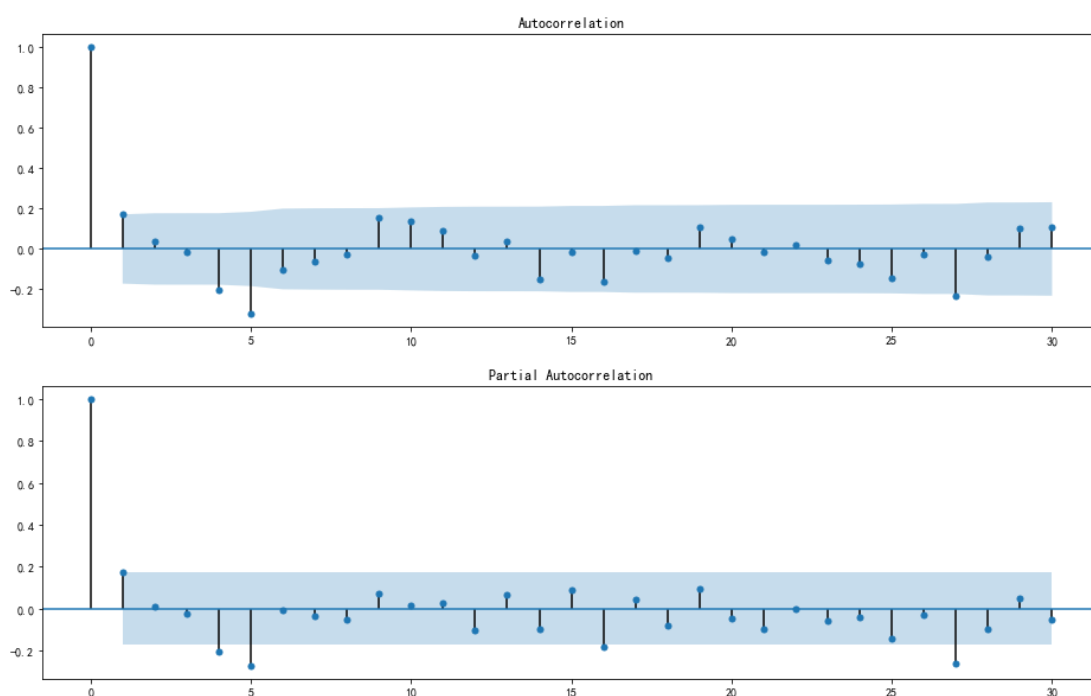


图 3.16 高频信号测试数据的 ACF 和 PACF

高频信号的单位根平稳性检验结果如下，可以看出不论是训练数据还是测试数据的检验统计量的值远小于 1%的关键值，即检验统计量的 p 值远小于 0.01，可以显著地拒绝原假设 H_0 ，认为训练数据和测试数据是平稳时间序列（是否是纯随机还需要后续检验）：

```

Test Statistic      -8.080697e+00
p-value            1.464139e-12
#Lags Used         1.700000e+01
Number of Observations Used  1.055000e+03
Critical Value (1%) -3.436564e+00
Critical Value (5%) -2.864283e+00
Critical Value (10%) -2.568231e+00
dtype: float64

```

图 3.17 高频信号训练数据的单位根平稳性检验法的统计结果

```

Test Statistic      -7.377945e+00
p-value            8.622829e-11
#Lags Used         4.000000e+00
Number of Observations Used  1.250000e+02
Critical Value (1%) -3.483779e+00
Critical Value (5%) -2.884954e+00
Critical Value (10%) -2.579257e+00
dtype: float64

```

图 3.18 高频信号测试数据的单位根平稳性检验法的统计结果

3.3 时间序列分析的纯随机性检验

3.3.1 平稳时间序列的分析价值

如果时间序列经过平稳性检验后为平稳序列，则可以利用对平稳时间序列的常用拟合方法，对平稳时间序列拟合建模，进而利用模型预测未来的变化走势。然而，并不是所有的平稳时间序列都有拟合建模价值。只有序列值之间拥有显著的相关关系，能反映历史数据对未来的变化走势的影响，才值得继续对该平稳序列进行相应的分析

如果序列值之间毫无相关关系，则说明该时间序列中并不蕴含历史信息，历史行为对未来的变化走势没有任何影响。这样的时间序列就是纯随机序列，其中也没有任何值得提取的信息。

为了确定平稳时间序列还有没有继续分析的价值，需要对平稳时间序列进行纯随机性检验，判断其是不是纯随机序列。

3.3.2 平稳时间序列的纯随机性检验

纯随机序列又叫作白噪声序列，其最显著的性质是纯随机性，序列的各项值之间毫无相关性，是随机无序的波动序列。

检验时间序列是否为纯随机序列，同样需要通过构造检验统计量来实现。由于平稳时间序列一般只存在短期相关性，即其中蕴含的显著相关关系，一般只存在于延迟期数相对较短的序列之间。所以我们只需要检验一个平稳时间序列是否存在短期相关关系，就可以判断该序列是否是纯随机序列。

所以序列的纯随机性的检验假设条件可以表示为：

原假设 H_0 ：延迟期数不大于 m 期的序列值之间相互独立；备选假设：延迟期数不大于 m 期的序列值之间存在相关关系。我们用 ρ 代表自相关系数，下标代表延迟的期数，则假设检验条件可以表示如下：

$$H_0: \rho_1 = \rho_2 = \dots = \rho_m = 0, \forall m \geq 1 \quad (3.15)$$

$$H_1: \exists \rho_k \neq 0, \forall m \geq 1, k \leq m \quad (3.16)$$

构造 LB 统计量，对这个联合假设进行检验，其中：

$$LB = n(n+2) \sum_{k=1}^m \left(\frac{\hat{\rho}_k^2}{n-k} \right)^2 \quad (3.17)$$

计算出 LB 统计量的值后，查找它对应的 P 值，若 P 值显著大于所设定的显著性水平 α ，则不能拒绝纯随机的原假设；反之，若 P 值小于显著性水平 α ，则可以从统计学角度显著拒绝原假设，从而认定该序列不是纯随机序列。

3.3.3 对分析信号实例进行纯随机性检验

对 3.3.2 节中进行过平稳性检验的滤波信号的训练数据再进行纯随机检验，从而确定训练数据有没有分析价值，由于平稳时间序列一般具有短期相关性，同时结合信号的 ACF 和 PACF 图得知，信号的相关性延迟期数较短，基本不会超过 10 期，因此在计算 LB 统计量的时候，我们选择延迟期数为 12 期。这里设置检验的显著性水平为 1%。

对低频信号的训练数据进行纯随机性检验，计算 LB 统计量和它对应的 P 值：

	LBvalue	pvalue
1	5.147328	2.328217e-02
2	7.537147	2.308498e-02
3	25.847337	1.026575e-05
4	34.718462	5.306842e-07
5	39.982566	1.505506e-07
6	56.141025	2.725965e-10
7	63.348575	3.226566e-11
8	64.736705	5.440617e-11
9	69.034924	2.353454e-11
10	69.097121	6.621620e-11
11	69.978849	1.233037e-10
12	73.528071	6.966225e-11

图 3.19 低频信号训练数据的 LB 统计量和 P 值

从统计检验结果可以看出，延迟阶数 3 阶以后，低频信号的训练数据 LB 统计量的 P 值显著小于 0.01，且随着延迟期数的增加，P 值有减小的趋势，可以判断低频信号的训练数据不是纯随机序列，有继续分析的意义。

对中频信号的训练数据进行纯随机性检验，计算 LB 统计量和它对应的 P 值：

	LBvalue	pvalue
1	4.700107	3.016073e-02
2	20.875601	2.930359e-05
3	22.581902	4.935311e-05
4	31.515378	2.402905e-06
5	50.250965	1.231226e-09
6	50.503269	3.726206e-09
7	68.236679	3.353653e-12
8	68.438493	1.004680e-11
9	68.961426	2.432737e-11
10	70.490274	3.565112e-11
11	80.266016	1.310865e-12
12	81.427035	2.203744e-12

图 3.20 中频信号训练数据的 LB 统计量和 P 值

从统计检验结果可以看出，延迟阶数 2 阶以后，中频信号的训练数据 LB 统计量 P 值显著小于 0.01，且随着延迟期数的增加，P 值有减小的趋势，同样可以认为中频信号的训练数据不是纯随机序列，有继续分析的意义。

对高频信号的训练数据进行纯随机性检验，计算 LB 统计量和它对应的 P 值：

	LBvalue	pvalue
1	29.856655	4.651972e-08
2	100.000153	1.928602e-22
3	100.022310	1.537087e-21
4	103.381138	1.874148e-21
5	124.729875	3.121975e-25
6	155.772676	4.649825e-31
7	176.775165	9.342856e-35
8	188.760304	1.484526e-36
9	189.552632	5.106666e-36
10	191.218638	1.090305e-35
11	195.603351	6.070311e-36
12	197.541949	1.048566e-35

图 3.21 高频信号训练数据的 LB 统计量和 P 值

从统计检验结果可以看出,延迟阶数 1 阶以后,高频信号的训练数据 LB 统计量 P 值显著小于 0.01,且随着延迟期数的增加,P 值有减小的趋势,同样可以认为中频信号的训练数据不是纯随机序列,有继续分析的意义。

综上,低频、中频和低频信号的训练数据均不是纯随机序列,有进行时间序列分析的价值。

3.4 ARIMA 模型拟合时间序列过程

3.4.1 ARIMA 模型建模的基本步骤

完成训练数据和测试数据的平稳性检验和纯随机性检验后,若确认原始数据是平稳非白噪声时间序列,即平稳非纯随机时间序列后,利用 ARIMA 模型对时间序列进行分析。通过原始数据是平稳时间序列还是差分平稳时间序列来确认 ARIMA 模型的参数 d (若 d 为 0, ARIMA 模型等同于 ARMA 模型);通过 ACF 和 PACF 图表现出来的特性,来大致估计 ARIMA 模型中参数 p 和 q 的值。确定好 p , d 和 q 的值后,便可以估计模型的系数,拟合模型,可以利用矩估计,极大似然估计和最小二乘估计等方法,确定模型中未知系数的最优估计值。

拟合得到模型后,要对模型进行有效性检验,主要包括对模型本身的显著性检验和对模型参数的有效性检验两部分。之后,在模型优化的过程中,建立多个不同参数 p 、 q 的拟合模型,并分别对它们进行模型的有效性检验,检验合格的模型中,选取经过评价准则评价最优的模型作为最终模型进行时间序列未来趋势的预测。

整个建模过程基于前面 2 节经过平稳性检验和纯随机性检验的中区热水二次泵 1 号的低频、中频和低频信号部分的训练数据。

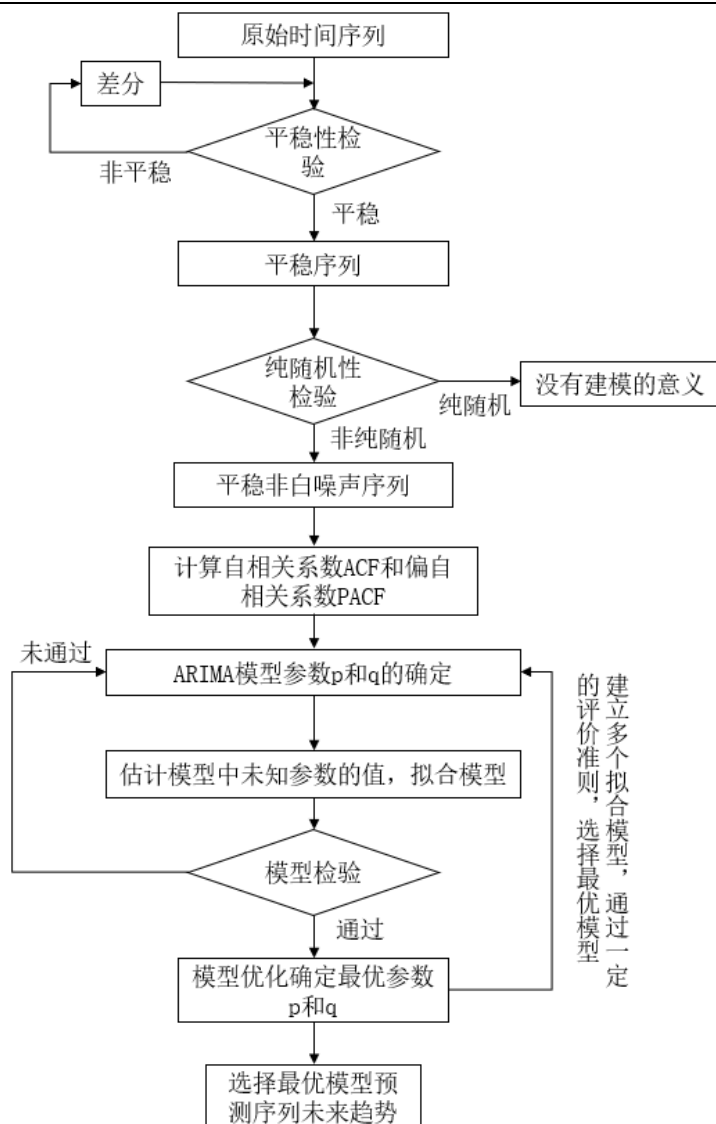


图 3.22 ARIMA 模型建模流程

3.4.2 ARIMA 模型参数的估计

根据原序列几阶差分后为平稳时间序列可以确定 ARIMA 模型的参数 d 。而确定 ARIMA 模型参数 p 和 q 需要通过观察平稳非白噪声序列的 ACF 和 PACF 图来估计。在 3.2.2 节中，曾利用 ACF 和 PACF 图判断时间序列的相关性，下面具体计算自相关系数和偏自相关系数的数学意义，并说明它对参数 p 和 q 确定的指导意义

通常，利用协方差函数和相关系数来判断两个变量之间的相关性，以此寻找其中的统计规律。当单独研究时间序列时，利用自协方差函数和自相关系数来评判同一事件在两个不同时间点下的相关程度。

对于时间序列 $\{X_t, t \in T\}$, 任取 $t, s \in T$, 定义时间序列 $\{X_t\}$ 的自相关系数 ACF 为:

$$\rho(t, s) = \frac{E(X_t - \mu_t)(X_s - \mu_s)}{\sqrt{DX_t \cdot DX_s}} \quad (3.18)$$

式中 E 和 D 分别代表计算期望和方差, μ_t 和 μ_s 是时间序列在 t 和 s 时刻的均值。 $\rho(t, s)$ 亦可以理解为平稳非白噪声序列滞后 (s-t) 阶的自相关系数。

但实际情形下, 对一个平稳的自回归模型来说, 假设求出滞后 k 阶的自相关系数, 得到的并不是单纯的 x_t 与 x_{t-k} 之间的相关系数, 这是因为 x_t 依然受到 $x_{t-1}, x_{t-2}, \dots, x_{t-k+1}$ 这 k-1 个随机变量的影响。同时, 这 k-1 个随机变量又都与 x_{t-k} 之间存在相关关系。所以, 自相关系数 ρ_k 里包含了其他相关变量对 x_t 与 x_{t-k} 的影响。如果想要计算单纯的 x_{t-k} 对 x_t 的影响, 则需要通过计算偏自相关系数来获得。

滞后 k 阶的偏自相关系数就是在给定了 $x_{t-1}, x_{t-2}, \dots, x_{t-k+1}$ 这 k-1 个中间随机变量的条件下, x_{t-k} 对 x_t 的相关影响程度大小的数字表述, 其计算式为:

$$\rho_{x_t, x_{t-k} | x_{t-1}, \dots, x_{t-k+1}} = \frac{E[(x_t - \hat{E}x_t)(x_{t-k} - \hat{E}x_{t-k})]}{E[(x_{t-k} - \hat{E}x_{t-k})^2]} \quad (3.19)$$

式中, $\hat{E}x_t = E[x_t | x_{t-1}, \dots, x_{t-k+1}]$, $\hat{E}x_{t-k} = E[x_{t-k} | x_{t-1}, \dots, x_{t-k+1}]$ 为给定 $x_{t-1}, x_{t-2}, \dots, x_{t-k+1}$ 这 k-1 个中间随机变量的条件下, 变量 x_t 和 x_{t-k} 的期望值。

平稳的 p 阶自回归模型 AR(p)模型的自相关系数有两大显著特性: 拖尾性和呈现指数衰减。其中拖尾性指的是滞后任意 k 阶的自相关系数始终非 0, 不会在 k 大于某个常数之后就恒为 0。同时, AR(p)模型的偏自相关系数呈现 p 阶截尾性, 即当偏自相关系数的滞后阶数 k 大于 p 时, 偏自相关系数值始终为 0。

另一方面, 平稳的 q 阶移动平均模型 MA(q)模型的自相关系数呈现 q 阶截尾性。同时, MA(q)模型的偏自相关系数则呈现拖尾性。

所以 AR(p)模型和 MA(q)模型的自相关系数和偏自相关系数性质不同, 而 ARMA(p, q)模型是 AR(p)模型和 MA(q)模型的有机组合形式, 其自相关系数和偏自相关系数均呈现拖尾性。三者规律如下表所示:

表 3.1 自回归模型和移动平均模型的系数

拟合模型	自相关系数	偏自相关系数
自回归 AR(p)	拖尾	p 阶截尾
移动平均 MA(q)	q 阶截尾	拖尾
自回归移动平均 ARMA(p,q)	拖尾	拖尾

为了确定 ARIMA 模型参数 p 和 q 从而给模型定阶，需要绘制时间序列的 ACF 和 PACF 图，并根据上表的规律，选择正确的 p 和 q 参数来对平稳非白噪声序列进行拟合建模。

在实际中，由于样本的随机性，样本的相关系数并不是完美截尾，而是在小范围内浮动。同时因为平稳非白噪声序列是短期相关的，随着延迟阶数 k 的增大，自相关系数和偏自相关系数都将衰减到零值附近，并在小值范围内浮动。

所以在借助计算机的情况下，最理想的方法是通过 ACF 和 PACF 图判断大致的 p 和 q 范围后，在后续模型优化中通过网格搜索法，按照一定的评价准则确定。观察在 3.2.2 节中低频、中频和低频信号训练数据的 ACF 和 PACF 图，可以看出数据具有较弱的短期相关性，短期相关性基本没有超过 5 阶延迟，因此对于模型合理的 p 和 q 参数的范围不会太大，因此网格搜索选择 p 的范围为 0-5， q 的范围也为 0-5。

3.4.3 ARIMA 模型的评估准则及参数优化

在大致估计模型参数 p 和 q 的范围后，通过网格搜索法寻找最优的参数 p 和 q ，这里就涉及利用一个评估准则，来判断并选出模型最优的参数。一般情况下，可以通过度量拟合程度的似然函数值，以及模型中出现的未知参数个数来评价模型的优劣。

一般情况下，似然函数值大说明模型拟合程度高，模型质量较优。模型中出现的未知参数个数多，说明模型的决定变量多，能适应的变化也多，拟合的准确度因此也更高。但是不断增加模型未知参数，来提高模型拟合的准确度并不是一个可行的办法。随着模型出现的未知参数增多，模型的不稳定性也增大。而且参数增多后，增加了参数估计的难度，从而增大了估计的偏差。因此，配置最优的拟合准确度和未知参数个数后而获得的模型是建模的关键

AIC 准则由日本统计学家提出，常用来评价一个拟合模型的优劣。其数学形式如下：

$$AIC = -2 \ln(f(y|\theta_k)) + 2K \quad (3.20)$$

式中， θ_k 是第 k 个未知参数， y 是因变量，而 $f(y|\theta_k)$ 计算了模型的极大似然函数值。 K 则是模型中未知参数的个数。

一定情形下，能最小化 AIC 值的模型是该情形下的最优模型，所以增大极大似然函数值的同时，减少模型未知参数的个数，有利于获得更小的 AIC 值，即获得更优的模型。

但是如果一个观察对象容量较大,则从中获取有价值的信息难度就较大,就需要较高的拟合精度,从而对决定变量的数量要求也较高。而 AIC 准则中第一项拟合误差在计算时会受到样本容量正比例地影响,AIC 准则中第二项,参数个数前的系数始终是 2,与样本容量毫不相关。因此随着观察对象样本容量增加,AIC 准则或趋向于使得未知参数更多增加,实质上是不利于拟合较优的模型。

BIC 准则可以改善 AIC 准则的这一缺点,它的数学形式如下:

$$BIC = -2 \ln(f(y|\theta_k)) + \ln(n)K \quad (3.21)$$

可见,BIC 准则通过修改参数系数为样本容量的单值函数,从而加入样本容量对未知参数个数的影响,以此来改善 AIC 准则。

同理,一定情形下,能最小化 BIC 值的模型也是该情形下的最优模型。

最终对于低频信号的训练数据,选择 p 为 4, q 为 3, d 为 0 的 ARIMA 模型,简记为 ARMA(4, 3);对于中频信号的训练数据,选择 p 为 5, q 为 0, d 为 0 的 ARIMA 模型,简记为 ARMA(5, 0);对于高频信号的训练数据,选择 p 为 5, q 为 0, d 为 0 的 ARIMA 模型,简记为 ARMA(3, 2)。

3.4.4 ARIMA 模型的有效性校验

模型的有效性检验包括检验模型的显著性和参数的显著性两部分,其主要意义和方法如下:

(一)如果模型拟合后残差项为纯随机序列,则表明模型的残差项不再存在任何有价值的信息,拟合模型充分提取了原数据中的有价值信息,这时的模型是显著有效的。与此相反,如果模型拟合后残差项不是纯随机序列,则需要重新进行模型拟合。因此检验模型拟合后残差序列的纯随机性,就是检验模型的显著性,检验的方法同 3.3 节中叙述的相同。

(二)模型拟合后决定变量所对应的未知参数影响模型的表达,需要对这些参数进行统计学上的显著性检验,从而判定参数对模型影响的显著程度。在设定的显著性水平下,没有通过显著性检验的参数对模型的表达影响小,因此可以删去。不断拟合模型并删除没有通过显著性检验的参数,直到最终获得的拟合模型所有的参数都显著影响模型表达。

低频信号训练数据 ARMA 模型拟合后残差的 ACF 和 PACF 图如下,可以看出模型在 1 阶开始后就截尾,根据平稳时间序列具有短期相关性,因此可以基本判断残差是白噪声序列,不再有拟合价值,模型提取的信息充分。

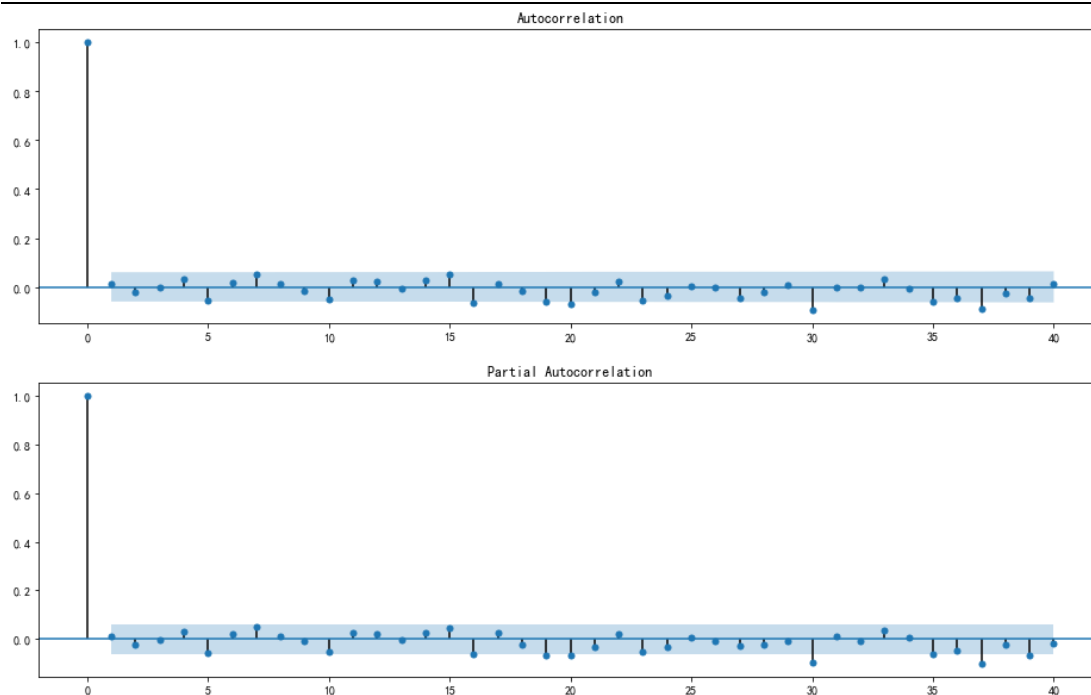


图 3.23 低频信号训练数据的 ACF 和 PACF 图

再对模型的残差进行基于 LB 统计量的纯随机性检验，结果见下图，模型残差的 LB 统计量和其 P 值远大于 0.01（显著性水平），因此不能拒绝原假设，即低频信号拟合模型的残差是纯随机序列：

