

中国大学生计算机设计大赛



软件开发类作品文档

作品编号： 2025020887

作品名称： 慧图工坊—机器学习与图像处理综合教学实验平台

作 者： 郭思宇、闫建瑾

版本编号： V1.0

填写日期： 2025 年 04 月 13 日

填写说明：

- 1、本文档适用于所有涉及软件应用与开发类的各个小类作品，包括：（1）Web 应用与开发（2）管理信息系统（3）移动应用开发（非游戏类）（4）算法设计与应用（5）企业赛项；
- 2、本文档为简要文档，不宜长篇大论，需简明扼要，建议设计二级目录，逻辑性强；
- 3、一级标题采用二号黑体，居中，二级标题采用三号黑体，靠左，根据需要可以设计三级标题，正文一律用五号宋体；
- 4、提交文档时，以 PDF 格式提交本文档；
- 5、本文档内容是正式参赛内容组成部分，务必真实填写。如不属实，将导致奖项等级降低甚至终止本作品参加比赛。
- 6、项目中涉及应用人工智能大模型的内容，务必在各个部分说明清楚。

目 录

第 1 章 需求分析.....	1
1.1 开发缘由.....	1
1.2 用户角色及功能需求.....	1
1.2.1 学生角色功能需求.....	1
1.2.2 教师角色功能需求.....	1
1.2.3 管理员角色功能需求.....	1
1.3 性能需求.....	1
第 2 章 概要设计.....	2
2.1 体系结构.....	2
2.2 技术设计.....	2
第 3 章 详细设计.....	4
3.1 界面设计.....	4
3.1.1 总体规划.....	4
3.1.2 平台登录.....	4
3.1.3 学生用户.....	4
3.1.4 教师用户.....	6
3.1.5 管理员用户.....	7
3.2 数据库设计.....	8
3.2.1 概念设计.....	8
3.2.2 实体关系.....	9
3.2.3 数据字典.....	10
3.3 后端设计.....	11
3.4 评分模块设计.....	14
3.4.1 图像分类评分模块.....	14
3.4.2 图像分割评分模块.....	15
3.4.3 目标识别评分模块.....	18
3.4.4 答案配置设计.....	20
3.5 创新点与技术难点.....	21
第 4 章 主要测试.....	22
第 5 章 安装与使用.....	23
5.1 安装.....	23
5.2 使用.....	23
第 6 章 项目总结.....	24
第 7 章 开发团队.....	24
7.1 学生开发团队.....	24
7.2 指导教师.....	24
参考文献.....	25

第1章 需求分析

1.1 开发缘由

近年来，人工智能技术的快速发展推动高等教育课程体系的深度变革。作为核心领域，机器学习与图像处理课程的教学目标已从理论传授向综合能力培养转型。然而，传统实验教学模式面临严重依赖人工核验、实验类型局限、教学反馈滞后等问题。

据教育部 2022 年《人工智能本科专业建设报告》显示，超过 73% 的高校反馈实验教学环节存在“理论实践脱节”与“评价体系缺失”问题。为此，我们开发了“慧图工坊—机器学习与图像处理综合教学实验平台”（下文简称“本实验平台”），目的是通过数字化、信息化的方式，解决教与学的矛盾，提高育人效果。

1.2 用户角色及功能需求

基于对实际教学环境的考察调研，开发组认为有学生、教师和管理员三类主要用户角色。所有用户均需要通过身份认证进入实验平台，分角色开放不同权限模块。

1.2.1 学生角色功能需求

学生是教学活动中的直接受众群体，有以下功能需求：

- (1) 实验选择：学生可查看平台上所有实验项目，加入实验、查看实验详情。
- (2) 结果上传：学生能在平台上提交实验结果，在教师允许范围内调整实验结果。
- (3) 成绩查看：学生可查看实验成绩和排名，实现学生对实验的深入理解。

1.2.2 教师角色功能需求

教师在教学活动中有引领性、指导性作用，有以下功能需求：

- (1) 实验创建与管理：教师可创建、管理实验项目，设置实验相关要求。
- (2) 学生管理：教师可管理学生信息，查看学生参与实验的情况。
- (3) 评分管理：教师通过设定答案或制作评分服务接口进行打分。
- (4) 成绩查看：教师可查看学生的实验成绩及排名，形成过程化评价结果。

1.2.3 管理员角色功能需求

管理员负责管理实验课程，同时负责系统维护，有以下功能需求：

- (1) 用户管理：管理员能够管理所有用户账户信息。
- (2) 数据维护：管理员可查看所有实验项目，删除失效实验内容，更新实验信息。
- (3) 质量控制：管理员可配合学校学院等行政管理部门进行教学质量监控和评估。

1.3 性能需求

实验平台针对学校信息化建设相关要求，实验平台从四个方面确定性能需求：

- (1) 高并发支持：实验平台必须支持多个用户同时在线操作。
- (2) 实时数据处理：立即对学生结果进行评分、实时反馈。
- (3) 跨平台兼容性：针对学生端不同的设备，适配不同的屏幕尺寸。
- (4) 系统稳定性：实验平台需要长时间运行，具备高可用性。

第2章 概要设计

2.1 体系结构

实验平台采用了前后端分离+评分服务的混合架构，总体上采用 B/S 架构模式，便于跨平台使用，支持不同操作系统的设备。基于用户角色进行出发，设计了如图 2.1 所示的模块，系统总体分为学生、教师和管理员三大角色，每个角色拥有独立的功能模块，彼此之间通过权限和共享数据进行通信。

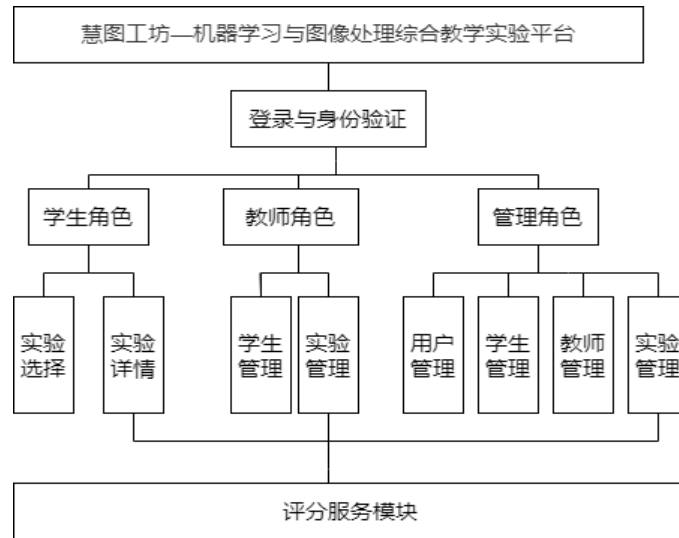


图 2.1 模块构架图

2.2 技术设计

针对系统的实际用途和功能实现需求，项目组经过充分调研与技术评估，最终确定前端表现层部分采用现代化的 Vue 与 Vite 技术栈进行开发。Vue 作为当前广泛使用的渐进式 JavaScript 框架，具备响应式数据绑定、高效组件化开发以及丰富的生态支持，能够有效提升前端开发效率和用户体验。结合 Vite 的快速构建与热重载特性，开发团队能够实现极佳的开发调试体验及快速的生产环境构建，保证系统前端界面响应迅速、交互流畅。此外，Vue 的组件化设计使得页面模块复用性高，维护升级更加便捷，满足实验平台日后迭代更新的需求。

在后端技术选型方面，项目组采用基于 Java JDK 的 Spring Boot 与 MyBatis 技术栈。Spring Boot 作为 Java 生态中最成熟且广泛应用的微服务框架，提供了自动化配置、内嵌服务器、丰富的扩展插件以及强大的社区支持，极大简化了后端服务的开发与部署流程。通过 Spring Boot 的依赖注入和模块化设计，系统业务逻辑结构清晰、模块边界明确，有效提升代码的可维护性和扩展性。MyBatis 作为一款轻量级持久层框架，兼具 SQL 语句灵活编写的优势及良好的映射功能，既满足了复杂数据库操作的灵活性，也保证了数据库访问性能与安全性。该技术组合能够稳定高效地支持实验平台的核心业务逻辑和数据处理，确保系统在用户量增加时具备良好的扩展能力。

评分服务部分则采用了基于 Python 的 Flask 框架驱动，形成了系统内的一个独立轻量级微服务。Flask 以其极简设计和灵活性著称，非常适合构建轻量级的 API 接口服务。评分模块的设计要求支持图像处理、数据计算和快速响应，Python 丰富的科学计算库（如

