



大学生论文检测系统

文本复制检测报告单 (全文标明引文)

No: ADBD2024R_20240603160531477951374875

检测时间: 2024-06-03 16:05:31

篇名: 基于机器视觉的人脸表情识别系统设计
作者: 汪哲文
指导教师: 苏连成
检测机构: 燕山大学教务处
文件名: 汪哲文燕山大学本科毕业设计0603-提交版.doc
检测系统: 大学生论文检测系统
检测类型: 大学生论文
检测范围: 中国学术期刊网络出版总库
中国博士学位论文全文数据库/中国优秀硕士学位论文全文数据库
中国重要会议论文全文数据库
中国重要报纸全文数据库
中国专利全文数据库
图书资源
优先出版文献库
大学生论文联合比对库
互联网资源(包含贴吧等论坛资源)
英文数据库(涵盖期刊、博硕、会议的英文数据以及德国Springer、英国Taylor&Francis 期刊数据库等)
港澳台学术文献库
互联网文档资源
源代码库
CNKI大成编客-原创作品库
机构自建比对库
时间范围: 1900-01-01至2024-06-03

⚠可能已提前检测, 检测时间: 2024/5/26 21:43:24, 检测结果: 7%

检测结果

去除本人文献复制比: 6.6%

跨语言检测结果: 0%

去除引用文献复制比: 5.4%

总文字复制比: 6.6%

单篇最大文字复制比: 0.7% (人脸面部表情识别的研究进展)

重复字数: [1568] 总段落数: [5]
总字数: [23773] 疑似段落数: [5]
单篇最大重复字数: [177] 前部重合字数: [629]
疑似段落最大重合字数: [561] 后部重合字数: [939]
疑似段落最小重合字数: [64]



指标: ☐ 疑似剽窃观点 ☒ 疑似剽窃文字表述 ☐ 疑似整体剽窃 ☐ 过度引用

相似表格: 0 相似公式: 没有公式 疑似文字的图片: 0

3% (68)	3% (68)	中英文摘要等 (总2300字)
18.1% (561)	18.1% (561)	第1章绪论 (总3108字)
14.2% (431)	14.2% (431)	第2章相关技术理论和研究软件 (总3029字)
7.2% (444)	7.2% (444)	第3章表情特征的提取与识别 (总6194字)
0.7% (64)	0.7% (64)	第4章人脸表情识别系统的设计实现 (总9142字)

指导教师审查结果

指导教师： 苏连成

审阅结果:

审阅意见: 指导老师未填写审阅意见

1. 中英文摘要等

总字数: 2300

相似文献列表

去除本人文献复制比: 3%(68) 去除引用文献复制比: 3%(68) 文字复制比: 3%(68) 疑似剽窃观点: (0)

1	基于卷积神经网络的人脸表情识别 曹天欣 - 《大学生论文联合比对库》 - 2023-05-16	3.0% (68) 是否引证: 否
2	基于卷积神经网络的人脸表情识别 曹天欣 - 《大学生论文联合比对库》 - 2023-05-31	3.0% (68) 是否引证: 否

原文内容

燕山大学本科毕业设计(论文)
基于机器视觉的人脸表情识别系统设计

学院： 电气工程学院
专业： 智能科学与技术

姓名：汪哲文

学号：202011030374

指导教师：苏连成

答辩日期：2024年6月

摘要

在人机交互和情感计算领域，人脸表情识别系统的研究和应用具有重要意义。本文提出了一种基于机器视觉的人脸表情识别系统，采用了Xception模型及其简化版本Mini-X模型进行表情特征提取与分类。首先，本文介绍了相关技术，包括支持向量机（SVM）和卷积神经网络（CNN），并详细讨论了OpenCV图像处理库和Qt Designer界面设计工具的应用。其次，本文详细阐述了系统的架构设计，包括数据层、逻辑层、控制层和表现层的功能和相互关系。在界面设计方面，通过Qt Designer实现了用户友好的交互界面，用户可以方便地进行图像加载、识别器选择和识别操作。

在系统实现过程中,重点描述了图像识别器、视频识别器和实时识别器的功能实现方法。通过对FER-2013数据集的模型训练,本文探讨了数据加载与预处理、模型设计与编译、模型训练与评估等步骤。实验结果表明,所提出的系统在表情识别的准确性和实时性方面具有显著优势,特别是在处理混合表情和微表情时,Mini-X模型展示了较高的计算效率和实用性。

关键词: 机器视觉; 人脸表情识别; 卷积神经网络; mini-X模型; 微表情; 实时识别

Abstract

In the field of human-computer interaction and affective computing, the research and application of facial expression recognition systems are of great significance. This paper proposes a facial expression recognition system based on machine vision, utilizing the Xception model and its simplified version, the Mini-X model, for expression feature extraction and classification. First, the paper introduces relevant technologies, including Support Vector Machine (SVM) and Convolutional Neural Networks (CNN), and discusses the application of the OpenCV image processing library and the Qt Designer interface design tool in detail.

Subsequently, the paper elaborates on the system architecture design, including the functionalities and interrelationships of the data layer, logic layer, control layer, and presentation layer. In terms of interface design, a user-friendly interactive interface is implemented through Qt Designer, allowing users to easily perform image loading, recognizer selection, and recognition operations.

During the system implementation process, the paper focuses on the functional realization of the image recognizer, video recognizer, and real-time recognizer. By training the model on the FER-2013 dataset, this paper explores steps such as data loading and preprocessing, model design and compilation, and model training and evaluation. Experimental results demonstrate that the proposed system has significant advantages in the accuracy and real-time performance of expression recognition. Particularly in handling mixed and micro-expressions, the Mini-X model shows high computational efficiency and practicality.

Keywords: Machine Vision; Facial Expression Recognition; Convolutional Neural Network; Mini-X Model; Micro-Expressions; Real-time Recognition

目录

摘要.....I	
Abstract.....II	
第 1 章绪论.....1	
1.1 课题背景概述.....1	
1.2 研究内容相关领域概述.....1	
1.3 研究目标和思路.....3	
1.4 本文结构.....4	
第 2 章相关技术理论和研究软件.....5	
2.1 支持向量机(SVM)算法.....5	
2.2 卷积神经网络(CNN)技术.....5	
2.3 面部动作编码系统.....6	
2.4 Python编程语言.....7	
2.5 OpenCV图像处理库.....7	
2.6 MATLAB矩阵编程语言.....7	
2.7 Qt Designer界面设计工具.....8	
2.8 本章小结.....8	
第 3 章表情特征的提取与识别.....9	
3.1 引言.....9	
3.2 Xception模型.....9	
3.3 FER-2013数据集.....11	
3.4 模型训练.....11	
3.4.1 数据加载与预处理.....11	
3.4.2 模型设计与编译.....12	
3.4.3 模型训练与评估.....13	
3.5 表情特征提取算法.....13	
3.6 表情特征分类算法.....15	
3.7 混合表情和微表情处理.....15	
3.8 本章小结.....16	
第 4 章人脸表情识别系统的设计实现.....19	
4.1 系统架构概述.....19	
4.2 系统界面设计.....21	
4.2.1 用户交互设计.....22	
4.3 人脸表情识别程序功能介绍.....23	
4.3.1 图像识别器.....23	
4.3.2 视频识别器.....24	
4.3.3 实时识别器.....25	
4.4 图像识别器功能实现.....26	
4.4.1 图像预处理.....26	
4.4.2 人脸捕捉.....27	
4.4.3 静态图像识别.....27	
4.5 视频识别器功能实现.....28	
4.5.1 视频流处理.....29	
4.5.2 补帧动画.....29	
4.6 实时识别器功能实现.....30	
4.7 本章小结.....31	
结论.....33	
参考文献.....34	
致谢.....37	
附录1 表情系统编码规则.....39	

2. 第1章绪论		总字数：3108
相似文献列表		
去除本人文献复制比：18.1%(561) 去除引用文献复制比：13.9%(433) 文字复制比：18.1%(561) 疑似剽窃观点：(0)		
1	基于注意力机制的人脸表情识别 高禹 - 《大学生论文联合比对库》- 2020-05-02	4.4% (137) 是否引证：否
2	基于深度学习的人脸面部表情识别技术研究 曹寅添 - 《大学生论文联合比对库》- 2023-06-03	4.3% (135) 是否引证：否

