



同濟大學

TONGJI UNIVERSITY

硕士学位论文

(专业学位)

基于深度学习的装配式混凝土结构 缺陷识别研究

资助项目：国家重点研发计划子课题

(2016YFC0701800)

姓 名：赵涛涛

学 号：1732344

所在院系：土木工程学院

学科门类：工程

学科专业：建筑与土木工程

指导教师：唐和生

二〇二〇年六月



同濟大學
TONGJI UNIVERSITY

A dissertation submitted to
Tongji University in conformity with the requirements for
the degree of Master of Engineering

**Research on defect detection of prefabricated
concrete structure based on deep learning**

Project Supported by the National Key R&D Program
(2016YFC0701800)

Candidate: Zhao Taotao

Student Number: 1732344

School/Department: College of Civil Engineering

Discipline: Engineering

Major: Architecture and Civil Engineering

Supervisor: Tang Hesheng

June, 2020

基于深度学习的装配式混凝土结构缺陷识别研究

赵涛涛

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；学校有权保存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：

年 月

同济大学学位论文原创性声明

本人郑重声明：所呈交的学位论文，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：

年 月 日

摘要

随着我国社会和经济实力的不断进步,人们对于建筑和居住环境提出了更高的要求,传统的粗放式建筑模式已不能顺应时代的需求。为此国家大力推动装配式建筑的发展,力图推动传统建筑模式向新型工业化建筑模式转型。装配式建筑采用工厂预制构件和现场组装的施工方式,竖向构件间多为灌浆套筒连接。由于施工工序复杂等原因,装配式建筑中灌浆套筒连接的施工容易产生质量缺陷,这必然会对结构的安全性产生一定的影响。局部动力响应信号蕴含着非常丰富的缺陷信息,然而传统的数据分析方法难以从这些局部响应中提取有效的缺陷信息,深度学习凭借其强大的数据挖掘、特征提取和分类识别能力,成为数据挖掘和缺陷识别的利器,也为上述问题提供了一个理想的解决方案。因此,本文将深度学习算法应用到装配式混凝土结构缺陷识别中,建立基于深度学习的装配式混凝土结构缺陷识别方法。该方法利用深度学习对装配式混凝土结构的局部响应信号进行逐层特征变换,最终提取出反映结构响应缺陷信息的概率特征,并结合定性分析和定量分析实现装配式混凝土结构的灌浆套筒缺陷识别。利用装配式框架结构和装配式剪力墙动力检测试验,系统地验证了该方法的可行性、有效性和准确性。主要研究工作如下:

(1) 梳理深度学习技术基础,并重点对其中卷积神经网络的基本结构、工作原理和关键算法进行研究,为基于深度学习的装配式混凝土结构灌浆缺陷识别的研究提供理论基础。

(2) 建立基于深度学习的装配式混凝土结构缺陷识别方法,该方法利用卷积神经网络直接提取装配式混凝土结构动力响应中的缺陷特征,结合定性分析和定量分析实现装配式混凝土结构的缺陷识别。

(3) 为验证该方法在装配式框架结构缺陷识别中的可行性和有效性,设计并完成一个缩尺装配式框架结构的动力检测试验。利用上述方法从缺陷存在性、缺陷位置和缺陷程度等三个层面对该框架进行缺陷识别。

(4) 为验证该方法在装配式剪力墙缺陷识别中的可行性和有效性,设计并完成一个足尺装配式剪力墙动力检测试验。利用上述方法从缺陷存在性和缺陷类型等两个方面对装配式剪力墙进行缺陷识别。

关键词: 装配式混凝土结构、灌浆套筒连接、缺陷识别、深度学习、卷积神经网络

ABSTRACT

With the continuous progress of China's social and economic strength, people have put forward higher requirements for architecture and living environment. The traditional extensive architecture model has been unable to meet the needs of The Times. Therefore, the state vigorously promotes the development of prefabricated building and tries to promote the transformation of traditional building model to new industrial building model. Prefabricated building adopts the construction method of factory prefabricated components and field assembly, and the vertical components are mostly connected by grouting sleeves. However, due to the complex construction process, the construction of joint nodes is prone to quality defects, which will inevitably have a certain impact on the safety of the structure. Local dynamic response signals contain abundant defect information. However, it is difficult for traditional data analysis methods to extract effective defect information from local response of prefabricated structure. Deep learning has become a sharp tool for data mining and defect recognition with its powerful data mining, feature extraction and classification recognition capabilities. Therefore, in this paper, the deep learning is applied to the structural defect detection of prefabricated concrete, and the method of structural defect detection of prefabricated concrete based on deep learning is established. The method uses deep learning to transform the local response signals layer by layer, and then extracts the defect information of the response signals. The dynamic test of prefabricated frame structure and prefabricated shear wall is designed, and the feasibility, effectiveness and accuracy of the method are systematically verified. The main research work is as follows:

(1) To sort out the technical basis of deep learning, and focus on the basic structure, working principle and key algorithm of convolutional neural network, so as to provide a theoretical basis for research on defect detection of prefabricated concrete structures based on deep learning.

(2) The defect detection method for prefabricated concrete structures based on deep learning is established. The method uses convolution neural network to directly extract the defect features in the dynamic response of prefabricated concrete structure, and combines qualitative analysis and quantitative analysis to realize the defect recognition of prefabricated concrete structure.

(3) In order to verify the feasibility and effectiveness of the proposed method in

defect detection of prefabricated frame structure, a dynamic test of reduced scale prefabricated frame structure is designed and completed. The proposed methods are used to identify the defects in the framework from three aspects: defect existence, defect location and defect degree.

(4) In order to verify the feasibility and effectiveness of this method in defect identification of assembled shear wall, a full-scale dynamic test of prefabricated shear wall was designed and completed. The proposed method is used to identify the defects of the assembled shear wall from the aspects of defect existence and defect type.

Key words: Prefabricated concrete structure, Grouting sleeve connection, Defect detection, Deep learning, Convolution neural network

目录

摘要.....	I
ABSTRACT.....	III
第 1 章 绪论.....	1
1.1 研究背景及意义.....	1
1.2 研究问题.....	2
1.3 装配式混凝土结构相关介绍.....	3
1.3.1 装配式结构节点连接类型.....	3
1.3.2 装配式混凝土结构灌浆缺陷成因.....	4
1.3.3 国内外灌浆套筒连接缺陷检测研究现状.....	5
1.4 深度学习及其在结构损伤识别中的应用研究现状.....	7
1.4.1 深度学习研究现状.....	7
1.4.2 深度学习在结构损伤/缺陷识别中的应用研究现状	9
1.5 研究内容.....	13
第 2 章 基于深度学习的装配式混凝土结构缺陷识别方法.....	15
2.1 前言.....	15
2.2 深度学习简介.....	15
2.3 卷积神经网络 (CNN) 简介	17
2.3.1 卷积神经网络.....	19
2.3.2 学习算法.....	27
2.3.3 反向传播算法.....	29
2.3.4 梯度下降算法.....	33
2.4 卷积神经网络设计.....	37
2.4.1 卷积神经网络拓扑结构.....	37
2.4.2 代价函数选取.....	41
2.4.3 激活函数选取.....	43
2.4.4 优化算法选取.....	45
2.5 基于深度学习的装配式混凝土结构缺陷识别方法.....	47
2.6 本章小结.....	48
第 3 章 装配式框架结构灌浆缺陷识别试验验证.....	51
3.1 前言.....	51
3.2 试验概况.....	51
3.2.1 试验模型介绍.....	51
3.2.2 缺陷引入方式.....	57
3.2.3 激振方法与试验仪器.....	58

3.2.4 各工况激振点及测点布置.....	60
3.2.5 试验过程.....	62
3.3 试验结果分析.....	62
3.3.1 加速度时程分析.....	62
3.3.2 数据前处理及数据集搭建.....	65
3.3.3 各工况缺陷及缺陷位置识别.....	67
3.3.4 不同缺陷程度识别.....	74
3.4 本章小结.....	79
第 4 章 装配式剪力墙缺陷识别试验验证.....	81
4.1 前言.....	81
4.2 试验概况.....	81
4.2.1 试验模型介绍.....	81
4.2.2 缺陷引入方式.....	87
4.2.3 激振方式与激振点布置.....	89
4.2.4 各工况测点布置.....	91
4.2.5 试验过程.....	92
4.3 试验结果分析.....	93
4.3.1 加速度时程分析.....	93
4.3.2 数据前处理.....	95
4.3.3 数据增强及数据集创建.....	96
4.3.4 各板带缺陷识别.....	98
4.4 本章小结.....	105
第 5 章 结论与展望.....	107
5.1 结论.....	107
5.2 进一步工作的方向.....	108
致谢.....	109
参考文献.....	112
个人简历、在读期间发表的学术论文与研究成果.....	117

第1章 绪论

1.1 研究背景及意义

建筑业是我国国民经济的支柱产业之一，与全国经济的发展、人民生活的改善有着密切的关系，其目前仍旧是一个劳动密集型、生产方式相对落后的传统产业。我国建筑结构形式多为混凝土结构，主要采用现场施工的传统生产方式，存在工业化程度低、设计建造粗放、建筑材料损耗及建筑垃圾量大、建筑全寿命周期能耗高等诸多问题。随着社会经济的发展，这种传统建造方式已不能够满足人们对高品质建筑和人居环境质量的美好需求，粗放的发展模式也已不能适应建筑业高质量发展的时代要求。装配式建筑以设计标准化、生产工厂化、施工装配化、装修一体化和管理信息化的“五化一体”的新型建造方式为主要特征^[1]，其不仅使资源和能源获得充分利用，能有效控制施工过程中出现的污染，而且能提升劳动生产的效率和质量安全水平，进而促进了建筑业和信息化的融合，同时也符合国家绿色、高质量发展的战略要求。发展装配式建筑就是要走出一条科技含量高、建筑质量优、经济效益好、人力资源优势得到充分发挥的新型建筑工业化的道路。为此，国家大力发展装配式建筑，就是促进传统粗放的建造方式向新型工业化建造方式转变，实现建造方式的变革。

从2013年发展改革委、住房城乡建设部发布《绿色建筑行动方案》开始，国家密集颁布关于推广装配式建筑的政策文件，在发展规划、标准体系、产业链管理、工程质量等多个方面作出了明确要求。例如，2016年2月，国务院颁发《关于进一步加强城市规划建设管理工作的若干意见》，标志着国家正式将推广装配式建筑提升到国家发展战略的高度。《意见》强调我国须大力推广装配式建筑，建设国家级装配式生产基地；加快政策支持力度，力争用10年左右时间，使装配式建筑占新建建筑的比例达30%。2017年3月，住建部出台《“十三五”装配式建筑行动方案》，明确到2020年，全国装配式建筑占新建建筑的比例达到15%以上，其中重点推进地区达到20%以上，积极推进地区达到15%以上，鼓励推进地区达到10%以上。2016年以来关于建筑工业化的国家政策如图1.1所示。在国家的政策指导下，各地方政府相继出台对于装配式建筑的推广文件。根据不完全统计，地级市以上政府和相关部门出台的相关文件超过100份。在当前的形势下，国家层面的扶持完善，地方政府落实到位，可以预见，未来十年将是装配式建筑高速发展的黄金时代。

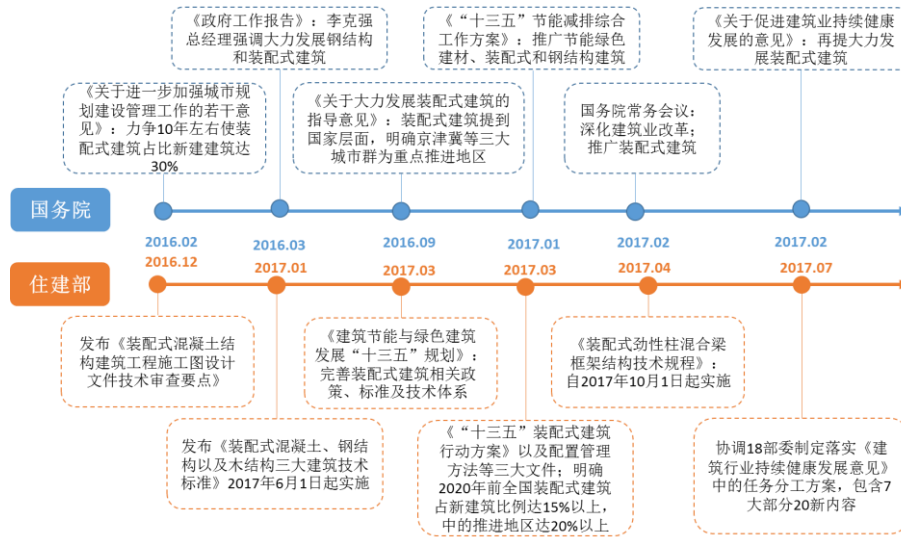


图 1.1 2016 年以来国家政策解读

装配式建筑主要有三种技术模式：装配式预制混凝土结构（PC，Precast Concrete）、钢结构和木结构。这三种技术各自有其技术特点和优势。其中，装配式混凝土结构建造成本较低，适合量大面广的多层、小高层办公楼、住宅建筑。装配式混凝土结构在传统技术框架基础上侧重于外墙板、内墙板、楼板等构件的预制化，预制化率为 40%~50%，如果延伸至现场装修一体化，成本可进一步压缩至接近传统技术成本，并能实现约 5 天建一层的高效率。在量大面广的多层建筑，尤其是住宅领域有广泛的应用场景。从住建部认定的首批 64 个装配式建筑示范项目来看，混凝土结构占比最大达 64%，共 41 项（钢结构 19 项，木结构 4 项）。

装配式混凝土结构连接节点的质量对结构整体性能的影响巨大，然而由于施工工序复杂，在连接节点的施工过程中容易产生质量缺陷，这必然会对结构的安全性产生一定的影响。为了保证装配式结构的整体性能，延长寿命周期，采取有效合理的措施与技术手段对梁柱连接节点进行缺陷检测和识别，及时发现存在质量缺陷的连接节点并加以维护，则显得尤为必要和迫切。因此，对装配式混凝土结构缺陷识别的研究既具有重要的理论意义，也具有非常重要的现实意义。

1.2 研究问题

装配式混凝土结构中连接节点直接影响到结构整体的性能，若能尽早发现连接节点的缺陷，就可以避免由于局部缺陷不断发展带来的各项损失。因此建立装配式混凝土结构的缺陷识别方法是非常必要的。

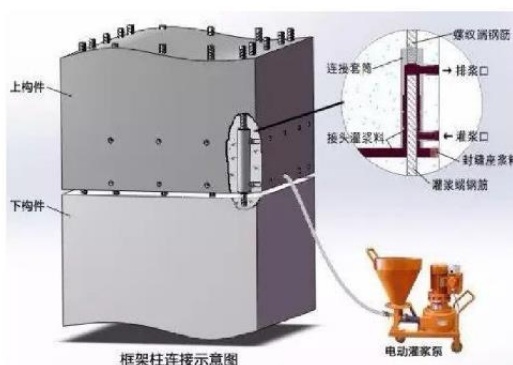
局部动力响应信号蕴含着非常丰富的缺陷信息，如何从局部动力响应中提取

缺陷信息成为缺陷识别研究的重点和难点,传统的数据分析方法难以从大量的数据中提取出有效的信息。深度学习凭借其强大的数据挖掘、特征提取和分类识别能力,成为数据挖掘和缺陷识别的利器,也为该问题提供了一个理想的解决方案。因此,本课题拟将深度学习算法应用到装配式混凝土结构缺陷识别中,对基于深度学习的装配式混凝土结构缺陷识别方法进行研究。旨在利用深度学习对结构局部响应信号进行逐层特征的变换,最后提取出能反映结构响应缺陷信息的特征,并将定性分析和定量分析相结合,实现装配式混凝土结构的缺陷识别。

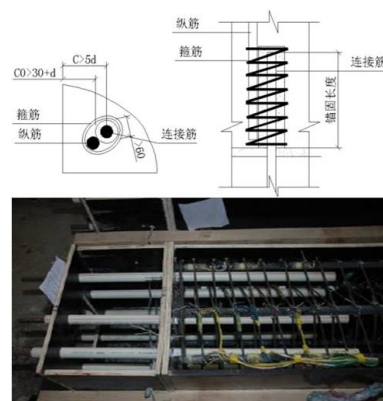
1.3 装配式混凝土结构相关介绍

1.3.1 装配式结构节点连接类型

装配式建筑的施工过程包含工厂预制和现场装配两个过程,需要通过可靠的连接方式将预制构件与现场后浇混凝土、水泥基灌浆料等连接成为一个完整的装配式混凝土结构体系,因此连接节点处的性能会直接影响到整体结构的承载能力和抗震性能^[2]。对于装配式建筑的节点连接方式,国内外的很多学者进行了相关研究。在现阶段的装配式混凝土建筑中,一般所采用的连接方式包括湿连接和干连接两类,湿连接包括浆锚连接、后浇混凝土连接和钢筋套筒连接等三种方式;干连接包括机械套筒连接、牛腿连接、焊接连接以及螺栓连接^[3]。各连接方式如图 1.2 所示。学者研究表明湿连接中整体性能更为优越,并且与现浇节点拥有等同的抗震性能^[4]。《装配式混凝土结构技术规程》规定当房屋高度大于 12m 或层数超过 3 层时,预制柱宜采用套筒灌浆连接^[5],钢筋灌浆套筒连接成为实际工程中应用最为广泛的一种连接方式。



(a) 灌浆套筒连接



(b) 浆锚连接

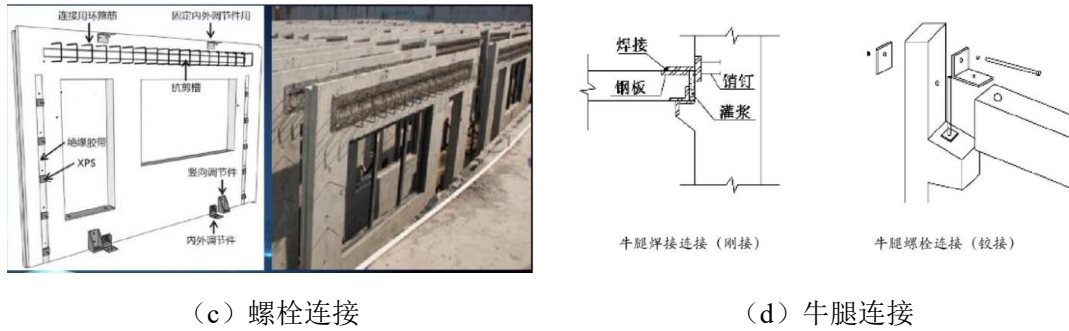


图 1.2 装配式结构节点构造连接方式

灌浆套筒的连接技术主要包括带肋钢筋、套筒和灌浆料等三部分，该方法将待连接的两个钢筋分别从套筒的两端插入其中，并在钢筋和套筒之间灌注专用灌浆料，来实现钢筋的连接。灌浆料通常具有微膨胀和高强度等特性，灌浆套筒连接即利用灌浆料的微膨胀特性使得套筒内侧和钢筋间的正向作用力进一步加强。钢筋中产生的应力则通过上述正向作用力和钢筋表面的摩擦力进行传递^[6]。

1.3.2 装配式混凝土结构灌浆缺陷成因

采用钢筋套筒灌浆连接接头不仅很好地解决了预制装配式混凝土结构中的纵向钢筋连接问题，并且能够有效地保证钢筋的传力效果，实现装配等同现浇的设计要求。但目前通常采用的套筒大多内部结构复杂，制作工艺繁琐，对施工安装过程中的对中精度要求较高，存在连接速度较慢、施工工序复杂等缺点。因此，在施工过程中会不可避免的出现灌浆缺陷的问题，根据已有研究^[7]，钢筋灌浆套筒连接主要的灌浆缺陷分为端部缺陷、水平缺陷、中部或均布缺陷以及偏心缺陷，缺陷形式如图 1.3 所示。

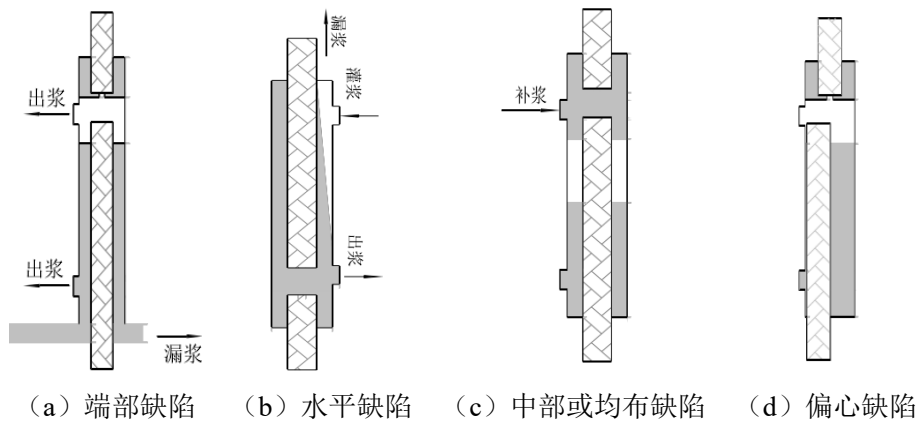


图 1.3 套筒灌浆连接缺陷图例

1.3.3 国内外灌浆套筒连接缺陷检测研究现状

《装配式混凝土结构技术规程》JGJ1-2014 作为行业标准由住建部在 2014 年发布,该规范在工程设计、验收等多个层面为我国装配式混凝土建筑提供技术支持。2018 年 12 月 6 日,《装配式混凝土建筑技术标准》GB/T 51231-2016 在住建部于南宁举办的“推动城市高质量发展系列标准发布”活动中发布。此标准对于《装配式混凝土结构技术规程》中所规定的技术内容和条文进行了补充与完善,同时丰富、发展了装配式混凝土结构的成熟的新技术、新工艺。在这两本国家标准依然采用监理人员现场监督的方法对灌浆质量进行保证,并采用平行试件拉伸试验的方法对质量进行验证^[8]。鉴于以上情况,诸多学者对套筒灌浆缺陷的无损检测技术展开研究,目前主要的检测技术有 X 射线工业 CT 技术、超声波法、冲击回波法、预埋钢丝拉拔法等。

(1) X 射线工业 CT 技术

X 射线具有较强的穿透性,可根据其在传播过程中的衰减、吸收和再生散射定律,将受到不同程度吸收的射线投射到 X 射线胶片上,经显影后获得与材料结构或缺陷相对应的不同图像,借以确定缺陷的种类、大小、数量和分布情况,从而判定缺陷的危害性和质量等级。由于放射线检测技术具有图像化的显示效果和检测高精度等优点,在混凝土缺陷检测领域很早就被尝试应用^[9]。

东南大学安琳^[10]等人首先利用我国现有的钢结构 X 射线探伤设备开展了预应力混凝土桥梁结构中灌浆空洞无损检测的室内试验,并成功应用于某实际桥梁的检测中,是我国首次采用 X 射线检测灌浆空洞的实例。

张富文等人展开了基于 X 射线工业 CT 技术检测套筒灌浆密实度的试验研究,结果表明该技术可以通过获取灌浆套筒内部的影像实现灌浆缺陷的检测^[11]。在实验室研究的基础上,该课题组利用便携式 X 射线技术检测了上海市某装配整体式混凝土剪力墙的套筒灌浆质量。结果表明该技术在部分套筒灌浆密实度的检测中具有良好的可行性,适用范围包括套筒居中以及梅花形布置的装配式剪力墙。该方法能通过 X 射线成像技术对套筒外形、未灌浆区以及灌浆密实区等进行观察。但是该方法也存在适用性不足的问题,即在套筒双排布置的装配式剪力墙灌浆密实度检测中,该方法难以区分灌浆缺陷的具体位置^[12]。

(2) 超声波法

混凝土缺陷无损检测技术中,最常用的技术是超声波法,混凝土结构无损检测已经广泛地使用了超声波法。20 世纪 40 年代末 50 年代初,加拿大学者 Leslie 等人和英国学者 Jones 等创造性地将超声波用于混凝土的检测中^[13]。20 世纪 50 年代,我国引进了超声波仪器,是为该领域研究的肇始。

潘绍伟^[14]等人开展了钢管混凝土拱桥超声波检测研究,研究表明,对于钢管混凝土拱桥内混凝土质量可采用超声波检测法,而采用低频超声检测,由声速、波幅和波形以及脉冲回响等参数综合评价,具有较高的准确性。在理论分析的基础上,超声检测成功应用到万县长江公路大桥钢管混凝土拱的检测,较可靠的检测出其存在的细微缺陷。

2014年,聂东来^[15]等人在装配式节点处浇筑混凝土之中首次使用超声波法。检测的指标包括与竖向钢筋连接的钢筋套筒灌浆料的密实度。通过测得的超声波声速与幅值,可以判断与竖向钢筋连接的灌浆套筒的密实性,但该方法并不能准确识别出灌浆缺陷的具体位置。

张苏承等人的研究表明该方法需要对接收信号进行经验性的判断,这极大地增加了人主观因素的影响;同时该方法在对距离侧面较近灌浆套筒的检测中,得到的结果并不理想^[16]。

(3) 冲击回波法

20世纪80年代,美国国家标准与技术研究院(NIST)联合康奈尔大学提出了冲击回波法(Impact-echo Method,简称IE法),并将其应用到混凝土无损检测中^[17]。该方法凭借其强大的穿透力、广泛的卓越频率分布以及现场操作方便的优势很快就在土木工程中得到广泛的应用。

2000年,JAEGER等^[18]首次在预应力孔道内部缺陷的检测中应用了冲击回波法,试验结果表明冲击回波法在后张法预应力孔道灌浆密实性的检测中具有很强的适用性。

2011年,OSLON等^[19]为评估预应力混凝土桥面的损伤,将冲击回波法应用到预应力管道灌浆密实度的检测中,结果表明该方法能够有效的评估预应力管道中灌浆的密实度,并以此为基础成功的评估了桥梁的损伤。

我国对冲击回波法的研究开始于20世纪80年代,南京水利科学研究院率先将该方法应用到混凝土结构厚度和内部缺陷的检测,并成功开发了一套完整的系统。近年来,我国诸多学者展开了将冲击回波法应用于装配式建筑连接节点灌浆缺陷监测的研究。

庞涛等^[20]等为检测为南京大地伊丽雅特湾项目158楼中预制剪力墙浆锚连接节点的灌浆密实度,采用冲击回波法进行了对应的模型试验和现场检测。检测结果证明了冲击回波法在检测浆锚节点的注浆密实情况和缺陷位置中的可行性和适用性。

刘辉等^[21]展开基于冲击回波法监测钢筋灌浆套筒连接灌浆密实度监测的试验研究。试验结果表明:在套筒居中布置的灌浆套筒密实度检测中,冲击回波法能够对试件的灌浆密实度仅能做定性判断,而无法做详细的定量分析;在双排布

置的灌浆套筒密实度检测中，冲击回波法无法定性地判断试件的灌浆密实度。

(4) 预埋钢丝拉拔法

预埋钢丝拉拔法是在套筒的出浆口预埋高强钢丝，养护一段时间后，对预埋钢丝进行拉拔，通过拉拔荷载值判断灌浆饱满程度。高强钢丝、拉拔设备是常用的检测仪器，该方法具有许多优点，包括操作简便，成本低廉，但依然有许多缺点，包括无法确定缺陷位置、用拉拔法并不能实现无损检测、需要 3 天左右的养护期以及测量方法的误差较大^[9]。

高润东等^[22]通过设计不同的参数对预埋在套筒灌浆料中的钢丝进行了拉拔试验，这些参数包括 3 种不同直径的高强钢筋、3 种锚固长度以及 2 种灌浆料养护时间。试验结果表明该方法是一种简单、有效的套筒灌浆饱满度检测方法，依据试验结果给出应该工程实际检验的建议为高强钢丝直径为 5mm，锚固长度为 30mm，灌浆养护龄期为 3 天。

以上所述的灌浆缺陷识别方法均为现阶段常用的装配式结构灌浆缺陷识别方法，这些方法具有可视化和局部试验完成特点部位检测的优点。但是，这些方法面临同样的问题，即技术设备复杂，器材成本较为昂贵，主观因素强等。同时其中部分检测方法对待检测构件的厚度、以及构件中套筒的分布均有一定的限制。利用装配式结构的动力响应则可以很好地克服以上困难，因此开展基于动力响应的装配式结构灌浆缺陷识别方法的研究极为必要。

1.4 深度学习及其在结构损伤识别中的应用研究现状

1.4.1 深度学习研究现状

深度学习是一种代表性学习方法，它使网络体系结构能够自主地从原始数据中学习高度抽象的特征，从而完成识别或分类任务，它是机器学习的一个分支，主要从人工神经网络演化而来。其发展的起源可追溯到 1943 年由 McCulloch 和 Pitts 提出的第一个人工神经细胞模型。1958 年 Rosenblatt 提出了由多个神经细胞组成的单层感知器结构，该结构可以通过感知器收敛算法进行学习，提高分类能力。1986 年 Rumelhart 等利用反向传播算法训练多层神经网络，使隐含层能够构造有用的特征进行分类^[23]。

在人工神经网络的基础上，Le Cun 等人^[24]于 1989 年开发了第一个深度卷积神经网络（Convolutional neural networks, CNN）——LeNet-5，通过反向传播算法（Back propagation, BP）对网络进行有监督的训练，来实现识别手写邮政编码的目的。在 Mnist 手写数字数据集的训练和测试中，LeNet-5 的识别准确率超过

99.65%。

1990年, Elman^[25]提出了一种具有局部存储单元和反馈连接的递归神经网络(Recurrent neural networks, RNN)处理时间序列数据。

然而, 由于梯度消失或爆炸等问题, 深度学习网络的训练比较困难, 这一挑战阻碍了深度学习的发展。直到 Hinton 等^[26]于 2006 年开发了深度信念网络(deep belief network, DBN)。他们提出了利用无监督预训练对网络的权值进行初始化, 再利用有监督训练对权值进行微调的解决方法, 由此打开了深度学习的大幕。

2011 年, 微软研究院^[27]的科研人员首次将深度学习应用到语音识别中, 使识别错误率下降 20%-30%。

2012 年, 首次参加 ImageNet 比赛的 Hinton 课题组^[28]就凭借 CNN 网络——AlexNet 打败了 Google 团队, 一举夺得冠军。AlexNet 首次将 ReLU 激活函数应用到卷积神经网络中, 使得 CNN 的收敛速度有了明显的提升且减少了梯度消失发生的可能性; 还扩展了 LeNet5 结构, 添加 Dropout 层减小过拟合。这些技术的联合应用使 AlexNet 在 1000 个不同类别的图像分类中获得了 15.3% 的错误率。

2012 年 6 月, 谷歌首席架构师 Jeff Dean 和斯坦福大学教授 Andrew Ng^[29]主导著名的 Google Brain 项目, 将 1.6 万台电脑的处理器相连接构建出一个全球最大的中枢神经网络, 使其具备自我学习的能力。

2015 年, Facebook 团队^[30]完成 Facenet 网络, Facenet 不同于传统的 CNN 方法。传统的 CNN 先通过网络进行处理, 然后将处理后的结构利用 SVM 方法进行分类。该方法是直接通过学习将特征变为欧式平面上的一个点, 然后通过比较点之间的距离来进行判断。Facenet 利用 9 层深度神经网络和 3 个无权重共享的局部连接层来实现人脸特征的提取, 最终实现在 LFW 数据集上的准确度已经达到了 99.6%, 目前是数据集上检测的最好记录。

在此之后越来越多的研究人员展开了对深度学习相关理论和应用的研究, 深度学习得到了爆发式的发展, 现在已出现多种不同功能的深度学习网络。比如侧重于图像中目标对象检测的 CapsuleNet; 侧重于分类任务的 VGG-Net、ZF-Net、GoogLeNet 和 ResNet; 侧重于目标检测的 MobileNet、SegNet 和 ShuffleNet; 侧重于图像、视频等处理的 GAN、f-GAN、EBGAN 和 InfoGAN; 侧重于细分任务的 U-Net、DeconvNet、CRF-RNN、ENet、PSPNet、RefineNet、fully convolutional network (FCN)、DenseNet 和 Deeplab^[31]。多种深度学习网络的出现使其在语音识别、图像识别、机器翻译以及自然语言处理等多个方面得到广泛应用。深度学习的发展历史如图 1.4 所示。

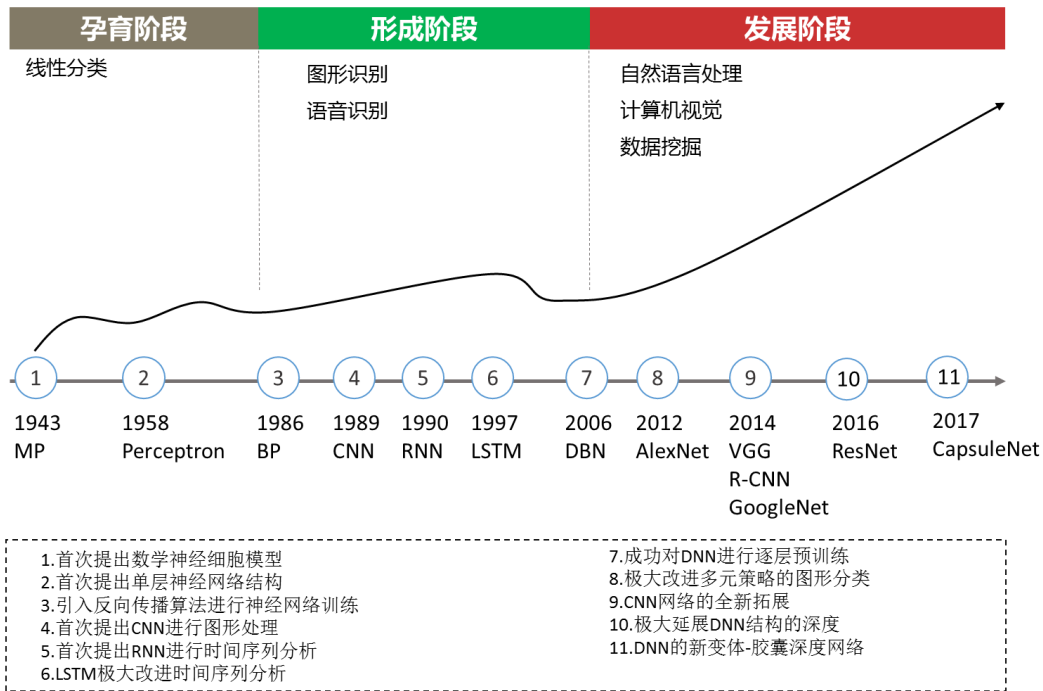


图 1.4 深度学习发展历史

1.4.2 深度学习在结构损伤/缺陷识别中的应用研究现状

结构损伤/缺陷识别是指利用结构性态的改变来识别结构中的损伤/缺陷，也是结构健康监测（Structural healthy monitoring, SHM）中的重要任务之一。传统的结构损伤/缺陷识别方法常通过处理结构的动力响应，获取相应的指标来评估结构的损伤/缺陷。但是随着结构健康监测系统在各种大型建筑上的广泛应用，结构的损伤识别常需要对大量的数据进行分析和处理，传统的数据处理方法难以有效地挖掘这些大数据的包含的特征信息。由于计算机硬件和计算能力的迅猛发展，利用深度学习对数据进行处理的方法为结构损伤/缺陷识别中的图像或振动响应数据特征挖掘、大数据分析、自主处理数据等方面提供了新的途径。因此许多人尝试将深度学习应用于土木工程结构的损伤/缺陷识别中^[32]，深度学习在结构损伤/缺陷识别中应用的主要形式表现为图像处理技术与结构时程响应处理两类，本节将分别对这两类方法进行回顾。

（1）图像处理技术

2017年，Cha等^[33]提出了一种基于卷积神经网络（CNN）的结构裂缝检测方法。采用40000张 256×256 像素分辨率的图像对CNN进行训练，记录精度达到98%。训练后的CNN与滑动窗口技术相结合，将不同条件下（如照片尺寸、强光点、阴影的裂缝）裂缝的照片进行测试，结果表明，该方法具有较好的性能，能够在实际情况下发现混凝土裂缝。

2018年，Dorafshan等^[34]比较了常用边缘检测器和深卷积神经网络（DCNN）

在混凝土结构图像裂缝检测中的性能。使用六种常见的边缘检测方案 (Roberts、Prewitt、Sobel、Laplacian of Gaussian、Butterworth 和 Gaussian)，并使用 AlexNet-DCNN 结构在完全训练、传递学习和分类器模式下，分析了 19 幅混凝土高清晰度图像 (3420 个子图像、319 个裂缝和 3101 个无裂缝) 的数据集。结果表明，DCNN 方法在混凝土损伤图像检测中的应用前景广阔。为了降低残差噪声，提出了一种 DCNN 与边缘检测器相结合的混合方法，将残差噪声降低 24 倍。

2018 年，Bao 等^[35]将计算机视觉和深度学习应用到异常数据的检测中，并提出了相应的检测方法。该方法首先将时间序列信号转换成图像向量，并作为训练数据集输入深度神经网络中，利用堆叠式自动编码器和贪婪分层训练技术对建立的深度学习网络进行训练。采用了中国某实际大跨度桥梁结构健康监测系统的加速度数据进行验证，结果表明，该方法能较准确地自动检测数据的多模式异常，总体检测精度达 87%。

2019 年，Tang 等^[36]设计了一个五层 CNN 来检测和分类来自 SHM 系统的异常数据。该方法将原始的时间序列数据分成若干部分，分别在时域和频域对数据进行可视化处理，然后将时域图像与频域图像叠加成一个双通道图像。将相应标记的图像作为 CNN 的输入，通过卷积的方法对异常特征进行描述和表示。采用了我国某斜拉桥的 SHM 系统实际加速度数据进行了验证，结果表明，该方法能够有效、准确地检测出 SHM 数据的多模式异常。

2019 年，Ni 等人^[37]提出了一种基于深度学习的混凝土裂缝宽度定量测量方法。该方法设计了双尺度卷积神经网络 (GoogLeNet 和 resnet) 来检测复杂场景图像中的裂纹，并验证了其准确性。在此基础上，提出了一种基于 Zernike 矩算子的裂缝宽度估计方法。并通过某室内加载试验和某结构的现场实测对该方法的有效性进行评估，结果表明该方法在混凝土裂缝宽度定量测量中展现出良好的性能。

2019 年，Xu 等人^[38]提出了一种基于改进的快速卷积神经网络 (faster R-CNN)，用于从图像中识别和定位受损钢筋混凝土柱的多类型地震损伤 (即混凝土开裂、混凝土剥落、钢筋外露和钢筋屈曲)。通过现场成像的方法建立图像数据集，采用数据增强的方法对数据集进行扩充。识别结果表明，R-CNN 能够自动识别和定位上述多类型地震损伤，总体平均精度达到 80%。

2019 年，Beckman 等人^[39]提出了一种基于快速卷积神经网络 (faster R-CNN) 的混凝土层裂损伤检测方法。通过深度传感器采集和建立了一个由 1091 幅 (853x1440 像素) 损伤图像组成的数据库，并利用该数据库对深度学习网络进行了训练和验证。计算结果表明该方法的平均精度为 90.79%。当试验构件和传感器之间的距离从 100cm 到 250cm 时，识别的平均精度误差 (MPE) 为 9.45%。在相

