

“中国软件杯” 大学生软件设计大赛
基于百度飞桨的遥感图像智能解译平台
详细设计文档

wust2022

WUST2022

参赛院校： 武汉科技大学

队伍名称： WUST CSC2022

队长： 张戈

队员： 郭臣、何欣怡、王宇航

目录

一、背景与意义.....	3
二、需求分析.....	3
2.1 基本功能要求	3
2.2 非功能性要求	4
2.3 系统性能	4
2.4 运行环境规定.....	4
三、系统架构设计.....	4
3.1 主体架构设计	4
3.2 web 页面设计.....	5
3.3 主服务设计.....	9
3.4 数据存储服务设计	10
3.5 图像解析服务设计.....	11
四、环境部署.....	11
4.1 环境配置.....	11
4.2 配置所有服务.....	12
五、 算法设计.....	16
5.1 变化检测.....	16
5.2 目标提取.....	19
5.3 目标检测.....	22
5.4 地物分类.....	25
六、项目创意及特色.....	28
七、系统测评.....	29
八、总结.....	29
附录.....	30
致谢.....	31

一、背景与意义

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，随着航天航空技术以及电子信息技术深入发展，遥感技术突破了数据获取的瓶颈，已走向全面应用的新阶段，并且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。

而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，亟需一种新的遥感图像分析与处理技术，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。近年来，以深度学习为代表的监督学习方法凭借其强大的特征学习能力，在遥感影像智能解译领域较传统方法取得了突破性进展，也为实现高效快速、自动化检测的应用需求，开辟了一个新的自动化、智能化的处理方法，实现对于遥感图像的自动分析和智能解译，是智慧城市、气象预测、环境保护和防灾减灾、农林业监测等场景的重要研究领域，对于城市规划、重点用地监控、环境保护有着重要意义。

二、需求分析

2.1 基本功能要求

使用百度 AI Studio 平台进行训练，基于国产化人工智能框架——百度飞桨 PaddlePaddle 框架进行开发，设计并开发一个可以通过深度学习技术实现对遥感图像自动解译的 WEB 系统，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：

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）

2.2 非功能性要求

- (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

2.3 系统性能

- (1) 响应时间：要求系统能在操作请求后 1 秒内作出相应，否则应基于相关提示信息。
- (2) 易使用性：要求系统能对非专业人员来说，操作简单易懂。
- (3) 高效性：要求系统模块数量合理，模块间低耦合、高内聚。
- (4) 可维护性：要求系统运行过程中，可进行维护管理等工作。
- (5) 高并发性：要求系统能运用一定的策略应对可能出现的高并发情况。

2.4 运行环境规定

2.4.1 环境

数据库环境：MySQL

Node.js 环境：V16.14.0

Python 运行环境：3.5+

2.4.2 程序运行方式

B/S 结构（Browser/Server，浏览器/服务器模式），客户机上只要安装一个浏览器，如 Chrome 或 Edge 等，服务器安装 MYSQL 数据库，并配置相应 PYTHON 的环境。浏览器通过 Web Server 同数据库进行数据交互。