	LBvalue	pvalue
1	0.132942	0.715401
2	0.700529	0.704502
3	0.707072	0.871539
4	1.790784	0.774168
5	5.230992	0.388345
6	5.618014	0.467309
7	8.397072	0.298885
8	8.624851	0.374924
9	8.880356	0.448391
10	11.453877	0.323266
11	12.264336	0.344106
12	12.732115	0.388809

图 3.24 低频信号拟合模型残差的 LB 统计量和 P 值

低频信号训练数据的 ARMA 模型拟合统计结果如下，在 $P>|t|$ 一列中可以看出 ARMA(4,3)模型除了 q 的第一个参数差一点满足 5%置信度的显著性检验，所有的参数都通过显著性检验，可以被认为是有价值的参数，整个参数组的选取基本合理。

Results: ARMA						
=====						
Model:	ARMA	BIC:	14739.8579			
Dependent Variable:	y	Log-Likelihood:	-7342.0			
Date:	2019-12-21 16:33	Scale:	1.0000			
No. Observations:	1073	Method:	css			
Df Model:	7	Sample:	4			
Df Residuals:	1062		3			
Converged:	0.0000	S.D. of innovations:	232.570			
AIC:	14700.0620	HQIC:	14715.138			

	Coef.	Std.Err.	t	P> t	[0.025	0.975]

ar.L1.y	-0.2277	0.0845	-2.6948	0.0072	-0.3933	-0.0621
ar.L2.y	-0.2884	0.0617	-4.6721	0.0000	-0.4094	-0.1674
ar.L3.y	0.7678	0.0729	10.5292	0.0000	0.6249	0.9107
ar.L4.y	0.1641	0.0323	5.0752	0.0000	0.1007	0.2275
ma.L1.y	0.1506	0.0822	1.8327	0.0671	-0.0105	0.3116
ma.L2.y	0.3663	0.0697	5.2546	0.0000	0.2297	0.5030
ma.L3.y	-0.6814	0.0789	-8.6313	0.0000	-0.8361	-0.5267

	Real	Imaginary	Modulus	Frequency		

AR.1	-0.4315	-0.9055	1.0031	-0.3208		
AR.2	-0.4315	0.9055	1.0031	0.3208		
AR.3	1.2061	-0.0000	1.2061	-0.0000		
AR.4	-5.0214	-0.0000	5.0214	-0.5000		
MA.1	-0.4414	-0.9156	1.0164	-0.3215		
MA.2	-0.4414	0.9156	1.0164	0.3215		
MA.3	1.4205	-0.0000	1.4205	-0.0000		
=====						

图 3.25 低频信号拟合 ARMA 模型的统计结果

综上，可以认为低频信号训练数据通过模型的有效性检验，是有效的 ARMA 模型。

中频信号训练数据 ARMA 模型拟合后残差的 ACF 和 PACF 图如下，可以看出模型在 1 阶开始后就截尾，同样地可以判断残差是白噪声序列，不再有拟合价值，模型提取的信息充分。

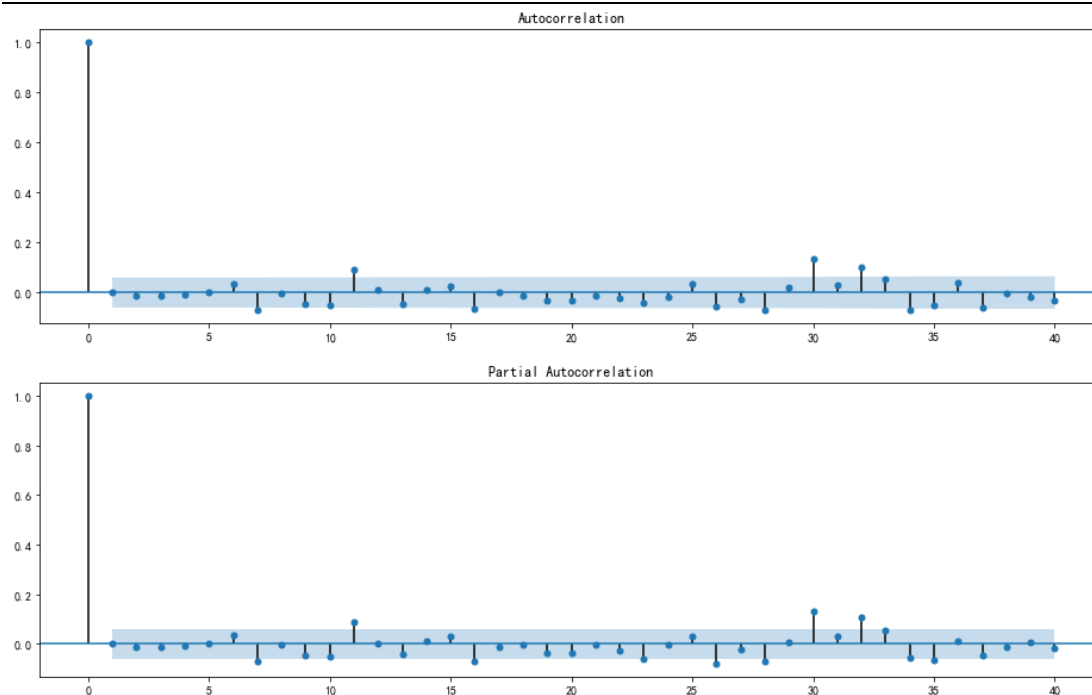


图 3.26 中频信号训练数据的 ACF 和 PACF 图

再对模型的残差进行基于 LB 统计量的纯随机性检验，结果见下图，模型残差的 LB 统计量和其 P 值远大于 0.01（显著性水平），因此不能拒绝原假设，即中频信号拟合模型的残差是纯随机序列：

	LBvalue	pvalue
1	0.005421	0.941308
2	0.161175	0.922574
3	0.373811	0.945598
4	0.423076	0.980544
5	0.423080	0.994670
6	1.890997	0.929445
7	7.203506	0.408003
8	7.208528	0.514311
9	9.331220	0.407278
10	12.000524	0.285021
11	20.951760	0.033874
12	21.040400	0.049792

图 3.27 中频信号拟合模型残差的 LB 统计量和 P 值

中频信号训练数据的拟合统计结果如下，在 $P > |t|$ 一列中可以看出 ARMA(5, 0)模型除了 q 的第 1 个和第 3 个参数离满足 5%置信度的显著性检验有一些差距外，其他所有的参数都通过显著性检验，可以被认为是有价值的参数，整个参数组的选取基本合理。

Results: ARMA						
Model:	ARMA	BIC:	16507.2410			
Dependent Variable:	y	Log-Likelihood:	-8232.7			
Date:	2019-12-21 15:43	Scale:	1.0000			
No. Observations:	1073	Method:	css			
Df Model:	5	Sample:	5			
Df Residuals:	1063		3			
Converged:	1.0000	S.D. of innovations:	538.928			
AIC:	16477.3997	HQIC:	16488.705			
	Coef.	Std.Err.	t	P> t	[0.025	0.975]
ar.L1.y	0.0433	0.0303	1.4283	0.1535	-0.0161	0.1028
ar.L2.y	0.1341	0.0302	4.4453	0.0000	0.0750	0.1933
ar.L3.y	0.0496	0.0304	1.6308	0.1032	-0.0100	0.1092
ar.L4.y	-0.1031	0.0302	-3.4176	0.0007	-0.1623	-0.0440
ar.L5.y	-0.1322	0.0303	-4.3630	0.0000	-0.1915	-0.0728
	Real	Imaginary	Modulus	Frequency		
AR.1	1.1404	-0.7583	1.3695	-0.0934		
AR.2	1.1404	0.7583	1.3695	0.0934		
AR.3	-0.6933	-1.3887	1.5521	-0.3237		
AR.4	-0.6933	1.3887	1.5521	0.3237		
AR.5	-1.6745	-0.0000	1.6745	-0.5000		

图 3.28 中频信号拟合 ARMA 模型的统计结果

综上，可以认为低频信号训练数据通过模型的有效性检验，是有效的 ARMA 模型。

高频信号训练数据 ARMA 模型拟合后残差的 ACF 和 PACF 图如下，可以看出模型在 1 阶开始后就截尾，同样地可以判断残差是白噪声序列，不再有拟合价值，模型提取的信息充分。

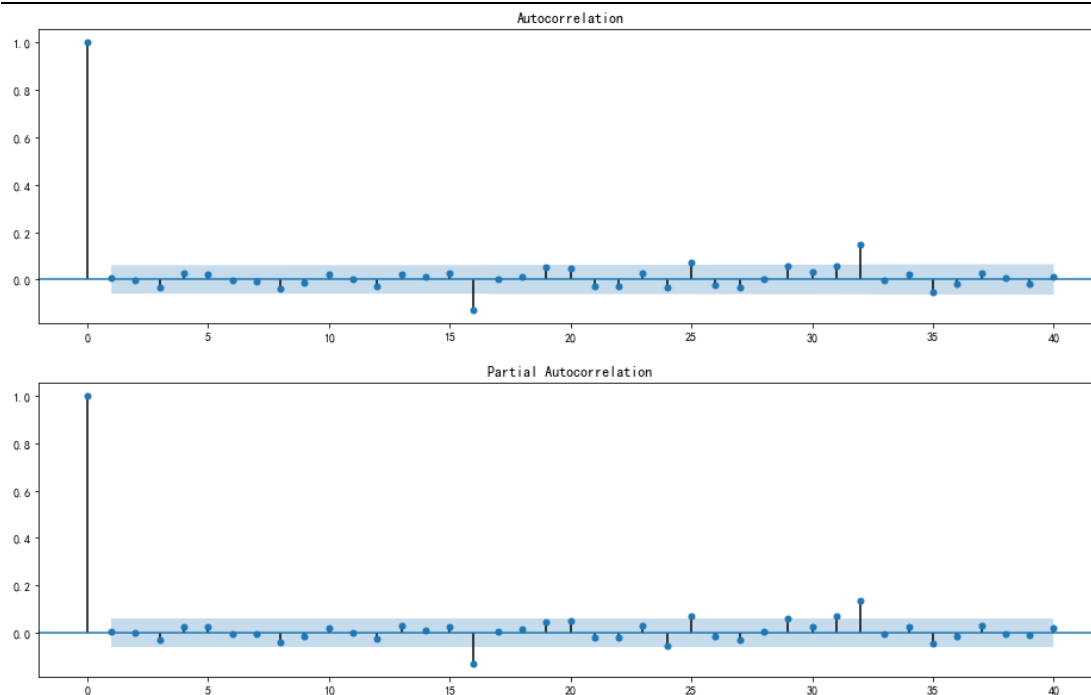


图 3.29 高频信号训练数据的 ACF 和 PACF 图

再对模型的残差进行基于 LB 统计量的纯随机性检验，结果见下图，模型残差的 LB 统计量和其 P 值远大于 0.01（显著性水平），因此不能拒绝原假设，即高频信号拟合模型的残差是纯随机序列：

	LBvalue	pvalue
1	0.005421	0.941308
2	0.161175	0.922574
3	0.373811	0.945598
4	0.423076	0.980544
5	0.423080	0.994670
6	1.890997	0.929445
7	7.203506	0.408003
8	7.208528	0.514311
9	9.331220	0.407278
10	12.000524	0.285021
11	20.951760	0.033874
12	21.040400	0.049792

图 3.30 高频信号拟合模型残差的 LB 统计量和 P 值

高频信号训练数据的拟合统计结果如下，在 $P > |t|$ 一列中可以看出 ARMA(3, 2)模型的 5 个参数都通过显著性检验，可以被认为是具有价值的参数，整个参数组的选取合理。

Results: ARMA

```

=====
Model:                ARMA                BIC:                15433.5743
Dependent Variable:  y                Log-Likelihood:    -7695.9
Date:                2019-12-21 16:22  Scale:              1.0000
No. Observations:   1073            Method:            css
Df Model:            5                Sample:            3
Df Residuals:       1065            S.D. of innovations: 321.647
Converged:           1.0000            HQIC:              15415.030
AIC:                15403.7218
=====

```

	Coef.	Std.Err.	t	P> t	[0.025	0.975]
ar.L1.y	1.1721	0.0970	12.0799	0.0000	0.9819	1.3623
ar.L2.y	-0.2722	0.1353	-2.0121	0.0445	-0.5374	-0.0070
ar.L3.y	-0.2217	0.0434	-5.1057	0.0000	-0.3068	-0.1366
ma.L1.y	-1.0730	0.1006	-10.6616	0.0000	-1.2702	-0.8757
ma.L2.y	0.4158	0.1110	3.7474	0.0002	0.1983	0.6333

	Real	Imaginary	Modulus	Frequency
AR.1	1.0201	-0.5829	1.1749	-0.0826
AR.2	1.0201	0.5829	1.1749	0.0826
AR.3	-3.2683	-0.0000	3.2683	-0.5000
MA.1	1.2902	-0.8604	1.5508	-0.0936
MA.2	1.2902	0.8604	1.5508	0.0936

=====

图 3.31 高频信号拟合 ARMA 模型的统计结果

综上，可以认为低频信号训练数据通过模型的有效性检验，是有效的 ARMA 模型。

3.4.5 ARIMA 模型的测试和故障监测指导

ARIMA 模型在训练集上拟合的结果可以分成拟合趋势部分和残差部分，分析结果可以看出，拟合的趋势相比残差弱一些，结合模型的有效性检验中已经证实残差是白噪声序列，说明低频、中频和低频信号部分的不确定性较强，确定性较弱。

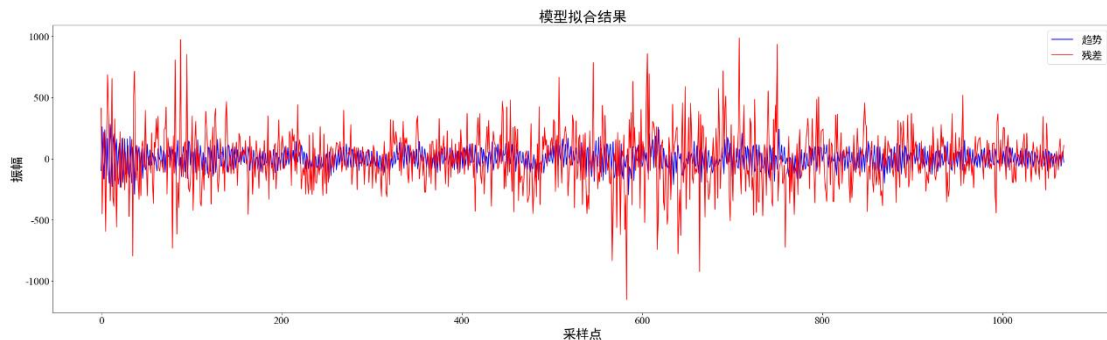


图 3.32 低频信号训练数据的拟合结果

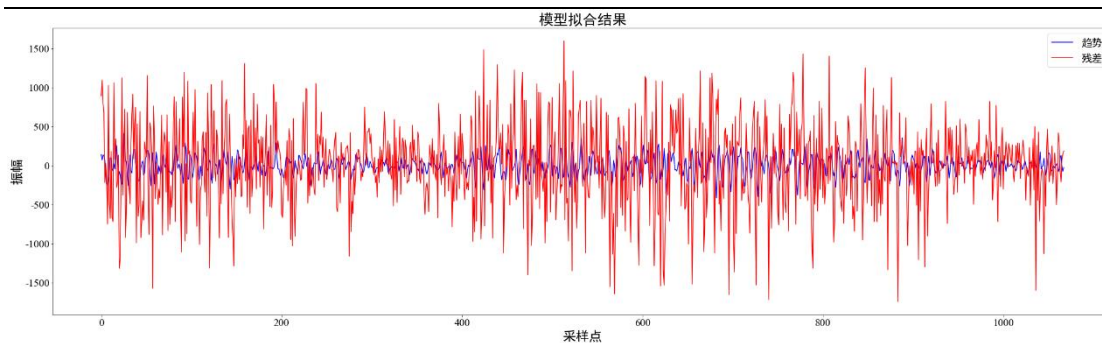


图 3.33 中频信号训练数据的拟合结果

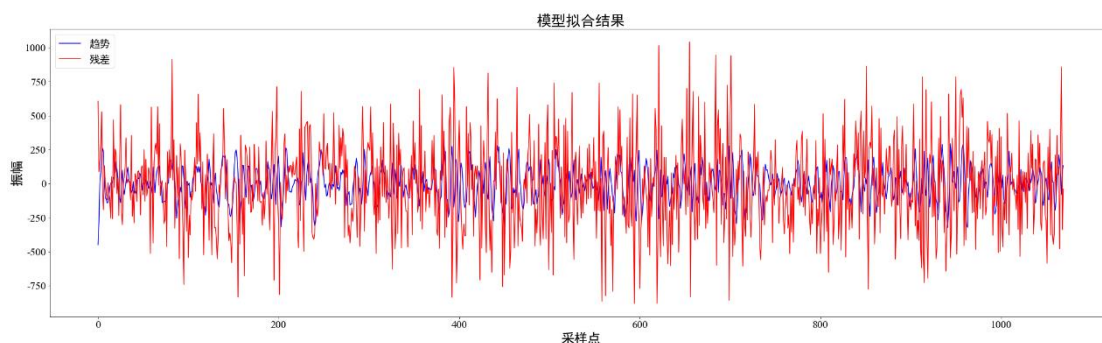


图 3.34 高频信号训练数据的拟合结果

对模型进行测试时，用拟合得到的 ARMA 模型向后预测等同于测试集长度的数据，并利用拟合数据的 95%置信区间上限和下限作为测试数据的阈值。可以看出向后预测时，低频信号的阈值有一定的波动，到了中频信号和高频信号时，阈值几乎趋于定值。总体来看，阈值的定值限值决定了音频信号波动的范围限值，可以反应出残差数据值更大的音频信号的浮动范围规律。因此，可以利用 95%置信区间上限和下限作为检验音频信号是否有异常的阈值。

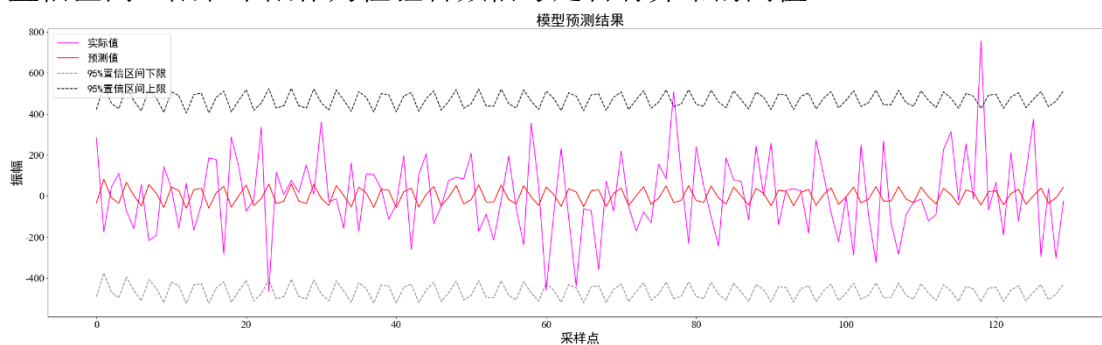


图 3.35 低频信号测试数据的预测结果

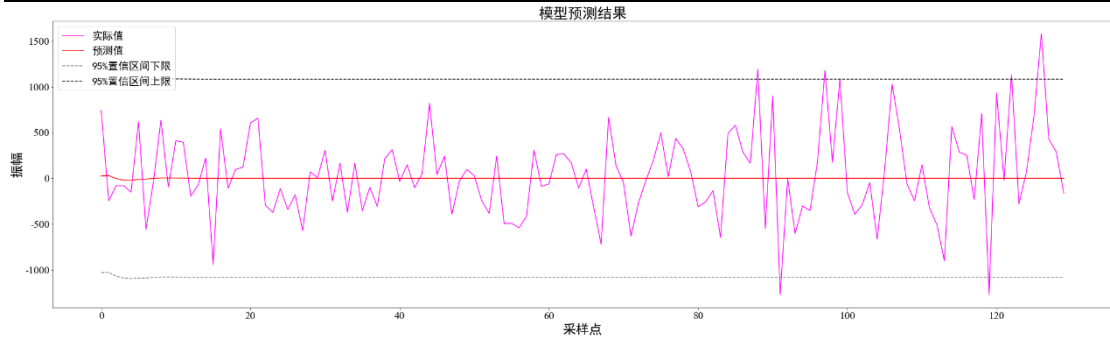


图 3.36 中频信号测试数据的预测结果

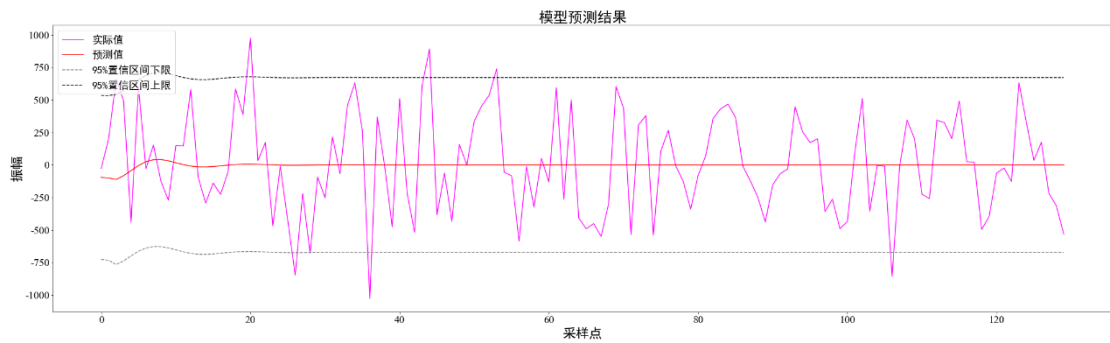


图 3.37 高频信号测试数据的预测结果

具体的阈值诊断方法通过对不同水泵的音频信号的低频、中频和高频部分信号分别建立 ARIMA 时间序列模型，并通过模型的预测数据的 95%置信区间的上限和下限作为阈值。获得新的音频信号数据如果在预测数据的 95%置信区间的上限和下限之间时，则音频信号正常。同时考虑一些不确定因素，观察测试数据中实际值在预测结果 95%置信区间上下限之外的数据点相对较少，同时考虑到置信区间亦不是 100%的置信区间。因此在进行实际故障巡检时，当获得新的音频信号数据的多数信号值（监测阈值：设定为全部信号的一定百分比）都在预测结果 95%置信区间的上限和下限之内时，认为音频信号无异样，此时水泵正常运行；反之，则认为音频信号有异样，此时水泵非正常运行，存在故障。

3.5 本章小结

本章详细介绍了如何对滤波获得的低频、中频和高频三个不同成分的信号利用 ARMA 或 ARIMA 模型进行时间序列拟合。并提出建立时间序列模型应该遵循的步骤，主要包括时间序列的平稳性检验和纯随机性检验以及时间序列的 ARIMA 模型拟合。其中 ARIMA 模型拟合的过程是算法建立的核心部分，包括了模型参数的估计、模型的有效性检验和模型结果测试等内容。整个算法分别在

低频、中频和低频三个成分的信号上进行了验证，并通过最终结果提出指导故障监测的合理方法。

第 4 章 基于深度学习图像处理的冷冻机房关键设备分类

本章开始研究基于图像故障监测和诊断算法开发，本章着重利用卷积神经网络训练一个针对冷冻机房内主要设备和装置进行分类的深度神经网络，根据此网络可以自动识别拍照得到的图片类型，并确定具体使用哪一个算法模块。