同距离范围内, 最大损伤深度测量的 MPE 为 3.24%。

(2) 结构响应处理技术

2018 年, 李雪松等^[40]提出一种基于卷积神经网络 (CNN) 的桥梁损伤识别方法。以一简支梁 9 个测点的加速度响应数据作为输入训练和测试 CNN, 并分析了不同激励大小、不同噪声环境对 CNN 识别结果的影响, 结果显示在弱激励和噪声环境下 CNN 在桥梁损伤识别中具有良好的适用性。

2018 年, 李贵凤等^[41]利用长短期记忆网络对桥梁的监测数据进行处理, 丰富了深度学习在桥梁损伤识别中的应用。为提高监测数据分析的时效性, 使用主成分分析获取简化特征值, 并以某大桥的仿真实验数据为例, 对比了基于 LSTM 的识别方法、多层感知机以及支持向量机的识别准确度, 结果显示, 基于 LSTM 的方法具有最高的识别准确度。

2018 年, 谢祥辉等^[42]提出基于堆栈降噪自动编码器的桥梁损伤识别方法, 该方法利用多个自动编码器对桥梁的加速度响应进行损伤特征的提取, 并利用 Softmax 来判断桥梁的损伤模式。最后利用连续梁桥试验, 对比了该方法与现有 BP 神经网络的识别精度和鲁棒性, 结果显示, 该方法具有更高的识别精度和鲁棒性。

2019 年, 赵焯等^[43]提出了一种基于卷积神经网络的叶片损伤识别方法。通过比较已有 CNN 的网络结构之后, 将 ResNet 选为基础模型结构, 并将批量梯度下降算法与 CNN 融合以提高学习效率。最后试验验证得到叶片损伤识别率达到了 96.75%。

2017 年, Lin 等人^[44]提出了一种基于卷积神经网络 (CNN) 的损伤识别方法, 通过深度学习从结构的时域响应数据中自动提取特征。为验证该方法的识别精度, 设计了一个简支欧拉-伯努利梁的数值试验, 采用梁在不同工况 (不同位置损伤、不同程度损伤以及不同激励位置) 下的响应时程对 CNN 进行训练和测试, 结果显示 CNN 网络在无噪声和有噪声数据集上都有很好的损伤识别精度。

2017 年, Abdeljaber 等^[45]开发了一种基于一维 CNN 的结构损伤识别系统, 该系统采用将特征提取和分类块融合成一个完整的学习模块, 从而实现了基于振动的损伤检测和损伤的实时定位。进行的大量振动台实验验证了该方法的实用性和计算效率。

2018 年, Pathirage 等^[46]提出了一种基于自动编码器的深度学习网络, 该网络可以通过振动响应识别结构损伤。以固有频率、振型等振动特性作为输入, 以结构损伤作为输出向量。对自编码器中的隐藏层进行逐层训练, 并进行微调以优化整个网络。对钢框架结构进行了数值和实验研究, 与传统的人工神经网络方法相比, 验证了该方法的准确性和有效性。

2018年, Rafiei 和 Adeli^[47]提出了一种利用传感器采集的结构环境振动响应,对结构系统进行整体和局部健康状况评估的方法。该模型采用同步压缩小波变换、快速傅里叶变换和无监督深度玻尔兹曼机对记录信号进行频域特征提取。使用概率密度函数评估整体和局部的健康状况。最后利用振动台实验数据,说明了该模型的有效性。

2018年, Lee 等^[48]比较了具有不同隐藏层、激活函数和优化算法的 DNN 体系结构,以测试不同组合的性能。对桁架结构进行了数值分析,采用结构响应作为训练和验证数据集,结果显示在包含两个隐藏层的结构中, softplus 激活函数和 Adam 优化算法是有效的。

2019年, Khodabandehlou 等^[49]建立一个 11 层二维 CNN 从加速度响应时程中提取特征,使得在有限的加速度测量下进行桥梁损伤状态分类成为可能。通过对某钢筋混凝土公路桥的模型在不同荷载作用下的振动台试验,验证了该方法的适用性和有效性。结果表明,该方法对结构条件的微小变化具有较强的鲁棒性和敏感性。

2019年, Duan 等^[50]提出了一种基于 CNN 的桥梁损伤检测方法。对某拱桥在不同损伤条件下的加速度响应进行了数值分析。分别以加速度响应时程和傅立叶谱作为训练数据集,比较了 CNN 损伤识别的性能。结果表明,使用傅立叶谱作为输入的 CNN 在各种损伤状态下的性能优于使用时程作为输入的 CNN 和使用傅立叶谱作为输入的传统神经网络。

2019年, Gulgec 等^[51]设计了一个 CNN 拓扑来对模拟的损伤和健康的情况进行分类,并在损伤存在时进行定位。为选择一个特征提取器,我们训练并比较了 50 个不同学习速率、卷积层和全连通层的 CNNs,并最终确定了最优的 CNN。通过对未损坏和损坏的钢结构连接进行有限元模拟,对所提方法的性能进行了评估。通过使用应变分布作为几种不同损伤场景下和不同载荷下的结果来训练样本。基于所提出的研究结果,该框架在损伤诊断和定位具有较高的准确性、鲁棒性和计算效率。

2019年, Zhou 等^[52]提出了一种基于刚度退化的损伤指数分类方法的深度学习网络(DLN),用于损伤识别。训练数据是通过 30,000 个真实的、随机产生的迟滞环的数值模拟获取,训练后的 DLN 模型的性能是通过 1800 个额外的模拟三层“虚拟”建筑和一个三层实体建筑的实验数据来评估。结果表明,DLN 有潜力在不需要详细工程分析的情况下,提供快速、实时的警报或其他关于损坏状态和紧急响应。

2019年, Wang 等^[53]提出了由 ResNet 和 DenseNet 组成的双路径网络。利用多维振动信号建立了空间时频数据集,通过包含环境噪声的监测数据对不同的铁

路事件进行分类。通过铁路现场安全监测试验，验证了该方法的有效性。

2019年，Kim和Sim^[54]提出了一个基于深度学习的框架，该框架由一个快速R-CNN和一个区域建议网络组成，用于频率域模式识别中的自动峰值提取，该体系仅使用计算机软件生成的信号进行训练。利用梁、桁架和斜拉索三种试验结构的加速度响应进行了实验室和现场试验，以验证所提出的峰值检测器的识别性能。结果表明，该框架能够成功地识别出大部分的显著峰值，且具有较高的精度。

2019年，Wu和Jahanshahi^[55]提出了一种基于深度卷积神经网络(CNN)的方法来估计线性单自由度(SDOF)系统、非线性SDOF系统和三层多自由度(MDOF)钢框架的动态响应。考虑了多种噪声信号的情况，并以传统的多层感知器(multilayer perceptron, MLP)算法为参考，提出了CNN方法。数值模拟和实验结果表明，该方法能够准确地预测结构响应，对噪声数据的鲁棒性优于MLP算法。

综上所述，随着深度学习技术的蓬勃发展以及计算能力的不断增强，国内外土木工程领域的诸多学者纷纷展开了将深度学习技术应用于结构损伤/缺陷识别中的研究，可以预见其在结构损伤/缺陷识别领域拥有广阔的发展空间和应用价值。深度学习在结构损伤/缺陷识别中应用的主要形式表现为图像处理技术与结构时程响应处理两类，卷积神经网络在这两方面的应用都表现出很强的实用性，故利用卷积神经网络对装配式结构的缺陷识别具有一定的研究前景和潜在价值。基于图像处理的识别方法需要采集响应的图像作为数据集，然而在实际结构中很难对装配式混凝土结构的缺陷进行图像的采集，因此本文将采用卷积神经网络对时程响应进行处理进而实现装配式混凝土结构的缺陷识别。

1.5 研究内容

随着我国装配式建筑的逐步推广，越来越多的新建工程将会采用装配式结构的形式，以往的探伤研究中少有针对装配式混凝土结构节点缺陷检测方面的相关研究。本文依托于国家重点研发计划子课题：装配式结构整体性检测评价技术，将时下推广的装配式建筑与前沿的检测方法相结合，丰富了研究资料不多的装配式结构识别领域，为装配式结构在我国的推广起到了积极的促进作用。

第一章为绪论，主要阐述了本课题的研究背景，对目前装配式结果灌浆套筒缺陷识别技术的发展情况进行梳理，并对深度学习的发展路径以及其在结构损伤和缺陷识别领域的应用进行了详细的梳理。

第二章为基于深度学习的装配式混凝土结构的缺陷识别方法。本章首先阐述了深度学习的基本信息、基本思路以及一些常见的算法，其中包括使用最为广泛

的卷积神经网络中的卷积运算、非线性激活、下采样等，为后续对基于深度学习的装配式混凝土结构灌浆缺陷识别的研究提供理论基础。其次通过对卷积神经网络拓扑结构以及其中的关键函数进行介绍和比选，最终建立了基于深度学习的装配式混凝土结构的缺陷识别方法。

第三章为装配式框架结构灌浆缺陷识别试验验证。设计具有灌浆缺陷的装配式框架结构，并对其进行动力检测试验，采集加速度响应。以加速度时程作为卷积神经网络的输入，对第三章建立的识别方法在装配式框架结构灌浆缺陷识别中的可行性、有效性和识别准确性进行验证。

第四章为装配式剪力墙缺陷识别试验验证。设计具有不同缺陷类型（灌浆缺陷、坐浆层缺陷以及灌浆和坐浆层均有缺陷）的装配式剪力墙，并对其进行动力检测试验，采集加速度响应。以加速度时程作为卷积神经网络的输入，对第三章建立的识别方法在装配式剪力墙缺陷识别中的可行性、有效性和识别准确性进行验证。

第五章为结论与展望。本章对全文的研究工作进行归纳总结，根据以上研究成果进行概括得出结论，并对下一步的研究方向进行展望。

第2章 基于深度学习的装配式混凝土结构缺陷识别方法

2.1 前言

如绪论所述,由于深度学习在大数据处理和特征提取等方面的优越性能,其在结构损伤/缺陷识别领域的应用也逐渐成为学者们关注的热点。本文将深度学习与装配式混凝土结构的灌浆缺陷识别问题相结合,对基于深度学习的装配式混凝土结构的缺陷识别展开研究。为更好地进行后续的研究,本章将首先对深度学习的基本思路以及一些常见的算法进行研究,为后续对基于深度学习的装配式混凝土结构灌浆缺陷识别的研究提供理论基础。在此基础上,将建立基于深度学习的装配式混凝土结构缺陷识别方法。该方法利用卷积神经网络直接提取装配式混凝土结构动力响应中的缺陷特征,并利用全连接层将其映射成缺陷概率作为输出,最终实现装配式混凝土结构的缺陷识别。本章将对基于深度学习的装配式混凝土结构缺陷识别方法做详细介绍,并对该方法中采用的卷积神经网络进行介绍以及其中的关键函数进行比选。

2.2 深度学习简介

深度学习是机器学习(Machine Learning)的一个研究方向,而机器学习是人工智能的核心。三者的关系如图2.1所示。机器学习是一门多领域交叉学科,是指用某些算法指导计算机利用已知数据得出适当的模型,并利用此模型对新的情境给出判断的过程^[56]。传统机器学习的研究方向主要包括如线性回归(Linear Regression)、K均值(K-means)、决策树(Decision Trees)、随机森林(Random Forest)、支持向量机(Support Vector Machine, SVM)以及人工神经网络(Artificial Neural Networks, ANN)^[56]。

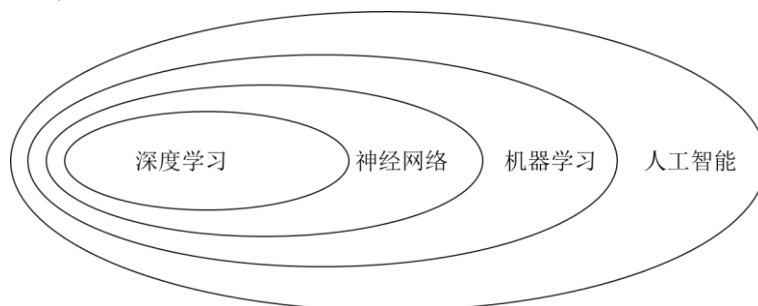


图 2.1 深度学习、机器学习和人工智能的关系示意图

深度学习源于神经网络的研究，其概念最早由多伦多大学的 Hinton 提出，是指利用一定的训练方法对样本数据展开训练，得到含有多个层级的深度神经网络结构。深度学习构建的深度神经网络成功地模拟了人类大脑的分层结构，从而能很好地模拟了人类的学习过程。从本质上来讲深度学习即为更深层次的人工神经网络，因此也可以称之为深度神经网络(Deep Neural Networks, DNN)。

与传统的神经网络类似，深度学习构建的深度神经网络通常由一个输入层 (Input layer)、多个隐藏层 (Hidden layer) 和一个输出层 (Output layer) 组成。隐藏层的功能是把输入数据的特征，抽象到另一个维度空间，来展现其更抽象化、更高阶的特征^[57]。深度学习通过对输入层的信息进行逐层特征提取，即上一层提取的特征 (输出) 传递到下一层，并将其作为下一层的学习对象 (输入) 继续提取更为高阶的特征，自动地实现对输入信息本质特征的提取，深度学习模型示意图如图 2.2 所示。

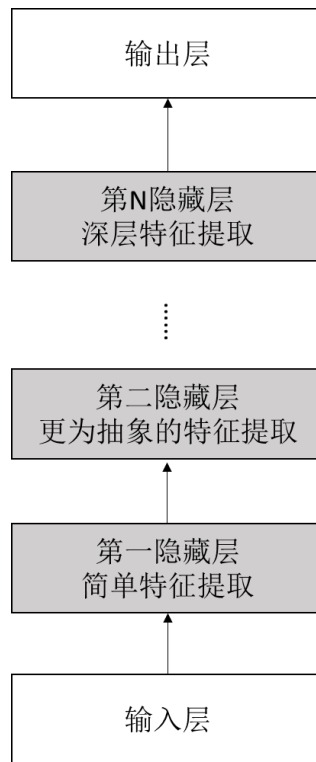


图 2.2 深度学习模型的示意图

根据深度神经网络中的信息流动方向，深度学习方法可以分为反馈深度网络 (Feed-back deep networks, FBDN)、前馈深度网络 (Feed-forward deep networks, FFDN)、和双向深度网络(Bi-directional deep networks,BDDN)。典型的前馈深度神经网络包括多层感知机(Multi layer perceptrons,MLP)、卷积神经网络(Convolutional neural networks,CNN)。典型的反馈深度网络有反卷积网络(Deconvolutional networks, DN)、层次稀疏编码网络(Hierarchical sparse coding,HSC)。典型的双向

深度网络包括深度玻尔兹曼机(Deep Boltzmann machines,DBM)、深度信念网络(Deep belief networks,DBN)、栈式自编码器(Stacked auto-encoders,SAE)^[58]。深度神经网络分类结构如图 2.3 所示。

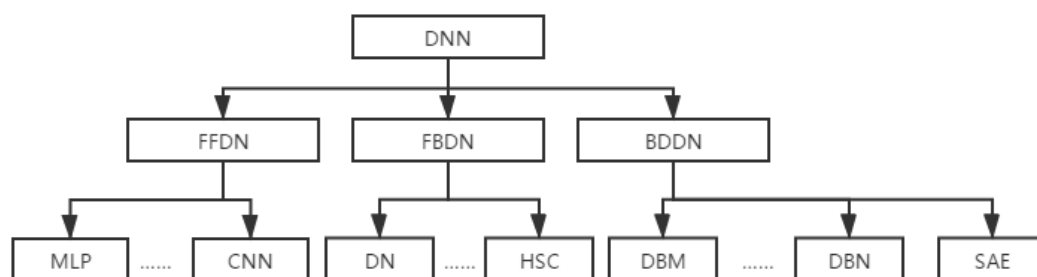


图 2.3 深度神经网络分类结构

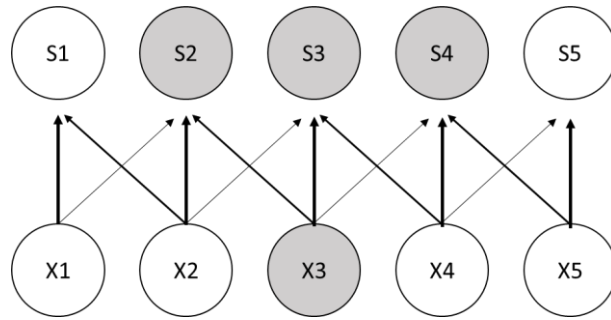
2.3 卷积神经网络 (CNN) 简介

卷积神经网络是一种常用的深度学习方法。上世纪 60 年代, Hubel 等人对猫的视觉形成机理进行了研究, 并发现了感受野 (Receptive Field) 的概念, 即通过对感受野的多次激发使得视觉信息从视网膜传递到大脑^[59]。到 80 年代, 日本科学家 Fukushima 在感受野概念的基础上提出了一个自组织的多层神经网络模型——Neocognitron。该模型通过对视觉模型进行分解, 使其转化为多层依次相连的特征平面进行处理, 即每一层的响应都由上一层的感受野激发而来。这一操作使得视觉系统模型化, 也增加视觉系统的抗干扰能力, 当物体出现轻微的变形或者移动时, 该系统也能完成识别^[59]。之后的 10 年里, 卷积神经网络一直停滞不前, 直到 1998 年, LeCun 利用反向传播算法 (BP) 对神经网络进行有监督的训练, 成功地提出了第一个卷积神经网络——LeNet-5^[24]。LeNet-5 利用彼此交替的卷积层与下采样层对原始图像进行特征的提取以及利用最后的全连接层对原始图像的特征进行分类, 最终实现手写邮政编码的识别。卷积层中的卷积核即承担了感受野的功能, 实现对局部区域信息的激发。在此之后, 越来越多的学者对卷积神经网络进行升级优化, 从而出现了一大批优秀卷积神经网络, 如 AlexNet 网络、VGGNet 网络、GoogleNet 网络以及 ResNet 网络。

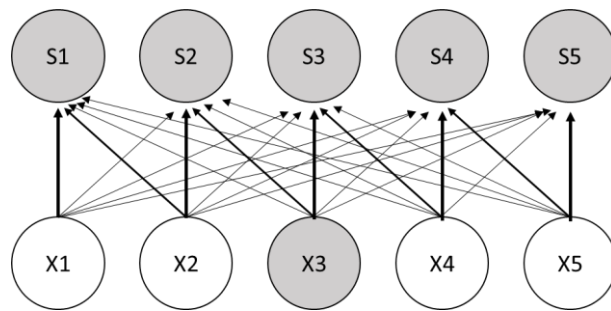
相比于其他深度神经网络, 卷积神经网络的三大核心思想为: 局部感受野、权值共享以及下采样^[60], 这三大核心思想不仅极大地简化了网络参数还提升了网络的稳定性。

(1) 局部感受野: 与传统的神经网络不一样, 卷积神经网络中的神经元仅与下一层的部分神经元连接, 这样就可以使得神经元间的权重参数大幅减少。这是因为图像的空间联系是局部的, 相距较远的像素间关联性很弱, 因此神经元无

需感受整幅图像。随着网络的逐渐深入，高层的网络综合不同的局部神经元即可获取图像的整体信息。局部感受器的实现过程如图 2.4 所示。



(a) 局部感受野

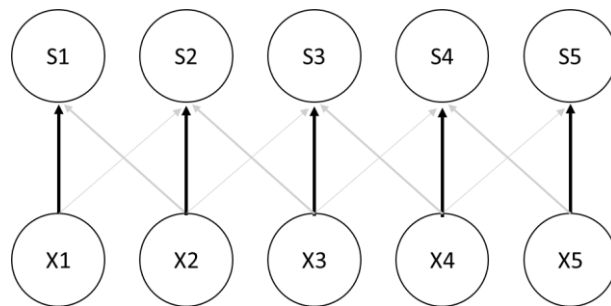


(b) 全局感受

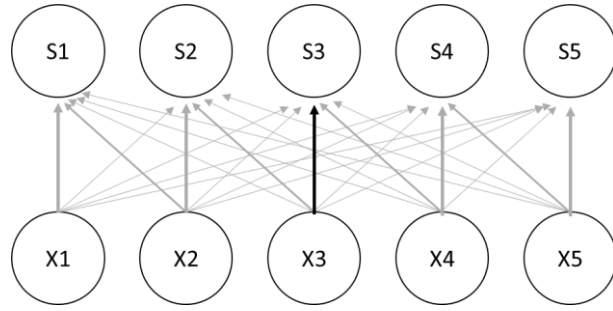
图 2.4 局部感受野

(2) 权值共享：区别于传统神经网络，卷积神经网络的多个神经元共用一个卷积核或者一个权重系数。当用同样的卷积核对图像进行卷积处理时，该层所有的神经元都可以检测到出现在图像不同位置处的相同特征，这一操作也使该网络能够在图像在轻微移动之后实现准确的识别，即获取了平移不变性。权值共享的实现过程如无权值共享模型

图 2.5 所示，图中的箭头表示模型中参数的连接方式。从 (b) 图中可以看出，黑色箭头为该模型中第三个输入与上层的连接参数，由于该模型无权值共享，因此该参数仅使用一次。而从 (a) 图中可以看出，该模型中第三个输入与上层的连接参数被用到了所有的输入与上层的连接。



(a) 权值共享模型



(b) 无权值共享模型

图 2.5 参数共享实现

(3) 下采样：下采样是指使用某一位置的相邻输出的总体统计特征来代替网络在该位置的输出，减少输出的总像素。因此可以降低过度拟合的概率，同时提升计算速度。

2.3.1 卷积神经网络

典型的卷积神经网络通常由一个输入层、若干卷积层（其中通常有非线性激活函数的介入和使用）和池化层、全连接层以及最后的输出层组成，非线性激活层通常与卷积层同时出现。典型的卷积神经网络如图 2.6 所示。

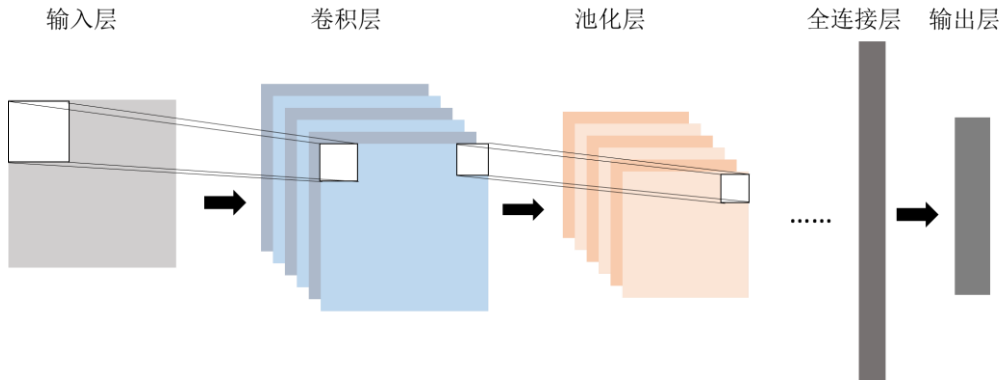


图 2.6 典型的卷积神经网络结构

(1) 卷积层

卷积层是卷积神经网络的核心，其目的在于利用卷积核进行信号特征的提取。卷积是对两个实变函数的一种数学运算，函数 x 与 w 的卷积表达式如(2.1)所示。

$$s(t) = \int x(a) \times w(t - a) da \quad (2.1)$$

卷积常采用星号表示：

$$s(t) = (x * w)(t) \quad (2.2)$$

参考式(2.1)离散形式的卷积运算可以定义为：

$$s(t) = (x * w)(t) = \sum x(a)w(t - a) \quad (2.3)$$

在实际应用中，离散形式卷积运算的输入信号一般是二维甚至多维的数组，而卷积核是特定的多维数组，其一般由学习算法优化计算得到。一个二维数组的卷积运算过程如图 2.7 所示。从图 2.7 中可以看出，离散卷积运算本质上就是矩阵的乘法，因此若一个神经网络算法中的涉及到矩阵乘法且对矩阵结构无特殊要求，则该网络就适用于卷积运算，这极大地增加了卷积神经网络在深度学习中的应用范围。

一般情况下，通常称卷积运算中的第一个参数为输入(input)；通常称第二个参数为卷积核(kernel function)；输出通常称为特征映射(feature map)。根据卷积表达式可以确定，输出中的每个元素都是由输入中每个小区域的元素加权相加得到，其中的权值则由卷积核来确定，因此卷积核在卷积中起着过滤器或特征提取的作用。卷积层具有强大的特征学习能力，一般情况下，深层次的网络能够不断迭代地从底层网络的特征中提取更高层次的特征，这样的方式在很大的程度上简化了繁琐的特征提取任务。

在任何卷积网络的实现中都有一个重要性质，那就是能够隐含地对输入用零元素进行填充(pad)使得它加宽。如果没有这个性质，宽度在每一层就会缩减。对输入进行零填充允许我们对核的宽度和输出的大小进行独立的控制。如果没有零填充，我们就被迫面临二选一的局面，要么选择网络空间宽度的快速缩减，要么选择一个小型的核——这两种策略都会极大得限制网络的表示能力。在卷积计算过程中，需要特别主要三种不同的零填充方式。MATLAB 将第一种称为有效(valid)卷积，这是一种不使用零填充的情况，同时卷积核也仅对输入中能够完整包含整个核的位置进行处理。虽然这类情况的输出更为规范，但是输出的大小在每一层都会缩减。如果输入的图像宽度是 m ，核的宽度是 k ，那么输出的宽度就会变成 $m-k+1$ 。如果卷积核非常大的话缩减率会非常显著。因为缩减数大于 0，这限制了网络中能够包含的卷积层的层数。当层数增加时，网络的空间维度最终会缩减到 1×1 ，这种情况下增加的层就不可能进行有意义的卷积。第二种特殊的情况是只进行足够的零填充来保持输出和输入具有相同的大小。在 MATLAB 的术语中，这称为相同(same)卷积。在这种情况下，只要硬件支持，网络就能包含任意多的卷积层，这是因为卷积运算不改变下一层的结构。然而，这样就会造成在输入中靠近边界的部分元素对输出的影响相对较小，进而就会导致边界中的元素出现欠表达的情况。这使得第三种极端情况产生了，在 MATLAB 中称为全(full)卷积。它进行了足够多的零填充使得每个像素在每个方向上恰好被访问了 k 次，最终输出图像的宽度为 $m+k-1$ 。各卷积类型对比如图 2.8 所示。

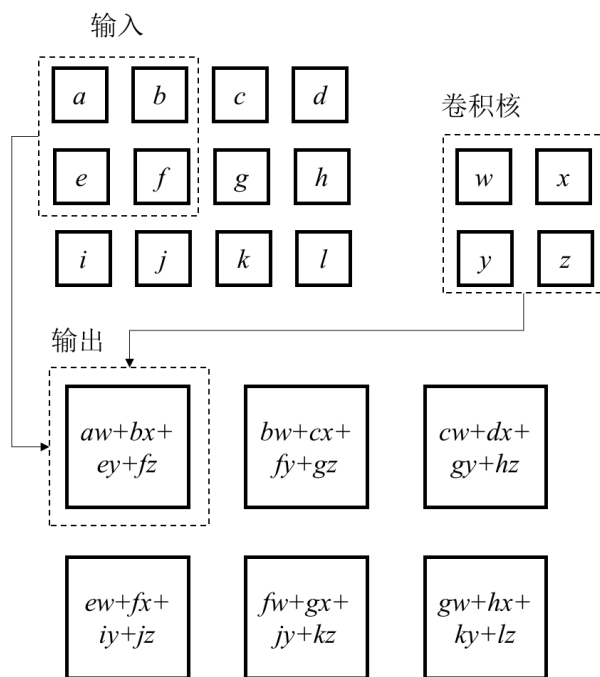


图 2.7 二维数组的卷积运算过程

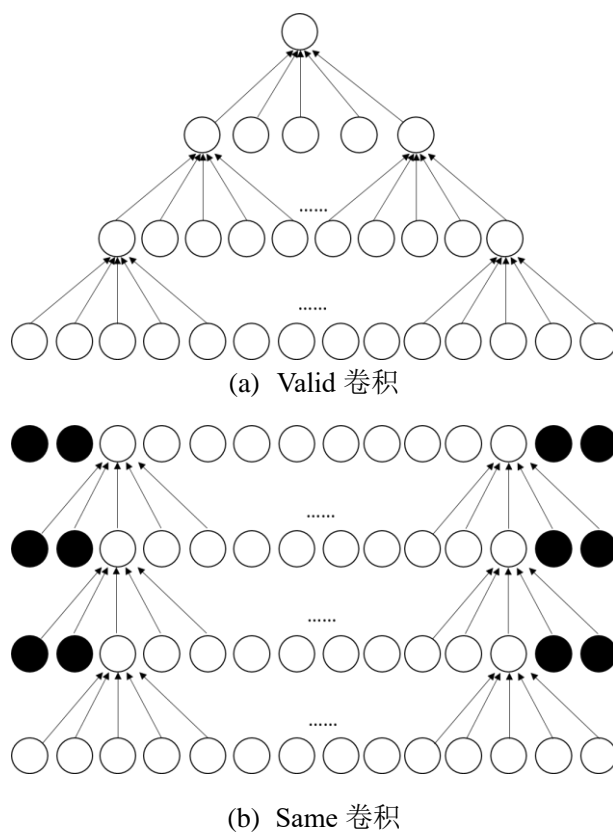


图 2.8 Valid 卷积与 Same 卷积对比

(2) 非线性变换—激活函数

为提升线性模型的表达能力，需对卷积层中得到的特征进行筛选，筛选的过

程是将卷积层中得到的特征进行非线性映射，这即需要非线性激活函数的介入。激活函数的作用机理需结合人功神经元计算模型进行阐述，人工神经元的计算模型如图 2.9 所示。

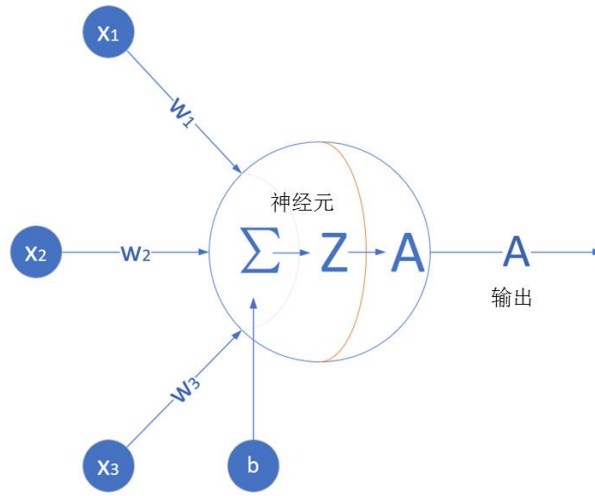


图 2.9 人工神经元计算模型

图中 X_1, X_2 和 X_3 是外界的输入信号， w_1, w_2 和 w_3 分别是三个输入信号的权重值， b 是偏置项， A 是该神经元的输出。该神经元的输出 A 如式(2.4)。

$$A = f(Z) = f\left(\sum X_i \times w_i + b\right) \quad (2.4)$$

式中 $f(Z)$ 即是激活函数，从中可以看出一个神经元作用机理首先需要对输入和权值做内积和（即卷积），其次为卷积增加偏置项，经过激活函数最终得到输出，这个输出在神经网络中又成为下一个神经元的输入。激活函数的作用就是把输入的线性数据利用自身的非线性特征映射成为非线性特征，并实现特征筛选的功能。

研究表明，深度学习网络的性能不仅与网络结构有直接的关系，而且也会受到各层神经元激活函数的影响^[62]。卷积神经网络中常见非线性函数包含 Sigmoid 函数、Tanh 函数、softsign 函数、Relu (Rectified Linear Unit)函数^[63]。接下来将对各个激活函数做详细的介绍。

(a) Sigmoid 函数

Sigmoid 函数表达式如式(2.5)所示。

$$y = \frac{1}{1 + e^{-x}} \quad (2.5)$$

Sigmoid 函数是神经网络中最早出现的一类激活函数，它是一个单调递增的函数，能够将经过神经元计算得到的输出映射到(0,1)的范围内，适合于在概率输出。其图像如图 2.10 所示，当输出 x 趋近于负无穷时，输出 y 趋近于 0；当输入

x 趋近于正无穷时，输出 y 趋近于 1。Sigmoid 函数具有连续，光滑，严格单调，输出稳定等优点，使得其成为深度学习中应用十分广泛的激活函数之一。当然，Sigmoid 函数具有非常严重的饱和性，即在曲线的两端 ($x \ll 0$ 或 $x \gg 0$) sigmoid 函数的梯度接近于 0，容易导致训练减缓，甚至出现梯度消失的现象。

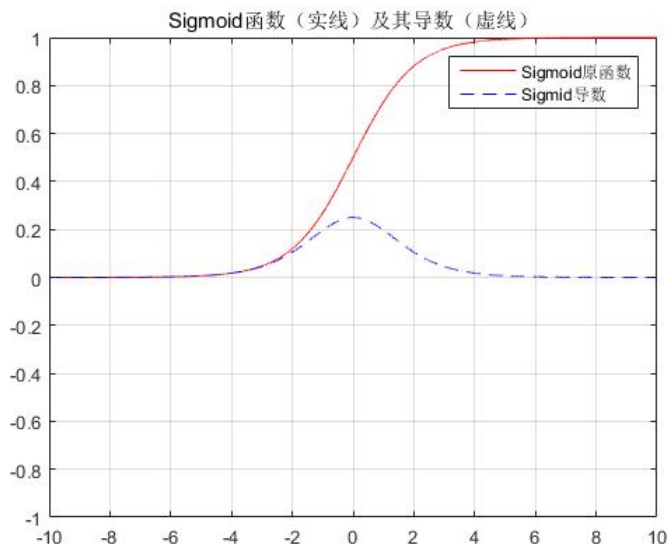


图 2.10 Sigmoid 函数图像

(b) Tanh 函数(双曲正切函数)

Tanh 函数从本质上来看是 Sigmoid 函数的变形，其表达式如式(2.6)所示

$$y = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.6)$$

Tanh 函数是一个单调递增的函数，能够将经过神经元计算得到的输出映射到(-1,1)的范围内，其图像如图 2.11 所示。Tanh 函数的输出是以 0 为中心，因此 Tanh 函数比 sigmoid 函数收敛更为迅速。但是，Tanh 函数面临着与 sigmoid 函数同样的缺点，即 Tanh 函数也具有严重的饱和性，容易出现梯度消失的现象。

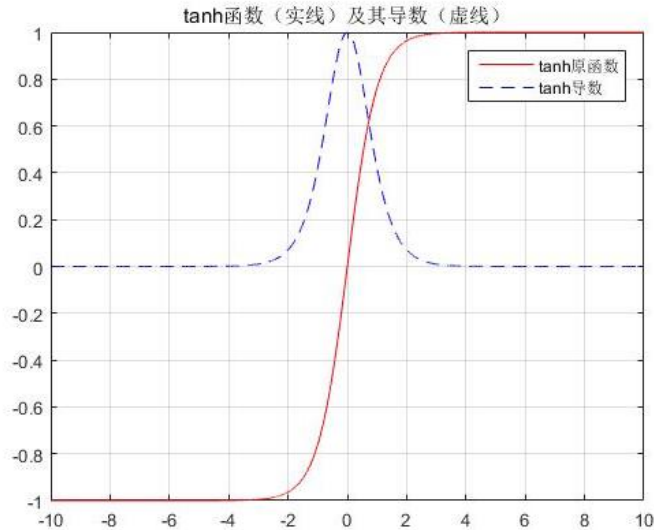


图 2.11 Tanh 函数图像

(c) Softsign 函数

Softsign 函数表达式如式(2.7)所示

$$y = \frac{x}{1+|x|} \quad (2.7)$$

Softsign 函数是 Tanh 函数的一个代替函数。它具有反对称的性质、可微分的性质，并且能够返回(-1,1)之间的值，其图像如图 2.12 所示。与 Tanh 函数相比，softsign 函数能够更好的解决梯度消失的问题。

(d) Relu 函数

Relu 函数全程为 Rectified Linear Unit，是由 Krizhesky、Hinton 等人于 2012 年提出的一种激活函数^[28]，其表达式如式(2.8)所示

$$y = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (2.8)$$

从表达式看出，Relu 函数是一个分段函数，也是一个取最大值的函数。Relu 函数有很多优点，第一 Relu 函数克服了 Sigmoid 函数的部分缺点，即不会出现正区间内梯度消失的现象；其次 Relu 函数在计算过程中仅包含线性计算，而无需指数运算，因此无论在正向传播还是在反向传播中，均具有较快的计算速度；最后，相对于 Sigmoid 函数和 Tanh 函数，Relu 函数具有较快的收敛速度。但是相对地，由于 Relu 函数在负区间中的输出为 0，在训练过程中极容易出现权重无法更新的现象，即出现部分神经元死亡的现象。

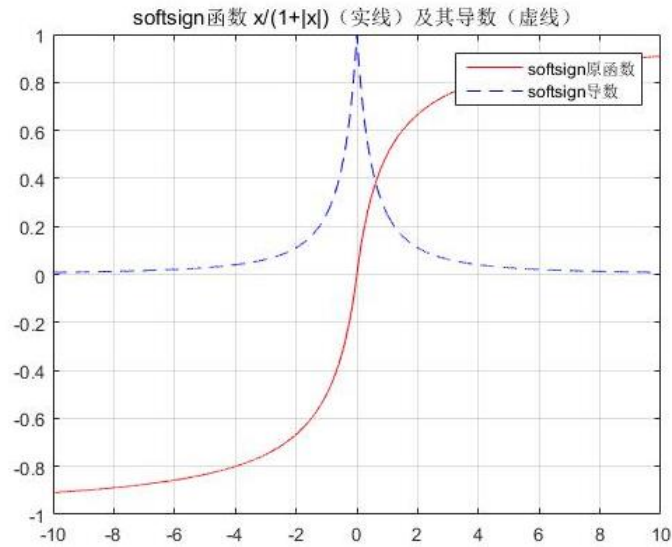


图 2.12 Softsign 函数图像

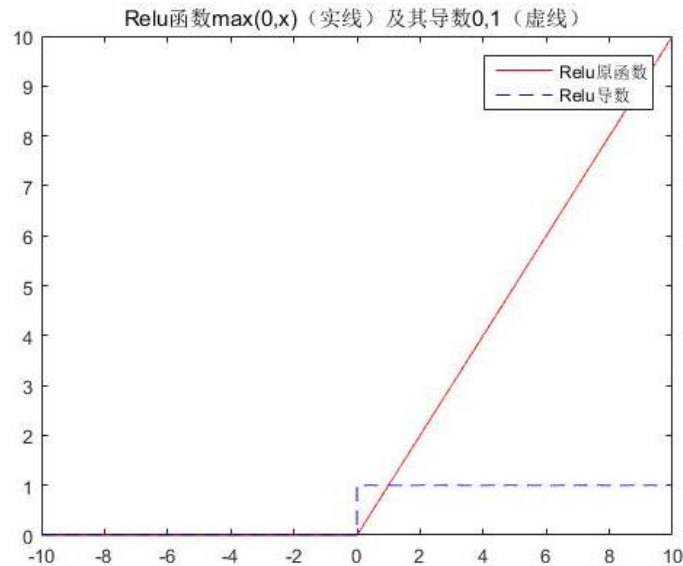


图 2.13 Relu 函数图像

(e) Leaky Relu 函数

为解决 Relu 函数可能存在的权值无法更新现象，有学者提出 Leaky Relu 函数，其表达式(2.9)所示

$$y = \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases} \quad (2.9)$$

Leaky Relu 函数图像如图 2.14 所示。Leaky Relu 函数不仅继承了 Relu 函数的很多优点，例如收敛速度快，较快的计算速度等，Leaky Relu 函数给予负输入值一个非零的斜率，这解决了 Relu 函数中极易出现的神经元死亡现象。