三、系统架构设计

3.1 主体架构设计

根据赛题要求，我们设计了 B/S 结构（Browser/Server，浏览器/服务器模式），完成基

于百度飞桨的遥感图像智能解译平台的开发。

主要由以下四部分组成：

- (1) 用于用户操作的 web 页面
- (2) 响应用户操作的主服务
- (3) 用于数据存储的数据库服务
- (4) 用于解析遥感图像的解析服务

相关流程如图 1 所示：



图 1 项目框架设计图

3.2 web 页面设计

web 页面基于 vue-element-admin 后台前端解决方案实现。主要开发功能页有：登录/注销、目标提取、地物分类、目标检测、变化检测。

3.2.1 登录页面

输入用户名和密码登录使用基于百度飞桨的遥感图像智能解译平台，登录界面如图 2 所示。



图 2 登录界面

默认账号 admin，密码 111111。

3.2.1 主页面

页面全局包括左侧功能栏、顶部状态栏以及右侧导航栏。中部空白部分为功能页面主体，如图 3 所示。

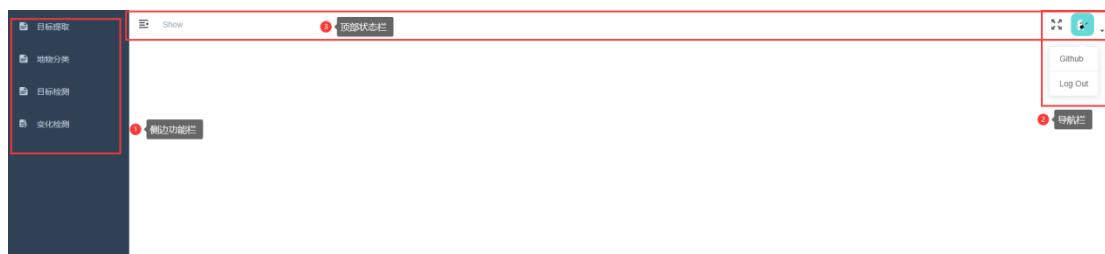


图 3 主界面

3.2.3 变化检测页面

变化检测页面主要由检测图片域、结果分析域、原始图片域以及变化图片域，共 4 部分组成。原始图片域为上传并展示用于变化检测的原始图像，变化图片域为上传并展示用于变化检测的经过时间后已变化的图像，检测图片域用以根据用户上传的原始图像和变化图像进行检测并展示检测结果，结果分析域为展示变化面积所占比例。

主要功能为：使用变化检测技术对同区域两个时期的卫星图像变化情况完成分析，依次上传原始图片和变化图片后，点击“check”按钮，系统标记出变化的区域（变化的区域标记

为白色，未标记的区域为黑色），同时对于变化的情况进行分析，显示变化面积所占比例，变化检测效果如图 4，5 所示。



图 4 变化检测页面一



图 5 变化检测页面二

3.2.4 目标提取页面

目标提取页面主要由检测图片域、结果分析域以及原始图片域，共 3 部分组成。原始图片域为上传并展示用于目标提取的原始图像，检测图片域用以根据用户上传的原始图像进行道路提取并展示检测结果，结果分析域为展示道路分割面积所占比例。

主要功能为：使用图像分割技术对卫星图像中指定对象完成分割，上传图片后，对于指定的对象进行提取标记，对于指定对象进行标记提取，同时进行结果分析，显示提取出的道路所占面积，目标提取效果如图 6 所示，对于图像中的道路进行分割提取。

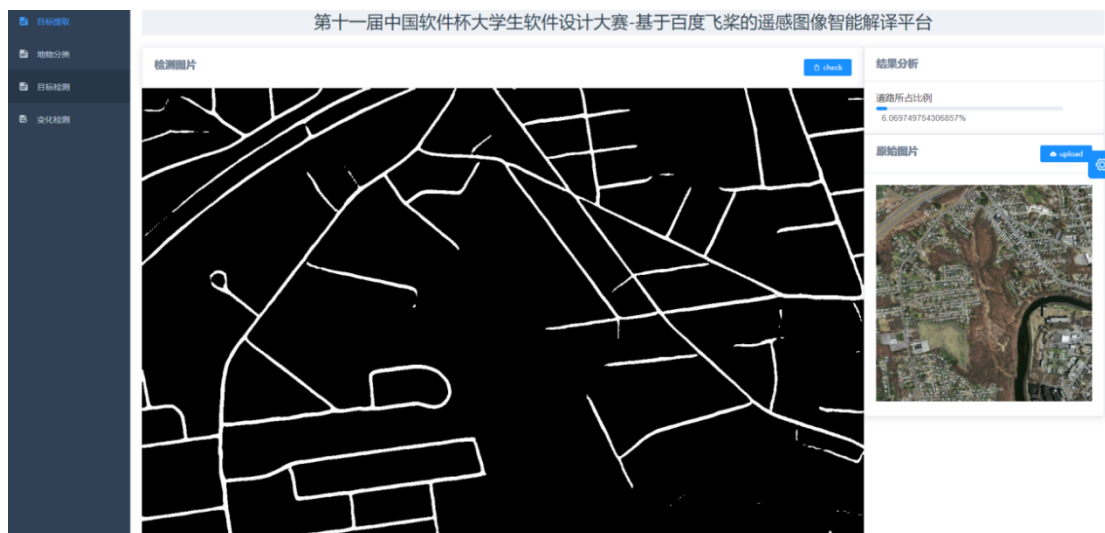


图 6 目标提取界面

3.2.5 目标检测页面

目标检测页面同样由检测图片域、结果分析域以及原始图片域，共 3 部分组成。原始图片域为上传并展示用于目标检测的原始图像，检测图片域用以根据用户上传的原始图像进行目标检测并展示检测结果，结果分析域为罗列检测出的对象的类别和置信度。