4.1 深度学习和张量库 Pytorch 介绍

4.1.1 深度学习及其发展

深度学习是机器学习的一个分支，它以人工神经网络为架构，是一种对数据进行表征学习的算法^[56]。它可以将原始数据转换为能够被机器学习来有效开发的一种形式，避免了人为手动提取特征的麻烦，允许计算机学习使用特征的同时，也学习了如何提取特征，即学习如何学习。

深度学习相比于传统的机器学习，拥有非常强的特征提取能力，因此在处理大体量的训练数据时，能获得比传统机器学习更高的准确性。特别是在计算机视觉和语音识别等领域，比起传统机器学习，深度学习显著地展现出自己的锋芒。例如图像处理，彩色图像可以用一个三维矩阵表示，分别代表图像的高度，图像的宽度和图像的色彩三通道，矩阵的每个点代表图像像素的值，这就使得一张彩色图像所包含的信息量很大，人为地去规定它的特征变得十分困难，也不够准确。而深度学习则可以更好地处理这样的拥有很大信息量的数据。

至今，已经出现过数种深度学习架构，较为经典且应用较多的 2 类架构是卷积神经网络（CNN）和递归神经网络（RNN）。并且，随着硬件技术的进步，高性能的图形处理器（GPU）不断更新，使得数值和矩阵运算的速度大幅度提高，也助力了深度学习神经网络结构的复杂化和处理信息能力的增强。随着基本框架和理论地不断完善，深度学习在人工智能领域的地位也越来越重要。

4.1.2 深度学习框架 Pytorch

Pytorch 是 Torch 的 Python 版本，是由 Facebook 开源的神经网络框架，是一个的对多维矩阵数据进行操作的经典张量库，在机器学习和其他数学密集型领域有广泛应用。但由于 Torch 使用了一种不是很大众的语言 Lua 作为接口，使得其一直没有被推广^[57]。Torch 的开发团队考虑到 Python 在计算科学领域的优势地位

和完整的生态环境以及便捷易学的接口,在对其中的张量进行重构并增加许多新功能的基础上,封装 Torch 后提供 Python 接口,推出了现在的 Pytorch。至今,Pytorch 推出只有 2 年多,其 Windows 平台版本推出仅仅 1 年多,但是却引起了广泛的关注,成为与 Tensorflow 齐名的最受欢迎的深度学习框架之一。

在 Pytorch 推出之前,几乎霸占深度学习框架受欢迎程度榜首的是 Google 公司开发的 Tensorflow。Tensorflow 基于静态计算图,先定义再运行,一次定义后可以多次运行,即网络结构流程的搭建和网络内容的修正是分离的。而 Pytorch 是基于动态计算图的,在运行过程中被定义,可以多次运行并多次构建,即网络结构流程的搭建是在网络内容不断被更新的时候同时生成的。

这样的动态图的设计使得 Pytorch 比 Tensorflow 更加简洁,其设计按照张量、封装变量、层和模块三个从低到高的抽象层次,数学上代表高维数组、自动求梯度、神经网络架构三个具象实意。同等复杂度的问题下,用 Pytorch 搭建的神经网络训练时间代价远低于 Tensorflow,并且运算过程更容易实现 GPU 加速,不需要复杂的环境配置,只要安装过由 NVIDIA 推出的通用并行计算架构 CUDA 便可以实现在 GPU 上封装变量,进行神经网络训练。

因此,综合考虑环境配置的便捷和设计思路的精简两方面的优势,本论文的深度学习图像分类算法选择利用 Pytorch 框架来实现。

4.2 卷积神经网络

4.2.1 定义和基本结构

卷积神经网络(CNN)是一种前馈神经网络,基本结构一般包括多个卷积层和末端的多个全连接层组成,其中全连接层的最后一层也叫作 softmax 层,全连接层即对应了传统的人工神经网络输入层与输出层之间的隐藏层。通常可以认为卷积层的主要目的是提取特征,而全连接层的主要目的是分类或回归。

通常卷积层包括 3 个主要功能和要素,分别是卷积部分、激活函数部分、池化部分。

卷积部分是一组平行的特征图,通过在输入图像上按照一定的滑动步长滑动多个相同尺寸不同参数的卷积核(卷积核通常为尺寸远小于原图像的矩阵),同时执行一定的运算而得到。每个卷积核计算产生一张特征图,多个卷积核计算得到的多张特征图叠在一起获得最后输出的结果。其中,一张特征图中的所有元素都是通过一个卷积核运算得到的,因此一张特征图共享相同的权重和偏置项,这样也显著降低了卷积神经网络的复杂程度。

激活函数部分的目的是增强整个神经网络的非线性特性，从而使得模型拟合能力增强，适应性强。常见的激活函数主要有 Sigmoid 函数、双曲正切函数还有线性整流函数 ReLU。传统的人工神经网络多利用 Sigmoid 函数、双曲正切函数作为激活函数，对于深度较小的传统人工神经网络效果较好。但是随着网络深度的增加，在训练过程中反向传播以更新网络权重和偏置参数的时候，这两大函数的梯度的自身结构会造成网络的“梯度弥散”，致使接近输出层的隐藏层内梯度值大，参数更新速度快，而接近输入层的隐藏层内梯度值小，参数更新速度慢，甚至停滞。而数 ReLU 函数的梯度非 0 即 1，可以避免“梯度弥散”的出现。对于层数较多的卷积神经网络，选择 ReLU 函数作为激活函数解决了梯度弥散问题，同时计算时只需要判断输入是否大于 0，从而加快了计算速度，且由于梯度很稳定，收敛速度也优于 Sigmoid 函数和 Tanh 函数。总体来说，ReLU 函数更有利于网络结构的稳定性。不过，ReLU 函数的梯度有可能为 0，这可能会导致某些神经元的梯度为 0，导致其参数永远不能被更新，特别是在学习率较大时这一情况较容易出现。为了改善这一问题，提出了 Leaky ReLU 函数，它与 ReLU 函数的不同是其横坐标小于 0 的部分值为 α ，通常取 $\alpha = 0.01$ 。虽然从理论上讲，Leaky ReLU 函数继承了 ReLU 函数的全部优势，且不会造成“神经元死亡”的不稳定现象。但是至今为止，还没有在实践操作中证明 Leaky ReLU 函数总是好于 ReLU 函数。

四种激活函数的表达式分别为：

Sigmoid 函数：

$$f(x) = \frac{1}{1 + e^{-x}}$$

Tanh 函数：

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

ReLU 函数：

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases}$$

Leaky ReLU 函数：

$$f(x) = \begin{cases} 0.01x, & x < 0 \\ x, & x \geq 0 \end{cases}$$

将 4 种函数及让它们的导数绘制成下面的图片所示，可以看出 Sigmoid 函数和 Tanh 函数的导数的值基本上都小于 1，在深度神经网络更新权重和偏置项的反向传播过程中，由于链式法则效应，连续多层小于 1 的梯度相乘会使得梯度越来越小，渐渐趋于 0 为消失。同时由于深度神经网络初始化权值和偏置项的绝对值多数小于 1，因此随着网络层数的加深，梯度弥散现象不可避免。而 ReLU 函

数和 Leaky ReLU 函数的导数值在横坐标大于 0 的时候都等于 1，在深度神经网络反向传播更新权重和偏置项的时候不会出现梯度弥散现象。

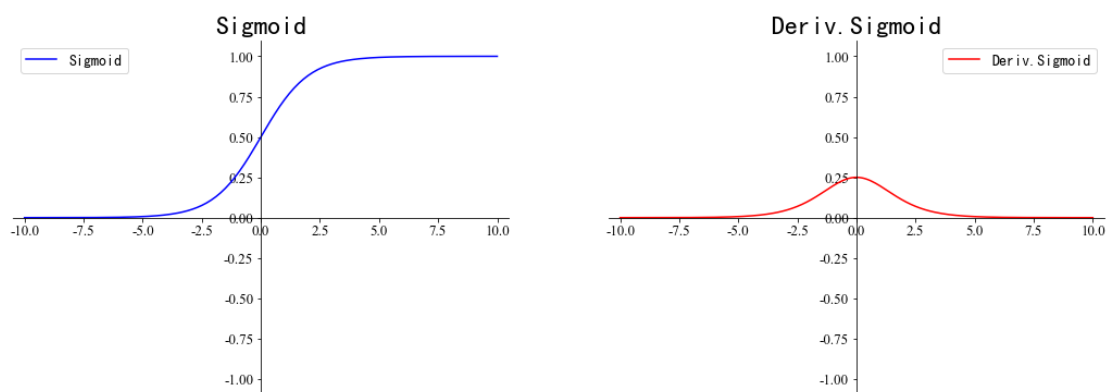


图 4.1 Sigmoid 函数和它的导函数图

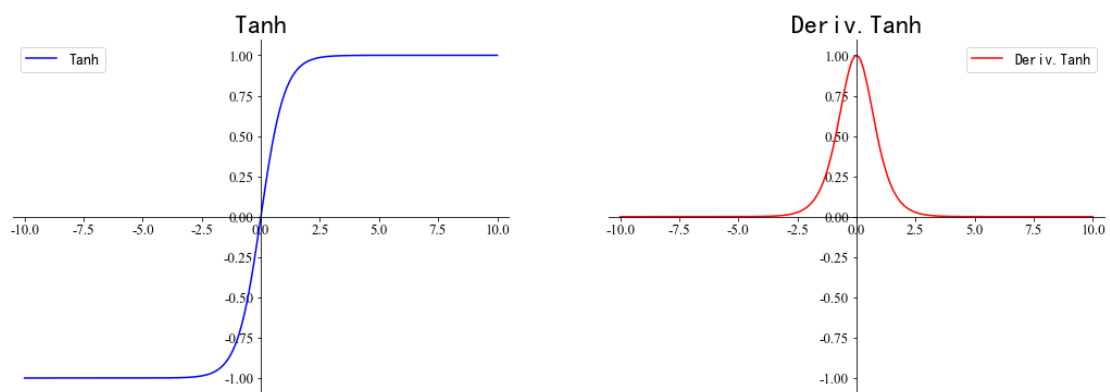


图 4.2 Tanh 函数和它的导函数图

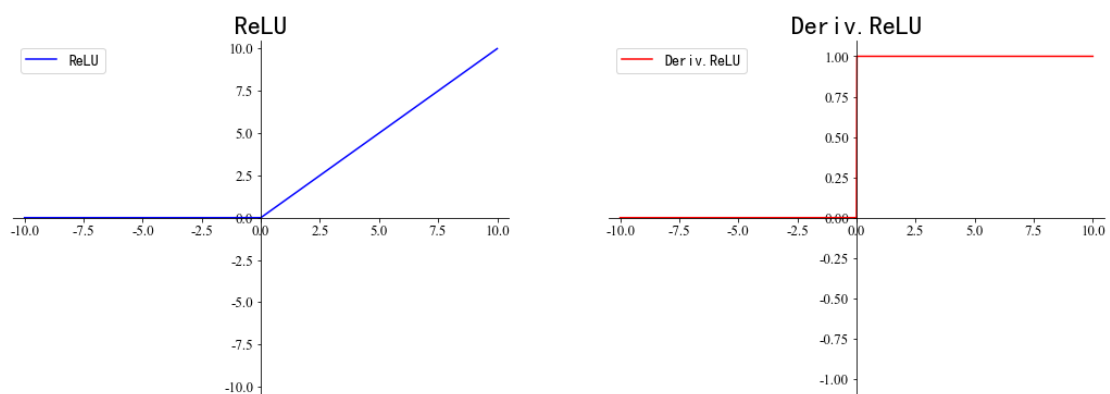


图 4.3 ReLU 函数和它的导函数图

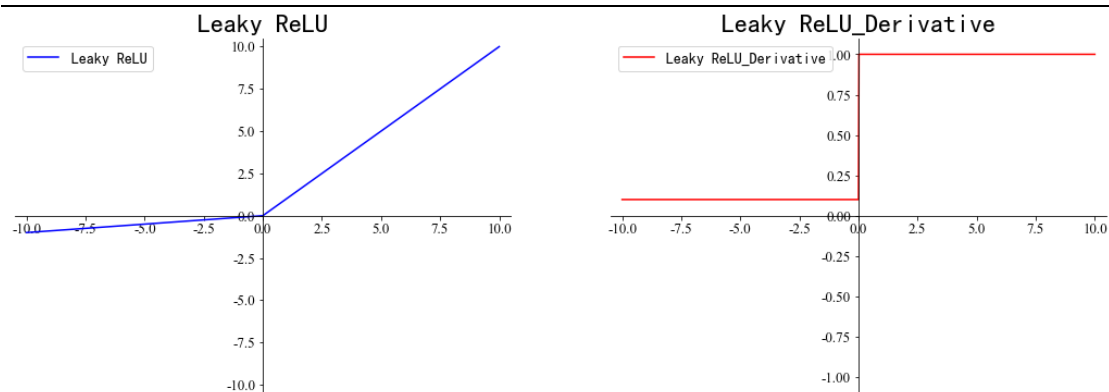


图 4.4 Leaky ReLU 函数和它的导函数图（其中 $\alpha = 0.1$ ，为了显示清楚）

池化部分可以看作一种形式的降采样，其中最常见 2 种池化函数是最大化池化和平均池化。以图像数据为例，它可以将输入的图像数据（可能经过卷积和激活函数处理）划分为多个矩形区域，对每个矩形区域求最大值或者求平均值作为输出结果。一般情况下，一个特征在原始图像内的精确位置相比于它和其他特征的相对位置来说，并不是很关键，因此池化可以在卷积的基础上进一步有效地减小数据的空间大小，保留数据的一些重要特征。其中，由于在实践中表现的更好，最大化池化是应用最多的池化操作，它可以保证数据的重要特征几乎不被损失的条件，减小数据的体量，从而达到提取关键特征的目的。

全连接层通常包括线性运算部分，激活函数部分，有时还会有 dropout 部分。深度神经网络中卷积层提取的是局部特征，全连接层的目的是把之前提取的所有局部特征重新通过权值矩阵组装成完整的特征矩阵。一般情况下，使用 2 层及以上全连接层可以很好地解决非线性拟合问题。

全连接层的线性运算部分与传统人工神经网络的输入层到隐藏层再到输出层的线性运算部分完全一样，通过权值和偏置项将输入结果线性的组合成输出结果。激活函数的作用与卷积层内的相同，也可以选择不同的形式，来增加网络的非线性逼近能力。比较特殊的是 dropout 部分，它是按照一定的概率（一般是 50%，因为这种情况下随机生成的网络结构最多）将神经元单元暂时从网络中丢弃，使其不再对前向和反向传输的数据响应，即将神经元单元的值设置为 0，但权值依然保留。Dropout 丢弃神经元只是暂时的，对于设置多个 mini-batch 的深度神经网络，相当于每一个 mini-batch 都在训练不同的网络，每次训练 dropout 都会按照指定概率随机选择来丢弃神经元。Dropout 部分可以有效防止模型过拟合，让神经网络的泛化能力更强，同时也减少了网络复杂度，从而加快了计算速度。

4.2.2 经典卷积神经网络 AlexNet

对卷积神经网络的定义和基本结构有了了解后,不得不提到卷积神经网络中的经典网络 AlexNet。AlexNet 诞生于 2012 年,是 2012 年 ImageNet 竞赛的冠军 Hinton 和他的学生 Alex Krizhevsky 所设计的^[58]。它继承了 LeNet 的卷积、降采样和权值共享的优秀思想,并将其发扬光大,同时首次在卷积神经网络中应用 ReLU 作为激活函数、采用 dropout 方法防止过拟合、全部采用重叠的最大池化实现降采样(池化核的尺寸大于池化核在数据上滑动的步长)以及引入 LRN(局部响应归一化)技术。其中,LRN 技术是数据在经过激活函数激活、池化后进行的一种处理方法,它借鉴了神经生物学中“侧抑制”的概念,指的是被激活的神经元抑制相邻神经元。它通过创建局部神经元活动的竞争机制,增强响应值较大的神经元,削弱响应值较小的神经元。实践表明了 LRN 技术可以有效增强模型的泛化能力,特别是在使用具有线性性质的 ReLU 作为激活函数时,效果更加明显。

AlexNet 包含特征提取模块和分类模块这两大模块。其中,特征提取模块包括 5 个卷积层,每个卷积层都有卷积运算部分和 ReLU 激活函数激活部分,第 1、2 和 5 层的卷积层有重叠最大化池化运算部分。分类模块包括 3 个全连接层,其中最后一个全连接层是线性部分运算加一个 softmax 层。全连接层包括 dropout 操作、线性运算部分还有 ReLU 激活函数激活部分。其中 softmax 层将线性运算的结果映射到范围为 (0,1) 的实数上,表示对应结果的概率,同时保证了所有结果的和为 1。

Softmax 层的计算如下,下标 j 表示第 j 个输出结果,在分类问题中可以理解为第 j 个类别,此时 softmax 结算结果为该类别的概率:

$$\text{softmax}(z_j) = \frac{e^{z_j}}{\sum_N e^{z_j}} \quad (4.1)$$

其中, z_j 为线性运算部分的计算结果,可以表述为输入值与权重的成绩之和再加上偏置项:

$$z_j = w_j \cdot x + b_j \quad (4.2)$$

其中 w_j 为第 j 个输出结果的权重, b_j 为其偏置项, x 为对应输入。

4.3 冷冻机房关键设备图片准备

4.3.1 分类涉及内容和目的

根据整个图像处理的流程思路,后续图像处理的重点对象是水泵本体和冷冻水管上的温度表和压力表。同时考虑到希望能尽量细化冷冻机房内的自动巡检故障诊断可能涉及的主要设备和部件,为将来的继续研究提供一定的支持。所以,本章的设备分类涉及4个设备:卧式水泵、立式水泵、表盘、压缩机。因为对于不同的设备有不同的巡检需求,对于表盘图像,后续要对其进行自动读数;对于水泵图像,后续要调用它所对应的红外图像进行温度示数识别。虽然本论文研究对象主要集中在水泵等输配侧,对于冷水机组压缩机暂时没有巡检需求,但是考虑到将来对整个自动化巡检技术路线的完善,并实现整个冷冻机房的自动化巡检,因此压缩机也需要进行拍照。设备分类的目的是在自动巡检过程中机器人拍摄到某个设备的图像后,判断图像类别,然后依据此来选择后续对该图像进行处理的具体算法。

4.3.2 关键设备图像类型和训练预处理

表盘图像实例如下所示,在不同的管道上呈现的形态和安置方式大致相同,表盘本身构造完全相同。

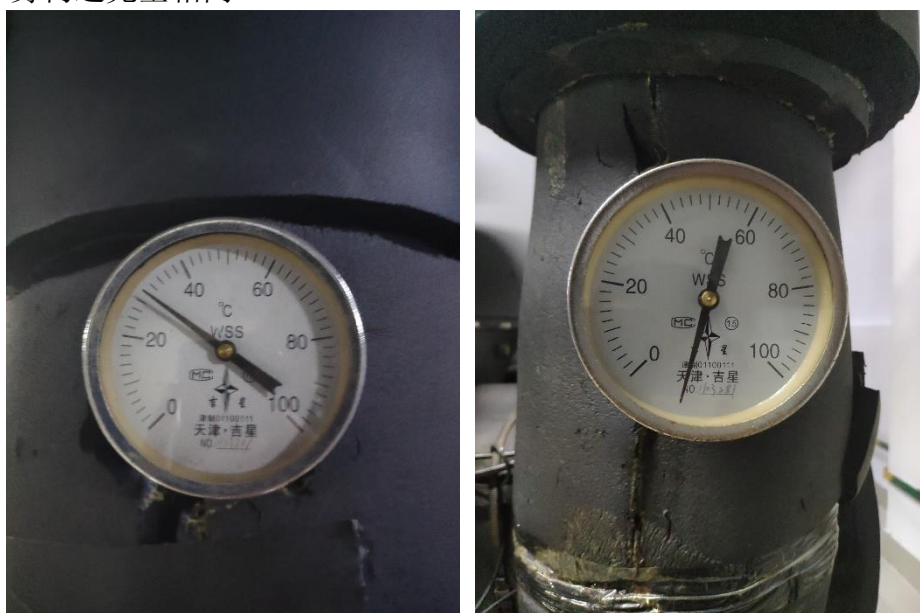


图 4.5 表盘图片示例

不同角度拍摄到的立式水泵图像大致如下,立式水泵在本论文所研究的机房中主要承担输送生活热水的功能。

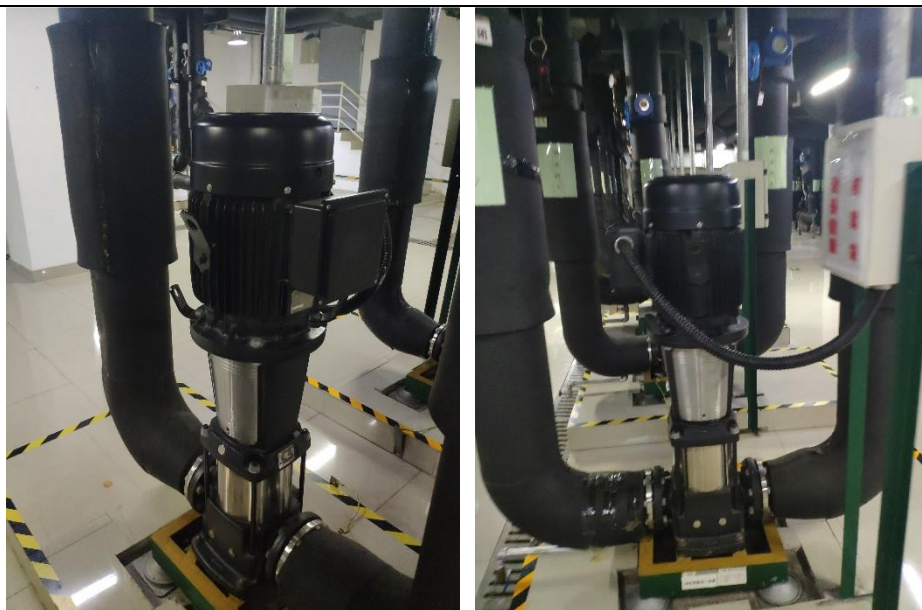


图 4.6 立式水泵图片实例

卧式水泵是该机房水泵的主体形式，新风水循环水泵、天棚水循环水泵、地源水循环水泵等冷冻水侧循环泵以及冷却水循环泵都是卧式水泵，其部分不同角度的大致图像如下：

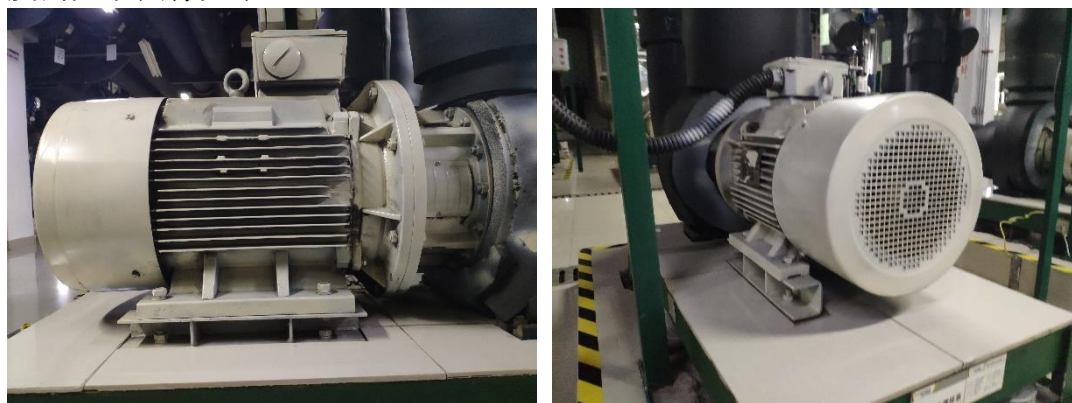


图 4.7 卧式水泵图片示例

该机房共有 8 台螺杆机组，其中 2 台只供热水，2 台只供冷水，辅助降温新风。另外 4 台都是地源热泵机组，2 台用于天棚的制冷季供冷和制热季供热，另外 2 台用于新风的制冷季降温 and 供热季加热。不同角度下拍摄的压缩机图像大致如下：

