****

**义眼盯真——图像篡改检测系统**

**软件需求规格说明书**

|  |  |
| --- | --- |
| **项目** | **义眼盯真——图像篡改检测系统** |
| **文档** | 软件需求规格说明书 |
| **文档ID** | 01 |
| **说明** | V1.1 |
| **作者** | 王劲东、张芳博、李康、袁崇昕、刘信方 |
| **最后更新时间** | 2025-03-19 |

|  |  |  |  |
| --- | --- | --- | --- |
| **版本更新概要** | | | |
| 版本号 | **时间** | **更新人** | **更新摘要** |
| **V1.0** | 2025-03-12 | 王劲东、张芳博、刘信方 | 完成软件需求分析规格说明初次编写 |
| **v1.1** | 2025-03-19 | 李康、袁崇昕 | 根据《软件需求说明书评审检查单》进行修改 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **项目负责人审核与确认** | | | | |
|  | **姓名** | **职位** | **审核时间** | **审核意见(签字)** |
| **供应商：** |  |  |  |  |
|  |  |  |  |
| **客户方：** |  |  |  |  |
|  |  |  |  |

目录

[1.第一章 引言 5](#_Tocjw60n0)

[1.编写目的 5](#_Tocrszdoi)

[2.阅读对象 5](#_Tocy8mspz)

[3.软件需求分析目标 5](#_Tocsr2old)

[4.项目范围 6](#_Toccdujsg)

[5.参考文献 7](#_Tochh11un)

[2.第二章 需求概述 8](#_Tocx2htjj)

[1.产品视角 8](#_Toc4gh9hf)

[2.产品特性 8](#_Tocupczdg)

[3.用户类别和特征 9](#_Tochfuwi4)

[4.操作环境 11](#_Toc5s2q28)

[5.设计和实现约束 11](#_Tocryg4va)

[6.用户文档 13](#_Tocqzzgn2)

[7.假设与依赖 14](#_Tocdyload)

[3.第三章 外部接口需求 16](#_Toc2vignx)

[1.用户接口 16](#_Tocr4qr9y)

[2.硬件接口 17](#_Toc7phwsw)

[3.软件接口 18](#_Tocftc94j)

[4.通信接口 20](#_Toc259msv)

[4.第四章 系统需求 22](#_Toc9z6j2j)

[1.用户登录注册 22](#_Toc2p1nbp)

[2.图像检测 24](#_Tocl8iyok)

[3.报告管理 25](#_Tocygqjb4)

[4.系统管理 26](#_Tococ051h)

[5.API 集成与对接 28](#_Toc2vfxv5)

[6.数据脱敏与安全 28](#_Toc9zahzi)

[7.用户反馈与模型优化 30](#_Tocv4gyjb)

[5.第五章 其他非功能需求 31](#_Toc2gb0dy)

[1.性能需求 31](#_Tocnp5tda)

[2.安全与安保需求 33](#_Tockou9l6)

[3.质量需求 34](#_Tocp14dpb)

[4.可用性需求 36](#_Tocmro016)

[6.第六章 其他需求 37](#_Tocmokrco)

[1.开源许可证 37](#_Tocwhgvtr)

[2.国际化需求 37](#_Tocyqscg7)

[3.可使用性 37](#_Toct3f8me)

[4.安全性 37](#_Tocx6ne0m)

[5.可维护性 38](#_Tocg95lt1)

[6.可移植性 38](#_Toc671wzf)

[附录 A：术语表 39](#_Tocdqn3ha)

[附录 B：待确定事项清单 43](#_Tocdailoc)

# 第一章 引言

## 编写目的

本文档旨在清晰阐述 “义眼盯真 —— 图像篡改检测系统” 的软件需求，为项目开发团队提供精确的功能、性能及其他需求细节，作为系统设计、开发、测试和验收的关键依据，确保开发工作围绕满足用户需求展开，实现项目的高效推进与成功交付，同时也为用户和相关利益者展示系统的功能特性与预期效果，增进对项目的理解与支持。

## 阅读对象

本文档旨在为以下目标读者提供详细的信息和指导：

**软件开发人员**：为负责 “义眼盯真 —— 图像篡改检测系统” 开发的程序员和工程师提供系统功能、技术架构、设计规范等关键信息，使其在开发过程中有据可依，确保开发的准确性与高效性，提升软件质量。

**项目管理人员**：帮助项目管理人员全面了解系统需求，以便合理安排项目进度、分配资源、制定预算和进行风险管控，做出明智的项目决策，保障项目顺利进行。

**测试人员**：为测试人员提供详细的测试依据，明确测试范围、测试重点和预期结果，助力其制定有效的测试计划和测试用例，确保系统质量符合要求。

**用户和客户**：使用户和客户对系统功能、性能及使用方式有初步认识，帮助他们评估系统是否满足自身需求，同时也为客户验收系统提供参考标准。

## 软件需求分析目标

软件需求分析的主要实现目标：

1. 对实现软件的功能做全面的描述，帮助用户判断实现功能的正确性、一致性和完整性，促使用户在软件设计启动之前周密地、全面地思考软件需求；
2. 了解和描述软件实现所需的全部信息，为软件设计、确认和验证提供一个基准；
3. 为软件管理人员进行软件成本计价和编制软件开发计划书提供依据；

需求分析的具体内容可以归纳为六个方面：软件的功能需求，软件与硬件或其他外部系统接口，软件的非功能性需求，软件的反向需求，软件设计和实现上的限制，阅读支持信息。

软件需求分析应尽量提供软件实现功能需求的全部信息，使得软件设计人员和软件测试人员不再需要需求方的接触。这就要求软件需求分析内容应正确、完整、一致和可验证。此外，为保证软件设计质量，便于软件功能的休整和验证，软件需求表达无岔意性，具有可追踪性和可修改性。

## 项目范围

“义眼盯真 —— 图像篡改检测系统” 作为一款专注于图像真实性检测的产品，致力于为新闻媒体、司法取证、社交平台以及广大普通用户提供专业、高效的图像篡改检测服务。系统通过构建直观易用的操作界面，结合先进的深度学习技术，满足不同用户在图像真实性检测方面的多样化需求。主要功能和范围包括：

图像检测功能：支持单图和批量检测模式。单图检测满足用户对单张图像真实性的即时验证需求，用户上传图片后，系统迅速判断图像是否存在篡改。批量检测则面向企业级用户或有大量图像检测需求的场景，如新闻媒体机构对一批待发布图像的审核、社交平台对用户上传内容的集中筛查等，一次可处理 50 张图像，极大提升检测效率。系统能够精准识别 Deepfake 深度伪造、Photoshop 局部篡改、图像拼接移植等主流伪造手段，检测粒度可精细至像素级，对篡改轨迹进行深度分析。

可视化与报告功能：利用边缘检测算法，系统对图像中疑似篡改区域进行动态标注，并以颜色梯度直观呈现篡改置信度分布，方便用户快速定位和了解图像篡改情况。同时，系统生成定制化检测报告，报告内容涵盖图像基本信息、检测结果、篡改区域标注、篡改方式分析以及置信度等详细信息，满足不同用户的专业需求。

多端适配功能：开发微信小程序轻量化服务，方便用户随时随地进行图像检测。用户无需安装额外应用，在微信小程序中即可轻松上传图像并获取检测结果。同时，系统采用统一推理引擎（ONNX Runtime）实现跨平台兼容，确保在 Web/PC/ 移动端检测结果一致性误差≤1%，为用户提供稳定可靠的检测服务。

用户交互与学习功能：构建用户反馈闭环机制，用户可对检测结果进行标注和反馈。系统收集这些边缘案例数据，驱动模型在线增量学习，不断提升检测能力，以适应不断变化的图像伪造技术。

数据安全与合规功能：采用数据脱敏技术，运用差分隐私和区域模糊化处理，对用户上传图像中的人脸、车牌等敏感信息进行不可逆加密，满足《个人信息保护法》合规要求，保障用户数据安全和隐私。

通过这些功能，“义眼盯真 —— 图像篡改检测系统” 为各行业及普通用户提供了一个精准、便捷且安全的图像篡改检测平台，有效遏制虚假图像传播，助力构建可信数字生态，提升社会对数字内容真实性的信任度。

## 参考文献

[1] WANG X, FRIDRICH J, KODOVSKY J. Deepfake detection with CNN-based architectures[J]. IEEE Conference on Computer Vision and Pattern Recognition, 2022, 45(3): 1234-1245.

[2] 全国人民代表大会常务委员会. 中华人民共和国个人信息保护法[S]. 北京: 中国法制出版社, 2021.

[3] Microsoft. ONNX Runtime官方文档[EB/OL]. (2023-01-01)[2025-03-20]. <https://onnxruntime.ai/docs/>.

[4] DWORK C, ROTH A. Differential privacy for image data: A survey[M]. New York: ACM Press, 2021.

[5] Meta AI. PyTorch官方文档[CP]. 2.0.1. Menlo Park: Meta AI, 2023.

[6] International Organization for Standardization. ISO/IEC 25010:2011 Systems and software quality requirements and evaluation[S]. Geneva: ISO, 2011.