NumPy、Pandas、OpenCV）为图像处理和算法实现提供了强有力的支撑。通过 Flask，评分模块可独立部署、独立维护，且能够通过 RESTful 接口与后端服务实现无缝交互，有效实现了系统的松耦合设计。采用 Python 作为评分模块的语言选择，不仅利用了其在人工智能和图像处理领域的技术优势，也使得评分算法的快速迭代和优化成为可能，满足教学评估对算法准确性和灵活性的高要求。

三部分技术栈的合理结合，形成了系统的混合式异构架构模式。该模式充分发挥了各自技术栈的优势：前端 Vue+Vite 提供灵活、现代的用户交互体验，后端 Spring Boot+MyBatis 保障业务逻辑的稳定和高效运行，评分模块 Flask 则专注于专业化的图像与数据评分处理。各模块间通过明确的接口协议实现协同工作，既保证了系统整体性能的最优化，也为后续各模块的独立升级和技术替换预留了充分空间。该异构架构在分布式环境下运行时，能够显著提升系统的弹性和容错能力，确保平台在面对多用户并发访问时仍保持良好响应速度和服务稳定性。

实验平台前后端分别部署于 2 台基于 Linux 操作系统的服务器，前端服务器支持公网访问，方便教师和学生随时随地通过互联网访问系统。Linux 系统以其安全稳定、资源利用高效和强大的网络服务能力成为服务器端部署的首选。系统架构遵循经典的浏览器/服务器（B/S）设计模式，前端页面通过浏览器访问，后端和评分模块作为独立服务响应请求。系统支持前端、后端、评分模块的分布式部署，将不同功能模块分别部署在不同服务器或容器环境中，有效分担单一服务器的运行压力，防止单点故障导致整体服务中断。分布式部署还方便根据访问负载动态调整资源，提升系统扩展性和高可用性。

此外，分布式部署模式符合现代企业级应用设计规范，有利于系统的安全防护和权限管理。前端与后端之间通过严格的接口认证和权限校验保证数据安全，评分模块作为独立的服务可通过网络隔离和访问控制增强整体系统的安全防御能力。同时，该设计方便系统未来结合云计算和容器化技术，实现更灵活的资源调度和自动化运维，提升系统运维效率和服务质量。

综上所述，项目组选择 Vue+Vite 技术栈作为前端开发基础，Spring Boot+MyBatis 作为后端核心框架，以及 Python Flask 驱动评分服务，形成了一套结构清晰、功能完整且高效稳定的实验教学平台。系统的混合异构架构与分布式部署策略不仅提升了系统的整体性能和可维护性，还确保了平台能够适应不断变化的教学需求和信息化建设趋势，充分发挥技术优势，实现教学管理与自动化评分的深度融合，为现代实验教学提供坚实的信息化支撑。

第3章 详细设计

3.1 界面设计

3.1.1 总体规划

平台页面以模块化设计为主要风格，如图 3.1 所示，页面分为功能区、数据操作区和全局导航区。功能区设置在页面左侧，顶部显示当前用户，为不同用户提供不同服务入口。数据操作区是页面中最大的区域，是各类服务和数据的具体显示区域，为数据显示设置了自动分页功能，方便用户查看。全局导航区在页面顶部，用户登录后使用，集成了弹窗提示功能，设置操作帮助和系统信息入口，方便各类用户使用查询。

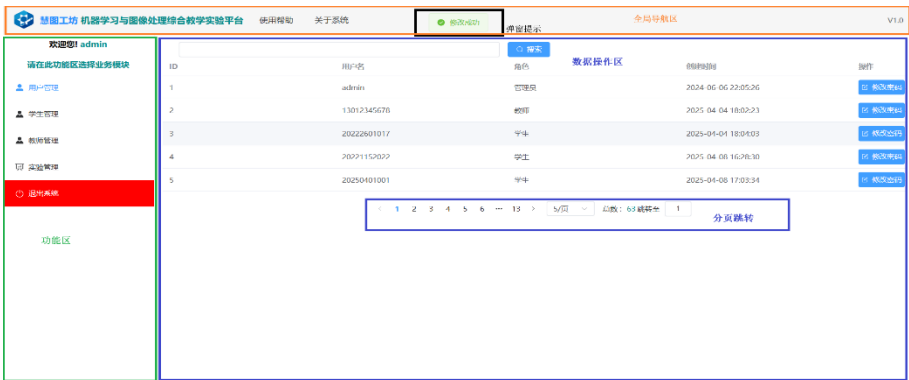


图 3.1 页面布局示意

3.1.2 平台登录

各类用户在系统注册后，可凭借自己的学号和默认密码登录平台。如图 3.2 所示，登录时需要勾选“我了解此系统使用规则和方法”“我保证不向他人传播题目和答案”的复选框才能正确登录实验平台，设计了异步等待和错误自动刷新规则，满足人机交互要求。



图 3.2 登录页面

3.1.3 学生用户

学生登录平台后在侧边栏选择“加入实验”，如图 3.3 所示，页面将展示目前平台上所有已发布的所有实验。点击对应实验后的“加入实验”按钮并二次确认即可加入实验。



图 3.3 学生实验选择加入页面

如图 3.4 所示，“我的实验”页面将显示学生已经选择加入的实验，点击“详情”按钮可进入如图 3.5 所示的“实验详情”页面来查看教师要求、下载附件和上传答案，也可在排行榜中查看自己的分数和排名。



图 3.4 “我的实验”页面



图 3.5 实验详情页面

提交作业实验结果需要点击“实验详情”页面的“点此上传作业”按钮。如图 3.6 所示，在对话框中选择对应文件后点击确认按钮，上传成功将减少一次上传机会，同时根据教师评判服务的标准当场计算实验分数和排名。

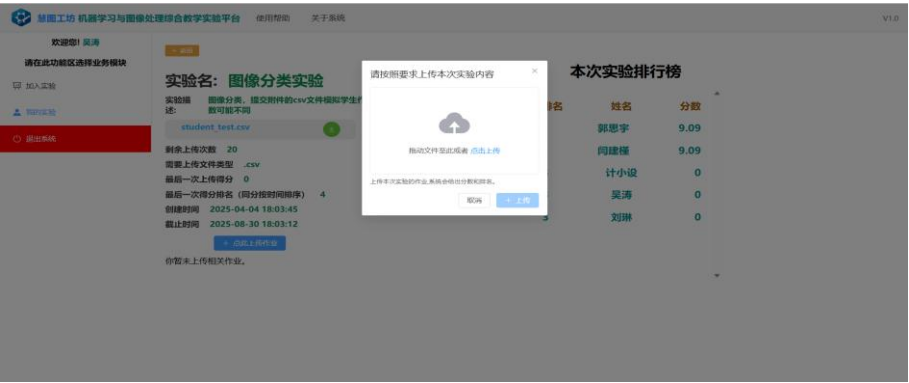


图 3.6 实验作业上传

3.1.4 教师用户

教师用户登录平台后可以在侧边栏选择“学生管理”，如图 3.7 所示，将展示目前平台上所有已经注册的学生列表，教师在这里可以对学生进行单独或批量化增删改查操作。

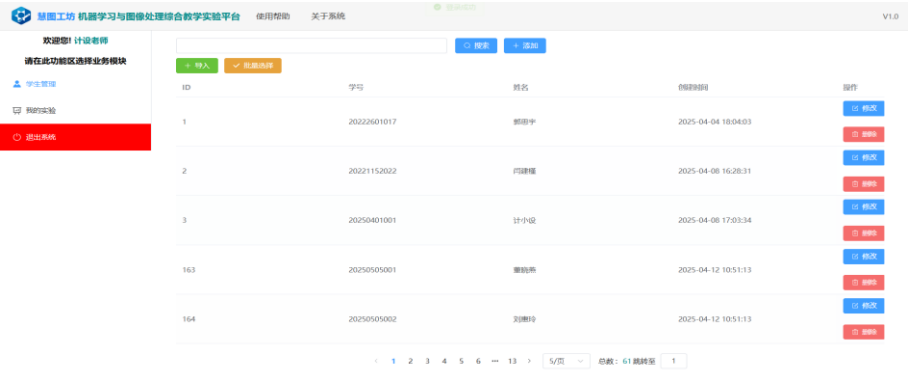


图 3.7 学生管理页面

教师登录后在侧边栏点击“我的实验”，如图 3.8 所示，默认显示当前教师发布的所有实验及相关要求等信息；教师用户可以增删改查实验项目，图 3.9 所示的是增加实验的页面。教师用户点击对应实验后的“详情”按钮，可以进入如图 3.10 所示的“实验详情”页面下载实验资源、查看学生排行榜、上传实验附件操作。

