

概要设计说明书

1、引言

1.1、背景

遥感数据具有覆盖范围及时域广、获取迅速、动态信息强等特点，已广泛应用于城市规划、气象预测、环境保护、防灾减灾、农林业监测等领域并取得了良好的经济和社会效益。

近年来，随着遥感技术的进一步发展和新一代高分辨率卫星系统相继投入应用，我国遥感领域已步入了高分辨率影像的快车道，对遥感数据的分析应用服务需求与日俱增。传统方式对高分辨率卫星遥感图像的特征刻画能力差且人工成本高。

随着人工智能的发展和落地应用，以地理空间大数据为基础，利用人工智能技术对遥感数据智能分析与解译成为未来发展趋势。

1.2、定义

名词概念	释义
项目	当前遥感场景下所需分析的全部遥感图像及对应分析结果的集合
组	单一遥感图像及其所有分析结果的集合，从属于项目
图层	遥感图像或分析结果在系统中的表现形式，为系统最小层级
综合分析台	某一项目中全部遥感图像及对应分析结果的整体展示

2、任务概述

2.1、目标

- 紫宸系统基于一站式AI开发平台AI Studio对上传遥感图片进行AI智能分析，旨在全面、可视化展示遥感影像在系统下的分析结果，实现一站式分析的目标
- 系统在对单个或单组图片实现变化检测、地物分类、目标检测、目标提取等功能的基础上，对于分析结果进行数据呈现以及多种结果的分析整合，实现对传入遥感图像的多维度、多颗粒度的全面分析

2.2、用户和特点

系统用户定位于有遥感图像分析需求的专业人员或学生

特点

- 有较高的素质可以对图片的分析结果做出判断
- 有较多的图像处理分析等需求
- 希望能够更加有效率的对一定数量量级的遥感图像进行批量分析处理
- 希望能够对某一遥感图像进行全面分析

3、需求规定

3.1、需求整体说明

本系统针对遥感图像的特征分析及对图像整合分析的需求，提供综合分析，变化检测、地物分类和目标提取以及目标检测的核心功能。同时本系统满足基本可用性，安全性和性能需求，在一定的技术和资金限制下，为用户提供高效、准确的遥感图像分析。

3.1.1、用例模型

	A	B	C
	用例	Actor	简要说明
1			
2	项目存储	用户	用户登录，可以新建项目到项目库，在项目库中进行操作，并保存，便于复盘和批量操作
3	视角切换	用户	切换轴测和平面视角，对可视化结果有一个直观的了解
4	综合分析	用户	将项目中所有已上传图片 and 对应全部分析结果按照图层顺序进行整合
5	对象导出	用户	可以将项目中的某图片的分析结果导出
6	变化检测	用户	可以调用变化检测算法完成图片的变化检测分析并展示结果
7	地物分类	用户	可以调用地物分类算法完成图片的地块分类分析并展示结果
8	目标提取	用户	可以调用目标提取算法完成图片的道路切割分析并展示结果
9	目标检测	用户	可以调用目标检测算法完成图片的目标检测分析并展示结果

3.2、功能描述

项目库	视角切换
<ul style="list-style-type: none">以项目为基础，每一项目中可存储多个待分析图片；待分析图片在经过分析后将会在项目中自动	<ul style="list-style-type: none">系统对于任意分析均提供轴测和平面两种视角

<p>生成组从属于项目层级</p> <ul style="list-style-type: none">新建项目后用户可自由在项目下上传图片 and 整理已分析结果所有项目将会存储在“最近项目”标签下项目库中可通过搜索查找目标项目	<ul style="list-style-type: none">轴测视角通过纵向排列的方式全面展现多张图片的对比与分析结果平面视角通过图层相叠的方式呈现，可通过拖拽操纵轴进行局部分析结果的查看和对比
<p>综合分析</p> <ul style="list-style-type: none">将项目中所有已上传图片 and 对应全部分析结果按照图层顺序进行整合提供轴测与平面两种视角	<p>对象导出</p> <ul style="list-style-type: none">可在相应功能模块下对于相应分析结果及原始图片以png或tiff格式进行导出图层中所有图片可单独导出在综合分析模块可将所有图片以组为单位全部导出地物分类模块的数据分析结果可以csv格式导出
<p>变化检测</p> <ul style="list-style-type: none">给定两张不同时间拍摄的相同位置（地理配准）的遥感图像，可定位出其中建筑变化的区域可同时上传多组项目内图片进行同步分析并得到相应结果	<p>地物分类</p> <ul style="list-style-type: none">对待分析遥感图像进行分析展示地块及其他对象在图像表达上的划分输出图片同时可生成不同对象在相应地块上面积覆盖占比可视化图表颜色与对象的对应关系清晰在图层展现
<p>目标提取</p> <ul style="list-style-type: none">对待分析遥感图像进行道路轮廓的提取并加以区分可根据分析结果中的模拟对比图层与原始图像进行对比直观观察路网分布情况	<p>目标检测</p> <ul style="list-style-type: none">对待分析遥感图像识别目标对象类型在图像中的位置可同时识别遥感图像中多种目标对象类型

3.3、质量指标描述

3.3.1、性能

项目几乎无并发需求，因此性能方面只考虑低并发场景。

- 资源相关类接口（如上传图片，图片处理等接口）：
根据资源大小以及网络质量影响功能性能，在普通大小（2000*2000像素、10mb及以下、.png/.jpg/.jpeg类型图片）、网络状态正常（带宽5mbps及以上）的情况下，接口处理速度在2s左右。
- 非资源相关类接口（其他接口）：
网络状态正常（带宽5mbps及以上）的情况下，接口处理速度在1s以内。