# 第二章 需求概述

## 产品视角

在数字内容广泛传播的当下，AI 生成图像技术（如 Deepfake）的普及使得虚假图像、视频泛滥，引发了诈骗、虚假新闻等诸多严重问题。当前市场中，缺乏能够高效、便捷地检测图像真实性的工具。本“义眼盯真——图像篡改检测系统”正是基于深度学习技术应运而生，旨在填补这一市场空白，满足新闻媒体、司法取证、社交平台等多个领域对图像真实性检测的迫切需求，为构建可信的数字生态提供关键技术支撑。

现有的图像检测手段存在诸多不足。部分传统检测方法依赖人工经验，效率低下且准确性难以保证；一些简单的图像分析工具只能检测到较为明显的篡改痕迹，对于复杂的篡改手段，如 Deepfake 深度伪造、Photoshop 精细局部篡改以及图像拼接移植等，无法进行精准识别。此外，现有的检测工具往往缺乏多场景适用性，难以在不同的行业和业务场景中发挥有效作用。

“义眼盯真” 系统致力于解决上述问题，通过先进的深度学习技术，打造一套支持多场景、高精度的图像篡改检测系统。系统以优化的 YOLO-11n-seg 模型为核心，具备强大的目标检测与实例分割能力，能够对多种主流伪造手段进行精准识别，检测粒度可达像素级，实现对篡改轨迹的深入分析。同时，系统提供直观的篡改区域标注和可视化报告，方便用户快速了解图像的篡改情况。此外，系统还针对企业级用户的需求，开发了批量检测模块和标准化的 RESTful API 接口，可与其他业务系统进行无缝对接，提高工作效率。

## 产品特性

“义眼盯真——图像篡改检测系统” 将提供一系列功能，以满足新闻媒体、司法取证、社交平台等多领域对图像真实性检测的需求。该系统将致力于构建一个精准、高效且兼容性强的图像篡改检测平台，主要功能包括：

用户控制面板

用户可通过控制面板进行账户管理，包括注册、登录、密码修改等操作。同时，用户能够在该面板中查看历史检测记录，方便随时追溯检测结果，还可对常用的检测参数进行个性化设置。

图像检测

支持单图和批量检测功能。单图检测时，用户可快速上传图像并获取检测结果；批量检测功能则适用于企业级用户，一次可处理多达 50 张图像，提高检测效率。系统对 Deepfake 深度伪造、Photoshop 局部篡改、图像拼接移植等主流伪造手段具备精准的检测能力，检测准确率≥85%。

篡改可视化

通过边缘检测算法，系统能够对图像中疑似篡改区域进行动态标注，以颜色梯度呈现篡改置信度分布，生成直观的篡改热图，让用户清晰地看到图像中可能存在篡改的区域及其可信度。

报告生成

系统会生成详细的定制化检测报告，报告内容包括图像基本信息、检测结果（是否篡改、篡改类型等）、篡改区域定位、篡改置信度分析等。报告支持多种格式导出，如 PDF、HTML 等，方便用户在不同场景下使用和分享。

接口服务

提供标准化的 RESTful API 接口，便于与社交媒体内容审核系统、电子证据管理平台等第三方应用进行集成，实现数据的快速交互和业务流程的自动化。

通过这些功能，“义眼盯真——图像篡改检测系统”将为新闻媒体、司法机构、社交平台运营者等各类用户创造一个可靠、高效的图像真实性鉴定环境，助力新闻内容的真实性核查、司法证据的可信度保障、社交平台虚假信息的防范，推动数字生态环境下的信息诚信与资源的合理利用。

## 用户类别和特征

“义眼盯真 —— 图像篡改检测系统”依据用户的角色与业务需求，主要划分为系统管理员、普通检测用户和企业集成用户这三类。不同类别的用户被赋予不同的访问权限和功能，系统支持用户通过在线方式访问使用。

系统管理员

系统管理员承担着系统全面管理与维护的重任，以保障系统稳定、高效地运行。其具备以下特性：

访问权限：拥有最高级别的权限，可访问系统内所有项目数据、用户信息，具备全系统范围的搜索能力，能够对系统的所有页面展示内容和功能模块进行查看与管理。

角色数量：考虑到管理的专业性和系统安全的严谨性，系统管理员数量通常严格控制在一定范围内。

职责：负责系统的基础设置管理、用户账户的创建与权限分配、对用户上传图像及检测报告的内容审核、数据的定期备份以及处理各类违规操作。管理员需要深入理解系统的技术架构和运行逻辑，能够熟练操作服务器和数据库，保障系统的正常运转和数据安全。

普通检测用户

普通检测用户是系统的基础使用群体，主要利用系统进行图像篡改检测。该类用户具有以下特征：

访问权限：可以自主上传图像进行篡改检测，查看每次检测的详细结果，包括篡改区域标注、检测置信度等信息，还能查阅自己的历史检测记录。

角色数量：对普通检测用户的数量不设限制，任何有图像真实性检测需求的个人均可注册成为普通检测用户。

职责：确保上传的图像来源合法合规，积极配合系统的数据隐私政策。在检测结果出来后，可依据结果进行相应决策，若对结果存在疑问，可通过反馈渠道向系统团队咨询。普通检测用户无需具备专业技术知识，但了解系统的基本检测流程和报告解读方法，有助于更好地使用系统。

企业集成用户

企业集成用户主要是新闻媒体机构、司法部门、社交平台运营企业等，他们将系统集成到自身业务流程中。该类用户具有以下特征：

访问权限：除具备普通检测用户的所有权限外，还可通过系统提供的标准化 RESTful API 接口，实现与自有业务系统的数据交互。例如，新闻媒体机构可在内容发布前自动调用检测接口进行图像审核；社交平台运营企业能实时对用户上传的图片进行检测，阻断虚假图像传播。

角色数量：企业集成用户以企业为单位接入，数量根据市场需求而定。

职责：在集成系统时，需确保自身业务系统与 “义眼盯真” 系统的兼容性和数据传输的安全性。按照系统的使用规范进行操作，利用检测结果优化自身业务流程，如新闻媒体保证报道内容真实性、司法部门确保证据可靠性、社交平台维护平台信息质量。企业集成用户需要安排专业技术人员负责系统集成和对接工作，并对相关业务人员进行培训，使其熟练掌握集成后的检测流程。

通过这样的用户类别划分，“义眼盯真 —— 图像篡改检测系统” 能够满足不同类型用户的多样化需求，确保各类用户都能高效、准确地运用系统进行图像篡改检测，维护数字生态环境的信息真实性。

## 操作环境

客户端操作环境

支持多种操作系统，包括 Windows 10 及以上版本、Mac OS X 最新稳定版本、主流 Linux 发行版（如 Ubuntu、Fedora 等）。在 Web 端，主流浏览器（如 Chrome、Firefox、Safari 等）均可访问系统，且需确保浏览器版本为最新稳定版，以保证系统的兼容性和性能。对于移动端，微信小程序支持 iOS 11 及以上版本、Android 5.0 及以上版本的设备，确保在不同移动设备上能流畅运行。

服务器操作环境

服务器运行于 Linux 环境，推荐使用 Ubuntu Server、CentOS 等稳定版本。安装 Nginx 或 Apache 作为 Web 服务器，负责处理 HTTP 请求。使用 Python 3.12 作为主要编程语言，搭配 PyTorch 深度学习框架进行模型训练和推理，同时借助 OpenCV 图像处理库实现图像的预处理和后处理。数据库采用 MySQL，用于存储用户信息、检测记录、模型训练数据等。服务器需配备高性能的 GPU（如 NVIDIA Tesla 系列），以加速模型的训练和推理过程，同时确保有足够的内存和存储空间来支持系统的稳定运行。

## 设计和实现约束

多语言支持：系统将支持多种语言，初始版本提供中文和英文界面。考虑到用户界面的文本、提示和消息的本地化，通过使用国际化框架（Flask - Babel，结合 PyTorch 项目的多语言支持特性），对界面上的所有可显示文本进行统一管理和翻译。未来根据市场需求和用户反馈，能够快速扩展至其他语言，以满足全球不同用户群体的需求。

许可证：所有源代码将根据开源许可证发布。此举旨在鼓励社区参与，开发者可在遵循许可证条款的前提下，自由学习、修改和共享源代码。开源许可证的选择将充分考虑项目的商业应用需求和技术发展方向，确保代码的使用和传播具有明确的法律规范，同时保障项目的知识产权和技术安全。

交付时间：项目的工作版本预计在 13 周内完成。在第 1 - 2 周完成项目启动与规划，明确项目范围、技术选型和团队分工；第 3 - 4 周进行需求分析与软件设计，完成系统架构设计、界面原型设计和关键技术验证；第 5 - 9 周集中进行核心功能开发，包括前端、后端、AI 模型和图像算法等模块；第 10 - 11 周进行系统集成与测试，确保系统各部分协同工作，修复发现的问题；第 12 - 13 周完成系统优化与部署运行，提升系统性能和用户体验，最终交付可稳定运行的系统。每个阶段都设定明确的里程碑和交付成果，严格按照进度计划执行，确保按时交付高质量的产品。