3	基于深度学习的人脸面部表情识别技术研究 曹寅添 - 《大学生论文联合比对库》 - 2023-06-03	4.3% (135) 是否引证: 否
4	刘婷婷-104110120-抑郁症患者对情绪图片识别的研究 刘婷婷 - 《大学生论文联合比对库》 - 2015-05-17	3.3% (104) 是否引证: 否
5	201903160028_王晓阳_基于人脸识别的课堂考勤系统设计与实现_梁虎 王晓阳 - 《大学生论文联合比对库》 - 2023-05-14	3.1% (97) 是否引证: 否
6	人脸面部表情动画技术研究及在影视动画中的应用 刘娟; - 《青海大学学报(自然科学版)》 - 2011-02-20	2.5% (78) 是否引证: 否
7	人面部表情识别方法综述 杜宁; - 《黑龙江科技信息》 - 2011-10-05	2.5% (77) 是否引证: 否
8	基于人脸识别的应用锁开发 李博涵 - 《大学生论文联合比对库》 - 2023-05-24	1.3% (41) 是否引证: 否
9	远程体温检测与人脸识别系统设计 苏炳书 - 《大学生论文联合比对库》 - 2023-06-14	1.3% (39) 是否引证: 否
10	基于深度学习的视频图像情绪分析系统设计与实现 李轩龙 - 《大学生论文联合比对库》 - 2023-06-08	1.3% (39) 是否引证: 否
11	人脸表情识别方法研究 钟扬 - 《大学生论文联合比对库》 - 2023-06-13	1.0% (31) 是否引证: 否
12	基于多群体遗传算法的人脸表情识别 朱娅妮;吴国华;杜加友; - 《杭州电子科技大学学报》 - 2013-02-15	0.9% (28) 是否引证: 否

原文内容

第 1 章绪论

1.1 课题背景概述

情感计算作为一个跨多学科的研究领域，随着人工智能技术的迅速发展和应用场景的不断扩展，人们对于机器能够准确理解人类情感和意图的需求日益增加。作为人类情感的重要载体，人脸表情识别项目具有极高的理论研究价值和应用价值。

在虚拟现实、智能家居、社交媒体等领域，人脸表情识别系统可以为用户提供更加智能、个性化的服务体验，从而提升产品的竞争力和用户满意度。在商业应用方面，人脸表情识别系统可以为企业提供情感分析和用户行为分析等关键数据，帮助企业更好地洞察消费者需求和市场动态，从而优化产品设计与营销策略。此外，人脸表情识别系统在安防监控、心理健康辅助诊断等领域也具有重要的应用前景。

达尔文[1]在其研究中指出表情系统在不同性别、不同种族的人群中具有极高的一致性。Ekman和Frisen[2]提出了面部表情编码系统（FACS），用44个运动单元（AU）来描述面部动作，并指出6种基本情感类别：愤怒、厌恶、恐惧、快乐、悲伤、惊讶。

1.2 研究内容相关领域概述

人脸表情识别系统的研究涉及多个学科领域，包括计算机视觉、模式识别、机器学习等。在计算机视觉领域，研究人员致力于开发高效的人脸检测和关键点定位算法，以实现对人脸表情的精确识别和分析。同时，模式识别领域的研究则集中于设计和优化表情特征提取和分类算法，以实现对不同表情类别的准确识别和区分。

一个人脸表情识别系统一般包括3个环节，即人脸检测、特征提取、表情分类。建立一个FER系统，第1步需要对人脸进行检测与定位，这一环节的研究实际上已成为一个独立的方向[3]。通过多学科交叉和技术融合，研究人员不断突破技术瓶颈，推动着人脸表情识别系统的实际应用和发展。

1.2.1 人脸识别算法概述

人脸识别算法主要致力于从图像或视频中准确地提取人脸信息，并进行人脸的检测、识别等操作。常见的人脸识别算法包括基于特征的方法、基于模型的方法、以及基于深度学习的方法。其中，基于特征的方法如局部二值模式（LBP）和主成分分析（PCA）早期取得了一定的成功；基于模型的方法如主动形状模型（ASM）和主动外观模型（AAM）则在形状和纹理的建模上具有优势。近年来，基于深度学习的方法如卷积神经网络（CNN）在人脸识别领域取得了显著的成果，通过深度网络能够自动提取高层次特征，大幅提升了识别精度。

1.2.2 表情识别算法概述

表情识别算法主要研究如何从人脸图像或视频中识别出人物的表情状态，常见的方法包括基于特征的方法、基于深度学习的方法等。基于特征的方法通过手工设计特征如几何特征和纹理特征，分析面部的关键点变化；而基于深度学习的方法如卷积神经网络（CNN）和递归神经网络（RNN）则通过大量数据训练模型，自动学习和提取有效特征。

表情识别算法主要包括两个阶段：特征提取和情感分类。特征提取阶段通过分析面部表情中的关键特征点，例如眉毛、眼睛和嘴巴等，提取出数值化的特征向量。情感分类阶段则使用分类算法，将特征向量与训练好的模型进行比对，最终确定面部表情所代表的情感类别。近年来，研究人员提出了一些先进的深度学习架构，如卷积神经网络（CNN）、长短期记忆网络（LSTM）等，用于表情识别中的特征提取和情感分类，显著提升了系统的识别准确率和鲁棒性。

1.2.3 现有研究的优缺点分析

现有的人脸表情识别研究方法主要分为基于特征的方法、基于模型的方法和基于深度学习的方法。

基于特征的方法的优点在于简单高效、计算复杂度低，且解释性强，便于调试。然而，这些方法对光照、表情幅度变化和遮挡的鲁棒性较差，且严重依赖手工设计的特征，难以应对复杂表情。

基于模型的方法利用面部结构化信息，能提高识别精度并适应不同的面部形状和姿态变化，但其模型建立和优化过程复杂，对计算资源要求较高，且在处理非标准化图像时泛化能力有限。

基于深度学习的方法能够自动学习和提取高层次特征，减少了对手工特征设计的依赖，且在大规模数据集上训练后具有高精度和鲁棒性。然而，深度学习方法对大规模标注数据集依赖较大，数据收集和标注成本高，训练和推理过程需要大量计算资源，同时模型的内部机制较为复杂，不易解释。

综合分析，未来的研究方向应集中在混合方法的探索上，结合传统特征提取和深度学习优势，以提高系统的整体性能和鲁棒性。

1.3 研究目标和思路

本研究旨在探索建立一个基于机器视觉的人脸表情识别系统，以提升人机交互的智能化水平。通过对人脸图像进行特征提取和分类，实现对不同表情的准确识别。

它主要包括两个技术环节：首先是人脸的检测与定位，然后进行表情特征提取和识别，即从输入的图像中检测和定位人脸。然后再提取出对表情，识别有用的信息并进行分类识别[4]。

为实现此目标，本研究综合运用了多种先进技术和方法。包括支持向量机（SVM）算法和卷积神经网络（CNN）技术，结合实际需求和技术特点，对这些算法和模型进行了优化，以提高识别的准确率和处理速度，创造一个高效、准确的人脸表情识别系统。

1.3.1 课题的理论与实践意义

人脸表情识别技术在人机交互领域展现出巨大的应用潜力。通过精准识别人脸表情，能够显著提升人机交互的自然性和智能化水平，为用户提供个性化且人性化的服务体验。此外，这项技术在医学、心理学和安防等领域同样发挥着重要作用。例如，利用人脸表情识别技术，可以进行情绪监测、精神疾病早期诊断、以及安防监控等应用，从而对社会生活和工作产生深远影响。因此，深入研究和广泛应用人脸表情识别技术，不仅推动相关领域的创新发展，而且具备重要的理论和实践意义。

1.3.2 研究目标

本研究的核心目标是设计并实现一个高效且精准的人脸表情识别系统，满足多种实际应用需求。具体目标包括：提高表情识别的准确率和处理速度，增强系统的鲁棒性和稳定性，同时降低系统的成本和复杂度，使其具备更广泛的推广和应用前景。通过实现这些目标，本研究旨在推动人机交互技术的发展，为用户提供更优质的服务体验，并在相关应用领域中发挥积极作用。