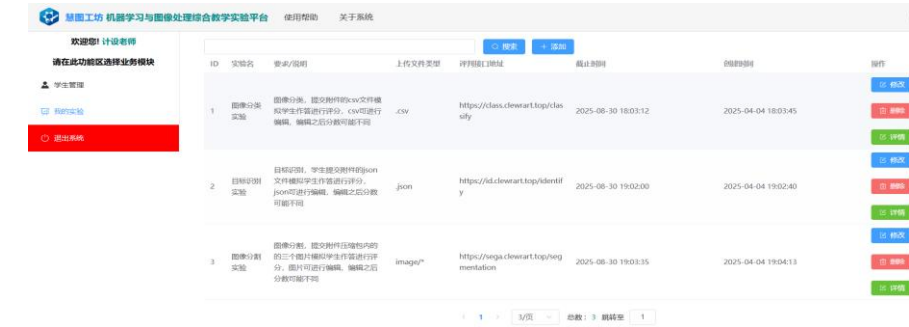


图 3.8 实验管理页面

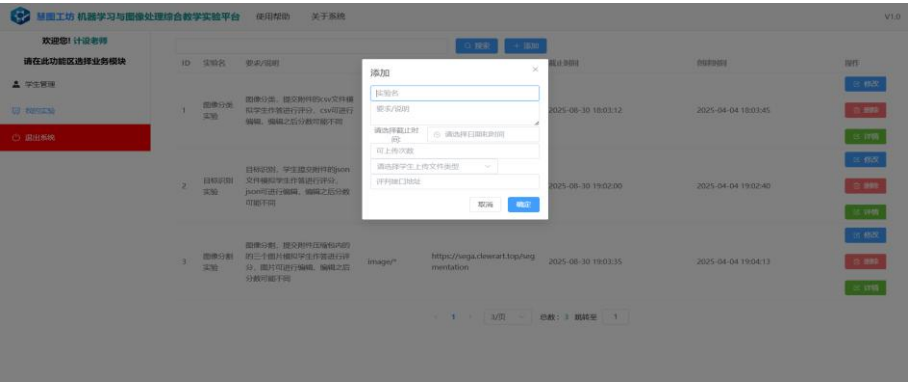


图 3.9 实验增加页面



图 3.10 实验详情页面

3.1.5 管理员用户

管理员用户是平台最高权限用户，在管理权限上可视为教师用户的超集。特别设计如图 3.11 所示的“用户管理”功能，用于对平台各类账户密码进行修改和管理。“实验管理”功能管理所有实验项目，操作逻辑同教师用户一致；本处不再赘述。“教师管理”“学生管理”功能的操作与设计与教师用户一致，本处不再赘述。

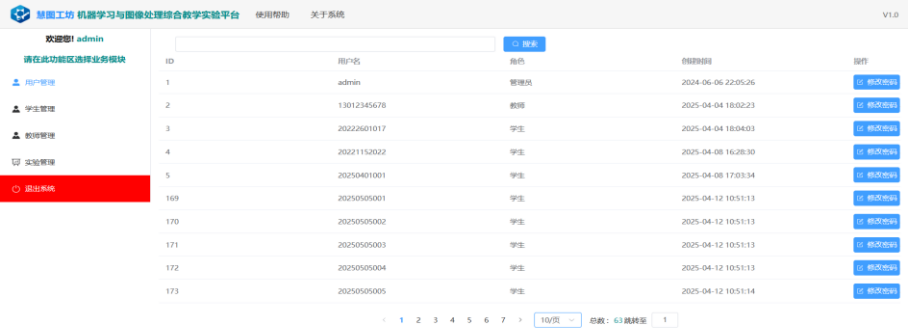


图 3.11 管理员“用户管理”功能

3.2 数据库设计

实验平台的数据库采用 MySQL 8 系列数据库，设计多个数据表及关系。

3.2.1 概念设计

本实验平台的数据库在概念设计阶段采用实体-关系模型（Entity-Relationship Model, ER 模型）进行建模，遵循“高内聚、低耦合”的设计原则，通过对实验教学全过程中核心对象的建模，有效支撑系统一系列功能，满足现代教学场景下多角色、多任务的管理需求，为后续系统功能模块的实现与优化提供坚实的数据支撑与结构保障。

平台通过抽象化的方式设计系统涉及的主要实体、属性及其之间的联系。该平台主要面向教师和学生两类用户群体，为实验课程教学全生命周期提供全面的信息支撑，包括实验任务发布、学生参与、文档提交与系统自动评分等多个核心功能模块。为了满足这些功能需求，平台在数据库层面设计了多个互相关联的数据实体，确保平台运行的逻辑合理性、数据一致性以及系统的可维护性和可扩展性。

系统的用户信息采用统一管理的方式，所有登录系统的人员，无论是教师还是学生，均首先在 `user` 表中建立用户记录。`user` 表存储每个用户的基础信息，包括用户名、加密后的密码、用户角色（如 `teacher` 或 `student`）、用户注册时间等，便于系统统一控制用户登录权限、身份识别及后续的数据调用。用户角色字段（`role`）在系统中起到区分权限和功能范围的作用，是进行权限管理和页面分发的重要依据。

基于角色的差异，系统分别设计了 `teacher` 表和 `student` 表，用于存储教师和学生的扩展信息。两者通过与 `user` 表中的 `id` 字段建立一对一的逻辑关系实现继承结构。`teacher` 表记录教师的真实姓名、联系电话和创建时间等；`student` 表则包括学生的姓名、学号以及账户创建时间等字段。通过这种结构，系统既能统一管理所有用户的公共属性，又能细致记录不同角色用户的特定信息，确保数据组织的清晰与规范。

实验相关信息存储在 `experiment` 表中，每条记录对应一个由教师发布的实验任务。实验信息包括实验名称、实验说明、提交文件类型要求、实验评测的接口地址（`judge_url`）、允许的总上传次数、截止时间以及创建时间等内容。`experiment` 表中的 `teacher_id` 字段作为外键与 `teacher` 表建立关联，用于表示该实验由哪位教师发布，从而实现教师与实验之间的一对多关系。

考虑到实际教学过程中，一个学生可能参与多个实验，一个实验也可能有多名学生参与，因此平台设计了 `experiment_student` 中间表，用于实现学生与实验之间的多对多关系。该表不仅承担关系映射的功能，还额外记录了学生参与实验的动态信息，包括该学生在某实验中剩余的上传次数（`upload_count`）、实验得分（`score`）以及记录生成时间等。该设计为后续的实验成绩统计、上传限制控制和学习过程分析提供了坚实的数据基础。

平台允许学生和教师上传与实验相关的各类文档，如实验报告、测试数据、评测模型等，文档数据通过 `document` 表进行统一管理。`document` 表中的每一条记录代表一个上传的文档文件，包含所属实验编号（`experiment_id`）、上传者用户编号（`user_id`）、用户角色（`user_role`）、原始文件名（`origin_name`）、系统保存名（`save_name`）、文件存储路径（`file_path`）和上传时间等信息。其中，`experiment_id` 和 `user_id` 均作为外键分别关联到 `experiment` 表和 `user` 表，实现文档与上传者及对应实验之间的明确联系。通过设置 `user_role` 字段（枚举类型），系统可以清晰地区分上传者是学生还是教师，为后续的功能区分、权限控制和文档分类提供支持。

在整体数据结构设计中，系统重视数据的完整性与一致性。在所有表的设计中，关键字段均设置为非空字段并建立主键，相关实体之间通过外键进行连接和约束。例如，

experiment 表中的 teacher_id 引用 teacher 表的 id 字段，document 表中通过 experiment_id 和 user_id 分别引用实验和用户实体，experiment_student 表同时引用 experiment 和 student，确保系统数据的关联性和逻辑严密性。这种设计不仅有助于维护数据的准确性，也便于在系统运行过程中进行数据的联合查询、统计分析和事务管理。

3.2.2 实体关系

user 表存储所有用户信息，teacher 和 student 表分别存储教师和学生的详细信息，并通过 user 表的 id 字段关联。

experiment 表记录实验的基本信息，与 teacher 表通过 teacher_id 字段关联，表示实验由教师创建。experiment_student 表实现学生与实验的多对多关系，记录学生参与实验的情况及得分。document 表存储学生或教师上传的文档信息，与 experiment 表和 user 表分别通过 experiment_id 和 user_id 字段关联。