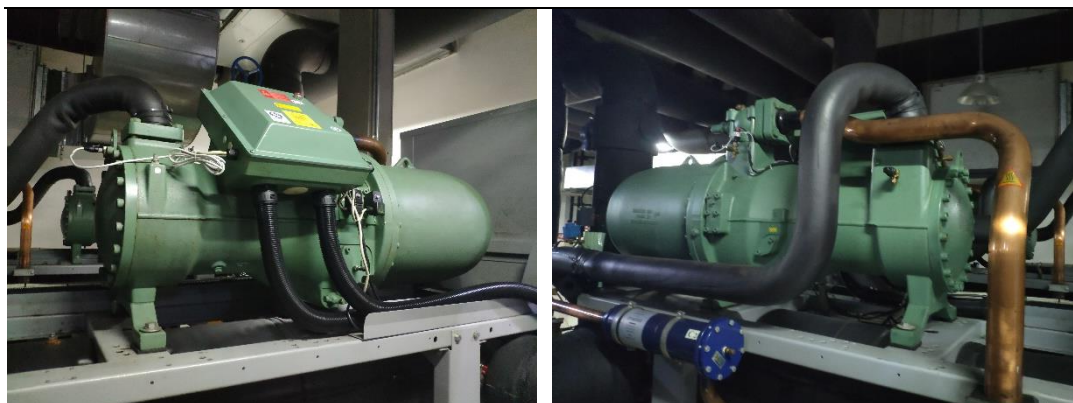


图 4.8 压缩机图片示例

Pytorch 在读取图片时，利用的是它的 torchvision 库中的 ImageFolder 模块。将所有图片分为训练集和验证集两个文件夹，每个文件夹下不同的设备再分别创建一个文件夹，然后利用 ImageFolder 模块读入相应的图片数据就可以。读入的同时，要对图片进行预处理，torchvision 库中的 transforms 模块基于 Python 自带的图像处理库 pillow，可以对训练集图像和验证集图像进行预处理。训练深度神经网络时，图像的预处理非常关键，合理的图像预处理可以增强网络的泛化性能。

AlexNet 要求图像的输入宽高尺寸为 $224*224$ ，但是普遍认为原始论文有误，合理的输入尺寸应该为 $227*227$ 。经过实际检验发现，无论是 $224*224$ 的输入还是 $227*227$ 的输入，在经过多个卷积层和全连接层后，其输出结果的长度都为设定的输出个数（即要分类的类别数）。且在全连接层之前，其经过多个卷积层后输出的张量尺寸都为 $(1, 256, 6, 6)$ ，分别代表输入图像的数目，图像的通道数（深度）、图像的高度、图像的宽度。这也解释了将卷积结果“展平”输入第一个全连接层后的数目等于 9216 ($9216=256*6*6$) 的原因。所以本论文也选用图像的宽高尺寸为 $224*224$ 作为输入网络的图像尺寸。

确定了网络的图像输入尺寸后，需要将输入图像的宽高尺寸剪裁到大小为 $224*224$ 。为了尽可能增强原始图像的表达能，先将训练集的原始图像的宽高尺寸的较小值缩放到 256，然后随机地在较小值的图片上裁剪出宽高尺寸为 $224*224$ 的图像，同时对图像按照一定概率进行水平翻转。然后，对图像的 RGB 三个通道分别进行标准化，标准化之前将像素范围在 $(0,255)$ 之间的原始图片转化成像素范围在 $(0,1.0)$ 之间 Pytorch 单精度张量形式的图片。同时，图片的尺寸格式也从 $(H*W*C)$ 变成了 $(C*H*W)$ ，其中 C 代表图像的颜色通道数，H 代表图像高度，W 代表图像宽度。AlexNet 给出的标准化所选用的三个通道的均值参数和标准差参数是基于 ImageNet 上的数据集抽样的统计结果，具有一定的普适性。但考虑到本论文研究的特殊性，所有图片都在一个特定的冷冻机房拍摄，因此决定对本论文自己的数据集重新计算标准化所需要的全部参数。

通过以上的图像预处理操作，使得训练集图像的随机性加大，数据量加大，从而大大减轻了原始图像过拟合的可能，提升泛化能力。

以卧式水泵为例，首先将图片进行尺寸变换，将原始尺寸的图片的较短边缩放到长度为 256 个像素点，得到下图所示图片：



图 4.9 尺寸变换到最短边为 256 的图片

然后对尺寸变换后的图片随机剪裁为宽和高的长度都为 224 个像素点的图片，裁剪后所示图片如下：



图 4.10 随机剪裁为高和宽都等于 224 的图片

对剪裁后的图片随机进行水平翻转，翻转概率为 50%，即训练集中大约有一半的图片进行水平翻转操作，水平翻转后的图片如下所示：



图 4.11 水平翻转后的图片

之后，将图片转化成 Pytorch 内置的张量格式，张量格式的图片如下所示：



图 4.12 格式转变为单精度张量形式的图片

然后对张量形式的图片进行标准化，将图片的三个色彩通道分别进行标准化。标准化的具体过程如下：每个色彩通道分别求所有像素的平均值和标准差，然后该色彩通道的每个像素点按照下面公式计算标准化后的像素值：

$$Pixel_{out}[channel] = \frac{Pixel_{in}[channel] - Pixel_{mean}}{Pixel_{std}} \quad (4.3)$$

其中，下标 in 代表标准化前的像素值，out 代表标准化后的像素值，mean 代表该通道像素值的平均值，std 代表该通道像素值的标准差。

图像标准化即将图像的香菇丝数据进行中心化处理，使得训练之后模型的泛化效果更好，是增强训练模型性能的一个方法。

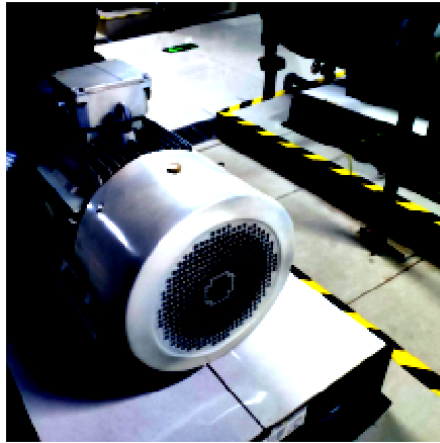


图 4.13 标准化后的图片

4.4 模型的训练准备

4.4.1 AlexNet 模型迁移学习

迁移学习,是利用已经训练好的模型参数来帮助新模型训练。通过迁移学习,新模型只需要在原有模型的基础上进行训练,可以充分利用原有模型的训练结果,并在此基础上适应性修正需要更新的参数来适应新任务。

从 Pytorch 加载获取的 Alexnet 模型已经在具有 1000 个类的 ImageNet 数据集上进行过训练。在对冷冻机房的设备进行分类时,首先需要加载原始模型的特征提取模块,其次根据本论文所要求的实现 4 分类的目的,根据原始模型的分​​类模块组成重新搭建一个分类模块,并将模块的最后一个全连接层的输出改为 4。

模型搭建完成后,因为 Alexnet 之前已经在 ImageNet 上做过较为深度的训练,因此再次使用模型进行训练时,可以关闭特征提取模块的梯度计算,不更新其中的参数;只对分类模块中的梯度重新计算,更新其中的参数。因为特征提取模块是为了提取图像特征,其中最重要的部分就是卷积层中的卷积部分,其上卷积核的参数已经得到过非常好的训练,在提取特征上有很强的泛化能力,所以再次使用模型时不需要对它们的参数进行更新计算。训练分类模块是为了根据提取的特征,重新计算属于某一类别的概率,因此其中的 3 个全连接层中的线性运算部分要根据不同的问题进行更新。通过这样的技巧,减少了模型训练的负担,是对 AlexNet 模型的迁移学习。

4.4.2 模型损失函数

定义好迁移学习的 AlexNet 模型后，训练还需要定义评价模型训练好坏的损失函数，模型参数的优化更新方法等。

针对分类问题，损失函数形式为“交叉熵损失”。对于卷积神经网络的分类问题，交叉熵损失函数利用负对数似然损失计算经过 Softmax 层处理后的输出向量（属于某一类的概率）和真实的类别经过“独热编码”（One-Hot）后的向量之间的损失。计算公式如下：

$$L = -\sum_{j=1}^C y_j \ln\left(\frac{e^{o_j}}{\sum_{k=1}^C e^{o_k}}\right) \quad (4.4)$$

其中， y_j 是经过独热编码后长为分类类别数目 C ，且只有真实类别对应位置为 1 的向量的第 j 个位置的值；上标 o_j 是神经网络的第 j 个输出值（输出值总个数是要分类的类别总数 C ）。

交叉熵损失可以评判训练结果与真实标签的接近程度，接近程度越高，损失值越小；反之，则损失值越大。

4.4.3 模型优化算法

优化算法采用带有动量的随机梯度下降算法，只更新分类模块的梯度，同时采用学习率衰减和权重衰减的技巧。

学习率的大小决定梯度的大小，从而影响参数的更新速度。学习率衰减的目的是使训练初期，参数更新的速度快一些，随着训练的进行，学习率渐渐下降从而使得参数更新的慢一些。这样可以在训练前期，提升训练速度从而降低训练时间，避免收敛到局部最优值；而训练后期，提高训练质量，使得模型的训练效果更好，能准确地获得全局最优的参数值。参数更新如下式所示：

$$w \leftarrow w - \eta \frac{\partial C_0}{\partial w} \quad (4.5)$$

其中 w 是训练参数， η 是学习率， C_0 是损失函数值。

权重衰减技巧通过在损失函数中加入 L2 正则化，从而控制权重的值，使其不会过大，加入 L2 正则化后，损失函数变为：

$$C = C_0 + \frac{\lambda}{2n} \sum_w w^2 \quad (4.6)$$

其中 n 是训练集的样本数，如果采用了 mini-batch 的训练方法，则 n 为每个 mini-batch 中的样本数。 λ 是权重衰减系数。

加入 L2 正则化后，参数更新如下式所示：

$$w \leftarrow w - \eta \frac{\partial C_0}{\partial w} - \frac{\eta \lambda}{n} w = \left(1 - \frac{\eta \lambda}{n}\right) w - \eta \frac{\partial C_0}{\partial w} \quad (4.7)$$

可见，权重 w 的值在更新中有向值减小方向变化的趋势。通过权重衰减的技巧，可以控制权重的值不会过大，从而控制了过拟合，提高了训练模型的泛化性能。

动量的目的是为了加速学习，本质上是积攒了历史梯度的影响，如果当前次的梯度与前一次梯度方向相似，这种趋势在当前更新中会被加强；反之，则当前次的梯度方向会被减弱。

更新速度变为：

$$v = \rho * v + g \quad (4.8)$$

其中 g 是本次计算得到的梯度值， ρ 为动量修正参数， v 为上次计算得到的梯度更新大小。

4.5 模型训练结果和预测展示

4.5.1 模型训练过程

模型总共在 GPU 上训练 20 轮，其中每轮代表训练集中的所有图像都被模型训练过。初始学习率设置为 0.005，学习率衰减设置为每 5 代衰减为之前的 10%，权重衰减系数设置为 0.0001，动量参数设置为 0.9。训练过程在 403 张包含卧式水泵、立式水泵、表盘、压缩机 4 个设备的图片上完成，并在同样的 20 张包含上述 4 个设备的图片上进行验证。部分训练过程如下，train loss 和 val loss 记录了每一轮训练的训练损失和验证损失。Acc 记录了训练和验证的精度。最终训练损失和验证损失都几乎降为 0，训练和验证精度也达到了 100%，训练总共耗时 15 分 10 秒。

```

Epoch 0/19
-----
train Loss: 0.0077 Acc: 0.9404
val Loss: 0.0000 Acc: 1.0000
Epoch 1/19
-----
train Loss: 0.0015 Acc: 0.9950 Epoch 11/19
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 2/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 Epoch 12/19
-----
train Loss: 0.0020 Acc: 0.9926 Epoch 13/19
val Loss: 0.0001 Acc: 1.0000 -----
Epoch 3/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 Epoch 14/19
-----
train Loss: 0.0039 Acc: 0.9826 Epoch 15/19
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 4/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 Epoch 16/19
-----
train Loss: 0.0018 Acc: 0.9926 Epoch 17/19
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 5/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 Epoch 18/19
-----
train Loss: 0.0003 Acc: 0.9975 Epoch 19/19
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 6/19 train Loss: 0.0001 Acc: 0.9975
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 7/19 train Loss: 0.0006 Acc: 0.9950
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 8/19 train Loss: 0.0002 Acc: 0.9975
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 9/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 10/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 11/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 12/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 13/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 14/19 train Loss: 0.0008 Acc: 0.9950
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 15/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 16/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 17/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 18/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000 -----
Epoch 19/19 train Loss: 0.0000 Acc: 1.0000
val Loss: 0.0000 Acc: 1.0000
Training complete in 15m 10s
Best val Acc: 1.000000

```

图 4.14 自定义 Alexnet 训练和验证结果

4.5.2 模型对新图像的分类验证

利用 Pytorch 的 torch 模块的 save 方法和 load_state_dict 方法实现模型的保存和加载，加载后将模型设置为评估模式，并选择性地转移到 GPU 上，然后对新图片进行预测分类。训练模型对新设备图片分类结果如下所示：

图片类别识别结果为：卧式水泵



图 4.15 卧式水泵识别结果

图片类别识别结果为：压缩机



图 4.16 压缩机识别结果

图片类别识别结果为：表盘



图 4.17 表盘识别结果

4.6 本章小结

本章主要利用冷冻机房关键设备的可见光图像实现对设备的分类,为巡检机器人利用基于图像处理的故障诊断算法提供基础。整章首先介绍了深度学习的概念和应用广泛的框架 Pytorch,然后介绍了卷积网络的基本结构和其中的经典网络 AlexNet。之后,对要分类的关键设备图像进行图像预处理,修改 AlexNet 模

型的分类模块并关闭特征提取模块的梯度更新来构造迁移学习新模型，并设定模型的损失函数和优化算法。最后进行模型训练和模型测试，训练过程准确度达到 100%，测试中模型表现同样优秀。

第5章 基于图像处理的冷冻机房表盘示数读取

冷冻机房内有大量的指针式温度表盘和压力表盘,这些表盘指示了冷冻水温度以及进出口压力等,是帮助判断冷水机组和其相应部件是否正常工作,以及管道是否存在保温不利等问题的重要参考。同时开发基于图像处理的表盘示数读取算法可以实现这些数值的自动监测,有相应的工程应用价值。因此本章主要对冷冻机房的温度表盘和压力表盘进行基于图像处理的示数读取相关算法进行开发。

5.1 冷冻机房表盘图像示数读取算法思路

分析图像的表盘指针读数情况,从图像中获取表盘指针的示数,主要考虑进行标准化拍照。所谓标准化是考虑到最终冷冻机房的照片拍摄由巡检机器人实现,所以可以考虑拍摄角度可调节,可以保证拍摄的所有压力表、温度表等照片都可以保证表盘在照片中处于垂直状态,表盘照片拍摄清晰。

在上述假设条件得到保证后,读取图像中指针读数的主要思路是通过识别图像中指针的位置,计算它与某一已知位置(如水平位置、垂直位置等)之间的夹角。通过夹角度数与实际表盘指针与示数的对应位置关系,借助数学关系推算出图像中指针的读数,从而实现表盘图像的自动读数功能。整个实现流程图如下所示,实现过程涉及的图像的缩放、旋转等几何变换,灰度图像的二值化、二值图像的逻辑“与”运算,对二值图像的联通域进行标记,并利用标记的不同联通域提取二值图像的域内特征,利用这一特征进行图像分割。后续还有对提取的表盘指针进行切分提取针尖,判别指针属于哪一个计算域,利用每个域各自的角度计算函数计算指针与指定轴的夹角,多次截取图像不同位置计算夹角求平均值,将夹角指针根据角度与表盘示数的对应关系转换为相应的温度和压力,并将读出的温度和压力根据阈值返回正常与否的判断。

在后续的小节中,会对整个过程中涉及的方法与实现过程进行详细的说明,并通过实例演示该读数算法的可靠性。

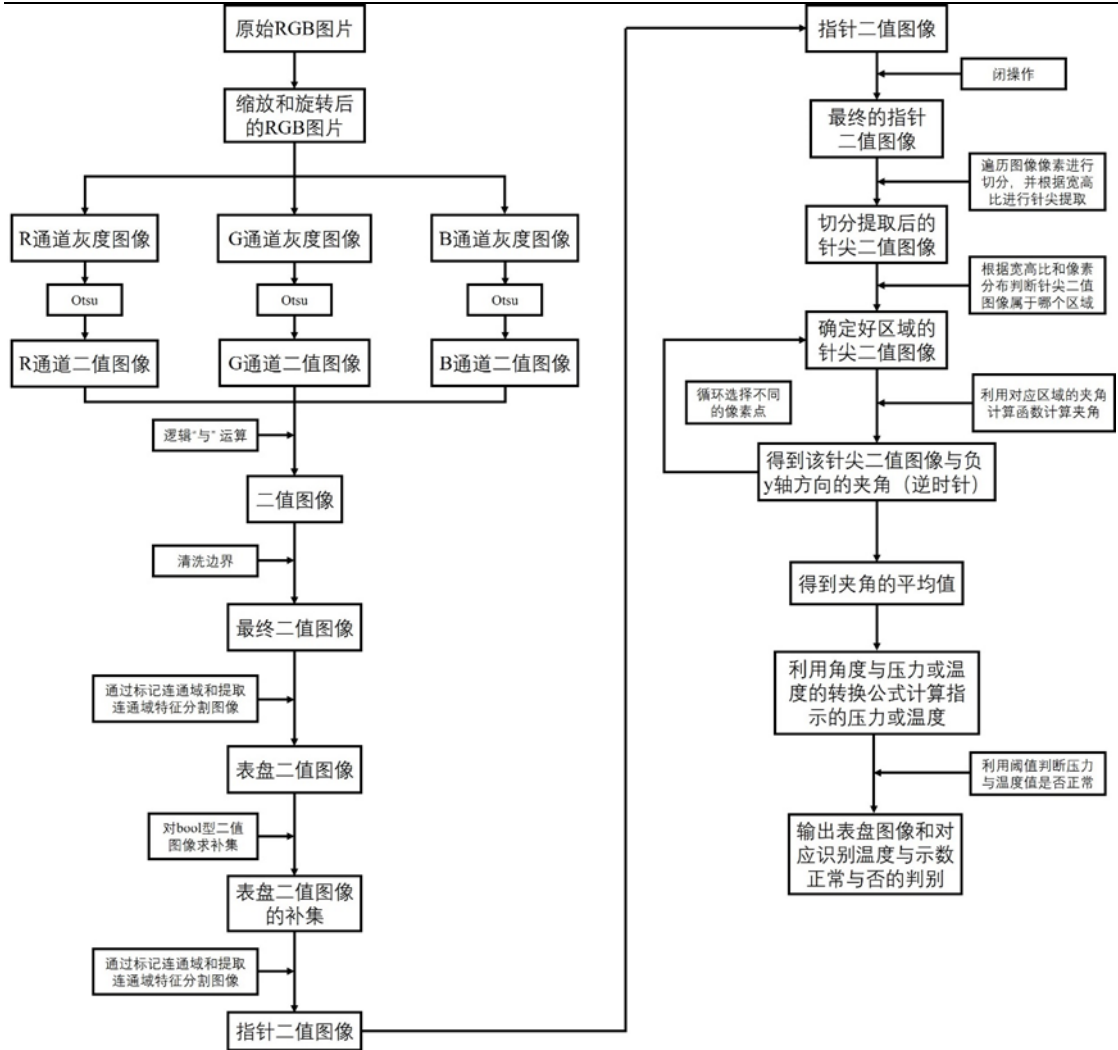


图 5.1 表盘指针图像示数读取流程图

5.2 表盘图像的图像预处理操作

首先，我们拿到表盘的可见光图像后，应对图像的颜色分布进行观察。本论文研究所针对的机房，是一个封闭空间，所有表盘的可见光照片的背景基本处于白色和黑色混合的情形，且表盘自身均为白色的盘底和黑色的指针。考虑到表盘的鲜明的对比色情况，将表盘图像进行适当的预处理操作，可以初步提取我们想要的图像信息，为后续的图像处理操作提供相应的支持。

初始的表盘 RGB 图像的预操作主要分为图像的几何变换，即缩放和旋转操作；还有十分重要的彩色图像的二值化。

5.2.1 表盘图像的缩放和变换操作

图像的缩放主要是调整图像的分辨率，在不损失空间信息的前提下，改变图像的尺寸大小。图像的旋转主要是将默认读入的宽高比小于 1.0 的非正常观察角度的表盘图像通过旋转变换为正常图像，通常是通过逆时针旋转 90 度实现。特别说明，在图像处理中，图像的原点坐标在左上角，因此高度方向是从上到下，宽度方向是从左到右。

原始表盘图像如图所示，类型为 uint8，尺寸为 3024*4032*3（尺寸的三个数字分别代表高、宽和通道数）。



图 5.2 原始表盘 RGB 图像

经过缩放和旋转的表盘图像如下，类型为 uint8，尺寸为 806*605*3。其中缩放大小为 0.2，旋转角度为逆时针 90 度。



图 5.3 经过缩放和旋转的表盘 RGB 图像

5.2.2 表盘图像的二值化操作

表盘图像的二值化操作主要目的是初步区分开表盘图像中的物体和背景，通常一张表盘图像中主要物体是表盘和它周围的其他物体还有背景部分，通过二值化的操作作为后续的精确保留做准备。

我们通过对原始彩色 RGB 图像的 3 个通道分别操作来实现表盘图像的二值化，首先提取原始图像的 R、G 和 B 三个通道，得到三个颜色通道对应的灰度图像。然后对三个颜色通道的灰度图像分别进行二值化后，将三个二值化图像进行逻辑“与”运算，获得最终的二值图像。这样的二值化步骤，相比于直接将 RGB 空间的图像转化为灰度图像后进行二值化操作，基本不会有太大差别，但相对更多地保留了一些图像信息。

然后，我们利用 Otsu 法（大津法，又叫最大类间方差法）对图像进行二值化。该算法假定图像根据前景像素和背景像素可以分成两部分，再次基础上计算分来两部分的最优阈值，使它们的类内方差最小，同时类间方差最大。使用 Otsu 算法，可以有效地区别图像的前景和背景，下面是 Otsu 算法的基本数学原理。

类内方差定义如下所示：

$$\sigma_w^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t) \quad (5.1)$$

其中， ω_i 是被阈值 t 分开的两个类的概率，而 σ_i^2 是这两个类的方差。

Otsu 证明了最小化类内方差和最大化类间方差的效果是相同的，即类内方差和类间方差的和是定值，类内方差取得最小值时，类间方差一定取得最大值。类间方差定义如下：

$$\sigma_b^2(t) = \sigma^2 - \sigma_w^2(t) = \omega_1(t)\omega_2(t)[\mu_1^2(t) - \mu_2^2(t)] \quad (5.2)$$