1.3.3 研究思路

本研究旨在通过深入研究和分析人脸表情识别算法和技术，设计并优化适用于人脸表情识别的算法和模型。具体研究思路如下：

- (1) 深入了解SVM算法和CNN技术等主流方法在表情识别中的应用效果，分析其优缺点。
- (2) 结合实际应用需求和技术特点，对现有算法和模型进行有针对性的优化与改进。这可能涉及算法结构的调整、特征提取方法的改进、以及模型训练过程的优化等。
- (3) 设计并进行系统的实验验证，通过性能评估不断优化和改进系统，确保识别效果的稳定性和高效性。
- (4) 将优化后的算法和模型集成到实际应用系统中，进行全面测试和调整，确保系统在各种应用场景中的实用性和可靠性。

1.4 本文结构

本文将按照以下章节顺序进行介绍：

第一章：从课题背景、相关领域概述、研究目标和思路等方面对研究主题进行阐述，并概述本文的结构。

第二章：主要介绍支持向量机、卷积神经网络、Python编程语言、OpenCV图像处理库和Qt Designer界面设计工具等相关技术理论和研究软件。

第三章：详细介绍Xception模型、FER-2013数据集、模型训练、表情特征提取算法、表情特征识别算法以及混合表情和微表情处理等内容。

第四章：介绍人脸表情识别系统的架构概述和系统界面设计，详细描述系统的组成结构和用户界面设计。

第五章：详细介绍人脸表情识别程序的功能介绍和实现方法，包括图像识别器、视频识别器和实时识别器等。

指 标	
疑似剽窃文字表述	
1. 常见的人脸识别算法包括基于特征的方法、基于模型的方法、以及基于深度学习的方法。其中，	
2. 它主要包括两个技术环节：首先是人脸的检测与定位，然后进行表情特征提取和识别，即从输入的图像中检测和定位人脸。	
3. 第2章相关技术理论和研究软件	总字数：3029
相似文献列表	

1	人脸面部表情识别的研究进展	5.8% (177)
	何良华, 邹采荣, 包永强, 赵力 - 《电路与系统学报》- 2005-02-28	是否引证：否
2	基于深度学习的野生动物识别算法研究	1.9% (58)
	林凯存 - 《大学生论文联合比对库》- 2023-06-19	是否引证：否
3	基于人脸五官结构特征的表情识别研究	1.8% (56)
	马飞; 刘红娟; 程荣花; - 《计算机与数字工程》- 2011-09-20	是否引证：否
4	在线多目标跟踪算法设计与实现	1.8% (56)
	朱利民 - 《大学生论文联合比对库》- 2020-06-03	是否引证：否
5	2220161832-方馨-第6组	1.8% (56)
	方馨 - 《大学生论文联合比对库》- 2020-06-04	是否引证：否
6	244-3019210102-曹皓然	1.5% (46)
	曹皓然 - 《大学生论文联合比对库》- 2023-05-31	是否引证：否
7	旋转机械故障诊断系统开发	1.3% (38)
	李浩文 - 《大学生论文联合比对库》- 2023-06-14	是否引证：否
8	194020118-计算机-刘宇杰	1.2% (36)
	计算机 - 《大学生论文联合比对库》- 2023-05-18	是否引证：否

原文内容

第 2 章相关技术理论和研究软件

2.1 支持向量机(SVM)算法

支持向量机(SVM)是一种常用的机器学习算法，它被广泛应用于分类和回归问题中。SVM的基本思想是通过找到一个最优的超平面来将不同类别的样本分开。它的核心概念是支持向量，即距离超平面最近的训练样本点。SVM算法在人脸表情识别中具有良好的性能和应用前景。

2.1.1 SVM算法原理

SVM算法基于统计学习理论中的结构风险最小化原则，旨在找到一个能够正确分类样本的最大间隔超平面。

SVM算法的思路可以概括为：对于非线性可分样本，首先通过定义核函数将非线性变换从输入空间转化成高维空间，接着在高维空间中求解最优化的线性分界面。对于线性不可分的情况[6]，即使在线性不可分的情况下，SVM也可以通过引入松弛变量和软间隔来构建分类器，并通过选择适当的核函数来处理非线性分类问题。

2.1.2 SVM算法在人脸表情识别中的应用

在人脸表情识别中，SVM算法通过将人脸图像映射到高维特征空间，构建出有效的分类器，能够对人脸表情进行精确识别。具体来说，人脸图像中的特征可以被抽象为高维空间中的点集，SVM通过学习这些特征点的分布和边界，建立一个能够区分不同表情类别的决策边界。

SVM算法的泛化能力使其能够有效地应对不同表情的变化和图像中的噪声干扰，确保分类器在处理未见数据时仍能保持较高的准确性。此外，通过选择合适的核函数，SVM可以处理复杂的非线性表情识别问题，这使得它在面对多样化的表情特征时，依然能够保持出色的识别性能。

2.2 卷积神经网络(CNN)技术

卷积神经网络（CNN）是一种深度学习模型，在计算机视觉领域中被广泛应用于图像处理和特征提取任务。相比于传统的全连接神经网络，CNN通过局部感知和权值共享的方式极大地减少了参数数量，同时能够有效地提取图像的局部特征并保持平移不变性。其中，卷积层通过使用不同的滤波器在输入数据上进行滑动运算，从而捕获输入数据的局部特征；池化层则通过降采样操作减少参数数量，同时保留重要的特征信息；激活函数引入非线性，并增加模型的表达能力；全连接层在经过多层卷积和池化操作后，将特征映射到具体的标签类别上。这种多层结构的CNN能够实现对复杂数据的高效分类和表征学习。

2.2.1 CNN技术原理

CNN基于局部感知和权值共享的设计理念。卷积操作通过滑动卷积核，捕捉输入图像的局部特征，从而提取空间信息。随着卷积层的增加，CNN逐层学习从低级到高级的特征，例如边缘、纹理、形状等。通过多层卷积和池化操作，CNN逐步学习到更抽象和高级的特征表示，使其在图像分类和识别任务中表现出色，成为计算机视觉领域的核心技术。

2.2.2 CNN技术在人脸表情识别中的应用

本研究将CNN技术引入人脸表情识别系统中，旨在自动学习和提取人脸图像的特征，以实现对不同表情的准确分类。通过学习大量标注好的人脸表情图像，CNN能够自动地提取出关键的面部特征，并建立这些特征与各个表情类别之间的映射关系。CNN具有对图像进行可靠分类的能力，并且对于噪声数据具有较好的鲁棒性。此外，CNN还能够处理人脸图像中的时间和空间信息，以实现微表情的识别。

通过结合SVM算法和CNN技术，此人脸表情识别系统能够更好实现对不同表情的准确识别和分类。

2.3 面部动作编码系统

美国心理学家Paul Ekman和Friesen较早地对脸部肌肉群的运动及其对表情的控制作用做了深入研究，开发了面部动作编码

系统(FacialActionCodingSystem, FACS)来描述面部表情[2]。根据人脸的解剖学特点,将人脸划分成若干既相互独立又相互联系的单元,即运动单元(AU),在分析了这些运动单元的运动特征后,得出了每个单元所控制的主要区域,以及可能与之相关的表情。

很多特征的形成可能被多种表情诱发,如眼眉下移就可能是被生气或厌恶引起,而眼眉上移可能是因为吃惊、害怕和伤心而引起,所以日常生活的表情往往表现为混合情感的结果[5]。

2.4 Python编程语言

Python占据人脸表情识别系统中最重要的部分。其简洁易读的语法使开发人员能够快速理解和编写代码,从而加速算法的开发和优化过程。Python的第三方库和工具,如NumPy、Pillow等,为图像处理和数据分析提供了强大支持,为特征提取和结果可视化等关键步骤、图像处理、特征提取、模型训练等关键模块提供了便利。此外,Python在深度学习领域的应用也是不可或缺的,Keras等框架提供了高效的神经网络训练和推理能力,使得人脸表情识别系统能够更准确地识别和分类不同的表情。

2.5 OpenCV图像处理库

OpenCV (Open Source Computer Vision Library) 是一款在计算机视觉领域广泛使用的开源图像处理库,为表情识别系统提供了强大的图像处理和计算机视觉功能。通过OpenCV,可以实现图像加载、预处理、特征提取以及人脸检测等功能。

在人脸检测是系统中,OpenCV提供的人脸检测器能够准确地定位图像中的人脸区域,并提取相关特征用于后续的表情分类。OpenCV的高效性和可靠性为本系统赋予了较高的准确性和鲁棒性,为实现可靠的人脸表情识别提供了重要保障。