整体设计通过外键约束确保数据一致性，支持教师创建实验、学生参与实验、文档上传与评分等功能，满足实验教学的需求。图 3.12 所示的是实验平台数据库 UML 图。

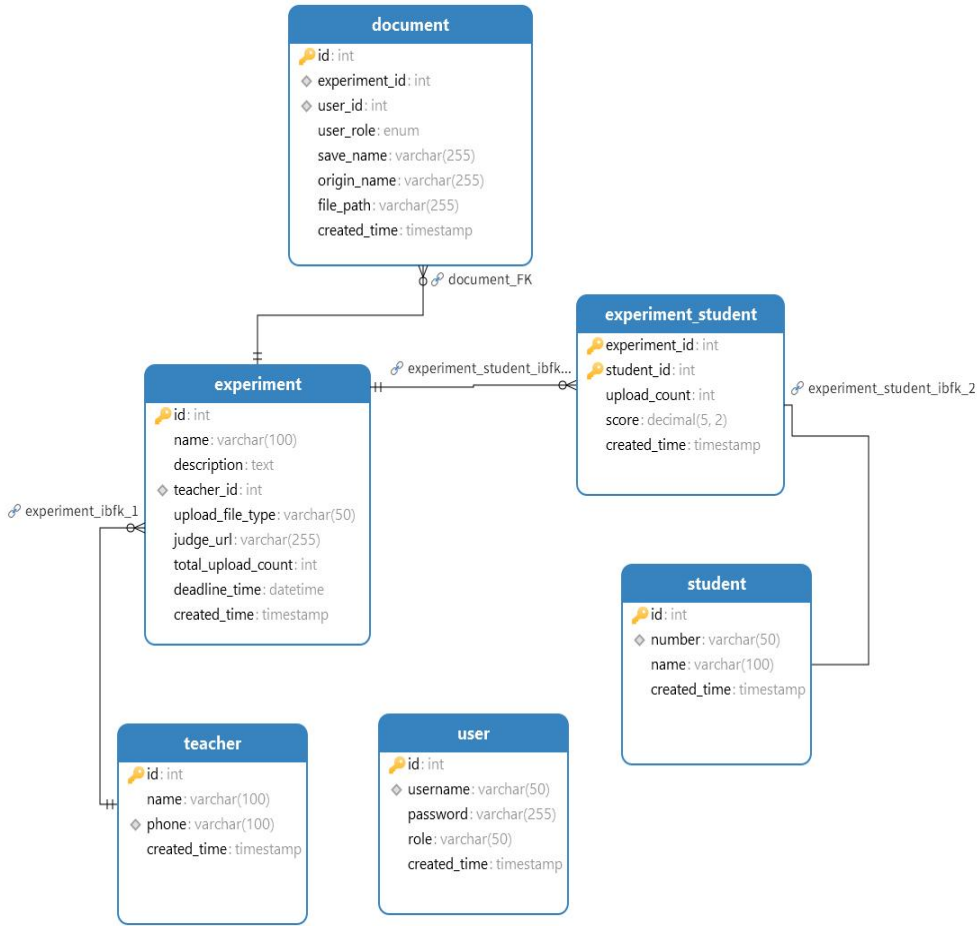


图 3.12 实验平台数据库 UML 图

3.2.3 数据字典

数据字典如下表 3.1-3.6 所示。

表 3.1 系统数据表 user 数据字典

数据表名	数据字段名	内容	数据类型	长度	主键
User	Id	记录系统用户序号	Int	11	Y
	Username	记录系统用户名	Varchar	50	
	Password	记录用户加密后的密码	Varchar	255	
	Role	记录用户角色组	Varchar	50	
	Created_time	记录用户创建时间	Timestamp		

表 3.2 实验系统数据表 teacher 数据字典

数据表名	数据字段名	内容	数据类型	长度	主键
Teacher	Id	记录系统用户号	Int	11	Y
	Name	记录教师用户名	Varchar	100	
	Phone	记录教师用户电话	Varchar	100	
	Created_time	记录用户创建时间	Timestamp		

表 3.3 实验系统数据表 student 数据字典

数据表名	数据字段名	内容	数据类型	长度	主键
Student	Id	记录系统用户号	Int	11	Y
	Name	记录学生用户名	Varchar	100	
	Number	记录学生用户学号	Varchar	50	
	Created_time	记录用户创建时间	Timestamp		

表 3.4 实验系统数据表 experiment_student 数据字典

数据表名	数据字段名	内容	数据类型	长度	主键
Experiment_student	Experiment_id	记录实验号	Int	11	Y
	Student_id	记录参与实验学生用户号	Int	11	Y
	Upload_count	记录剩余上传次数	Int	11	
	Score	记录学生分数	Decimal	5,2	
	Created_time	记录实验创建时间	Timestamp		

表 3.5 实验系统数据表 experiment 数据字典

数据表名	数据字段名	内容	数据类型	长度	主键
Experiment	Id	记录实验号	Int	11	Y
	Name	记录教师用户名	Varchar	100	
	Description	记录实验描述	Text		
	Upload_file_type	记录提交文件格式要求	Varchar	50	
	Judge_url	记录实验判别接口	Varchar	255	
	Total_upload_count	记录实验上传次数设定	Int	11	
	Deadline_time	记录实验截止时间	Datetime		
	Created_time	记录实验创建时间	Timestamp		

表 3.6 实验系统数表 document 数据字典

数据表名	数据字段名	内容	数据类型	长度	主键
Document	Id	记录文档号	Int	11	Y
	Experiment_id	记录实验号	Int	11	
	User_id	记录用户号	Int	11	
	User_role	记录用户角色	Enum	2	
	Save_name	记录文档存储名	Varchar	255	
	Origin_name	记录文档原文件名	Varchar	255	
	File_path	记录文档存储路径	Varchar	255	
	Created_time	记录文档上传时间	Timestamp		

3.3 后端设计

后端部分选用 Spring Boot 与 MyBatis Plus 作为基础架构，充分发挥二者在现代微服务与快速开发场景中的优势。Spring Boot 通过“一站式”自动配置、内嵌容器与约定优于配置（Convention over Configuration）理念，大幅度减少了繁琐的 XML 注解配置，使得项目启动脚手架仅需少量代码即可完成环境搭建；同时内置 Tomcat、Jetty 等容器，提供开箱即用的运行支持，简化部署流程、缩短交付周期。在项目启动时，Spring Boot 会根据 classpath 中的依赖自动加载合适的配置类，并通过 @SpringBootApplication、@EnableAutoConfiguration、@ComponentScan 等注解实现对 Bean 容器的扫描与管理，开发者只需聚焦于业务逻辑，无需手动管理底层依赖和基础设施。

MyBatis Plus 则在 MyBatis 核心框架之上提供了一套增强工具集，它保留了原生 SQL 的可控性与灵活性，同时通过代码生成、通用 CRUD 接口和条件构造器等措施大幅简化了

数据库访问层的开发。借助 MyBatis Plus，开发者可以通过实体类与 Mapper 接口的映射关系，快速生成常用的增删改查方法，无需编写重复的 XML 映射文件；同时凭借原生的 @Mapper 注解与 @SqlParser 等扩展特性，业务中涉及的复杂查询与批量操作也能通过自定义 SQL 或者插件化方式轻松完成。在该架构下，绝大多数数据访问逻辑可以在几行代码内搞定，开发效率成倍提升，且由于依然使用手写 SQL 或者 XML 动态 SQL，系统在性能调优与 SQL 优化方面拥有充分的自主权与可观测性。

项目采用前后端分离部署模式，前端页面通过 Vue3、Element Plus 框架构建，在浏览器端负责呈现界面并收集用户输入；后端系统则完全以 RESTful API 的形式对外暴露业务接口。每个业务模块对应一组专用 URI，例如/api/experiments、/api/users、/api/documents 等，在 Controller 层使用 @RestController、@RequestMapping、@GetMapping、@PostMapping 等注解定义路由与请求方法，接收前端参数、调用 Service 层执行核心业务，并将结果以 JSON 形式返回。这样的交互方式使得前端、后端团队可以独立并行开发，接口契约清晰易测，并且未来如需替换前端技术栈或接入第三方客户端，也只需遵循既定的 API 协议即可，无需对后端代码进行大规模重构。

在安全与权限控制方面，系统引入 Spring Security 框架，通过拦截器（Filter）与认证管理器（Authentication Manager）对用户请求进行统一拦截与鉴权。登录时，后端根据用户提供的用户名与密码，查询数据库并验证身份，若验证通过，则生成 JWT（JSON Web Token）并返回给前端；后续所有接口调用均需在 HTTP Header 中携带该 Token，后端通过 @PreAuthorize、@RolesAllowed 等注解及自定义的 Authentication Entry Point 和 Access Denied Handler 完成基于角色的权限判定与异常处理。无状态的认证方式不仅减轻服务器存储会话的压力，也让多节点部署时的用户状态管理变得更加简洁高效。