主要功能为：使用目标检测技术对卫星图像中指定对象完成检测，上传图片后，检测图片中指定对象所示的类别以及置信度，检测效果如图 7 所示。



图 7 目标检测界面

3.2.6 地物分类页面

地物分类页面也由检测图片域、结果分析域以及原始图片域，共 3 部分组成。原始图片

域为上传并展示用于地物分类的原始图像，检测图片域用以根据用户上传的原始图像进行地物分类并展示检测结果，结果分析域为罗列出每个类别在图片中所占像素比例。

主要功能为：使用图像分割技术对卫星图像每个像素完成分类，对于图片中各个位置所示的类别进行分类，大致可分为：建筑、耕地、林地、其他，不同的类别标记为不同的颜色，分类结果如图 8 所示。

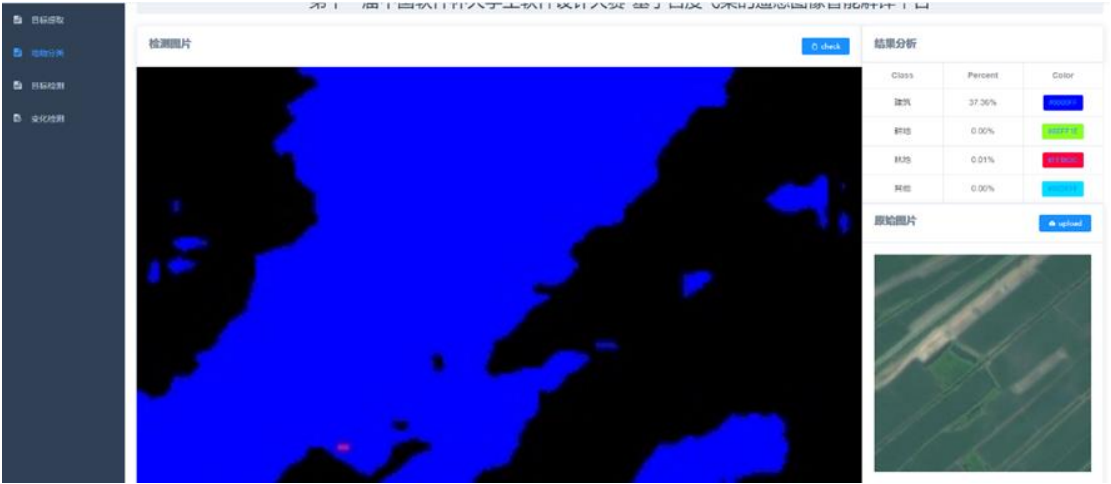


图 8 地物分类界面

3.3 主服务设计

主服务用于响应用户的操作命令、接收后台服务传来的数据以及将数据转发到对应的服务路由。

正如在 web 页面中所述，主服务基于 vue-element-admin 后台前端解决方案实现，去除了 vue-element-admin 中的冗余功能，根据任务要求进行了功能性的适配，构建快速、轻便的 web 服务。主服务任务流图可如图 9 所示：