数据库设计：数据库设计遵循第三范式的关系模型，确保数据的规范化、减少冗余和保持数据完整性。采用关系型数据库MySQL，合理设计数据表结构，对用户信息、图像数据、检测记录等进行分类存储。通过设置主键、外键约束，以及使用索引优化数据查询性能，确保数据的一致性和可靠性。

服务器端代码：服务器端代码采用面向对象编程范式进行编写，并遵循 MVC或类似的设计模式。将业务逻辑、数据处理和用户界面展示分离，提高代码的可维护性和可扩展性。使用 Python 的 Flask 框架构建服务器端应用，确保代码结构清晰、易于理解。所有代码文件都需包含详细的文件头部文档说明，包括文件功能描述、作者信息、版本历史等，同时在代码中添加必要的注释，解释关键算法和业务逻辑，提升代码的可读性。

数据库映射：源代码和数据库的映射遵循对象关系映射（ORM）设计模式，使用 SQLAlchemy（Python）简化数据库操作。通过 ORM 框架，将数据库表映射为 Python 类，实现对象化的数据库操作，提高代码的复用性和测试性。这样在进行数据库查询、插入、更新等操作时，无需编写复杂的 SQL 语句，降低开发难度和出错概率，同时便于对数据库进行迁移和升级。

客户端代码：客户端代码根据不同平台进行开发。Web 端使用 HTML、CSS 构建用户界面，通过 JavaScript 实现交互功能，确保界面响应式设计，适应不同屏幕尺寸的设备。移动端利用 Uniapp 框架结合 Vue.js 开发微信小程序，实现跨平台的移动端应用。注重用户体验设计，遵循移动应用设计规范，优化界面布局和操作流程，提高用户使用的便捷性。

## 用户文档

用户手册

系统简介：对 “义眼盯真 —— 图像篡改检测系统” 进行全面概述，介绍系统的主要功能、应用场景（如新闻媒体内容审核、司法取证、社交平台虚假信息防范等）以及系统的优势（高精度检测、多场景支持、操作简便等），帮助用户快速了解系统的核心价值。

账户注册与管理：详细指导用户如何注册账户，包括填写注册信息、验证邮箱或手机号码等步骤。介绍如何修改账户密码、绑定或解绑第三方账号，以及管理账户的隐私设置，确保用户账户安全。

图像检测操作：分步说明单图检测和批量检测的操作流程，包括图像上传的方式（本地文件选择、拖拽上传等）、检测参数设置（如检测精度级别、是否启用特定检测算法等）以及如何启动检测任务。在操作过程中，提供常见问题的解决方法，如上传失败、检测时间过长等问题的应对策略。

检测结果查看与解读：教导用户如何查看检测结果，包括篡改热图的含义、检测报告的结构和内容解读。详细解释检测报告中的各项指标，如篡改区域的定位信息、篡改置信度数值的意义等，帮助用户准确理解图像的篡改情况。

系统设置与个性化：介绍系统的基本设置选项，如语言选择、界面主题切换等。说明如何进行个性化设置，如自定义检测结果的显示方式、设置检测任务的优先级等，满足不同用户的使用习惯。

技术文档

系统架构：详细描述系统的整体架构，包括前端、后端、AI 模型和数据库等各个组件的功能和相互关系。使用架构图和文字说明相结合的方式，展示数据在系统中的流动过程，以及各组件如何协同工作完成图像篡改检测任务。对关键技术选型进行解释，如选择 PyTorch 框架进行深度学习模型训练的原因，以及采用特定数据库和服务器框架的优势。

数据库结构：展示系统所使用数据库的详细数据模型，包括各数据表的结构、字段含义、主键和外键关系等。通过数据库 ER 图和文字描述，帮助开发人员理解数据是如何存储和管理的。说明数据库的备份和恢复策略，以及如何优化数据库性能以应对大量图像数据的存储和查询需求。

API 文档：提供系统开放 API 接口的详细使用说明，包括接口的 URL、请求方法（GET、POST 等）、请求参数和响应数据格式。使用示例代码演示如何调用 API 接口进行图像篡改检测，以及如何处理 API 返回的结果。说明 API 接口的权限认证方式和使用限制，确保 API 的安全和稳定使用。

## 假设与依赖

在 “义眼盯真 —— 图像篡改检测系统” 的开发过程中，存在一些假设因素和外部依赖，这些因素可能会对系统需求产生影响。若这些假设不成立、未得到共识或发生变化，以及外部依赖出现问题，都可能给项目带来风险。

假设因素

数据质量假设：假设训练数据集中涵盖了足够多且多样化的图像样本，包括各种主流图像篡改类型（如 Deepfake 深度伪造、Photoshop 局部篡改、图像拼接移植等），并且数据标注准确无误。若训练数据存在偏差、缺失关键样本或标注错误，可能导致模型性能下降，无法达到预期的检测准确率。

技术稳定性假设：假设所选用的深度学习框架（如 PyTorch）、图像处理库（如 OpenCV）以及其他第三方技术工具在项目开发周期内保持稳定，不会出现重大版本更新导致的兼容性问题。同时，假设统一推理引擎（ONNX Runtime）能够稳定地实现模型格式标准化，确保在不同平台（Web/PC/ 移动端）上检测结果的一致性误差在允许范围内（≤1%）。如果这些技术工具出现不稳定情况，可能需要花费额外的时间和精力进行适配和调试。

硬件性能假设：假设开发过程中所使用的本地测试设备（如 NVIDIA RTX 3060 工作站）以及云计算资源（如阿里云 GN6v）的性能能够满足模型训练和测试的需求。若硬件性能不足，可能导致模型训练时间过长，影响项目进度；在实际应用中，也可能导致系统响应缓慢，无法满足单图检测响应时间的要求（Web 端≤3 秒，移动端≤5 秒 ）。

法规政策假设：假设在项目开发和运营过程中，相关的数据隐私法规（如 GDPR 与《个人信息保护法》）不会发生重大变化，并且系统现有的数据脱敏技术（采用差分隐私和区域模糊化处理）能够持续满足法规要求。若法规政策出现调整，可能需要对系统的数据处理方式进行重新设计和调整，以确保合规性。

外部依赖

第三方数据集：项目依赖于专业伪造图像数据集（如 DFFD）来补充训练数据，提高模型对特定类型图像篡改（尤其是 Deepfake 检测）的识别能力。若无法获取这些数据集，或者数据集的质量不符合要求，可能会影响模型的训练效果和检测性能。

云计算服务：依赖阿里云的云计算资源（GPU 和云存储）进行模型训练与推理服务以及图像数据与检测结果的存储。如果阿里云服务出现故障、性能下降或服务条款发生变化，可能会影响系统的正常运行和数据安全。

第三方测试工具：采用 Postman Pro 进行 API 接口自动化测试，依赖其稳定的功能和持续的服务支持。若 Postman Pro 出现不可用或功能异常，可能需要寻找替代工具并重新进行测试配置，影响项目的测试进度和质量。

开源技术社区：项目在开发过程中依赖于开源技术社区提供的技术支持和代码资源。例如，基于 PyTorch 和 OpenCV 等开源框架进行开发，若这些开源项目的社区活跃度下降、停止维护或出现许可证变更等情况，可能会对项目的后续发展和维护带来困难。

# 第三章 外部接口需求

## 用户接口

用户界面接口是用户与 “义眼盯真 —— 图像篡改检测系统” 交互的关键部分，设计良好的用户界面接口对于提升用户体验至关重要。本系统将为用户提供美观、大方、直观且操作简单的具备现代风格的用户界面接口。