其中， ω_i 是被阈值 t 分开的两个类的概率， μ_i 是每个类的均值。

假定灰度图的像素直方图有 L 个整数像素值（取值范围为 $[0,255]$ 之间）计算类概率 $\omega_1(t)$ 和 $\omega_2(t)$ 的表达式如下：

$$\omega_1(t) = \sum_{i=0}^{t-1} p(i) \quad (5.3)$$

$$\omega_2(t) = \sum_{i=t}^{L-1} p(i) \quad (5.4)$$

其中， $p(i)$ 是 i 像素值下的像素占该图像全部像素的比例，且 $\omega_1 + \omega_2 = 1$ 。

计算每个类的均值 $\mu_1(t)$ 和 $\mu_2(t)$ 的表达式如下：

$$\mu_1(t) = \frac{\sum_{i=0}^{t-1} ip(i)}{\omega_1(t)} \quad (5.5)$$

$$\mu_2(t) = \frac{\sum_{i=t}^{L-1} ip(i)}{\omega_2(t)} \quad (5.6)$$

具体实现 Otsu 算法的过程可以分成如下几个步骤：

计算每个强度等级下的直方图和概率；

设定初始的 $\omega_i(0)$ 和 $\mu_i(0)$ ；

遍历所有可能的阈值，从 $t=1$ 一直到灰度图像的最大强度（即最大像素）：更新 ω_i 和 μ_i ，然后计算类间方差 $\sigma_b^2(t)$

所求的阈值 t 为最大类间方差 $\sigma_b^2(t)$ 下的阈值。

原始图像和它的 R、G、B 三个通道的灰度图像分别如下所示。

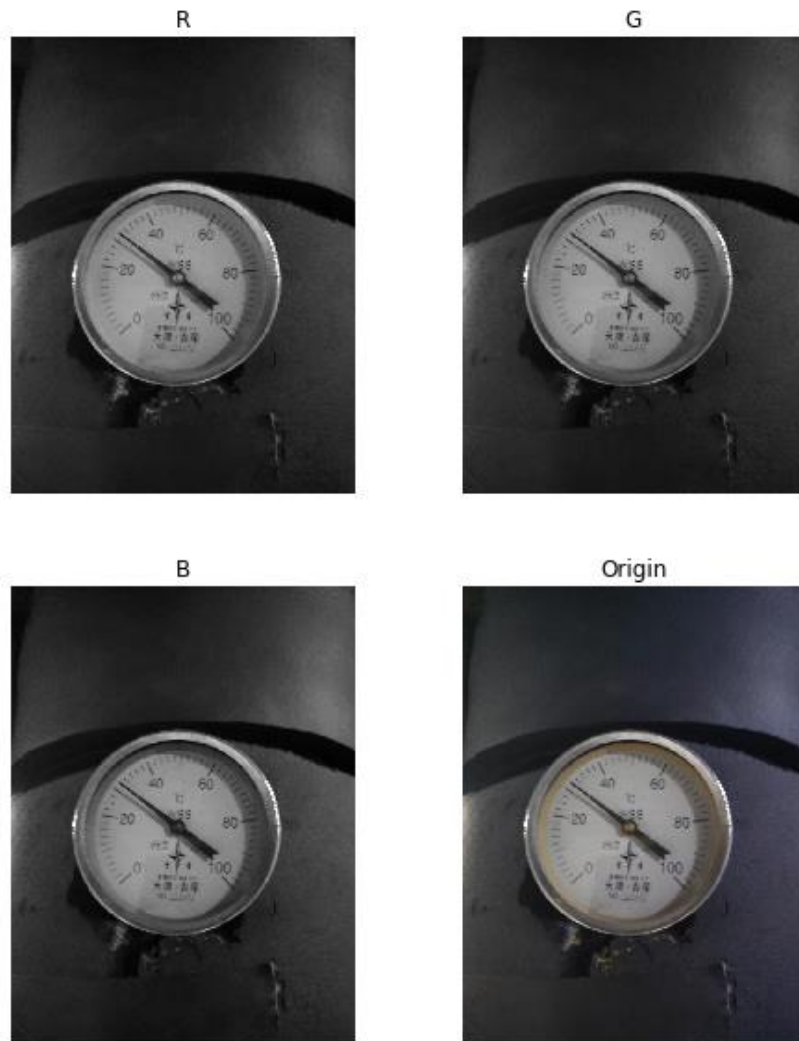


图 5.4 原始图像和它的 RGB 三个通道的灰度图

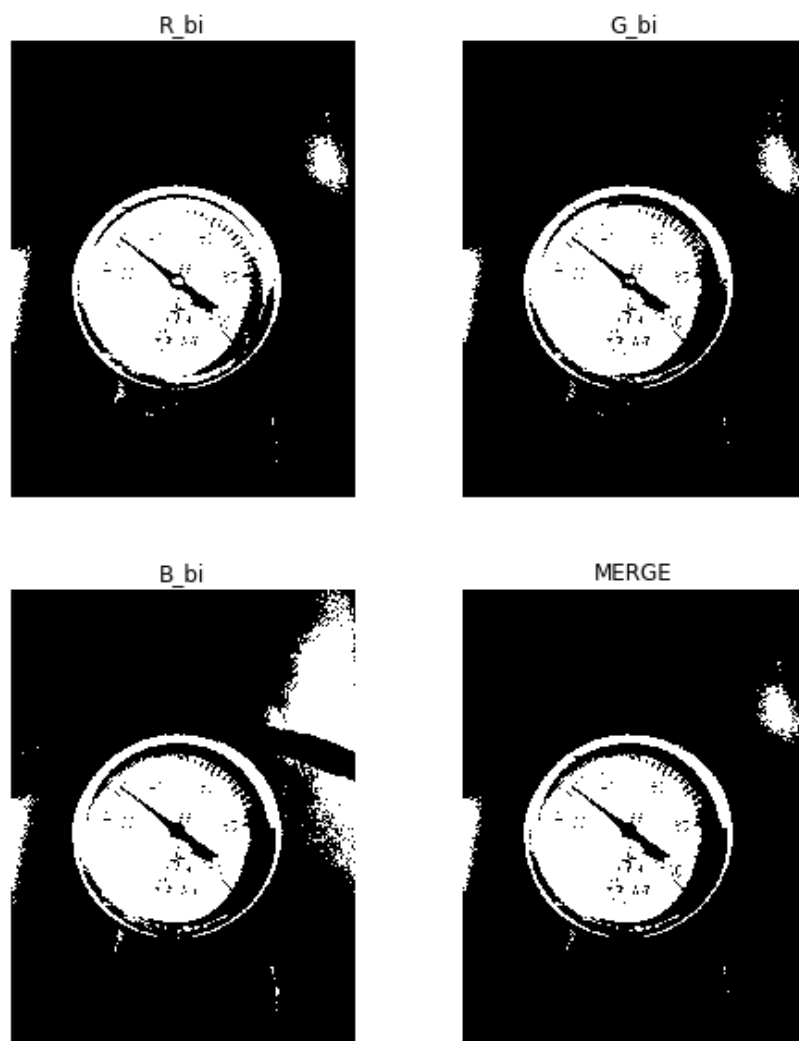


图 5.5 原始图像的 RGB 三通道图像的 Otsu 后的二值图像和逻辑“与”运算后的合成图像

对比展示将原始 RGB 图像利用 RGB 转灰度公式直接灰度化后，再二值化得到的结果和原始 RGB 图像三个彩色通道分别二值化后再合并得到的结果如下：

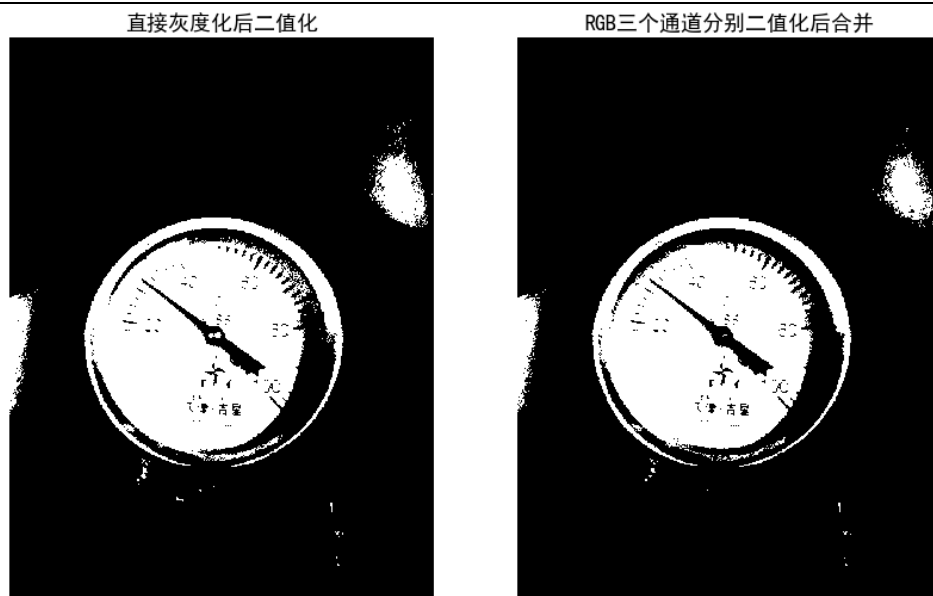


图 5.6 直接灰度化后二值化与 RGB 三通道分别二值化合并的结果

RGB 三个通道分别二值化合并得到的图像比直接灰度化后二值化得到的图像中，两者都为 bool 型图像，虽然没有明显的不同，但指针部分更加显著一些，因此本论文选择 RGB 三通道分别利用 Otsu 算法二值化后再合并的方法，对表盘图像进行二值化操作。

5.3 表盘图像中指针的提取

5.3.1 基于区域标记和区域特征统计的连通域提取方法

获得表盘的二值图像后，下一部任务是将表盘的指针部分从图像中分割出来。由于 bool 型二值图像像素只有 2 种类型，分别是 True 和 False，也可以用 1 和 0 表示，1 代表白色点，skimage 库默认为物体部分，0 代表黑色点，skimage 库默认为背景部分。Python 的图像处理库 skimage 中的 label 方法，可以用整型数字代表的数组标记二值图像中的连通域。其中的 regionprops 方法又可以对标记好的二值图像中的所有的联通域进行特征分析，这些特征包括连通域的像素数，连通域的矩形框选框以及其面积，连通域的中心坐标等等，最终返回所有联通域的特征分析结果集合，调用相应的特征只需在每个连通域的结果集合后选择需要的方法即可。本节的内容主要借助 skimage 中的 label 和 regionprops 功能，也是本章中至关重要的一部分。

5.3.2 表盘图像中表盘部分的提取

观察表盘的二值图像可以看出，其中表盘部分就是最大的连通域，我们只需要利用 label 方法标记连通域，然后通过 regionprops 方法提取连通域区域特性中像素数最多的部分，这一部分在 label 标记后的图像中其像素值均为同一正整数，且这一正整数出现次数最多。然后将表盘二值图像中不等于这一连通域对应正整数的部分赋值为 0，即为黑色像素，等于这一部分的赋值为 1，即为白色像素。

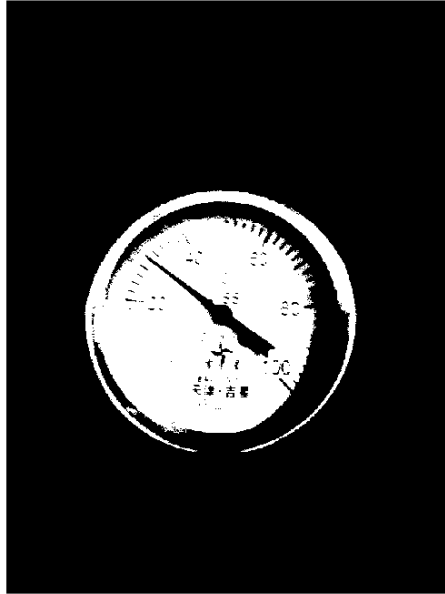


图 5.7 经过区域特征提取后的表盘部分图像

经过这一次区域特征提取后，表盘图像中小的连通域被去除，整个图像变的干净了许多，只留下了表盘部分的图像。

5.3.3 表盘图像中指针部分的提取

表盘部分提取成功后，我们要想办法提取指针部分，从上一小节中表盘部分提取后的二值图像中可以看到，指针部分是背景颜色（黑色），为了提取指针部分，我们先对表盘部分提取后的二值图像求补集，将其中的指针部分的颜色变成物体颜色（白色）。求补集后的图像如下，为了显示图像的标记，在图像中保留了高度和宽度部分的坐标。

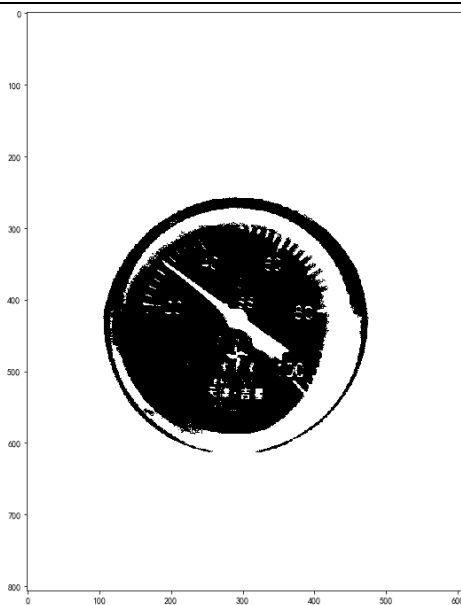


图 5.8 表盘部分图像求补集后的图像

求补集后的图像中，除了表盘外的白色背景大连通域以外，最大的连通域就是指针部分。因此类似于提取表盘部分的方法，我们这次提取区域特征后排序，找到连通域区域特性中区域像素数第二大的区域，这一部分在 label 标记后的图像中其像素值均为同一正整数，且这一正整数出现次数第二多。然后将表盘二值图像中不等于这一连通域对应正整数的部分赋值为 0，即为黑色像素，等于这一部分的赋值为 1，即为白色像素。

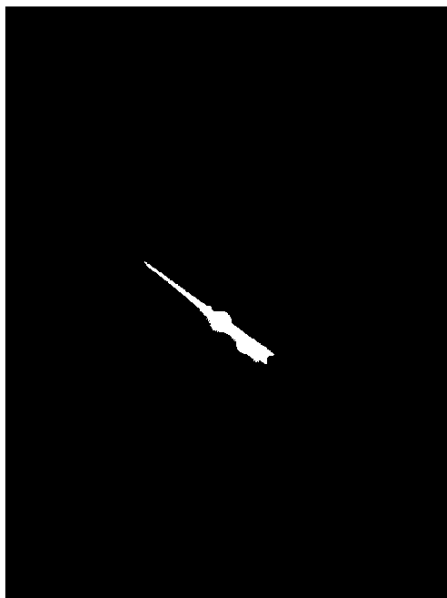


图 5.9 经过区域特征提取后的指针部分图像

经过这一次区域特征提取后，表盘图像中指针部分的连通域被提取出来，其余部分被处理为背景。2 次标记图像连通域和根据区域特征提取所要区域的图像操作成功地提取出表盘二值图像中的指针部分。

5.4 指针二值图像的形态学操作

对指针图像进行形态学操作，主要目的是消除图像中的一些不整齐的部分，使得图像边界更圆滑，为后续计算指针角度算法的设计提供高质量的图像支持。

形态学操作主要是基于像素区域的形态特征来获取图像中有价值的部分，使得接下来的图像处理能够关注如形状边界，内部连通域等有价值的图像形态部分。

二值图像的基本形态学运算有图像腐蚀、图像膨胀 2 种，同时，对这两种方法按照实施顺序进行组合可以获得开操作和闭操作这两中形态学运算方法。

5.4.1 形态学操作的腐蚀和膨胀

所有的图像形态学操作都是针对图像中的前景物体进行的，本论文研究的表盘图像，我们要提取的物体颜色主体是白色。因而我们用白色像素表示物体，用黑色像素表示背景。

本章图像处理的基础是 Python 中的 skimage 库，其在对二值图像进行形态学操作时，默认白色的像素点（二值图像中灰度值为 1 的像素）是前景，即是物体本身；而黑色的像素点（二值图像中灰度值为 0 的像素）是背景。这一假设与本论文研究的表盘图像的情形吻合，可以在不转换前景和背景的前提下进行相应的操作。

图像腐蚀操作描述为：对平面 Z^2 上的元素集合 A 和 S ，利用 S 对 A 进行腐蚀操作，记作 $A \ominus S$ ，数学表述如下：

$$A \ominus S = \{z | (S)_z \subseteq A\} \quad (5.7)$$

公式的直观意义为：当腐蚀算子 S 在平面 Z^2 上移动时，若 S 的原点平移在平面某一位置时 S 全部在集合 A 中，则所有这些位置的点的集合即为用 S 腐蚀 A 后得到的图像。

腐蚀操作可以侵蚀消除物体的边界，且腐蚀后的物体与原本物体的形状和腐蚀算子本身形状有关。如果物体 A 整体上大于腐蚀算子 S ，用 S 对 A 腐蚀最终结果使得 A 的尺寸比操作之前有所减小，类似损失掉一圈边界的样子，减小程度取决于腐蚀元素 S 的形状；如果物体 A 整体上小于腐蚀算子 S ，则用 S 对 A 腐蚀后物体 A 将消失；如果物体本身只有一些联通部分小于腐蚀算子，则用 S 对 A 腐蚀后，物体 A 会在较为狭长和细小和联通部分分离为两部分图像。

图像膨胀操作描述为：对平面 Z^2 上的元素集合 A 和 S ，利用 S 对 A 进行膨胀操作，记作 $A \oplus S$ ，数学表述如下：

$$A \oplus S = \left\{ z \mid (\hat{S})_z \cap A \neq \phi \right\} \quad (5.8)$$

公式的直观意义为：当膨胀算子 S 在平面 Z^2 上移动时，若 S 的原点平移到平面某一位置时 S 对于其自身的原点的映像 \hat{S} 和 A 存在至少单个像素的重合的部分，则所有这些位置的点的集合即为用 S 膨胀 A 后得到的图像。

膨胀操作可以扩大外延物体的边界，且膨胀后的物体与原本物体的形状和膨胀算子本身形状有关。如果在图像二值化后，其中一些狭长和细小的部分断开，此时就可借助膨胀桥接断裂的缝隙，将两个靠的非常近的物体重新连接成一个物体，保证二值图像与原始灰度或彩色图像的一致性。

5.4.2 形态学操作的开操作和闭操作

开操作对图像先腐蚀然后膨胀，闭操作对图像先膨胀然后腐蚀。两者都结合了腐蚀和膨胀的不同特性，是图像形态学操作中最常用的两种方法。

开运算可以移除小亮点，即消除细毛刺，并使狭窄的连接断开，从而使得图像的轮廓变得更光滑

闭运算可以移除小黑点，即填充小的孔洞，并弥合狭窄的间断，从而使得图像的轮廓变得更加光滑。

5.4.3 指针图像的开运算和闭运算

腐蚀算子的选择对开运算和闭运算的结果影响很大，通常对于边界不规则的物体，圆盘算子的结果更好。确定圆盘算子后，下一步要确定圆盘算子的尺寸。取圆盘算子的半径分别为 2、3、4，计算开、闭运算后指针图像的结果。

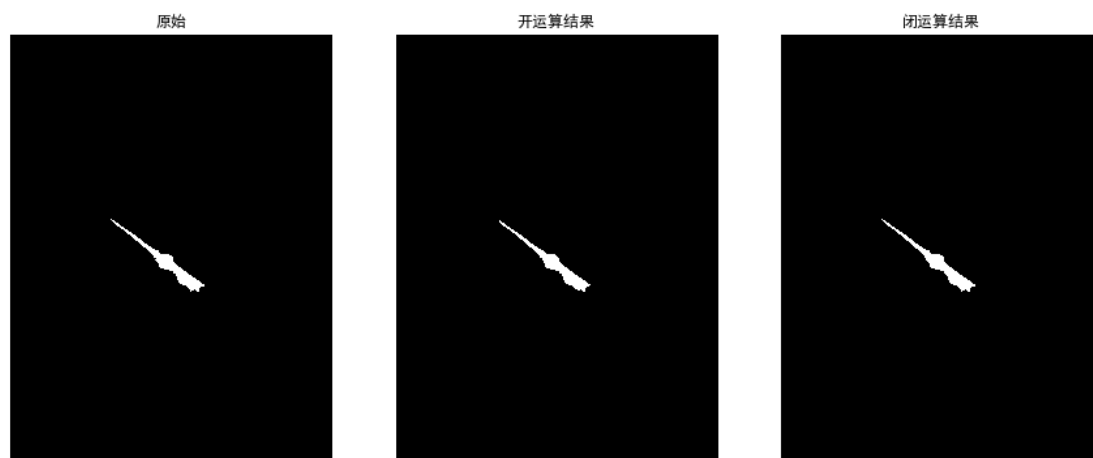


图 5.10 半径为 2 的圆盘算子对原图做开闭运算后的结果

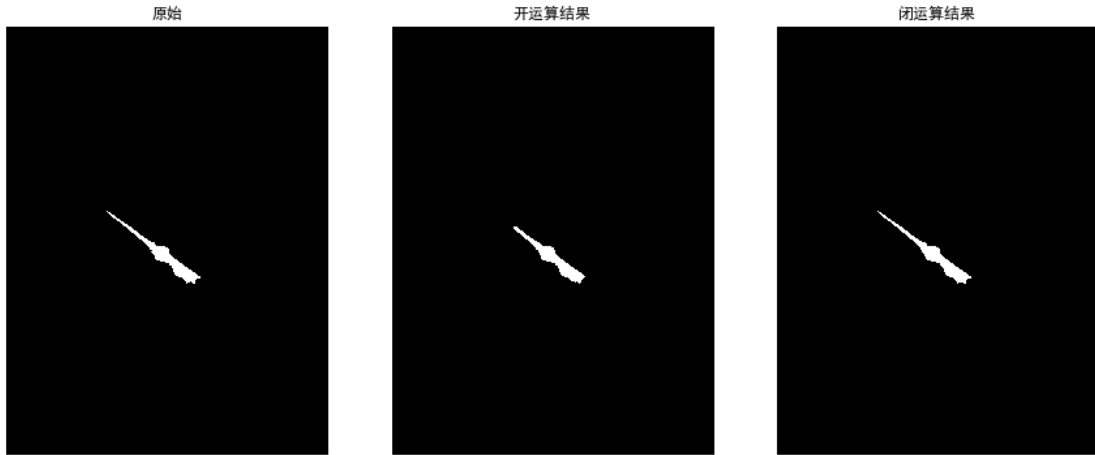


图 5.11 半径为 3 的圆盘算子对原图做开闭运算后的结果

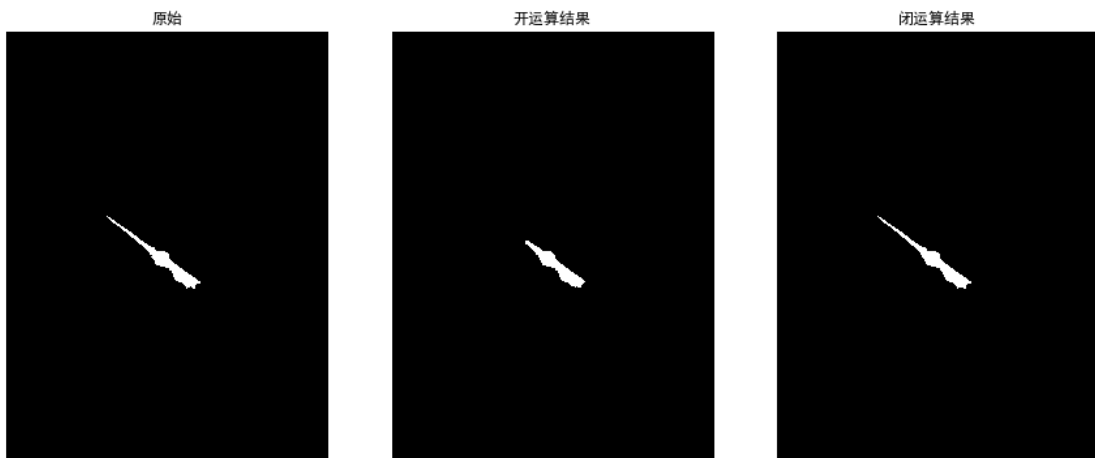


图 5.12 半径为 4 的圆盘算子对原图做开闭运算后的结果

观察 3 次开运算和闭运算的结果，可以看出对于细长的指针来说，圆盘算子的半径越大，开运算会明显地使指针的针尖变形，闭运算则不会明显地改变指针的整体结构。所以，对细长的指针来说，我们选择闭运算来使得指针针尖的部分细小的断裂部分弥合，整个针尖变的更光滑。

5.5 针尖部分图像的提取

得到指针的闭运算后的图像后，需要提取指针的针尖部分，指针的针尖提取包括两部分内容。首先要获得图像中有指针的部分，将图像裁剪到指针在图像中位置最大化的情况。其次要根据裁剪后图像的宽高比是否大于某个数值，决定需要在宽度方向上继续裁剪还是在高度方向上继续裁剪。

5.5.1 依据像素裁剪指针图像

由于指针部分的二值图像中只有 2 种像素形式，指针为白色像素点的集合，所以只需要沿着特定方向遍历图像，找到白色像素点的时候退出循环遍历，就可以确定白色像素点在指针部分图像中的位置。

