**基于百度飞桨的遥感图像智能解译平台**

**概要设计说明书**



二〇二二年六月

**概要设计说明书**

目录

[1 引言 4](#_Toc108008610)

[1.1 编写目的 4](#_Toc108008611)

[1.2 项目背景 4](#_Toc108008612)

[1.3 定义（专门术语、缩写） 4](#_Toc108008613)

[1.4 参考资料 5](#_Toc108008614)

[2 任务概述 5](#_Toc108008615)

[2.1 目标 5](#_Toc108008616)

[2.2 运行环境 5](#_Toc108008617)

[2.3 需求概述 5](#_Toc108008618)

[2.4 条件与限制 6](#_Toc108008619)

[3 总体流程 6](#_Toc108008620)

[3.1 总体架构设计 6](#_Toc108008621)

[3.1.1 系统架构设计 7](#_Toc108008622)

[3.1.2 系统技术架构 9](#_Toc108008623)

[3.1.3系统模块设计 9](#_Toc108008624)

[3.2 功能分配 9](#_Toc108008625)

[3.3 处理流程 10](#_Toc108008626)

[3.3.1 用户管理 11](#_Toc108008627)

[3.3.2 通用功能 14](#_Toc108008628)

[3.3.3 遥感图片分析 19](#_Toc108008629)

[4 接口设计 24](#_Toc108008630)

[4.1 外部接口 24](#_Toc108008631)

[4.1.1本系统需要调用的外部接口： 24](#_Toc108008632)

[4.1.2本系统对外提供的接口 24](#_Toc108008633)

[4.2 内部接口 25](#_Toc108008634)

[5 技术亮点 25](#_Toc108008635)

[5.1 新手引导，易于使用 25](#_Toc108008636)

[5.2 批处理+进度条 26](#_Toc108008637)

[5.3 图像渲染 27](#_Toc108008638)

[5.4 图像分析 27](#_Toc108008639)

[5.5 日志打印 28](#_Toc108008640)

[5.6 数据检验 28](#_Toc108008641)

[6 数据结构设计 30](#_Toc108008642)

[6.1 数据库数据结构设计 30](#_Toc108008643)

[6.1.1概念结构设计 30](#_Toc108008644)

[6.1.2 逻辑结构设计 30](#_Toc108008645)

[6.1.3 数据字典设计 30](#_Toc108008646)

[6.2数据库部署结构 31](#_Toc108008647)

[6.2.1现有部署描述 31](#_Toc108008648)

[6.2.2部署方案分析 32](#_Toc108008649)

[7 出错设计 32](#_Toc108008650)

[8 安全保密设计 33](#_Toc108008651)

[8.1系统面临的安全威胁 33](#_Toc108008652)

[8.2系统安全方案 33](#_Toc108008653)

[8.2.1服务器及客户端系统安全 33](#_Toc108008654)

[8.2.2应用系统安全 34](#_Toc108008655)

[9 维护设计 34](#_Toc108008656)

[9.1 维护事项收集 34](#_Toc108008657)

[9.1.1 系统本身的故障 34](#_Toc108008658)

[9.1.2 新需求 34](#_Toc108008659)

[9.2 分析设计维护方案 35](#_Toc108008660)

[9.3 系统维护 35](#_Toc108008661)

[9.4 系统更新、重新发布 35](#_Toc108008662)

# 1 引言

## 1.1 编写目的

该概要说明书主要编写的目的是为了让项目整体的概况得到一个具体的解释，读者主要是开发人员以及提出要求的人员，方便他们对该项目的设计概况得到整体的了解。概要设计报告中将对系统所做的所有概要设计进行详细的说明。在后面的的详细设计中，程序设计员可以参考该概要设计报告，对系统进行详细设计。在以后的软件测试以及软件维护阶段也可参考此说明书,以便于了解在概要设计过程中所完成的各模块设计结构，或在修改时找出在本阶段设计的不足或错误。

## 1.2 项目背景

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。随着人工智能及深度学习技术快速发展，通过应用深度学习技术可以加速遥感领域智能化应用，促进遥感数据处理走向智能化，使我国遥感事业更好地服务国计民生。该软件系统主要用于对遥感图片进行相关处理。

## 1.3 定义（专门术语、缩写）

目标提取（使用图像分割技术对卫星图像中指定对象完成分割）

变化检测（使用图像分割技术对同区域两个时期的卫星图像变化情况完成分析）

目标检测（使用目标检测技术对卫星图像中指定对象完成检测）

地物分类（使用图像分割技术对卫星图像每个像素完成分类）

AI Studio平台（飞桨AI Studio人工智能学习与实训社区）

PaddlePaddle（百度研发的深度学习平台）

Flask(使用Python 编写的轻量级 Web 应用程序框架)

Vue(套用于构建用户界面的渐进式框架)

## 1.4 参考资料

需求规格及使用说明书

# 2 任务概述

## 2.1 目标

主要目标就是设计出符合中软杯比赛的要求的程序，主要包括四个功能，目标提取，变化检测，目标检测，地物分类，除去四个功能以外，还要满足软件运行在WEB平台注重用户界面的视觉、交互及分析结果的创新展示。

## 2.2 运行环境

服务器运行在Linux环境下，操作系统Debian 10.2 64bit，python3.7.5，mysql 5.7,

客户端运行在个人电脑的浏览器，对性能没有要求。

## 2.3 需求概述

该项目需要实现目标提取、变化检测、目标检测和地物分类四大分析功能，软件运行在WEB平台。其中“变化检测”一项功能，需要保证检测结果的准确度尽可能高，Web软件平台要尽可能地利于用户操作，界面要尽可能的美观。