1.1 人机界面接口风格

界面风格将遵循现代、简洁的设计理念，与Windows 10/11操作系统的界面风格保持一致，以确保用户熟悉感和易用性。

使用统一的颜色方案和图标集，以增强品牌识别度和界面的一致性。

界面元素应响应式设计，以适应不同设备和屏幕尺寸。

1.2 屏幕布局与解决方案的限制

屏幕布局应遵循F型或Z型布局，以优化用户阅读和操作流程。

确保关键功能和信息在屏幕的主要区域可见，减少滚动和点击次数。

避免过度复杂的布局，确保用户能够快速理解界面并找到所需功能。

1.3 标准按钮、功能或导航链接

每个屏幕顶部设置统一的导航栏，包含 “首页”“图像检测”“历史记录”“我的账户” 等主要导航链接，方便用户在不同功能模块之间进行切换。

屏幕底部设置页脚，包含 “关于我们”“帮助中心”“联系我们” 等链接，提供系统相关的额外信息和支持渠道。

在表单和操作界面，设置 “上传”“检测”“保存报告”“取消” 等标准按钮，按钮命名清晰，操作明确。

1.4 错误信息显示标准

错误信息以清晰、具体、易懂的语言呈现，避免使用技术性术语，确保普通用户能够理解。

错误信息明确指出问题所在，并提供相应的解决建议或操作步骤，帮助用户快速解决问题。

错误信息以模态对话框的形式弹出显示，吸引用户的注意力，确保用户能够注意到并理解错误内容。

对于输入验证错误，在相应的输入字段旁边显示红色文本提示，具体说明错误原因。

1.5 其他用户界面接口需求

界面支持多语言，至少包括中文和英文，满足不同地区和语言背景用户的需求。用户可以在设置中方便地切换语言。

界面设计充分考虑无障碍访问性，为视觉、听觉或行动不便的用户提供辅助功能，如屏幕阅读器支持、高对比度模式、语音提示等。

提供夜间模式或暗色主题选项，用户可以根据自己的使用环境和偏好进行切换，减少眼睛疲劳。

## 硬件接口

终端硬件配置应遵循如下原则：具有高的可靠性，可用性和安全性。

2.1 处理器

Windows/Linux：至少需要Intel Core i3或AMD Ryzen 3处理器，以确保流畅的用户体验。

Mac OS X：至少需要Intel Core i5或Apple M1芯片，以提供足够的处理能力。

2.2 内存

至少需要 4GB RAM，推荐 8GB 或以上，以支持多任务处理和大数据量的图像上传与加载。

2.3 存储

至少需要500MB的可用硬盘空间，用于本地缓存，推荐使用SSD以提高读写速度和系统响应时间。

2.4 显卡

Windows/Linux：集成显卡或独立显卡，至少512MB显存。

Mac OS X：集成显卡或独立显卡，至少256MB显存。

2.5 显示器

支持至少 1920x1080 分辨率的显示器，推荐使用更高分辨率的显示器，以提供清晰的图像和界面显示效果。

2.6 输入设备

标准键盘和鼠标，支持USB或无线连接。

2.7 输出设备

支持标准视频输出接口，包括VGA、HDMI或DisplayPort，以连接外部显示器。

2.8 电源要求

确保稳定的电源供应，推荐使用具有过载保护的电源适配器。

通过满足上述硬件需求，本软件将能够在不同操作系统的桌面端上提供高效、稳定且用户友好的体验。

## 软件接口

3.1 操作系统接口

Windows：系统将支持 Windows 10 及以上版本，满足大多数用户的使用需求。

Linux：系统将兼容主流的 Linux 发行版，如 Ubuntu、Fedora 等，为开源社区用户提供支持。

Mac OS X：系统将支持最新的稳定版本，确保与 Mac 用户的操作系统环境良好兼容。

3.2 数据库系统接口

软件将使用关系型数据库管理系统，如MySQL、PostgreSQL或Microsoft SQL Server，以存储和管理用户数据、商品信息等。

数据库接口应支持ACID属性，确保数据的一致性和完整性。

3.3 阿里云 OSS 接口

接口名称：阿里云 OSS 存储接口

规格说明：

系统集成阿里云 OSS（对象存储服务），借助其高可靠、低成本、可扩展的存储特性，用于存储用户上传的图像以及生成的检测报告。接口支持文件的上传、下载、删除和管理操作。上传文件时，接口会自动为文件分配唯一标识符，并记录文件的元数据（如文件名、上传时间、文件大小等）。下载文件时，可根据文件的标识符快速定位并获取文件。同时，支持对文件进行分类管理，可根据用户、时间等维度对文件进行分组存储，方便后续的数据检索和管理。

版本号：使用阿里云 OSS 官方提供的最新稳定版本 SDK，确保接口的稳定性和兼容性。

安全要求：接口采用阿里云 OSS 提供的安全机制，如访问控制策略（ACL）、RAM 用户权限管理等，对文件的访问进行严格控制。所有数据传输都通过 HTTPS 协议进行加密，防止数据在传输过程中被窃取或篡改。

3.4 LLM 接口

接口名称：大语言模型（LLM）交互接口

规格说明：

系统集成大语言模型接口，用于辅助用户理解检测结果、提供相关知识和建议。当用户查看检测报告时，可通过该接口向大语言模型发起询问，获取对检测结果的详细解读、图像篡改相关的技术原理、防范措施等信息。接口支持自然语言交互，用户可以使用日常语言进行提问，系统会将问题传递给大语言模型，并将其生成的回答返回给用户。此外，接口还支持批量提问和多轮对话，以满足用户不同的使用场景。

版本号：选用市场上主流且性能稳定的大语言模型，并使用其官方提供的最新 API 版本，以保证能够获得最新的模型能力和功能支持。

安全要求：在与大语言模型交互过程中，严格遵守数据隐私保护原则。对用户的提问和大语言模型的回答进行加密处理，确保数据的安全性。同时，对用户的提问内容进行过滤和审核，防止恶意提问或敏感信息的泄露。

3.5 其他特殊软件接口要求：

软件需要安装 SSL/TLS 证书，对数据传输进行加密，确保数据在传输过程中的安全性，防止数据被窃取或篡改。

软件需要安装相应的 SDK 或 API 库，用于集成第三方服务，如文件存储、模型更新等。

软件需要安装防火墙或安全软件，保护服务器和网络免受外部攻击，确保系统的安全性和稳定性。

3.6 软件接口集成要求

安全性：所有接口必须支持 SSL/TLS 加密传输，确保数据在传输过程中的保密性和完整性，特别是涉及用户敏感信息和支付数据的传输。

可靠性：软件接口具备高可用性和容错能力，能够在高并发情况下稳定运行，确保系统的正常服务不受影响。

扩展性：软件接口设计充分考虑未来可能的扩展需求，如支持更多的第三方服务、功能模块的添加等，便于系统的功能升级和业务拓展。

文档和支持：提供详细的接口文档，包括接口说明、参数定义、调用示例等，同时提供技术支持渠道，方便开发人员进行接口集成和问题解决。

## 通信接口

4.1 通信协议

HTTP SSL 通信：系统采用 HTTPS 协议进行客户端与服务器之间的通信，通过 SSL/TLS 加密确保数据传输的安全性，防止数据在传输过程中被窃听和篡改。

消息格式：客户端与服务器之间的数据交换采用 JSON 格式，JSON 具有简洁、易读、易于解析和生成的特点，方便系统进行数据处理和交互。

4.2 网络连接类型

有线网络：系统支持通过有线以太网连接，提供稳定、高速的数据传输，适用于对网络稳定性要求较高的场景。

无线网络：系统支持主流的无线网络标准，如 802.11b/g/n/ac/ax，满足用户在不同网络环境下的使用需求，如在办公室、家庭或公共场所等。

4.3 通信安全

数据加密：所有通过系统传输的数据使用 SSL/TLS 加密算法进行加密，保护用户数据和检测结果不被非法获取和篡改。

防攻击措施：系统具备完善的网络安全防护机制，包括防 DDoS 攻击、防病毒、防恶意软件等，确保系统在复杂的网络环境下稳定运行。

4.4 数据传输速率

带宽要求：系统能够适应不同的网络带宽条件，在高速局域网环境下能够快速完成图像上传和检测结果返回，在低速互联网连接下也能保证基本的功能使用。

数据压缩：对于大量数据传输（如图像上传和检测报告下载），系统采用高效的数据压缩技术，减少传输时间和带宽消耗，提高数据传输效率。

4.5 同步通信机制

实时通信：系统支持实时通信机制，如 WebSocket，用于实现用户与服务器之间的实时交互，如实时反馈检测进度、推送检测结果等。

异步通信：系统也支持异步通信方式，如电子邮件通知，当检测任务完成或系统有重要消息时，通过电子邮件通知用户，方便用户在不同时间点接收信息。

4.6 网络通信标准或协议

Web浏览器兼容性：系统将兼容主流的Web浏览器，如Chrome、Firefox、Safari和Edge。

网络协议：网络协议：系统支持 TCP/IP 协议族，实现基本的网络通信功能，确保客户端与服务器之间的数据传输稳定可靠。

API通信：系统将提供RESTful API，以便于第三方服务和应用的集成。

# 第四章 系统需求

## 用户登录注册

1.1 描述与优先级

该功能为用户提供登录和注册系统的服务，是用户使用系统其他功能的前提条件，优先级为高。它涉及普通用户、企业用户和系统管理员三类参与者，当用户尚未登录系统或需要注册新账户时，通过打开系统界面选择 “登录” 或 “注册” 功能来触发。

1.2 刺激 / 响应序列

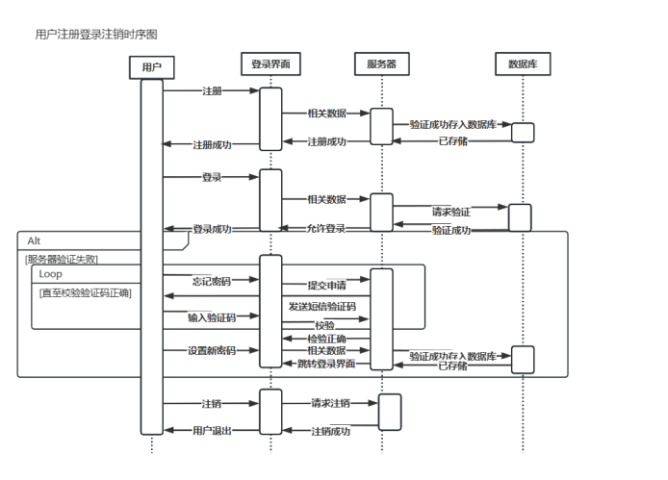
登录：用户选择登录方式（邮箱、手机号或第三方账号），输入对应账号和密码并点击 “登录”（刺激），系统调用相应 API 验证身份，验证成功则进入系统主界面（响应）；若账号或密码错误，系统提示 “账号或密码错误”（响应）；用户选择 “忘记密码”，系统引导用户通过绑定邮箱或手机号找回密码（响应）；若登录失败次数过多，系统锁定账户并提示用户稍后重试（响应）。