首先需要分别确定指针部分在整体图像中的上下边界，即要寻找对应的像素坐标。确定上边界先沿着指针部分图像的高度方向依次遍历图像，在每一个高度值下，再沿着宽度方向遍历图像，找到第一个值为 1 的像素点后，返回这一像素点的高度方向位置坐标。确定下边界先沿着高度的反方向依次遍历图像，在每一个高度值下，再沿着宽度方向遍历图像，找到第一个值为 1 的像素点后，返回这一像素点的高度方向位置坐标。

将原图像的高度方向裁剪到这两个高度位置坐标之间，即完成了指针的上下边界提取。

其次确定指针分体在整体图像中的左右边界，同样需要分别确定。同时确定左右边界的过程在上下边界提取后的指针部分图像中进行操作，可以减少遍历的运算量。确定左边界先沿着图像的宽度方向依次遍历图像，在每一个宽度值下，再沿着高度方向遍历图像，找到第一个值为 1 的像素点后，返回这一像素点的宽度方向位置坐标。确定右边界先沿着宽度的反方向依次遍历图像，在每一个宽度值下，再沿着高度方向遍历图像，找到第一个值为 1 的像素点后，返回这一像素点的宽度方向位置坐标。

将提取好上下边界的指针部分图像的宽度方向裁剪到这两个宽度位置坐标之间，即完成了指针的左右边界提取。

提取后上下边界和左右边界后，就完成了指针部分图像的裁剪。

5.5.2 指针图像的边缘检测

对裁剪后的指针部分图像利用 Canny 算法进行边缘检测，来检查指针的轮廓是否清晰。Canny 算法在获取梯度时使用了 Sobel 算子，主要算法步骤分为四步：

- (1) 先利用高斯平滑滤波器平滑图像；
- (2) 使用水平和垂直的 Sobel 算子计算图像梯度，并计算得到边缘强度(梯度的二范数)；
- (3) 遍历每个像素点以寻找可能的边缘。考虑到边缘可能会指向不同方向，通过跟踪水平和垂直方向的 Sobel 算子的踪迹和相对尺度把所有点归为 4 类：水平、垂直、对角线、反对角线。对于每个点，判断它的法线方向(通过插值确定潜在边缘方向)和反方向的点像素是否大于它。

对于潜在边缘方向边缘强度大于本身点的，认定为候选边缘，反之则排除这一点。这一步骤又被叫做“非极大值抑制”。

- (4) 使用“滞后阈值”来限定多大的亮度梯度是边缘，分别设定低阈值 t_1 ，高阈值 t_2 ， $t_1 \leq t_2$ 。因为边缘的梯度相对更大，因此首先标识像素值大于 t_2 的点为边缘点，而小于 t_1 的点为非边缘点。在 t_1 与 t_2 值之间的点，从已确定的边缘点展开 8 个领域方向的搜索，可以到达的点为边缘点，反之则是非边缘点。因此在 t_1 与 t_2 值之间的像素点，只有连接到一个高于高阈值时才会被保留

通常，直接利用 Sobel 算子产生的边缘有强弱之分，计算量小，对于一般的二值图像边缘检测的效果好。而 Canny 算法产生的边缘细，且没有强弱之分，相对计算量大，但是准确度高。其中的第一步选择高斯滤波器对图像进行模糊时设置滤波器的标准差参数至关重要，对于一般的二值图像，只需要极小的标准差，若标准差很大，则图像失真严重。对于较为复杂的灰度图，可以设置较高的标准差，这样对图像的平滑效果较好，有利于后续的计算过程进行边缘提取。

本论文采用标准差为 0.1 的 Canny 算法进行边缘提取，来检查指针图像的边缘是否足够连续，为之后的角度计算提供参考。



图 5.13 剪裁后图片及其 Canny 检测后的图片

对比上图，可以看出指针图像有很清晰的边缘，连续程度很高，对后续的角度计算也提供了足够的图像质量支持。

5.5.3 针尖部分的提取

获得裁剪后的指针部分图像后，就可以提取指针部分图像的针尖部分，即指针的顶端部分。针尖部分是指针比较均匀变化的部分，因此利用针尖部分进行后续角度的计算更为准确可靠。

提取针尖要对图像进行再次裁剪，首先要判断裁剪后的指针图像的宽高比，当指针图像的宽高比大于某一数值时（对应与指针与水平方向夹角较小），在宽度方向对指针图像进行裁剪；反之当宽高比小于某一数值时（对应与垂直方向夹角较小），在高度方向对指针图像进行裁剪。为了保证只在指针趋向于 12 点钟附近时（此时指针与垂直方向夹角较小）在高度方向对指针图像进行裁剪，因此要确保区分按高度方向和宽度方向裁剪的宽高比的阈值大于表盘指针处于 0 刻度时对应的宽高比。因为 0 刻度与垂直方向的夹角大约在 40° 到 45° 之间，对应宽高比在 1 到 1.2 之间，所以确定 1.2 为区分两种裁剪方式的阈值。即当指针部分图像的高度大于宽度的 1.2 倍，将其在高度方向裁剪为原来的 $1/3$ ；反之，当指针部分图像的高度小于等于宽度的 1.2 倍，将其在宽度方向裁剪为原来的 $1/3$ 。



图 5.14 在宽度方向裁剪后的针尖部分图片

5.6 判定指针在表盘中所处的区域

有了针尖的部分的图像后，下一步就是计算指针相对于高度方向的夹角（这一方向对应与表盘中的示数的逆时针递增方向相吻合）。

计算指针的夹角的核心思路是通过在宽度方向上截取这一方向上的指针部分中点，截取多个中点后，利用两点求反正切三角函数的方法计算夹角。因此，为了保证算法的可靠性和普适性，在计算指针夹角之前，需要人为地对指针目前指示所处区域进行分区。对于不同分区的指针，在计算角度时需要调整计算方法。

5.6.1 表盘 4 个区域的确定

结合上一节剪裁的内容,对指针的分区主要考虑到指针从 0 旋转到满刻度的过程中指针与高度方向的夹角会发生变化,在计算这一夹角时需要进行不同的调整。

特别地,当指针旋转到 12 点钟附近的位置时,图像的高宽比会大于 1.2,此时要在高度方向上对图像进行剪裁,之后计算角度时,直接在宽度方向计算角度会有没有像素的宽度方向出现。因此,当在高度方向裁剪图像后,需要将图像逆时针旋转 90 度,然后计算角度,最后计算真实角度时只需要进行一定的补偿即可。

另外,指针在刻度盘的左半部分和右半部分时,计算与高度方向夹角时也不相同,需要分开考虑,设置不同的分区。

综上,计算指针角度时,将指针分为 4 个区域。如下图温度表盘所示,指针从 0 刻度到最大刻度变化的过程中,依次经过区域 1 到区域 4。下面具体介绍分区的算法流程,整个流程写成函数代码的形式,具体代码参加附录 A。

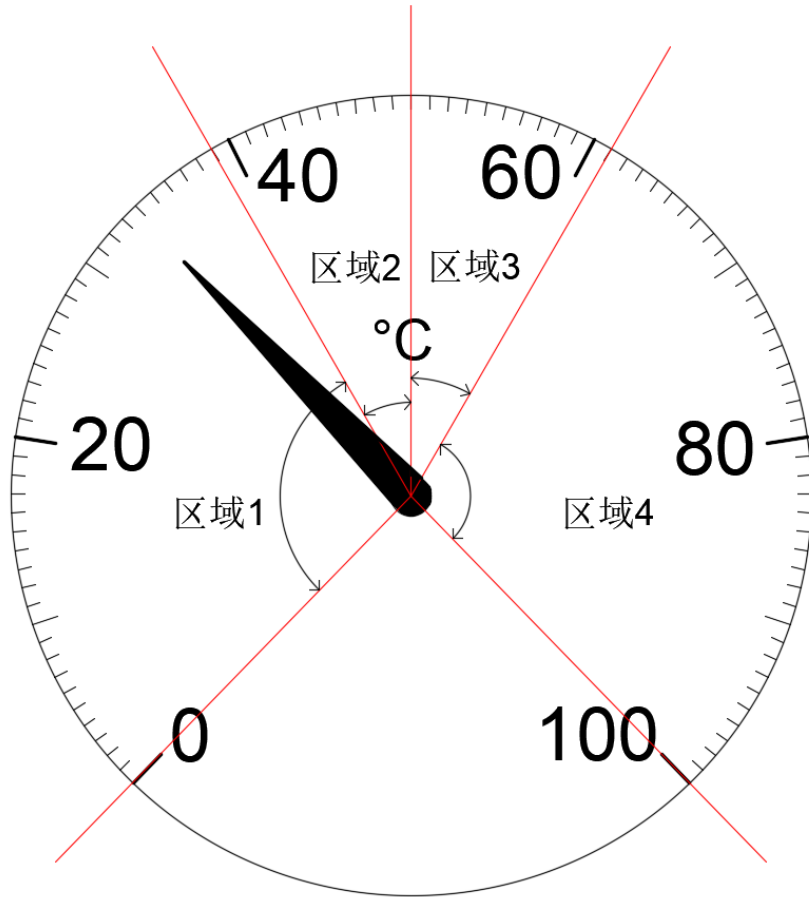


图 5.15 区域划分示意图 (温度表为例)

5.6.2 依据高宽比初步划分区域 1、4 和区域 2、3

指针部分图像的高宽比决定了指针趋向于 12 点钟位置的程度，上一节设定当指针部分图像的高宽比大于 1.2 时，将其在高度方向裁剪。指针部分图像的高宽比等于 1.2 时，表明指针与水平方向夹角为 50° ，这时指针出现在图 5.15 的区域 2 和区域 3 中。而当指针部分图像的高宽比小于 1.2 时，这时指针出现在图 5.15 的区域 1 和区域 4。

这样，通过高宽比将指针初步划分为区域 1、4 和区域 2、3 两种情形。

5.6.3 依据指针形状特性划分区域 1-4

有了区域 1、4 和区域 2、3 的划分，再进一步确定指针具体属于哪个区域。

对于区域 1、4，位于区域 1 的指针图像沿着宽度是由细渐渐变粗，位于区域 4 的指针图像沿着宽度是由细渐渐变粗。因此，将指针图像在宽度方向上截取 2 个坐标，前者可以是小于图像一半宽度的一处坐标，后者可以是大于图像一半宽度的一处坐标。为了保证识别准确性，本论文取第一个宽度坐标 w_1 为图像宽度的八分之一，取第二个宽度坐标 w_2 为图像宽度的八分之七，其中 $w_1 < w_2$ 。然后在这两个宽度坐标下，沿着高度方向进行遍历，统计像素值为 1（白色）的像素数。如果在宽度 w_1 下白色像素数小于宽度 w_2 下白色像素数，则指针位于区域 1；反之，指针位于区域 4。

对于区域 2、3，以 12 点钟方向划分，位于区域 2 的指针在图像中呈现出左倾的特征，位于区域 3 的指针在图像中呈现出右倾的特征。为了计算方便，首先将位于这两个区域的指针图像逆时针旋转 90° ，这时位于区域 2 的指针在图像中偏向下方，位于区域 3 的指针在图像中偏向上方。因此，将指针图像在宽度方向上截取 2 个坐标，前者可以是小于图像一半宽度的一处坐标，后者可以是大于图像一半宽度的一处坐标。同样取第一个宽度坐标 w_1 为图像宽度的八分之一，取第二个宽度坐标 w_2 为图像宽度的八分之七，其中 $w_1 < w_2$ 。然后在这两个坐标下，沿着高度方向遍历，找到第一个像素值为 1 的点的高度坐标，和最后一个像素值为 1 的点的高度坐标，然后求高度平均值得到指针在这一高度方向上的中点坐标（编写一个求中点函数）。如果在宽度 w_1 下的中点坐标值大于宽度 w_2 下的中点坐标值，表明指针偏向下方，则指针属于区域 2；如果在宽度 w_1 下的中点坐标值小于宽度 w_2 下的中点坐标值，表明指针偏向上方，则指针属于区域 3。如果两者相等，则表明指针指向 12 点钟方向。

5.7 计算指针的指示角度

确定好指针在表盘中所处的区域后，下一步就是计算指针与图像高度方向（传统的负 y 轴方向）沿逆时针方向的夹角。每个区域的指针采用不同的计算方法计算夹角，下面依次说明。

5.7.1 区域 1 内指针的夹角计算

对于区域 1 内的指针，只需要在宽度方向上找两个坐标 y_1 和 y_2 ，使得 $y_1 < y_2$ ，然后利用上一节提到的求中点函数计算 y_1 和 y_2 宽度坐标下的高度方向指针中点坐标分别为 x_1 和 x_2 。当指针与高度方向夹角小于 90° ， $x_1 > x_2$ ；反之， $x_1 < x_2$ 。计算角度公式如下：

$$\text{Angle} = \arctan\left(\frac{y_2 - y_1}{x_1 - x_2}\right), \text{其中 } \text{Angle} \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right) \quad (5.9)$$

反正切函数的计算结果在 $-\pi/2$ 到 $\pi/2$ 之间。如果计算结果落在 $-\pi/2$ 到 0 之间，说明实际指针与高度方向夹角大于 $\pi/2$ ，由于正切函数是以 π 为周期的周期函数，只需将负数计算结果增加 π 即可得到实际夹角。计算公式如下：

$$\begin{cases} \text{Angle} = \text{Angle}, \text{Angle} \in (0, \frac{\pi}{2}) \\ \text{Angle} = \text{Angle} + \pi, \text{Angle} \in (-\frac{\pi}{2}, 0) \end{cases} \quad (5.10)$$

最后将弧度制转化为角度制即可得到以度数为单位的夹角。特别地如果 $x_1 = x_2$ ，则直接得到夹角为 90° ，无需计算。

5.7.2 区域 2 内指针的夹角计算

对于区域 2 的指针，首先需要将指针逆时针旋转 90° 到区域 1，寻找坐标 y_1 和 y_2 的方法，以及计算坐标 x_1 和 x_2 的方法同区域 1，只是 x_1 肯定大于 x_2 。最后只需给计算结果增加 $\pi/2$ 即可得到实际夹角。计算角度公式如下：

$$\text{Angle} = \arctan\left(\frac{y_2 - y_1}{x_1 - x_2}\right) + \frac{\pi}{2}, \text{其中 } \text{Angle} \in \left(\frac{7\pi}{9}, \pi\right) \quad (5.11)$$

最后将弧度制转化为角度制即可得到以度数为单位的夹角。特别地如果 $x_1 = x_2$ ，则直接得到夹角为 180° （为了防止某些取到的点高度坐标相等，保证算法稳定性），无需计算。

5.7.3 区域 3 内指针的夹角计算

对于区域 3 的指针，首先需要将指针逆时针旋转 90° 到区域 1，寻找坐标 y_1 和 y_2 的方法，以及计算坐标 x_1 和 x_2 的方法同区域 1，只是不同于区域 2 的指

针旋转后的结果，区域 3 的指针旋转后 x_1 肯定小于 x_2 。最后只需给计算结果增加 $3\pi/2$ 即可得到实际夹角。计算角度公式如下：

$$\text{Angle} = \arctan\left(\frac{y_2 - y_1}{x_1 - x_2}\right) + \frac{3\pi}{2}, \text{其中 } \text{Angle} \in \left(\pi, \frac{11\pi}{9}\right) \quad (5.12)$$

最后将弧度制转化为角度制即可得到以度数为单位的夹角。特别地如果 $x_1 = x_2$ ，则直接得到夹角为 180° （为了防止某些取到的点高度坐标相等，保证算法稳定性），无需计算。

5.7.4 区域 4 内指针的夹角计算

对于区域 4 内的指针，同区域 1 一样，首先要在宽度方向上找两个坐标 y_1 和 y_2 ，使得 $y_1 < y_2$ ，然后利用上一节提到的求中点函数计算 y_1 和 y_2 宽度坐标下的高度方向指针中点坐标分别为 x_1 和 x_2 。当指针与高度方向夹角大于 270° ， $x_1 < x_2$ ；反之， $x_1 > x_2$ 。计算角度公式如下：

$$\text{Angle} = \arctan\left(\frac{y_2 - y_1}{x_1 - x_2}\right), \text{其中 } \text{Angle} \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right) \quad (5.13)$$

与区域 1 有所不同，如果计算结果落在 0 到 $\pi/2$ 之间，说明实际指针与高度方向夹角小于 $3\pi/2$ ，只需将计算结果增加 π 即可得到实际夹角。如果计算结果落在 $-\pi/2$ 到 0 之间，说明实际指针与高度方向夹角大于 $3\pi/2$ ，由于正切函数是以 π 为周期的周期函数，只需将负数计算结果增加 2π 即可得到实际夹角。计算公式如下：

$$\begin{cases} \text{Angle} = \text{Angle} + \pi, \text{Angle} \in \left(0, \frac{\pi}{2}\right) \\ \text{Angle} = \text{Angle} + 2\pi, \text{Angle} \in \left(-\frac{\pi}{2}, 0\right) \end{cases} \quad (5.14)$$

最后将弧度制转化为角度制即可得到以度数为单位的夹角。特别地如果 $x_1 = x_2$ ，则直接得到夹角为 270° ，无需计算。

5.7.5 多组点计算角度后求平均

对于 4 个区域，计算任何一个区域的角度，取点的过程都进行多次，沿着宽度方向按照一定的步长保证每次取到的一组点 y_1 和 y_2 的关系都是 $y_1 < y_2$ 。对于每组点都用对用区域的求角度函数进行计算，最终保存计算得到的角度。为了防止有异常值的出现，求平均值采用所有角度 0.4 和 0.6 分位数之间的点，使得计算的角度更接近比较可信的中位数。

5.8 角度与温度或压力示数的对应

角度与温度或压力的对应关系，以本论文研究的朗诗物业高层建筑的冷冻机房内的温度表和压力表为例进行对应

5.8.1 角度与温度示数的对应

通过在线量角器网站导入图片进行角度测量，取两个锚点，一个锚点对应温度示数为 50°C 的点，为表盘的 12 点钟方向；另一个锚点对应温度示数为 0°C 的点。示数 222° 表明 0 刻度与高度方向夹角为 $222-180=42^{\circ}$ 。考虑到拍摄程度的歪斜，对多张图片测试结果基本都集中在 42° 到 46° 之间，所以取 44° 为 0 刻度修正值。



图 5.16 在线量角器测量温度表 0 刻度与 12 点钟刻度角度

计算温度与角度的对应关系，只需要将 0°C 到 50°C 温度之间的角度量出，然后计算每度角对应的温度值即可。

由于表盘的安置，表盘与绝对垂线会有一定程度的倾斜夹角，如图可以看出该表盘的 12 点钟刻度与正垂线有 3° 的夹角，当 12 点钟刻度在正垂线右侧时，定义该角度为负数；反之，定义该角度为正数。因此表盘的安装修正角度为 -3° 。虽然差异不大，但各个压力表盘可能具有不同的安装修正角度，对于识别精度要求高的冷冻机房，实际使用可以针对冷冻机房的温度表盘进行一定的分类，对安

装修正角度在不同范围的温度表盘分别给予一个安装修正角度，保证识别的精确性。



图 5.17 在线量角器测量温度表 12 点钟刻度与正垂线角度

有了 0 刻度修正角度、角度与温度的对应关系和表盘的安装修正角度，计算出的角度就可转换为温度示数。

5.8.2 角度与压力示数的对应

测量方法同 5.8.1 节中温度表的测量。一个锚点对应压力示数为 0.8MPa 的点，为表盘的 12 点钟方向；另一个锚点对压力示数为 0MPa 的点。示数 130° 表明 0 刻度与高度方向夹角。考虑到拍摄程度的歪斜，对多张图片测试结果基本都集中在 128° 到 132° 之间，所以取 130° 为 0 刻度修正值。

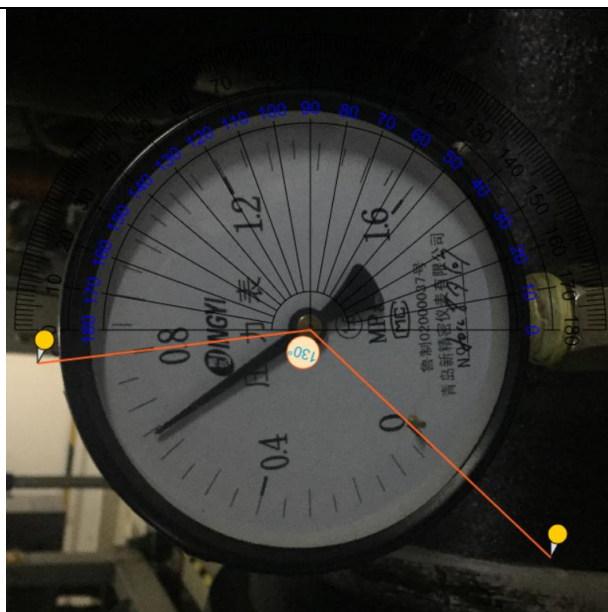


图 5.18 在线量角器测量压力表 0 刻度与 12 点钟刻度角度

压力表盘安装修正角计算同 5.8.1 节温度表盘的计算, 从下图可以看出 12 点钟刻度在正垂线左侧, 所以定义该角度为正数。因此表盘的安装修正角度为 6° 。虽然差异不大, 但各个压力表盘可能具有不同的安装修正角度, 对于识别精度要求高的冷冻机房, 实际使用可以针对冷冻机房的压力表盘进行一定的分类, 对安装修正角度在不同范围的的压力表盘分别给予一个安装修正角度, 保证识别的精确性。

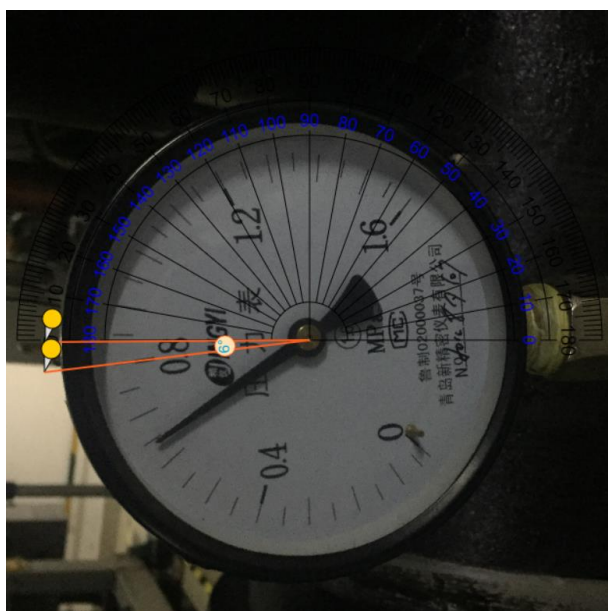


图 5.19 在线量角器测量压力表 12 点钟刻度与正垂线角度

有了 0 刻度修正角度、角度与压力的对应关系和表盘的安装修正角度，计算出的角度就可转换为温度示数。

5.9 输出结果验证

5.9.1 温度表读数结果展示

温度表识别验证结果如下，算法识别结果为 29.59°C，与温度表的实际值基本一致。



图 5.20 温度表读数测试结果

5.9.2 压力表读数结果展示

压力表识别验证结果如下，算法识别结果为 0.62MPa，与压力表的实际值基本吻合。



图 5.21 压力表读数测试结果

5.10 本章小结

本章主要实现对冷冻机房的温度表和压力表的可见光图像进行示数识别，通过对表盘图像进行一定的缩放和变换以及二值化等预处理操作后，通过两次最大联通域的提取操作后获得指针部分图像。然后对指针部分图像进行形态学操作来获得更连续的轮廓，之后对指针图像按照像素搜索进行一定的剪裁，对指针进行边缘检测后根据宽高比的大小裁剪得到针尖部分的图像，针尖部分的图像像素稳定，更方便通过数学方法计算角度。获得针尖部分的图像后，将表盘划分为4大区域，需要根据指针的位置确定其所在表盘的区域。不同的区域通过不同的数学方法计算指针与垂直方向的夹角，且计算夹角通过多个像素点计算求平均的方法提高准确度。有了计算得到的角度，再根据角度与温度和压力值的对应关系就可以得到温度表和压力表的示数，并根据该示数是否超过设定的阈值限制来判断温度表和压力表所监测的温度和压力值是否正常。