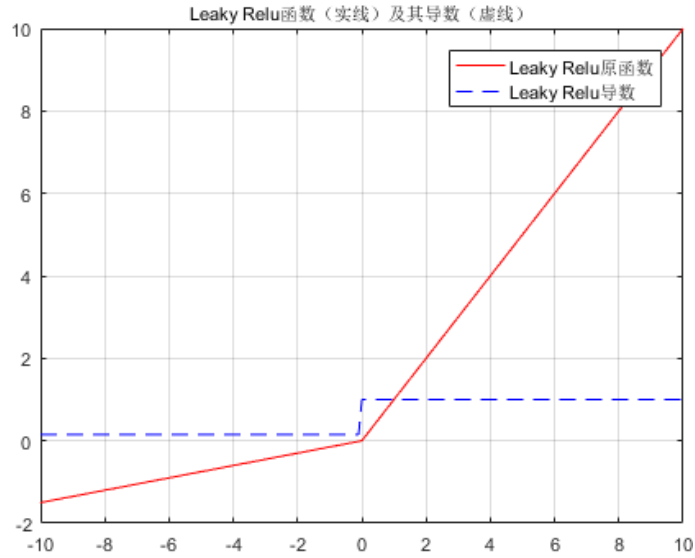


图 2.14 Leaky Relu 函数图像

(3) 下采样

下采样是卷积神经网络中的重要组成部分，又称池化（pooling），其通常紧随于卷积层之后，主要用于过滤部分冗余特征，并且能够实现减少参数数量的作用，从而防止出现过拟合的现象。同时，下采样层进一步提升了卷积神经网络的平移不变性。常见的下采样操作有两种，第一种为最大池化（Max Pooling），取每个池化窗口的最大值，其操作过程如图 2.15 所示；第二种为平均池化（Average Pooling），取每个池化窗口的平均值，其操作过程如图 2.16 所示。

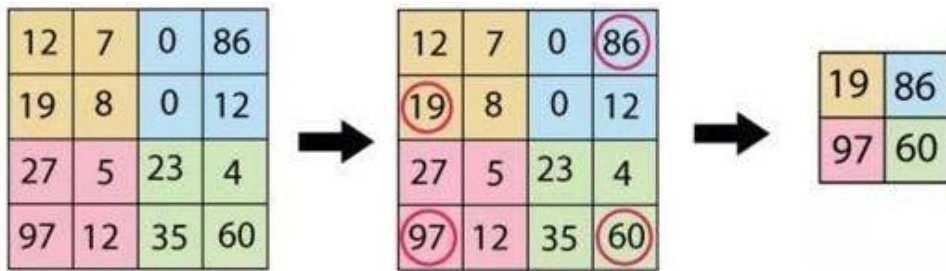


图 2.15 最大池化操作过程

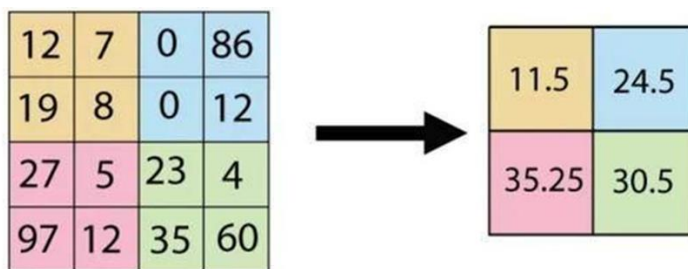


图 2.16 平均池化操作过程

2.3.2 学习算法

深度学习算法是机器学习的一个分支,1997年,Mitchell给出了机器学习的简单定义“对于某类任务 T 和性能度量 P,一个计算机程序被认为可以从经验 E 中学习是指,通过经验 E 改进后,它在任务 T 上由性能度量 P 衡量的性能有所提升^{[64]-[65]}。”在机器学习中,任务 T 通常情况下定义为机器学习系统应该如何处理样本。样本是指我们从某些希望机器学习系统处理的对象或事件中收集到的已经量化的特征(feature)的集合^{[64]-[65]}。常见的机器学习任务包括:分类、回归、转录、机器翻译以及异常检测等等。为评估机器学习的能力,需要对其性能设计量化的度量 P,对于不同的任务,需要建立与之相适应的度量 P。一般情况下,常采用准确率来度量机器的性能,而准确率是指该模型输出正确结果的样本比率。机器学习可以大致分类为无监督(unsupervised)算法、监督(supervised)算法和结合以上两种算法的半监督学习(semi-supervised learning)。

在监督学习算法(supervised learning algorithm)中,参与训练的数据集中的每条样本不仅包含一个输入,还包含一个输出值(一般是输入的特征,称之为标签),监督学习算法通过对数据集中的输入和标签进行不断地学习和训练,得到一个最优的模型,形成输入与输出之间的映射,从而产生一个推断的功能,并对全新的无标签的样本进行特征映射和判断^[66]。监督学习算法的步骤为:1.确定样本的类型;2.获取训练集;3.确定输入样本的特征如何表示;4.确定相应的学习算法;5.在训练集上运行学习算法;6.评估模型的准确性。常见的监督学习算法包括:线性回归(linear regression)、逻辑回归(logistic regression)、支持向量机(Support Vector Machines)以及决策树(decision trees)。

无监督学习算法(unsupervised learning algorithm)通过对含有很多特征的数据集进行训练来获取该数据集中有用的结构性质^[67]。常用的无监督学习算法主要有三种:聚类、关联和降维。在无监督学习中,应用最多、使用范围最广的是聚类分析,聚类的目的就是将相似的样本聚成一个组,这个组中的所有样本都具有一个或多个同样的特征,聚类分析如图 2.17 所示。常见的聚类算法包括 K 均值聚类、层次聚类、t-SNE(t-distributed stochastic neighbor embedding)聚类、DBSCAN(Density-Based Spatial Clustering of Applications with Noise)聚类算法。降维也是无监督学习中的重要组成部分,其主要思想是在保证数据特有特征的基础上,对数据进行分解,从而实现数据转化为低维的目的。常见的降维算法包括主成分分析、NMF(Non-negative Matrix Factorization,非负矩阵分解)等方法。

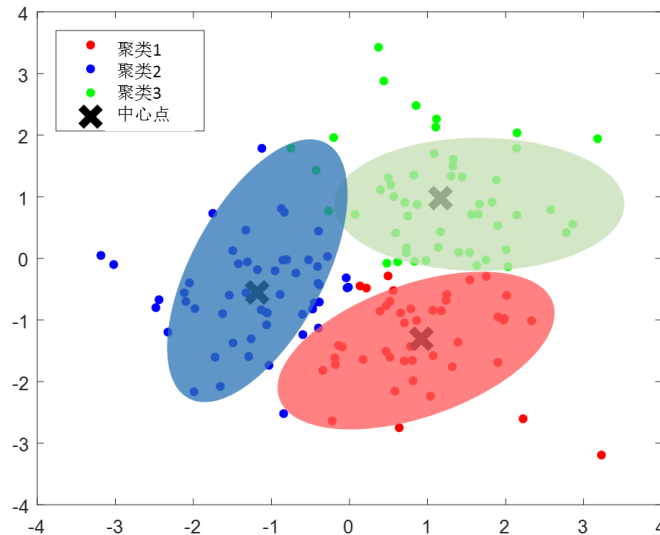


图 2.17 聚类分析示意图

半监督学习算法 (semi-supervised learning algorithm) 将无监督学习和监督学习的特点进行了完美的融合。在半监督学习算法中, 训练集中的大部分样本不含有标签, 而其中的小部分样本带有标签。

监督学习和无监督学习在很多方面有很大的区别, 从样本层面来看, 监督学习的数据集可以分为训练集和测试集。通过对训练集中的输入与相应的标签进行学习, 得到最优的映射函数, 并用测试中的数据样本进行测试。而无监督学习的样本只需要一组数据, 只需要分析数据集本身, 不需要提前设定标签。有监督学习的核心是分类, 确定每个样本的分类, 而无监督的核心是聚类, 通过对样本的特征分析, 将样本划分成由类似对象组成的类^[68]。针对监督学习和无监督学习的选择过程如图 2.18 所示。

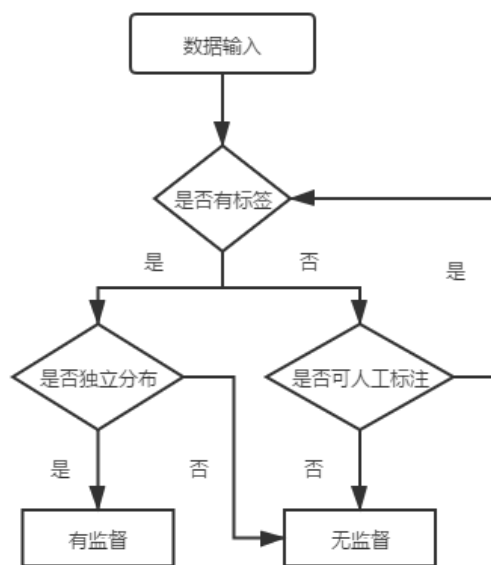


图 2.18 监督学习与无监督学习选择

2.3.3 反向传播算法

反向传播算法（Back Propagation, BP 算法）即误差反向传播算法，是一种常见的训练神经网络的方法，通常与优化方法相互配合使用。该算法最早由 David E. Rumelhart 和 James L. McClelland 等于 1986 年提出^[69]。它的基本思想可概括为如下三步：(1)信号的正向传播；(2)计算每一层的误差；(3)更新参数。接下来以三层神经网络对反向传播算法进行解释，如图 2.19 所示。

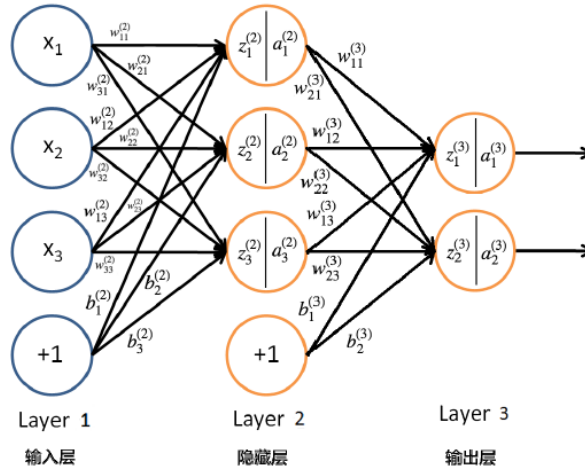


图 2.19 三层神经网络实例

对各记号含义解释如下：

n_l : 表示第 l 层神经元的个数；

f : 表示神经元的激活函数；

$W^{(l)}$: 表示第 $l-1$ 层到第 l 层的权重矩阵；

$w_{ij}^{(l)}$: 是权重矩阵 $W^{(l)}$ 中的元素，表示第 $l-1$ 层中第 j 个神经元到第 l 层中第 i 个神经元的连接权重；

$b^{(l)} = (b_1^{(l)}, b_2^{(l)}, \dots, b_{n_l}^{(l)})$: 表示第 $l-1$ 层到第 l 层的偏置；

$z^{(l)} = (z_1^{(l)}, z_2^{(l)}, \dots, z_{n_l}^{(l)})$: 表示第 l 层神经元的状态；

$a^{(l)} = (a_1^{(l)}, a_2^{(l)}, \dots, a_{n_l}^{(l)})$: 表示第 l 层神经元的输出值；

(1) 正向传播

如图 2.19 所示，该网络中第 2 层神经元的状态及激活值计算如式(2.10)-(2.15)所示。

$$z_1^{(2)} = w_{11}^{(2)} x_1 + w_{12}^{(2)} x_2 + w_{13}^{(2)} x_3 + b_1^{(2)} \quad (2.10)$$

$$z_2^{(2)} = w_{21}^{(2)} x_1 + w_{22}^{(2)} x_2 + w_{23}^{(2)} x_3 + b_2^{(2)} \quad (2.11)$$

$$z_3^{(2)} = w_{31}^{(2)} x_1 + w_{32}^{(2)} x_2 + w_{33}^{(2)} x_3 + b_3^{(2)} \quad (2.12)$$

$$a_1^{(2)} = f(z_1^{(2)}) \quad (2.13)$$

$$a_2^{(2)} = f(z_2^{(2)}) \quad (2.14)$$

$$a_3^{(2)} = f(z_3^{(2)}) \quad (2.15)$$

总结可得，第 l ($2 \ll l \leq L$) 层神经元的状态及激活值以向量的形式表达如式(2.16)和(2.17)所示。

$$z^{(l)} = W^{(l)} a^{(l-1)} + b^{(l)} \quad (2.16)$$

$$a^{(l)} = f(z^{(l)}) \quad (2.17)$$

因此，对于 L 层神经网络而言，信息的前向传递过程如式(2.18)所示。

$$x = a^{(1)} \rightarrow z^{(2)} \rightarrow a^{(2)} \rightarrow \dots \rightarrow a^{(L-1)} \rightarrow z^{(L)} \rightarrow a^{(L)} = y \quad (2.18)$$

(2) 误差反向传播

当训练样本从输入出发，经过各隐藏层得到输出之后，需要以神经网络中得到的输出与真实的输出（即标签）之间的损失为基础，对神经网络中的各个参数进行优化，度量这种损失的工具即为损失函数，也称之为代价函数或 lost 函数。对于分类问题而言，常见的代价函数包括二次代价函数和交叉熵代价函数。接下来以二次代价函数为例，对反向传播算法进行介绍。对某一个训练样本 $(x^{(i)}, y^{(i)})$ 来说，二次代价函数的表达式如式(2.19)所示

$$\begin{aligned} E_{(i)} &= \frac{1}{2} \|y^{(i)} - o^{(i)}\|^2 \\ &= \frac{1}{2} \sum_{k=1}^{n_L} (y_k^{(i)} - o_k^{(i)})^2 \end{aligned} \quad (2.19)$$

式中各字母含义如下所示：

$y^{(i)} = (y_1^{(i)}, y_2^{(i)}, \dots, y_{n_L}^{(i)})$ ：表示第 i 个训练样本的期望输出向量（标签），该向量是一个 n_L 维的向量。

$o^{(i)}$ ：表示神经网络对输入 $x^{(i)}$ 产生的实际输出向量。

则 N 个训练样本的总体（平均）代价函数可写为：

$$E_{total} = \frac{1}{N} \sum_{i=1}^N E_{(i)} \quad (2.20)$$

如图 2.19 所示，该网络中， $n_L = 2$ ， $y^{(i)} = (y_1^{(i)}, y_2^{(i)})$ 。则展开到隐藏层可得代价函数为

$$\begin{aligned}
 E_{(i)} &= \frac{1}{2}((y_1^{(i)} - f(z_1^{(3)}))^2 + (y_2^{(i)} - f(z_2^{(3)}))^2) \\
 &= \frac{1}{2}(y_1^{(i)} - f(w_{11}^{(3)}a_1^{(2)} + w_{12}^{(3)}a_2^{(2)} + w_{13}^{(3)}a_3^{(2)} + b_1^{(3)}))^2 \\
 &\quad + \frac{1}{2}(y_2^{(i)} - f(w_{21}^{(3)}a_1^{(2)} + w_{22}^{(3)}a_2^{(2)} + w_{23}^{(3)}a_3^{(2)} + b_2^{(3)}))^2
 \end{aligned} \tag{2.21}$$

根据式(2.21)可以看出，代价函数仅与权重矩阵和偏置向量有关，因此为使总体代价最小，需要从总体代价出发，对每一层的权重和偏置量进行逐层地优化，即误差的反向传播。接下来将其分为输出层和隐藏层的参数更新对反向传播算法进行介绍，为简单起见，去掉 $E_{(i)}$ 的下标，直接记为 E 。

(a) 输出层的权重参数更新

根据求导的链式法则，对式(2.21)中的输出层的权值参数求偏导可得：

$$\begin{aligned}
 \frac{\partial E}{\partial w_{11}^{(3)}} &= \frac{1}{2} \times 2(y_1 - a_1^{(3)}) \left(-\frac{\partial a_1^{(3)}}{\partial w_{11}^{(3)}}\right) \\
 &= -(y_1 - a_1^{(3)}) f'(z_1^{(3)}) \frac{\partial z_1^{(3)}}{\partial w_{11}^{(3)}} \\
 &= -(y_1 - a_1^{(3)}) f'(z_1^{(3)}) a_1^{(2)}
 \end{aligned} \tag{2.22}$$

做如下定义

$$\begin{aligned}
 \delta_i^{(l)} &= \frac{\partial E}{\partial z_i^{(l)}} \\
 &= -(y_1 - a_1^{(3)}) f'(z_1^{(3)})
 \end{aligned} \tag{2.23}$$

则式(2.22)可以变换为

$$\begin{aligned}
 \frac{\partial E}{\partial w_{11}^{(3)}} &= \frac{\partial E}{\partial z_1^{(3)}} \frac{\partial z_1^{(3)}}{\partial w_{11}^{(3)}} \\
 &= \delta_1^{(3)} a_1^{(2)}
 \end{aligned} \tag{2.24}$$

推广到一般情况，假设神经网络共 L 层，则：

$$\begin{aligned}
 \delta_i^{(L)} &= -(y_i - a_i^{(L)}) f'(z_i^{(L)}) (1 \leq i \leq n_L) \\
 \frac{\partial E}{\partial w_{ij}^{(L)}} &= \delta_{ij}^{(L)} a_j^{(L-1)} (1 \leq i \leq n_L, 1 \leq j \leq n_{L-1})
 \end{aligned} \tag{2.25}$$

将式(2.25)化为矩阵形式，则：

$$\begin{aligned}
 \delta^{(L)} &= -(y - a^{(L)}) \odot f'(z^{(L)}) \\
 \nabla_{w^{(L)}} E &= \delta^{(L)} (a^{(L-1)})^T
 \end{aligned} \tag{2.26}$$

式中 \odot 表示 Hadamard 积，其运算规则为把对应位置的元素分别相乘。如

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \odot \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} a_{11}b_{11} & a_{12}b_{12} \\ a_{21}b_{21} & a_{22}b_{22} \end{pmatrix} \quad (2.27)$$

(b) 隐藏层的权重参数更新

利用 $\delta_i^{(l)}$ 的定义可得：

$$\begin{aligned} \frac{\partial E}{\partial w_{ij}^{(l)}} &= \frac{\partial E}{\partial z_i^{(l)}} \frac{\partial z_i^{(l)}}{\partial w_{ij}^{(l)}} \\ &= \delta_i^{(l)} \frac{\partial z_i^{(l)}}{\partial w_{ij}^{(l)}} \\ &= \delta_i^{(l)} a_j^{(l-1)} \end{aligned} \quad (2.28)$$

对 $\delta_i^{(l)}, 2 \leq l \leq L-1$ 推导如下

$$\begin{aligned} \delta_i^{(l)} &= \frac{\partial E}{\partial z_i^{(l)}} \\ &= \sum_{j=1}^{n_{l+1}} \frac{\partial E}{\partial z_j^{(l+1)}} \frac{\partial z_j^{(l+1)}}{\partial z_i^{(l)}} \\ &= \sum_{j=1}^{n_{l+1}} \delta_j^{(l+1)} \frac{\partial z_j^{(l+1)}}{\partial z_i^{(l)}} \end{aligned} \quad (2.29)$$

又因为

$$z_j^{(l+1)} = \sum_{i=1}^{n_l} w_{ji}^{(l+1)} a_i^{(l)} + b_j^{(l+1)} = \sum_{i=1}^{n_l} w_{ji}^{(l+1)} f(z_i^{(l)}) + b_j^{(l+1)} \quad (2.30)$$

求偏导得：

$$\begin{aligned} \frac{\partial z_j^{(l+1)}}{\partial z_i^{(l)}} &= \frac{\partial z_j^{(l+1)}}{\partial a_i^{(l)}} \frac{\partial a_i^{(l)}}{\partial z_i^{(l)}} \\ &= w_{ji}^{(l+1)} f'(z_i^{(l)}) \end{aligned} \quad (2.31)$$

结合式(2.29)和式(2.31)可得：

$$\delta_i^{(l)} = \sum_{j=1}^{n_{l+1}} (\delta_j^{(l+1)} w_{ji}^{(l+1)}) f'(z_i^{(l)}) \quad (2.32)$$

式(2.32)可以利用第 $l+1$ 层的 $\delta^{(l+1)}$ 计算第 l 层的 $\delta^{(l)}$ ，是反向传播算法的核心。将其写成矩阵形式为：

$$\delta^{(l)} = ((W^{(l+1)})^T \delta^{(l+1)}) \odot f'(z^{(l)}) \quad (2.33)$$

(c) 输出层和隐藏层的偏置参数更新

对偏置参数求偏导可得：

$$\begin{aligned}\frac{\partial E}{\partial b_i^{(l)}} &= \frac{\partial E}{\partial z_i^{(l)}} \frac{\partial z_i^{(l)}}{b_i^{(l)}} \\ &= \delta_i^{(l)}\end{aligned}\quad (2.34)$$

写成矩阵形式则为：

$$\nabla_{b^{(l)}} E = \delta^l \quad (2.35)$$

上面三部分将反向传播算法的推导过程进行详细的介绍，可以总结出四个核心公式：

$$\delta_i^{(L)} = -(y_i - a_i^{(L)}) f'(z_i^{(L)}) \quad (2.36)$$

$$\delta_i^{(l)} = \sum_{j=1}^{n_{l+1}} (\delta_j^{(l+1)} w_{ji}^{(l+1)}) f'(z_i^{(l)}) \quad (2.37)$$

$$\frac{\partial E}{\partial w_{ij}^{(L)}} = \delta_j^{(L)} a_j^{L-1} \quad (2.38)$$

$$\frac{\partial E}{\partial b_i^{(l)}} = \delta_i^{(l)} \quad (2.39)$$

反向传播算法是利用式(2.36)-(2.39)四个公式求代价函数对参数的偏导的过程，其具体步骤如下：

- (1) 初始化权值矩阵和偏置向量；
- (2) 利用“正向传播”公式，如式(2.16)和(2.17)所示，计算每层的状态和激活值；
- (3) 利用式(2.36)计算输出层的 $\delta^{(L)}$ ；
- (4) 利用式(2.37)从第 L-1 层到第 2 层依次计算隐藏层的 $\delta^{(l)}$ ；
- (5) 按式(2.38)和式(2.39)求代价函数对参数的偏导数。

2.3.4 梯度下降算法

反向传播算法在计算过程中，为实现代价函数最小化的目的，需要对神经网络中的诸多参数进行优化，从而达到一个最优解，那么优化算法的选择则尤为重要。梯度下降算法(Gradient Descent Optimization)是目前使用最为广泛的优化算法，除此之外，梯度下降法还延伸出很多变式。下文将对梯度下降算法的原理及其各种变式的迭代原理做简单阐述。

在数学中，梯度是指某一可微的函数在该点处的方向导数沿着该方向取得最大值，对于一个二元函数 $f(x, y)$ ，该函数的梯度记为 ∇f 。梯度下降法背后的基本

原理为函数增长最快的方向就是梯度的方向，反之，则函数减小最快的方向势必是梯度的反方向，因此，对于深度学习网络中的最小化优化问题来说，权值参数和偏置需要沿着与梯度方向的相反方向前进一定的步长，即可以实现代价函数的最快下降。

假设梯度下降法的目标函数为 $J(\theta)$ ，则梯度下降法关于参数 θ 的更新式如下：

$$\theta = \theta - \eta \nabla_{\theta} J(\theta) \quad (2.40)$$

式中 η 表示学习率。

根据梯度下降法计算过程中采用数据量的不同，可以分为采用在整个训练集的数据进行计算的批量梯度下降算法 (Batch Gradient Descent)，该算法会占用大量的内存；采用训练集中的一个训练样本计算的随机梯度下降算法 (Stochastic Gradient Descent)，该算法可能会导致收敛速度过慢；采用训练集中一个小批量样本计算小批量梯度下降算法 (Mini-batch Gradient Descent)，该算法结合上述两种算法的优点，因此是目前最常用的梯度下降算法^[70]。梯度下降算法面临着如下两点问题：第一，对于负责的非线性结构，梯度下降算法极易收敛于局部最优；第二，需要人为设定学习率，过小的学习率会导致收敛速度过慢，而过大的学习率则训练震荡，出现发散的情况。鉴于以上两点问题，学者们基于梯度下降算法提出了很多优化算法^[71]。主要的改进思路有如下两点，一引入动量的概念，通过对指向方向相同维度的参数进行不断累积的更新从而实现加速收敛速度的目的；二给优化算法给予自适应改变学习率的功能，对不同的参数自适应地选取不同的学习率。梯度下降算法的改进路径如图 2.20 所示。

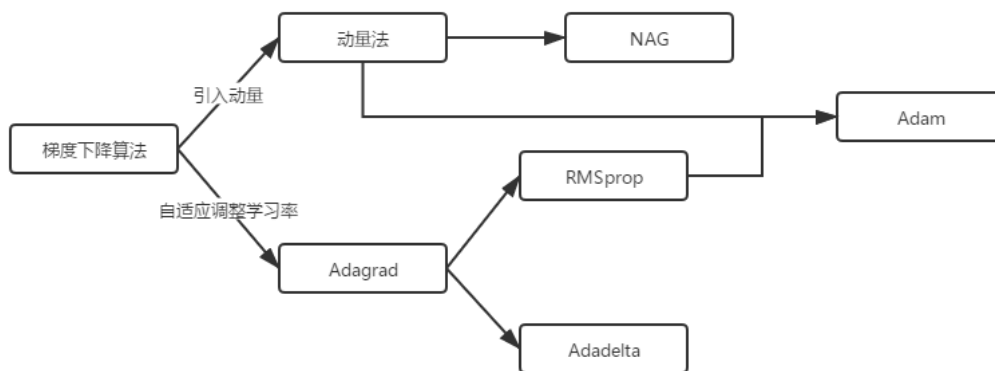
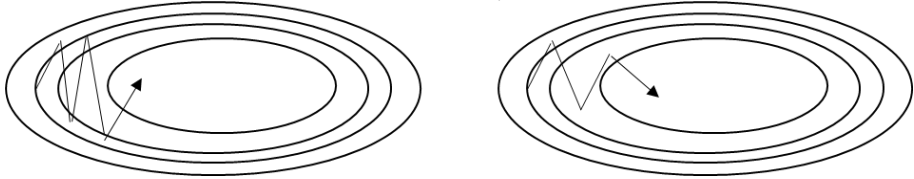


图 2.20 梯度下降算法改进路径

(1) 动量法(Momentum)

梯度下降法能够解决很多常规问题，但是在某些函数区域中，如表面曲线在一个维度上比在另一个维度上要陡峭得多。在这些情况下，梯度下降法在沟谷的斜坡上震荡，而只在底部向局部最优方向缓慢前进，因此引入动量 Momentum 来

加速梯度下降法的收敛速度。动量法在更新式引入与上一时刻步长相关的项，使得指向方向相同维度的参数进行不断累积的更新，而约束了梯度方向不一致变化的维度参数，从而实现加速收敛的作用，动量法作用机理如所示，其迭代公式如式(2.41)所示。

$$\begin{aligned} v_t &= \gamma v_{t-1} + \eta \cdot \nabla_{\theta} J(\theta) \\ \theta &= \theta - v_t \end{aligned} \quad (2.41)$$


(a) 梯度下降法
(b) 动量法

图 2.21 梯度下降法和动量法收敛对比

式中， γ 为动量因子参数，常取值 0.9。

(2) Nesterov accelerated gradient (NAG)

NAG 在动量法的基础上，引入了预测的功能。在计算参数梯度时，从代价函数中减去上一时刻的动量项，从而实现预估下一个参数的目的。NAG 的作用机理如所示。其迭代公式如式(2.42)所示。

$$\begin{aligned} v_t &= \gamma v_{t-1} + \eta \cdot \nabla_{\theta} J(\theta - \gamma v_{t-1}) \\ \theta &= \theta - v_t \end{aligned} \quad (2.42)$$

式中 γ 为动量因子参数，与动量法类似，常取值 0.9。

(3) Adagrad

Adagrad 算法能够自适应地调节学习率，对不同的参数自适应地选取不同的学习率。对具有较大偏导数的参数选用较大的学习率，而对于那些较小偏导数的参数选用较小的学习率。为实现此目的，Adagrad 算法引入累积平方梯度，接下来会对 Adagrad 算法如何实现自适应调节学习率进行分析。

令 $g_{t,i}$ 为在第 t 步迭代下参数 θ_i 的梯度，则：

$$g_{t,i} = \nabla_{\theta} J(\theta_{t,i}) \quad (2.43)$$

Adagrad 算法考虑到历史累积梯度的信息，则更新量为：

$$\theta_{t+1,i} = \theta_{t,i} - \frac{\eta}{\sqrt{G_{t,i} + \varepsilon}} \cdot g_{t,i} \quad (2.44)$$

式中

$G_{t,i}$ 是一个对角矩阵，其对角元素表示历史梯度的平方和；

ε 是平滑因子，它通常约为 $1e^{-8}$ 。

因此，Adagrad 算法的迭代公式的向量表示为

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{G_t + \varepsilon}} \odot g_t \quad (2.45)$$

由于 Adagrad 算法将历史梯度的平方和作为分母，会导致学习率在训练过程中不断地减小。

(4) RMSprop (root mean square prop)

为解决 Adagrad 算法学习率不断下降的问题，Hinton 提出 RMSprop 算法。不同于 Adagrad 算法将所有的历史梯度平方和进行累积，RMSprop 算法通过增加衰减系数 ρ 来控制历史梯度平方和累积的量。历史梯度平方和累积量为：

$$E[g^2]_t = \rho E[g^2]_{t-1} + (1 - \rho) g_t^2 \quad (2.46)$$

式中 $E[g^2]_t$ 表示添加衰减系数后的迭代次数 t 之前历史梯度平方和。用 $E[g^2]_t$ 将 Adagrad 算法中的 G_t 代替，即为 RMSprop 的迭代公式：

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{E[g^2]_t + \varepsilon}} \cdot g_t \quad (2.47)$$

Hinton 建议 ρ 设置为 0.9。

(4) Adam (Adaptive Moment Estimation)

Adam 算法^[72]将 RMSprop 算法和 Momentum 算法相结合。它除了会对衰减的历史梯度平方进行累积，而且会对衰减的历史梯度进行累积。两种不同累计值分别为：

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (2.48)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (2.49)$$

式中

m_t 为梯度的一阶矩估计；

v_t 为梯度的二阶矩估计；

β_1 为一阶矩估计的指数衰减率，常取 0.9；

β_2 为二阶矩估计的指数衰减率，常取 0.99。

由于 m_t 和 v_t 的初值均设置为 0，会使得其在迭代过程中向初值的偏移，因此需要对其做偏置校正。

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \quad (2.50)$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (2.51)$$

再对参数进行更新，可保证迭代过程较为平稳。

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t + \varepsilon}} \hat{m}_t \quad (2.52)$$

尽管梯度下降算法已经得到大力的发展和改进，但是后续的诸多算法也存在各自的问题，例如，Adam 算法可能会导致不收敛等问题。因此，在深度学习网络中需要结合实际，对优化算法进行筛选。

2.4 卷积神经网络设计

上文已对深度学习中的技术基础等进行了详细的阐述，在此基础上，下文将对基于深度学习的装配式混凝土结构缺陷识别方法进行介绍，并对其中的核心算法——卷积神经网络的拓扑结构及关键函数进行介绍和比选。

2.4.1 卷积神经网络拓扑结构

卷积神经网络已发展出多种不同的结构，典型的包括 LeNet-5、AlexNet、Facenet 等。其中 LeNet-5 以简洁的结构、高效的计算效率等优势极大地影响了后人的研究。本节将以此网络为基础，参考 Deep Learn Toolbox 中的卷积神经网络建立基于深度学习的装配式混凝土结构缺陷识别方法中采用的卷积神经网络结构。

LeNet-5 由 Le Cun 等人于 1989 年提出，主要用于手写邮政编码的识别。该网络是一个简洁、高效的卷积神经网络，为后面的诸多网络奠定了基础。Deep Learn Toolbox 是 Matlab 中应用最为广泛的深度学习工具包，该工具包中的卷积神经网络（CNN）由 Rasmus Berg Palm 等人在 LeNet-5 的基础上搭建而成，该网络由 1 个输入层、2 个卷积层、2 个池化层和 1 个全连接输出层组成，Deep Learn Toolbox 中 CNN 网结构图如图 2.22 所示。并在 MNIST 数据集的识别测试中得到了 1.22% 的错误率，略微领先于 DBN 和 SAE^[74]。

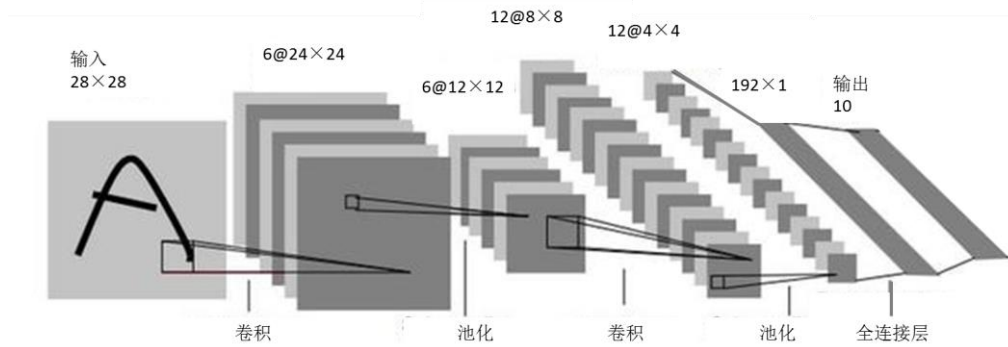


图 2.22 Deep Learn Toolbox 中 CNN 网络结构图

MNIST 数据集是一个最常见、也是最为入门级别的计算机视觉数据集，它由 70000 张带标签的手写数字图像组成，其中 60,000 张为训练集，剩余 10000 张图像组成测试集，70000 张图像均为像素大小为 28×28 的灰度图片。该数据集已被广泛用作机器学习算法的标准检验数据集，MNIST 数据集中的典型图像如图 2.23 所示。

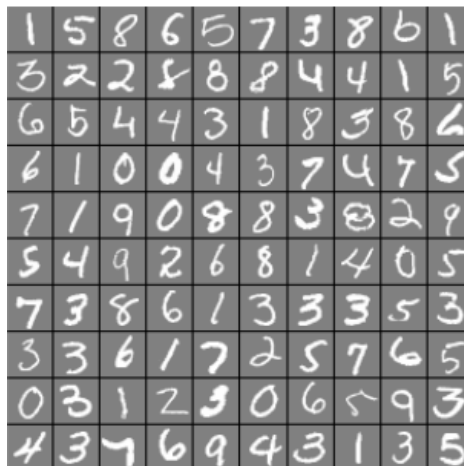


图 2.23 MNIST 数据集中的典型图像

在卷积神经网络的设计中，随着网络层数的增加，卷积核的尺寸应该随之而减小，这主要有两个方面的原因：第一从输出特征层面来看，对于层次越深的层来说，其输出的特征图将会越小，过大的卷积核会导致难以提取数据的高纬度特征；第二从计算性能层面来看，过大的卷积核会导致权重参数过多，进而影响计算速度。因此本文以 LeNet-5 的网络结构为基础，参考 Deep Learn Toolbox 中的 CNN 网络结构，设计基于深度学习的装配式混凝土结构缺陷识别方法中的卷积神经网络，其结构如图 2.24 所示。

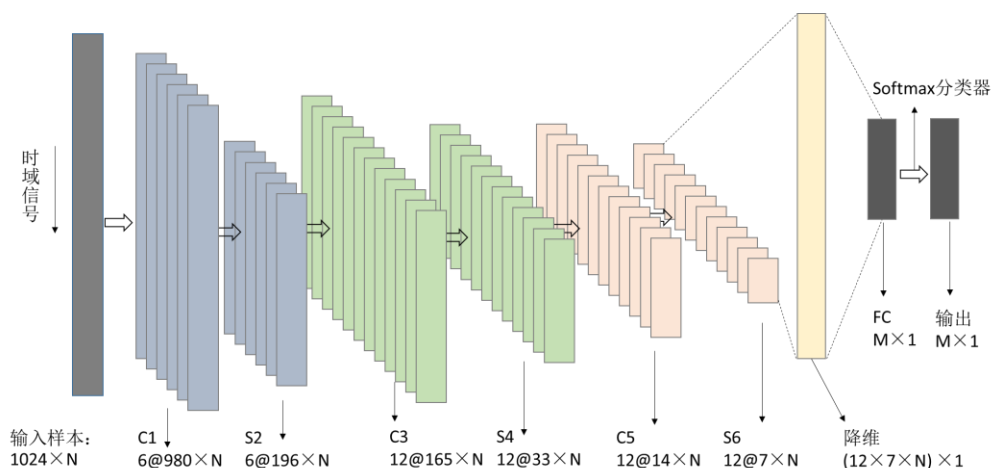


图 2.24 本文 CNN 基础框架

图中 C 表示卷积层；S 表示池化层；FC 表示全连接层；N 表示输入样本中的测点数；M 表示输出的节点数（M 和 N 的具体数值需结合实际问题确定）。从图 2.24 中可以看出，该 CNN 结构由 1 个输入层、3 个交替出现的卷积层和池化层、1 个全连接层、1 个输出层组成。该网络利用卷积层和池化层逐层对加速度响应信号进行特征提取，最后利用全连接层中的 Softmax 分类器计算缺陷概率作为网络的输出向量，并基于输出的缺陷概率完成装配式混凝土结构的缺陷识别。卷积网络结构中各层的详细参数汇总如表 2.1 所示，网络中各层的含义解释如下。

（1）输入层

输入层以加速度时程信号作为样本输入，每个样本均为 $1024 \times N$ 的矩阵（矩阵中 1024 表示加速度时程的信号数量，N 为信号测点的个数）。为降低 CNN 在训练过程中的不稳定性 and 加快收敛速度，加速度时程信号在输入 CNN 之前采用了幅值归一化的前处理。

（2）C1 卷积层

C1 为第一个卷积层，本 CNN 结构中 C1 层包含 6 个卷积核，每个卷积核的大小都是 45，卷积步长为 1，因此每个卷积核都可以计算得到一个大小为 $980 \times N$ 的特征映射。一个输入样本在经过 C1 卷积层的卷积计算之后共得到 6 个 $980 \times N$ 的特征映射。