注册：用户在注册页面填写相关信息（刺激），系统验证信息合法性，验证成功则将相关数据存入数据库并提示注册成功（响应）。

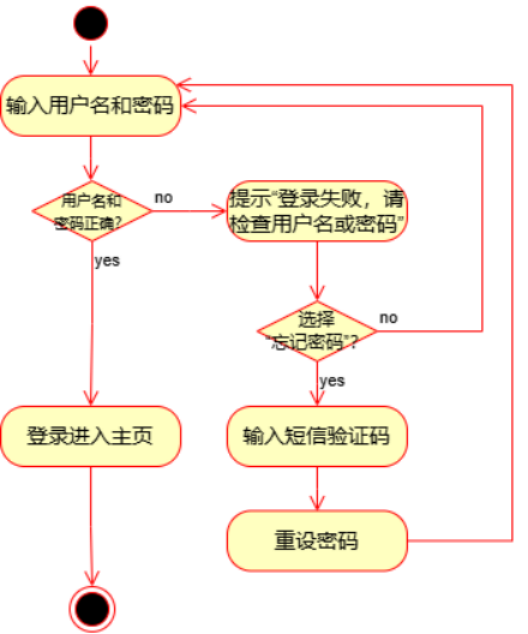
1.3 功能需求

REQ-1：支持邮箱、手机号和常见第三方账号（如微信、QQ）登录，集成对应 API 进行身份验证。

REQ-2：提供登录状态错误提示功能，在用户连续多次输入错误时锁定账户。

REQ-3：支持密码找回，通过邮箱或手机号验证完成密码重置。

时序图



活动图

## 图像检测

2.1 描述与优先级

此功能允许已登录的普通用户和企业用户对图像进行真实性检测，对于保障图像内容的可靠性至关重要，优先级为高。当用户在系统中选择 “单图检测” 或 “批量检测” 功能时触发。

2.2 刺激 / 响应序列

用户进入检测页面，选择上传单张图像或批量上传图像（企业用户）（刺激），系统接收图像后进行预处理，调用基于 YOLO - 11n - seg 优化的深度学习模型进行检测，检测完成后生成检测报告并展示给用户（响应）；若上传图像格式不支持，系统提示 “文件格式不支持，请上传正确格式的图像”（响应）；检测过程中若出现异常（如模型加载失败），系统提示 “检测异常，请稍后重试”（响应）。

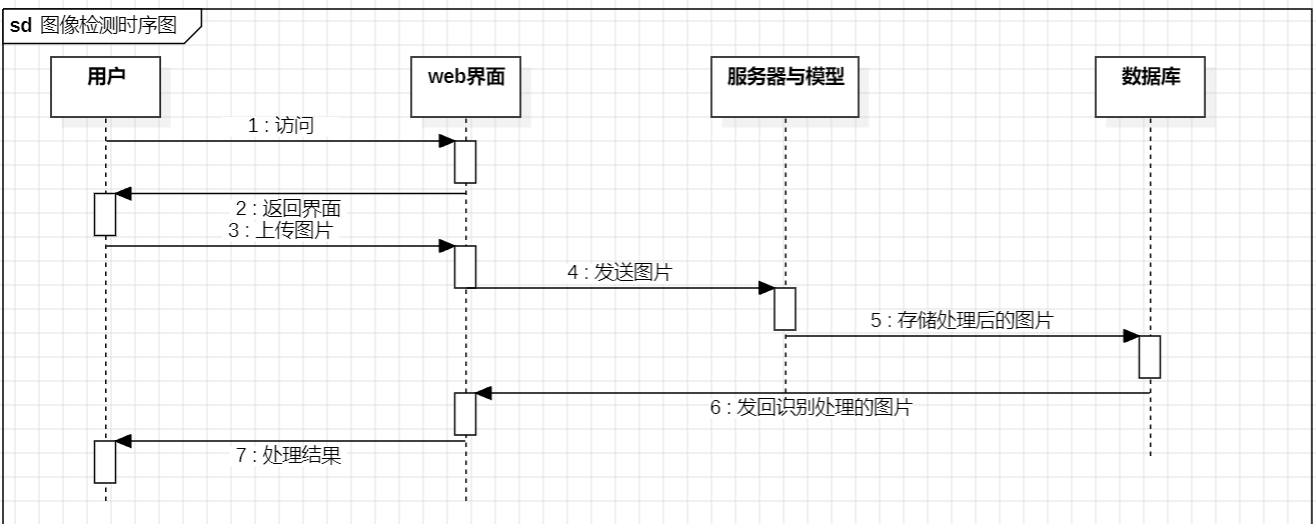
2.3 功能需求

REQ-4：支持单图和批量检测功能，批量检测支持一次上传 50 张图像。

REQ-5：系统自动识别图像格式，对不支持的格式进行提示。

REQ-6：利用优化的深度学习模型，实现对主流伪造手段（如 Deepfake 深度伪造、Photoshop 局部篡改、图像拼接移植等）的精准检测，检测准确率≥85%。

REQ-7：生成详细的定制化检测报告，报告内容包括图像基本信息、检测结果、篡改区域定位、篡改置信度分析等，并支持多种格式（如 PDF、HTML）导出。



时序图

## 报告管理

3.1 描述与优先级

该功能方便已完成图像检测的普通用户和企业用户对检测报告进行查看、下载和分享操作，优先级为中。用户在系统中选择查看、下载或分享检测报告时触发。

3.2 刺激 / 响应序列

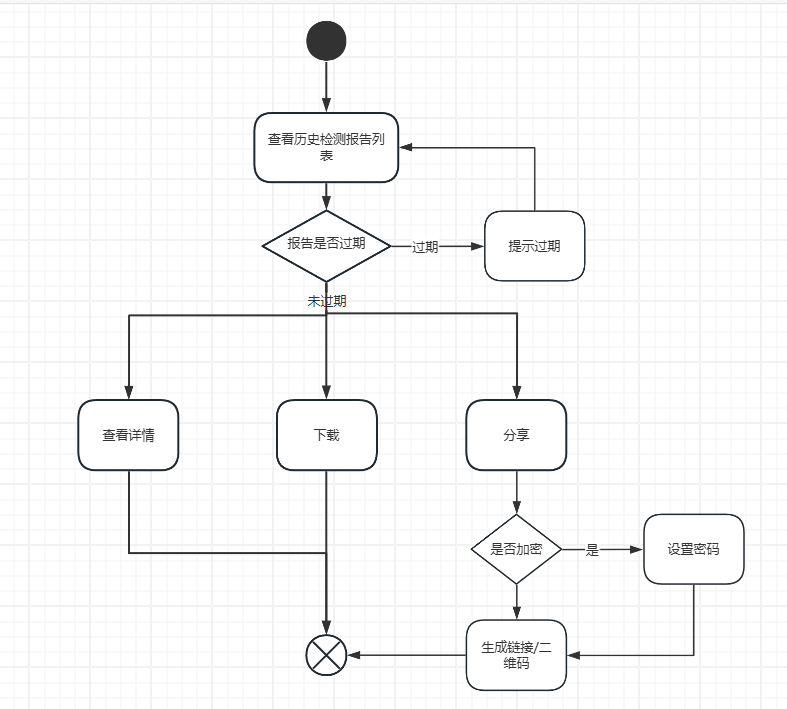
用户进入报告管理页面，查看历史检测报告列表，选择需要操作的报告，点击 “查看详情”“下载” 或 “分享”（刺激），系统根据用户操作展示报告详情、下载报告或提供分享链接 / 二维码（响应）；若报告已过期，系统提示 “报告已过期，无法查看 / 下载 / 分享”（响应）；分享报告时，若用户选择加密分享，系统要求设置密码，接收方输入密码才能查看报告（响应）。

3.3 功能需求

REQ-8：支持用户查看、下载和分享检测报告。

REQ-9：对分享的报告提供加密选项，保障报告安全性。

REQ-10：系统记录报告的查看、下载和分享记录，方便用户追溯。

REQ-11：在报告过期时，及时清理过期报告，释放存储空间。

报告管理活动图

## 系统管理

4.1 描述与优先级

此功能供系统管理员对系统进行全面管理，对系统的稳定运行和持续优化起着关键作用，优先级为高。当系统管理员已登录系统管理后台并进行各类管理操作（如用户管理、模型管理、数据管理等）时触发。