来自需求分析说明书：

表1 需求对应表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | PaddleRS-0010 | 注册 | 3.1注册/登录 |  |
| 2 | PaddleRS-0020 | 登录 | 3.1注册/登录 |  |
| 3 | PaddleRS-0030 | 修改信息 | 3.1注册/登录 |  |
| 4 | PaddleRS-0040 | 导航界面 | 3.2导航界面 |  |
| 5 | PaddleRS-0050 | 图像上传 | 3.3 图像上传/下载 |  |
| 6 | PaddleRS-0060 | 图像下载 | 3.3 图像上传/下载 |  |
| 7 | PaddleRS-0070 | 显示图片详情 | 3.4图片详情 |  |
| 8 | PaddleRS-0080 | 图片批处理 | 3.5图片批处理 |  |
| 9 | PaddleRS-0090 | 图片渲染 | 3.6图片渲染 |  |
| 10 | PaddleRS-0100 | 目标提取 | 3.7目标提取 |  |
| 11 | PaddleRS-0110 | 变化检测 | 3.8变化检测 |  |
| 12 | PaddleRS-0120 | 目标检测 | 3.9 目标检测 |  |
| 13 | PaddleRS-0130 | 地物分类 | 3.10地物分类 |  |
| 14 | PaddleRS-0140 | 系统性能指标 | 4.系统性能指标 |  |
| 15 | PaddleRS-0150 | 系统接口需求 | 5系统接口需求 |  |

## 2.4 条件与限制

条件：

1、系统运行顺畅无卡顿，无闪退等严重BUG

2、UI界面美观、逻辑合理、交互友好、可视化美观

硬件限制：

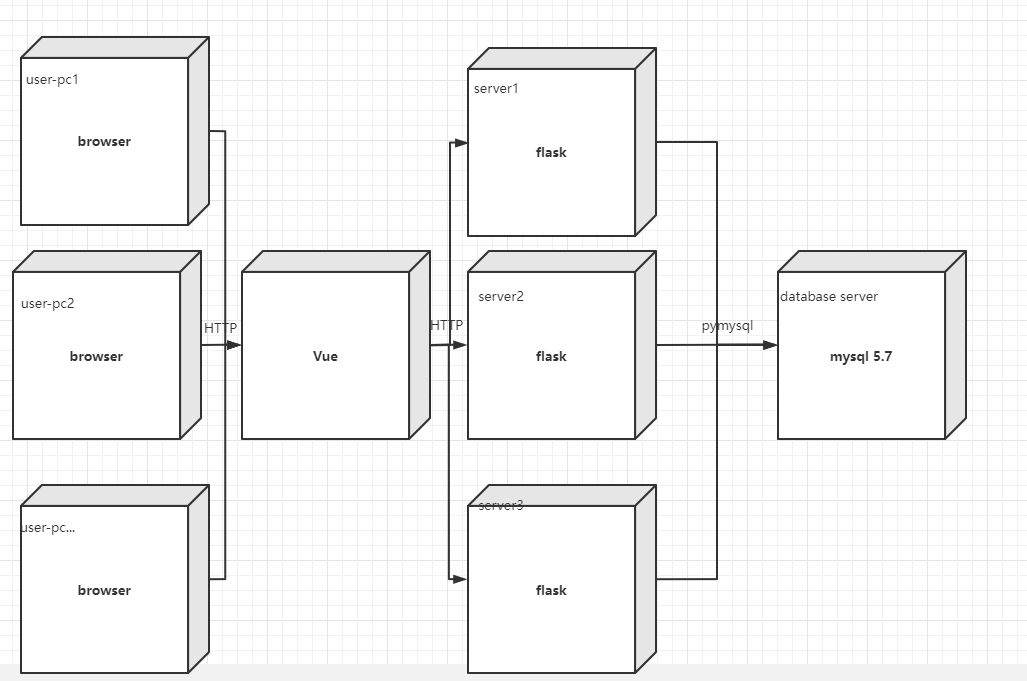
客户端：普通PC 内存大于等于2GB

Web服务器：内存大于等于8GB RAM（4核），硬盘大于等于40GB

模型训练服务器：GPU:TeslaV100.VideoMem:32GB CPU：4Cores.RAM:32GB.Disk:100GB

# 3 总体流程

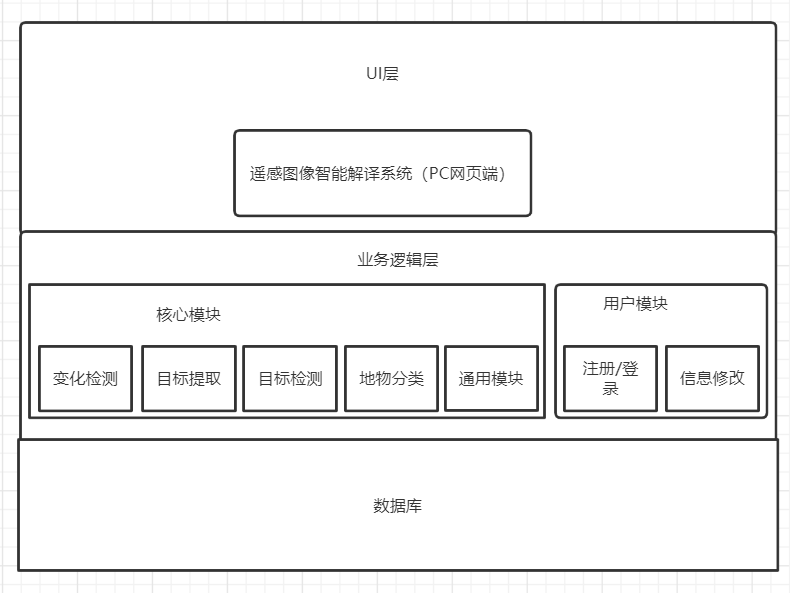
## 3.1 总体架构设计



### 3.1.1 系统架构设计

根据项目需求，本系统面向PC端用户采取C/S(客户端/服务端)架构设计，软件体系结构为分层结构，一共分为五层，用户界面层，控制层，业务逻辑层，持久层，数据库。

系统逻辑架构设计图



每个层次功能与作用介绍如下：

**1.用户层：**

该板块为用户与系统进行人机交互的PC端Web界面，使用Vue实现。Vue 是一套用于构建用户界面的渐进式框架。与其它大型框架不同的是，Vue 被设计为可以自底向上逐层应用。Vue 的核心库只关注视图层，不仅易于上手，还便于与第三方库或既有项目整合。另一方面，当与现代化的工具链以及各种支持类库结合使用时，Vue 也完全能够为复杂的单页应用提供驱动。（官方介绍）