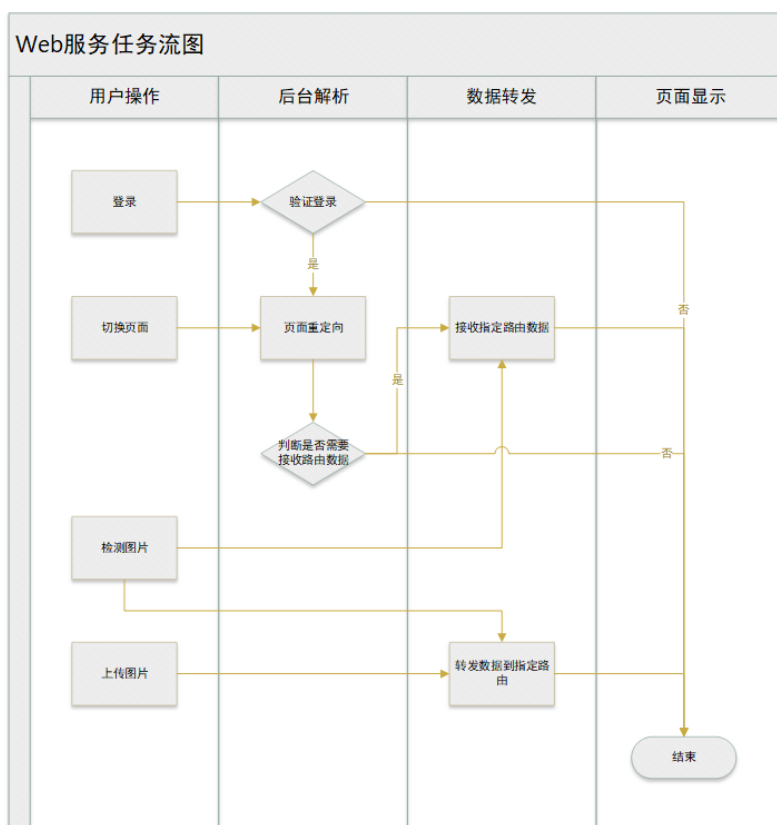


图 9 主服务任务流程图

3.4 数据存储服务设计

针对图像的存储、加载以及日志记录，我们使用 MySQL 对每个上传的图片进行了数据存储。

创建名为 updata 的数据库并创建 5 张 数据表为 changedetection1、changedetection2、objectdetection、targetextraction、terrainclassification，分别对应每个功能页面的图片上传域，保证图片上传时，同步记录到对应的数据表中。功能页面打开时，也会自动从数据库中读取最近一张历史图片。

各个数据表及其格式统一如下：

名	类型	长度	是否允许 null	是否为键
id	int	0	否	是
imgSrc	longtext	0	是	否

对应每张数据表，我们开通了存储、查询路由接口。数据表-接口对应如下：

数据表	图像存储路由接口	图像查询路由接口
changedetection1	post_storage_image	post_find_image1
changedetection2	post_storage_image2	post_find_image2
objectdetection	post_objectdetection_image	find_objectdetection_image

targetextraction	post_targetextraction_image	find_targetextraction_image
terrainclassification	post_terrainclassification_image	find_terrainclassification_image

3.5 图像解析服务设计

图像解析是整个项目最核心的功能,图像解析使用 PaddleRS 实现。为便于调用 PaddleRS,我们沿用了 Python 语言,并使用 Django 开发后台服务。

Django 本身基于 MVC 模型,即 Model (模型)+View (视图)+Controller (控制器)设计模式,MVC 模式使后续对程序的修改和扩展简化,并且使程序某一部分的重复利用成为可能。

Python 加 Django 快速开发、部署,搭建起一个图像解析服务,根据任务要求,我们开放了四个路由接口,功能-接口映射表如下:

功能名	路由接口
变化检测	cdrecvlmg
目标检测	odrecvlmg
目标提取	terecvlmg
地物分类	tcrecvlmg

四、环境部署

系统部署介绍以 Ubuntu / 20.04 LTS amd64 (64bit)为例。

4.1 环境配置

操作系统: Ubuntu / 20.04 LTS amd64 (64bit)

CPU: 2 核

内存: 8GB

Python: 3.8.10

4.1.1 安装 paddlepaddle

深度学习框架 paddlepaddle 的安装,飞桨官网介绍的很详细,无论 Linux 还是 Window 都可根据官网:

<https://www.paddlepaddle.org.cn/install/quick?docurl=/documentation/docs/zh/install/pip/windows-pip.html>, 安装所需版本。

注意, paddlepaddle、CUDA 以及 cuDNN 版本一定要对应, 否则会出现错误。

4.1.2 安装 Node.js

Ubuntu 下安装 Node.js 可通过 Node.js 的二进制发行版 <https://github.com/nodesource/distributions/blob/master/README.md> 安装。本次安装 Node.js v16.x, 相关安装命令如下:

<pre>curl -fsSL https://deb.nodesource.com/setup_16.x sudo -E bash -</pre>
<pre>sudo apt-get install -y nodejs</pre>

安装完毕后, 运行命令 `node -v` 查看 Node.js 的安装版本, 运行命令 `npm -v` 查看 npm 的安装版本。

4.1.3 安装 MySQL

Ubuntu 的镜像包中存有 MySQL APT Repository, 可以通过命令 `apt install mysql-server` 安装。使用命令 `netstat -tap | grep mysql`, 查看是否安装成功, 如果 mysql 的 socket 处于 LISTEN 状态则表示安装成功。

注意, 以上为 root 用户权限下的安装方法, 非 root 用户, 待测。

4.2 配置所有服务

4.2.1 克隆存储库

1. 使用命令 `git clone https://github.com/Shelly111111/CSC2022.git`, 将代码从云端克隆到本地;
2. 使用命令 `cd CSC2022` 切换到 CSC2022 目录下。

4.2.2 配置数据存储服务

代码克隆到本地好以后, 需要先在 MySQL 中建好数据存储服务所需的数据库以及数据表。

1. 使用命令 `cd CSC2022/sql`, 切换到 sql 目录下;

2. 切换到 root 下，使用命令 `mysql -u root -p` 并输入密码，登录 MySQL；
3. 使用命令 `show databases;`查看 MySQL 中的数据库；
4. 使用命令 `create database updata;`新建一个名为 updata 的数据库；
5. 使用命令 `use updata;`切换到 updata 数据库；
6. 使用命令 `show tables;`查看 updata 数据库下的所有表；
7. 分别使用以下命令导入所有表；

source changedetection1.sql
source changedetection2.sql
source objectdetection.sql
source targetextraction.sql
source terrainclassification.sql

8. 最后使用命令 `show tables;`查看所有表是否导入成功，若输出如下图所示，则证明导入成功。

```
mysql> show tables;
+-----+
| Tables_in_updata |
+-----+
| changedetection1 |
| changedetection2 |
| objectdetection  |
| targetextraction |
| terrainclassification |
+-----+
5 rows in set (0.00 sec)
```

9. 打开 `CSC2022/service/mysql/nodejs-orm/index.js` 文件，修改相关字典值为你的 MySQL 的配置。

```
2 // 数据库连接设置
3 let orm_config = {
4   host: 'localhost', //数据库地址
5   port: '3306',
6   user: 'root', //用户名，没有可不填
7   password: '', //密码，没有可不填
8   database: 'updata' //数据库名称
9 }
```

10. 使用命令 `cd ../service` 切换到 service 目录下。
11. 使用命令 `npm install` 安装依赖
12. 使用命令 `node app.js` 运行服务，运行结果如下图所示，则表示运行成功。

```
e:~/Downloads/CSC2022/service$ node app.js
Example app listening on port 3001!
```

4.2.3 配置图像解析服务

1. 在 CSC2022 目录下创建文件夹 data。
2. 由于模型太大，我们并未将模型上传到 GitHub 上，使用百度网盘链接：
<https://pan.baidu.com/s/1j3LmdBGRD836Fmeaa6lQmw> 提取码：b5t0，下载并解压到 data 目录下。

目录结构应为：

```
data/  
  -exp/  
  -od_exp/  
  -tc_exp/  
  -te_exp/
```

data 目录下应为四个文件夹分别是 exp、od_exp、tc_exp、te_exp。

3. 使用命令 `cd PaddleRS` 切换到 PaddleRS 目录下。
4. 使用命令 `pip3 install -r requirements.txt` 安装所有 python 第三方依赖包。
5. 使用命令 `python manage.py runserver 8000` 运行图像解析服务。出现如下图所示字样，证明运行成功。

```
June 27, 2022 - 10:25:48  
Django version 4.0.5, using settings 'testdj.settings'  
Starting development server at http://127.0.0.1:8000/  
Quit the server with CONTROL-C.
```