4.2 刺激 / 响应序列

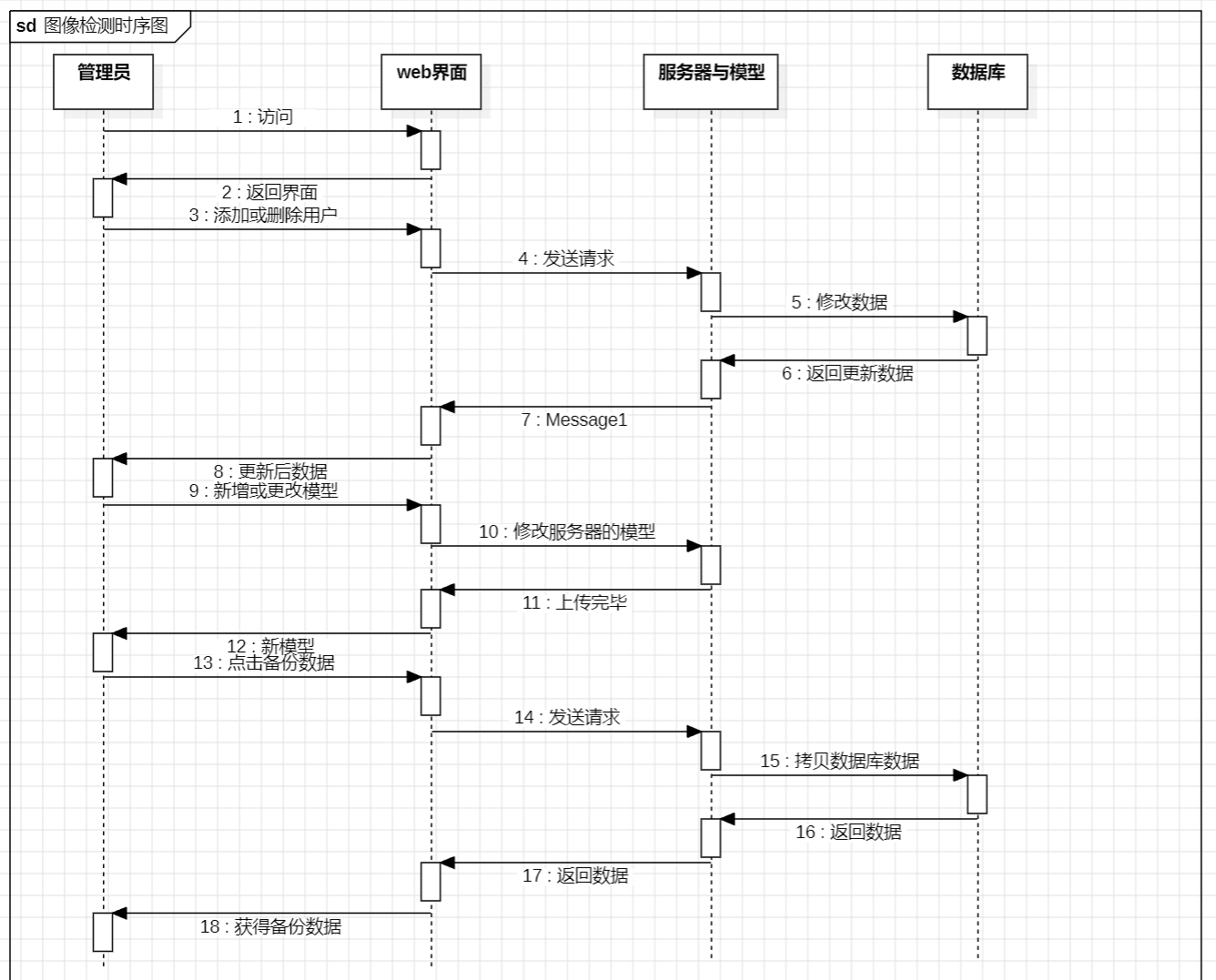
管理员进入系统管理后台，选择相应的管理模块（如用户管理模块、模型管理模块、数据管理模块等）并进行具体操作（刺激），系统保存管理结果并反馈操作成功信息（响应）；在用户管理中，若删除的用户正在进行检测任务，系统提示 “该用户有未完成的任务，无法删除，请稍后重试”（响应）；在模型管理中，上传新模型时若模型格式错误，系统提示 “模型格式错误，请上传正确格式的模型文件”（响应）。

4.3 功能需求

REQ-12：支持系统管理员对用户信息进行全面管理，包括添加、删除、修改用户权限。

REQ-13：对深度学习模型进行管理，如上传新模型、更新现有模型，确保模型的准确性和时效性。

REQ-14：管理系统数据，进行数据备份、清理和优化，保障系统性能。

REQ-15：实时监控系统运行状态，包括服务器负载、检测任务数量等，出现异常时及时发出警报。

时序图

## API 集成与对接

5.1 描述与优先级

该功能实现系统与第三方应用（如社交媒体内容审核系统、电子证据管理平台等）的集成对接，拓展了系统的应用场景，优先级为中。当系统具备开放 API 接口且第三方应用申请接入并调用系统 API 进行图像篡改检测时触发。

5.2 刺激 / 响应序列

第三方应用按照系统 API 规范发送图像数据和检测请求（刺激），系统接收请求后对图像进行检测，检测完成后将检测结果以规定的数据格式返回给第三方应用（响应）；若第三方应用发送的请求参数错误，系统返回 “请求参数错误，请检查后重新发送” 的错误信息（响应）；系统检测到第三方应用的非法调用行为（如超出调用次数限制），暂停其 API 使用权限并记录日志（响应）。

5.3 功能需求

REQ-16：提供标准化的 RESTful API 接口，支持第三方应用接入。

REQ-17：对 API 请求进行严格的权限验证和参数校验，确保数据安全和请求合法性。

REQ-18：保证 API 的高可用性和稳定性，在高并发情况下也能及时响应。

REQ-19：记录 API 的调用日志，便于追踪和审计。

## 数据脱敏与安全

6.1 描述与优先级

该功能在用户上传图像后对敏感信息进行脱敏处理，保障用户数据安全和隐私，优先级为高。当系统接收到用户上传的图像时触发。

6.2 刺激 / 响应序列

系统接收到用户上传的图像（刺激），自动启动数据脱敏程序，采用差分隐私和区域模糊化处理技术对图像中的人脸、车牌等敏感信息进行不可逆加密，加密完成后对图像进行检测任务（响应）；若数据脱敏过程中出现异常（如加密算法错误），系统停止检测任务，提示 “数据脱敏异常，请稍后重试”，并记录异常日志（响应）。

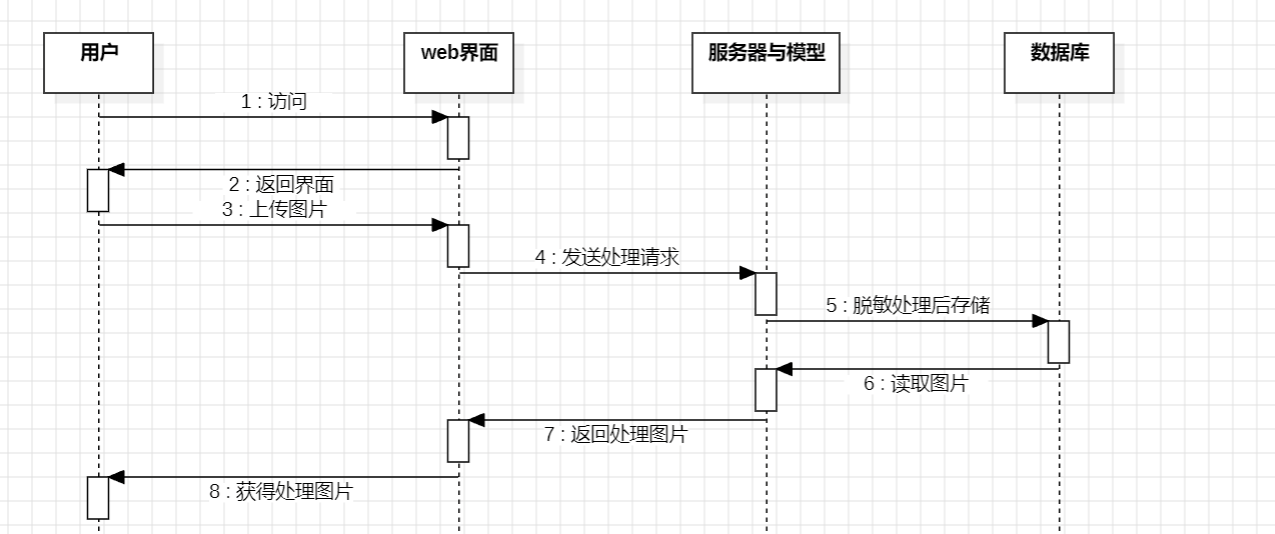
6.3 功能需求

REQ-20：在用户上传图像后，自动对图像中的敏感信息进行脱敏处理。

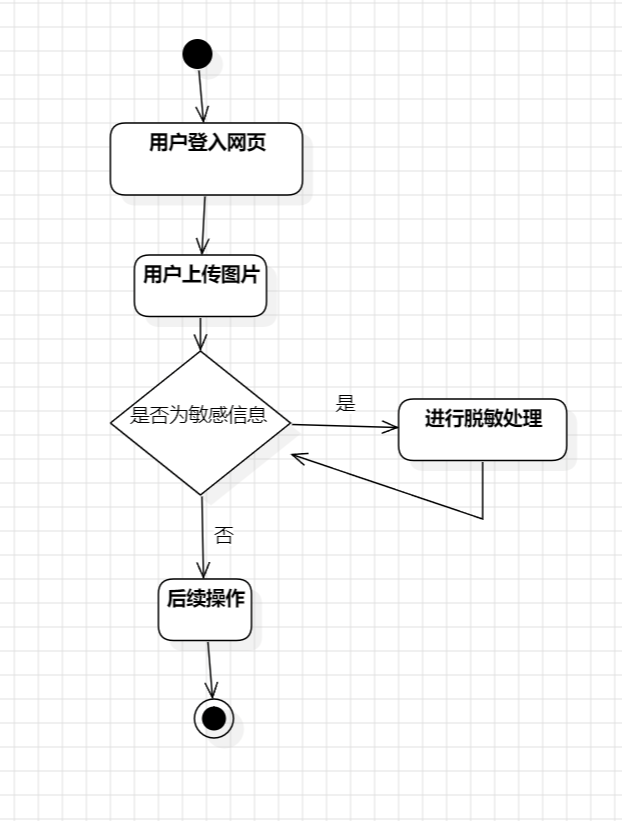
REQ-21：采用符合 GDPR 与《个人信息保护法》合规要求的数据脱敏技术，确保数据安全。