2.6 MATLAB矩阵编程语言

MATLAB (MATrix LABoratory) 是一种专为数值计算、符号计算、数据可视化、算法开发以及用户界面构建的应用程序设计语言和环境。广泛应用于工程计算、科学计算、数据分析、机器学习、人工智能等领域。MATLAB的核心特点是它的矩阵和数组为基础的运算,这使得处理大量数据或进行向量和矩阵运算非常高效。

MATLAB提供了简洁直观的语法,用户可以通过调用内置函数和命令轻松完成各种复杂操作,如矩阵运算、傅立叶变换和数据拟合等。其强大的数据可视化能力,使得用户能够快速生成高质量的图表和图像,从而更直观地分析和展示数据。

2.7 Qt Designer界面设计工具

Qt Designer是一款用于创建图形用户界面(GUI)的可视化设计工具,为表情识别系统提供了便捷的界面设计方案。通过Qt Designer,用户可以快速创建主窗口、菜单栏、工具栏等各种GUI元素,从而实现用户友好的界面设计。

使用Qt Designer设计系统的用户界面,使得用户能够方便地进行图像输入、表情识别以及结果展示。Qt Designer生成的UI文件与Python代码的无缝集成,使得界面设计与后端逻辑的交互变得更加简单,同时大大提高了开发效率。通过Qt Designer设计的界面不仅能够提升系统的用户体验,还能够使界面设计与功能实现更加紧密地结合,从而提高了整个系统的质量和性能。Qt Designer的使用极大地简化了界面开发的流程,为表情识别系统的设计和实现提供了强有力的支持。

2.8 本章小结

本章对SVM算法、CNN技术、Python语言、OpenCV库、MATLAB和Qt Designer进行了全面介绍。通过对这些关键技术和工具的介绍,为后续的人脸表情识别系统设计和实现奠定了基础。本章深入探讨了每个技术和工具的原理、应用场景以及在人脸表情识别中的作用。这些技术将在后续章节中进一步应用和优化,以实现一个高效、准确的人脸表情识别系统。

指 标

疑似剽窃文字表述

1. 神经网络, CNN通过局部感知和权值共享的方式极大地减少了参数数量,同时能够有效地提取图像的局部特征并保持平移不变性。
2. 根据人脸的解剖学特点,将人脸划分成若干既相互独立又相互联系的单元,即运动单元(AU),在分析了这些运动单元的运动特征后,得出了每个单元所控制的主要区域

4. 第3章表情特征的提取与识别	总字数: 6194
------------------	-----------

相似文献列表

去除本人文献复制比: 7.2%(444) 去除引用文献复制比: 7.2%(444) 文字复制比: 7.2%(444) 疑似剽窃观点: (0)

1	基于深度学习的人脸表情识别系统 李响 - 《大学生论文联合比对库》- 2023-06-14	2.4% (149) 是否引证: 否
2	基于Gabor特征的面部微表情识别 姚朝祥 - 《大学生论文联合比对库》- 2023-05-25	1.4% (89) 是否引证: 否
3	基于Gabor特征的面部表情识别 姚朝祥 - 《大学生论文联合比对库》- 2023-06-09	1.4% (89) 是否引证: 否
4	网络安全-19270833-叶进挺 网络安全 - 《大学生论文联合比对库》- 2023-05-17	1.0% (62) 是否引证: 否
5	11-2220194133-林中华	1.0% (61)

林中华 - 《大学生论文联合比对库》 - 2023-05-15		是否引证: 否
6	基于深度学习的白血病细胞识别研究	0.9% (55)
杨海若 - 《大学生论文联合比对库》 - 2023-05-30		是否引证: 否
7	基于openmv的安防门禁系统设计	0.8% (49)
王星云 - 《大学生论文联合比对库》 - 2023-06-15		是否引证: 否
8	基于卷积神经网络的交通标志识别研究	0.6% (40)
罗晨晓 - 《大学生论文联合比对库》 - 2023-06-13		是否引证: 否
9	基于深度学习的肺炎检测系统的设计与实现	0.6% (35)
孙晟昊 - 《大学生论文联合比对库》 - 2023-06-12		是否引证: 否
10	基于深度学习的图像分类实验	0.6% (35)
王宪超 - 《大学生论文联合比对库》 - 2023-06-13		是否引证: 否
11	采用迁移学习的小样本锂电池荷电状态估算方法	0.4% (27)
李龙;燕旭滕;张钰声;冯雅琳;胡红利;段羽洁;崔晨辉; - 《西安交通大学学报》 - 2023-11-10		是否引证: 否

原文内容

第 3 章表情特征的提取与识别
3.1 引言

人脸表情识别作为计算机视觉领域的一个核心研究方向，在情感识别、人机交互、智能监控等领域具有广泛的应用前景。在实现准确识别的过程中，表情特征的提取与识别是至关重要的步骤之一。本章的主要目标在于介绍基于深度学习的方法，以实现对表情特征的提取与识别。

首先，本章将深入探讨Xception模型及其简化版本Mini-X模型的原理和应用，以及本研究所采用的FER-2013数据集。接着，我们将详细讨论模型训练的流程，包括数据的加载与预处理、模型的设计与编译，以及模型的训练与评估。

进一步地，本章将介绍表情特征提取算法、表情特征分类算法，以及处理混合表情和微表情的方法。通过本章的研究，我们将建立一个完整的人脸表情识别系统，为后续系统设计提供坚实的基础。

3.2 Xception模型

Xception模型是一种卷积神经网络模型，其在图像分类任务中展现了卓越的性能。在人脸表情识别任务中，Xception模型通过在大量的表情图像数据上进行训练，能够有效地学习到不同表情之间的特征表示，从而实现对表情的准确识别。其在图像分类任务中表现出的优秀性能为人脸表情识别系统的设计提供了强大的技术支持。本节将详细介绍Xception模型的结构和原理，并探讨其在人脸表情识别任务中的应用。

3.2.1 模型结构与原理

Xception模型是在InceptionV3模型基础上进行改进的，它包括了36个卷积层和14个模块。Xception模型采用了深度可分离卷积来替代Inception V3中的卷积模块。深度可分离卷积将标准的卷积操作分解为“深度卷积”和“逐点卷积”两个独立步骤，并引入了残差学习结构。这种设计使得模型能够更好地捕获图像中的局部和全局特征，同时减少了参数数量和计算复杂度。通过构建深度可分离卷积块，Xception模型能够有效地提取图像特征，并实现对图像的准确分类。相比于Inception V3，Xception模型的结构相对简单，但在各种图像分类任务中表现出了很高的性能。具体的网络结构如图31所示。

图31 Xception 模型的主要网络模块示意图

3.2.2 Mini-X模型原理

Mini-X模型是对Xception模型的简化版本，旨在减少模型的复杂度和计算成本，同时保持较高的性能水平。

Mini-X模型采用了Xception模型的关键部分，并进行了适当的简化和优化。首先，通过裁剪网络层数，Mini-X模型减少了模型的深度，从而减少了参数数量和计算复杂度。其次，Mini-X模型减少了每个卷积层的通道数，降低了模型的宽度，进一步减少了计算量。最后，Mini-X模型通过精心设计和参数调整，保持了在表情识别任务中的性能，并在计算效率上有所提升。

3.2.3 Mini-X模型在人脸表情识别中的应用

Mini-X模型在人脸表情识别任务中具有广泛的应用前景。在人脸表情识别任务中，Mini-X模型能够提取人脸图像中的特征，实现对不同表情的准确识别和分类。

其简洁而有效的结构使得Mini-X模型能够在资源受限的环境下实现快速且准确的表情识别，为人脸表情识别系统的实现提供了有力支持。

3.3 FER-2013数据集

经过对比研究，本文选择了FER-2013数据库作为研究数据源，并采用卷积神经网络的有监督学习算法进行模型训练。以此对课题进行研究。

FER-2013数据集是一个常用的公开数据集，用于人脸表情识别任务的研究。FER-2013数据库包含了来自Kaggle竞赛网站的35887张人脸图像，涵盖了七种不同的情感类别，包括愤怒、厌恶、恐惧、快乐、悲伤、惊讶和平静。每张图像的大小为48x48像素，灰度图像，以CSV格式存储。这个数据集的特点是图像数量较多，情感类别丰富，并且图像质量较高，适合用于训练和评估表情识别模型。