Vue 只关注视图层， 采用自底向上增量开发的设计。

Vue 的目标是通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件。

**2. 业务层：**

可以从逻辑上分为三个板块：

遥感图像分析模块（每个模块都有图片批处理和图片渲染功能）：

（1）目标提取（使用图像分割技术对卫星图像中指定对象完成分割）

（2）变化检测（使用图像分割技术对同区域两个时期的卫星图像变化情况完成分析）

（3）目标检测（使用目标检测技术对卫星图像中指定对象完成检测）

（4）地物分类（使用图像分割技术对卫星图像每个像素完成分类）

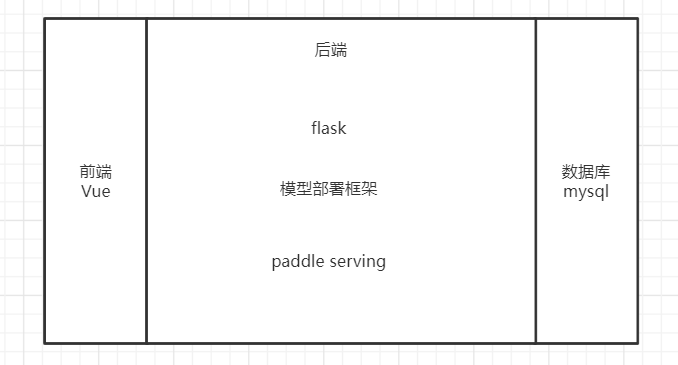
用户管理模块：用户登录和用户注册以及用户个人信息修改界面

图片上传/下载模块

**3. 数据库层：**

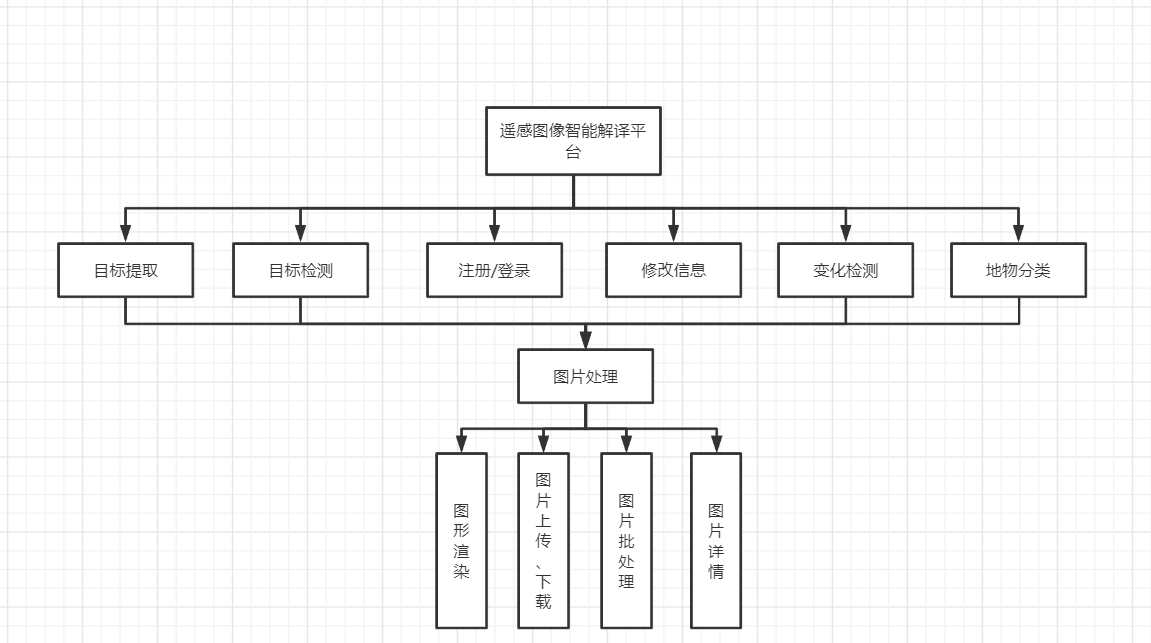
该层使用MySql数据库产品进行构建，同时增加一个备份子库。因为业务不是很复杂无需添加redis等充当缓存。