为保证数据一致性，服务层在方法上使用 @Transactional 注解管理事务，MyBatis Plus 与 Spring Transaction 整合后，可针对分布式事务场景获后续扩展微服务以及多数据源进一步接入 Seata、Atomikos 等分布式事务解决方案。数据库连接池结合 Spring Boot 的自动配置，只需在 application.yml 中配置少量参数，即可获得高可用连接池实例。

性能优化还体现在缓存与异步处理上。针对频繁读取但不经常变动的数据，如系统配置、实验模板、教师信息等，系统可借助 Spring Cache 实现二级缓存，将热点数据缓存在内存或分布式缓存中，降低数据库访问压力。对于耗时操作，如大文件处理、自动评分任务的异步执行，则使用 Spring Boot Starter 提供的 @Async 注解或集成消息队列（RabbitMQ、Kafka 等），将任务发送到消息中间件，由工作线程或独立的微服务进程异步消费并执行，避免阻塞主请求线程，提高系统吞吐量与响应速度。

在模块化设计方面，项目以 Maven 多模块或 Gradle 子项目的方式进行组织，将公共依赖、实体（entity）、DTO（Data Transfer Object）、工具类（utils）等提取为基础模块，业务功能则按实验管理、用户管理、文档管理、权限管理等领域划分子模块，各自独立构建与测试。这样不仅有助于职责分离，也让团队协作更具条理；当业务量或团队规模扩大时，可将部分子模块拆分为独立微服务，借助 Spring Cloud 或者 Kubernetes 进行容器化部署与服务编排，实现更高水平的弹性伸缩与故障隔离。

部署方面，由于 Spring Boot 支持打包为可执行 JAR 或 WAR，结合 Docker 容器化部署后，CI/CD（持续集成/持续交付）流程可通过 GitHub Actions 等自动化工具完成镜像构建、单元测试、集成测试与灰度发布，最终将镜像推送到私有或公共的容器仓库，并在 Kubernetes 集群或 Docker Compose 上一键部署，确保从代码提交到生产环境的全链路自动化与可回溯。

总的来说，本系统后端以快速开发、模块清晰、性能可观测与易于运维为目标，通过一系列成熟的技术栈和最佳实践相结合，为实验教学平台提供了稳定可靠且高效可扩展的

后端服务支撑。未来可根据业务需要进一步引入微服务拆分、多活部署等先进架构，实现更高层次的伸缩能力与技术演进。支持跨平台部署与并发处理，提升系统性能与稳定性，满足复实验平台的业务承载需求。后端 UML 图如图 3.13 所示。

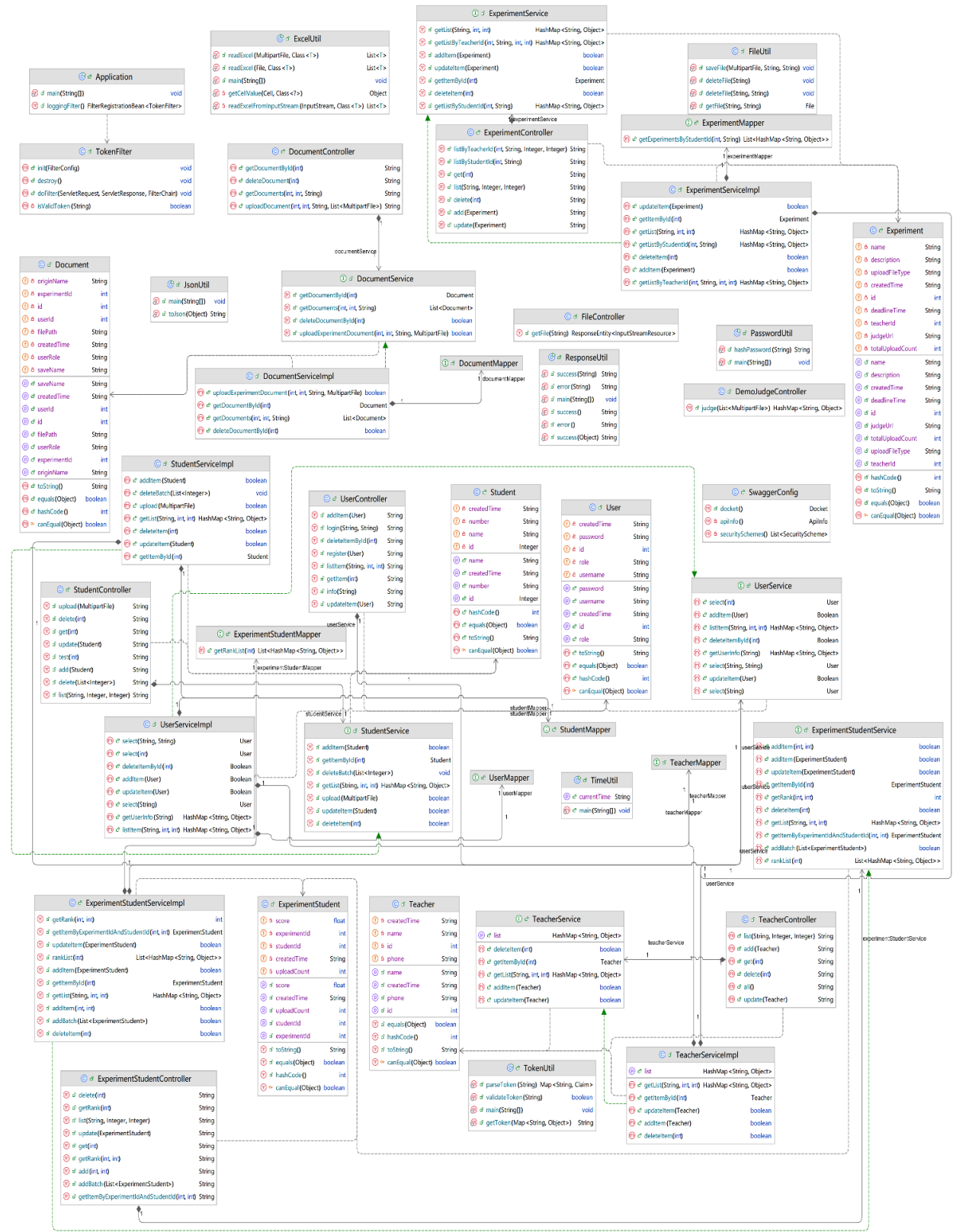


图 3.13 后端 UML 图

3.4 评分模块设计

3.4.1 图像分类评分模块

图像分类评分模块使用 Flask 框架搭建了一个轻量级的 Web 服务,旨在为图像处理与机器学习类实验课程提供自动化评分功能。该模块专为图像分类任务而设计,其核心功能是接收学生提交的分类预测结果文件,解析并与教师预设的标准答案进行比对,通过准确率转换为百分制得分反馈给平台。评分模块作为系统中的重要组成部分,具备部署简单、响应快速、可维护性强等特点,为学生与教师之间的教学反馈提供了高效、实时的支持。

在实际使用中,学生通常会完成图像分类实验任务,并根据实验平台的接口要求,将其模型预测的分类结果保存为 CSV(字符分隔值,Comma-Separated Values)格式的文件进行提交。该文件一般由两列构成,第一列为样本编号或图片 ID,第二列为对应预测的类别标签。教师端则在平台后台上传标准答案文件,也为 CSV 格式,表示每一个图像样本的真实分类标签。在评分流程启动时,Flask 应用首先通过接口接收学生上传的文件内容,并使用标准 Python 工具将其解析为 Python 字典类型的数据结构,以样本编号为键,对应的类别标签为值。这种格式化后的数据使得评分逻辑更为直观与高效。

该模块的核心评分逻辑由一个名为 `calc_score` 的函数实现。它接收两个字典类型的参数: `answer` 表示教师的标准答案, `real` 表示学生的作答结果。首先,通过 `len(answer)` 获取教师答案中样本的总数量,作为评分基准。随后,定义变量 `count` 用于统计学生作答中正确分类的样本数。通过遍历学生提交的每一个样本键值对,如果该样本编号在教师标准答案中存在,并且对应的标签也一致,则将计数器 `count` 增加 1。遍历结束后,以正确分类的样本数量除以总样本数得到准确率,并乘以 100 转换为百分制得分。通过 `round(..., 2)` 保留两位小数作为最终评分结果,以提供精确、清晰的得分反馈。主要评分代码如下,评分流程如图 3.14 所示。

```
def calc_score(answer:dict,real:dict):    /* 转换学生作答、教师预设答案为字典类型 */
    base=len(answer)                    /* 获取教师预设答案字典的长度 */
    count=0                             /* 初始化学生作答正确答案计数 */
    for r in real:
        if r in answer and answer[r]==real[r]: /* 若作答中存在与答案的对应的键值 */
            count=count+1                /* 则正确答案计数加 1, 直至检测完成 */
    return round(count/base * 100,2)      /* 以匹配率作为标准返回百分制分数 */
```