3.3.2、安全性

- 使用gorm防止SQL注入
- 配置阿里云轻量资源服务器安全组，防止黑客恶意使用其他端口
- 对用户密码有强制性要求（必须含有大小写字母、数字，长度要求等），单向加密密码
- token多重验证，密钥仅在后端程序中保存，后端只接受自己签发的token，防止黑客伪造token

3.3.3、易用性

- 便于管理
 - 以项目为基础，每一项目中可存储多个待分析图片；待分析图片在经过分析后[将会在项目中自动生成组](#)从属于项目层级
- 功能整体逻辑统一
 - 所有功能操作逻辑基本相同，降低学习成本
- 功能入口明显
 - 系统功能结构扁平化，所有入口均清晰可见可见

3.3.4、可靠性

- 要求系统平均故障间隔时间大于3个月，且平均修复时间小于6小时。
- 要求严重错误不得出现超过1次，大错误不得超过3次，错误不得超过5次。严重错误指数据完全丢失或损坏，大错误表示出现批量数据错误，错误表示单个用户某项某次数据错误。

3.3.5、可修改性

当系统的功能发生改变时，要易于修改而且修改的代价要尽可能低，所以系统要保持模块化结构，当某一功能模块发生修改时不能影响到其他的功能模块。系统要限制用户可能的选择。系统要有高内聚，低耦合的特点，不同模块间的联系不能过多，防止出现连锁反应。系统开发过程要制定文档规范，编写健全的文档，使修改易于进行。

3.3.6、可拓展性

采用模块化开发，开发人员需要添加新功能，只需要在相应的模块进行添加代码即可，不影响整体的架构。面向接口编程，新的模块被加入系统中时，只需要符合接口编程的规范以及路由注册规范，原有的功能模块不必被修改，系统很容易被扩展。

3.3.7、可移植性

程序可在满足运行环境规定的任何机器上运行，易于移植。

3.4、输入输出要求

- 输入要求
 - 账号格式仅支持字母开头，由小写英文字母和数字组成的4-16位字符格式
 - 密码格式仅支持长度8-20位，仅可且必须包括数字、大写字母、小写字母
 - 图片处理模块仅支持 .png/.jpg/jpeg 类型图片

- 结果分析名称不超过50个字符
- 输出要求
 - 输出的信息必须与输入数据库中的信息保持字符类别一致，数据库支持字符集：utf8mb4。

3.5、数据管理能力要求

数据分别存储在阿里云RDS MySQL云数据库和阿里云OSS对象存储服务下，由阿里云服务保证数据安全，若出现数据丢失，管理员也可以进行数据恢复操作。

3.6、故障处理要求

- 若因为各种原因出现处理异常情况，应保证此次异常不会影响整体系统以及用户数据
- 应保证整体系统不会崩溃，接口出错概率不超过1%

4、运行环境规定

4.1、设备

服务器要求 内存 >4G 存储容量 >512G

网络环境 有要求

显存要求 NVIDIA TESLA T4 10-16G

4.2、支持依赖

系统技术栈

前端：mui axios mobx react typescript vite

后端：go v1.15, aliyun-oss-go-sdk v2.2.2, jwt-go v3.2.0, gin v1.7.7, gorm.io/datatypes v1.0.6, gorm.io/driver/mysql v1.3.3, gorm.io/gorm v1.23.5

阿里云轻量应用服务器，阿里云 OSS 对象存储，阿里云 RDS MySQL 云数据库

算法：flask paddlepaddle PaddleRS opencv-python numpy 以及相关依赖，详见requirements.txt 文件

4.3、接口

4.3.1、用户接口

在用户界面部分，界面经过专门设计，简洁明了，符合用户的交互习惯。不同模块的界面布局差异小，降低用户的学习成本。前端部分代码的编写要符合HTML5的规范，可以在浏览器上正常显示。界面上有帮助按钮，用户可以随时查看系统帮助。前端部分采用React的框架，有专门的设计，不同的界面之间具有相同的风格。系统中涉及的主要用户接口如下

- (1) 用户可以通过网络访问网页首页，进行遥感图像分析，也可以按已有的项目批量处理图片。
- (2) 系统管理员可以通过操作数据库管理引擎，来实现对于数据库文件进行定时备份和恢复等操作。

(3) 登录系统必须通过特定的网络端口号才能与数据库进行通信，从而进行账户密码等匹配操作。

4.3.2、外部接口

本智能遥感平台分析系统采用B/S的三层架构样式，所以用户可以在PC用浏览器对于本系统进行访问。本系统的部署需要一些外部接口的支持：

(1) 服务器端需安装go, paddlepaddle,paddleRS, mui axios mobx react typescript vite等依赖，客户端需要安装浏览器。

(2) 应用程序服务器要保证8G的运行内存，30G以上的硬盘空间，8G以上的显卡显存。云数据库服务器需要保证400G以上的硬盘空间。

4.3.3、后端接口

后端使用go语言开发，主要使用gin路由框架与gorm数据处理框架开发实现接口。

- 用户类型接口
 - 用户注册
 - 用户登录
 - 获取用户信息
- 项目类型接口
 - 创建项目
 - 搜索项目
 - 获取最近项目
 - 将项目移入回收站
 - 从回收站中恢复项目
 - 将回收站中项目彻底删除
 - 在回收站中搜索项目
 - 获取回收站中最近项目
- 图片和图片组类型接口
 - 在项目中上传图片
 - 修改已上传图片名称
 - 删除已上传图片
 - 修改图片组名称
 - 删除图片组
 - 获取项目下所有图片和图片组
- 图片处理相关接口
 - 综合分析
 - 地物分类
 - 变化检测

- 目标检测
- 目标提取

4.3.4、算法接口

python实现地块分类，道路切割，目标检测，变化检测四个算法接口，借助paddlepaddle框架和PaddleRS进行项目部署。