### 3.1.2 系统技术架构



### 3.1.3系统模块设计

以下为遥感图像智能解译平台的功能架构设计图



## 3.2 功能分配

用户管理模块：

用户注册：用户第一次访问系统，需要自己自定义账户和密码。

用户登录：用户输入自己已注册的账户和密码，来进入系统。同时服务器设置了拦截器，用来防止非法跳转。

信息修改：用户可以在这里修改自己的用户名，注册的电话号码，邮箱以及密码

遥感图像分析模块：

（1）目标提取（使用图像分割技术对卫星图像中指定对象完成分割）

（2）变化检测（使用图像分割技术对同区域两个时期的卫星图像变化情况完成分析）

（3）目标检测（使用目标检测技术对卫星图像中指定对象完成检测）

（4）地物分类（使用图像分割技术对卫星图像每个像素完成分类）

（5）图片处理模块，为以上四个分析模块共有的模块。具有以下功能：

① 图像上传/下载：为了对遥感图像进行编译和检测，首先要将图片上传至服务器，再交给模型处理。然后可以下载处理好的遥感图片

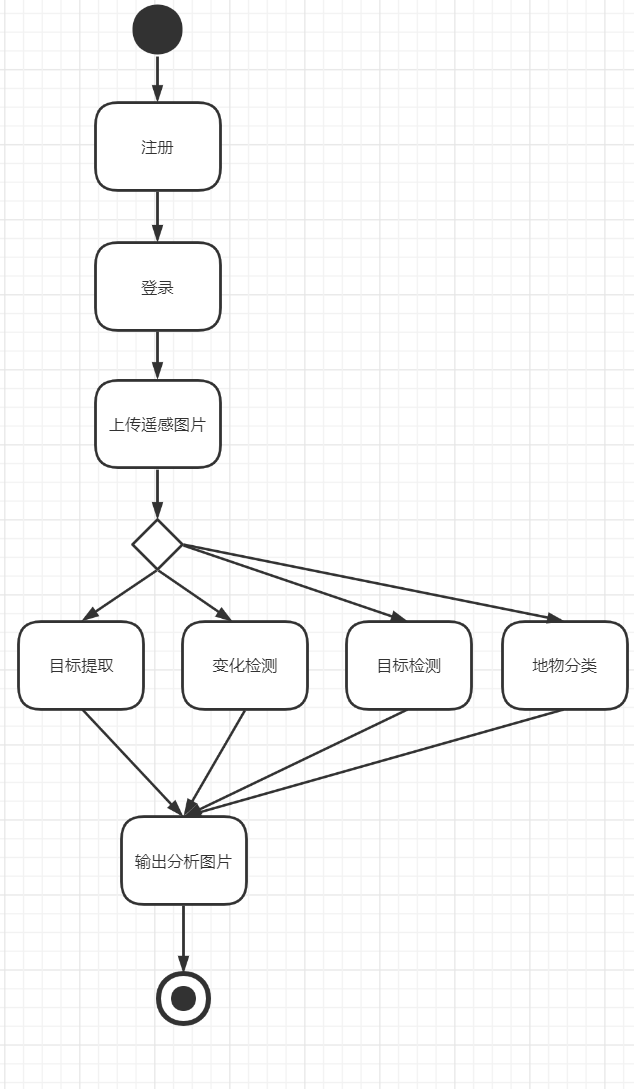
② 图片详情：系统对上传和处理后图片有个基本的分析，分析图片长、宽、位深、大小和类型，处理后的图片额外有处理时长和目标名称（仅目标提取和目标检测）