（3）S2 池化层

S2 为第一池化层，池化的尺寸为 5×1 。池化方式采用平均池化的方法来降低 C1 得到的 6 个特征映射的大小，C1 卷积层计算得到的输出经过 S2 池化层的平均池化之后共得到 6 个 $196 \times N$ 的特征映射；这样即保证了特征映射不会因为输入的小方差变化而发生大规模改变，同时也极大地提升了网络的计算效率。

（4）C3 卷积层

C3 卷积层是第二个卷积层，包含了 12 个卷积核，每个卷积核的大小都是

32, 卷积步长为 1, S2 池化层和 C3 卷积层之间的各节点采用全连接的方式。因此 S2 池化层的输出在经过 C3 卷积层的卷积计算之后可以得到 12 个 $165 \times N$ 的特征映射。

(5) S4 池化层

S4 是第二池化层, 池化尺寸为 5×1 , 与 S2 池化层一样采用平均池化的方法来降低 C3 得到的 12 个特征映射的大小。C3 卷积层中得到的输出经过 S4 池化层的平均池化之后共得到 12 个 $33 \times N$ 的特征映射。

(5) C5 卷积层

C5 卷积层是第三个卷积层, 包含了 12 个卷积核, 每个卷积核的大小为 20, 卷积步长为 1, S4 池化层和 C5 卷积层之间的各节点采用全连接的方式。因此 S4 池化层的输出在经过 C5 卷积层的卷积计算之后可以得到 12 个 $14 \times N$ 的特征映射。

(6) S6 池化层

S6 是第三池化层, 池化尺寸为 2×1 , 与 S4 池化层一样采用平均池化的方法来降低 C5 得到的 12 个特征映射的大小。C5 卷积层计算得到的输出经过 S6 池化层的平均池化之后共得到 12 个 $7 \times N$ 的特征映射。

(7) FC 全连接层

FC 为全连接层, 其作用是对 S6 得到的特征映射做降维处理, FC 将 S6 池化层的特征映射中的所有元素处理成一个 M 维的列向量, 再通过对该列向量乘以权值系数并加上对应的偏置, 再利用 Softmax 分类器计算网络最后的输出, 即输出包含 M 个节点。

Softmax 分类器主要用于多分类问题, 能够将分类器前级输出单元的输出映射到 $(0,1)$ 区间内。Softmax 函数的表达式如式(2.53)所示。

$$S_i = \frac{e^{V_i}}{\sum_{i=1}^C e^{V_i}} \quad (2.53)$$

式中

S_i 为当前输入为第 i 种分类的概率;

V_i 为 Softmax 分类器前级输出单元的第 i 个输出;

e 为自然对数

C 为总的类别个数。

根据式(2.53)可知, Softmax 分类器的输出表征了不同类别之间的相对概率, 这样即可以根据概率的大小对输入样本进行分类。同时, Softmax 函数能够保证各类别的概率之和为 1, 因此, 当其中一个类别的概率增大时, 其他类别的概

率会相应地减小，故 Softmax 分类器在对分类问题中具有明显的优越性。

表 2.1 卷积神经网络结构的详细参数

编号	类型	核尺寸	核数量	步长	输入	输出
C1	卷积层	45	6	1	1024×N	6@980×N
S2	池化层	5	-	1	6@980×N	6@196×N
C3	卷积层	32	12	1	6@196×N	12@165×N
S4	池化层	5	-	1	12@165×N	12@33×N
C5	卷积层	20	12	1	12@33×N	12@14×N
S6	池化层	2	-	1	12@14×N	12@7×N
FC	全连接层	-	-	-	12@7×N	M×1

卷积神经网络的基本结构确定之后，还需要对代价函数、激活函数、优化算法以及学习率等参数进行进一步的选取，下面的章节将会基于上述 CNN 基本框架对以上参数进行对比选取。

2.4.2 代价函数选取

卷积神经网络中的代价函数是用来衡量网络输出的预测值与真实值（标签）之间差异的工具，网络的训练也就是通过优化网络中的权值和偏置等参数来实现代价函数最小的目的，代价函数也成为损伤函数、Lost 函数和 Cost 函数。对于分类问题而言，常见的代价函数包括二次代价函数(Quadratic Cost Function, QCF)和交叉熵代价函数(Cross-Entropy Cost Function, CECF)。二次代价函数和交叉熵代价函数的表达式分别如式(2.54)和式(2.55)所示。

$$E = \frac{1}{2N} \sum_{i=1}^N \|y^{(i)} - o^{(i)}\|^2 \quad (2.54)$$

$$E = -\frac{1}{N} \sum_{i=1}^N (y^{(i)} \times \ln(o^{(i)}) + (1 - y^{(i)}) \times \ln(1 - o^{(i)})) \quad (2.55)$$

式中

E 表示代价函数值；

N 表示样本总数；

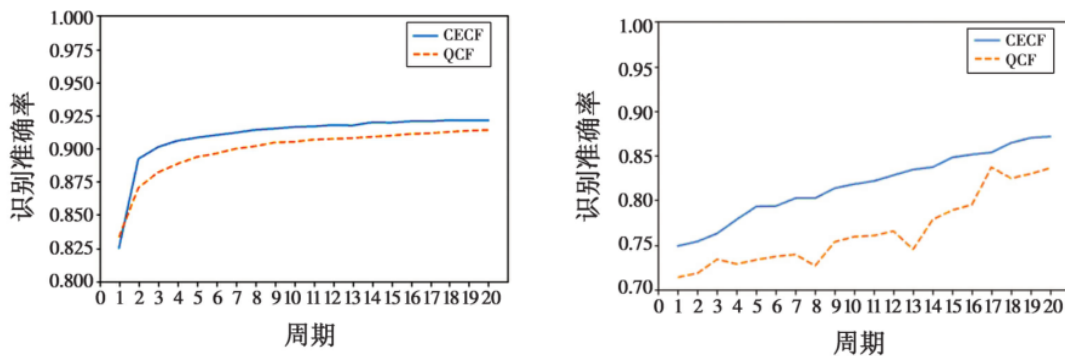
$y^{(i)}$ 表示第 i 个样本的真实输出（标签）；

$o^{(i)}$ 表示第 i 个样本的预测输出。

在 CNN 的训练过程中，当 CNN 的预测值与实际值相差较大时，为使训练过程能够更快收敛，在反向传播算法中各参数的调整幅度需要相应地增大。然而，根据 2.3.3 对反向传播算法的推导过程可以看出，二次代价函数在反向传播过程

中容易受到激活函数导数 $f'(z)$ 的影响（如式(2.23)和式(2.24)所示）。因此当使用二次代价函数时，会出现误差越大，参数的调整幅度减小，训练速度降低的现象。当使用交叉熵函数作为代价函数时，在反向传播过程中可以很好地避免这种现象的发生，很好地解决了激活函数影响更新效率的问题。关于交叉熵代价函数方向传播问题的推导见文献^[75]，此处不做详细推导。

为对比不同代价函数在 CNN 中的表现，李仲德^[76]等人基于 AlexNet 卷积神经网络，利用通用的计算机视觉数据集 MNIST 数据集和 CIFAR-10 数据集分别对二次代价函数和交叉熵代价函数在图像识别中的准确率进行了测试，对比结果如图 2.25 所示（图中蓝色线条表示交叉熵代价函数的识别准确率，红色线条表示二次代价函数的识别准确率）。



(a) MNIST 数据集识别准确率结果对比 (b) CIFAR-10 数据集识别准确率结果对比

图 2.25 二次代价函数和交叉熵代价函数图像识别准确率对比^[76]

从图 2.25 中可以看出，交叉熵代价函数不仅能有效地提高图像识别的准确率，而且训练速度更快，明显地加快了 CNN 的训练过程。

赵宏等人^[75]还利用 TensorFlow 搭建卷积神经网络，利用 notMNIST 数据集对代价函数（二次代价函数和交叉熵代价函数）和激活函数（Sigmoid 激活函数、Tanh 激活函数、ReLU 激活函数和 PReLU 激活函数）的不同组合的识别准确率进行了对比。Sigmoid 函数和 Tanh 函数作为激活函数时两种代价函数的识别准确率对比如表 2.2 所示，其余两种激活函数的识别准确率与之类似，此处不再赘述。表中结果显示无论采用何种激活函数，交叉熵代价函数的识别准确率均高于二次代价函数的识别准确率。

综上所述，交叉熵代价函数在训练效率和识别准确度两方面比二次代价函数都更为优秀，因此，本文设计的 CNN 中也将采用交叉熵代价函数。

表 2.2 不同激活函数时两种代价函数的识别准确率对比^[75]

激活函数	代价函数	迭代次数	识别准确率
Sigmoid 函数	二次代价函数	800	76.41%
	交叉熵函数	800	87.50%
	二次代价函数	1400	78.16%
	交叉熵函数	1400	85.16%
	二次代价函数	2000	77.16%
	交叉熵函数	2000	89.72%
Tanh 函数	二次代价函数	800	90.76%
	交叉熵函数	800	93.87%
	二次代价函数	1400	89.64%
	交叉熵函数	1400	92.31%
	二次代价函数	2000	93.75%
	交叉熵函数	2000	95.66%

2.4.3 激活函数选取

上文 2.3.1 节中已经对激活函数在神经网络中的作用机理、各类型激活函数及其优缺点进行了详细的阐述，但是如何在一个特定的 CNN 中选取最为适宜的激活函数依然需要结合实际情况进行考虑。因此，本节将结合前人的研究与试验的实际情况选取 CNN 中的激活函数。卷积神经网络中常见非线性函数包含 Sigmoid 函数、Tanh 函数、和 Relu (Rectified Linear Unit) 函数^[63]，因此本节将重点对这三类激活函数进行比较和选择，这三类激活函数的表达式及优缺点对比如表 2.3 所示。

为对比这三种激活函数在 CNN 中的性能表现，田娟等人^[77]基于 Caffe 卷积神经网络，就识别准确率和计算速度两个维度，利用 Mnist 数据集对常用激活函数的综合性能进行比较分析。不同激活函数的识别准确率如表 2.4 所示。

表 2.4 中结果显示，Sigmoid 函数的识别准确率最低，而 Tanh 函数和 Relu 函数的识别准确率相当，且高于 Sigmoid 函数。

表 2.3 各激活函数对比

激活函数	表达式	优点	缺点
Sigmoid 函数	$y = \frac{1}{1 + e^{-x}}$	1.连续, 光滑, 严格单调, 输出稳定。	1.由于函数的饱和性, 导致梯度消失; 2.输出不关于原点对称, 导致梯度下降权重更新时出现 z 字型的下降。
Tanh 函数	$y = \frac{e^x - e^{-x}}{e^x + e^{-x}}$	1.解决了输出原点对称问题; 2.收敛速度比 Sigmoid 函数快。	1.由于函数的饱和性, 导致梯度消失。
Relu 函数	$y = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases}$	1.解决了部分梯度消失的问题; 2.收敛速度更快。	1.容易出现部分神经元死亡现象。

表 2.4 各激活函数识别准确率^[77]

迭代次数	Sigmoid 函数	Tanh 函数	Relu 函数
1,000	97.33%	98.00%	98.04%
2,000	98.14%	98.53%	98.77%
3,000	97.91%	98.56%	98.87%
4,000	98.66%	98.95%	99.04%
5,000	98.72%	98.87%	98.95%
6,000	98.87%	99.00%	99.10%
7,000	98.86%	98.95%	99.02%
8,000	98.86%	98.95%	99.02%
9,000	98.90%	99.00%	99.13%
10,000	98.87%	99.09%	99.07%

同时, 该团队还对比了三类激活函数在信号正向传播和反向传播过程中所消耗的时间, 对比结果如表 2.5 所示

表 2.5 三类激活函数正向传播与反向传播耗时对比^[77]

	激活函数	耗时
正向传播(forward)	Sigmoid 函数	3.49694ms
	Tanh 函数	5.09306ms
	Relu 函数	0.24092ms
反向传播(backward)	Sigmoid 函数	0.06916ms
	Tanh 函数	0.07278ms
	Relu 函数	0.05238ms

表 2.5 中结果显示：在正向传播过程中，Sigmoid 函数与 Tanh 函数的耗时较多，这是因为这两类函数涉及到非线性的幂次计算，导致了计算速度较慢、耗时较长的现象。Relu 函数凭借其函数形式不涉及非线性计算，求导方便等优势在耗时方面表现出较好的性能。

综合上述两个方面可以得到，相比于 Sigmoid 函数，Tanh 函数和 Relu 函数均取得了更好的效果。因此本节接下来的内容将会结合后续装配式框架结构动力试验对激活函数做进一步选择。

随机挑选装配式框架结构动力检测试验中的工况 6 为例，将此工况下框架结构中各测点的加速度时程划分为包含 2000 个输入样本的数据集。加速度时程的前处理、数据集划分等操作将会在 3.3.2 中做详细阐述，故此处不做展开介绍。以 2.4.1 节中设计的 CNN 基本框架和 2.4.2 节中选取的交叉熵代价函数为背景，分别采用 Tanh 函数和 Relu 函数作为激活函数，训练次数达 10000 次后，对比两类激活函数的灌浆缺陷识别准确率，并以识别准确率作为依据进行激活函数的选取。工况 6 中 CNN 训练次数达 10000 后，不同学习率下两类激活函数的灌浆缺陷识别准确率如表 2.6 所示。

表 2.6 不同学习率下三类激活函数的识别准确率

学习率 \ 激活函数	0.005	0.01	0.02	0.05
Tanh 函数	76.11%	97.31%	94.08%	88.74%
Relu 函数	60.59%	70.97%	70.54%	64.4%

学习率为 0.005 时，Tanh 函数和 Relu 函数的准确率分别是 76.11% 和 60.59%；学习率为 0.01 时，Tanh 函数和 Relu 函数的准确率分别是 97.31% 和 70.97%；学习率为 0.02 时，Tanh 函数和 Relu 函数的准确率分别是 94.08% 和 70.54%；学习率为 0.05 时，Tanh 函数和 Relu 函数的准确率分别是 88.74% 和 64.4%。可以得出如下结论：在三种不同学习率的情况下，Tanh 函数和 Relu 函数都有较好的识别准确率，即都能够较好的识别装配式混凝土结构的灌浆缺陷。但是在相同学习率的情况下，Tanh 函数的识别准确率均明显地超过了 Relu 函数。因此可以判断出 Tanh 函数更为适合本文的卷积神经网络，故选取 Tanh 函数作为 CNN 中卷积层的激活函数。

2.4.4 优化算法选取

卷积神经网络的训练是指通过不断优化网络中的权重参数和偏置参数来实现代价函数最小化的目的，为了能够快速、完善地完成网络的训练过程，需要对采用的优化算法以及学习率等算法进行选择。上文 2.3.4 节中对反向传播算法中

的参数优化算法——梯度下降算法及各种变式的优化原理、参数更新公式和各自的优缺点做了详细的阐述。本节将基于这些优化算法的优缺点，并结合试验的实际情况选取 CNN 中的优化算法。

为对比不同优化算法的性能，Nitish 等人^[78]利用多种卷积神经网络，包括 (ResNet-32、DenseNet、PyramidNet 和 SENet)，并在 CIFAR-10 数据集上对梯度下降算法和 Adam 算法的性能进行测试，实验结果如图 2.26 所示 (图中蓝色线条表示梯度下降法计算得到的识别错误率，红色线条表示 Adam 算法计算得到的识别错误率)。

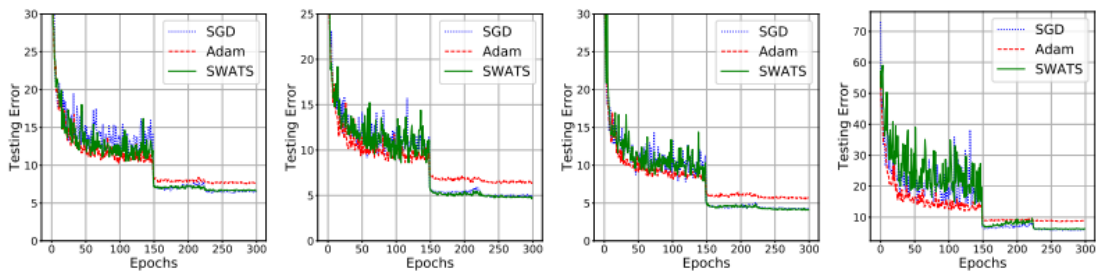


图 2.26 不同卷积神经中梯度下降算法和 Adam 算法错误率^[78]

图中结果显示虽然 Adam 算法的收敛速度比梯度下降法要快，但最终收敛的结果并没有梯度下降法好。

同时，Wilson 等人^[79]于 2017 年发表的论文也对比了多种优化算法的综合性能。他们利用 VGG 卷积神经网络，并在 CIFAR-10 上对 6 种优化算法进行对比，对比结果如图 2.27 所示。

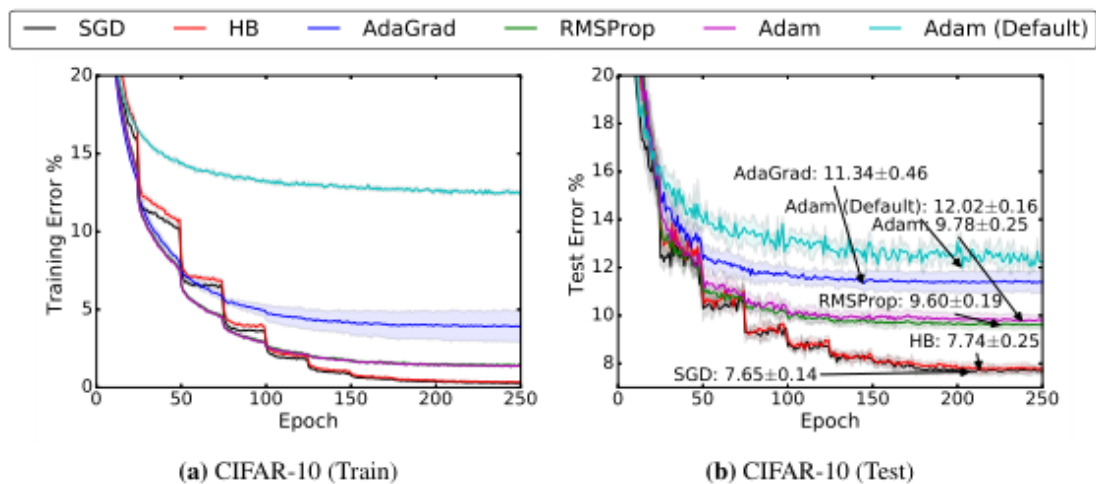


图 2.27 不同优化算法识别错误率对比^[79]

结果显示，梯度下降算法的优化结果明显优于其他的自适应算法，如 Adam

算法、RMSProp 算法等。

本节先后尝试了 2.3.4 节中的多种优化算法，结果发现几种自适应改进的优化算法在运行结果上与传统梯度下降算法较为接近。因此本文选取小批量梯度下降算法作为卷积神经网络的优化算法，根据上节中对学习率的讨论，确定本文小批量梯度下降算法中的学习率为 0.01。

2.5 基于深度学习的装配式混凝土结构缺陷识别方法

上节以经典的卷积神经网络——Deep Learn Toolbox 为基础，对其中的关键参数、函数和算法进行了对比选优，最终设计出卷积神经网络拓扑结构。本节将对基于深度学习的装配式混凝土结构缺陷识别方法中的 3 个关键要素（卷积神经网络拓扑结构、数据集搭建方法以及卷积神经网络的训练和测试）进行介绍，该识别方法的流程如图 2.28 所示。

（1）拓扑结构

该方法利用上节中建立的卷积神经网络直接提取局部动力响应中的特征，此功能通过交替出现的卷积层和池化层完成，卷积层通过沿着时间轴的卷积计算得到动力响应的特征映射，而池化层则对上层的特征映射进行统计总结，过滤部分冗余特征。将上述子结构重复 3 次得到卷积神经网络的隐式层次特征结构。网络最顶层的全连接层采用传统的分类器——Softmax 分类器，以卷积层中的特征映射作为输入，将其映射成缺陷概率作为输出，从而实现装配式混凝土结构的缺陷识别。

（2）数据集搭建与样本标签设计

数据集在基于深度学习的装配式混凝土结构缺陷识别方法中占据非常重要的地位。该方法首先需要对采集到的局部动力响应进行前处理并搭建数据集。数据集可以分为训练集和测试集两部分，训练集主要用于卷积神经网络的训练，而测试集主要用于卷积神经网络的性能评价。数据集包含大量的样本，每个样本均对应于一个能够反应该样本特征的标签。本文将缺陷概率作为样本的标签，根据卷积神经网络计算得到的样本实际输出进行装配式混凝土结构的缺陷识别。

该方法需要利用大量带标签的样本对其中的卷积神经网络进行训练。当原始采集到的数据量较少时，则需要对原始数据进行数据增强来扩充数据集。常用的数据增强方法有图像的旋转、缩放等，本文将结合试验情况进行数据增强，详细的数据增强过程会在后文进行详细阐述，此处不做赘述。

（3）卷积神经网络的训练和测试

该方法的实现有两个核心步骤：卷积神经网络的训练阶段和测试阶段。在卷

积神经网络的训练阶段，通过不断地优化网络中的参数（包括各个卷积层的权值参数和偏置参数等）使得衡量卷积神经网络实际输出与样本标签之间差值的代价函数最小。另一方面，在测试阶段，测试集中的样本会输入到经过训练的卷积神经网络中，经过一系列计算之后得到该样本的实际输出，通过检验测试集中样本的实际输出与样本标签间的差异性，即可实现装配式混凝土结构的缺陷识别。通过对测试集中所有样本的识别结果进行统计即可对该方法的性能进行评估。

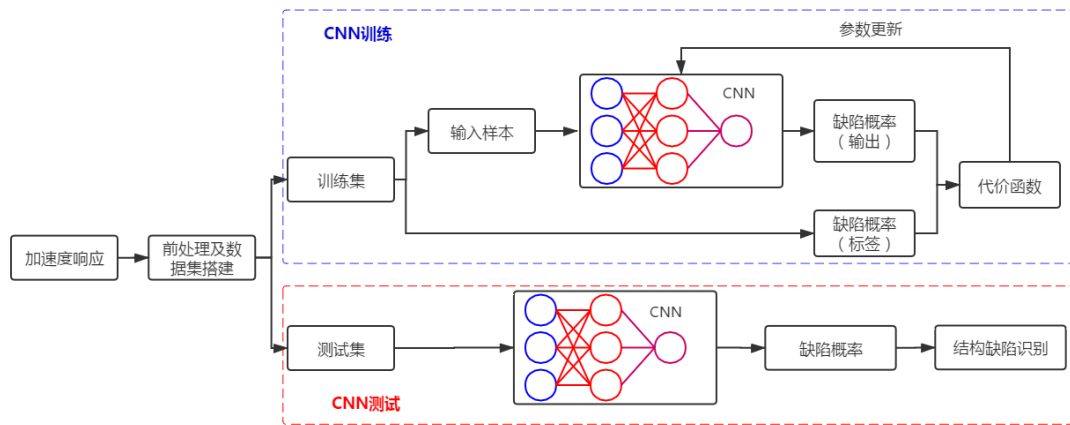


图 2.28 基于深度学习的装配式混凝土结构灌浆缺陷识别流程

2.6 本章小结

本章首先对深度学习网络的分类和其作用进行了梳理，并对其中的卷积神经网络进行了详细的讨论，为后续建立基于深度学习的装配式混凝土结构缺陷识别方法打下了理论基础。在此基础上，建立了基于深度学习的装配式混凝土结构缺陷识别方法，并重点对其中的卷积神经网络拓扑结构和关键函数等进行了比选。主要内容如下：

(1) 对卷积神经网络的基本框架进行详细讨论，并重点对卷积神经网络的基本结构、工作原理和其中的关键算法进行研究；

(2) 对卷积神经网络中的学习算法进行了讨论：对学习算法的定义和分类进行了阐述。通过对两种不同学习算法的工作原理以及常见算法的介绍，最终对两种学习算法的选择原则进行讨论；

(3) 对卷积神经网络中的反向传播算法进行了理论推导：以常见的二次代价函数为例，利用求导的链式法则对反向传播算法在误差从后往前传播的有效性进行了证明，最终推导出反向传播算法的 4 个核心公式及其工作的步骤；

(4) 对卷积神经网络中梯度下降算法的工作原理、发展路径、变体以及各自的优势进行了梳理：对梯度下降算法的工作原理进行了简单的介绍，通过查阅

文献梳理出其发展出来的各种变体，并对比了这些变体算法的性能；

(5) 建立了基于深度学习的装配式混凝土结构缺陷识别方法，该方法利用卷积神经网络直接提取装配式混凝土结构动力响应中的缺陷特征，并通过全连接层将其映射成缺陷概率作为输出，最终实现装配式混凝土结构的缺陷识别；

(6) 本文以 Deep Learn Toolbox 中的 CNN 网络结构为基础框架，并对卷积神经网络中的关键函数（代价函数、激活函数和优化算法）进行比较选优，最终建立该方法中的卷积神经网络。

第3章 装配式框架结构灌浆缺陷识别试验验证

3.1 前言

第二章建立了基于深度学习的装配式混凝土结构缺陷识别方法,为验证该方法的可行性、有效性和识别的准确性,本章将以装配式混凝土框架结构的缩尺模型作为研究对象,对该方法进行试验验证。在该装配式框架结构的预制柱中设计不同程度的套筒灌浆缺陷,进行现场激励,采集梁柱连接节点处的加速度响应。利用采集到的响应对上文中的卷积神经网络进行训练和测试,定义评价指标,对该方法的识别效果进行分析与研究。

3.2 试验概况

3.2.1 试验模型介绍

本试验采用的模型是装配式混凝土框架结构 1/2 缩尺试验模型,该模型为一横向单跨纵向双跨的 2 层装配式混凝土框架结构,平面尺寸为 1.5m×3.0m,每层层高为 1.5m,平面柱网如图 3.1 所示。

预制柱截面尺寸为 250mm×250mm,高为 1480mm;座浆层厚度为 20mm;叠合梁截面尺寸为 150mm×200mm,长度为 1250mm。基础地梁截面尺寸为 300mm×300mm,一层与二层的现浇板板厚为 70mm。各构件的详细信息汇总如表 3.1 所示。

表 3.1 各构件详情汇总

构件类型	截面尺寸	长度或高度	数量	制作方式
柱	250mm×250mm	1480mm	12	预制
梁	150mm×200mm	1250mm	12	预制加部分现浇
基础地梁	300mm×300mm	-	1	预制
板	70mm	-	2	现浇

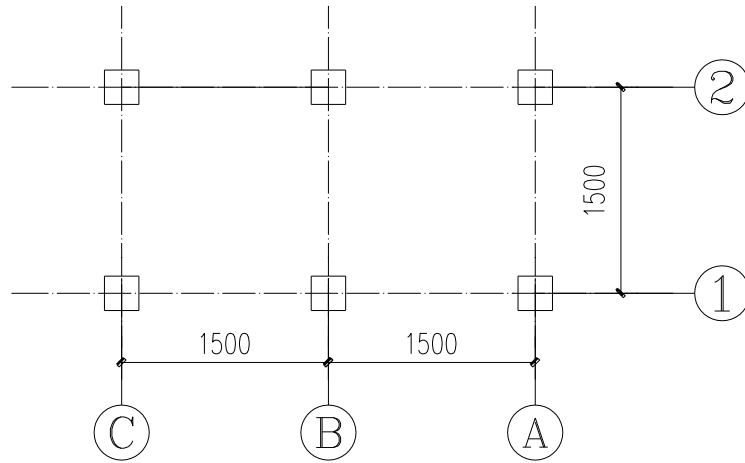
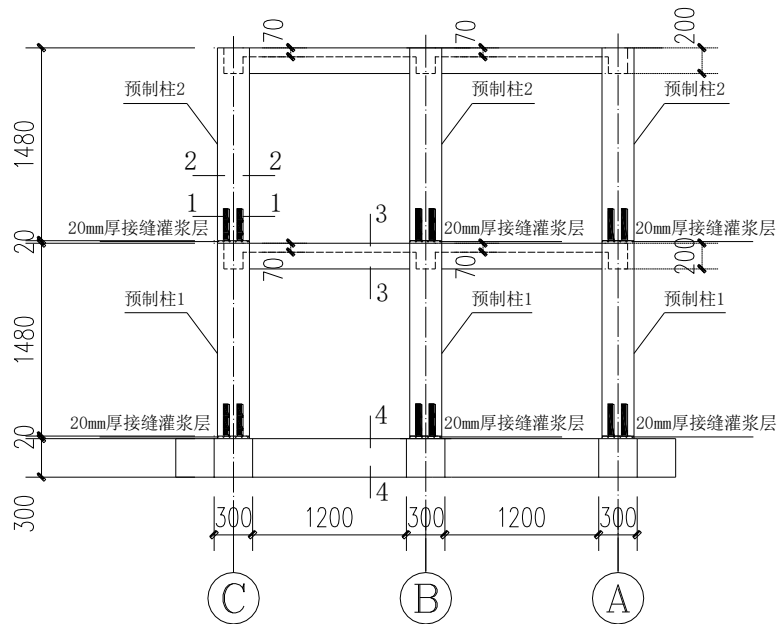


图 3.1 框架模型平面柱网图

在基础地梁与一层预制柱连接位置、一层预制柱与二层预制柱连接位置，构件的内部纵筋采用全灌浆套筒连接的连接方式，连接处均设置 20mm 厚的座浆层（接缝灌浆层），立面尺寸如图 3.2 和图 3.3 所示。



A-C轴立面图

图 3.2 框架模型 A-C 轴立面图

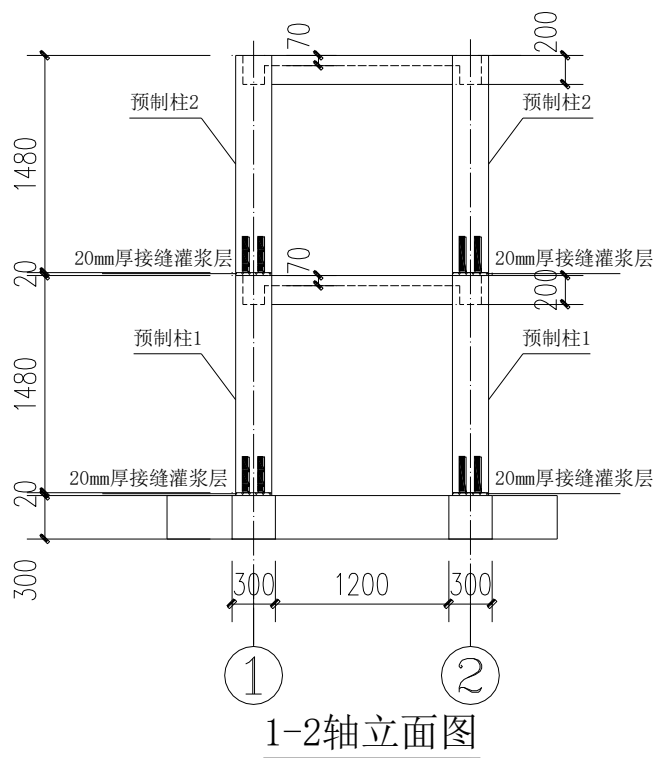


图 3.3 框架模型 1-2 轴立面图

模型中梁柱混凝土均采用 C30 混凝土，所用的纵筋及箍筋均采用 HRB400 级钢筋。

在图 3.2 中，截面 1-1 表示包含套筒段的预制柱横截面，这部分的混凝土保护层厚度为 20mm，套筒选用球墨铸铁材料制作的全灌浆套筒，由于纵筋选用的是直径为 12mm 的钢筋，参照《装配式混凝土结构技术规程》^[5]，使用的全灌浆套筒外径为 44mm，内径为 34mm，套筒总长为 250mm，下部纵筋伸入套筒长度为 110mm，上部纵筋伸入套筒长度为 120mm。截面 2-2 为不含套筒段的预制柱横截面，这部分的混凝土保护层厚度为 32mm，纵筋采用 4 根直径为 12mm 的钢筋，底筋与顶筋各两根，箍筋采用的是 8mm 的钢筋，箍筋间距加密区为 100mm，非加密区为 200mm。截面 3-3 为叠合梁的横截面，混凝土保护层厚度为 20mm，叠合梁预制部分高度为 130mm，现浇部分为 70mm，顶部纵筋采用 2 根直径为 10mm 的钢筋，腰筋采用 2 根直径为 6mm 的钢筋，底部纵筋采用 2 根直径为 10mm 的钢筋，箍筋采用的是 8mm 的钢筋，箍筋间距为 100mm。截面 4-4 为基础地梁的横截面，混凝土保护层厚度为 25mm，底部纵筋和顶部纵筋均采用 4 根直径为 22mm 的钢筋，腰筋采用 2 根直径为 12mm 的钢筋，箍筋采用的是 8mm 的钢筋，箍筋采用的是 8mm 的钢筋，箍筋间距为 100mm。4 个截面的截面图如图 3.4 所示。

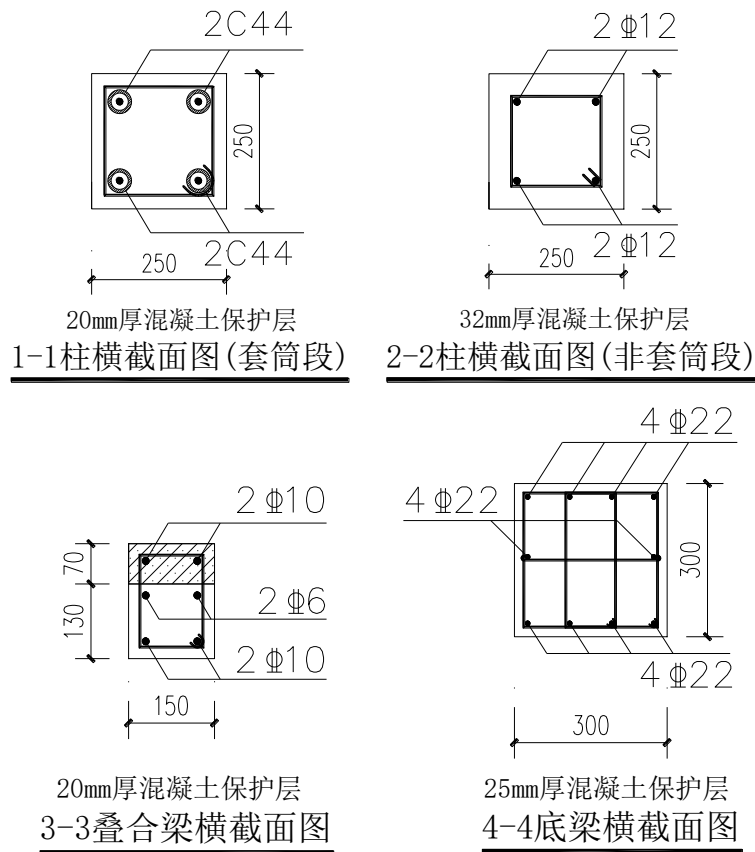


图 3.4 截面图

根据《JGJ 1-2014 装配式混凝土结构技术规程》^[5]，在靠近梁柱节点处与预制柱柱底处会有一段箍筋加密区，因此，该框架的六根预制柱将依据套筒段区域长度进行箍筋加密区与非加密区的布置，具体的深化设计如图 3.5 所示。

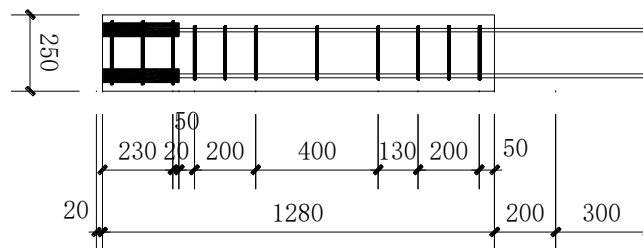


图 3.5 预制柱箍筋立面布置图

叠合梁的截面尺寸为部分的箍筋按照 100mm 的间距进行均匀布置，底部纵筋外伸 200mm，从而保证叠合梁与预制柱之间的现浇节点具有良好的受力性能。叠合梁的具体深化设计如图 3.6 所示。

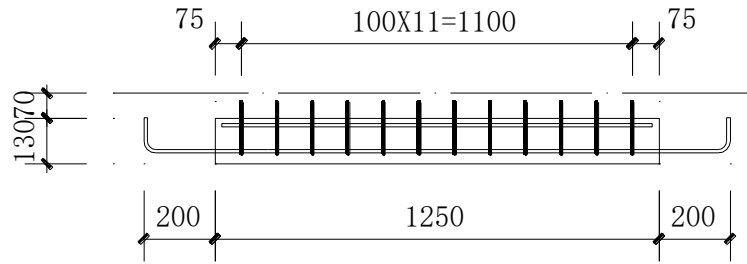


图 3.6 叠合梁钢筋立面布置图

在装配式框架结构完成整体装配过程中，需要进行楼板的现浇。在本试验中，现浇板与叠合梁的现浇部分（共 70mm）一起整体浇筑，板筋选用 6mm 直径的钢筋，板筋间距为 200mm，板的配筋方式采用的是双层双向配筋。一层板与二层板的配筋相同，如所示。