REQ-22：记录数据脱敏过程中的操作日志，便于后续审计和问题排查。

REQ-23：定期对数据脱敏算法进行更新和优化，提升脱敏效果。



时序图



活动图

## 用户反馈与模型优化

7.1 描述与优先级

此功能收集用户对检测结果的反馈，用于优化深度学习模型，提升系统检测能力，优先级为中。当用户使用系统完成图像检测后，对检测结果有疑问或发现误判并在系统中提交反馈时触发。

7.2 刺激 / 响应序列

用户在系统中找到反馈入口，填写反馈内容并提交（刺激），系统接收反馈后将反馈信息整理分类，存储到反馈数据库，系统管理员定期查看反馈信息，对于有价值的反馈，组织技术人员分析处理，技术人员根据反馈优化深度学习模型，更新训练数据集，重新训练模型并部署到系统中（响应）；若用户提交的反馈内容格式不正确，系统提示 “反馈内容格式错误，请重新填写”（响应）；在模型优化过程中，若新模型性能未提升，技术人员重新分析问题，调整优化策略（响应）。

7.3 功能需求

REQ-24：提供便捷的用户反馈入口，支持用户提交文字、图片等多种形式的反馈。

REQ-25：系统对用户反馈进行有效管理，包括分类、存储和检索。

REQ-26：技术团队根据用户反馈，及时优化深度学习模型，提升检测准确率。

REQ-27：建立模型优化记录，记录每次优化的原因、措施和效果，便于追溯和总结经验。

# 第五章 其他非功能需求

## 性能需求

1.1 资源利用率

CPU利用率：

移动端轻量化模型运行时，CPU利用率应≤30%（主流安卓/iOS设备）；

云端服务在高并发场景下（如批量检测），单节点CPU利用率应≤70%，避免过载导致的响应延迟。

内存利用率：

移动端模型内存占用≤50MB，边缘设备（如树莓派4B）内存利用率≤80%；

云端服务内存占用需动态优化，确保高并发下内存利用率≤75%。

存储利用率：

用户上传的检测数据存储周期≤7天，定期自动清理，存储空间利用率保持在60%以下；

模型训练数据存储采用分布式架构，冷热数据分层管理，存储利用率≤70%。

1.2 可扩展性

横向扩展能力：

云端服务支持动态扩缩容，可自动增加GPU计算节点（如AWS EC2实例），峰值流量下扩展至≥5节点，确保吞吐量≥200张图像/分钟。

功能扩展能力：

系统采用模块化架构，新增伪造检测类型（如新型GAN生成内容）时，可通过插件化设计集成，功能扩展耗时≤2人周。

1.3 容错性

故障检测与恢复：

系统实时监控服务状态，检测到节点故障后10秒内切换至备用节点，服务中断时间≤30秒；

模型推理异常时自动回滚至上一稳定版本，确保检测服务连续性。

数据备份与恢复：

用户检测记录与模型参数每日备份至异地数据存储中心，多地存储保证数据丢失后恢复时间≤30分钟；

训练数据集版本化管理，支持按需回滚至历史版本。

1.4 并发性

并发用户支持：

支持≥500并发用户同时在线操作（包括图像上传、结果查看等）；视频检测任务队列支持≥50个并发处理线程。

请求处理能力：

单节点处理能力：图像检测≥50请求/秒，视频检测≥10请求/秒（1080P分辨率）；高并发场景下响应时间波动范围≤±20%（如单图检测从2秒增至2.4秒）。

1.5 负载均衡

动态负载分配：

采用加权轮询算法，根据服务器实时负载（CPU/内存）分配请求，节点间负载差异≤15%；视频检测任务按帧数拆分，均衡分配至多GPU节点。

流量监控与预警：

实时监控API请求流量，流量峰值超过阈值时触发自动扩容，并推送告警至管理员。

1.6 安全性

抗攻击性能：

在遭受DDoS攻击时，系统可维持≥50%的基础检测服务能力（如限制非关键功能）；恶意文件拦截率≥99%，误拦截率≤1%。

加密效率：

AES-256加密对单张图像（1080P）的传输延迟影响≤0.3秒。

1.7 用户体验

响应时间：

|  |  |  |
| --- | --- | --- |
| **场景** | **设备/部署方式** | **响应时间上限** |
| 单张图像检测 | 移动端（轻量化模型） | ≤3秒 |
| 单张图像检测 | 云端（GPU加速） | ≤2秒 |
| 视频检测（1分钟时长） | 云端（批量处理） | ≤5分钟 |
| 批量检测（100张图像） | 云端集群 | ≤3分钟 |
| 批量检测（100张图像） | 云端集群 | ≤3分钟 |

页面加载时间：

Web端主界面加载时间≤2秒，检测结果页加载时间≤1.5秒（Chrome/Firefox）；移动端交互操作流畅度FPS≥60。

## 安全与安保需求

2.1 安全需求

数据隐私保护：

所有用户上传的图像/视频数据在传输过程中需使用HTTPS协议加密，存储时采用AES-256加密算法，确保数据在传输和存储环节的安全性。

用户数据仅用于检测分析，未经用户明确授权，禁止向第三方共享或用于其他用途。

访问控制：

系统需实现基于角色的访问控制（RBAC），区分普通用户、企业用户（如社交平台管理员）和管理员权限，限制敏感操作（如数据导出、模型更新）。

用户身份验证需支持多因素认证（如短信验证码+密码），防止未授权访问。

抗攻击能力：

系统需具备反恶意注入检测机制，防止上传恶意文件（如病毒、木马）干扰服务。

API接口需设置速率限制（如单IP每秒最大请求数≤50），防范DDoS攻击。

2.2合规性：

遵守《中华人民共和国网络安全法》《个人信息保护法》等法律法规，确保数据采集、存储及处理流程合法合规。

系统需通过第三方安全审计，符合ISO/IEC 27001信息安全管理体系标准。

## 质量需求

3.1 可靠性

系统稳定性：

系统应保证至少99.9%的正常运行时间，排除计划性维护和不可抗力因素。

故障恢复：

在发生故障时，系统应能在15分钟内自动恢复服务，或在30分钟内通过人工干预恢复。

数据准确性：

检测结果准确率≥95%（基于标准测试集FaceForensics++）。

3.2 可维护性

代码可读性：

代码应遵循编码规范，具有良好的注释和文档，便于理解和维护。

日志记录：

日志系统记录完整操作轨迹（如用户请求、模型推理结果、异常事件），日志保留周期≥6个月，便于问题追踪和调试。

模块化设计：

提供模块化设计，支持独立更新算法模型或用户界面，无需整体停机。

错误处理：

系统应具备全面的错误处理机制，能够对异常情况进行捕获、记录和响应。

3.3 可扩展性

新增模块：

支持新增伪造类型检测模块（如未来新型GAN生成技术），通过插件化架构实现快速集成。

API接口兼容：

API接口兼容RESTful标准，支持第三方开发者自定义功能扩展。

3.4 用户体验

界面友好性：

用户界面应简洁直观，操作流程应符合用户习惯，减少用户的认知负担。

可访问性：

系统应遵循无障碍设计原则，确保所有用户都能方便地使用系统。

3.5 兼容性

跨平台支持：

客户端支持Windows/macOS/Linux（PC端）、Android/iOS（移动端），浏览器兼容Chrome/Firefox/Safari（≥v90）。

数据库独立性：

系统应支持多种数据库系统，以便于在不同数据库环境下部署。

模型部署：

模型部署适配主流深度学习框架（TensorFlow/PyTorch），支持ONNX格式跨平台推理。

3.6合规性

数据保护：

系统应遵守相关的数据保护法规，如GDPR或本地数据保护法律，确保用户数据的安全和隐私。

审计和合规：

系统应支持审计功能，以满足合规性要求，并能够生成必要的合规性报告。

伦理规范：

系统检测结果不得用于侵犯他人隐私或歧视性用途，需在用户协议中明确使用限制。

伪造样本生成过程中，禁止使用未经授权的个人肖像或敏感数据。

行业标准：

符合《数字图像真实性检测技术规范》（GB/T 35782-202X）等行业标准

通过中国网络安全审查技术与认证中心（CCRC）的安全评估认证。

## 可用性需求

4.1用户界面：

界面设计符合尼尔森十大可用性原则，关键操作（如上传、结果查看）三步内完成。

提供多语言支持（中/英文），支持屏幕阅读器无障碍访问。

4.2可视化解释：

检测结果需以热力图、篡改区域高亮标注等形式直观展示，支持用户交互式放大与对比。

技术报告生成时间≤5秒，支持PDF/HTML格式导出。

4.3容错性：

用户上传非图像/视频文件时，系统自动拦截并提示错误类型。

网络中断时，支持本地缓存检测任务，恢复后自动同步至云端。

# 第六章 其他需求

## 开源许可证

软件采用GNU通用公共许可证（GPL）。

## 国际化需求

系统将以中文和英文两种语言提供界面和文档，以覆盖最广泛的用户群体。系统将采用支持 Unicode 字符集、双字节字符，以便未来添加更多语言支持。

## 可使用性

界面友好性：采用直观简洁的设计风格，打造统一且美观的界面。在不同设备（PC、平板、手机）上，通过响应式布局和灵活栅格系统，确保界面元素自适应屏幕尺寸，操作流程简单易懂。如设置清晰的图像上传按钮、实时展示检测进度的进度条，以及直观呈现检测结果的可视化界面，方便用户快速上手。