③ 图形渲染：系统对处理后的图片进行渲染，和用户有更直观的视觉体验。

④ 批处理：支持图片批处理，按照提示上传压缩过的图片（zip格式）可以批量生成处理后的图片。

## 3.3 处理流程

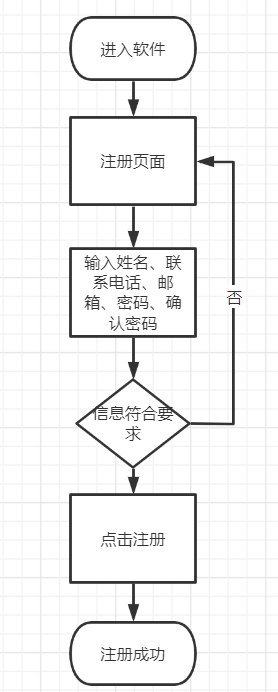
整体流程：



### 3.3.1 用户管理

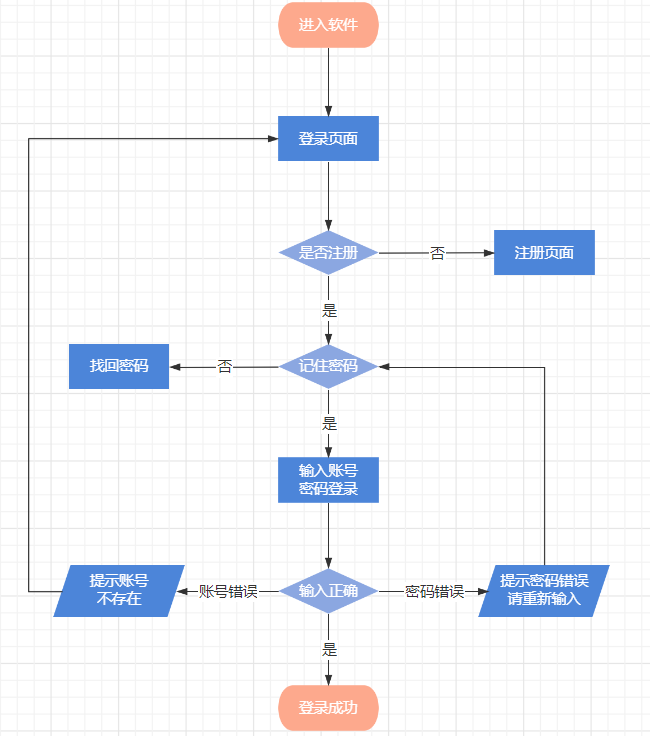
**1. 用户注册**

程序流程：



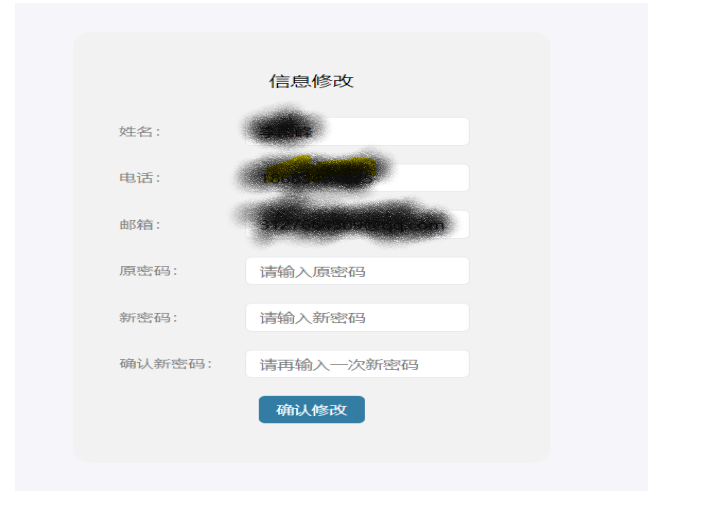
**2.用户登录**

程序流程：



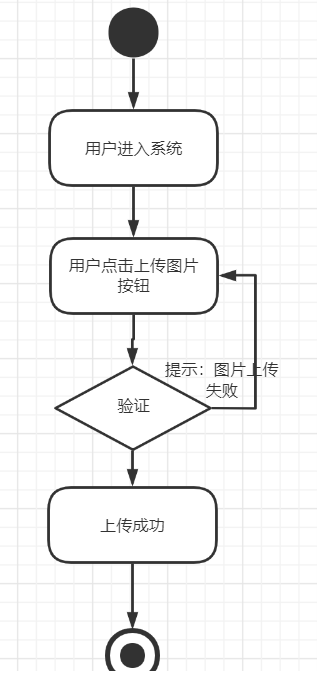
**3.修改信息**

用户可以通过修改信息界面进行对自己账户信息的修改，同时也可以对账户的账号和密码进行修改



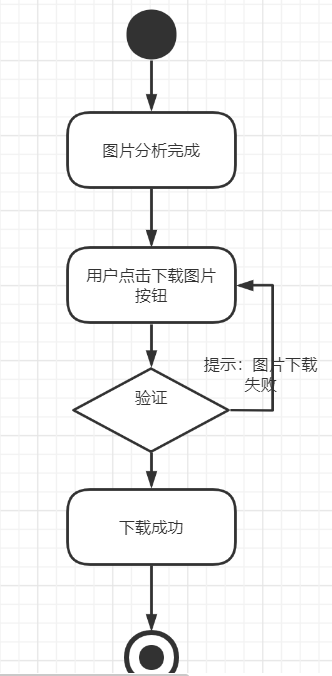
### 3.3.2 通用功能

**1. 图片上传：**



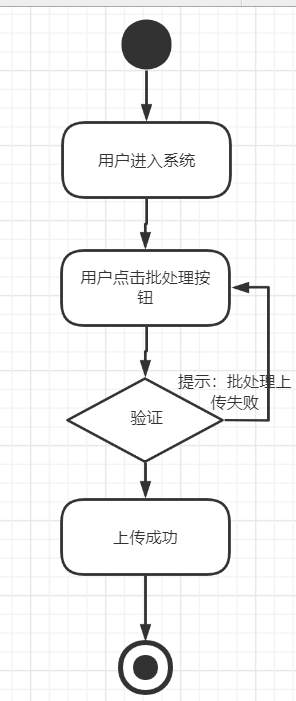
**2. 图片下载：**

程序流程：



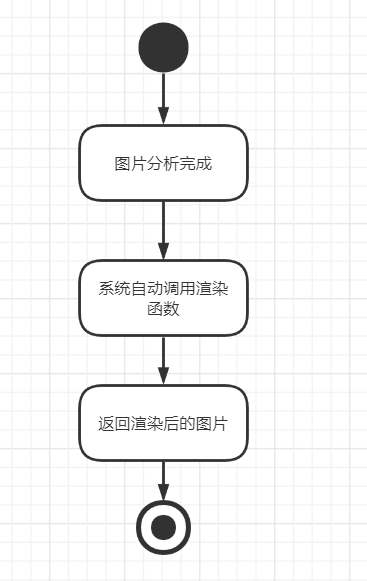