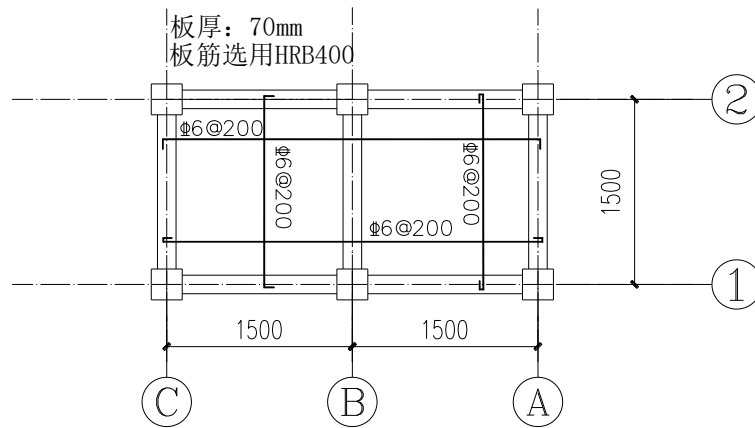


图 3.7 框架模型一层板与二层板配筋平面图

预制梁构件和预制柱构件均由专业的预制装配式构件制造工厂制作完成，预制构件的制作过程及预制构件示意图如图 3.8 所示。



(a) 预制梁钢筋及支模

(b) 预制柱钢筋及支模

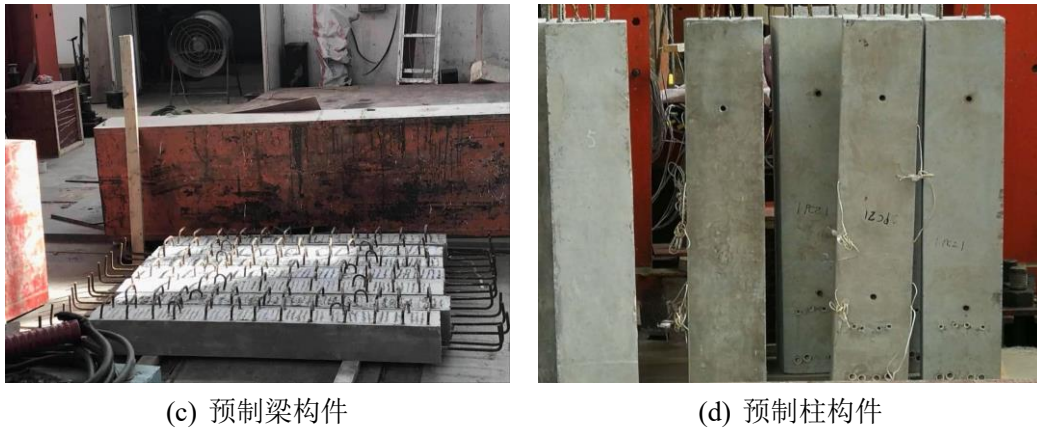


图 3.8 预制梁柱构件支模及预制构件

预制构件在工厂制作并养护之后，需要在试验室对其进行组装，构件的组装过程如图 3.9 所示。



图 3.9 试验室预制构件装配过程

由前文可知，装配式建筑中的节点连接尤为关键。在本试验中，地梁与柱以及柱与柱之间的钢筋连接全部采用全灌浆套筒的连接方式，灌浆过程如图 3.10 所示。



图 3.10 套筒灌浆过程

装配完成的框架结构实体如图 3.11 所示。



图 3.11 装配式框架结构

3.2.2 缺陷引入方式

为了避免一层梁、柱、板与节点浇筑和二层结构吊装与浇筑过程中出现结构性破坏或倒塌，一层所有预制柱内的套筒均完成灌浆料的正确灌浆施工，将存在灌浆质量缺陷的套筒布置在二层预制柱中。由于在进行灌浆施工的过程中，若注入的灌浆料不足以填充套筒与钢筋连接部分的所有间隙，待达到强度后硬化的灌浆料将会增加二次灌浆的难度^[73]。因此，该框架结构中存在灌浆质量缺陷的套筒设计为不进行灌浆施工。为了能够同步获取存在与不存在灌浆质量缺陷的套筒所在连接节点的振动响应数据，梁两侧的二层预制柱将设计成一侧存在套筒灌浆质量缺陷、另一侧不存在套筒灌浆质量缺陷。

试验模型第二层的初始缺陷引入及激振点（JZ）布置（激振点布置在跨中），如图 3.12 所示（图中空心圆点表示未灌浆的套筒）：在第二层中，1 号柱中套筒全部灌浆，2 号柱中 1 个套筒灌浆，3 号柱中套筒全部灌浆，4 号柱中 3 个套筒灌浆，5 号柱中套筒全部灌浆，6 号柱中 2 个套筒灌浆。

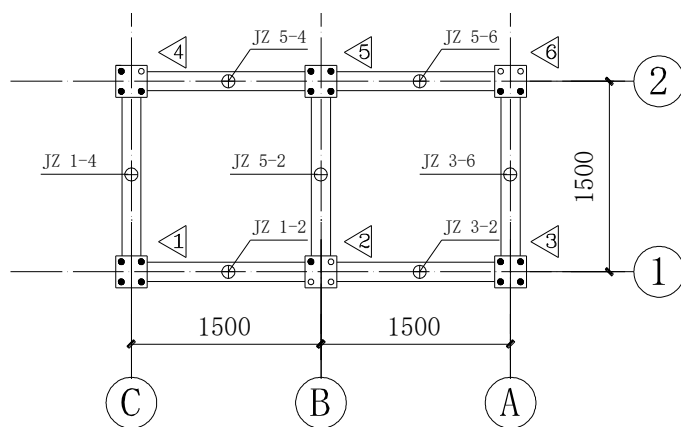


图 3.12 第二层平面缺陷引入图及激振点布置

试验模型第一层由于工厂制作加工及拼装误差（非人为引入），部分钢筋未能完成灌浆，本文仍将其视作初始缺陷引入，如图 3.13 所示（图中空心圆点表示未灌浆的套筒）：

在第一层中，由于施工原因，1 号柱中 3 个套筒灌浆，2 号柱中套筒全部灌浆，3 号柱中 2 个套筒灌浆，4 号柱中 3 个套筒灌浆，5 号柱中 3 个套筒灌浆，6 号柱中套筒全部灌浆。

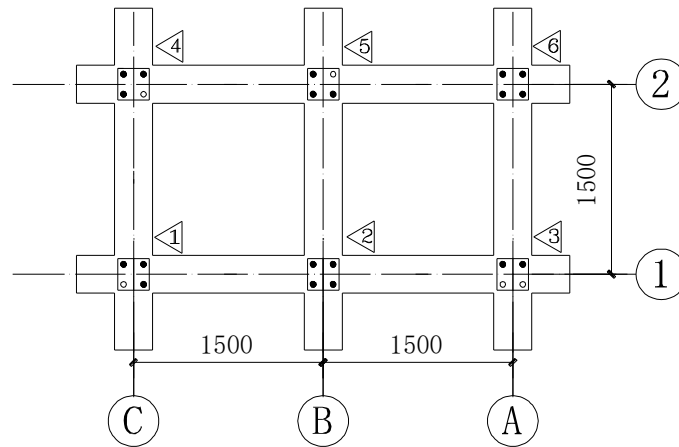


图 3.13 第一层平面缺陷分布图

框架结构中具体缺陷引入汇总如表 3.2 所示，表中 0 表示该柱所有套筒均已灌浆，1 表示该柱有一个套筒未灌浆，2 表示该柱有两个套筒没灌浆，依此类推。


表 3.2 缺陷引入汇总表

未灌浆套筒数目	1号柱	2号柱	3号柱	4号柱	5号柱	6号柱
第二层	0	3	0	1	0	2
第一层	1	0	2	1	1	0

3.2.3 激振方法与试验仪器

此次试验采用激振器激振，测点响应数据的采集采用加速度传感器。信号源、功率放大器、激振器均采用扬州科动提供的仪器，其参数如表 3.3 所示，加速度计由同济大学试验室提供；数据采集系统采用北京东方振动和噪声技术研究所配套的采集系统，采样频率为 1024Hz。

表 3.3 激振仪器参数

信号源	型号: KD5602				
工作状态	输出模式	失真度	显示	频率调整	频率范围
手动改变频率或自动扫频, 上下限频率可设定	正弦、对数、线性	<2% ± 0.1dB	数显电压和频率	手动调频或自动扫频	10Hz~20KHz
功率放大器	型号: KD5702				
额定输出功率	额定输出电压	额定输出电流	工作频率	信噪比	配用激振器
200W	14V _{RMS}	15A	20Hz~10KHz	>80dB	KDJ-10、20
激振器	型号: KDJ-20				
最大激振力	力常数	最大振幅	频率范围	最大电流	电流电阻
200N	14N/A	±5mm	DC~2KHz	14A	0.5Ω

在本次试验中, 为了使结构响应达到最大, 提高信噪比, 先对结构进行扫频激振并做傅里叶变换分析, 以确定各测点响应最大时地激振频率, 采用仪器峰值激振力 200N 进行激振, 结构扫频试验自谱分析如图 3.14 所示, 由扫频试验自谱图可以发现, 当激振频率为 182Hz 时, 各个测点均有最大响应, 各测点加速度时程曲线的能量也主要分布在 182Hz 的频带附近, 故采样频率选为 1024Hz 是足够精确的, 能够涵盖 0~512Hz 频带的响应。为了使激振器激振过程中结构达到较大的响应, 激振器激振频率选择为 182Hz, 并采用正弦激励。

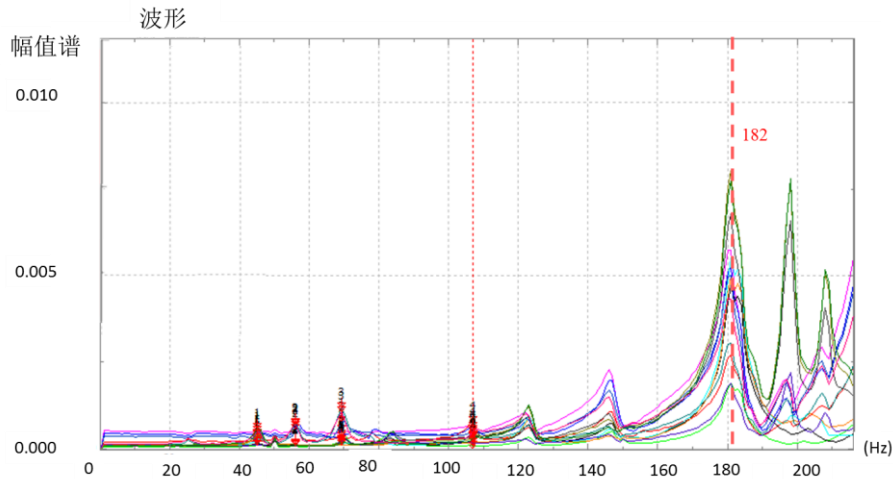


图 3.14 装配式框架结构扫频试验自谱分析

3.2.4 各工况激振点及测点布置

根据试验模型的边界条件和灌浆套筒的缺陷程度,本次试验共设计了 7 种不同的工况,分别对应于图 3.12 中的 7 个激振点。在每个工况中,激振点均位于梁的中点处,每次激振均会同时采集激振点两侧连接节点(叠合梁与预制柱的连接节点)处的加速度响应。例如对于工况 1 来说,激振点为 JZ 1-2,每次激振需要同时采集柱 1(无缺陷柱)、柱 2(缺陷柱)以及柱 1 与柱 2 中间连接梁上的结构加速度响应。7 个工况的划分详情如表 3.4 所示。

表 3.4 工况划分详情

工况	激振点	无缺陷柱编号	缺陷柱编号(未灌浆套筒个数)
工况 1	JZ 1-2	1	2(3)
工况 2	JZ 3-2	3	2(3)
工况 3	JZ 5-4	5	4(1)
工况 4	JZ 5-6	5	6(2)
工况 5	JZ 1-4	1	4(1)
工况 6	JZ 5-2	5	2(3)
工况 7	JZ 3-6	4	6(2)

每个工况均布置 14 个测点,这 14 个测点均匀、对称地分布在激振点两侧的预制柱(包括无缺陷柱以及缺陷柱)和叠合梁上,即激振点的每侧都布置 7 个测点,编号较小的测点(编号 1-7)布置在无灌浆缺陷的连接节点处,编号较大的测点(编号 8-14)布置在存在灌浆缺陷连接节点处。对于布置在无灌浆缺陷节点的 7 个测点来说,其中 3 个测点位于激振点所在的梁上,布置间距为 150mm,与激振点距离最近的测点距激振点为 175mm;剩余 4 个测点均匀分布于预制柱

上, 布置间距为 320mm。以工况 5 为例对激振点与测点的布置与编号进行说明, 如图 3.15 所示。其余各工况的测点和激振点布置方式等均与工况 5 一致, 在此不展开赘述。图中预制柱上的加速度传感器测量方向均平行于地面, 叠合梁上的加速度传感器测量方向均垂直于地面。试验所用加速度传感器和数据采集仪均由同济大学试验室提供, 其灵敏度等参数如表 3.5 所示。

表 3.5 各测点加速度传感器参数

	测点号	标定值 (mV/m/s ²)	增益	量程 (±m/s ²)	输入耦合	
预制柱 1	1	62.89	100	1.59	电压 DC	 加速度传感器
	2	62.11	100	1.61	电压 DC	
	3	70.74	100	1.41	电压 DC	
	4	64.94	100	1.54	电压 DC	
	5	71.53	100	1.4	电压 DC	
	6	71.74	100	1.39	电压 DC	
	7	73.15	100	1.37	电压 DC	
预制柱 4	8	73.15	100	1.36	电压 DC	 数据采集仪
	9	71.58	100	1.4	电压 DC	
	10	72.67	100	1.38	电压 DC	
	11	57.8	100	1.73	电压 DC	
	12	60.98	100	1.64	电压 DC	
	13	59.52	100	1.68	电压 DC	
	14	76.05	100	1.31	电压 DC	

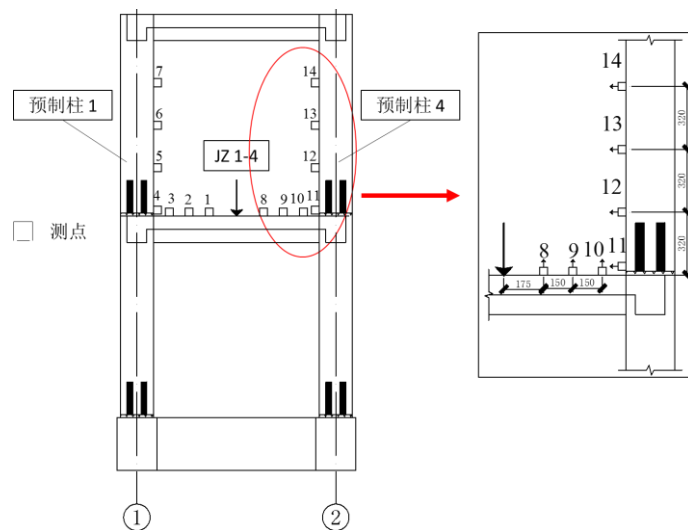


图 3.15 工况 5 中激振点及测点布置与编号立面图

3.2.5 试验过程

试验过程分为三个步骤：

- (1) 在激振点布置激振器；
- (2) 在激振点两侧的预制柱和叠合梁上的测点处布置加速度传感器；
- (3) 激振并同时采集各测点处的加速度时程，每个工况的采集时长均为 60s，每个工况中每个测点处均会采集到 61440 个加速度信号，因此每个工况都会采集到 14 条加速度时程，每条时程均包含 61440 个加速度信号。

以工况 1 为例，试验过程如图 3.16 所示。

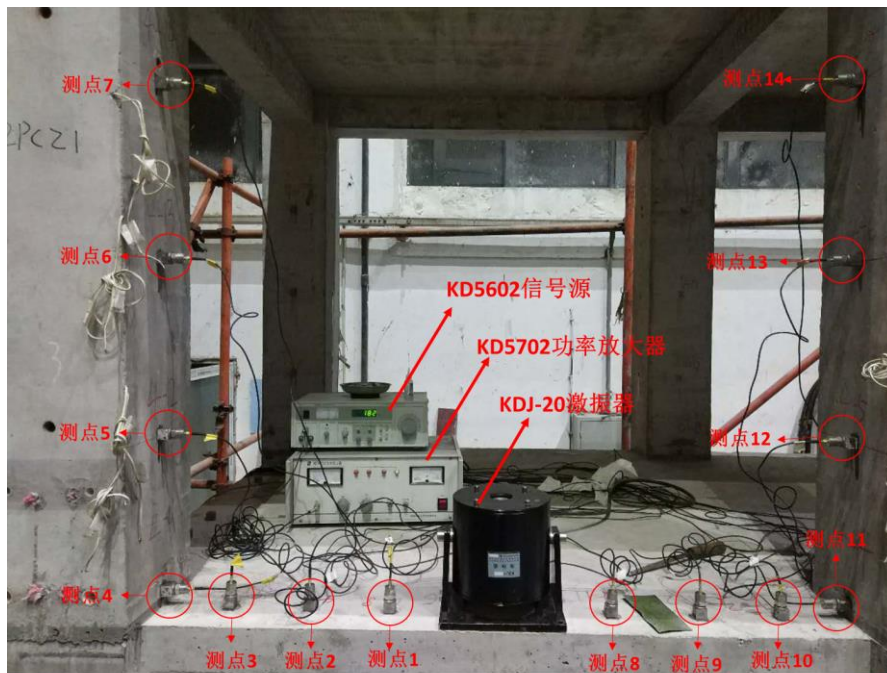


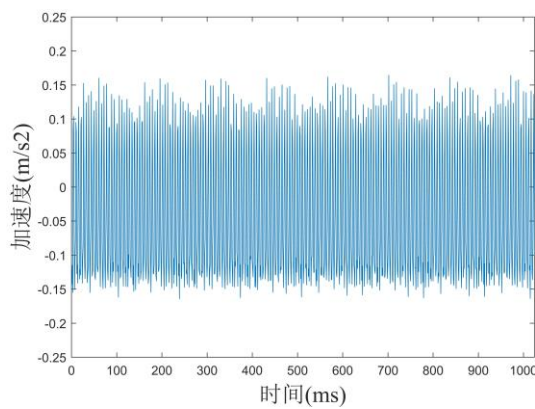
图 3.16 工况 1 试验过程

3.3 试验结果分析

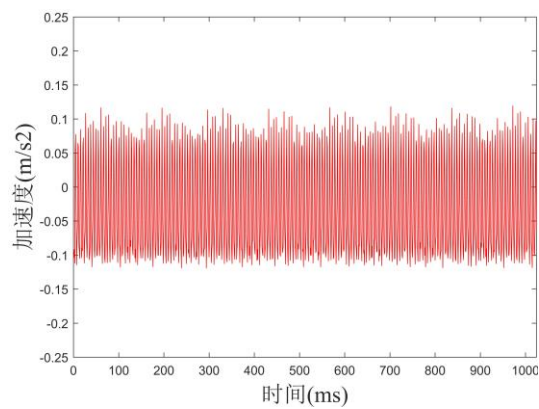
3.3.1 加速度时程分析

试验中采集得到了各工况下个测点的加速度响应，首先对其时程曲线进行对比分析，由于激振器为连续激振，因此各工况中各测点的加速度时程曲线形式相近，因此节选工况 1 中各测点的代表性片段进行分析，工况 1 中各测点的代表性加速度时程曲线如图 3.17 所示。通过对比在相同激振条件下工况 1 中不同测点的加速度时程响应曲线可以发现，梁上各测点（布置在无缺陷柱一侧的测点 1、2、3 以及布置在缺陷柱一侧的测点 8、9、10）的加速度响应随着测点与激振点

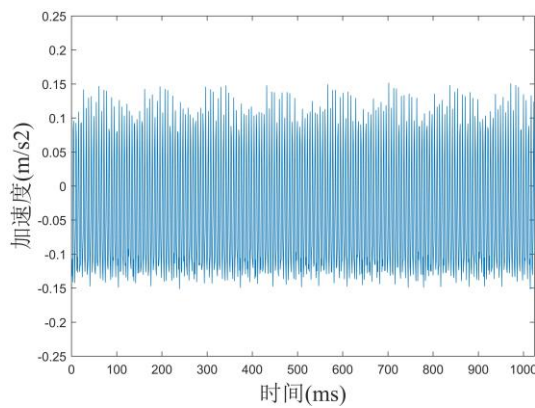
之间距离的增加而逐渐减小。无缺陷柱上测点 5、6、7 的加速度时程的幅值较为接近，而测点 4 的加速度时程幅值则相对较小；同样地，缺陷柱上测点 12、13、14 的加速度时程的幅值较为接近，而测点 11 的加速度时程幅值则相对较小。这是由于测点 4 和测点 11 非常接近梁和柱的连接处，这导致该处的刚度出现局部增加的现象，进而造成了加速度响应的减小。接下来对工况 1 中布置在无缺陷节点和有缺陷节点中相同位置处测点（测点 1 对应于测点 8、测点 2 对应于测点 9，以此类推）的加速度响应进行对比分析可以发现：相对于测点 1，测点 8 的加速度时程幅值有一定的下降，测点 9、测点 10 和测点 11 也出现相同的趋势；然而相对于测点 5，测点 12 的的加速度时程幅值有一定的上升，测点 13 和测点 14 也出现相同的趋势。



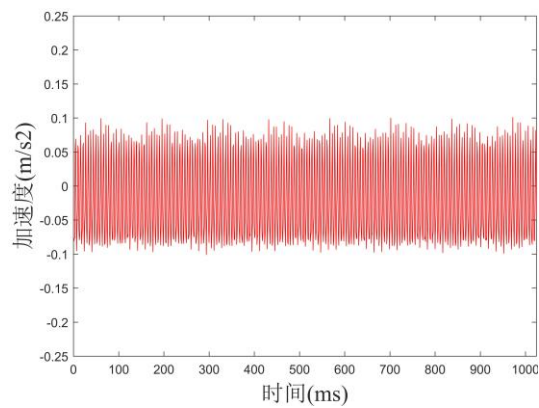
(a) 测点 1 代表性加速度时程曲线



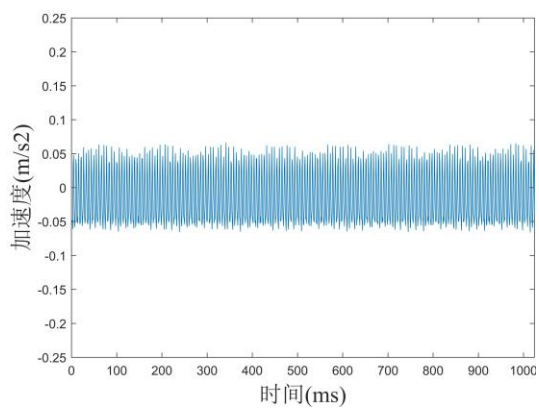
(b) 测点 8 代表性加速度时程曲线



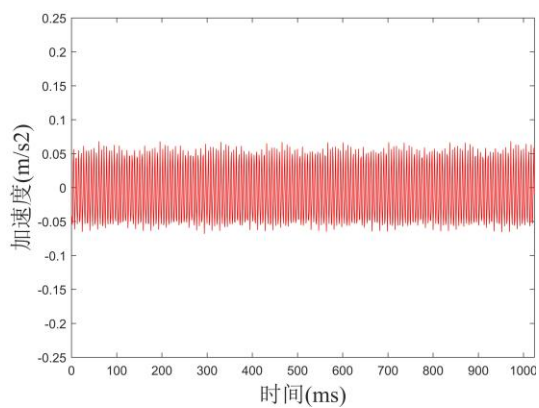
(c) 测点 2 代表性加速度时程曲线



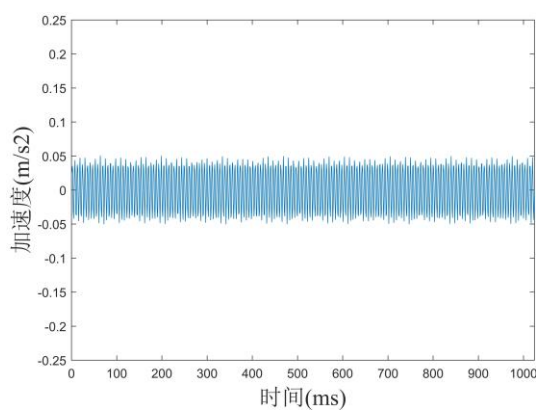
(d) 测点 9 代表性加速度时程曲线



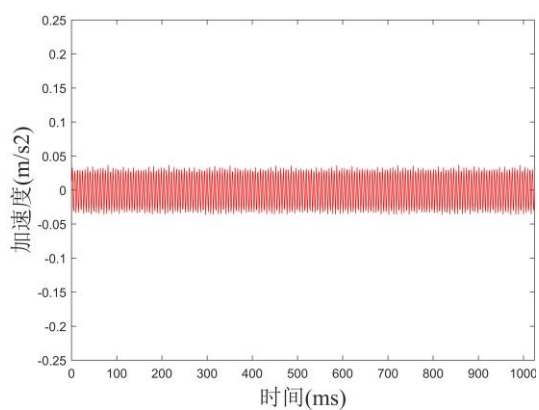
(e) 测点 3 代表性加速度时程曲线



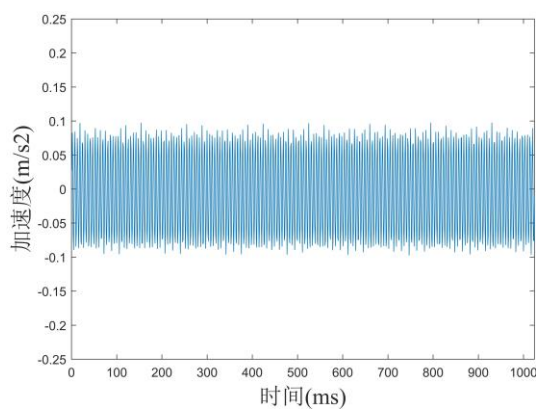
(f) 测点 10 代表性加速度时程曲线



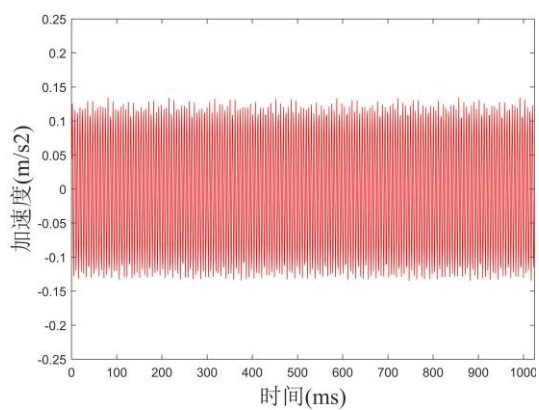
(g) 测点 4 代表性加速度时程曲线



(h) 测点 11 代表性加速度时程曲线



(i) 测点 5 代表性加速度时程曲线



(j) 测点 12 代表性加速度时程曲线

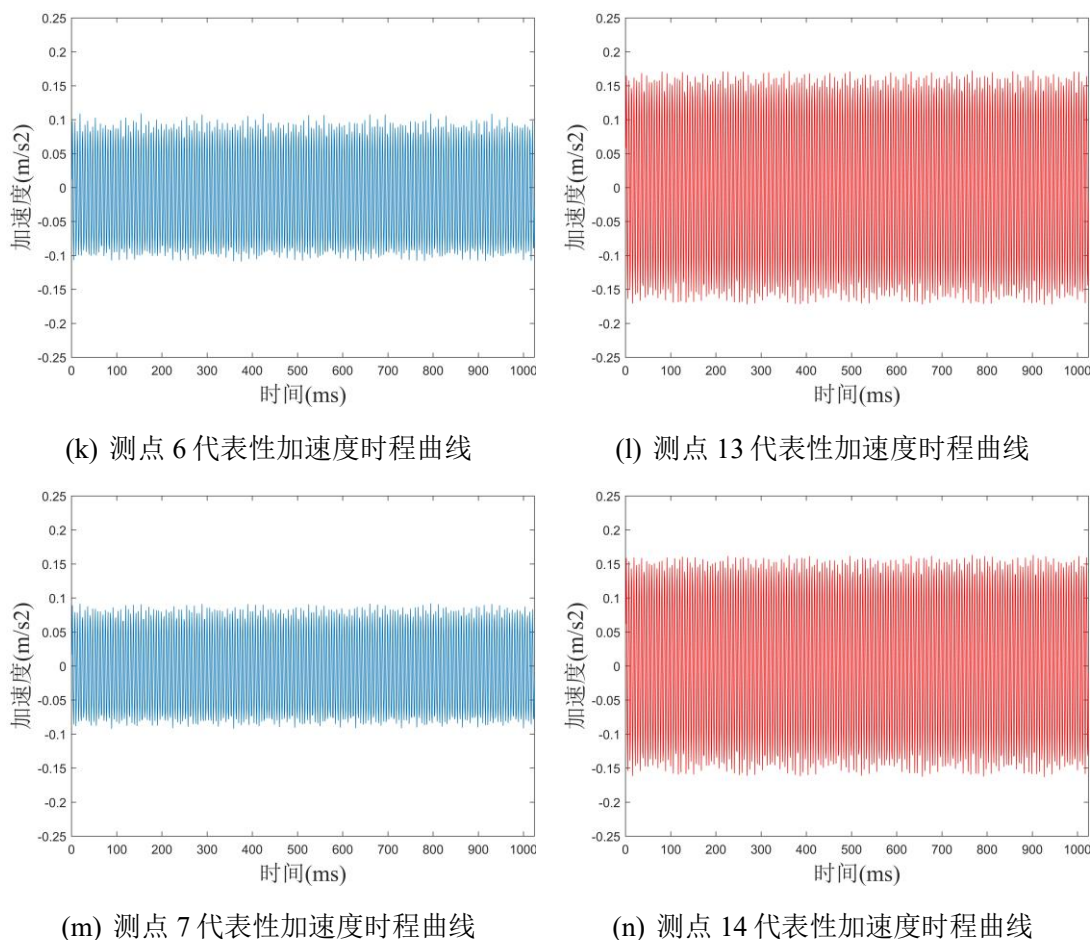


图 3.17 工况 1 各测点代表性加速度时程曲线对比

3.3.2 数据前处理及数据集搭建

根据上文中的试验概况可得,装配式框架结构激振试验中每个工况都会采集得到 14 条加速度时程,每条加速度时程均包含 61440 个信号,需要对其进行相前处理。下文将着重阐述加速度时程信号前处理的方法以及搭建用于训练和测试卷积神经网络数据集的方法。

(1) 数据前处理

根据 3.3.1 节中对各工况中加速度时程曲线的分析可知,虽然每个工况中各测点的加速度时程曲线在整体形式上非常接近,但是不同工况以及不同测点间的加速度幅值具有较大差异。因此在利用卷积神经网络进行装配式混凝土框架缺陷识别之前,需要把所有加速度时程的幅值调整一致,即对其做幅值归一化的前处理。本文将依照如下公式对所有加速度时程做幅值归一化处理。

$$y_i = \frac{x_i}{\max_{i=1,2,\dots,N} (|x_i|)} \quad (3.1)$$

式中：

x 表示原始加速度时程；

y 表示幅值归一化之后的加速度时程。

这样的前处理操作仅对加速度时程的幅值做了统一的调整，同时也保证处理之后的加速度时程与原始时程在形式上保证一致，不会导致丢失过多的信息；同时将所有测点加速度时程的幅值调整一致也保证了后续卷积神经网络训练过程中的稳定性，减少由于测点间加速度时程幅值相差过大而造成的收敛速度过慢甚至不收敛的现象。后续的相关操作，诸如数据集搭建、卷积神经网络的训练与测试以及装配式混凝土框架结构浆缺陷识别均会采用幅值归一化之后的加速度时程，而不会直接采用原始的加速度时程。

(2) 数据集搭建

如图 2.28 所示，数据集的搭建是基于深度学习的装配式混凝土结构缺陷识别方法的开端，其重要性不言而喻。

本文以试验中各测点的加速度时程作为输入样本。如 3.2.5 节所述，每个工况均采集到 14 条加速度时程，每条加速度时程均包含 61440 个信号，其中前 7 条加速度时程于无缺陷的预制柱和叠合梁上采集得到，称之为无缺陷响应，后 7 条加速度时程于有缺陷的预制柱和叠合梁上采集得到，称之为有缺陷响应。截取自无缺陷响应以及有缺陷响应中时长 1s 内 7 个测点的加速度信号组成一个样本，因此每个样本的尺寸都为 1024×7 。为扩充样本的个数，每个样本在产生过程中信号会出现一定的交叉。本文将无缺陷响应和有缺陷响应分别划分成 5000 个样本，即每个工况都会产生 10000 个样本。同一个工况中的 10000 个样本将组成该工况下的数据集(Data Set)，因此整个试验会产生样本个数和样本尺寸都一致，但样本来源不同的 7 个数据集，以工况 1 为例数据集搭建为图 3.18 所示。为保证训练集和测试集中无缺陷样本和缺陷样本数量保持一致，本文从每个数据集的 5000 个无缺陷样本中随机挑选 80% 以及从该数据集中的 5000 个有缺陷样本中随机挑选 80%，共 8000 个样本组成训练集 (Train Set)。同时每个数据集中的剩余样本，共 2000 个 (包括 1000 个无缺陷样本和 1000 个有缺陷样本) 组成测试集 (Test Set)，各工况数据集中样本个数如表 3.6 所示。在神经网络的训练过程中，每次迭代都会从训练集的样本中随机选取一定数量的样本作为输入对该网络进行训练。训练结束之后，会依次把测试集中的样本作为输入对训练完成的卷积神经网络进行测试，来验证该方法在装配式混凝土框架结构缺陷识别问题中的有效性。

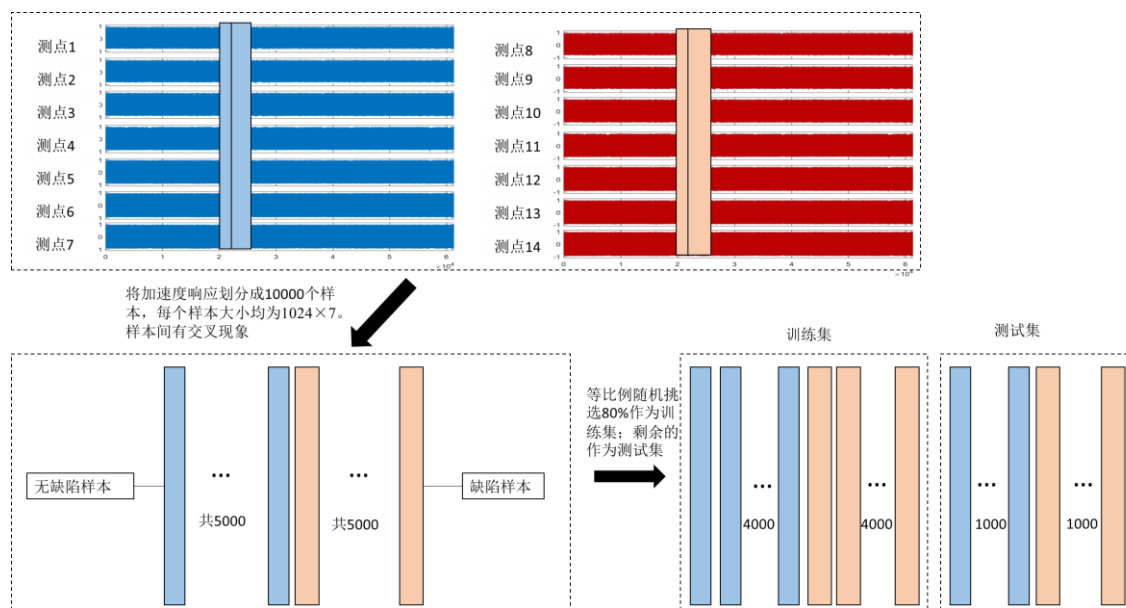


图 3.18 工况 1 数据集搭建示意图

表 3.6 各工况数据集中样本个数汇总

工况	训练集			测试集		
	有缺陷样本	无缺陷样本	总计	有缺陷样本	无缺陷样本	总计
工况 1	4000	4000	8000	1000	1000	2000
工况 2	4000	4000	8000	1000	1000	2000
工况 3	4000	4000	8000	1000	1000	2000
工况 4	4000	4000	8000	1000	1000	2000
工况 5	4000	4000	8000	1000	1000	2000
工况 6	4000	4000	8000	1000	1000	2000
工况 7	4000	4000	8000	1000	1000	2000

3.3.3 各工况缺陷及缺陷位置识别

目前，结构的缺陷/损伤识别主要可以分为五大内容^[80]：（1）识别结构系统中是否存在缺陷/损伤；（2）完成结构缺陷/损伤的定位；（3）判别结构缺陷/损伤的类型；（4）完成结构缺陷/损伤程度的标定与评价；（5）预测结构的剩余使用寿命。因此本节将针对每一个工况，对其中的样本是否为缺陷样本以及缺陷所在的位置进行识别。

（1）样本标签设置

卷积神经网络作为一个监督学习算法，每个输入样本都需要一个已知的输出，即为该样本的标签。参考文献^[44]，本节中样本标签设置为一个包含 8 个元素的向量，每个元素都代表缺陷位于相应测点编号的概率，即前 7 个元素分别代表了缺陷位于试验中 7 个测点的概率，而第 8 个元素则表示该样本为无缺陷样本的概率。

率。因此，样本标签由以下规则生成：当缺陷位于第 i 个测点处时，标签向量中的第 i 个元素为 1，其余元素均为 0；当该样本为无缺陷样本时，标签向量中的第 8 个元素为 1，其余元素均为 0。也就是说，标签向量中某元素的取值为 1 意味着缺陷位于该测点处的可能性为 100%，其余的 0 元素则意味着缺陷位于这些测点处的可能性为 0。

由于本次试验主要针对装配式混凝土框架结构中灌浆缺陷的识别，缺陷均位于测点 4 处，因此各工况中有缺陷样本的标签为 [0,0,0,1,0,0,0,0]，而无缺陷样本标签为 [0,0,0,0,0,0,0,1]。这也是未来需要做更为深化的研究，即人为引入多种不同位置的缺陷。各工况中样本的标签如表 3.7 所示。

表 3.7 各工况中样本标签汇总表

	工况	缺陷所处位置	标签
无缺陷样本	工况 1	无	[0,0,0,0,0,0,0,1]
	工况 2	无	[0,0,0,0,0,0,0,1]
	工况 3	无	[0,0,0,0,0,0,0,1]
	工况 4	无	[0,0,0,0,0,0,0,1]
	工况 5	无	[0,0,0,0,0,0,0,1]
	工况 6	无	[0,0,0,0,0,0,0,1]
	工况 7	无	[0,0,0,0,0,0,0,1]
有缺陷样本	工况 1	测点 4	[0,0,0,1,0,0,0,0]
	工况 2	测点 4	[0,0,0,1,0,0,0,0]
	工况 3	测点 4	[0,0,0,1,0,0,0,0]
	工况 4	测点 4	[0,0,0,1,0,0,0,0]
	工况 5	测点 4	[0,0,0,1,0,0,0,0]
	工况 6	测点 4	[0,0,0,1,0,0,0,0]
	工况 7	测点 4	[0,0,0,1,0,0,0,0]

这样设置样本标签即可以直观地反映出各个样本是否存在缺陷，同时可以表征该样本中缺陷所处的测点位置，为后续装配式混凝土结构缺陷识别提供方便。同时，由于卷积神经网络使用 Softmax 分类器计算网络最后的输出。由上文所述可知，Softmax 分类器将来自上层的输入映射成一个向量，其中的每个元素都在 0 到 1 的范围之间，即表示卷积神经网络识别的缺陷概率。

基于上述标签设置，本节中卷积神经网络各层的特征映射大小如表 3.8 所示。

表 3.8 卷积神经网络结构的详细参数

编号	类型	核尺寸	核数量	步长	输入	输出
C1	卷积层	45	6	1	1024×7	6@980×7
S2	池化层	5	-	1	6@980×7	6@196×7
C3	卷积层	32	12	1	6@196×7	12@165×7
S4	池化层	5	-	1	12@165×7	12@33×7
C5	卷积层	20	12	1	12@33×7	12@14×7
S6	池化层	2	-	1	12@14×7	12@7×7
FC	全连接层	-	-	-	12@7×7	8×1

(2) 系统环境及超参数设置

上文已成功设计出用于基于深度学习的装配式混凝土缺陷识别方法中的卷积神经网络，并成功搭建了7种工况的数据集，本节将利用数据集中的训练集对卷积神经网络进行训练和参数的优化。本文采用64位 Window 10 + Matlab (Version 2018a)的系统环境，该计算机的处理器为第八代 Intel Core i7 主频可达3.2GHz，系统内存容量为8GB。