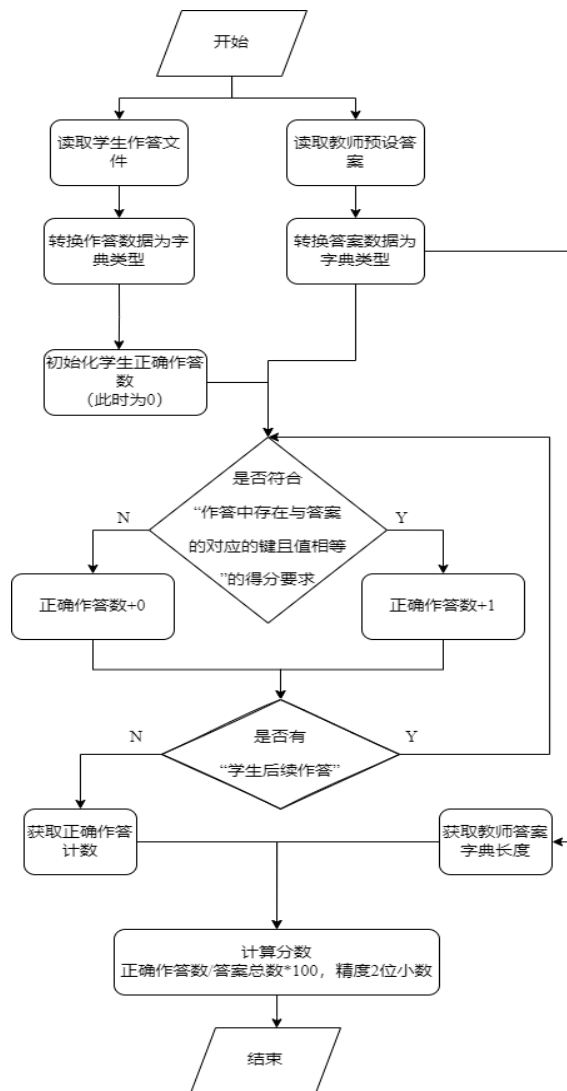



图 3.14 图像分类评分算法流程图

3.4.2 图像分割评分模块

图像分割自动化评分接口基于 Flask 框架搭建了一个轻量级 Web 服务，主要用于自动化评估学生提交的图像分割任务结果。该服务提供统一的接口，用以接收学生上传的分割结果图片集合，并将这些结果与教师预设的标准答案图片进行对比，最终计算得分反馈给教学管理平台。图像分割作为计算机视觉领域中的关键任务，其评分标准的科学性和自动化程度直接影响教学质量和效率，故本评分模块设计注重准确度、鲁棒性及运行效率。

在评分流程中，学生提交的图像分割结果通常是一组同名图像文件，每张图像代表一个待评分的样本。教师端则提供一套对应的标准答案图像集合，文件名与学生提交保持一致。评分模块首先对所有图像统一预处理，包括去除图像透明通道、灰度化处理及二值化操作。去除透明通道可消除图像中无效的透明像素影响，灰度化将彩色图像转换为单通道灰度图，简化数据结构，便于后续计算；二值化则通过设定阈值，将像素值分为前景与背景两部分，使得图像的前景区域轮廓清晰，方便进行像素级的比较。预处理后，所有图像均采用一致的格式和标准，保证评分计算的公平和准确。

接下来，评分模块通过遍历教师标准答案图像集合的每一个图像名称，在学生提交的结果集中查找同名图像。如果学生作答中存在该同名文件，则进一步调用 Dice 相似系数函数

计算两张图像的相似度。Dice 相似系数 (Dice Similarity Coefficient, DSC) 是评价图像分割结果准确性的重要指标, 其取值范围在 0 到 1 之间, 值越接近 1 表明学生提交的图像与标准答案在分割区域上越一致, 重叠部分越大。Dice 相似系数计算公式:

$$Dice(S, T) = \frac{2 \times |S \cap T|}{|S| + |T|}$$

其中 S 表示学生同名作答结果集中的像素数量, T 表示教师答案集中同名作答结果的像素数量; $|S \cap T|$ 表示学生作答与教师答案的交集像素数量。

引入 numpy 库进行计算 Dice 系数的代码如下:

```
def dice_coefficient(y_true, y_pred):
    intersection = np.sum(y_true * y_pred)  /* 求两张图片的交集像素数 */
    union = np.sum(y_true) + np.sum(y_pred) /* 求两张图片的总像素数 */
    if union > 0:                             /* 如果交并集总像素数大于 0 */
        return 2.0 * intersection / union    /* 计算 Dice 系数 */
    else:
        return 1.0                          /* 两图像均为空图, 认为 Dice 系数为 1 */
```

该函数首先将输入的两个二值化图像进行逐像素相乘, 得到交集像素的数量, 随后计算两张图像的前景像素和, 若和不为零, 则通过公式计算出 Dice 系数, 否则在两张图像均为空的情况下直接返回 1.0, 表示完美匹配。此设计既避免了除零错误, 也合理地处理了无前景图像的特殊情况, 保证评分的均一、稳定性。

```
for key in answer_file_dict:                /* 遍历所有标准答案图像 */
    if key in student_file_dict:            /* 若学生提交了同名图像 */
        /* 计算 Dice 系数 */
        dice = dice_coefficient(answer_file_dict[key], student_file_dict[key])
        total_dice += dice                  /* 累加 Dice 系数 */
    score = round(total_dice / len(answer_file_dict) * 100, 2) /* 得到百分制分数 */
评分流程如图 3.15 所示:
```