**3. 批处理：**

程序流程：



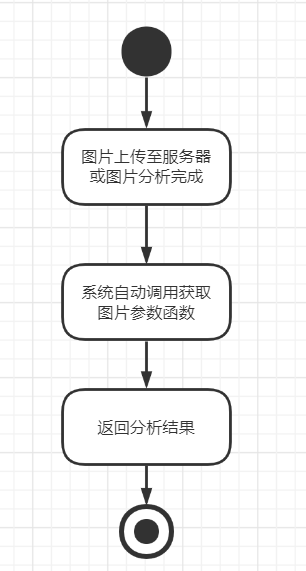
**4. 图形渲染**

程序流程：



**5. 图片详情**

程序流程：



### 3.3.3 遥感图片分析

四个功能的基本流程差不多，以变化检测为例

图片举例：

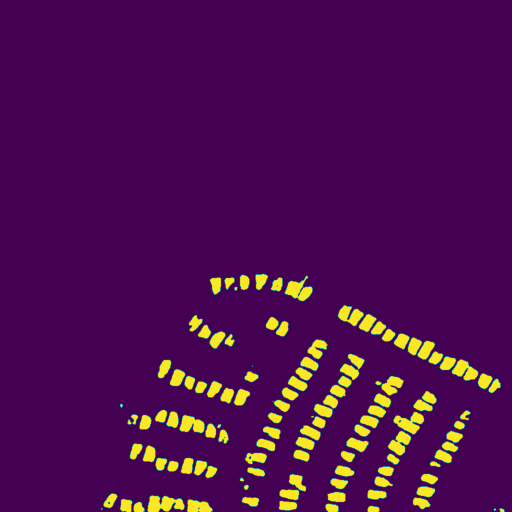
（变化前图片）A



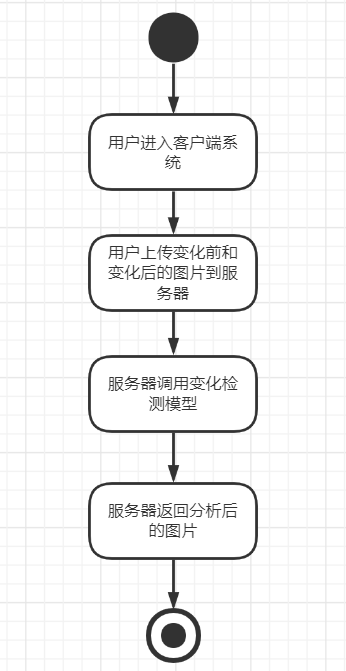
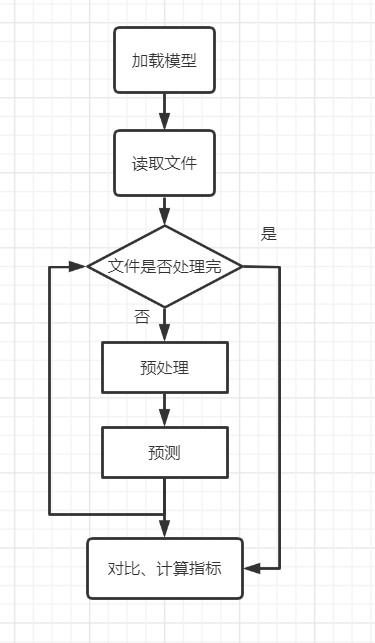
（变化后图片）B



变化检测输出：



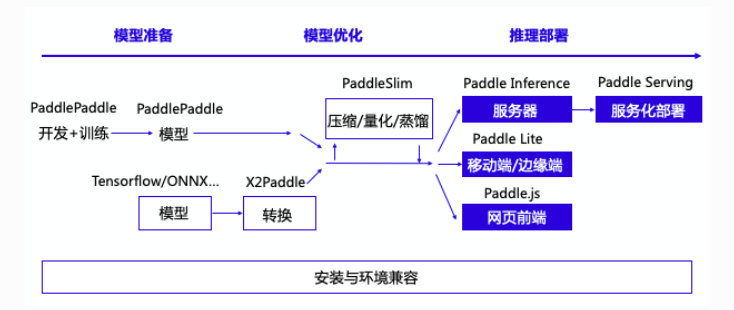
变化检测：



模型内部情况

服务器调用模型及远程交互用到了Paddle serving的概念。Paddle serving介绍; paddle serving调用了paddle inference做远程服务。paddle serving可以认为是一个示例，都会使用paddle serving作为统一预测服务入口。如果在web端交互，一般是调用远程服务的形式，可以使用paddle serving的web service搭建。