图32是几个示例图片。

图32 FER-2013示例图片

3.4 模型训练

人脸表情识别的模型训练是构建表情识别系统的关键步骤之一。在模型训练阶段，使用前文提到的FER-2013数据集来训练此人脸表情识别模型。主要代码如下所示。

3.4.1 数据加载与预处理

将读取图像数据并将其转换为模型可接受的格式。这包括了灰度化、大小调整和归一化等操作，以确保图像数据的一致性和可训练性。其中，灰度化处理有助于降低数据维度和计算复杂度，而大小调整和归一化则有助于提高模型的训练效率和性能。主要代码如下表31所示。

表31 数据加载与预处理代码

```
cnn.py
regularization = l2(l2_regularization)
img_input = Input(input_shape)
x = Conv2D(8, (3, 3), strides=(1, 1), kernel_regularizer=regularization, use_bias=False)(img_input)
x = BatchNormalization()(x)
x = Activation('relu')(x)
x = Conv2D(8, (3, 3), strides=(1, 1), kernel_regularizer=regularization, use_bias=False)(x)
x = BatchNormalization()(x)
x = Activation('relu')(x)
```

3.4.2 模型设计与编译

在模型设计与编译阶段，选择合适的卷积神经网络架构，并设置模型的结构和参数。针对人脸表情识别任务，本研究针对人脸表情识别任务选择了基于深度可分离卷积的Mini-X模型。

Mini-X模型包括了多个卷积层、批归一化层和激活函数层，其中分离卷积层用于减少计算量，残差连接用于加速训练过程。模型的输出层采用了全局平均池化层和softmax激活函数，用于输出分类结果。卷积层主要代码如下表32所示。

表32 卷积层代码

```
cnn.py
residual = Conv2D(16, (1, 1), strides=(2, 2),
padding='same', use_bias=False)(x)
residual = BatchNormalization()(residual)
x = SeparableConv2D(16, (3, 3), padding='same',
kernel_regularizer=regularization,
use_bias=False)(x)
x = BatchNormalization()(x)
(续表32)
x = Activation('relu')(x)
x = SeparableConv2D(16, (3, 3), padding='same',
kernel_regularizer=regularization,
use_bias=False)(x)
x = BatchNormalization()(x)
x = MaxPooling2D((3, 3), strides=(2, 2), padding='same')(x)
x = layers.add([x, residual])
```

在模型设计过程中，首先初始化正则化器，使用给定的l2_regularization值初始化L2正则化器，用于权重衰减，以防止模型过拟合。然后，接收输入图像，并通过连续的卷积层（带ReLU激活和批归一化）开始构建模型的基础。接下来，利用残差连接进行通道调整和下采样，然后进行分离卷积和批归一化，用于特征提取，并通过最大池化层进行下采样，减半图像的空间维度。最后，将原始输入经过处理后的特征与当前层的特征相加，利用残差学习加速训练过程和提高性能。

最后一个卷积层不使用正则化，用于生成类别预测的特征图。

全局平均池化将空间维度压缩为单个值，保留通道信息，常用于减少过拟合和简化模型。Softmax激活转换特征为概率分布，表示每个类别的预测概率。

3.4.3 模型训练与评估

使用训练集对模型进行训练，并通过测试集对模型进行评估。定义训练的迭代次数和批处理大小，并使用随机梯度下降优化算法来最小化损失。

在训练完成后，通过测试集对模型进行评估，计算模型在测试集上的准确率、精确率和召回率等指标。这些指标将有助于评估模型的性能和泛化能力，以及其在实际应用中的有效性。

通过以上步骤，不断调整模型的结构和参数，并进行多次训练和评估，以达到满意的性能水平为止。

3.5 表情特征提取算法

表情特征提取算法是人脸表情识别系统中的关键步骤之一，其目标是从人脸图像中提取具有代表性的特征信息，以便后续的表情分类任务。本研究选择CNN的Mini-X模型来提取表情特征。

在具体的实现中，对每个人脸图像进行如下处理：

- (1) 计算人脸区域的坐标和尺寸。
- (2) 提取图像中对应面部区域的部分。
- (3) 将面部区域调整到预定义的大小。
- (4) 转换调整后的图像数据类型为浮点数，以便于模型处理。
- (5) 添加额外的维度，使图像数据满足模型输入的形状要求。
- (6) 使用前文训练的模型对处理后的面部图像进行表情分类预测，得到一个表示不同表情概率的向量。

主要代码如表33所示

表33 表情特征提取代码

```
visualmodule.py
def getEmotion(cls, image: CVImage, fases: Sequence[Rect]) -> list[tuple]:
    img_np = np.expand_dims(image, 2)
    result = []
    for x1, y1, width, height in fases:
        x2, y2 = x1 + width, y1 + height
        face = img_np[y1:y2, x1:x2]
        try:
            face = cv2.resize(face, CLASSIFIER_EMOTION_SIZE)
        except:
            continue
        face = cls.uint2float(face)
        face = np.expand_dims(face, 0)
        face = np.expand_dims(face, -1)
        emotion = CLASSIFIER_EMOTION.predict(face)[0]
        result.append((x1, y1, x2, y2, emotion))
    return result
```

3.6 表情特征分类算法

表情特征分类算法是人脸表情识别系统中的核心部分，旨在将从人脸图像中提取的特征信息与预定义的表情类别进行匹配，从而实现对不同表情的准确分类。表情特征提取的结果是一个包含了该表情对于7个表情标签的归属程度的向量，包含了该表情对于7个表情标签的归属程度，如下所示。

[1.01e-01, 1.65e-03, 2.37e-01, 2.29e-05, 6.50e-01, 8.67e-05, 8.95e-03]

7个表情的归属度均为百分数，其和为1。

一般来说归属度最高的表情就是最终认定的表情结果。最后将信息转化为标签并输出。

在具体的实现中，可以使用MATLAB对表情特征提取出的表情向量进行进一步处理。例如：

(1) 快乐和悲伤在一定程度上是矛盾的，而恐惧和惊讶在面部表情相似度很高。因此针对不同表情之间的相似度和冲突性，可以引入注意力模块进行标签特征化。

(2) 快乐表情的特征非常明显，而厌恶表情的特征与多个表情有一定的共同之处。因此对于表情的显著性不同，可以使用样条插值进行归一化处理，以更好地反映各个表情在图像中的重要性。

3.7 混合表情和微表情处理

在实际应用中，人脸图像可能会出现混合表情或微表情的情况，这给表情识别系统带来了挑战。本研究采用了Mini-X模型和非线性映射的结合策略来处理这一问题。

首先，Mini-X模型能够提取图像中丰富的表情特征，包括微表情的细微变化。然后，通过非线性映射，可以将复杂的混合表情分解为单个表情成分，并将其准确地归类到相应的表情类别中。这样的处理方式可以有效地提高表情识别系统对复杂表情的识别和处理能力。

本研究以最显著表情的显著性的80%为阈值，超过此阈值的表情可以组成“显著表情集”（不超过3个），每个显著表情集均对应一个混合表情或微表情。包含挑衅、冷酷、无聊、冷漠、焦虑、不安、憎恨、紧张、激动、满足、震惊、悲愤、绝望、嘲笑、疑虑、沮丧、惊喜、失望、复杂、矛盾、反常等（详见附录1）。这种处理方式可以提高表情识别系统对复杂表情的识别和处理能力，从而更准确地反映人脸图像中的情感状态。

主要代码如表34所示

表34 表情特征分类与微表情处理代码

```
visualmodule.py
def microexpression(cls, origin: tuple[int]) -> str:
    result = []
    confidence = 0
    emotions = tuple(EMOTION_MAP[i](origin[i]) for i in range(7))
    limit = floor(0.8 * max(emotions))
    emotions = sorted(enumerate(emotions), key=lambda x: x[1], reverse=True)[:3]
    for i, value in emotions:
        if value < limit:
            break
        result.append(i)
        confidence += origin[i]
    name = tuple(sorted(result))
    detail = tuple(EMOTION_LABELS[i] for i in result)
    return EMOTION_MICRO_LABELS[name].format(*detail, round(confidence * 100))
```

3.8 本章小结

本章介绍了人脸表情识别系统中表情特征的提取及其识别过程。首先，本章介绍了Xception模型及其简化版本Mini-X模型的原理和应用，以及采用的FER-2013数据集。接着，本章详细讨论了模型训练的过程，包括数据加载与预处理、模型设计与编