*注意：若您部署互联网服务，请使用命令 `python manage.py runserver` 您的内网 ip:8000 运行图像解析服务。

4.2.4 配置主服务

切换到 CSC2022 目录下。

1. 使用命令 `npm install` 安装依赖。
2. 运行 `npm run dev` 启动服务。出现如下图所示字样，证明运行成功。

```
You may use special comments to disable some warnings.  
Use // eslint-disable-next-line to ignore the next line.  
Use /* eslint-disable */ to ignore all warnings in a file.  
  
App running at:  
  - Local:   http://localhost:9527/  
  - Network: http://192.168.1.100:9527/  
  
request invoke:/dev-api/vue-element-admin/user/info
```

*注意：如果不是本地访问，请依次修改：

(1) CSC2022/src/views 下的 ChangeDetection、ObjectDetection、TargetExtraction、TerrainClassification 包下的 components/BoxCard.vue 中 uploadUrl 值为“你的公网 ip:端口/upload”。图示为变化检测代码 uploadUrl 所在位置。

```
125     data() {
126       return {
127         imageUrl: "",
128         imageUrl2: "",
129         imageUrl3: "",
130         dialog1Visible: false,
131         dialog2Visible: false,
132         fileList: [],
133         uploadUrl: "http://192.168.1.100:8000/upload",
134         base64: [],
135         percent: 0
136       };
137     },
```

(2) 修改 CSC2022/src/utils/request.js 中第 19 行 baseUrl 值为“你的公网 ip:3001”。

```
17 export const instance = axios.create({
18   timeout: 5000,
19   baseUrl: 'http://192.168.1.100:3001'
20 });
21
```

(3) 修改 CSC2022/src/api/update.js 中第 5 行 url 值为“你的公网 ip:8000”。

```
1 import { instance } from '../utils/request'
2
3 import axios from 'axios'
4
5 url = 'http://192.168.1.100:8000'
6
7 export function sendImage(imgSrc1, imgSrc2) {
8   var data = new URLSearchParams();
9   data.append('img1', imgSrc1);
10  data.append('img2', imgSrc2);
11  return axios.post(url + '/cdrecvImg', data)
12 }
13
```

五、算法设计

5.1 变化检测

5.1.1 任务背景

相对于传统基于人海战术的目视解译方法,基于深度学习的遥感图像识别技术可以自动分析图像中的地物类型,在准确率和效率方面展现出极大的潜力。此次赛题变化检测部分要求参赛者利用提供的训练数据,实现对多时相图像中的建筑变化检测。具体而言,多时相遥感图像建筑物变化检测任务是给定两张不同时间拍摄的相同位置(地理配准)的遥感图像,要求定位出其中建筑变化的区域。

5.1.2 数据集介绍

本数据集包含赛事的变化检测任务所需训练(training)和验证(evaluation)数据,以及测试(testing)影像对。训练/验证集,其中共含有 637 个遥感影像对以及对应的二值变化标签,每个时相的影像均包含 RGB 三个波段(亮度值已量化到 0-255),长、宽均为 1024。测试集含有 363 个遥感影像对,但不包含对应的二值变化标签。

数据集链接: <https://aistudio.baidu.com/aistudio/datasetdetail/134796>。

训练图样例如下:

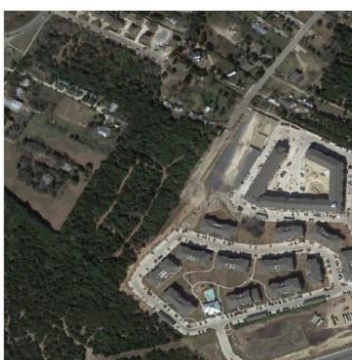
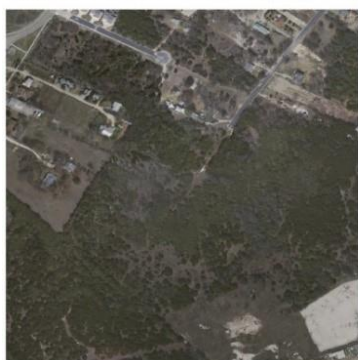
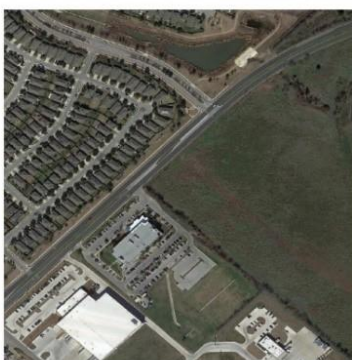
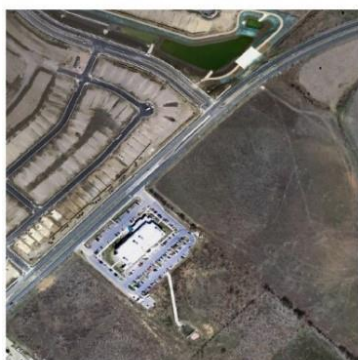
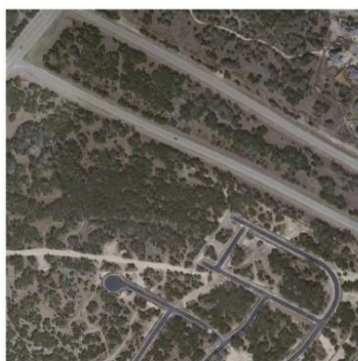
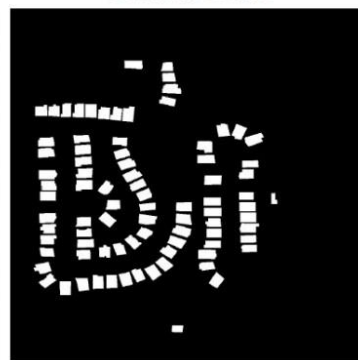
Image 1



Image 2



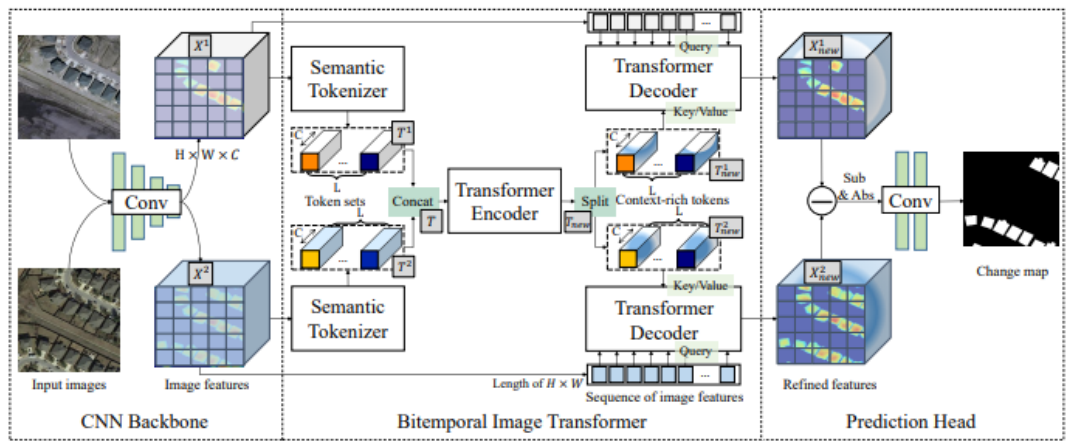
Ground Truth



5.1.3 算法设计

1. 模型架构

BIT-CD 针对时空中长距离信息难以联系的问题引入了位时图像变换器，有效的对位时图像中的长范围背景进行建模。此外，BIT-CD 还将 BIT 纳入一个基于深度特征差分的 CD 框架中，以更好地模拟上下文。



2. 数据划分和增强

本团队对训练的数据进行了按照 19: 1 的比例进行随机划分，划分为训练集和验证集，使用 PaddleRS 内置 transforms 对数据进行动态增强和预处理，以增强模型的泛化能力，包括训练时 RandomCrop、RandomHorizontalFlip、RandomVerticalFlip 和 Normalize，验证时 Normalize。

3. 学习率下降策略

本团队使用 Adam 优化器优化学习率，下降策略使用 0.95 的 $\text{step}/\text{DECAY_STEP}$ 次方作为学习率的系数。DECAY_STEP 设置为 1000。

4. 模型训练与预测

本团队训练了 1000 次，每 10 个 epoch 保存一次模型权重并进行验证，验证的最优模型在验证集上的 mIOU 为 0.91。

使用训练时保存的最优模型进行预测，本团队使用了 0.5 的阈值对预测图进行二值划分，划分后的预测图在测试集上的得分和 F1 值为 0.85。本团队尝试了大津法对图像进行二值阈值分割，发现结果并不理想。