本文采用的优化算法为梯度下降算法，在介绍卷积神经网络的训练过程之前，需要对其中涉及到的几个关键性概念进行介绍。主要涉及三个概念，第一是“一批样本(Batch)”及“批处理大小(Batch_size)”，第二是“迭代(Iteration)”，第三是“一代训练(Epoch)”。在训练过程中为加快训练的收敛速度，并减少计算机的计算负荷，通常会使用训练集中的一小部分样本对网络的权重参数与偏置参数等进行一次更新，这一小部分样本则被称为一批样本，即一个 Batch，每批样本中样本的个数称之为批处理大小，即 Batch_Size。而使用一批样本对网络进行一次参数更新的过程，则被称为一次迭代，迭代的次数称之为迭代次数。使用训练集中的全部样本对网络进行一次完整的训练，则被称为“一代训练”，即一个 Epoch。迭代次数可以根据批处理大小，Epoch 以及训练集中样本的个数换算而成，换算公式如式(3.2)所示。

$$I_N = E_N \times \frac{T_S}{B_S} \quad (3.2)$$

式中：

I_N 表示 Iteration_Number，即迭代次数；

E_N 表示 Epoch_Number，即训练代数；

T_S 表示 Train Set_Size，即训练集中样本的个数；

B_S 表示 Batch_Size，即批处理大小。

由于本次试验中产生的训练集样本个数相对较少，因此为保证有足够数量批的样本用于训练，批处理大小设置为5，也就是说每次迭代都使用5个样本对卷

积神经网络进行训练；训练代数设置为 20，也就是说用训练集中的全部样本对网络进行 20 次完整的训练。因此，每个工况中卷积神经网络训练的迭代次数达 $20 \times 8000 / 5 = 32000$ 次。各工况卷积神经网络训练过程中的超参数设置如表 3.9 所示。

表 3.9 各工况卷积神经网络训练过程中的超参数设置

工况	数据集大小	训练集大小	批处理大小	训练代数	迭代次数
工况 1	10000	8000	5	20	32000
工况 2	10000	8000	5	20	32000
工况 3	10000	8000	5	20	32000
工况 4	10000	8000	5	20	32000
工况 5	10000	8000	5	20	32000
工况 6	10000	8000	5	20	32000
工况 7	10000	8000	5	20	32000

(3) 训练过程

为实现每个工况中缺陷及其位置的识别，需要对每个工况中卷积神经网络分别进行训练。对每个工况而言，将该工况训练集中的样本分批次迭代 32000 次使得衡量卷积神经网络真实输出与样本标签之间差值的代价函数最小。随着迭代次数的不断增大，网络的代价函数会逐渐减小，并收敛至稳定水平。7 个工况卷积神经网络训练中的代价函数如图 3.19 所示。

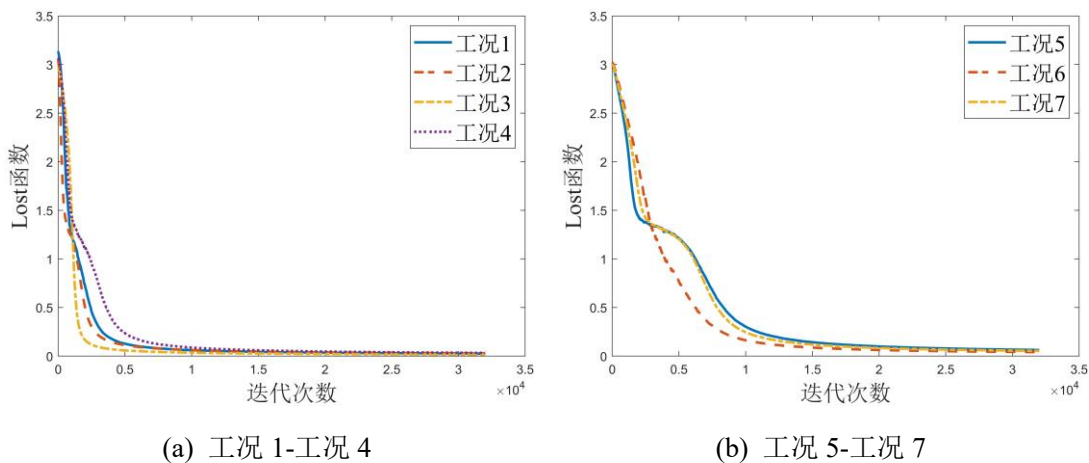


图 3.19 卷积神经网络训练过程中代价函数下降图

从图 3.19 中可以看出，7 个工况的卷积神经网络在迭代 32000 次之后均实现了收敛。对比纵向 4 个工况（工况 1-工况 4）可以发现，工况 3 的收敛速度最快，在迭代 3000 次之后即实现了收敛。工况 1 和工况 2 紧随其后，在迭代 6000 次之后也实现了收敛，工况 4 最晚实现收敛，在迭代 9000 次之后实现了收敛。对比

横向3个工况(工况5-工况7)可以发现,工况6的收敛速度最快,在迭代13000次之后实现了收敛。工况7也在迭代18000次之后实现了收敛,而工况5的收敛速度最慢,在迭代20000次之后也实现了收敛。

从收敛速度来看,纵向(A-C轴)4个工况(工况1-工况4)中代价函数的收敛速度明显大于横向(1-2轴)3个工况(工况5-工况7)中代价函数的收敛速度。这是由于纵向4个工况中无缺陷柱和有缺陷柱的边界条件有一定的区别,例如工况1中的无缺陷柱是1号柱(边柱)而有缺陷柱是2号柱(中柱)。这种差别导致了两者在加速度响应方面的差别更大,进而更容易实现灌浆缺陷及其位置的识别,因此纵向的4个工况中卷积神经网络的训练也能更快收敛。

(4) 识别结果判断依据

由于本节仅涉及到对工况缺陷及缺陷位置的识别,即只进行定性分析。由于样本标签中每个元素代表缺陷位于该测点处的概率,因此本节对缺陷及其位置的识别逻辑如下:当某样本输出向量中的第8个元素大于0.95时,该样本被识别为有缺陷的样本;而当某样本输出向量中的第*i*个元素($i=1,2,\dots,7$)大于0.95时,该样本被识别为有缺陷样本,且缺陷位于测点*i*处;当某样本输出向量中没有元素大于0.95时,则表示识别失效。

准确识别装配式混凝土框架结构缺陷的定义为能够准确识别出是否有缺陷以及缺陷所处的测点位置,相应地识别准确率则定义为准确识别的样本数与总样本数的比值,识别准确率如式(3.3)所示。本节将基于上文中对准确识别灌浆缺陷以及识别准确率的定义,对卷积神经网络的识别效果进行评价。

$$\alpha = \frac{S}{N} \times 100\% \quad (3.3)$$

式中:

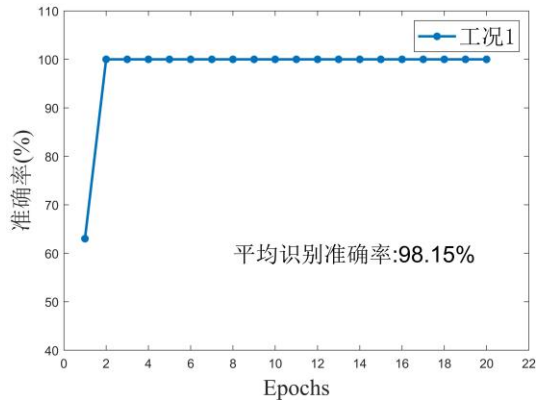
α 表示识别准确率;

S 表示准确识别的样本数;

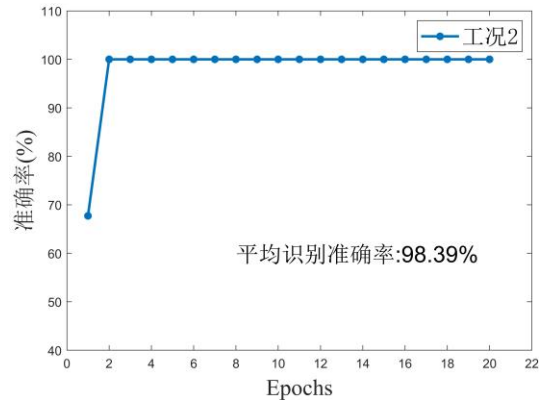
N 表示总样本数。

(5) 识别效果评价

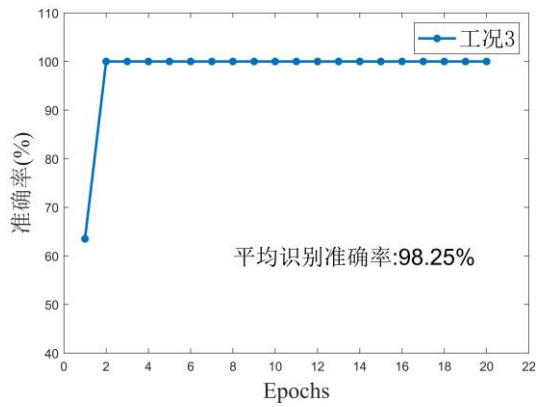
训练识别准确率代表了对输入样本产生正确识别的比例,表示训练后的网络在训练集上能正确识别灌浆缺陷的平均水平。以一代训练中的8000个样本为一个单元,对7个工况的训练识别准确率进行比较,卷积神经网络训练过程中的识别准确率如图3.20所示。



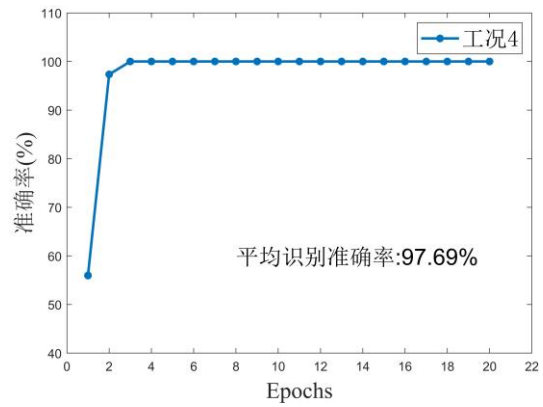
(a) 工况 1



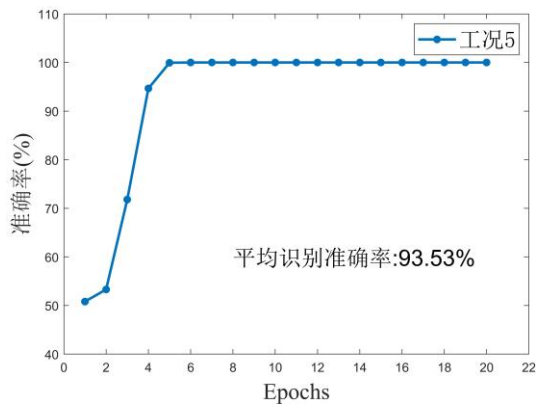
(b) 工况 2



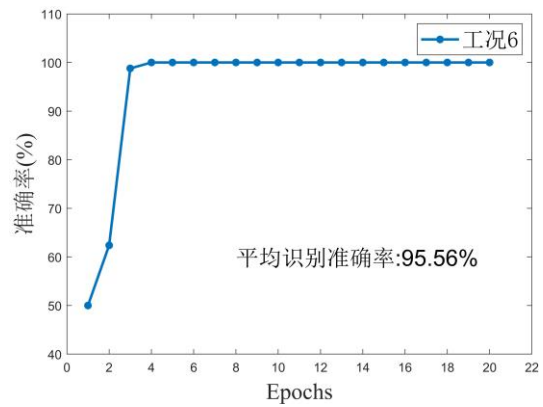
(c) 工况 3



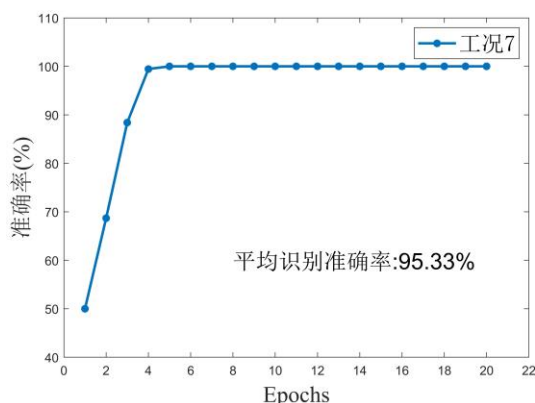
(d) 工况 4



(e) 工况 5



(f) 工况 6



(g) 工况 7

图 3.20 7 个工况中卷积神经网络的训练识别准确率变化图

从图 3.20 中可以看出, 7 个工况的卷积神经网络在训练过程中的识别准确率随着训练代数的增长在不断地上升, 直到识别准确率达 100%。其中纵向 4 个工况 (工况 1-工况 4) 均在训练到第 2 代时已经实现识别准确率达 100%。而横向 3 个工况 (工况 5-工况 7) 也在训练到第 5 代前实现识别准确率达 100%。这也与上文中纵向工况收敛速度较快的现象一致。同时图 3.20 中也展示了各工况在训练过程中的平均识别准确率, 即 20 代训练过程中各自识别准确率的平均值。其中纵向 4 个工况 (工况 1-工况 4) 的平均识别准确率较高, 位于 98% 左右; 横向 3 个工况 (工况 5-工况 7) 的平均识别准确率则相对较低, 位于 95% 左右。这也说明 7 个工况的卷积神经网络在训练过程中均能实现较好的识别准确率。但是通过对比 7 个工况的平均识别准确率可以发现: 纵向 4 个工况 (工况 1-工况 4) 的识别准确率明显高于横向 3 个工况 (工况 5-工况 7) 的识别准确率。

将各工况测试集中的 2000 个样本依次输入训练完成的卷积神经网络中, 根据输出和上文的定义对识别结果进行评价。随机从 7 个工况中各自挑选一个有缺陷样本和一个无缺陷样本, 各自的输出向量如图 3.21 所示。从图 3.21 中可以看出, 7 个工况中典型无缺陷样本的输出向量表现出同样的规律, 即第 8 个元素均是该向量 8 个元素中的最大值, 且都接近于 1, 这说明这 7 个样本都识别为无缺陷。同时 7 个工况中典型有缺陷样本的输出向量表现出同样的规律, 即第 4 个元素均接近于 1, 这说明这 7 个样本都被识别为有缺陷, 且缺陷位于测点 4。因此可以得出结论, 对于上述随机挑选的 14 个样本而言, 卷积神经网络均能准确地实现缺陷及其位置的识别。

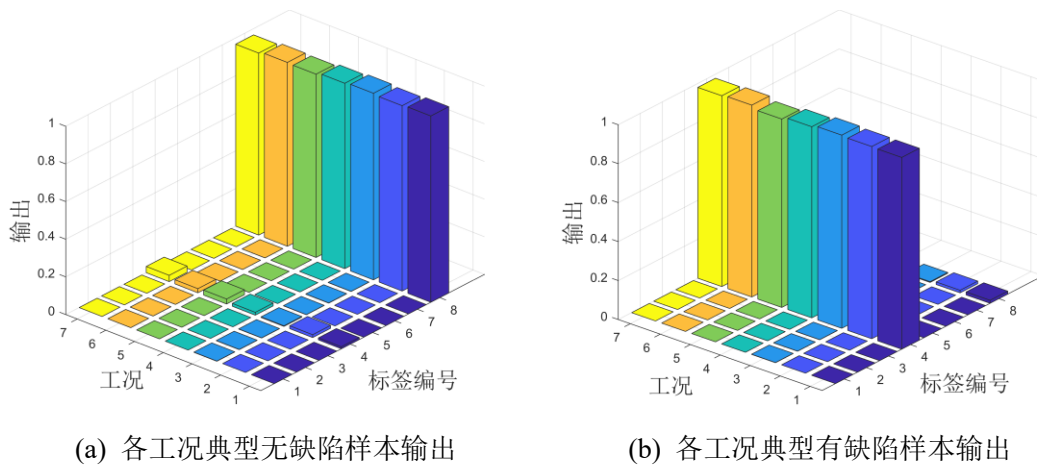


图 3.21 各工况典型样本输出

下文将分别对 7 个工况测试集中的 2000 样本的识别准确率进行分析和比较，7 个工况测试集的识别准确率如表 3.10 所示。

表 3.10 7 个工况测试集的识别准确率

工况	识别准确率
工况 1	100%
工况 2	100%
工况 3	100%
工况 4	100%
工况 5	100%
工况 6	100%
工况 7	100%

从表 3.10 中可以看出，7 个工况中测试集的识别准确率均为 100%。这说明基于深度学习的装配式混凝土结构缺陷识别方法在这 7 个工况中均能很好地实现灌浆缺陷及其位置的识别。因此可以判断上文中所建立的基于深度学习的装配式混凝土结构缺陷识别方法在装配式框架结构的缺陷及其位置的识别中具有非常优秀的表现。

3.3.4 不同缺陷程度识别

上节已利用卷积神经网络实现了对 7 种工况的缺陷及其位置的识别，本节将进一步对不同缺陷程度做进一步的对比识别。由于横向三个工况(工况 5-工况 7)中预制柱的边界条件等都一致，同时工况 5 的缺陷柱中有 1 根套筒未灌浆、工况 6 的缺陷柱中有 3 根套筒为灌浆、工况 7 的缺陷柱中有 2 根套筒未灌浆，即三个工况中缺陷柱涵盖了 25%、50%和 75%三个不同的缺陷程度，因此本节将选用这 3 个工况来完成不同缺陷程度的识别。

(1) 样本标签设置

在进行缺陷识别之前，首先需要将这3个工况数据集中的5000个缺陷样本组合成一个全新数据集，并等比例的随机挑选80%，作为训练集（共12000个样本，3个工况各4000个）；剩余的20%作为测试集（共3000个样本，3个工况各1000个）。

本节中样本标签设置为包含3个元素的向量，3个元素都分别代表该样本缺陷程度的概率，即第1个元素代表了样本缺陷是25%的概率，第2个元素代表了样本缺陷是50%的概率，第3个元素代表了样本缺陷是75%的概率。因此，样本标签由以下规则生成：当某样本缺陷为25%时，标签中的第1个元素取值为1，其余两个元素取值均为0，其余情况以此类推。本节中样本的标签设置如表3.11所示。

表 3.11 样本标签汇总

工况	未灌浆套筒数	缺陷程度	标签
工况 5	1	25%	[1,0,0]
工况 6	3	75%	[0,0,1]
工况 7	2	50%	[0,1,0]

基于上述标签设置，本节中卷积神经网络各层的特征映射大小如表3.12所示。

表 3.12 卷积神经网络结构的详细参数

编号	类型	核尺寸	核数量	步长	输入	输出
C1	卷积层	45	6	1	1024×7	6@980×7
S2	池化层	5	-	1	6@980×7	6@196×7
C3	卷积层	32	12	1	6@196×7	12@165×7
S4	池化层	5	-	1	12@165×7	12@33×7
C5	卷积层	20	12	1	12@33×7	12@14×7
S6	池化层	2	-	1	12@14×7	12@7×7
FC	全连接层	-	-	-	12@7×7	3×1

(2) 系统环境及超参数设置

本节中对卷积神经网络训练所采用的系统环境与3.3.3节中一致，但是相关超参数的设置会有一些的变化，具体超参数设置如表3.13所示。

表 3.13 超参数设置

数据集大小	训练集大小	测试集大小	批处理大小	训练代数	迭代次数
15000	12000	3000	5	20	48000

(3) 训练过程

卷积神经网络训练过程中的 Lost 函数下降图如图 3.22 所示。

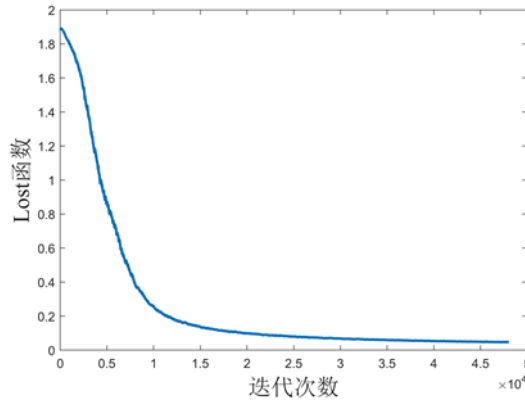


图 3.22 卷积神经网络训练过程中代价函数下降图

从图 3.22 中可以看出，在迭代 30000 次之后，代价函数逐渐收敛。与 3.3.4 节中各工况缺陷识别的代价函数相比，本节中代价函数的收敛速度相对较慢，这是由于对缺陷程度的识别过程更为复杂，数据集中的样本组成更为复杂，这两个因素造成了代价函数收敛较慢的现象。

(4) 识别结果判断依据

与 3.3.3 节相比，本节需要对各工况的缺陷程度进行识别，也更加偏向于定量地分析，因此 3.3.3 节中对准确识别的定义将不再适用于本节，故下文将提出全新的定义。为实现不同缺陷程度的识别，需要评估样本输出向量与三种不同缺陷程度样本标签之间的差异，本节引入欧式距离来计算这种差值。欧式距离也称欧几里得距离，其度量了 n 维空间中两点之间的距离。N 维空间中两点 (x_1, x_2, \dots, x_n) 和 (y_1, y_2, \dots, y_n) 间的欧式距离如式(3.4)所示。

$$\rho = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2} \quad (3.4)$$

假设某样本的输出为 $\mathbf{O}=[o_1, o_2, o_3]$ ，25%缺陷的样本标签为 $\mathbf{A}=[a_1, a_2, a_3]$ ，50%缺陷的样本标签为 $\mathbf{B}=[b_1, b_2, b_3]$ ，75%缺陷的样本标签为 $\mathbf{C}=[c_1, c_2, c_3]$ 。输出 \mathbf{O} 与样本标签 \mathbf{A} 、 \mathbf{B} 、 \mathbf{C} 之间的欧式距离分别按照式(3.5)-式(3.7)计算。

$$\Delta_1 = \sqrt{(o_1 - a_1)^2 + (o_2 - a_2)^2 + (o_3 - a_3)^2} \quad (3.5)$$

$$\Delta_2 = \sqrt{(o_1 - b_1)^2 + (o_2 - b_2)^2 + (o_3 - b_3)^2} \quad (3.6)$$

$$\Delta_3 = \sqrt{(o_1 - c_1)^2 + (o_2 - c_2)^2 + (o_3 - c_3)^2} \quad (3.7)$$

在某样本输出向量的三个欧式距离中若 Δ_1 远小于 Δ_2 和 Δ_3 ，则该样本被识别为 25%的缺陷程度； Δ_2 若远小于 Δ_1 和 Δ_3 ，则该样本被识别为 50%的缺陷程

度；若 Δ_3 远小于 Δ_1 和 Δ_2 ，则该样本被识别为 75% 的缺陷程度；若 Δ_1 、 Δ_2 和 Δ_3 中没有一个元素远小于其余两个，则该样本为无效识别。下文将基于上述定义，对基于深度学习的装配式混凝土结构缺陷识别方法在不同缺陷程度识别中的效果进行评价。

随机从测试集中挑选三个样本（每个工况各一个），各自的输出向量如图 3.23 所示，其与三种样本标签间的欧式距离如表 3.14 所示。从图中可以看出，工况 5 样本的输出向量中第 1 个元素为 0.98，而其余两个元素均接近于 0；工况 6 样本的输出向量中第 3 个元素为 0.98，而其余两个元素均接近于 0；工况 7 样本的输出向量中第 2 个元素为 0.97，而其余两个元素均接近于 0。这与各工况的实际标签极为接近。根据表中的欧式距离可以看出，工况 5 样本被识别为 25% 缺陷，工况 6 样本被识别为 75% 缺陷而工况 7 样本被识别为 50% 缺陷，这与实际情况一致，下节将对测试集中所有样本的识别结果进行评价。

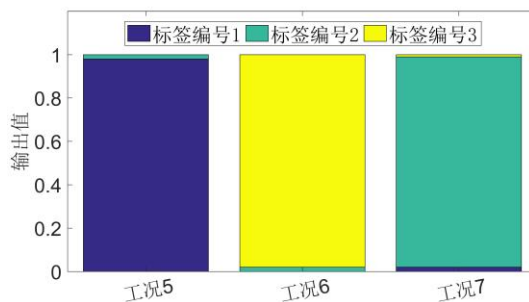


图 3.23 缺陷程度识别典型样本输出向量

表 3.14 缺陷程度识别典型样本识别结果

样本所在工况	输出向量	参考基准	欧式距离	识别结果
工况 5	[0.98,0.02,0]	25%缺陷	0.028	25%缺陷
		50%缺陷	1.38	
		75%缺陷	1.4	
工况 6	[0,0.02,0.98]	25%缺陷	14	75%缺陷
		50%缺陷	1.38	
		75%缺陷	0.028	
工况 7	[0.02,0.97,0.01]	25%缺陷	1.37	50%缺陷
		50%缺陷	0.038	
		75%缺陷	1.38	

(5) 识别效果评价

将测试集中的样本依次输入训练完成的卷积神经网络中，根据上文的定义对识别结果进行评价。混淆矩阵（Confusion matrix）是用来度量模型分类预测能

力的重要分析工具，每一列各自表示了一个类别的真实数量，每一行则各自表示了一个类别的实际识别结果数^[81]。其形式如式(3.8)所示。

$$C = [c_{ij}], \quad ij = 1, 2, \dots, K \quad (3.8)$$

式中：

C 表示混淆矩阵；

K 表示总分类数；

c_{ij} 表示第 i 类样本被分类成第 j 样本的数量。

从上式可以发现混淆矩阵的对角元素表示为分类成功，而非对角元素则表示分类错误。除此之外，混淆矩阵中有两个非常重要的系数——精准率(Precision, p)和召回率(Recall, r)。一个样本的识别结果可以分为如下四类，真阳性(TP)，假阳性(FP)，真阴性(TN)，假阴性(FN)。以75%的缺陷程度为例，解释如表3.15所示。

表 3.15 对于 75%缺陷的 4 种识别状态

识别缺陷 \ 真实缺陷	75%缺陷	非 75%缺陷
75%缺陷	TP	FP
非 75%缺陷	TN	FN

精准率表示所有识别为75%缺陷的样本中，真实缺陷也是75%的样本比例；而召回率则表示所有真实缺陷为75%的样本中，准确识别为75%的样本比例。两者的计算公式如式(3.9)和式(3.10)所示。

$$p = \frac{TP}{TP + FP} \quad (3.9)$$

$$r = \frac{TP}{TP + FN} \quad (3.10)$$

利用这两个系数可以全面的对分类结果进行评价，即精准率和召回率越高，说明模型的性能越好灌浆缺陷识别结果的混淆矩阵如表3.16所示。

表 3.16 混淆矩阵

		真实缺陷			精准率
		25%	50%	75%	
识别缺陷	25%	1000	0	0	100%
	50%	0	1000	0	100%
	75%	0	0	1000	100%
召回率		100%	100%	100%	

从表 3.16 中可以看出,25%缺陷、50%缺陷和 75%缺陷的召回率均为 100%;25%缺陷、50%缺陷和 75%缺陷的精准率均为 100%。25%缺陷的召回率为 100%,说明真实缺陷是 25%的 1000 个样本中的所有样本均被准确识别为 25%缺陷,没有样本被识别为 50%缺陷和 75%缺陷。同时 25%缺陷的精准率为 100%,说明在被识别为 25%缺陷的所有样本中,真实缺陷也是 25%的样本量占比为 100%,真实缺陷是 50%或者 75%但被识别为 25%缺陷的样本量占比为 0%,两者的具体比例不是本文关心的重点,此文不在详述。50%缺陷和 75%缺陷的召回率和精准率的含义与前文一致,此处不在赘述。精准率和召回率均表明,卷积神经网络在三种缺陷程度上的识别性能均为 100%,说明该网络对三种不同的缺陷程度具有较好的识别性能;且召回率和精准率之间未出现显著的差异,说明该网络未表现出偏向性。

3.4 本章小结

为验证基于深度学习的装配式混凝土结构缺陷识别方法的有效性,本章设计出一个缩尺装配式混凝土框架模型并人为引入灌浆缺陷,通过现场激励获取该框架的局部加速度响应。首先利用上述方法完成各工况的缺陷识别,其次完成不同工况间缺陷程度的识别,并分别提出评价依据对识别效果进行评价。主要内容与结论如下:

(1) 系统地介绍了该试验的试验对象为一缩尺装配式框架模型并人为引入灌浆缺陷。根据灌浆缺陷程度和边界条件,试验共包括 7 个不同的工况,利用激励器激励的方法分别对 7 种工况进行激励获取结构局部加速度响应;

(2) 利用上述方法分别对试验中的 7 种工况进行缺陷识别,并使用识别准确率对识别效果进行评价。结果表明该方法在 7 种工况中的缺陷识别准确率均为 100%,即该方法在装配式框架结构缺陷识别中具有良好的性能。

(3) 利用上述方法对工况 5-工况 7 中的 3 种不同缺陷程度进行识别,并通过混淆矩阵对三种不同缺陷程度的识别效果进行评价,结果显示该方法在装配式框架结构不同缺陷程度的识别中具有良好的性能,同时对三种不同缺陷程度未表

现出明显的偏向性。

第4章 装配式剪力墙缺陷识别试验验证

4.1 前言

第三章设计了一个具有不同缺陷程度的装配式框架结构缩尺模型，并对其进行动力试验验证了第二章中建立的基于深度学习的装配式混凝土结构缺陷识别方法在装配式框架结构缺陷识别问题中的有效性，并对识别准确度进行了测试。但是在实际工程中，纯装配式框架的应用相对较少，往往搭配装配式剪力墙结构同时应用，为拓展基于深度学习的装配式混凝土结构缺陷识别方法的使用范围，本章将以装配式剪力墙的足尺模型作为研究对象，对该方法在装配式剪力墙缺陷识别方面做进一步试验验证。

4.2 试验概况

4.2.1 试验模型介绍

为验证该方法的有效性，本次试验需引入一个无缺陷的剪力墙和一个存在灌浆缺陷的剪力墙进行对照试验。两个剪力墙的墙体截面尺寸均为1580mm×200mm，高度均为2400mm；为更好地还原剪力墙结构的构造，需要在剪力墙上方布置顶梁，两个剪力墙中顶梁的截面尺寸均为300mm×300mm，长度为1980mm；同时分别为两个剪力墙布置相同的基础地梁，基础梁的截面尺寸为300mm×500mm。墙体正视图如图4.1所示，基础梁俯视图如图4.2所示。各构件的详细信息汇总如表4.1所示。

表 4.1 各构件详情汇总

构件类型	截面尺寸	长度或高度	数量	制作方式
剪力墙	1580mm×200mm	2400mm	2	预制
顶梁	300mm×300mm	1980mm	2	预制
基础梁	300mm×500mm	2580mm	2	预制

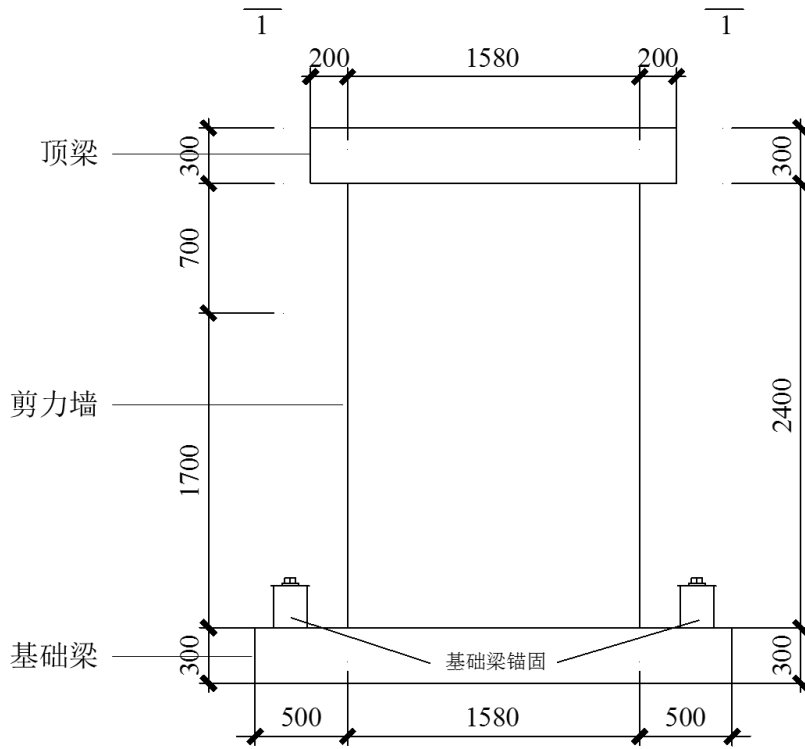


图 4.1 剪力墙尺寸图

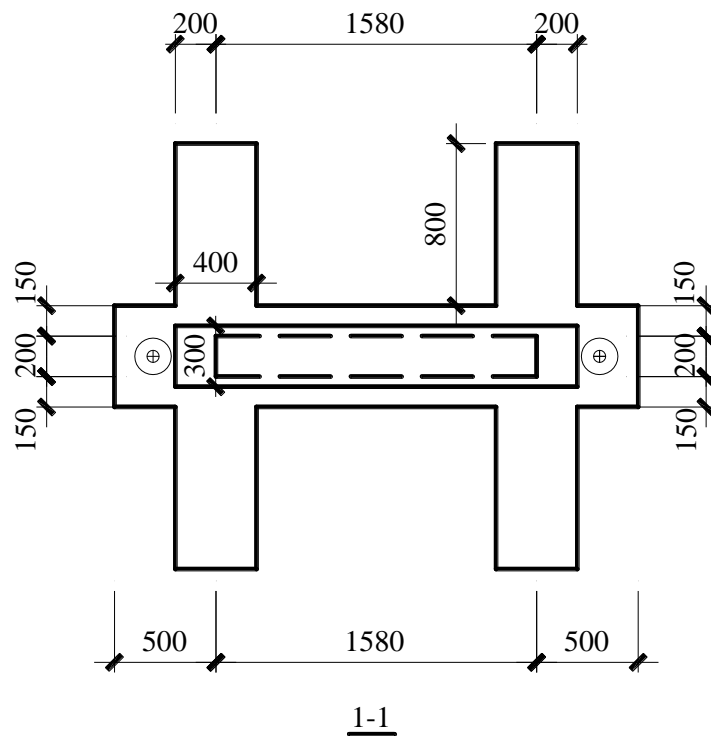


图 4.2 基础梁俯视图

在剪力墙与基础梁的连接处，构件的内部钢筋的连接采用全灌浆套筒连接的方式，连接处均设置 20mm 厚的座浆层（接缝灌浆层），同时为保证剪力墙的输出

平面稳定性，在顶梁与基础梁之间设置4根钢管作为侧向支撑。剪力墙正视图如图4.3所示，侧视图如图4.4所示。

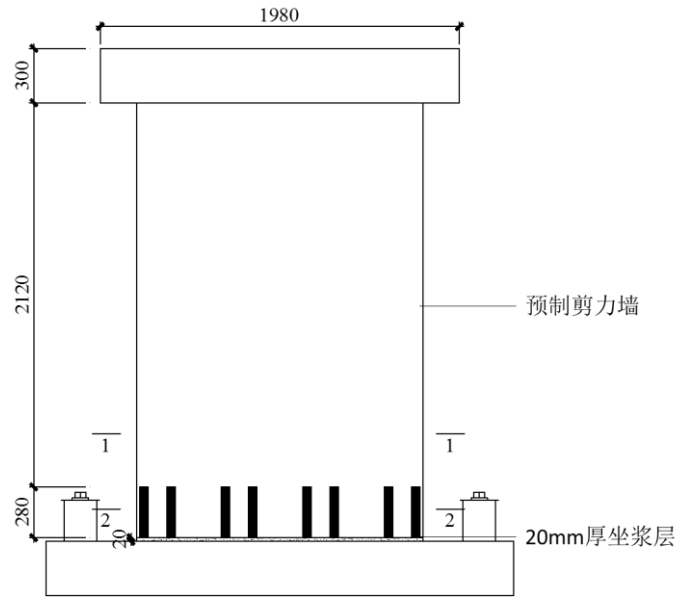


图 4.3 剪力墙正视图

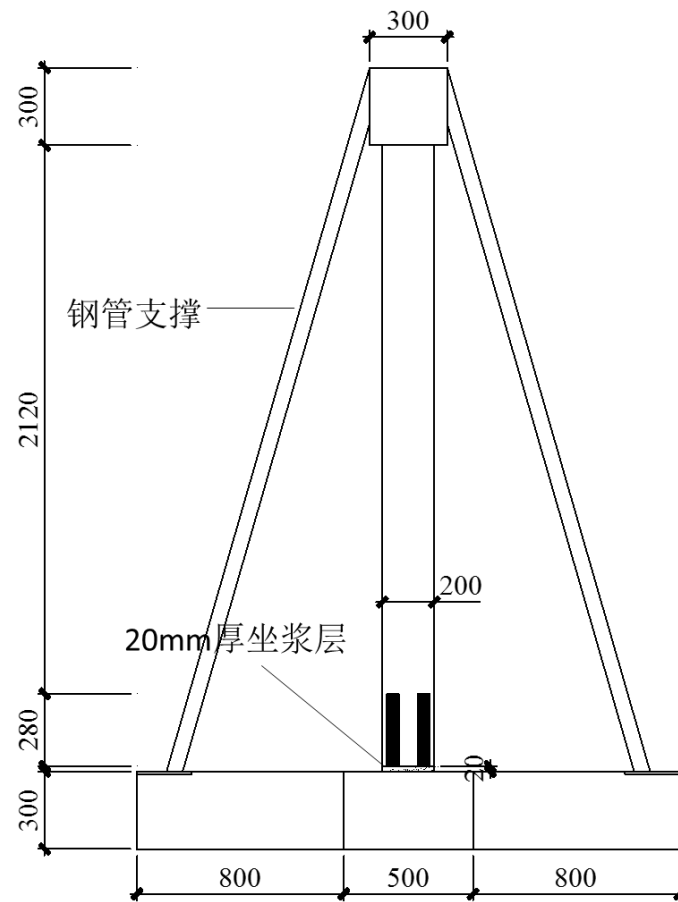


图 4.4 剪力墙侧视图

模型中梁柱混凝土均采用 C30 混凝土，所用的纵筋及箍筋均采用 HRB400 级钢筋，剪力墙和顶梁的配筋如所示。

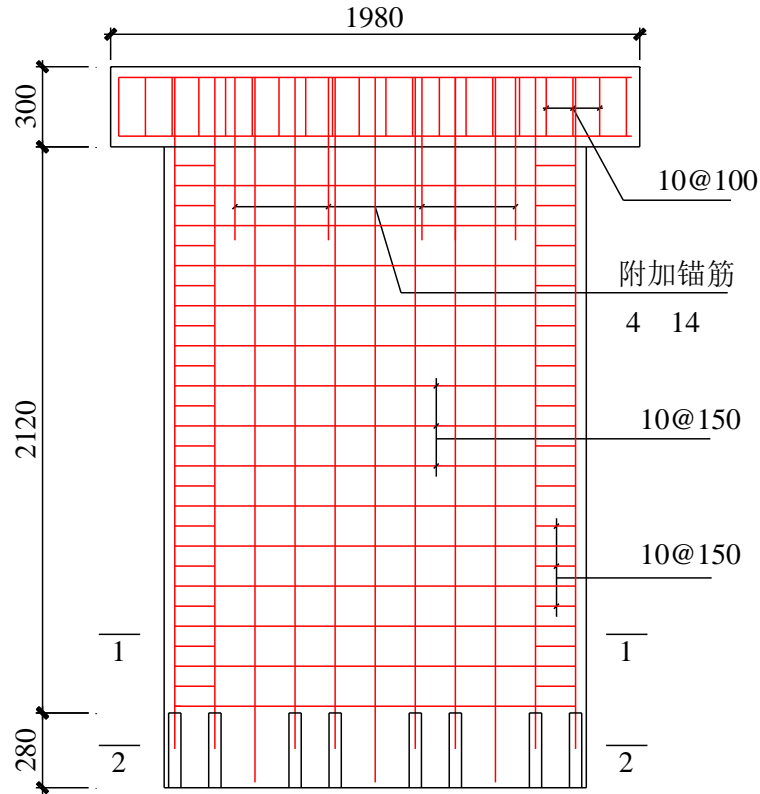


图 4.5 剪力墙与顶梁配筋图

在图 4.5 中，截面 1-1 为不含套筒段的预制剪力墙横截面，这部分的混凝土保护层厚度为 23mm。纵向钢筋采用直径为 14mm 的钢筋，每片墙有 22 根；箍筋采用直径为 10mm 的钢筋，箍筋间距为 150mm，在剪力墙的两端设置角柱，该区域箍筋间距为 75mm，1-1 截面图如图 4.6 所示。