在模型训练完毕之后，紧接着使用PaddleSlim来对模型进一步处理，之后再通过paddle serving部署模型。示例如下：



# 4 接口设计

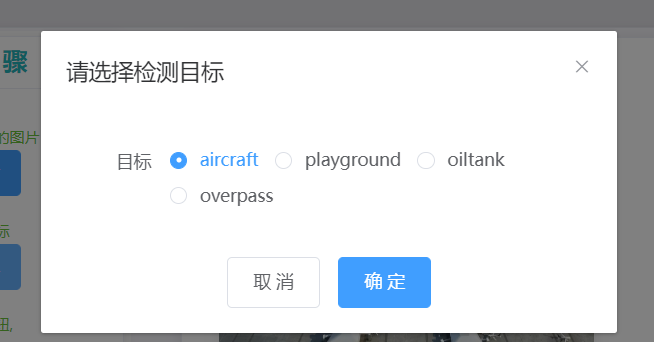
## 4.1 外部接口

### 4.1.1本系统需要调用的外部接口：

模型调用接口: 处理分析的遥感图片时，需输入特定值来调用部署的模型，当调用特定的模型接口处理完数据后，会将处理结果返回给调用端。详情如下：

目标检测的外部接口：

目标检测用户需选择检测目标，选择的检测目标将传回服务器作为模型调用的初始值。



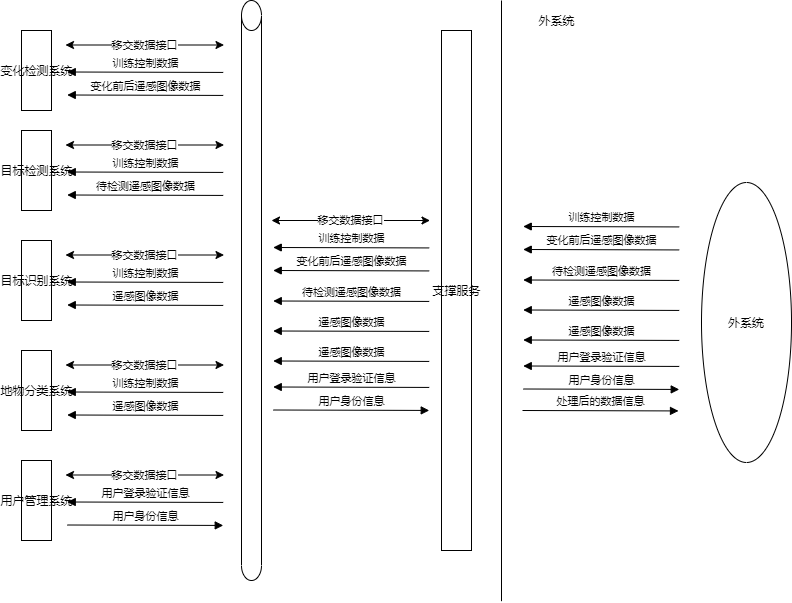
目标提取

### 4.1.2本系统对外提供的接口

网站访问接口：支持在PC端通过服务器的IP地址和对应端口，外部访问服务器网站的特定资源，如用户登录、遥感图像处理等功能的使用。

图像上传/下载接口：支持用户在PC端借助浏览器，将遥感图像上传到后台，也支持用户下载其想要的后台图片到网页或者指定文件存储路径中。

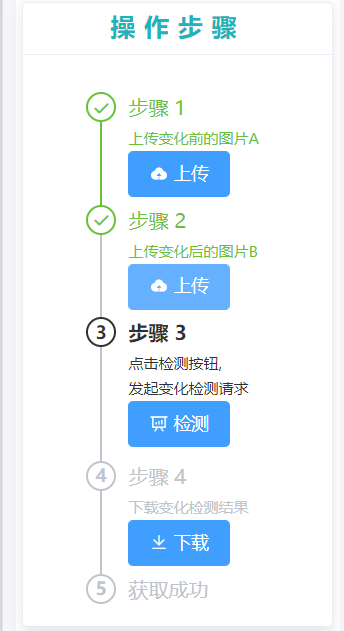
## 4.2 内部接口



# 5 技术亮点

## 5.1 新手引导，易于使用

用户体验良好，每一步均有操作指引和注意事项，而且用户操作后按钮会改变颜色。





## 5.2 批处理+进度条

有文件的批处理，可以批量上传下载图片并且有进度条显示。



## 5.3 图像渲染

有数据结果的图像渲染，使用户更能直观了解。



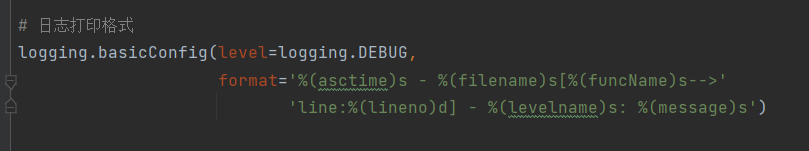
## 5.4 图像分析

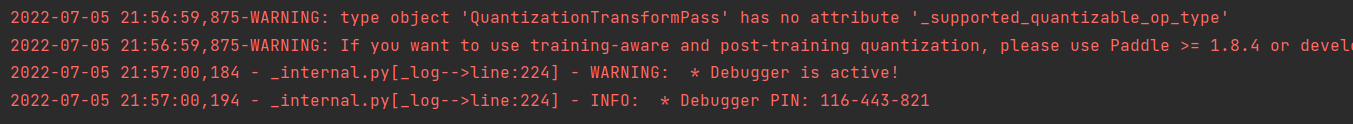
有数据图像的分析展示，可以让用户了解分析的时间和分析后图片的基本信息。



## 5.5 日志打印

服务端加入了日志打印，可以更方便快捷的查看报错信息。





## 5.6 数据检验

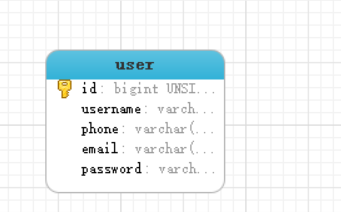
系统会自动判断用户输入的数据信息是否合理，并提示其修改。



# 6 数据结构设计

## 6.1 数据库数据结构设计

### 6.1.1概念结构设计



### 6.1.2 逻辑结构设计

用户信息（id，用户名，手机号，邮箱，密码）

### 6.1.3 数据字典设计

（1）数据项

数据项名称：id

别名：无

类型长度：varchar（10）

描述：用户的唯一识别码

数据项名称：密码

别名：password

类型长度：varchar（255）

描述：用户登录的密码，为MD5加密后的字符串

数据项名称：用户名

别名：username

类型长度：varchar（255）

描述：用户注册时输入的用户名

数据项名称：手机号

别名：phone

类型长度：varchar（255）

描述：用户注册时输入的手机号，有前端数据检验，存入数据库为标准电话号码

数据项名称：电子邮件

别名：eamail

类型长度：varchar（255）

描述：用户注册时输入的邮箱地址，有前端数据检验。

（2）数据结构

数据结构名称：用户信息

含义说明：定义了不同用户的相关信息

组成：id，账号，密码，手机号，邮箱

（3）外部实体

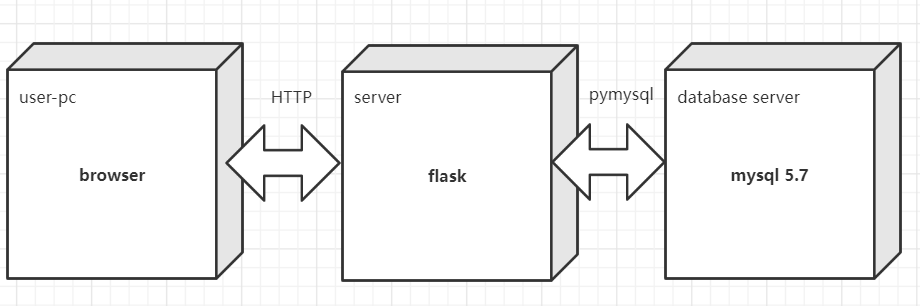
名称：用户

说明：使用本系统的人

输入数据流：注册的个人信息

输出数据流：上传的图片

## 6.2数据库部署结构



### 6.2.1现有部署描述

本项目中的数据库部署方案采用单体架构

### **6.2.2部署方案分析**

最简单也是最早的通用部署方案，在用户量比较低时的架构方法。即一个应用服务器完成单一的功能。

缺点分析：

代码耦合，开发维护困难

无法针对不同模块进行针对性优化

无法水平扩展

单点容错率低，并发能力差

# 7 出错设计

用一览表的方式说明每一种可能出错的情况出现时，系统输出信息的形式、含义级处理的方法。

由于输入信息不符合规范（如：商品数量为负数等），称之为软错误；

由于硬件方面的错误（如：网络传输超时、硬件出错等），称之为硬错误；

对于一些关键的操作（如：删除操作），应该提供提示确认机制；

对于数据、测试文档，都是要提供相应的保密措施设置

|  |  |  |
| --- | --- | --- |
| 错误类型 | 子项 | 错误原因 |
| 数据库错误 | 连接 | 连接超时 |
| 连接断开 |
| 数据库 | 数据库代码错误 |
| 数据库溢出 |
| 系统部分自定义错误 | 权限错误 | 管理员权限设置故障 |
| 输入错误 | 用户ID错误/为空 |
| 用户密码错误/为空 |
| 验证码错误/为空 |