第 6 章 基于红外图像的水泵温度示数提取

上一章基于可见光图像开发了冷冻机房内温度和压力表盘的示数读取算法，本章主要利用水泵的红外图像来开发可以读取水泵最高温度，并对其进行监测的算法。

6.1 红外图像温度读数思路

6.1.1 光学文字识别工具 pytesseract

pytesseract 是 Python 接口的一个光学文字识别工具，可以读取图像中的数字，在不同的语言环境下亦可以读取不同的文字。使用 pytesseract 库应当事先安装好 Tesseract-OCR 软件，并将其添加到环境变量中。

在实践中，由于 pytesseract 识别图片中的内容时，对字体图片的尺寸，即分辨率有一定的要求。适当放大图片会增加识别精度，但是具体的放大倍数会获得不同程度的识别精度，因此确定一个适宜本章研究内容的放大倍数，也是需要考虑的一个问题。

6.1.2 红外图像温度识别流程

红外图像的温度读数主要流程是利用拍摄到的有温度条记录信息的红外图像，利用确定定位裁剪获得的最大温度示数，配合一定的图像处理操作，进行一定倍数的图像放大后，利用 Python 的光学文字识别库 pytesseract 对读数进行识别，并判断温度是否超过阈值，返回带有读数信息和判别结果的图像。所有过程通过 Python 的代码实现并主要借助 pytesseract 库和 skimage 库。

6.2 红外图像中的温度示数提取

6.2.1 红外图像的读取

红外图像的基本形式如下图所示，右侧的温度条代表了不同色彩所代表的温度高低，图像主体的色彩都可以在右侧的温度条找到对应的色彩，从而判断色彩所对应的温度值。

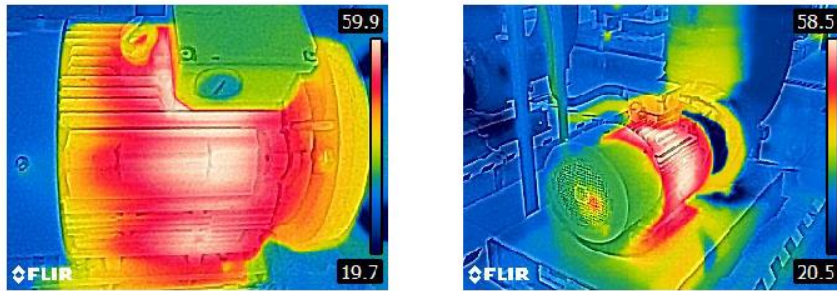


图 6.1 水泵红外图像

6.2.2 红外图像温度示数裁剪

由于使用的 FLIR 红外摄像机拍摄的红外图像的分辨率都是 240*320，然后色彩通道是三通道。所以温度条的最顶端的温度示数在图片中的位置不变，提取到这个温度后再进行识别就是水泵此刻运行的温度上限。这一最高温度在图片中的坐标位置为高度方向从 5 到 25，宽度方向从 279 到 315。以图 6.1 中右侧水泵红外图像为例，裁剪后的温度示数图像如下所示：



图 6.2 水泵最高温度示数 RGB 图片

6.3 红外图像中温度示数的预处理和识别

6.3.1 温度示数图像的二值化处理

首先先将彩色图片转化为灰度图片，灰度图像如下：



图 6.3 水泵最高温度示数灰度图片

在灰度图片的基础上，与上一章的二值化方法相同，利用 Otsu 算法求得最佳的二值化阈值后，对灰度图像进行二值化，得到如下结果：



图 6.4 水泵最高温度示数二值化图片

6.3.2 二值化图像的放大与类型转换操作

在进行识别前，需要将二值化后的图片进行放大，否则 `pytesseract` 的识别精度会大大降低。原始二值图像的类型为 `bool` 型，放大后为 `float64` 类型，但是 `float64` 类型的识别精度不高，需要将图像中像素值为 0、1 的像素放大到区间范围为 `[0,255]` 之间，再将其转换成 `uint8` 类型的图像后，才可以利用 `pytesseract` 进行识别。如何确定一个最佳的放大倍数，从而保证算法的适用性更强，是温度示数识别需要考虑的一个重要问题。

6.3.3 温度示数的识别及验证

由于不同的放大倍数会带来不同的识别精度，所以结合网格搜索法的思想，在可能的放大倍数的范围内按照一定的间隔取多个不同的放大倍数，在 100 张红外图像上进行训练，得到最大识别精度下的放大倍数为最终确定的较为合理的图像放大倍数。

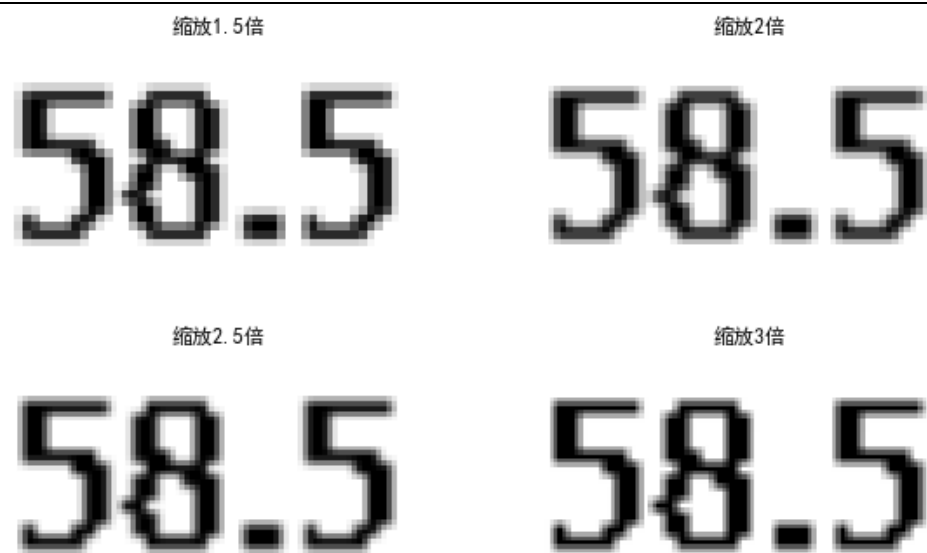


图 6.5 缩放不同倍数后的图片示数

进行最优放大倍数搜索的范围为 1 到 5 之间，每次递增 0.1，记录每个不同的放大倍数下，识别算法在不同的 100 张红外图像中识别示数的结果，将识别结果与实际温度指示对比，计算出一个放大倍数下的图像识别精度，即正确识别的结果占识别图片的总数之比。寻找识别精度最高的放大倍数，即为算法最终确定的温度示数二值图像的放大倍数。

表 6.1 放大倍数与识别准确度对应结果

放大倍数	识别准确度	放大倍数	识别准确度
1.0	0.74	3.1	0.94
1.1	0.96	3.2	0.94
1.2	1	3.3	0.82
1.3	1	3.4	0.94
1.4	1	3.5	0.62
1.5	1	3.6	0.8
1.6	1	3.7	0.88
1.7	1	3.8	0.62
1.8	1	3.9	0.68
1.9	1	4.0	0.62
2.0	1	4.1	0.58
2.1	0.98	4.2	0.6
2.2	0.96	4.3	0.58
2.3	0.98	4.4	0.58

2.4	1	4.5	0.54
2.5	0.98	4.6	0.54
2.6	0.98	4.7	0.6
2.7	0.96	4.8	0.56
2.8	0.98	4.9	0.58
2.9	1	5.0	0.58
3.0	0.9		

观察计算结果，可以看出不进行适当的图像缩放的话，识别准确度会相当的低，随着放大倍数的增大，识别准确度会显著提高。当放大倍数位于 1.2 到 2.0 之间，识别准确度达到 100%；当放大倍数位于 2.1 到 2.9 之间，识别准确度也很高。但当放大倍数大于 3.0 后，识别准确度开始显著下降，识别效果变得很差。

因此，综合不同的放大倍数下训练结果的识别准确度，本算法最终选择放大倍数 1.5 倍，为温度示数二值图像的放大修正算法。

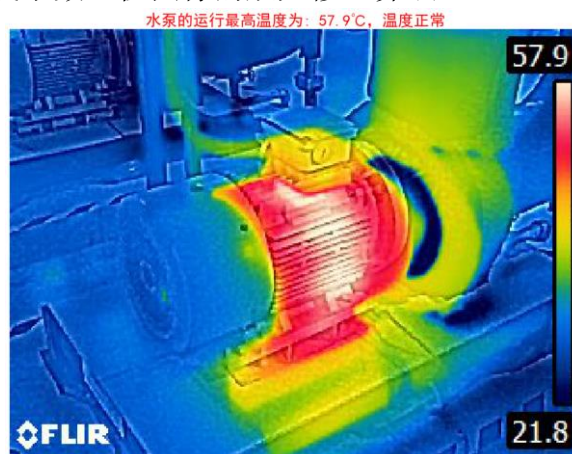


图 6.6 卧式水泵温度识别结果

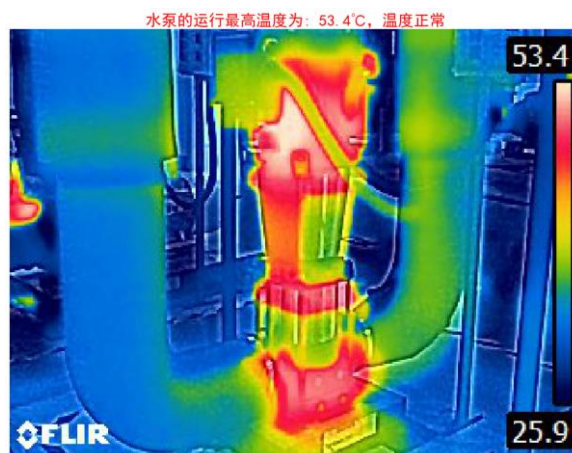


图 6.7 立式水泵温度识别结果

6.4 本章小结

本章主要实现的是利用水泵红外温度图像，识别其中的最大温度示数，判断水泵的最高温度。首先对水泵的红外图像进行裁剪得到温度示数，对温度示数的图像进行二值化处理、图像放大、图像类型转化等操作后，利用 `pytesseract` 工具进行光学文字识别读取温度示数。其中温度示数图像放大的倍数会影响 `pytesseract` 工具的识别精度，通过网格搜索改变图像放大倍数的方法在 100 张水泵红外图像上进行检验，来寻找最优的放大倍数。最后在最优的放大倍数下完成整个水泵红外图像温度示数读取的过程，并通过识别的示数是否超过设定的温度阈值来判定水泵运行温度是否正常。

第7章 结论与展望

7.1 结论

冷冻机房的故障监测和诊断是保证暖通空调水系统良好和高效运行的一个重要辅助手段。利用机器人内置故障诊断算法进行自动巡检故障监测和诊断可以有效提升冷冻机房的智能化程度,也大大地辅助了物业工作人员日常的例行巡检,是智能化技术在工业场景应用的一个发展方向。

本论文通过开发非侵入式的故障检测和诊断算法,作为机器人自动巡检的核心内置程序,完成了以下所述的内容,实现了相对应的功能:

- (1) 利用麦克风阵列采集冷冻机房不同水泵的音频设备,以机房内同一工位下的相同水泵中的一个为实例,实现了从水泵音频信号中分离和提取不同频率成分的音频部分形成多组新的信号,代表不同的水泵声音特征。在此基础上,利用音频信号的时域分析方法,建立音频信号的ARIMA时间序列拟合分析算法,并通过置信区间的阈值范围来预测未来音频信号的范围,从而判断一定运行时间内新采集的音频信号是否正常,以此判断水泵存在异常。以上音频信号分离和提取的部分以及音频信号时域分析算法部分均写成可应对多种可能情形的Python代码,方便巡检机器人调用。
- (2) 利用红外热成像仪和可见光摄像头采集冷冻机房内的立式水泵、卧式水泵、压缩机和表盘的红外图像和可见光图像。利用保留特征提取模块,修改分类模块的卷积神经网络 AlexNet 针对冷冻机房可能需要检测的上述4类图像训练一个图像分类器,图像分类器在可见光图像进行训练,对可见光图像的分类成功也即对红外图像的分类成功。训练分类器在403张包含上述4类的图片上进行训练,并在同样包含上述4类的20张图片上进行测试,其中训练集和测试集的准确率均为100%。图像分类器的目的是帮助巡检机器人判别不同的设备图像,从而为后续选择对应的图像处理算法做准备。
- (3) 针对冷冻机房的压力表和温度表的可见光图像,利用灰度图像二值化、图像合并、最大联通域提取、形态学操作和边缘检测等图像处理方法提取表盘的针尖部分。然后结合表盘的几何学位置特征和像素点位置计算角度的便宜性,对表盘进行多个区域划分,并在不同的区域利用

不同的反正切三角函数 (\arctan) 公式计算表盘的针尖部分与固定轴线的夹角, 从而换算表盘的示数。以上方法最终可以实现自动读取表盘图像指针示数的功能, 并通过可靠的设定阈值来监测示数正常与否。最后通过完整的 Python 代码形式, 为巡检机器人实现自动读取冷冻机房温度表和压力表的读数, 判别示数正常与否提供了相应算法。

- (4) 针对冷冻机房的水泵红外图像, 利用光学识别工具的 Python 接口 pytesseract 和图像切割、彩色图像灰度化、灰度图像二值化和图像缩放等图像处理方法对从红外图像提取出来的最高温度示数进行识别。为了提高识别精度, 对图像的最优放大倍数进行网格搜索。上述方法最终可以实现自动读取水泵红外图像的最高温度的功能, 并通过可靠的设定阈值来监测水泵运行温度是否异常。最后通过完整的 Python 代码形式, 为巡检机器人实现自动读取冷冻机房水泵运行最高温度, 并判别温度正常与否提供了相应算法。

以上所有内容和实现的功能共同构成了智能机器人冷冻机房自动巡检的算法, 并得到具体的验证, 可以实现相应功能的冷冻机房设备状态监测和故障诊断。

7.2 展望

本文的研究从图像和音频两大数据源开发了冷冻机房部分设备的故障监测和诊断算法, 涉及的范围较为全面, 实现的步骤也细致详尽, 并贴合于工业自动化的应用情形。

综上所述, 本文的研究虽然取得了初步的成功, 但从研究角度和工业应用角度依然有一些方面可以改进, 这些改进方面如下所述。

研究方面值得改进的地方:

- (1) 在进行水泵音频信号和故障诊断的研究中, 采用了快速傅里叶变换和有限冲激响应滤波的方法进行音频信号的特征提取。这一方法在实现中会遇到实际采集的音频信号分解较为含糊, 界限不明确的问题, 在大致提取音频信号不同成分上有较好的适用性, 但缺乏针对性地具体解释成分特征, 不能深层次挖掘故障原因。因此如果想更有解释性地进行故障原因诊断时, 应该寻找一个更为明确的特征因子。
- (2) 在进行时间序列拟合各成分音频信号时, 对中高频特征的拟合很容易出现均一性, 分析发现是音频信号并不是时间强相关的序列, 从时间序列的角度看, 随机波动的成分较大。一方面可以体现出时间序列应用的普遍适应性较强, 但在出现异常波动时不能明确定位异常波动的

原因。另一方面值得继续研究音频信号为什么会出现较多的不明确随机波动成分。

工业应用角度值得改进的地方：

- (1) 在训练图像分类器时，没有分开压力表和温度表。而实际运行表盘示数读取算法时，由于量程的不同，针对压力表和温度表采用的是不同的校准算法。如果能直接通过图像分类器区分压力表和温度表，会使巡检机器人工作更为便利。对于论文研究的冷冻机房，发现温度表和压力表所安装的位置和安装方式的确有一定的差别。温度表倾向嵌入包装有保温层的管道内，其背景以保温层为主；而压力表倾向于通过金属杆伸出一定的高度进行安装，其背景以冷机和冷机间的通道为主。因此可以尝试改进图像分类器，针对压力表和温度表分别进行分类识别。
- (2) 表盘识别算法的核心是完整的提取较为清晰的表盘指针部分，其中最重要的两步操作是两次最大联通域的提取，且首次最大联通域的提取比较依赖图片拍摄的质量。拍摄的图片表盘占据的空间大，且背景颜色以深色为主，可以清晰地提取出表盘部分。算法在本文研究的冷冻机房中运行效果较好，如果扩大工业应用，需要在更多的机房测试，同时总结出应对不同背景环境冷冻机房的较为普适性的方法，从而开发更稳定的算法。

致谢

伴随着硕士毕业论文的截稿，我的两年半的同济硕士生活即将划上休止符。算上昔日在同济的本科时光，我与同济已经携手走过了六年半的时光。“逝者如斯夫”，驻足回首，逝去时光像雕刻家手中的刻刀一样，在我的人生塑像中镌刻了平淡而又浓重的一笔，留下了深浅交织的划痕。这些划痕折射出了成长中的悲欢，让人敬畏又叹惋：敬畏于像在溪流中淘到精致的鹅卵石一样，在时光中把自己打磨得坚韧又能展露锋芒；叹惋于像在一棵树上生长的枝干，时而交叉时而分离一样，时光中遇到的风景与伙伴也沧海桑田，或时常在眼前，或烙印在脑海。

完成这篇论文，我要感谢陪伴我度过两年半硕士生活的 A434 课题组大家庭。首先要感谢我的导师——许鹏教授。早在本科三年级的空气调节专业课时，我就对许老师儒雅的气质和风范印象深刻；也记得因为课上一次对空调负荷计算方法的小结文章而受到认可。最后获得保研资格后，我也毫不犹豫地选择了许老师作为我的导师，也很感激许老师接受了我。在之后的硕士生活中，我也很感激许老师的指导与激励，让我真正觉得我渐渐变得善于思考，学会分析，知道如何研究与探索，实践与创新。

其次，要感谢两年半硕士生活中 A434 课题组大家庭的成员们。感谢在我研一刚入学就给我很多指导建议的吴云涛师兄；感谢虽然科研紧张但也教我很多东西的冯帆师兄；感谢像大姐姐一样亲和的褚祎祎师姐的帮助。虽然只有一年的时光，但这一年他们的帮助让我更快地适应从本科转变为硕士的学习生活。感谢罗为师姐对我的关心，一起在图像处理上学习进步；感谢杨志伟师兄和我认真的交流，让学习生活也充满欢笑。与他们在一起将近两年的时光，留下很多快乐的记忆。感谢陈永保师兄与我一起做项目，欢快地练习英语口语；感谢顾洁帆师姐在就业和科研方面的建议；感谢沙华晶师姐帮助我解答难题，特别是 Energyplus 方面的问题。感谢我的同门陈喆，在电子设备上提供各种技术支持，同时也是我整个硕士生活交流学术最多的人，我拓宽了眼界，也学习到很多新鲜的知识。感谢王鸿鑫师弟，每逢求教，都能亲和耐心地讲解；感谢戴明坤师弟，和我交流人生；感谢李泓名师弟，在科研方面我向他学习了很多技术，帮助我进行论文算法的机器人集成。同时也感谢贴心的肖彤师妹、郭明月师妹和认真向我学习的何睿凯师弟和一起健身的陈志森师弟，还有和我讨论 Pytorch 的贾乾然师弟。正是这样的 A434 大家庭给我的学习生活带来了欣慰与温馨。此外，也要感谢在我采集音频和图像数据时，给我提供帮助的潘师傅。

聚散离合终有时，历来烟雨不由人。那些在我成长路上给予我温柔的人们，愿你们也能一直被世界温柔以待。告别学生时光的时日，愿自己对待学习可以求知若饥，虚心若愚；看待事物可以宠辱不惊，去留无意；保持品性可以怀瑾握瑜，心若芷萱。

本课题受 2017 年国家重点研发计划项目《多能互补集成优化的分布式能源系统示范》项目 2017YFB0903404 支持。

2020 年 3 月

参考文献

- [1] Yu Y, Woradetchjumroen D, Yu D. A review of fault detection and diagnosis methodologies on air-handling units[J]. *Energy and Buildings*. 2014, 82: 550-562.
- [2] Yan R, Ma Z, Zhao Y, et al. A decision tree based data-driven diagnostic strategy for air handling units[J]. *Energy and Buildings*. 2016, 133: 37-45.
- [3] Wang H, Chen Y, Chan CWH, et al. A robust fault detection and diagnosis strategy for pressure-independent VAV terminals of real office buildings[J]. *Energy and Buildings*. 2011, 43(7): 1774-1783.
- [4] Mulumba T, Afshari A, Yan K, et al. Robust model-based fault diagnosis for air handling units[J]. *Energy and Buildings*. 2015, 86: 698-707.
- [5] Fan C, Xiao F, Yan C. A framework for knowledge discovery in massive building automation data and its application in building diagnostics[J]. *Automation in Construction*. 2015, 50: 81-90.
- [6] Bagavathiappan S, Lahiri BB, Saravanan T, et al. Infrared thermography for condition monitoring - A review[J]. *Infrared Physics & Technology*. 2013, 60: 35-55.
- [7] Huda ASN, Taib S. Application of infrared thermography for predictive/preventive maintenance of thermal defect in electrical equipment[J]. *Applied Thermal Engineering*. 2013, 61(2): 220-227.
- [8] 丁邺, 付冬梅, 李志平等. 变压器故障的红外诊断方法与软件设计[J]. *化工自动化及仪表*. 2016, 43(08): 818-821.
- [9] 孙怡, 吴邦树, 刘仪卜等. 基于红外图像处理的电气设备故障诊断方法[J]. *机电工程技术*. 2016, 45(06): 59-62.
- [10] 井金剑, 翟春艳, 李书臣等. 基于红外图像识别的电气设备故障诊断[J]. *电子设计工程*. 2014, 22(12): 171-173.
- [11] 王淼, 杜伟, 孙鸿博等. 基于红外图像识别的输电线路故障诊断方法[J]. *红外技术*. 2017, 39(04): 383-386.
- [12] García Márquez FP, Tobias AM, Pinar Pérez JM, et al. Condition monitoring of wind turbines: Techniques and methods[J]. *Renewable Energy*. 2012, 46: 169-178.
- [13] Glowacz A, Glowacz Z. Diagnosis of the three-phase induction motor using thermal imaging[J]. *Infrared Physics & Technology*. 2017, 81: 7-16.
- [14] Nasiri A, Taheri-Garavand A, Omid M, et al. Intelligent fault diagnosis of cooling radiator based on deep learning analysis of infrared thermal images[J]. *Applied Thermal Engineering*. 2019, 163: 114410.
- [15] Zidani F, El Hachemi Benbouzid M, Diallo D, et al. Induction motor stator faults diagnosis

- by a current concordia pattern-based fuzzy decision system[J]. *IEEE Transactions on Energy Conversion*. 2003, 18(4): 469-475.
- [16] Choi S, Pazouki E, Baek J, et al. Iterative Condition Monitoring and Fault Diagnosis Scheme of Electric Motor for Harsh Industrial Application[J]. *IEEE Transactions on Industrial Electronics*. 2015, 62(3): 1760-1769.
- [17] Guesmi H, Ben Salem S, Bacha K. Smart wireless sensor networks for online faults diagnosis in induction machine[J]. *Computers & Electrical Engineering*. 2015, 41: 226-239.
- [18] Bandyopadhyay I, Purkait P, Koley C. A combined image processing and Nearest Neighbor Algorithm tool for classification of incipient faults in induction motor drives[J]. *Computers & Electrical Engineering*. 2016, 54: 296-312.
- [19] Myrans J, Everson R, Kapelan Z. Automated detection of faults in sewers using CCTV image sequences[J]. *Automation in Construction*. 2018, 95: 64-71.
- [20] Fuentes A, Yoon S, Kim S, et al. A Robust Deep-Learning-Based Detector for Real-Time Tomato Plant Diseases and Pests Recognition[J]. *Sensors*. 2017, 17(9): 2022.
- [21] Wang G, Sun Y, Wang J. Automatic Image-Based Plant Disease Severity Estimation Using Deep Learning[J]. *Computational Intelligence and Neuroscience*. 2017, 2017: 1-8.
- [22] . Segmentation Method for Induction Motor Fault Diagnosis[J].
- [23] Glowacz A, Glowacz Z. Diagnosis of the three-phase induction motor using thermal imaging[J]. *Infrared Physics & Technology*. 2017, 81: 7-16.
- [24] Vilela R, Metrolho JC, Cardoso JC. Machine and industrial monitorization system by analysis of acoustic signatures[M]: *IEEE*. 2004: 277-279.
- [25] Lei Y, Zuo MJ, He Z, et al. A multidimensional hybrid intelligent method for gear fault diagnosis[J]. *Expert Systems with Applications*. 2010, 37(2): 1419-1430.
- [26] Kankar PK, Sharma SC, Harsha SP. Fault diagnosis of ball bearings using machine learning methods[J]. *Expert Systems with Applications*. 2011, 38(3): 1876-1886.
- [27] Sakthivel NR, Sugumaran V, Nair BB. Comparison of decision tree-fuzzy and rough set-fuzzy methods for fault categorization of mono-block centrifugal pump[J]. *Mechanical Systems and Signal Processing*. 2010, 24(6): 1887-1906.
- [28] Pan Y, Chen J, Guo L. Robust bearing performance degradation assessment method based on improved wavelet packet - support vector data description[J]. *Mechanical Systems and Signal Processing*. 2009, 23(3): 669-681.
- [29] Wang X, Makis V, Yang M. A wavelet approach to fault diagnosis of a gearbox under varying load conditions[J]. *Journal of Sound and Vibration*. 2010, 329(9): 1570-1585.
- [30] Yang M, Makis V. ARX model-based gearbox fault detection and localization under varying load conditions[J]. *Journal of Sound and Vibration*. 2010, 329(24): 5209-5221.
- [31] Shao Y, Mechefske CK. Gearbox vibration monitoring using extended Kalman filters and hypothesis tests[J]. *Journal of Sound and Vibration*. 2009, 325(3): 629-648.