译、以及模型训练与评估。并且本章介绍了表情特征提取算法、表情特征分类算法以及处理混合表情和微表情的方法。

表情特征提取算法处理人脸图像中的特定区域，通过图像调整和格式转换等操作提取关键表情信息。表情特征分类算法对提取的特征进行归类和匹配，实现对不同表情的准确分类。为处理混合表情和微表情，本研究结合了Mini-X模型和非线性映射的方法，有效识别和处理复杂表情情况。本章建立了一个完整的人脸表情识别系统，并为后续的系统设计和实现提供了重要参考和指导。

指 标
疑似剽窃文字表述
1. 采用了深度可分离卷积来替代Inception V3中的卷积模块。深度可分离卷积将标准的卷积操作分解为“深度卷积”和“逐点卷积”两个独立步骤，并引入了残差学习结构。
2. 相比于Inception V3，Xception模型的结构相对简单，但在各种图像分类任务中表现出了很高的性能。
3. 训练完成后，通过测试集对模型进行评估，计算模型在测试集上的准确率、精确率和召回率等指标

5. 第4章人脸表情识别系统的设计实现	总字数：9142
相似文献列表	
去除本人文献复制比：0.7%(64) 去除引用文献复制比：0.7%(64) 文字复制比：0.7%(64) 疑似剽窃观点：(0)	
1 基于卷积神经网络的人脸表情识别	0.4% (33)
成萱 - 《大学生论文联合比对库》- 2023-05-21	是否引证：否
2 基于视频图像的社区排队人数统计	0.3% (31)
宁杰 - 《大学生论文联合比对库》- 2023-06-12	是否引证：否
原文内容	

第 4 章人脸表情识别系统的设计实现

4.1 系统架构概述

本研究设计了一个基于机器视觉的人脸表情识别系统，其核心目标是实现对面脸图像中表情的准确识别和分类。系统的架构分为四个主要层级：数据层、逻辑层、控制层和表现层。如图41所示。

图41 系统架构设计

4.1.1 数据层

数据层（Data Layer）负责存储底层数据常量和模型文件。这些数据常量可能包括系统所需的基础数据，而模型文件则包含了用于人脸和表情识别的预训练模型等。此外，该层还提供了关于识别器框架的抽象定义，提供静态类型检查，确保识别器的一致性和正确性。二者为并列关系，互不从属。

4.1.2 逻辑层

逻辑层（Logic Layer）包含了系统的核心逻辑功能，负责识别和处理工作。在这一层中，定义了通用逻辑执行函数，用于图像处理、表情检测等任务。此外，实现了不同类型的识别器，包括图像识别器、视频识别器和实时识别器，每个识别器负责执行特定类型的识别任务。数据层为逻辑层提供了基础框架和填充支持。

4.1.3 控制层

控制层（Control Layer）是整个系统的枢纽，负责接收来自表现层的数据和指令，将工作指令传递给正确的逻辑执行机构，并将逻辑层反馈的数据传递回表现层。控制层中的控制器提供了接口，用于协调不同识别器的工作状态，以及处理来自用户界面的事件和操作。

4.1.4 表现层

表现层（Presentation Layer）是系统与用户交互的界面。它启动用户交互界面，并将用户的操作传递给控制层进行处理。通过与用户的交互，表现层将用户的需求传达给控制层，同时将逻辑层的结果反馈给用户。通过依赖注入将UI控制指令传递到控制层。

图42 程序文件结构

综上所述，该系统的分层架构使得系统具有清晰的功能划分和模块化特性，有助于系统的开发、维护和扩展。

4.2 系统界面设计

基于PyQt框架设计的系统界面，用于人脸表情识别系统的用户交互。如图43所示。界面主要分为以下几个部分：

人脸图像框：位于界面中央，用于实时显示人脸图像。该区域采用QLabel控件实现，并带有黑色边框。

数据框：位于界面底部，用于显示系统的一些数据信息，如识别结果等。同样采用QLabel控件实现。

图43 人脸表情识别系统界面示意图

识别类型选择：在界面左侧，提供了三种识别类型的选择按钮：图像识别、视频识别和实时识别。用户可以根据需求选择相应的识别类型。

开始识别按钮和模型载入按钮：位于界面左侧，提供了开始识别按钮和模型载入按钮。用户可以点击开始识别按钮启动识

别过程，而模型载入按钮用于加载模型。

通过这个用户界面，用户可以直观地进行人脸表情识别的操作，选择不同的识别类型，并随时查看识别结果。

4.2.1 用户交互设计

系统的用户交互设计旨在提供简单直观的界面，使用户能够方便地使用人脸表情识别功能。

如图44所示，系统采用了信号和槽机制，实现了界面组件与控制器之间的交互。当用户单击开始识别按钮或选择不同的识别器类型时，相应的信号将触发控制器中的方法，从而启动图像加载、识别或摄像头捕获等功能。同时，控制器会根据识别器的工作状态更新界面上的显示信息，使用户能够清晰地了解识别过程和结果。

图44 用户交互设计

主要组件包括图像框、数据框、模型载入按钮、开始识别按钮以及识别器选择按钮组。

(1) 用户可以通过单选按钮选择不同的识别器类型：

- 图像识别器：用于从静态图像中识别人脸表情。
- 视频识别器：用于从视频中实时识别人脸表情。
- 实时识别器：用于从摄像头实时捕获图像并识别人脸表情。

(2) 图像框用于显示加载的图像或识别结果。用户可以单击该区域选择图像文件或启动摄像头。

(3) 用户点击模型载入按钮后，系统会载入模型数据和信息，同时通过依赖注入将UI控制指令传递到控制层。

(4) 用户点击开始识别按钮后，系统将开始对图像或视频中的人脸进行表情识别。

(5) 数据框用于显示识别信息，如加载状态、识别结果等。

4.3 人脸表情识别程序功能介绍

本章将详细介绍所设计的人脸表情识别系统的功能及其实现方法。该系统包括图像识别器、视频识别器和实时识别器三大功能模块，分别用于静态图像、视频流以及实时摄像头图像的表情识别。

4.3.1 图像识别器

图像识别器是人脸表情识别系统的核心功能之一，能够对输入的静态图像进行表情识别。用户可通过图像上传或输入指定路径，系统将对图像进行处理并输出识别结果。如图45、图46所示。

图45选择图片

图46图像识别结果

4.3.2 视频识别器

视频识别器是人脸表情识别系统的另一重要功能，能够对输入的视频流进行实时表情识别。通过对视频帧的连续处理，系统能够捕获并分析视频中人脸的表情变化，实现对视频内容的表情识别。该功能适用于视频监控、智能交通等场景，为实时场景下的表情分析提供了有力支持。如图47所示。

图47视频识别结果

4.3.3 实时识别器

实时识别器是人脸表情识别系统的实时交互功能，能够对摄像头捕获的实时图像进行表情识别。通过与计算机连接的摄像头，系统能够实时捕获用户的面部表情，并进行实时分析和识别。用户可通过摄像头实时查看自己的表情识别结果，实现即时反馈和交互体验。如图48所示。

图48实时识别结果

4.4 图像识别器功能实现

图像识别器是人脸表情识别系统的一个核心组件，负责对输入的静态图像进行预处理、检测人脸并识别表情状态。该模块的实现包括三个重难点：图像预处理、人脸捕捉和静态图像识别。

4.4.1 图像预处理

图像预处理是图像识别的第一步，其目的是标准化输入图像，以提高人脸检测和表情识别的准确性和效率。在该系统中，预处理步骤包括图像读取、格式转换、图像缩放等。在开始识别前，需要先使用Keras应用程序接口导入前文训练的人脸识别模型和表情识别模型。

使用PyQt GUI应用程序工具包获取文件路径，再使用NumPy矩阵数值计算扩展将PNG或JPG格式的图像文件转换成矩阵数据，最后将矩阵导入CV2计算机视觉和机器学习软件库进行储存。同时将图像展示在GUI的图像框内。

4.4.2 人脸捕捉

人脸捕捉是图像识别的第二步，目的是从预处理后的图像中检测并定位人脸区域。在此模块中，本研究使用预训练的级联分类器来执行人脸检测，识别图像中的人脸区域。

将图像转换为灰度图像，并将灰度图像输入到模型中计算，得到包含人脸位置信息的数据包。

4.4.3 静态图像识别

静态图像识别是图像识别的最后一步，其目的是对捕捉到的人脸区域进行表情识别。在此阶段，本研究使用预训练的模型来分类人脸表情，如愤怒、快乐、悲伤等。模型通过对图像进行特征提取和表情分类，从而识别出图像中人脸的表情状态。