| 查找错误 | 为查找到符合要求的  记录 |
| 链接错误 | 内部连接错误 | 网页链接错误 |
| 文件/图片链接错误 |
| 外部链接错误 | 页面休整或者其他错  误无法打开 |
| 连接网址维护或  其他错误无法打开 |
| 服务器错误 | 服务器宕机 | 服务器遭受网络攻击 |
| 客户端传入数据太大，服务器内存超出 |
| 客户端传入数据太大，服务器CPU过载 |
| 服务器无响应 | 网络断开或网络较弱 |
| 服务器返回错误 | 客户端发送数据存在问题 |

补救措施主要包括：

1．对于软错误（如客户非法操作），需要在添加/修改操作中及时对输入数据进行验证，分析错误的类型，并且给出相应的错误提示语句，传送到客户端的浏览器上；

2．对于硬错误（如内存溢出），错误类型较少而且比较明确，所以可以在可能出错的地方中输出相应的出错语句，并将程序重置，最后返回输入阶段；

3．后备技术说明准备采用的后备技术，当原始数据丢失时启用的副本的建立和启动的技术，例如周期性的把磁盘信息记录在案；

4．恢复及在启动技术说明将使用的恢复再启动技术，是软件从故障点恢复执行或是软件从头开始重新开始运行的方法

5. 服务器在宕机一段时间后，自动重新启动服务。

# 8 安全保密设计

系统的安全设计的主要对象是系统自身的应用安全、数据安全、服务器操作系统和数据库的安全管理维护。

## 8.1系统面临的安全威胁

本系统需要考虑系统及数据可能面临的以下安全威胁：

1、非人为因素：服务器意外断电、损坏、硬盘出错或损坏，网络中断等；

2、人为因素：操作失误，恶意攻击，病毒破坏等；

3、信息泄露、信息窃取、假冒等；

4、系统软件安全漏洞。

## 8.2系统安全方案

针对上述安全威胁，系统的安全运行依赖网络和服务器系统的安全，系统本身需要设计相应的安全监控功能。

### 8.2.1服务器及客户端系统安全

对于数据库系统，进行相应的安全配置维护管理，根据实际情况及时进行安全策略调整，定期进行数据库系统的有关备份。

由于客户端计算机用途很开放，很容易受到病毒感染、恶意攻击等，可能会进一步影响到服务器，因此，对客户端计算机也要采取安全措施，进行相应的安全配置管理，如设置有效的系统密码，设置较高的浏览器级别，及时打补丁，安装反病毒程序，定期查杀病毒，根据实际情况及时采取安全措施。

服务器拥有2Gbps的防护能力，实时监控网络流量，当服务器遭受的攻击流量超过一定流量时，基于优秀特征识别算法（有效抵御 SYN Flood、ICMP Flood 等常见的 DDoS 攻击。）将自动启动 DDoS 清洗。

### 8.2.2应用系统安全

1、身份认证

用户需要注册登录后才能正常使用该系统，用户的密码采用MD5加密后存储到后端数据库中。普通用户只能登录客户端

2、用户权限控制

系统提供用户角色使用权限，即用户作为普通用户登录本系统进行遥感图片的提交和处理，以及查看处理结果等。

角色分为以下几种：

服务器系统管理员： 模型参数设置、用户管理、数据库管理

普通用户：  使用该系统进行遥感图像处理

3、安全管理和权限制度设计

明确系统的安全管理机构/部门、人员及职责，负责管理系统安全保密工作。制定系统安全保密管理制度，并严格加以执行及监督，实现资源的合理配置和统一管理，实现统一的访问控制策略，确保系统的安全运行、安全审查。

在外部安全上，使用防火墙为本系统提供一个安全的运行环境。

在系统内部，本系统用户，机构、角色、权限根据实名制层级设置，提高系统数据操作的安全保护。

4、拦截器设计

防止非法跳转

# 9 维护设计

## 9.1 维护事项收集

### 9.1.1 系统本身的故障

系统已经在使用中，但在使用过程中发现经常出现几种错误，需要对这些错误进行收集、整理、归类，并在此过程中对相关使用个人和部门及时给与支持。

### 9.1.2 新需求

随着不断地迭代更新，系统功能和结果可能发生改变，为适应新的需求，要对现有系统进行升级或二次开发。这时候需要及时、准确了解需求，并归类整理成文档。

## **9.2 分析设计维护方案**

在对问题有足够的认识，并确认有足够资源可以按时完成的情况下，需要把需求转化为用于开发的技术文档，如系统维护概要说明书、系统维护功能清单、系统维护详细说明书；之后依据现有人员列出详细的开发计划。

## **9.3 系统维护**

项目主管把握整个项目进度，负责开发任务的分配、跟踪、检视、提供必要的支持等。 开发人员主要写代码、作测试，美工负责页面设计和美化，保障页面风格一致性。测试人员，在系统上线之前做充分的测试，保障系统安全、稳定、高效运行。

## **9.4 系统更新、重新发布**

系统开发完毕，经过反复、周密的测试，由项目负责人将系统重新发布出去，但要尽量保证新旧系统切换过程中还影响业务部门使用。因此建议前期新老系统并行，待新系统稳定之后再关掉老系统（停机更新）。