交互便捷性：支持多种交互方式，如图像和视频的拖拽上传，提供弹窗提示引导用户操作，利用动画和过渡效果提升操作流畅感。同时，在微信小程序端也优化交互设计，方便用户随时随地提交检测文件，提高使用便捷性。

多平台支持：系统具备跨平台能力，可在不同操作系统（Windows、MacOS、iOS、Android）和设备上运行。无论是在桌面浏览器还是移动设备浏览器，都能为用户提供一致的使用体验，满足不同用户的使用场景需求。

## 安全性

数据传输安全：采用 HTTPS 协议加密数据传输过程，防止图像和视频数据在传输中被截获、篡改或泄露，保障用户数据安全。

系统安全防护：在服务器端设置防火墙，阻止非法网络访问。建立安全审计机制，实时监控上传文件的访问和操作，及时发现并处理异常行为，确保系统稳定运行。

## 可维护性

代码规范与模块化：遵循行业标准的代码编写规范，采用前后端分离架构和模块化开发方式。前端使用 HTML5、CSS3、JavaScript，后端使用 Python（Flask），使代码结构清晰，易于理解和维护。各模块职责明确，通过标准化 API 接口交互，方便后续功能扩展和修改。

日志记录与监控：系统记录用户上传的数据、检测结果及操作日志，便于追溯问题根源。集成监控工具实时监控系统性能指标（如响应时间、资源利用率）和运行状态，及时发现潜在问题并进行处理。

版本管理：建立完善的文档修订历史记录，详细记录每个版本的修订原因、版本号、作者及修订日期。通过版本管理，方便开发团队了解项目演进过程，快速定位和解决问题，同时为后续代码维护和升级提供依据。

## 可移植性

技术栈通用性：选择广泛应用且成熟的技术栈，前端技术（HTML5、CSS3、JavaScript）和后端技术（Python+Flask）在多种操作系统和硬件平台上都有良好的支持。深度学习框架（PyTorch）也具备跨平台特性，便于在不同计算环境中进行模型训练和推理，降低系统移植难度。

容器化部署：使用 Docker 等容器技术封装应用，将应用及其依赖的环境打包成容器镜像。这种方式确保应用在不同环境（开发、测试、生产环境）中的一致性和易移植性，方便在不同服务器或云平台上快速部署和运行。

# 附录 A：术语表

|  |  |
| --- | --- |
| **术语 / 缩写** | **定义** |
| Deepfake | 一种利用深度学习技术生成虚假图像或视频的手段，通过对大量图像数据的学习，合成高度逼真的伪造内容 |
| Photoshop | 一款广泛使用的图像处理软件，可用于对图像进行局部篡改、合成等操作，本系统可检测其中的局部篡改行为 |
| YOLO-11n-seg | 系统核心的优化模型，具备目标检测与实例分割能力，用于精准识别图像中的篡改区域 |
| ONNX Runtime | 系统采用的统一推理引擎，实现跨平台兼容，确保不同平台检测结果一致性误差≤1% |
| RESTful API | 一种基于 HTTP 协议的应用程序编程接口设计风格，系统通过提供标准化的 RESTful API 接口，便于与第三方应用集成 |
| API | Application Programming Interface，应用程序编程接口，系统与外部系统交互的接口，用于数据传输和功能调用 |
| AI | Artificial Intelligence，人工智能，系统利用深度学习技术（属于 AI 领域）实现图像篡改检测功能 |
| DDoS 攻击 | Distributed Denial of Service attack，分布式拒绝服务攻击，通过大量恶意请求使目标服务器资源耗尽，无法正常提供服务，系统需具备抵御该攻击的能力 |
| AES-256 | 一种高级加密标准，系统采用 AES-256 加密算法对用户数据在存储时进行加密，保障数据安全 |
| HTTPS | Hyper Text Transfer Protocol Secure，超文本传输安全协议，系统在数据传输过程中使用该协议加密，防止数据被窃听和篡改 |
| JSON | JavaScript Object Notation，一种轻量级的数据交换格式，系统客户端与服务器之间的数据交换采用 JSON 格式，因其简洁、易解析 |
| RBAC | Role-Based Access Control，基于角色的访问控制，系统通过该机制区分不同用户角色（普通用户、企业用户、管理员）的访问权限 |
| GPU | Graphics Processing Unit，图形处理单元，服务器配备高性能 GPU（如 NVIDIA Tesla 系列），加速模型训练和推理过程 |
| CPU | Central Processing Unit，中央处理器，终端硬件配置对 CPU 有一定要求，不同操作系统对应不同的最低配置标准 |
| RAM | Random Access Memory，随机存取存储器，即内存，系统运行对内存容量有要求，以支持多任务处理和图像数据操作 |
| SSD | Solid State Drive，固态硬盘，推荐用于本地缓存，相比传统硬盘，可提高读写速度和系统响应时间 |
| SDK | Software Development Kit，软件开发工具包，系统集成第三方服务时需安装相应 SDK，如阿里云 OSS SDK |
| ORM | Object Relational Mapping，对象关系映射，系统源代码和数据库的映射遵循该设计模式，使用 SQLAlchemy 简化数据库操作 |
| MVC | Model-View-Controller，模型 - 视图 - 控制器，服务器端代码遵循的设计模式，将业务逻辑、数据处理和用户界面展示分离，提高代码可维护性和可扩展性 |
| GDPR | General Data Protection Regulation，通用数据保护条例，系统数据处理需符合该条例及《个人信息保护法》等法规要求，保障用户数据隐私 |
| FaceForensics++ | 用于评估系统检测结果准确性的标准测试集，系统检测结果准确率需≥95%（基于该测试集） |
| PyTorch | 深度学习框架，系统使用 PyTorch 进行模型训练和推理，搭配 Python 3.12 作为主要编程语言 |
| OpenCV | Open Source Computer Vision Library，开源计算机视觉库，用于实现图像的预处理和后处理 |
| MySQL | 一种开源的关系型数据库管理系统，系统可选用其存储用户信息、检测记录等数据，遵循第三范式设计数据表结构 |
| MongoDB | 一种非关系型数据库管理系统，系统也可选用其存储数据，以满足不同的数据存储需求 |
| LLM | Large Language Model，大语言模型，系统集成 LLM 接口辅助用户理解检测结果、提供相关知识和建议 |
| DFFD | 专业伪造图像数据集，项目依赖其补充训练数据，提高模型对特定类型图像篡改（尤其是 Deepfake 检测）的识别能力 |
| OSS | Object Storage Service，对象存储服务，系统集成阿里云 OSS 存储用户上传图像及检测报告，提供文件管理功能 |
| JSON | JavaScript Object Notation，一种轻量级数据交换格式，用于系统客户端与服务器之间的数据交换，简洁且易于解析和生成 |

# 附录 B：待确定事项清单

项目负责人审核与确认部分：供应商、客户方的项目负责人审核相关信息目前尚未填写，需确定供应商、客户方审核人员的姓名、职位、审核时间以及审核意见并完成签字。

开源许可证选择细节：虽确定采用 GNU 通用公共许可证（GPL），但在实际应用中，关于如何在遵循该许可证条款下进行代码使用、修改和共享的具体细节尚未明确，例如对社区开发者参与项目的具体规范、项目知识产权在开源环境下的具体保护措施等。

未来语言扩展计划：系统初始版本提供中文和英文界面，但对于未来扩展其他语言的具体规划尚未确定，包括预计扩展的语言种类、时间节点以及实现方式等。

服务器硬件资源动态调整阈值：在性能需求的负载均衡部分，实时监控 API 请求流量时，对于触发自动扩容的流量峰值阈值尚未明确，需确定具体数值以便系统能够准确进行动态负载分配和预警。

数据脱敏算法更新周期：数据脱敏与安全部分提到定期对数据脱敏算法进行更新和优化，但未明确更新周期，需确定具体时间间隔，以保障数据脱敏效果的有效性和及时性。

模型优化效果评估标准：在用户反馈与模型优化中，技术团队根据用户反馈优化深度学习模型，但对于模型优化效果的评估标准未详细说明，需确定具体量化指标来衡量每次优化后模型性能的提升情况。