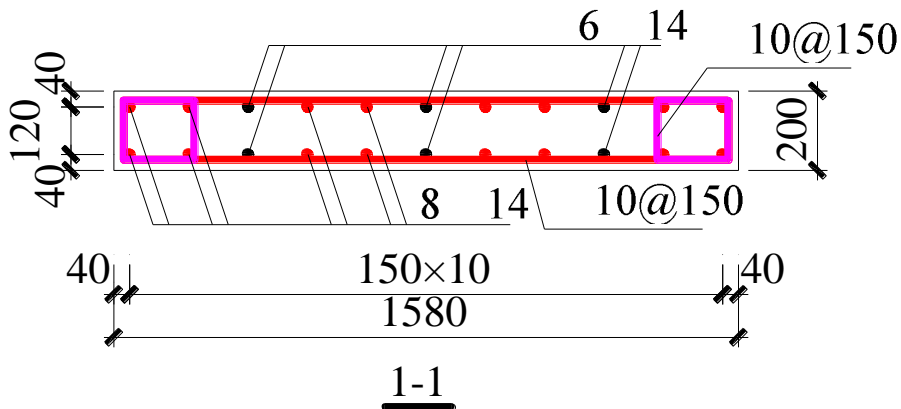


图 4.6 1-1 截面图

在图 4.5 中，截面 2-2 表示包含套筒段的预制剪力墙横截面，这部分的混凝土

土保护层厚度为 20mm，套筒选用球墨铸铁材料制作的全灌浆套筒，由于纵筋选用的是直径为 14mm 的钢筋，参照《装配式混凝土结构技术规程》^[5]，使用的全灌浆套筒外径为 46mm，内径为 34mm，套筒总长为 280mm，下部接触梁中预留钢筋伸入套筒长度为 130mm，上部纵筋伸入套筒长度为 130mm，2-2 截面图如图 4.7 所示。

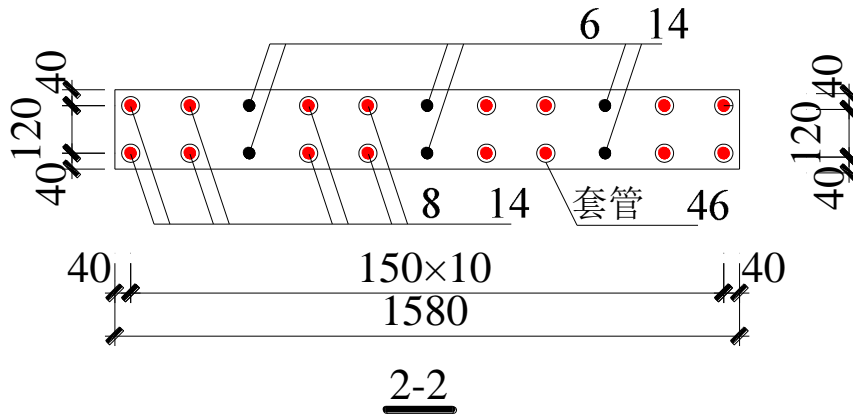


图 4.7 2-2 截面图

预制剪力墙构件和预制基础梁构件均由同济大学实验室预制完成，制作完成的预制剪力墙和预制基础梁如图 4.8 所示，



(a) 预制剪力墙构件

(b) 预制基础梁构件

图 4.8 预制构件

预制构件养护完成之后，需要在实验室对其进行组装，构件的组装过程如图 4.9 所示。



图 4.9 构件组装过程

在本试验中,基础梁和剪力墙之间的钢筋连接采用了全灌浆套筒的连接方式,灌浆过程如图 4.10 所示。



图 4.10 灌浆过程

装配完成的剪力墙如图 4.11 所示。



图 4.11 装配式剪力墙

4.2.2 缺陷引入方式

为更直观地验证基于深度学习的装配式剪力墙缺陷识别方法的有效性,本次试验需引入一个无缺陷的剪力墙和一个存在灌浆缺陷的剪力墙进行对照试验,并将无缺陷剪力墙和存在灌浆缺陷的剪力墙都划分成4个板带,之后的研究都以板带作为基本单元进行分析,板带的划分如图4.12所示。

鉴于框架结构同样的理由,剪力墙中存在灌浆质量缺陷的套筒同样设计为不进行灌浆施工。本次试验在灌浆缺陷剪力墙中共设计4种不同的灌浆缺陷程度,其中板带1中有3个套筒未灌浆、板带2中有1个套筒未灌浆、板带3中有2个套筒未灌浆、板带4中有4个套筒未灌浆。而无缺陷剪力墙的4个板带中的所有套筒都进行了灌浆施工。无缺陷剪力墙和缺陷剪力墙中各板带套筒灌浆数量对比图如图4.13所示(图中空心圆点表示未灌浆的套筒)。

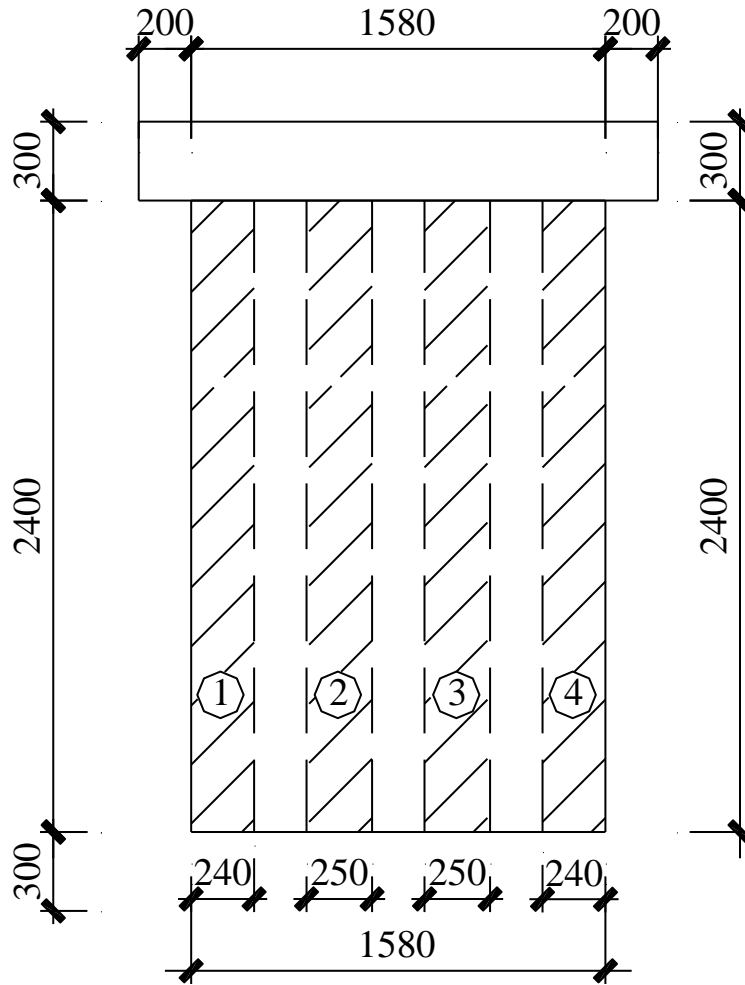
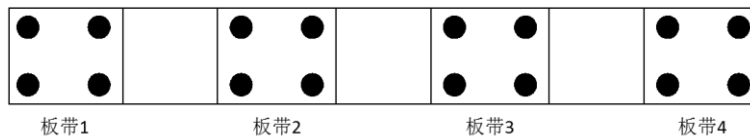
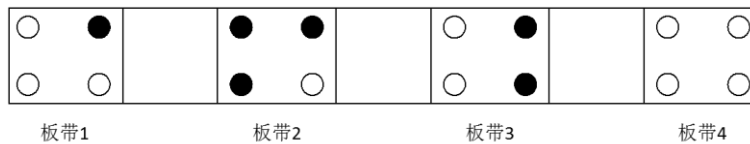


图 4.12 板带划分示意图



(a) 无缺陷剪力墙套筒灌浆分布图



(b) 灌浆缺陷剪力墙套筒灌浆分布图

图 4.13 各板带中套筒灌浆数量对比图

为丰富试验中装配式剪力墙中缺陷的类型以及进一步加大缺陷的程度，在对无缺陷剪力墙和灌浆缺陷剪力墙做完动力测试之后，根据灌浆缺陷剪力墙中各板带未灌浆套筒的个数，分别在两片剪力墙的坐浆层中凿开对应数量的贯穿孔洞来

创造全新的缺陷类型。例如由于灌浆缺陷剪力墙的 1 板带中有 3 个套筒未灌浆，因此在无缺陷剪力墙 1 板带下面的坐浆层中开凿 3 个贯穿孔洞来模拟坐浆层有缺陷的 1 板带，同样地在灌浆缺陷剪力墙 1 板带下面的坐浆层中开凿 3 个贯穿孔洞来模拟坐浆层和套筒灌浆均有缺陷，其他板带以此类推。坐浆层中开凿贯穿孔洞数量如图 4.14 所示（图中实线矩形框表示原始无灌浆缺陷剪力墙的坐浆层，虚线矩形框表示原始有灌浆缺陷剪力墙的坐浆层，同心圆表示在坐浆层中开凿的贯穿孔洞）。

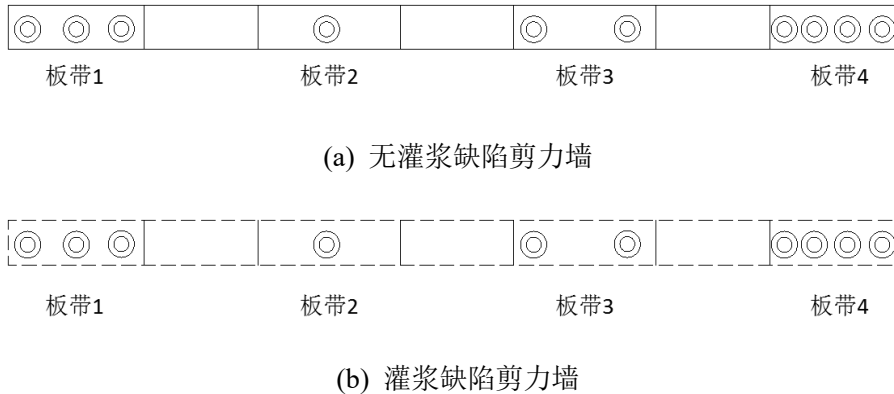


图 4.14 坐浆层中开凿贯穿孔 S 洞示意图

剪力墙中人为引入缺陷汇总如表 4.2 所示。

表 4.2 剪力墙缺陷汇总表

剪力墙缺陷程度	板带 1	板带 2	板带 3	板带 4
无缺陷	0+0	0+0	0+0	0+0
灌浆缺陷	3+0	1+0	2+0	4+0
坐浆层缺陷	0+3	0+1	0+2	0+4
灌浆缺陷以及坐浆层缺陷	3+3	1+1	2+2	4+4

表中加法公式的含义解释如下：公式整体表示板带的缺陷水平，加号前的数字表示剪力墙中的未灌浆套筒的数量，加号后的数字表示坐浆层中开凿孔洞的数量。因此表中板带 1 列中的“3+0”表示板带 1 在该工况下存在的缺陷包括 3 个套筒未灌浆和坐浆层未开凿孔洞，其他公式以此类推。

4.2.3 激振方式与激振点布置

为控制锤击力度，本试验采用单摆锤击的方式，将 870g 重的不锈钢小球制作成一个长为 500mm 的单摆对板带进行锤击，如图 4.15 所示。本次试验为保证每次激振时的激振点和激振幅度相同，在每个板带的中央并且距离剪力墙底部

1700mm 处布置挂点用以固定单摆上端，因此每个板带的激振点距离剪力墙底部为 1200mm，各板带挂点和激振点如图 4.16 所示。激振过程中通过控制小球与剪力墙之间的距离来保证激振的一致性，本次试验中小球与剪力墙之间的距离控制为 200mm。试验中测点响应数据的采集采用加速度传感器，数据采集系统采用北京东方振动和噪声技术研究所配套的采集系统，采集频率为 1024Hz。

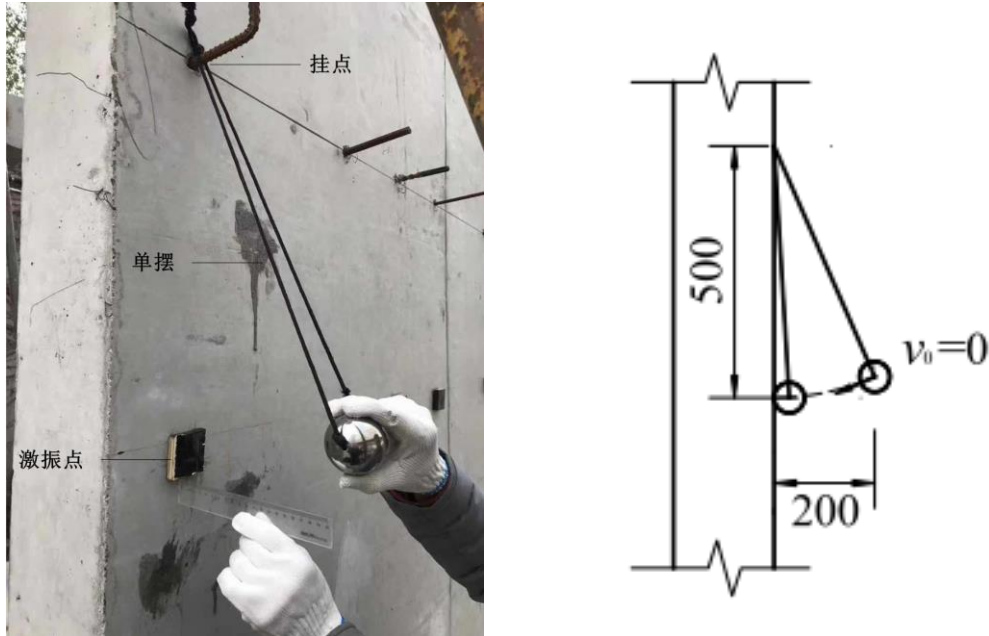


图 4.15 单摆锤击

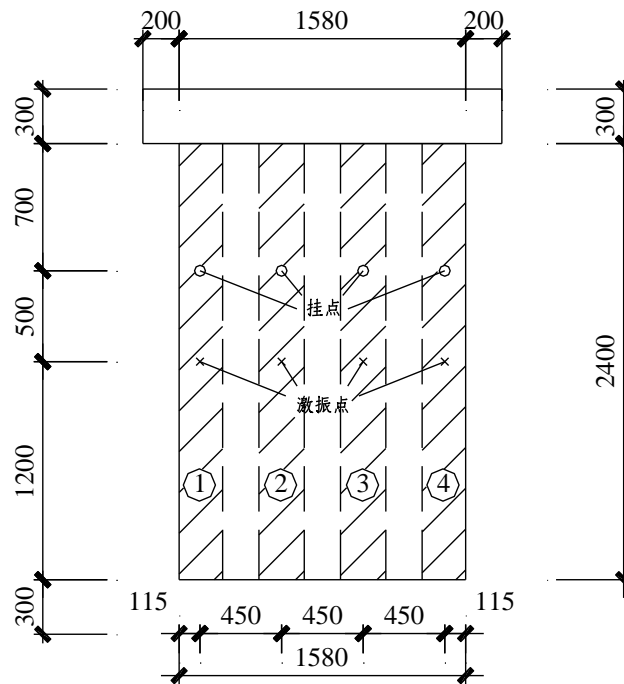


图 4.16 各板带挂点和激振点布置图

4.2.4 各工况测点布置


根据各板带边界条件和剪力墙的缺陷水平,本次试验共分为16个不同的工况,其中工况1-4分别对无缺陷剪力墙中的4个板带进行动力测试,工况5-8则分别对灌浆缺陷剪力墙的4个板带进行动力测试,工况9-12则分别对坐浆层缺陷剪力墙的4个板带进行动力测试,工况13-16则分别对灌浆缺陷加坐浆层缺陷剪力墙的4个板带进行动力测试。在每个工况中,激振点均位于板带的中央,每次激振都会采集该板带下方接近灌浆套筒的加速度响应。例如工况1是对无缺陷剪力墙中的板带1进行激振,工况5是对灌浆缺陷剪力墙中的板带1进行激振,16个工况的划分详情如表4.3所示。

表4.3 工况汇总表(表中公式的含义如上文所述)

剪力墙状态	工况	板带	缺陷水平
无缺陷	工况1	板带1	0+0
	工况2	板带2	0+0
	工况3	板带3	0+0
	工况4	板带4	0+0
灌浆缺陷	工况5	板带1	3+0
	工况6	板带2	1+0
	工况7	板带3	2+0
	工况8	板带4	4+0
坐浆层缺陷	工况9	板带1	0+3
	工况10	板带2	0+1
	工况11	板带3	0+2
	工况12	板带4	0+4
灌浆缺陷加坐浆层缺陷	工况13	板带1	3+3
	工况14	板带2	1+1
	工况15	板带3	2+2
	工况16	板带4	4+4

本试验采用加速度传感器采集测点处的加速度响应,每个工况均布置4个测点,这4个测点均匀地分布在板带中央,由下到上4个测点的编号依次为1、2、3、4,其中测点1位于剪力墙底部,测点布置间距为300mm。以工况1为例对激振点和测点的布置与编号进行说明,如图4.17所示。其余各工况的测点和激振点布置方式等均与工况1一致,在此不展开赘述。图中剪力墙上的加速度传感器测量方向均平行于地面,试验所用数据采集仪由同济大学试验室提供,所用加速度传感器由扬州科动公司提供,其灵敏度等参数如表4.4所示。

表 4.4 各测点加速度传感器参数

测点号	标定值 (mV/m/s ²)	增益	量程 (±/m/s ²)	输入耦合	 加速度传感器
1	982.7	10	1.02	ICP	
2	1050.2	10	0.95	ICP	
3	992	10	1.01	ICP	
4	1068.3	10	0.94	ICP	

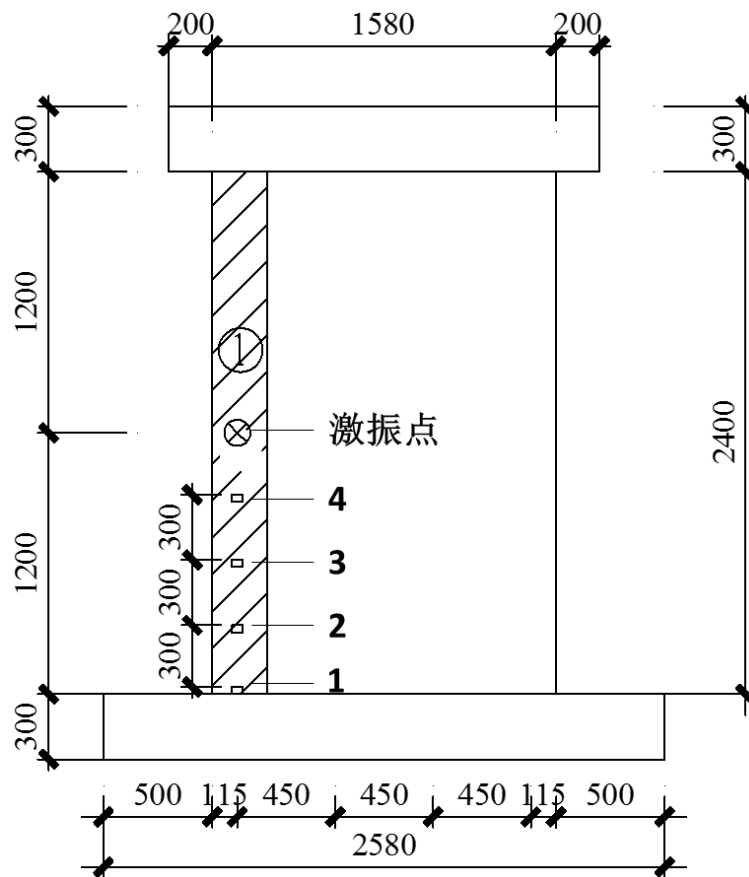


图 4.17 工况 1 激振点及测点布置图

4.2.5 试验过程

试验过程分为三个步骤：

- (1) 在挂点处布置激振单摆；
- (2) 在板带的各测点处布置加速度传感器；

(3) 在激振点激振并同时采集各测点处的加速度时程；每个工况都激振 50 次，每次激振采集时长为 4s，因此每次激振中每个测点处会采集到 4096 个信号，故每个工况都会采集得到 50 组加速度响应，每组响应中都有 4 条包含 4096 个信

号的加速度时程。

以工况 1 为例，试验过程如图 4.18 所示。

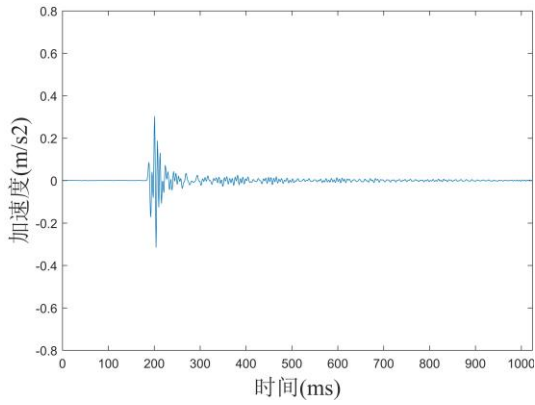


图 4.18 工况 1 试验过程

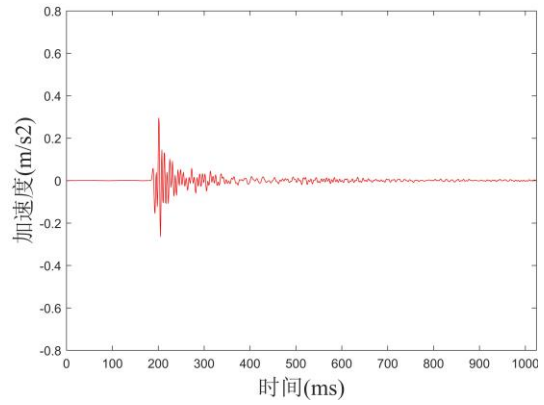
4.3 试验结果分析

4.3.1 加速度时程分析

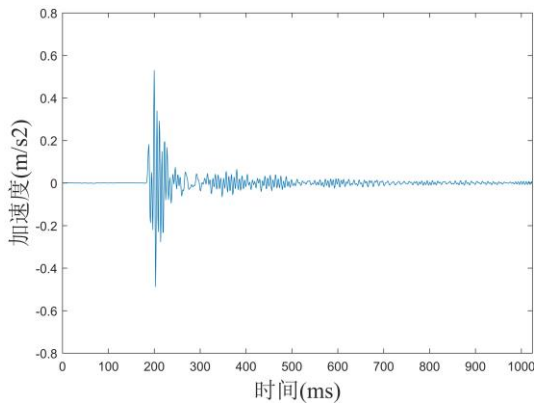
试验中采集得到了各工况下个测点的加速度响应，首先对时程曲线进行对比分析。由于无缺陷剪力墙和灌浆缺陷剪力墙中对应的板带需要进行对比分析，因此选取工况 1 和工况 5 中代表性激振中各测点的加速度响应时程进行分析，其中工况 1 的测试对象为无缺陷剪力墙中的板带 1，工况 5 的测试对象为缺陷剪力墙中的板带 1。工况 1 和工况 5 中各测点的加速度时程如所示。



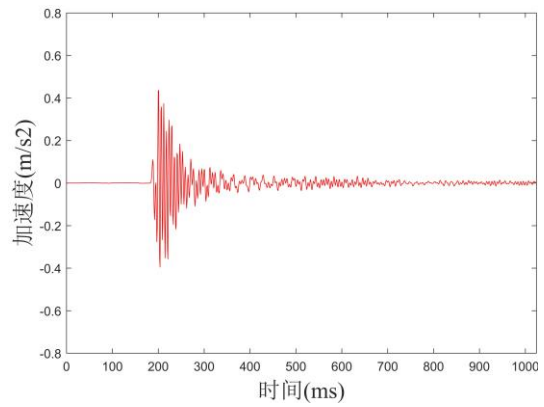
(a) 工况 1 测点 1 代表性加速度时程曲线



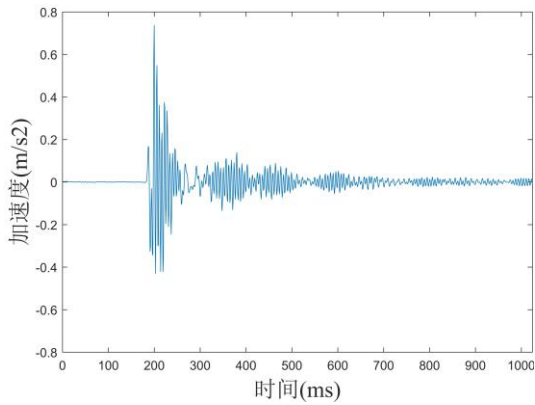
(b) 工况 5 测点 1 代表性加速度时程曲线



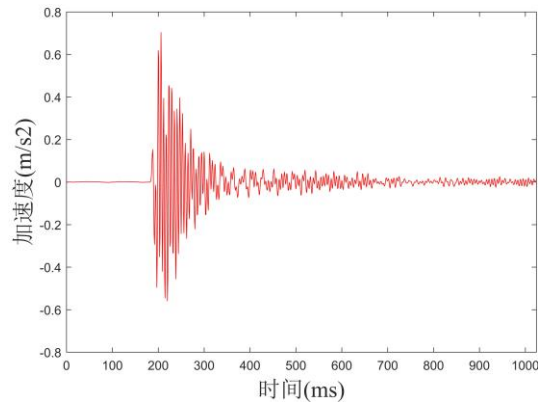
(c) 工况 1 测点 2 代表性加速度时程曲线



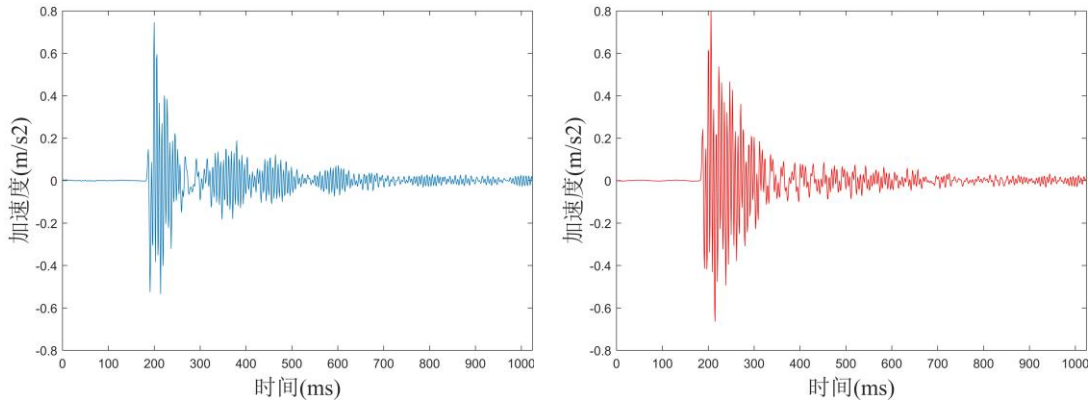
(d) 工况 5 测点 2 代表性加速度时程曲线



(e) 工况 1 测点 3 代表性加速度时程曲线



(f) 工况 5 测点 3 代表性加速度时程曲线



(g) 工况 1 测点 4 代表性加速度时程曲线 (h) 工况 5 测点 4 代表性加速度时程曲线

图 4.19 工况 1 和工况 5 各测点代表性加速度时程曲线

通过对比工况 1 中不同测点的加速度时程响应曲线可以发现, 剪力墙上测点 3、4 的加速度时程响应曲线在幅值和形式上都非常接近, 与其相比, 测点 2 的加速度时程响应曲线在幅值上有一定的下降, 而测点 1 的加速度时程响应曲线在幅值上下降更为剧烈。同样对比工况 5 中不同测点的加速度时程响应曲线可以发现同样的规律。对比工况 1 和工况 5 中同样测点的加速度时程曲线可以发现: 各测点的加速度时程曲线在形式上有较大的差异, 但是在幅值上较为接近, 说明响应时程的形式对缺陷的敏感度强于响应时程的幅值。

4.3.2 数据前处理

由于单摆锤击造成的墙体加速度响应会迅速衰减, 且为保证将激振造成的响应完整采集, 每次采集过程中都会在激振前采集部分环境噪声。因此, 需要从完整信号中截取出墙体加速度最为显著的部分。经过分析大量的加速度时程曲线, 本文将从每组试验采集的完整信号中截取时长为 1s 的片段 (每个测点都有 1024 个信号) 作为后续分析的基础, 以工况 1 中的测点 2 为例, 数据截取过程如图 4.20 所示。

装配式剪力墙动力检测试验采集到大量的加速度时程, 根据节中对各工况中加速度时程曲线的分析可知, 虽然每个工况中各测点的加速度时程曲线在整体形式上非常接近, 但是不同工况以及不同测点间的加速度幅值具有较大差异。因此在利用卷积神经网络进行灌浆缺陷识别之前, 需要把所有加速度时程的幅值调整一致, 即对其做幅值归一化的前处理。参考上文, 根据式(4.1)对试验中采集到的所有加速度时程做幅值归一化的前处理。

$$y_i = \frac{x_i}{\max_{i=1,2,\dots,N} (|x_i|)} \quad (4.1)$$

式中：

x 表示原始加速度时程；

y 表示幅值归一化之后的加速度时程。

这样的前处理操作仅对加速度的幅值做了统一的调整，同时也保证处理之后的加速度时程与原始时程在形式上保证一致，不会导致丢失过多的信息；同时将所有测点加速度时程的幅值调整一致也保证了后续卷积神经网络训练过程中的稳定性，减少由于测点间加速度时程幅值相差过大而造成的收敛速度过慢甚至不收敛的现象。后续的相关操作，诸如数据集搭建、卷积神经网络的训练与测试以及装配式剪力墙结构的缺陷识别均会采用幅值归一化之后的加速度时程，而不会直接采用原始的加速度时程。

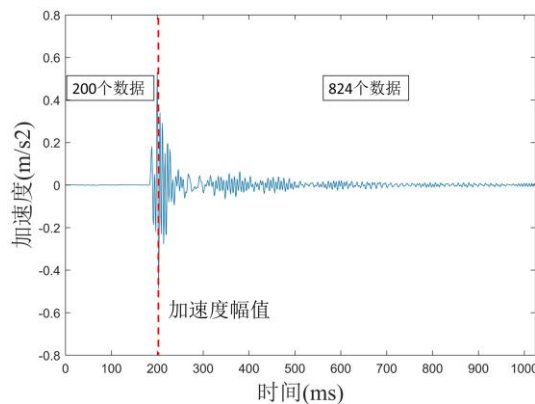


图 4.20 数据截取过程示意图

4.3.3 数据增强及数据集创建

为了使卷积神经网络具有更好的泛化性能，需要获取更多的数据样本，但是在试验中获取的数据非常有限。解决这个问题的一种方法是在真实数据集的基础上创建人造数据，并将其添加到训练集中，这就是“数据增强”的含义。典型的数据增强有如下方法：对于图像识别而言，包括了从原始图像中随机剪切一块，随机旋转小角度，水平翻转，调整对比度等方法^[82]。对于语音识别任务，在输入中加入噪声也是一种提高鲁棒性的数据增强方法^[83]。对于加速度时程而言，Lin 等人通过对原始加速度时程进行随机采样、重新组合的方式实现数据增强的目的^[44]。借鉴该方法，本文做出如下的数据增强操作：

(1) 从每个工况的 50 组加速度时程中随机挑选 4 组，每组都有 4 个测点的加速度时程；

(2) 从 (1) 中的 4 组时程中随机挑选 1 个不重复测点的加速度时程，因此可以得到 4 个不同测点的时程；

(3) 将 (2) 中 4 个测点的加速度时程按照测点的顺序组合成一个全新的时

程，这样即完成了一次数据增强。

数据增强的操作流程如图 4.21 所示。

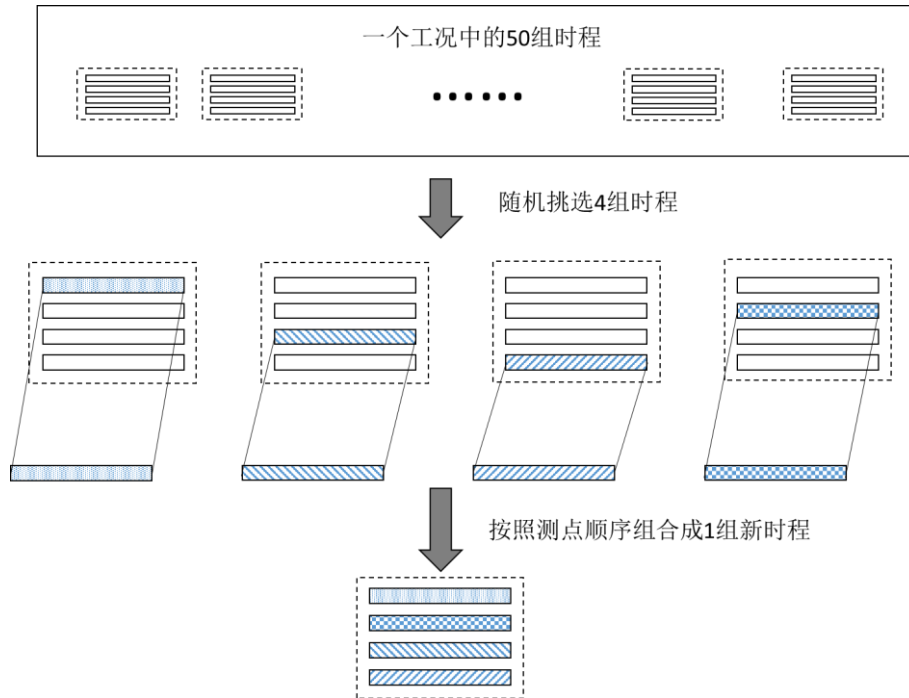


图 4.21 数据增强流程图

经过数据前处理之后，每个工况均得到 50 个大小相同的样本，每个样本中都有 4 条加速度时程，每条时程均包含 1024 个信号。利用上述数据增强技术分别将所有工况中的样本数扩充至 1000 个为后续数据集的创建提供基础。

由于剪力墙结构的缺陷识别全部以板带作为研究的基础，故本节会把同一板带在不同工况下得到的样本组合成一个数据集，共产生 4 个不同的数据集。每个数据集中都有 4000 个样本，包含了 1000 个无缺陷的样本、1000 个仅灌浆缺陷的样本、1000 个仅坐浆层缺陷的样本以及 1000 个灌浆缺陷以及坐浆层缺陷的样本。与上文一致，随机挑选每个数据集中 80% 的样本，共 3200 个样本（4 种缺陷样本类型等比例选取）作为训练集；剩余的 800 个样本则充当测试集。4 个板带的数据集中样本个数如表 4.5 所示。

表 4.5 各板带数据集中样本个数汇总

		板带 1	板带 2	板带 3	板带 4
训练集	无缺陷	800	800	800	800
	灌浆缺陷	800	800	800	800
	坐浆层缺陷	800	800	800	800
	灌浆加坐浆层缺陷	800	800	800	800
	汇总	3200	3200	3200	3200
测试集	无缺陷	200	200	200	200
	灌浆缺陷	200	200	200	200
	坐浆层缺陷	200	200	200	200
	灌浆加坐浆层缺陷	200	200	200	200
	汇总	800	800	800	800

4.3.4 各板带缺陷识别

(1) 样本标签设置

本节中所有的分析以板带为基本单元,与上节中设置的样本标签采用同样的策略,用向量作为每个板带中样本的输出。本节中样本标签设置为包含 4 个元素的向量,4 个元素都分别代表该样本缺陷类型的概率,即第 1 个元素代表了样本无缺陷的概率,第 2 个元素代表了样本是灌浆缺陷的概率,第 3 个元素代表了样本是坐浆层缺陷的概率,第 4 个元素则代表了样本是灌浆以及坐浆层缺陷的概率。因此,这些样本标签由以下规则生成的:当某样本为无缺陷样本时,标签中的第一个元素取值为 1,其余元素取值均为 0,其他情况以此类推。本节中样本的标签设置如表 4.6 所示。

表 4.6 各板带数据集样本标签汇总表

板带	工况	缺陷程度	样本标签
板带 1	工况 1	无缺陷	[1,0,0,0]
	工况 5	灌浆缺陷	[0,1,0,0]
	工况 9	坐浆层缺陷	[0,0,1,0]
	工况 13	灌浆加坐浆层缺陷	[0,0,0,1]
板带 2	工况 2	无缺陷	[1,0,0,0]
	工况 6	灌浆缺陷	[0,1,0,0]
	工况 10	坐浆层缺陷	[0,0,1,0]
	工况 14	灌浆加坐浆层缺陷	[0,0,0,1]
板带 3	工况 3	无缺陷	[1,0,0,0]
	工况 7	灌浆缺陷	[0,1,0,0]
	工况 11	坐浆层缺陷	[0,0,1,0]
	工况 15	灌浆加坐浆层缺陷	[0,0,0,1]
板带 4	工况 4	无缺陷	[1,0,0,0]
	工况 8	灌浆缺陷	[0,1,0,0]
	工况 12	坐浆层缺陷	[0,0,1,0]
	工况 16	灌浆加坐浆层缺陷	[0,0,0,1]

基于上述标签设置，本节中卷积神经网络各层的特征映射大小如表 4.7 所示。

表 4.7 卷积神经网络结构的详细参数

编号	类型	核尺寸	核数量	步长	输入	输出
C1	卷积层	45	6	1	1024×4	6@980×4
S2	池化层	5	-	1	6@980×4	6@196×4
C3	卷积层	32	12	1	6@196×4	12@165×4
S4	池化层	5	-	1	12@165×4	12@33×4
C5	卷积层	20	12	1	12@33×4	12@14×4
S6	池化层	2	-	1	12@14×4	12@7×4
FC	全连接层	-	-	-	12@7×4	4×1