通过数据包的信息，对每个人脸图像进行如下处理：

- (1) 计算人脸区域的坐标和尺寸。
- (2) 提取图像中对应面部区域的部分。
- (3) 将面部区域调整到预定义的大小。
- (4) 转换调整后的图像数据类型为浮点数，以便于模型处理。
- (5) 添加额外的维度，使图像数据满足模型输入的形状要求。
- (6) 使用前文训练的模型对处理后的面部图像进行表情分类预测，得到一个表示不同表情概率的向量。
- (7) 使用非线性映射对表情向量进行进一步处理，分析图像对应的表情（或混合表情/微表情）。
- (8) 将表情信息整理成文本内容，并添加到图像上。

由于OpenCV不能添加ASCII字符之外的文本信息，因此需要先将彩色图像转为NumPy矩阵，再转换为Pillow图像。使用

Pillow图像处理库添加中文文本后，再依次转为NumPy矩阵和OpenCV图像。

主要代码如表41所示：

表41 图像识别器部分代码

```
recognizer.py
def detectImg(cls) -> tuple[Result, str]:
    (续表41)
    cv2.waitKey(1)
    cls.vidio_origin.set(cv2.CAP_PROP_POS_FRAMES, cls.vidio_index)
    ret, frame = cls.vidio_origin.read()
    cls.vidio_index += 3
    if ret is False:
        return Result.FINISH, ''
    if cls.emotion_freq != 0:
        cls.emotion_freq -= 1
    else:
        cls.emotion_freq = 3
    img_gray = cv2.cvtColor(frame, cv2.COLOR_RGB2GRAY)
    faces = cls.getFace(img_gray)
    if len(faces) == 0:
        cv2.imwrite(PATH_RESULT, frame)
        return Result.FACE_NOT_FOUND_CONTINUE, ''
    cls.emotions = cls.getEmotion(img_gray, faces)
    result, text = cls.markEmotion(frame, cls.emotions)
    cv2.imwrite(PATH_RESULT, result)
    return Result.CONTINUE, text
```

最后，向控制层发出处理完成信号，控制层收到信号后将处理完成的图像展示在GUI的图像框内。

通过上述步骤，图像识别器能够对输入的静态图像进行预处理、检测人脸并识别表情，实现对静态图像中人脸表情的准确识别。这些功能模块的实现为整个系统提供了基础支撑，使得系统能够高效、准确地进行人脸表情识别。

4.5 视频识别器功能实现

视频识别器是人脸表情识别系统的重要组成部分之一，其功能是对输入的视频流进行处理，实时检测人脸并识别表情状态。下面将详细介绍视频识别器的功能实现。其中人脸捕捉和图像识别和前文相似，故不再赘述。

4.5.1 视频流处理

视频流处理是视频识别器的核心功能，主要包括从视频文件中读取视频流和对视频流进行处理两个步骤。

使用PyQt GUI应用程序工具包获取文件路径，再使用VideoCapture打开视频流。如果成功打开视频流，则将视频的第一帧图像展示在屏幕上。通过VideoCapture逐帧读取视频流，并将每一帧图像进行人脸检测和表情识别。同时，为了提高视频流的处理效率，在识别到人脸表情后会对连续的帧进行冷却，避免频繁的表情识别。

4.5.2 补帧动画

补帧动画是视频识别器的附加功能，旨在增强视频流的视觉效果，并在识别到人脸表情后添加一些动态效果。

为了提高视频流的处理效率，在识别到人脸表情后会对连续的帧进行冷却，避免频繁的表情识别。适当调整帧之间的间隔，实现连续帧之间的插帧处理，从而产生流畅的动画效果。

主要代码如表42所示：

表42 视频识别器部分代码

```
recognizer.py
def detectImg(cls) -> tuple[Result, str]:
    cv2.waitKey(1)
    cls.vidio_origin.set(cv2.CAP_PROP_POS_FRAMES, cls.vidio_index)
    ret, frame = cls.vidio_origin.read()
    cls.vidio_index += 3
    if ret is False:
        return Result.FINISH, ''
    if cls.emotion_freq != 0:
        cls.emotion_freq -= 1
    else:
        (续表42)
        cls.emotion_freq = 3
    img_gray = cv2.cvtColor(frame, cv2.COLOR_RGB2GRAY)
    faces = cls.getFace(img_gray)
    if len(faces) == 0:
        cv2.imwrite(PATH_RESULT, frame)
        return Result.FACE_NOT_FOUND_CONTINUE, ''
    cls.emotions = cls.getEmotion(img_gray, faces)
```

```
result, text = cls.markEmotion(frame, cls.emotions)
cv2.imwrite(PATH_RESULT, result)
return Result.CONTINUE, text
```

通过视频流处理和补帧动画两个关键步骤的实现，视频识别器可以对输入的视频流进行实时处理，并实现人脸表情的识别和展示，为用户提供生动、有趣的观看体验。

4.6 实时识别器功能实现

实时识别器是人脸表情识别系统的一个关键组件，其功能是从摄像头获取实时视频流，并对视频中的人脸进行实时识别和表情分析。其中实时视频流的获取方式等内容和前文相似，故不再赘述。

主要代码如表43所示：

表43 实时识别器部分代码

```
recognizer.py
def detectImg(cls) -> tuple[Result, str]:
cv2.waitKey(1)
cls.vidio_origin.set(cv2.CAP_PROP_POS_FRAMES, cls.vidio_index)
ret, frame = cls.vidio_origin.read()
cls.vidio_index += 3
if ret is False:
return Result.FINISH, ''
(续表43)
if cls.emotion_freq != 0:
cls.emotion_freq -= 1
else:
cls.emotion_freq = 3
img_gray = cv2.cvtColor(frame, cv2.COLOR_RGB2GRAY)
faces = cls.getFace(img_gray)
if len(faces) == 0:
cv2.imwrite(PATH_RESULT, frame)
return Result.FACE_NOT_FOUND_CONTINUE, ''
cls.emotions = cls.getEmotion(img_gray, faces)
result, text = cls.markEmotion(frame, cls.emotions)
cv2.imwrite(PATH_RESULT, result)
return Result.CONTINUE, text
```

4.7 本章小结

本章介绍了人脸表情识别系统的设计，着重介绍了人脸表情识别系统的实现细节。首先概述了系统的整体架构，包括数据层、逻辑层、控制层和表现层，详细阐述了每个层级的功能和作用。随后讨论了系统界面设计，通过Qt Designer界面设计工具，系统实现了用户友好的交互界面，其中包括开始识别按钮、模型载入按钮以及识别器类型选择按钮等功能组件。通过信号与槽的连接，将用户操作与系统逻辑关联起来，使得用户能够轻松进行图像加载、识别器选择和识别操作。

细节方面，从图像识别器、视频识别器到实时识别器，系统的功能逐步扩展，满足了用户对于不同场景下表情识别的需求。通过图像识别器，用户可以对静态图像进行表情识别，适用于对已有图片的分析和处理；视频识别器则进一步提升了系统的实用性，能够实时对视频流中的人脸表情进行识别，满足了对动态场景下表情的监测需求；而实时识别器则将功能扩展到了摄像头实时视频流的处理，为用户提供了实时、便捷的表情识别服务。这些功能的实现，不仅展现了系统在技术上的成熟和稳定，也为用户提供了更加丰富和灵活的使用体验。通过本章的介绍，读者可以全面了解系统的功能特点和实现方式，为进一步的研究和应用提供了重要参考。

结论

本研究针对人脸表情识别这一重要课题，提出并实现了一套基于深度学习的系统。首先，本文回顾了相关领域的研究进展，详细介绍了SVM算法和CNN技术在人脸表情识别中的应用。通过对现有方法的分析，我们选择了Xception模型及其简化版本Mini-X模型作为核心算法，利用FER-2013数据集进行训练与验证。

在系统设计方面，我们采用了模块化架构，包括数据层、逻辑层、控制层和表现层。每个层级均有明确的功能分工，确保系统的稳定性和可扩展性。表现层通过Qt Designer进行设计，实现了用户友好的交互界面，使用户能够方便地进行图像加载、模型选择和识别操作。

实验结果表明，Mini-X模型在保持较高识别准确率的同时，大大降低了计算复杂度，提高了系统的实时性能。通过对表情特征的精确提取和分类算法的优化，我们的系统能够有效识别多种表情，包括混合表情和微表情，在情感计算、人机交互、智能监控等领域展现出广阔的应用前景。

综上所述，本研究不仅在理论上对人脸表情识别技术进行了深入探讨，而且在实践中实现了一个高效、实用的系统，为未来相关研究和应用提供了有力的支持。未来的工作将进一步优化算法，扩展数据集，并探索更多的应用场景，以期实现更加智能和全面的人脸表情识别系统。

参考文献

- [1] Darwin C. The Expression of the Emotions in Man and Animals[M]. London: J Murray, 1872.
- [2] P. Ekman et al., "Facial Action Coding System", Consulting Psychologists Press, 1978.
- [3] 刘晓旻, 谭华春, 章毓晋. 人脸表情识别研究的新进展[J]. 中国图象图形学报, 2006, (10): 1359-1368.
- [4] 王志良, 陈锋军, 薛为民. 人脸表情识别方法综述[J]. 计算机应用与软件, 2003, (12): 63-66.