- [32] Wang F, Mechefske CK. Adaptive modelling of transient vibration signals[J]. *Mechanical Systems and Signal Processing*. 2006, 20(4): 825-842.
- [33] Wenyi Wang AKW. Autoregressive Model-Based Gear Fault Diagnosis[J]. *Journal of Vibration and Acoustics*. 2002, 124.
- [34] Lei Y, He Z, Zi Y. A new approach to intelligent fault diagnosis of rotating machinery[J]. *Expert Systems with Applications*. 2008, 35(4): 1593-1600.
- [35] Lei Y, He Z, Zi Y, et al. New clustering algorithm-based fault diagnosis using compensation distance evaluation technique[J]. *Mechanical Systems and Signal Processing*. 2008, 22(2): 419-435.
- [36] Tandon N, Choudhury A. A review of vibration and acoustic measurement methods for the detection of defects in rolling element bearings[J]. *Tribology International*. 1999, 32(8): 469-480.
- [37] Wang W, Kanneg D. An integrated classifier for gear system monitoring[J]. *Mechanical Systems and Signal Processing*. 2009, 23(4): 1298-1312.
- [38] Li W, Mechefske CK. Detection of Induction Motor Faults: A Comparison of Stator Current, Vibration and Acoustic Methods[J]. *Journal of Vibration and Control*. 2016, 12(2): 165-188.
- [39] Koo IS, Kim WW. The development of reactor coolant pump vibration monitoring and a diagnostic system in the nuclear power plant[J]. *ISA Transactions*. 2000, 39(3): 309-316.
- [40] Lei Y, Zuo MJ, He Z, et al. A multidimensional hybrid intelligent method for gear fault diagnosis[J]. *Expert Systems with Applications*. 2010, 37(2): 1419-1430.
- [41] Lei Y, He Z, Zi Y. Application of the EEMD method to rotor fault diagnosis of rotating machinery[J]. *Mechanical Systems and Signal Processing*. 2009, 23(4): 1327-1338.
- [42] Peng ZK, Tse PW, Chu FL. A comparison study of improved Hilbert - Huang transform and wavelet transform: Application to fault diagnosis for rolling bearing[J]. *Mechanical Systems and Signal Processing*. 2005, 19(5): 974-988.
- [43] Rafiee J, Rafiee MA, Tse PW. Application of mother wavelet functions for automatic gear and bearing fault diagnosis[J]. *Expert Systems with Applications*. 2010, 37(6): 4568-4579.
- [44] Baydar N, Ball A. DETECTION OF GEAR FAILURES VIA VIBRATION AND ACOUSTIC SIGNALS USING WAVELET TRANSFORM[J]. *Mechanical Systems and Signal Processing*. 2003, 17(4): 787-804.
- [45] Zhong ZM, Chen J, Zhong P, et al. Application of the blind source separation method to feature extraction of machine sound signals[J]. *The International Journal of Advanced Manufacturing Technology*. 2006, 28(9-10): 855-862.
- [46] Li Li LQ. Machine diagnosis with independent component analysis and envelope analysis[C]. Bangkok. 2000.
- [47] 刘朦月. 基于振动信号的电机轴承故障诊断方法研究[D]: 哈尔滨工业大学, 2016.
- [48] 李焯. 基于数据驱动的电机轴承故障诊断方法研究[D]: 沈阳理工大学, 2017.

- [49] Glowacz A, Glowacz W, Glowacz Z, et al. Early fault diagnosis of bearing and stator faults of the single-phase induction motor using acoustic signals[J]. *Measurement*. 2018, 113: 1-9.
- [50] Glowacz A. Fault diagnosis of single-phase induction motor based on acoustic signals[J]. *Mechanical Systems and Signal Processing*. 2019, 117: 65-80.
- [51] Delgado-Arredondo PA, Morinigo-Sotelo D, Osornio-Rios RA, et al. Methodology for fault detection in induction motors via sound and vibration signals[J]. *Mechanical Systems and Signal Processing*. 2017, 83: 568-589.
- [52] Zhang D, Entezami M, Stewart E, et al. Adaptive fault feature extraction from wayside acoustic signals from train bearings[J]. *Journal of Sound and Vibration*. 2018, 425: 221-238.
- [53] 杜世斌. 基于音频特征的电气设备故障监测算法研究[D]: 山东大学, 2014.
- [54] 杨毅明. 数字信号处理[M]: 机械工业出版社, 2017.
- [55] 王燕. 应用时间序列分析[M]: 中国人民大学出版社, 2012.
- [56] Yu LDAD. Deep Learning: Methods and Applications[J]. *Foundations and Trends® in Signal Processing*. 2014, 7: 197-387.
- [57] 陈云. 深度学习框架Pytorch入门与实践[M]: 电子工业出版社, 2018.
- [58] Alex Krizhevsky ISGH. ImageNet Classification with Deep Convolutional Neural Networks[J]. *Advances in neural information processing systems*. 2012, 25(2).

附录 A 部分代码节选

第一部分：卷积神经网络训练与预测（节选）

```
import torch
import torch.nn as nn
import torch.optim as optim
from torch.optim import lr_scheduler
from torch.autograd import Variable
from torchvision import datasets, models, transforms
import numpy as np
import time
import os

#建立模型
model_alexnet = models.alexnet(pretrained=True)
for para in model_alexnet.parameters():
    #循环 16 次
    para.requires_grad = False

model_alexnet.classifier = nn.Sequential(
    nn.Dropout(),
    nn.Linear(256 * 6 * 6, 4096),
    nn.ReLU(inplace=True),
    nn.Dropout(),
    nn.Linear(4096, 4096),
    nn.ReLU(inplace=True),
    nn.Linear(4096, 4)
)

if use_gpu:
    model_ft = alexnet.cuda()
criterion = nn.CrossEntropyLoss()
```

```
optimizer_ft = optim.SGD(model_ft.parameters(), lr=0.005, momentum=0.9,
weight_decay = 0.0001)
exp_lr_scheduler = lr_scheduler.StepLR(optimizer_ft, step_size=5, gamma=0.1)
model_ft = train_model(model=model_ft,
                        criterion=criterion,
                        optimizer=optimizer_ft,
                        scheduler=exp_lr_scheduler,
                        num_epochs=20)

#图像预处理
t_test = transforms.Compose([
    transforms.Resize(256),
    transforms.CenterCrop(224),
    transforms.ToTensor(),
    transforms.Normalize
    ([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
])

img_transform_test = t_test(img) #尺寸 torch.Size([3, 224, 224])
img_transform_new_PIL = img_transform_test.unsqueeze(0)

#图像识别
x_out = model_ft(img_transform_new_PIL.cuda())
_, preds = torch.max(x_out.data, 1)

label_dic = image_datasets['train'].class_to_idx
label_predict_num = preds.cpu().numpy()[0]
label_predict_text=list(label_dic.keys())[list(label_dic.values()).index(label_pred
ict_num)]

#绘图展示识别结果
font1={'fontsize':24,
      'fontweight':'bold',
      'verticalalignment':'baseline',
```

```
'horizontalalignment':'center'}
plt.rcParams['font.sans-serif']=['SimHei']
plt.rcParams['axes.unicode_minus']=False

fig = plt.figure(figsize=(18,12))
ax1 = fig.add_subplot(121)
ax1.imshow(img)
ax1.set_title('图片类别识别结果为: %s' % label_predict_text, fontdict = font1,
color='r')
ax1.axis('off')
```

第二部分：表盘识别函数的预处理和区域角度计算函数

```
from matplotlib import pyplot as plt
from skimage import filters
from scipy import arctan
from skimage import color
from skimage import transform
import math

#灰度图片二值化
def binarize_image(image,threshold):
    #image 图像为 float 类型的灰度图像
    image_binarize = image>threshold
    return image_binarize

def merge_image(image,show='false'):
    # 利用 otsu 法自动确定二值化阈值
    R = image[:, :, 0]
    G = image[:, :, 1]
    B = image[:, :, 2]
    otsu_r = filters.threshold_otsu(R)
    otsu_g = filters.threshold_otsu(G)
```

```

otsu_b = filters.threshold_otsu(B)
#对 float64 类型的图片进行二值化, 并转成
R_bi = binarize_image(R,threshold=otsu_r)
G_bi = binarize_image(G,threshold=otsu_g)
B_bi = binarize_image(B,threshold=otsu_b)
img_new = R_bi&G_bi&B_bi
if show=='true':
    fig,axes = plt.subplots(2,2,figsize=(8,12))
    ax0,ax1,ax2,ax3 = axes.ravel()

    ax0.imshow(R_bi,cmap='gray')
    ax0.set_title('R_bi')

    ax1.imshow(G_bi,cmap='gray')
    ax1.set_title('G_bi')

    ax2.imshow(B_bi,cmap='gray')
    ax2.set_title('B_bi')

    ax3.imshow(img_new,cmap='gray')
    ax3.set_title('MERGE')

    for ax in axes.ravel():
        ax.axis('off')
return img_new

# 将彩色图片转成灰度再根据 otsu 阈值转成二值
def color_gray_bi(image,show='false'):
    img_gray=color.rgb2gray(image) #彩色图片转成灰度图片
    otsu_th = filters.threshold_otsu(img_gray)
    img_bi = binarize_image(img_gray,otsu_th) #灰度图片转成二值图片

    if show == 'true':
        fig = plt.figure(figsize=(10,5))

```

```
ax = fig.add_subplot(111)
ax.imshow(img_bi,cmap='gray')
ax.axis('off')
return img_bi
```

#找中点函数

```
def find_median(y,bw):
    x1=None
    x2=None
    h = bw.shape[0]
    w = bw.shape[1]
    if y<0 or y>=w:
        return '坐标输入超出允许范围'
    for i in range(h):
        if bw[i,y]==1:
            x1=i
            break
    for i in range(h):
        if bw[i,y]==1 and bw[i+1,y]==0:
            x2=i
            break
    x = (x1+x2)//2
    return x
```

```
def recog_180lessmore_zone14(bw):
```

```
    h = bw.shape[0]
    w = bw.shape[1]
    w1 = round(w/4)
    w2 = round(w*3/4)
    pixel_w1_count=0
    pixel_w2_count=0
    for i in range(h):
        if bw[i,w1]==1:
            pixel_w1_count+=1
```

```

        if bw[i,w2]==1:
            pixel_w2_count+=1
    if pixel_w1_count<pixel_w2_count:
        return '区域 1 指针'
    else:
        return '区域 4 指针'

def recog_180lessmore_zone23(bw):
    bw_ro = transform.rotate(bw,90,resize=True)
    bw_ro = bw_ro.astype('bool')
    w = bw_ro.shape[1]
    w1 = round(w*1/3)
    w2 = round(w*2/3)
    pixel_w1_height=find_median(w1,bw_ro)
    pixel_w2_height=find_median(w2,bw_ro)
    if pixel_w1_height>pixel_w2_height:
        return '区域 2 指针'
    elif pixel_w1_height<pixel_w2_height:
        return '区域 3 指针'
    else:
        return '180 度指针'

#取 2 个点作为计算角度的基准点，设定计算角度函数
def find_angle_zone1(y1,y2,bw):
    if y1>=y2:
        return '输入的第一个点的坐标需要小于第二个点的坐标'
    x1 = find_median(y1,bw)
    x2 = find_median(y2,bw)
    try:
        angle = arctan((y2-y1)/(x1-x2))
        angle = 180*angle/math.pi
        if angle<0:
            angle += 180
    except:

```



```

        angle = 90
    return angle

def find_angle_zone2(y1,y2,bw):
    if y1>=y2:
        return '输入的第一个点的坐标需要小于第二个点的坐标'
    bw_ro = transform.rotate(bw,90,resize=True)
    bw_ro = bw_ro.astype('bool')
    x1 = find_median(y1,bw_ro)
    x2 = find_median(y2,bw_ro)
    try:
        angle = arctan((y2-y1)/(x1-x2))
        angle = 180*angle/math.pi+90 #把小于 90 度的角加 90 变到小于
180 度的角
    except:
        angle = 180
    return angle

def find_angle_zone3(y1,y2,bw):
    if y1>=y2:
        return '输入的第一个点的坐标需要小于第二个点的坐标'
    bw_ro = transform.rotate(bw,90,resize=True)
    bw_ro = bw_ro.astype('bool')
    x1 = find_median(y1,bw_ro)
    x2 = find_median(y2,bw_ro)
    try:
        angle = arctan((y2-y1)/(x1-x2))
        angle = 180*angle/math.pi+180+90 #把大于 90 度的角加 180 度
(从负变回自身) 再加 90 度变到大于 180 度的角
    except:
        angle=180
    return angle

def find_angle_zone4(y1,y2,bw):
    if y1>=y2:

```

```

    return '输入的的第一个点的坐标需要小于第二个点的坐标'
x1 = find_median(y1,bw)
x2 = find_median(y2,bw)
try:
    angle = arctan((y2-y1)/(x1-x2))
    angle = 180*angle/math.pi
    angle += 180 #把 0-90 度变到 180 到 270 度
    if angle<0:
        angle += 360 #把-90-0 度变到 270 到 360 度
except:
    angle = 270
return angle

```

第三部分：表盘示数识别函数（温度表为例）

```

def angle_calculation_temperature(image,filenameoutput,
ifsave=True,jpgname='TemperatureGaugeResult.jpg'):
    img_bi = gsf.color_gray_bi(image)
    #使用 regionprops 进行连通域提取(找到集合)
    clear_bi = clear_border(img_bi,show='false')
    label_image,num = label(clear_bi,return_num=True,connectivity=2)

    area_lis=[]
    region_result = regionprops(label_image)
    for i in range(num):
        area_lis.append(region_result[i].area)

    index_maxarea = area_lis.index(max(area_lis))

    label_image_c = label_image.copy()
    label_image_c[label_image_c!=(index_maxarea+1)]=0
    label_image_c[label_image_c==(index_maxarea+1)]=1
    label_image_c = label_image_c.astype('bool')
    label_image_complement = (1-label_image_c).astype('bool')

```

```

#使用 regionprops 进行连通域提取(对补集图像操作)
label_image2,num2 =
label(label_image_complement,return_num=True,connectivity=2)

area_lis2=[]
region_result2 = regionprops(label_image2)
for i in range(num2):
    area_lis2.append(region_result2[i].area)

sort_li = area_lis2.copy()
sort_li.sort(reverse=True)
second_max = sort_li[1]
index_secondmaxarea = area_lis2.index(second_max)

label_image2_c = label_image2.copy()

label_image2_c[label_image2_c!=(index_secondmaxarea+1)]=0
label_image2_c[label_image2_c==(index_secondmaxarea+1)]=1
label_image2_c = label_image2_c.astype('bool')
#对图形进行形态学操作（用算子处理集合）
img_open = binary_opening(label_image2_c, disk(3)) #求形态学开图像，
img_closing = binary_closing(label_image2_c, disk(3)) #求形态学闭图像
#对图像关键部分进行裁剪，目的是提取指针
bw = img_closing
h = bw.shape[0];w = bw.shape[1]
#本次循环决定 needle 的上下边界
for i in range(h):
    for j in range(w):
        if bw[i,j]==1:
            m1=i
            #n1=j
            break
        if bw[i,j]==1:
            break

```

```

for i in range(h-1,-1,-1):
    for j in range(w):
        if bw[i,j]==1:
            m2=i
            #n2=j
            break
    if bw[i,j]==1:
        break

bw1 = bw[m1:m2,:]
for i in range(w):
    for j in range(bw1.shape[0]):
        if bw1[j,i] == 1:
            n1 = i
            break
    if bw1[j,i]==1:
        break

for i in range(w-1,-1,-1):
    for j in range(bw1.shape[0]):
        if bw1[j,i] == 1:
            n2 = i
            break
    if bw1[j,i]==1:
        break

bw11 = bw1[:,n1:n2]
h1 = bw11.shape[0];w1 = bw11.shape[1]
if h1>1.2*w1:
    bw2 = bw11[0:round(bw11.shape[0]/2.5),:] #高度方向裁剪
round(bw11.shape[0]/2.5)
else:
    bw2 = bw11[:,0:round(bw11.shape[1]/2.5)] #宽度方向裁剪

```

```
#判断指针属于哪个区域
needle_zone = ""
if h1>1.2*w1:
    #属于区域 2 和区域 3
    re = gsf.recog_180lessmore_zone23(bw2)
    if re == '区域 2 指针':
        needle_zone = '区域 2'
    elif re == '区域 3 指针':
        needle_zone = '区域 3'
    else:
        needle_zone = '180 度'
else:
    #属于区域 1 和区域 4
    re = gsf.recog_180lessmore_zone14(bw2)
    if re == '区域 1 指针':
        needle_zone = '区域 1'
    else:
        needle_zone = '区域 4'
#计算不同选择点的角度值
h2 = bw2.shape[0];w2 = bw2.shape[1]
dic_angle = {}
ave = 0
if needle_zone == '区域 1':
    for i in range(round(1*w2/4),round(3*w2/4),5):
        for j in range(i,round(3*w2/4),5):
            if i!=j:
                an = gsf.find_angle_zone1(i,j,bw2)
                dic_angle.update({'宽度方向坐标'+str(i)+'-'+str(j):an})
elif needle_zone == '区域 4':
    for i in range(round(1*w2/4),round(3*w2/4),5):
        for j in range(i,round(3*w2/4),5):
            if i!=j:
                an = gsf.find_angle_zone4(i,j,bw2)
```

```

        dic_angle.update({'宽度方向坐标'+str(i)+'+'+str(j):an})
elif needle_zone == '区域 2':
    for i in range(round(1*h2/4),round(3*h2/4),5):
        for j in range(i,round(3*h2/4),5):
            if i!=j:
                an = gsf.find_angle_zone2(i,j,bw2)
                dic_angle.update({'高度方向坐标'+str(i)+'+'+str(j):an})
elif needle_zone == '区域 3':
    for i in range(round(1*h2/4),round(3*h2/4),5):
        for j in range(i,round(3*h2/4),5):
            if i!=j:
                an = gsf.find_angle_zone3(i,j,bw2)
                dic_angle.update({'高度方向坐标'+str(i)+'+'+str(j):an})
else:
    ave = 90

if ave==0:
    df_angle = pd.DataFrame([dic_angle]).T
    ave = df_angle[(df_angle[0]>df_angle.quantile(0.4)[0]) &
(df_angle[0]<df_angle.quantile(0.6)[0])][0].mean()
    #将角度值转化成指针示数,
    T_angle = 50/136 #136 度对应 50℃
    calibra = -3*T_angle #5 度校准
    T_point = (ave-44)*T_angle+calibra
    font1={'fontsize':24,
        'fontweight':'bold',
        'verticalalignment':'baseline',
        'horizontalalignment':'center'}
    plt.rcParams['font.sans-serif']=['SimHei']
    plt.rcParams['axes.unicode_minus']=False
    #温度输出图像并显示指针示数
    if T_point>50.0:
        fault_detector = '示数异常'
    else:

```

```

        fault_detector = '示数正常'
    fig = plt.figure(figsize=(18,12))
    ax = fig.add_subplot(111)
    ax.imshow(image)
    ax.set_title('温度表盘的读数为: %.2f°C, %s' %
(T_point,fault_detector),fontdict = font1,color='r')
    if ifsave:
        plt.savefig(filenameoutput+'\\'+ jpgname)

```

第四部分 水泵红外图片温度示数识别

```

import pytesseract
import os
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
from skimage import io,transform
from skimage.filters import threshold_otsu
from skimage import color

#二值化图像的函数
def binarize_image(image,threshold):
    #image 图像为 float 类型的灰度图像
    image_binarize = image>threshold
    return image_binarize

# 将 rgb 裁剪，转灰度，转二值，取补集
def crop_gray_binary_complement(image_f):
    img_rgb = io.imread(image_f)
    img_rgb_num = img_rgb[5:25,279:-5]
    img_gray=color.rgb2gray(img_rgb_num) #彩色图片转成灰度图片，
float64
    otsu_th = threshold_otsu(img_gray)
    img_bi = binarize_image(img_gray,otsu_th) #灰度图片转成二值图片，
bool

```

```
img_bi_com = 255-255*img_bi.astype('uint8')

return img_rgb,img_bi_com

def crop_gray_binary_complement_batch(image_f):
    img_rgb = io.imread(image_f)
    img_rgb_num = img_rgb[5:25,279:-5]
    img_gray=color.rgb2gray(img_rgb_num) #彩色图片转成灰度图片,
float64
    otsu_th = threshold_otsu(img_gray)
    img_bi = binarize_image(img_gray,otsu_th) #灰度图片转成二值图片,
bool
    img_bi_com = 255-255*img_bi.astype('uint8')

    return img_bi_com

#数字文字识别
def number_recog(image):
    text = pytesseract.image_to_string(image)
    try:
        int(text[0]) & int(text[-1])
    except:
        text = text[1:-1]
    try:
        print('最高温度值为:{t}'.format(t = float(text)))
        return float(text)
    except:
        print('识别结果不是数字')
        return 0

#定义温度检测函数
def temperature_detect(image_f,scale_factor,temp_alarm_thres=75.0):
    img_rgb,im_bi_com = crop_gray_binary_complement(image_f) #
```



```

im_sc = transform.rescale(im_bi_com, scale_factor, mode='reflect')
im_sc = (im_sc*255).astype('uint8')
result = number_recog(im_sc)
if result>temp_alarm_thres:
    fault_detector = '温度异常'
else:
    fault_detector = '温度正常'
font1={'fontsize':24,
      'fontweight':'bold',
      'verticalalignment':'baseline',
      'horizontalalignment':'center'}
plt.rcParams['font.sans-serif']=['SimHei']
plt.rcParams['axes.unicode_minus'] =False

fig = plt.figure(figsize=(16,12))
ax = fig.add_subplot(111)
ax.imshow(img_rgb)
ax.set_title('水泵的运行最高温度为: %.1f°C, %s' %
(result,fault_detector),fontdict = font1,color='r')
ax.axis('off')

return result

```

个人简历、在读期间发表的学术论文与研究成果

个人简历:

陈智博, 男, 1994 年 8 月生。

2017 年 7 月毕业于同济大学建筑环境与能源应用工程(暖通)专业 获学士学位。

2017 年 9 月入同济大学动力工程专业读专业硕士研究生。

已发表论文:

[1]陈智博, 沙华晶, 许鹏, 奚培峰. 中国公共建筑的建筑典型模型建立[J]. 建筑节能, 2020(02):97-99+106.

参与学术会议:

2018 年同济大学-冈山大学暑期夏令营及学术会议 冈山市 冈山县 日本
2018.07.22-2018.07.29