(2) 系统环境及超参数设置

上节把同一板带在不同工况下得到的样本组合成一个数据集，共产生了 4 个不同的数据集。本节将利用上节中创建的 4 个数据集分别对 4 个卷积神经网络进行训练和测试，本节中对卷积神经网络训练所采用的系统环境与 3.4.3 节中一致，但是相关超参数的设置会有一些的变化，具体超参数设置如表 4.8 所示。

表 4.8 超参数设置

板带	训练集大小	测试集大小	批处理大小	训练代数	迭代次数
板带 1	3200	800	5	50	32000
板带 2	3200	800	5	50	32000
板带 3	3200	800	5	50	32000
板带 4	3200	800	5	50	32000

(3) 训练过程

在 4 个板带的缺陷识别中，卷积神经网络训练过程中的 Lost 函数下降图如图 4.22 所示。

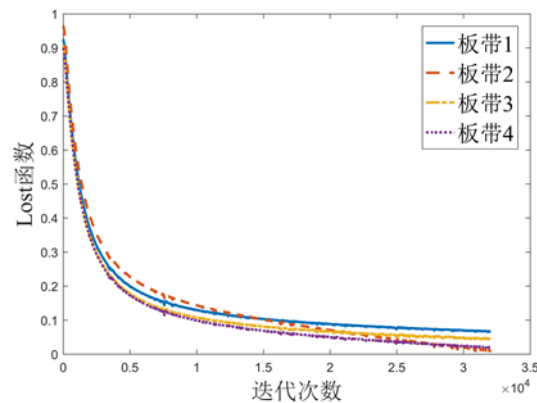


图 4.22 卷积神经网络训练过程中代价函数下降图

从图 4.22 中可以看出，4 个板带的卷积神经网络中的代价函数随着迭代次数的增加而不断减少，并最终实现收敛。同时，4 个代价函数的收敛速度非常接近。说明 4 个板带的卷积神经网络的训练过程较为理想。

(4) 识别结果判断依据

本节以一个板带为研究基础，对板带的 4 种不同状态（无缺陷、灌浆缺陷、坐浆层缺陷、灌浆以及坐浆层缺陷）进行识别。本节将沿用 3.3.4 节中定义的识别依据进行后续的板带缺陷的识别。即分别计算样本输出向量与四种不同状态样本标签间的欧式距离，然后根据 4 个欧式距离的大小进行剪力墙的缺陷识别。最后同样利用混淆矩阵对识别效果进行评价，关于混淆矩阵相关内容的介绍见 3.3.4 节。

随机从 4 个板带的测试集中各自挑选 4 个样本，各自的输出向量如图 4.23 所示，以板带 3 中典型样本输出与四种样本标签间的欧式距离如表 4.9 所示。

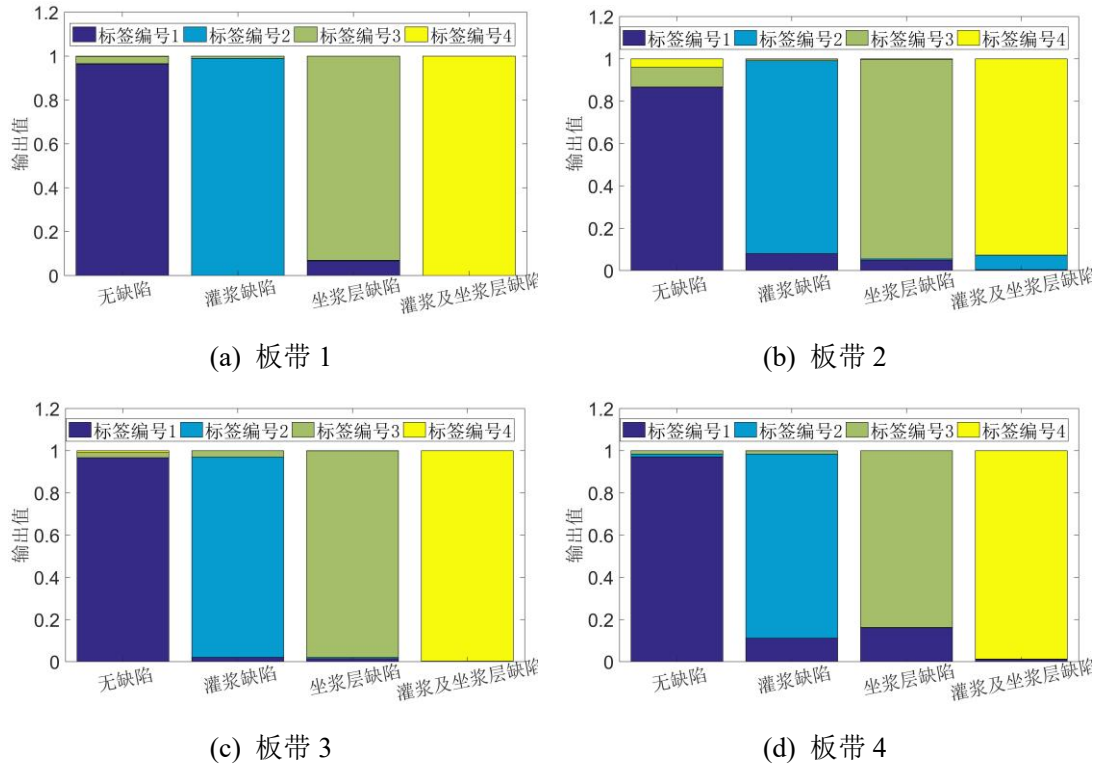


图 4.23 各板带典型样本输出

表 4.9 板带 3 中典型样本识别结果

样本真实缺陷	输出向量	参考基准	欧式距离	识别结果
无缺陷	[0.97,0,0.02,0.01]	无缺陷	0.04	无缺陷
		灌浆缺陷	1.39	
		坐浆层缺陷	1.37	
		灌浆以及坐浆层缺陷	1.38	
灌浆缺陷	[0.02,0.95,0.03,0]	无缺陷	1.37	灌浆缺陷
		灌浆缺陷	0.06	
		坐浆层缺陷	1.36	
		灌浆以及坐浆层缺陷	1.38	
坐浆层缺陷	[0.01,0.01,0.98,0]	无缺陷	1.39	坐浆层缺陷
		灌浆缺陷	1.40	
		坐浆层缺陷	0.03	
		灌浆以及坐浆层缺陷	1.40	
灌浆以及坐浆层缺陷	[0.01,0,0,0.99]	无缺陷	1.41	灌浆以及坐浆层缺陷
		灌浆缺陷	1.41	
		坐浆层缺陷	1.41	
		灌浆以及坐浆层缺陷	0.01	

从图和表中可以看出，板带 3 中的 4 个典型样本的识别结果均与实际缺陷一致，其余 3 个板带中 4 个典型样本的识别结果也均与实际缺陷一致，此处不展开

赘述。下节将对测试集中所有样本的识别结果进行评价。

(5) 识别效果评价

将各板带测试集中的样本依次输入训练完成的卷积神经网络中，根据上文的定义对识别结果进行评价。

板带 1 缺陷类型识别结果的混淆矩阵如表 4.10 所示。

表 4.10 板带 1 混淆矩阵

		真实缺陷				精准率
		无缺陷	灌浆缺陷	坐浆层缺陷	灌浆以及坐浆层缺陷	
识别缺陷	无缺陷	172	6	3	0	95.03%
	灌浆缺陷	3	181	1	7	94.27%
	坐浆层缺陷	9	0	179	2	94.21%
	灌浆以及坐浆层缺陷	1	0	5	189	96.92%
召回率		92.97%	96.79%	95.21%	95.45%	

从表 4.10 中可以看出，无缺陷、灌浆缺陷、坐浆层缺陷和灌浆及坐浆层缺陷的召回率分别为 92.97%、96.79%、95.21%和 95.54%；而无缺陷、灌浆缺陷、坐浆层缺陷和灌浆及坐浆层缺陷的精准率分别为 95.03%、94.27%、94.21 和 96.92%。灌浆缺陷的召回率为 96.79%，说明真实缺陷是灌浆缺陷的 200 个样本中有 96.79% 的样本被准确识别为灌浆缺陷，也有 3.21% 的样本被识别为坐浆层缺陷和灌浆及坐浆层缺陷。同时灌浆缺陷的精准率为 94.27%，说明在被识别为灌浆缺陷的所有样本中，真实缺陷也是灌浆缺陷的样本量占比为 94.27%，真实缺陷是坐浆层缺陷或者灌浆及坐浆层缺陷但被识别为灌浆缺陷的样本量占比为 5.73%，两者的具体比例不是本文关心的重点，此文不在详述。无缺陷、坐浆层缺陷和灌浆及坐浆层缺陷的召回率和精准率的含义与前文一致，此处不在赘述。精准率和召回率均表明，卷积神经网络在板带 1 中四种缺陷类型上的识别性能均位于 90%以上，说明该网络具有较好的识别性能，且召回率和精准率之间未出现显著的差异，说明该网络未表现出偏向性。

板带 2 缺陷类型识别结果的混淆矩阵如表 4.11 所示。

表 4.11 板带 2 混淆矩阵

		真实缺陷				精准率
		无缺陷	灌浆缺陷	坐浆层缺陷	灌浆以及坐浆层缺陷	
识别缺陷	无缺陷	184	0	3	4	96.34%
	灌浆缺陷	8	190	0	5	93.60%
	坐浆层缺陷	0	3	177	0	98.33%
	灌浆以及坐浆层缺陷	0	0	9	187	95.41%
召回率		95.83%	98.45%	93.65%	95.41%	

从表 4.11 中可以看出,无缺陷、灌浆缺陷、坐浆层缺陷和灌浆及坐浆层缺陷的召回率分别为 95.83%、98.45%、93.65%和 95.41%;而无缺陷、灌浆缺陷、坐浆层缺陷和灌浆及坐浆层缺陷的精准率分别为 96.34%、93.60%、98.33%和 95.41%。灌浆缺陷的召回率为 98.45%,说明真实缺陷是灌浆缺陷的 200 个样本中有 98.45% 的样本被准确识别为灌浆缺陷,也有 1.55% 的样本被识别为坐浆层缺陷和灌浆及坐浆层缺陷。同时灌浆缺陷的精准率为 93.60%,说明在被识别为灌浆缺陷的所有样本中,真实缺陷也是灌浆缺陷的样本量占比为 93.60%,真实缺陷是坐浆层缺陷或者灌浆及坐浆层缺陷但被识别为灌浆缺陷的样本量占比为 6.4%,两者的具体比例不是本文关心的重点,此文不在详述。无缺陷、坐浆层缺陷和灌浆及坐浆层缺陷的召回率和精准率的含义与前文一致,此处不在赘述。召回率表明,卷积神经网络在板带 2 中三种缺陷类型上的识别性能均位于 95%左右,说明该网络具有较好的识别性能;且召回率和精准率之间未出现显著的差异,说明该网络未表现出偏向性。

板带 3 缺陷类型识别结果的混淆矩阵如表 4.12 所示。

表 4.12 板带 3 混淆矩阵

		真实缺陷				精准率
		无缺陷	灌浆缺陷	坐浆层缺陷	灌浆以及坐浆层缺陷	
识别缺陷	无缺陷	164	0	12	0	93.18%
	灌浆缺陷	0	192	0	16	92.31%
	坐浆层缺陷	12	0	176	0	93.62%
	灌浆以及坐浆层缺陷	0	4	0	184	97.87%
召回率		93.18%	97.96%	93.62%	92.00%	

从表 4.12 中可以看出，无缺陷、灌浆缺陷、坐浆层缺陷和灌浆及坐浆层缺陷的召回率分别为 93.18%、97.96%、93.62%和 92%；而无缺陷、灌浆缺陷、坐浆层缺陷和灌浆及坐浆层缺陷的精准率分别为 93.18%、93.62%、92.16%和 97.87%。灌浆缺陷的召回率为 97.96%，说明真实缺陷是灌浆缺陷的 200 个样本中有 97.96% 的样本被准确识别为灌浆缺陷，也有 2.04% 的样本被识别为坐浆层缺陷和灌浆及坐浆层缺陷。同时灌浆缺陷的精准率为 93.62%，说明在被识别为灌浆缺陷的所有样本中，真实缺陷也是灌浆缺陷的样本量占比为 93.62%，真实缺陷是坐浆层缺陷或者灌浆及坐浆层缺陷但被识别为灌浆缺陷的样本量占比为 6.38%，两者的具体比例不是本文关心的重点，此文不在详述。无缺陷、坐浆层缺陷和灌浆及坐浆层缺陷的召回率和精准率的含义与前文一致，此处不在赘述。召回率表明，卷积神经网络在板带 3 中四种缺陷类型上的识别性能均位于 90%左右，说明该网络具有较好的识别性能；且召回率和精准率之间未出现显著的差异，说明该网络未表现出偏向性。

板带 4 缺陷类型识别结果的混淆矩阵如表 4.13 所示。

表 4.13 板带 4 混淆矩阵

		真实缺陷				精准率
		无缺陷	灌浆缺陷	坐浆层缺陷	灌浆以及坐浆层缺陷	
识别缺陷	无缺陷	185	1	5	4	94.87%
	灌浆缺陷	3	172	0	1	97.73%
	坐浆层缺陷	9	0	167	3	93.30%
	灌浆以及坐浆层缺陷	0	12	8	188	90.38%
召回率		93.91%	92.97%	92.78%	95.92%	

从表 4.13 中可以看出，无缺陷、灌浆缺陷、坐浆层缺陷和灌浆及坐浆层缺陷的召回率分别为 93.91%、92.97%、92.78%和 95.92%；而无缺陷、灌浆缺陷、坐浆层缺陷和灌浆及坐浆层缺陷的精准率分别为 94.87%、97.73%、93.30%和 90.38%。灌浆缺陷的召回率为 92.97%，说明真实缺陷是灌浆缺陷的 200 个样本中有 92.97% 的样本被准确识别为灌浆缺陷，也有 7.03% 的样本被识别为坐浆层缺陷和灌浆以及坐浆层缺陷。同时灌浆缺陷的精准率为 97.73%，说明在被识别为灌浆缺陷的所有样本中，真实缺陷也是灌浆缺陷的样本量占比为 97.73%，真实缺陷是坐浆层缺陷或者灌浆及坐浆层缺陷但被识别为灌浆缺陷的样本量占比 2.27%，两者的具体比例不是本文关心的重点，此文不在详述。坐浆层缺陷和灌浆及坐浆层缺陷的召回率和精准率的含义与前文一致，此处不在赘述。召回率表明，卷积神经网络

络在板带 4 中四种缺陷类型上的识别性能均超过 90%，说明该网络具有较好的识别性能；且召回率和精准率之间未出现显著的差异，说明该网络未表现出偏向性。

综合上述从 4 个板带的混淆矩阵可以看出，基于深度学习的装配式混凝土结构缺陷识别方法在剪力墙的各板带缺陷识别中的表现虽然略逊色于框架结构，但是其依然表现出稳定、良好的识别性能，且对不同的缺陷类型，该方法未表现为明显的偏向性。

4.4 本章小结

为进一步验证基于深度学习的装配式混凝土结构缺陷识别方法的有效性，本章设计出两个足尺装配式剪力墙模型，在两个剪力墙与基础梁的连接处人为引入灌浆缺陷、坐浆层缺陷和灌浆及坐浆层缺陷等三种不同的缺陷类型，通过摆锤激励的方法获取剪力墙的加速度响应。最后采用上述方法完成剪力墙缺陷的缺陷识别。主要内容与结论如下：

(1) 系统地介绍了该试验的试验方案：试验对象为两个足尺装配式剪力墙模型，在两个剪力墙与基础梁的连接处人为引入灌浆缺陷、坐浆层缺陷和灌浆及坐浆层缺陷等三种不同的缺陷类型。每个剪力墙模型均分为 4 个板带，通过摆锤激励的方法获取各板带的加速度响应；

(2) 结合第三章内容建立卷积神经网络，利用数据增强技术对幅值归一化后的加速度响应进行扩充，同时根据不同缺陷类型定义样本输出标签，将各板带的样本和输出标签组成数据集；

(3) 利用上述方法对剪力墙 4 个板带中的不同缺陷类型进行识别，并通过混淆矩阵对该方法的识别效果进行评价，结果显示该方法在装配式剪力墙结构 4 个板带不同缺陷类型的识别中具有良好的性能，同时对剪力墙的四种不同缺陷状态未表现出明显的偏向性。

第5章 结论与展望

5.1 结论

本文将深度学习与装配式混凝土结构缺陷识别相结合,提出了基于深度学习的装配式混凝土结构缺陷识别方法。该方法利用卷积神经网络对装配式混凝土结构的局部响应进行分析和特征提取,从而实现装配式混凝土结构的缺陷识别。为验证该方法的可行性、有效性和准确性,设计装配式框架结构和装配式剪力墙的动力检测试验。主要成果与结论如下:

1.梳理深度学习技术基础,并重点对其中卷积神经网络的基本结构、工作原理和关键算法等进行研究,为基于深度学习的装配式混凝土结构灌浆缺陷识别的研究提供理论基础。

2.建立基于深度学习的装配式混凝土结构缺陷识别方法,该方法利用卷积神经网络直接提取装配式混凝土结构动力响应中的特征,并通过全连接层将其映射成缺陷概率作为输出,从而实现装配式混凝土结构的缺陷识别。参考经典卷积神经网络,对其中的关键函数(代价函数、激活函数和优化算法)进行比较选优,最终设计出该方法中的卷积神经网络。

3.设计缩尺装配式混凝土框架动力检测试验,通过现场激振获取结构的局部加速度响应,利用基于深度学习的装配式混凝土结构缺陷识别方法进行装配式框架结构的缺陷识别,主要成果与结论如下:

(1)利用上述识别方法实现装配式混凝土框架结构缺陷及其位置的识别,结果表明该方法在框架结构动力检测试验中的7种工况的缺陷识别准确率均为100%,即该方法在装配式框架结构缺陷识别中具有良好的性能。

(2)利用上述识别方法实现装配式混凝土框架结构中三种不同缺陷程度的识别,并通过混淆矩阵对三种不同缺陷程度的识别效果进行评价,结果显示该方法在装配式框架结构不同缺陷程度的识别中具有良好的性能,同时对三种不同缺陷程度未表现出明显的偏向性。

4.设计足尺装配式混凝土剪力墙检测试验,通过现场激振获取剪力墙的局部加速度响应,利用基于深度学习的装配式混凝土结构缺陷识别方法进行装配式剪力墙的缺陷识别,主要成果与结论如下:

(1)由于试验中采集的加速度响应有限,利用数据增强技术对采集到的加速度进行扩充,并建立数据集用于后续分析;

(2) 利用上述识别方法实现装配式混凝土剪力墙四种不同缺陷状态（无缺陷、灌浆缺陷、坐浆层缺陷和灌浆以及坐浆层缺陷）的识别，通过混淆矩阵对该方法的识别效果进行评价，结果表明虽然该方法在装配式剪力墙的识别效果略逊色于框架结构，但是其依然表现出良好稳定的识别效果，并且对剪力墙的四不同缺陷状态未表现出明显的偏向性。

综上，本文将土木工程与新兴技术相集合，建立基于深度学习的装配式混凝土结构缺陷识别方法，并通过装配式框架结构和装配式剪力墙的动力检测试验分别从缺陷存在性识别、缺陷位置识别以及缺陷程度识别等几个层面验证了该方法的可行性、有效性以及准确性。

5.2 进一步工作的方向

本文结合卷积神经网络，进行了全灌浆套筒连接剪力墙连接缺陷的检测研究，获得了一些初步成果，检测效果达到了较为满意的结果。但由于作者能力和时间有限，仍然留有一些问题值得探讨。

(1) 本文仅探讨了应用范围较广的全灌浆套筒连接的缺陷检测，未研究其他的连接形式，如半灌浆套筒连接，浆锚连接以及螺栓连接等，这些连接方式在实际工程中也用一定程度的应用，有必要进行探讨研究。

(2) 本文中的缺陷引入方式是将部分套筒不灌浆来实现的，实际工程中，还存在着灌浆不充分导致的缺陷，以及由于座浆层处的不良施工所带来的缺陷，对于这些原因引起的缺陷，有待于进一步地研究。

(3) 缺陷检测完成后，对于缺陷的补救措施有许多，如针孔补灌，经试验证明针孔补灌后的装配式结构性能良好，可作为有效补救措施之一。本文未对此类补灌措施进行深入研究，有关此类的技术研究可做进一步探讨。

(4) 检测过程较为繁琐，且数据分析过程需大量样本支撑。受作者能力和时间限制，该方法仅在实验室进行了试验，而未能在实际工程中进行验证，因此，其在实际工程中的验证需要做进一步的研究。

(5) 受作者能力和时间限制，本文以卷积神经网络进行缺陷识别，若能结合其他深度学习网络的特点增加网络深度，有望进一步提升识别效果。

(6) 受作者能力和时间限制，本文仅对网络的输出进行分析，网络内部各层的学习状态依然不能可视化。因此将每一层的学习状态进行可视化，能助力其大范围推广和应用，这是未来研究的重点。

(7) 受作者能力和时间限制，本文对样本标签的设置较为单一化，后续可以做进一步深入的研究。

致谢

凡是过往，皆为序章。

时光荏苒，三年硕士生活已接近尾声，回眸往昔，感慨良多。很幸运能够来到同济大学，她所给予我的不仅仅是学识，更重要的是严谨求实、团结创新的作风和同舟共济、自强不息的精神。在此前后七年的学习生活中，有过苦难，有过彷徨，但更多的是无私的关怀和帮助，让我终生难忘。在此，衷心感谢在同济大学学习期间给予我帮助的老师 and 同学。

首先要衷心感谢我的导师唐和生老师，唐老师学术上认真严谨的态度深深的影响着我，使我在学习研究中能够严格要求自己。唐老师具有很强的人格魅力，工作之外的他和蔼谦逊，年轻有为，博闻强识，运动一流，不仅以老师的身份传道授业解惑，更以前辈的身份教会我生活中的道理，指明我人生的方向。本文能得以完成，从选题到研究方法的选择到最后论文的修改定稿，无一不是倾注了唐老师的心血。在此，对您耐心的指导和无私的帮助表示由衷感谢。

此外，要感谢薛松涛老师，薛老师名扬中外却平易近人，谈笑风生又不断传播乐观积极的人生态度，无微不至的关怀着教研室的每一位同学，在我研究生在读研期间给予了我前往日本东北大学参与国际论坛与考察的机会。还要感谢谢丽宇老师，谢老师在教研室日常活动与科研上给予了我们很大的支持。在此，对教研室三位老师均表示由衷的感谢与崇高的敬意。

感谢办公室茹海琳老师、曹阳老师和程春兰老师在工作 and 生活上给予的关怀和照顾。

感谢教研室毕业的李大伟师兄、陈杉杉师兄、凌塑奇师兄以及混凝土与砌体结构研究室的张璇师姐在论文与试验研究上给予我的指导和帮助。

感谢郭雪媛、班鑫磊、陈依珂师姐在生活和学习等多个方面给予的无私帮助，在研究生的三年时间内，像大姐姐一样关心我的学习、生活和工作；感谢帅气的王云骞、可爱的陈子暘、美丽的薛智奇和优秀的管帅在科研、生活中带来的支持，你们的一路陪伴为我提供了无穷的助力。

感谢博士师兄师姐杨鹏超、仝运佳、李宪之、张力和谢雅娟以及师弟康建飞、易卓然、陈豪、濮昱、杨虎、郭泰坤、姜康、郑志泉、曹跃辉以及师妹吴雅琴、曾璟琳、冉凌霄、黎思维、龚琳。与你们相处我非常开心，感谢你们为整个教研室营造了活泼快乐的氛围。感谢研究室已经毕业的赵金海、陈实、徐康乾、张睿师兄以及林庄慧师姐在科研和生活上给予的关心。

感谢好友张乾翔、王琪琪、景一君、陈彦府、张武杰等，谢谢你们丰富了我

的大学生活，让我毕生难忘。

感谢白梦圆、左佳欣、康乐乐以及刘毓帆在面临重要人生选择却时给予的温柔帮助。

还要感谢歌手林俊杰，他的声音陪伴我走过无数个难以入眠的夜晚。

特别要感谢父母和亲人的养育之恩，寸草之心，无以为报，希望父母亲人安康快乐，顺心顺意。

感谢所有曾经帮助、关心和包容过我的人，愿你们幸福安康、心想事成。

最后在今年这个特殊的日子，向奋战在疫情一线的所有的医护人员和工作人员致敬。

赵涛涛

2020年6月于同济大学樱花树下

参考文献

- [1] 文林峰. 大力发展装配式建筑的重要意义. 建设科技, 2016, Vol. Z1: 36-37
- [2] 王振. 我国预制装配式建筑的现状与发展. 工程技术:文摘版, 2016, Vol.19(10): 288
- [3] 石建光, 林树枝. 预制装配式混凝土建筑的结构体系和生产方式. 厦门科技, 2014, Vol. (01): 43-46
- [4] 吴迪. 装配式建筑节点连接方式研究综述. 中外建筑, Vol. (8): 150-151
- [5] 中国建筑标准设计研究院. JGJ 1-2014 装配式混凝土结构技术规程. 北京: 中国建筑工业出版社, 2014
- [6] 郑永峰, 郭正兴, 曹江. 新型灌浆套筒的约束机理及约束应力分布. 哈尔滨工业大学学报, 2015, Vol.47(12): 106-111
- [7] 郑清林, 王霓, 陶里. 灌浆缺陷对钢筋套筒灌浆连接试件性能影响的试验研究. 建筑科学, 2017, Vol. 33(5): 61-68
- [8] 中国建筑标准设计研究院. GB/T 51231-2016 装配式混凝土建筑技术标准. 北京: 中国建筑工业出版社, 2016
- [9] 李长太. 钢筋套筒连接灌浆密实度检测技术研究. 福建建材, 2018, Vol.000(009): 28-30
- [10] 安琳, 郑亚明. 灌浆空洞 X 射线无损检测的试验研究与应用. 中国土木工程学会混凝土工程耐久性研究和应用研讨会, 2006: 373-381
- [11] 高润东, 李向民, 张富文. 基于 X 射线工业 CT 技术的套筒灌浆密实度检测试验. 无损检测, 2017, Vol.39(04): 6-11+37
- [12] 张富文, 李向民, 高润东. 便携式 X 射线技术检测套筒灌浆密实度研究. 施工技术, 2017, Vol. 46(17): 6-9+61
- [13] 程朝霞, 徐银芳, 王毅翔. 超声波法检测混凝土强度的发展. 华中科技大学学报(城市科学版), 2003, Vol.(4): 95-98
- [14] 潘绍伟, 叶跃忠, 徐全. 钢管混凝土拱桥超声波检测研究. 桥梁建设, 1997, Vol.(1): 32-35.
- [15] 聂东来, 贾连光, 杜明坎. 超声波对钢筋套筒灌浆料密实性检测试验研究. 混凝土, 2014, Vol.(9): 120-123
- [16] 张苏承, 张棋, 蒋瑶瑶. 套筒灌浆密实度检测技术及存在问题. 施工技术, 2018, Vol.47(17): 138-139
- [17] Carino N. J., Sansalone M. & Hsu N. N. Flaw detection in concrete by frequency spectrum analysis of impact-echo waveforms. International advances in nondestructive testing, 1986, Vol. (12): 117-146
- [18] Jaeger B. J., Sansalone M. J. & Poston R W. Using impact-echo to assess tendon ducts. Concrete International, 1997, Vol.19 (2): 42-46
- [19] Olson L. D., Tinkey Y. & Miller P. Concrete bridge condition assessment with impact echo scanning. Emerging Technologies for Material, Design, Rehabilitation, and Inspection of Roadway Pavements, 2011: 59-66
- [20] 庞涛, 栾文彬. 装配整体式剪力墙结构的新型连接节点施工. 建筑施工, 2015, Vol37(5): 568-570

- [21] 刘辉, 李向民, 许清风. 冲击回波法在套筒灌浆密实度检测中的试验. 无损检测, 2017, Vol.39(4): 12-16
- [22] 高润东, 李向民, 王卓琳. 基于预埋钢丝拉拔法的套筒灌浆饱满度检测技术研究. 施工技术, 2017, Vol.46(17): 1-5
- [23] Ye X. W., Jin T. & Yun C. B. A review on deep learning-based structural health monitoring of civil infrastructures. *Smart Structures and Systems*, 2019, Vol.24 (5): 567-585
- [24] LeCun Y., Boser B., Denker J. S., et al. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1989, Vol.1 (4): 541-551
- [25] Elman J. L. Finding structure in time. *Cognitive science*, 1990, Vol.14 (2): 179-211
- [26] Hinton G. E., Osindero S. & Teh Y. W. A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation*, 2006, Vol.18 (7): 1527-1554
- [27] Dahl G. E., Yu D., Deng L., et al. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on audio, speech, and language processing*, 2011, Vol.20 (1): 30-42
- [28] Krizhevsky A., Sutskever I. & Hinton G. E. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*. 2012, Vol.25 (2): 1097-1105
- [29] Sutskever I., Martens J., Dahl G., et al. On the importance of initialization and momentum in deep learning. *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, Atlanta, 2013: 1139-1147
- [30] Schroff F., Kalenichenko D. & Philbin J. Facenet: A unified embedding for face recognition and clustering. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015: 815-823.
- [31] LeCun Y., Bengio Y. & Hinton G. Deep learning. *Nature*, 2015, Vol.521 (7553): 436-444
- [32] Vodrahalli K. & Bhowmik A. K. 3D computer vision based on machine learning with deep neural networks: A review. *Journal of the Society for Information Display*, 2017, Vol.25 (11): 676-694
- [33] Cha Y. J., Choi W. & Büyüköztürk O. Deep learning - based crack damage detection using convolutional neural networks. *Computer - Aided Civil and Infrastructure Engineering*, 2017, Vol.32 (5): 361-378
- [34] Dorafshan S., Thomas R. J. & Maguire M. Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete. *Construction and Building Materials*, 2018, Vol.186: 1031-1045
- [35] Bao Y., Tang Z., Li H., et al. Computer vision and deep learning-based data anomaly detection method for structural health monitoring. *Structural Health Monitoring*, 2019, Vol.18 (2): 401-421
- [36] Tang Z., Chen Z., Bao Y., et al. Convolutional neural network - based data anomaly detection method using multiple information for structural health monitoring. *Structural Control and Health Monitoring*, 2019, Vol.26 (1): e2296
- [37] Ni F. T., Zhang J. & Chen Z. Q. Zernike-moment measurement of thin-crack width in images enabled by dual-scale deep learning. 2019, Vol.34 (5): 367-384
- [38] Xu Y., Wei S., Bao Y., et al. Automatic seismic damage identification of reinforced concrete columns from images by a region-based deep convolutional neural network. *Structural Control*

- & Health Monitoring, 2019, Vol.26 (3): 1-22
- [39] Beckman G. H., Polyzois D. & Cha Y. J. Deep learning-based automatic volumetric damage quantification using depth camera. *Automation in Construction*, 2019, Vol.99: 114-124
- [40] 李雪松, 林逸洲, 马宏伟. 基于卷积神经网络的桥梁损伤识别方法应用研究. *青海大学学报*, 2018, Vol.(2): 41-46
- [41] 李贵凤. 基于深度学习的桥梁健康监测数据分析关键技术研究:[硕士学位论文]. 重庆: 重庆交通大学土木工程学院, 2018
- [42] 谢祥辉, 单德山, 周筱航. 基于堆栈降噪自动编码器的桥梁损伤识别方法. *铁道建筑*, 2018, Vol. 58(05): 1-5
- [43] 赵焯. 基于卷积神经网络的叶片损伤识别方法研究: [硕士学位论文]. 天津: 中国民航大学航空工程学院, 2019
- [44] Lin Y., Nie Z. & Ma H. Structural damage detection with automatic feature - extraction through deep learning. *Computer - Aided Civil and Infrastructure Engineering*, 2017, Vol.32 (12): 1025-1046
- [45] Abdeljaber O., Avci O. & Kiranyaz S. Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks. *Journal of Sound and Vibration*, 2017, Vol.388: 154-170
- [46] Pathirage C. S., Li J. & Li L. Structural damage identification based on autoencoder neural networks and deep learning. *Engineering Structures*, 2018, Vol.172: 13-28
- [47] Rafiei M. H. & Adeli H. A novel unsupervised deep learning model for global and local health condition assessment of structures. *Engineering Structures*, 2018, Vol.156: 598-607
- [48] Lee S., Ha J. & Zokhirova M. Background information of deep learning for structural engineering. *Archives of Computational Methods in Engineering*, 2018, Vol.25 (1): 121-129
- [49] Khodabandehlou H., Pekcan G. & Fadali M. S. Vibration - based structural condition assessment using convolution neural networks. *Structural Control and Health Monitoring*, 2019, Vol.26 (2): e2308
- [50] Duan Y., Chen Q., Zhang H. CNN-based damage identification method of tied-arch bridge using spatial-spectral information. *Smart Structures and Systems*, 2019, Vol.23 (5): 507-520
- [51] Gulgec N. S., Takáč M. & Pakzad S. N. Convolutional neural network approach for robust structural damage detection and localization. *Journal of Computing in Civil Engineering*, 2019, Vol.33 (3): 04019005
- [52] Zhou C., Chase J. G. & Rodgers G. W. Degradation evaluation of lateral story stiffness using HLA-based deep learning networks. *Advanced Engineering Informatics*, 2019, Vol.39: 259-268
- [53] Wang Z., Zheng H. & Li L. Practical multi-class event classification approach for distributed vibration sensing using deep dual path network. *Optics express*, 2019, Vol.27 (17): 23682-23692
- [54] Kim H. & Sim S. H. Automated peak picking using region - based convolutional neural network for operational modal analysis. *Structural Control and Health Monitoring*, 2019, Vol.26 (11): e2436
- [55] Wu R. T. & Jahanshahi M. R. Deep convolutional neural network for structural dynamic response estimation and system identification. *Journal of Engineering Mechanics*, 2019, Vol.145 (1): 04018125
- [56] 蒋艳凰, 赵强利. 机器学习方法. 北京: 电子工业出版社, 2009

- [57] 余凯, 贾磊, 陈雨强. 深度学习的昨天、今天和明天. 计算机研究与发展, 2013, Vol.050(009): 1799-1804
- [58] 尹宝才, 王文通, 王立春. 深度学习研究综述. 北京工业大学学报, 2015, Vol.41(01): 48-59
- [59] 李彦冬, 郝宗波, 雷航. 卷积神经网络研究综述. 计算机应用, 2016, Vol.36(9): 2508-2515
- [60] 何蓉. 基于卷积神经网络的音乐推荐系统: [硕士学位论文]. 南京: 南京邮电大学, 2019
- [61] 王伟伟. 基于深度学习的单目标跟踪算法的设计与实现: [硕士学位论文]. 西安: 西安电子科技大学, 2019
- [62] 蒋勇, 李淼. 基于遗传程序设计优化神经网络激励函数问题研究. 计算机应用与软件, 2008, Vol.(08): 10-11+38.
- [63] 周畅, 米红娟. 深度学习中三种常用激活函数的性能对比研究. 北京电子科技学院学报, 2017, Vol.(4): 27-32
- [64] Mitchell T. M. Machine learning. 1997
- [65] 刘青玉. 基于浅层学习和深度学习的传感器活动识别对比研究: [硕士学位论文]. 秦皇岛: 燕山大学, 2017
- [66] 吴佑寿, 赵明生. 激活函数可调的神经元模型及其有监督学习与应用. 中国科学 E 辑: 技术科学, 2001, Vol.31(3): 263-272
- [67] Ghahramani Z. Unsupervised learning. Summer School on Machine Learning. Springer, Berlin, Heidelberg, 2003: 72-112
- [68] Sathya R. & Abraham A. Comparison of supervised and unsupervised learning algorithms for pattern classification. International Journal of Advanced Research in Artificial Intelligence, 2013, Vol.2 (2): 34-38
- [69] McClelland J. L., Rumelhart D. E. & PDP Research Group. Parallel distributed processing. Explorations in the Microstructure of Cognition, 1986, Vol.2: 216-271
- [70] 孙娅楠, 林文斌. 梯度下降法在机器学习中的应用. 苏州科技大学学报: 自然科学版, 2018, Vol.(2): 26-31
- [71] Ruder S. An overview of gradient descent optimization algorithms. arXiv preprint arXiv, 2016, 1609.04747
- [72] Kingma D. P. & Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv, 2014, 1412.6980
- [73] 吕俊杰. 装配式建筑钢筋套筒灌浆施工中应注意的问题. 建筑工人, 2017, Vol.38(12): 6-8
- [74] Palm R. B. Prediction as a candidate for learning deep hierarchical models of data. Technical University of Denmark, 2012
- [75] 赵宏, 郭万鹏. 深度神经网络代价函数选择与性能评测研究. 软件, 2018, Vol.039(001): 14-20
- [76] 李仲德, 卢向日, 崔桂梅. 数字图像识别的代价函数选择和性能评价. 电光与控制, 2019, Vol.26(12): 44-48+68
- [77] 田娟, 李英祥, 李彤岩. 激活函数在卷积神经网络中的对比研究. 计算机系统应用, 2018, Vol.27(07): 45-51
- [78] Keskar N. S. & Socher R. Improving Generalization Performance by Switching from Adam to SGD. 2017.

- [79] Wilson A. C., Roelofs R. & Stern M. The Marginal Value of Adaptive Gradient Methods in Machine Learning. 2017
- [80] 徐林, 高三杰, 刘卫, 杨波. 无损检测技术的原理及应用. 科技视界, 2014, Vol.8(13): 87
- [81] 徐阳. 基于计算机视觉的桥梁结构局部损伤识别方法研究: [博士学位论文]. 哈尔滨: 哈尔滨工业大学土木工程学院, 2019.
- [82] Wu R., Yan S. & Shan Y. Deep Image: Scaling up Image Recognition. Computer ence, 2015
- [83] Sietsma J., Dow R. J. F. Creating artificial neural networks that generalize. Neural Networks, 1991, Vol.4 (1): 67-79

个人简历、在读期间发表的学术论文与研究成果

个人简历:

赵涛涛, 男, 1995 年 6 月生。

2017 年 6 月毕业于同济大学土木工程专业, 获学士学位。

2017 年 9 月入同济大学土木学院攻读硕士学位。

已发表论文:

[1] Tang H. S., Zhao T. T. & Xue S. T. Experimental Verification of the Local Damage Detection Method Based on Spatial-temporal Correlation in Dense Sensor Networks. 12th International Workshop on Structural Health Monitoring, Stanford, 2019: 1109-1116

[2] Tang H. S., Zhao T. T., Yang G. Q. & Xue S. T. Vehicle-Track System Identification for High-Speed Railway Using a Cross Entropy Approach. 7th Asia-Pacific Workshop on Structural Health Monitoring, Hongkong, 2017: 982-991