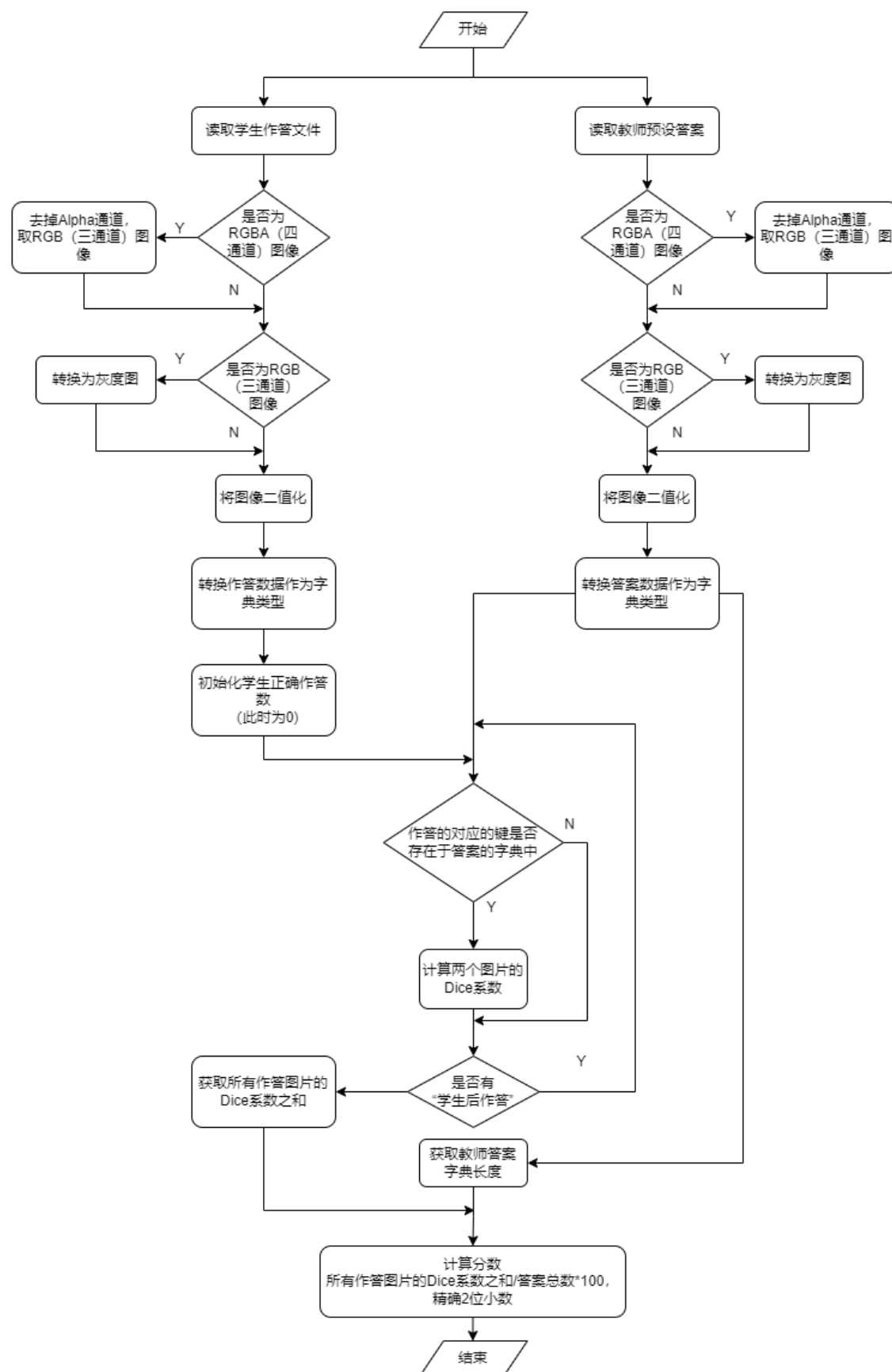


图 3.15 图像分割评分算法流程图

3.4.3 目标识别评分模块

目标识别自动化评分接口采用 Flask 框架搭建，旨在为图像目标检测任务的教学提供高效、准确的自动化评分服务。该服务通过设计轻量级 Web 接口，接收学生提交的目标检测结果文件，文件格式采用广泛使用的 JSON 格式，包含了目标的类别标签，还详细记录了检测框的位置和置信度信息，方便评分模块进行多维度的评价分析。评分模块的核心在于基于交并比（IoU，Intersection over Union）计算检测框的匹配情况，以及基于单类别的平均精确度（AP，Average Precision）计算准确度指标，最终综合生成全类别的平均精确率 mAP（mean Average Precision）作为评分标准，并将结果转换为百分制分数反馈。

相关参数解释如下：

(1) Iou 计算公式：

$$IoU = \frac{Area_{inter}}{Area_{box1} + Area_{box2} - Area_{inter}}$$

其中 $Area_{inter}$ 表示两个边界框重叠交集的面积， $Area_{box1}$ 表示第一个预测框的面积， $Area_{box2}$ 表示第二个预测框的面积。

(2) AP 计算公式：

$$AP = Pre_i \times \sum_{i=1}^n (Re_i - Re_{i-1})$$

其中精确率 $Pre = \frac{T}{T+F}$ ，召回率 $Re = \frac{T}{T+Fn}$

其中 T 表示预测正确的目标， F 表示错误预测的目标， Fn 表示漏检的目标，该值通常不单独计算，通过召回率的分母间接体现。

(3) mAP 计算公式：

$$mAP = \frac{1}{T} \times \sum_{t=1}^{TS} \left(\frac{1}{S} \sum_{c=1}^S AP_{t,c} \right)$$

其中： TS 表示 IoU 阈值的个数， S 表示类别总数， $AP_{t,c}$ 表示在第 t 个 IoU 阈值下，第 c 类的 AP 值

相关实现代码如下：

(1) Iou 计算代码：

```
def compute_iou(box1, box2):  
    /* 将 xywh 格式转换为 xyxy 格式 */  
    x1, y1, w1, h1 = box1  
    box1 = [x1, y1, x1 + w1, y1 + h1]  
    x2, y2, w2, h2 = box2  
    box2 = [x2, y2, x2 + w2, y2 + h2]  
    /* 计算交集区域 */  
    inter_left = max(box1[0], box2[0])  
    inter_top = max(box1[1], box2[1])  
    inter_right = min(box1[2], box2[2])  
    inter_bottom = min(box1[3], box2[3])
```

```

    if inter_right < inter_left or inter_bottom < inter_top:
        return 0.0
    inter_area = (inter_right - inter_left) * (inter_bottom - inter_top)
    area1 = (box1[2] - box1[0]) * (box1[3] - box1[1])
    area2 = (box2[2] - box2[0]) * (box2[3] - box2[1])
    /* 计算并集区域 */
    union_area = area1 + area2 - inter_area
    if union_area == 0:
        return 0.0
    return inter_area / union_area

```

(2) 使用 numpy 库计算 AP 的代码:

```

def compute_ap(gt_dict, detections, iou_threshold, category_id)
    tp = []
    fp = []
    total_gt = 0
    /* 统计所有图像中的符合教师答案数 */
    for img_id in gt_dict:
        total_gt += len(gt_dict[img_id].get(category_id, []))

    /* 以置信度降序排序结果 */
    filtered_detections = []
    for det in detections:
        if det['category_id'] == category_id:
            filtered_detections.append(det)
    sorted_detections = sorted(filtered_detections, key=lambda det: det['score'],
reverse=True)

    tp_cum = np.cumsum(tp)                /* 累计 TP */
    fp_cum = np.cumsum(fp)                /* 累计 FP */
    recall = tp_cum / (total_gt + 1e-6)    /* 计算召回率 */
    precision = tp_cum / (tp_cum + fp_cum + 1e-6) /* 计算精确率 */
    /* 计算 AP */
    ap = 0.0
    for i in range(1, len(precision)):
        ap += (recall[i] - recall[i-1]) * precision[i]
    return ap

```

(3) mAP 计算代码:

```

def get_score(teacher_answer, student_results):
    /* 构造答案标注的信息字典 */
    gt_dict = defaultdict(lambda: defaultdict(list))
    for ann in teacher_answer['annotations']:
        img_id = ann['image_id']                /* 提取图像 ID */
        cat_id = ann['category_id']              /* 提取类别 ID */

```

```

        gt_dict[img_id][cat_id].append({'bbox': ann['bbox']}) /* 存储对应的 bbox */
/* 提取答案标注中所有出现过的类别 ID，去重后构造集合 */
categories = set(ann['category_id'] for ann in teacher_answer['annotations'])

/* 设定 IoU 阈值范围：从 0.5 到 0.9，步长为 0.05 */
iou_thresholds = np.arange(0.5, 0.95, 0.05)
/* 存储每个 IoU 下的 mAP */
aps = []
/* 遍历每一个 IoU 阈值 */
for iou in iou_thresholds:
    category_aps = []
    /* 遍历所有类别并计算各类别的 AP */
    for cat_id in categories:
        /* 计算该类别在该 IoU 下的 AP */
        ap = compute_ap(gt_dict, student_results, iou, cat_id)
        category_aps.append(ap)
    /* 计算当前 IoU 阈值下的 mAP */
    mean_ap = np.mean(category_aps) if category_aps else 0
    aps.append(mean_ap)
/* 计算最终 mAP */
mAP = np.mean(aps)
return round(mAP, 4)

```

评分流程如图 3.16 所示：

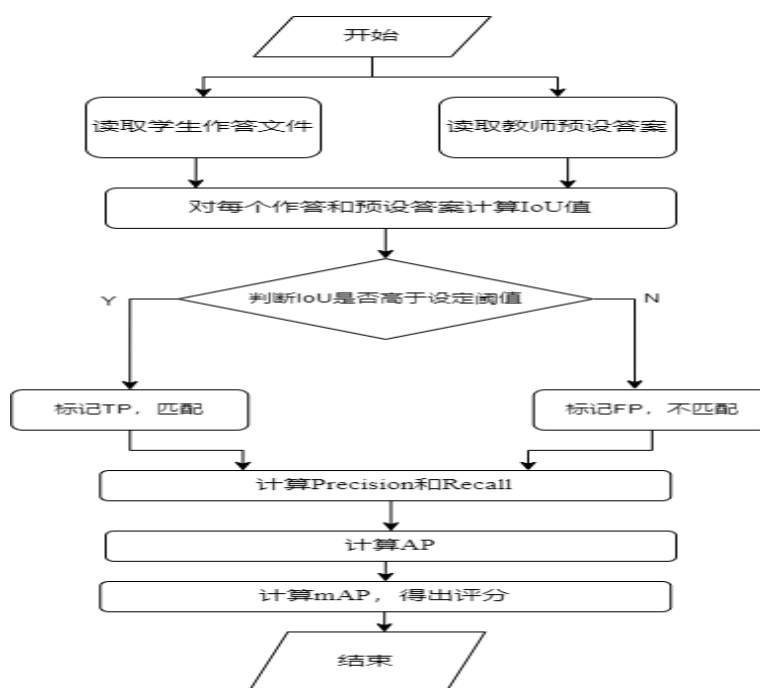


图 3.16 目标检测评分算法流程图

3.4.4 答案配置设计

针对机器学习与图像处理相关课程中关于图像分类、图像分割、目标识别三大主题自动化评分的迫切需求，结合本平台开发实际开发了针对上述主题的评分后端接口与实验平台联用，答案配置有如下要求：