- [5] 刘小沔. 基于深度学习的人脸图像识别技术的研究[D]. 中国科学院大学(中国科学院长春光学精密机械与物理研究所), 2019.
- [6] 施徐敢. 基于深度学习的人脸表情识别[D]. 浙江理工大学, 2015.
- [7] 苏悦, 杨春金, 王建霞. 基于表情识别技术的学生课堂状态检测 [J]. 河北工业科技, 2022, 39 (05): 396-403.
- [8] 张波, 兰艳亭, 李大威, 等. 基于卷积网络通道注意力的人脸表情识别 [J]. 无线电工程, 2022, 52 (01): 148-153.
- [9] 徐琳琳, 张树美, 赵俊莉. 基于图像的面部表情识别方法综述 [J]. 计算机应用, 2017, 37 (12): 3509-3516+3546.
- [10] 乔曦. 基于水下机器视觉的海参实时识别研究[D]. 中国农业大学, 2017.
- [11] 贾晔焯, 杨明强, 张鹏, 等. 微表情自动识别综述 [J]. 计算机辅助设计与图形学学报, 2014, 26 (09): 1385-1395.
- [12] 刘帅师, 田彦涛, 万川. 基于Gabor多方向特征融合与分块直方图的人脸表情识别方法 [J]. 自动化学报, 2011, 37 (12): 1455-1463.
- [13] 吴奇, 申寻兵, 傅小兰. 微表情研究及其应用 [J]. 心理科学进展, 2010, 18 (09): 1359-1368.
- [14] 薛雨丽, 毛峡, 郭叶, 等. 人机交互中的人脸表情识别研究进展 [J]. 中国图象图形学报, 2009, 14 (05): 764-772.
- [15] 马瑞, 王家威, 宋亦旭. 基于局部线性嵌入(LLE)非线性降维的多流形学习 [J]. 清华大学学报(自然科学版), 2008, (04): 582-585. DOI:10.16511/j.cnki.qhdxxb.2008.04.030.
- [16] 左坤隆, 刘文耀. 基于活动外观模型的人脸表情分析与识别 [J]. 光电子·激光, 2004, (07): 853-857.
- [17] 张炜, 王庆, 赵荣椿. 汽车牌照的实时识别 [J]. 信号处理, 2000, (04): 372-375.

附录1 表情系统编码规则

EMOTION_LABELS = {

0: '愤怒',
1: '厌恶',
2: '恐惧',
3: '快乐',
4: '悲伤',
5: '惊讶',
6: '平静'

EMOTION_MICRO_LABELS = {

(0, 1): '憎恨({0}/{1} {2}%)',
(0, 2): '紧张({0}/{1} {2}%)',
(0, 3): '挑衅({0}/{1} {2}%)',
(0, 4): '沮丧({0}/{1} {2}%)',
(0, 5): '震惊({0}/{1} {2}%)',
(0, 6): '冷漠({0}/{1} {2}%)',
(1, 2): '疑虑({0}/{1} {2}%)',
(1, 3): '反常({0}/{1} {2}%)',
(1, 4): '绝望({0}/{1} {2}%)',
(1, 5): '复杂({0}/{1} {2}%)',
(1, 6): '无聊({0}/{1} {2}%)',
(2, 3): '反常({0}/{1} {2}%)',
(2, 4): '复杂({0}/{1} {2}%)',
(2, 5): '复杂({0}/{1} {2}%)',
(2, 6): '疑虑({0}/{1} {2}%)',
(3, 4): '矛盾({0}/{1} {2}%)',
(3, 5): '惊喜({0}/{1} {2}%)',
(3, 6): '满足({0}/{1} {2}%)',
(4, 5): '震惊({0}/{1} {2}%)',
(4, 6): '失望({0}/{1} {2}%)',
(5, 6): '矛盾({0}/{1} {2}%)',
(0, 1, 2): '复杂({0}/{1}/{2} {3}%)',
(0, 1, 3): '嘲笑({0}/{1}/{2} {3}%)',
(0, 1, 4): '失望({0}/{1}/{2} {3}%)',
(0, 1, 5): '复杂({0}/{1}/{2} {3}%)',
(0, 1, 6): '冷酷({0}/{1}/{2} {3}%)',
(0, 2, 3): '反常({0}/{1}/{2} {3}%)',
(0, 2, 4): '复杂({0}/{1}/{2} {3}%)',
(0, 2, 5): '复杂({0}/{1}/{2} {3}%)',
(0, 2, 6): '焦虑({0}/{1}/{2} {3}%)',
(0, 3, 4): '矛盾({0}/{1}/{2} {3}%)',
(0, 3, 5): '激动({0}/{1}/{2} {3}%)',
(0, 3, 6): '挑衅({0}/{1}/{2} {3}%)',
(0, 4, 5): '悲愤({0}/{1}/{2} {3}%)',

```

(0, 4, 6): '失望({0}/{1}/{2} {3}%)',
(0, 5, 6): '矛盾({0}/{1}/{2} {3}%)',
(1, 2, 3): '反常({0}/{1}/{2} {3}%)',
(1, 2, 4): '绝望({0}/{1}/{2} {3}%)',
(1, 2, 5): '复杂({0}/{1}/{2} {3}%)',
(1, 2, 6): '疑虑({0}/{1}/{2} {3}%)',
(1, 3, 4): '矛盾({0}/{1}/{2} {3}%)',
(1, 3, 5): '反常({0}/{1}/{2} {3}%)',
(1, 3, 6): '挑衅({0}/{1}/{2} {3}%)',
(1, 4, 5): '悲愤({0}/{1}/{2} {3}%)',
(1, 4, 6): '失望({0}/{1}/{2} {3}%)',
(1, 5, 6): '矛盾({0}/{1}/{2} {3}%)',
(2, 3, 4): '矛盾({0}/{1}/{2} {3}%)',
(2, 3, 5): '不安({0}/{1}/{2} {3}%)',
(2, 3, 6): '反常({0}/{1}/{2} {3}%)',
(2, 4, 5): '复杂({0}/{1}/{2} {3}%)',
(2, 4, 6): '复杂({0}/{1}/{2} {3}%)',
(2, 5, 6): '矛盾({0}/{1}/{2} {3}%)',
(3, 4, 5): '矛盾({0}/{1}/{2} {3}%)',
(3, 4, 6): '矛盾({0}/{1}/{2} {3}%)',
(3, 5, 6): '惊喜({0}/{1}/{2} {3}%)',
(4, 5, 6): '矛盾({0}/{1}/{2} {3}%)',
}

```

说明：1. 总文字复制比：被检测论文总重合字数在总字数中所占的比例

2. 去除引用文献复制比：去除系统识别为引用的文献后，计算出来的重合字数在总字数中所占的比例

3. 去除本人文献复制比：去除作者本人文献后，计算出来的重合字数在总字数中所占的比例

4. 单篇最大文字复制比：被检测文献与所有相似文献比对后，重合字数占总字数的比例最大的那一篇文献的文字复制比

5. 复制比：按照“四舍五入”规则，保留1位小数

6. 指标是由系统根据《学术论文不端行为的界定标准》自动生成的

7. 红色文字表示文字复制部分；绿色文字表示引用部分（包括系统自动识别为引用的部分）；棕灰色文字表示系统依据作者姓名识别的本人其他文献部分

8. 本报告单仅对您所选择的比对时间范围、资源范围内的检测结果负责



 amlc@cnki.net

 <https://check.cnki.net/>