- (1) 图像分类：采用 csv 格式，教师答案重命名为 `answer.csv`，包含 ID 和 CLASS 字段，配置于教师自用实验接口服务器上，并将该接口地址在实验发布时写入。
- (2) 图像分割：采用泛图像格式，支持 RGB 彩色图像，教师答案图片应配置于教师自用实验接口服务器上的 `answer` 文件夹中，并将该接口地址在实验发布时写入。
- (3) 目标识别：采用 json 数据格式，教师答案重命名为 `answer.json`，答案文件中包含一个数组，每个分类的答案使用一个 JSON 数值（一对大括号）表示，每个数值中设置 `image_id`、`category_id`、`bbox` 三个字段来表示答案（学生提交时还应增加 `score` 置信度字段）答案配置于教师自用实验接口服务器上，并将该接口地址在实验发布时写入。

3.5 创新点与技术难点

实验平台在架构设计与功能实现中体现出多方面的创新性，同时也克服了多个技术难点，主要有以下四点。

- (1) 混合式系统架构设计：项目采用混合式架构，将 B/S 结构与微服务思想结合，前端使用 Vue3 构建自适应交互界面，后端 SpringBoot + Mybatis Plus 构建高稳定服务，Flask 驱动评分服务，实现灵活部署和合理性能。
- (2) 跨平台兼容性与适配性设计：系统针对多终端环境进行了兼容性优化，支持主流浏览器及多操作系统访问，让教学不因兼容性问题停滞。
- (3) 任务流闭环：系统实现了从学生上传、教师批阅、评分反馈的全流程自动闭环，显著优化了传统教学任务中繁琐的人工操作环节，让教师专注于教学教研。
- (4) 智能化评分机制：教师自定义评分维度与权重分配后自动评分，大大提升了评分一致性和效率，特别适用于教学实验使用。
- (5) 多角色权限与数据隔离：引入分层权限控制机制，区分教师、学生与管理员的功能边界，结合数据库字段控制与接口鉴权，严格保障实验教学可信度。

第4章 主要测试

在系统开发完成后，项目组对系统进行了包括功能测试、性能测试、安全性测试和兼容性测试在内的全面测试。测试工作主要集中于以下几个方面：

- (1) 功能测试：通过单元测试与人工操作模拟，验证了学生端的作业上传、作业评分和高数据量名单导入流程。初期发现上传文件类型控制不严格和导入格式错误问题，修正后增加了前端文件格式限制机制，修正了相关设计的数据类型。
- (2) 性能测试：采用 Apache Benchmark (ab) 工具对实验平台进行了并发压力测试。测试初期在并发请求数超过 20 的场景下，系统响应时间显著上升，个别接口响应时间超过 1000 秒，影响用户体验。项目优化数据库访问层引入 SQLAlchemy + pool 连接池处理机制，显著降低了连接创建开销。
- (3) 部署分离：单一服务器部署前后端和评分服务发现在 30 并发下负载过高，极度影响性能。开发组优化为分布式部署到多台服务器上，避免单一服务器配置升级增加成本，提升了请求分发效率与系统吞吐能力。经复测，并发访问 70 的条件下，系统各主要接口响应时间均控制在 600ms 以内，稳定性和流畅性显著提升，符合设计的性能要求。
- (4) 安全性测试：通过模拟非法访问与接口注入测试，初期发现存在会话过期问题和用户越权问题。通过完善登录管理机制与用户权限校验，有效解决相关漏洞。
- (5) 兼容性测试：分别使用 14 寸、16.1 寸和 23.8 寸计算机显示器通过切换 Windows、Debian Linux 系统分别使用 Google Chrome、Firefox 访问实验平台；界面展示与功能运行与设计一致。在 Apple iPad、MacBook Air 14 上使用 Apple Safari 浏览器访问实验平台，界面展示与功能运行与设计一致。系统兼容性好。

第5章 安装与使用

5.1 安装

本实验平台设计为基于浏览器访问的应用，用户无需在客户端安装任何软件，仅要求使用 Chrome 浏览器或基于 Chromium 内核的浏览器、FireFox 浏览器、Apple Safari 浏览器访问，实验平台不支持 IE 浏览器。

服务器部署中，首先需要配置 Nginx 作为反向代理服务器，部署 Java、Python 和 Vue 静态网站环境并将生产环境运行包配置好，设置转发规则。接着，通过检查 Nginx 日志，确认运行无错误。之后，前往服务器提供商处放行对应的端口，配置 DNS 解析指向服务器公网 IP 地址，实现通过访问实验平台。为确保通信安全，开发组申请了 HTTPS 证书，并启用加密传输，保护用户数据的隐私性。

完成配置后，用户能够顺利地通过浏览器访问实验平台。

5.2 使用

用户通过浏览器访问 <https://cloud.clewrart.top> 输入账号密码即可使用。实验平台在登录后可查看使用指南或参考本文档的详细设计部分说明，此处不再赘述。

第6章 项目总结

本项目针对高校机器学习和图像处理等相关 AI 类课程实验教学中存在的评价标准不统一、教学反馈滞后以及数字化混合教学力度不足等问题，设计并开发了一套基于 Vue 和 Spring Boot 技术的实验平台。通过角色分层设计与模块化功能划分，实验平台实现了实验教学全流程的数字化管理，有效解决了传统实验教学中人工评分效率低、主观性强、反馈周期长等痛点。

实践表明，本实验平台在试运行阶段显著提升了实验教学效率：教师实验管理耗时减少，更多时间用于教学和实验指导上；学生实验报告提交与评分反馈周期大大缩短，实时排行榜功能激发了学生竞争意识，进一步促进了学习主动性。

未来系统开发将围绕以下方向展开：其一，扩展更多计算机相关实验主题，如深度学习模型的支持、实验结果的自动分析等的评分接口，增强系统普适性；其二，引入人工智能辅助教学，基于学生实验数据生成个性化学习建议；其三，结合区块链技术实现实验过程、实验材料、评分结果的存证，进一步提升评价可信度和规范性，促进实验课程混合式教学改革深化深入，提高育人质量。

总之，本项目的完成不仅为高校机器学习和图像处理等 AI 类课程的教学提供了有效工具，也为人工智能领域的教学管理创新贡献了我们的探索与实践。未来，我们将继续优化系统，提升其功能与可用性，为高校计算机教育领域的发展贡献更多力量。

最后，衷心感谢开发组成员和指导老师的支持和帮助。限于我们的水平有限，难免有缺点和错误，望各位评委专家和师生批评指正。

第7章 开发团队

7.1 学生开发团队

开发组总负责人：郭思宇，男，西南林业大学大数据与智能工程学院计算机科学与技术专业大三学生，学习研究方向为物联网、网络工程和 CV 方向；华为 HCIA 认证工程师。本项目中负责全面统筹项目开发和部署工作、项目关键技术实现与目标识别评分开发，承担文档整理写作和路演任务，推进项目高效落地。

开发组成员：闫建瑾，男，西南林业大学大数据与智能工程学院计算机科学与技术专业大三学生，学习研究方向为算法和软件工程；曾获蓝桥杯全国软件和信息技术专业人才大赛软件赛道云南省一等奖等。本项目中负责项目模块优化和测试调优工作，承担图像分割评分模块开发优化任务。

7.2 指导教师

赵毅力，博士，副教授。西南林业大学大数据与智能工程学院计算机科学与工程系教师，主要研究方向为智能遥感影像处理与分析，长期从事计算机视觉和深度学习在遥感影像中的分析和应用工作以及森林生态智慧遥感研究。主要承担本科《面向对象程序设计》、《计算机图形学》、研究生《数据科学与机器学习》等课程教学。主持国家自然科学基金项目 1 项，云南省教育厅科学研究基金项目 1 项，云南省农业基础研究联合专项面上项目 1 项，共发表学术论文 40 余篇。

参考文献

- [1] 张益琿,曹艳琴. 循序渐进 VUE.JS 3.X 前端开发实战[M].清华大学出版社. 2023.06.
- [2] 韩志豪,刘晓英. 基于 Python 的机器视觉实验教学平台设计[J].计算机测量与控制学报. 2020 ,28 (03)
- [3] 赵晓侠、潘晟旻、寇卫利. MySQL 数据库设计与应用 [M].人民邮电出版社. 2022.03
- [4] 王以伍、舒晖. 基于 SpringBoot+Vue 前后端分离的高校实验室预约管理系统的设计与实现. 现代计算机. 2023 ,29 (01)
- [5]人工智能本科专业建设报告.中华人民共和国教育部.2022