考虑由于训练时使用随机翻转进行了训练增强以提高模型的泛化能力，本团队在测试时对测试集分别进行不翻转、水平翻转、垂直翻转、水平且垂直翻转的测试时增强，并对四种输出进行评价的加权投票，最终得分和 F1 为 0.855。

5. 小结

我们团队也是首次接触遥感图像变化检测，它与遥感图像分割有着相似之处，但也有很明显的不同。其除了要对图像中的像素进行语义分割，还要考虑两幅图片中位时的差异，BIT 很好得对此进行了探索。

我们参考了 BIT 模型并在其之上进行了尝试，但并没有过多的改动其模型网络本身，后续将会在此进行改进，也会更多的学习相应的知识。

5.2 目标提取

5.2.1 任务背景

从遥感影像图片中提取的信息已在广泛的领域得到应用，包括城市规划、作物和森林管理、救灾和气候建模。目前，大部分提取仍由人类专家执行，这使得该过程缓慢、成本高且容易出错。

此次赛题目标提取部分要求参赛者利用提供的训练数据，实现从遥感影像图片中分割道路。来自附近树木的障碍物、相邻建筑物的阴影、道路纹理和颜色的变化、道路等级的不平衡（由于道路图像像素相对较少）是阻碍当前模型分割从图像一端延伸到另一端的尖锐道路边界的困难和挑战。

5.2.2 数据集介绍

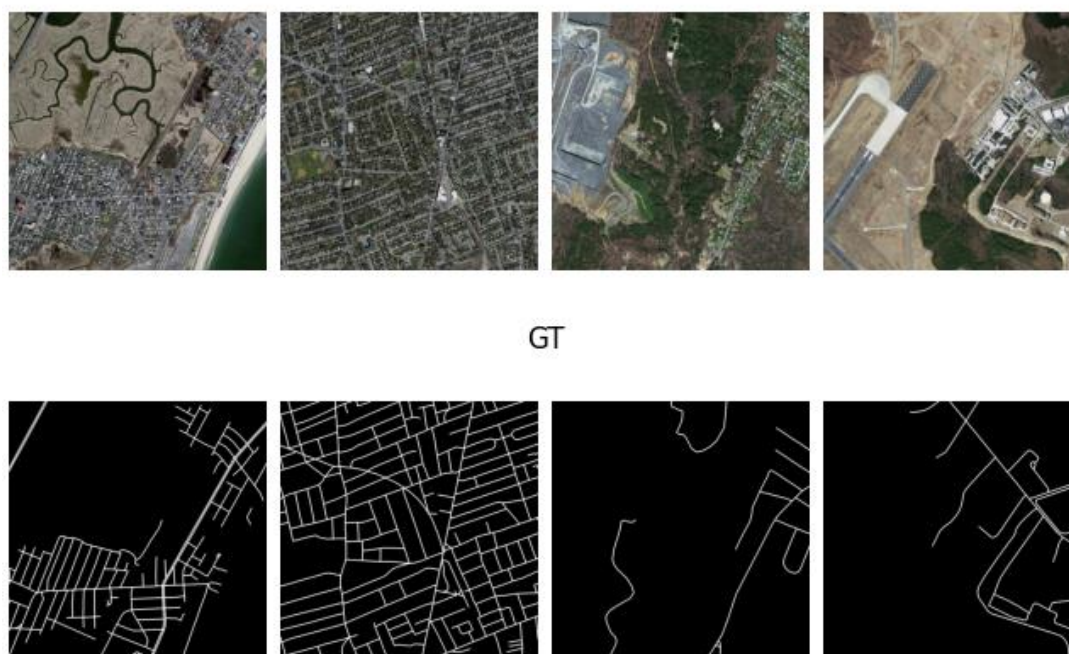
使用 Massachusetts Roads 道路提取数据集[1]。该数据集的训练集包含 804 幅大小为 1500x1500 的三波段影像，每幅影像均配有道路提取标签。

数据集链接：

<https://www.cs.toronto.edu/~vmnih/data/>

<https://aistudio.baidu.com/aistudio/datasetdetail/56961>

训练图样例如下：

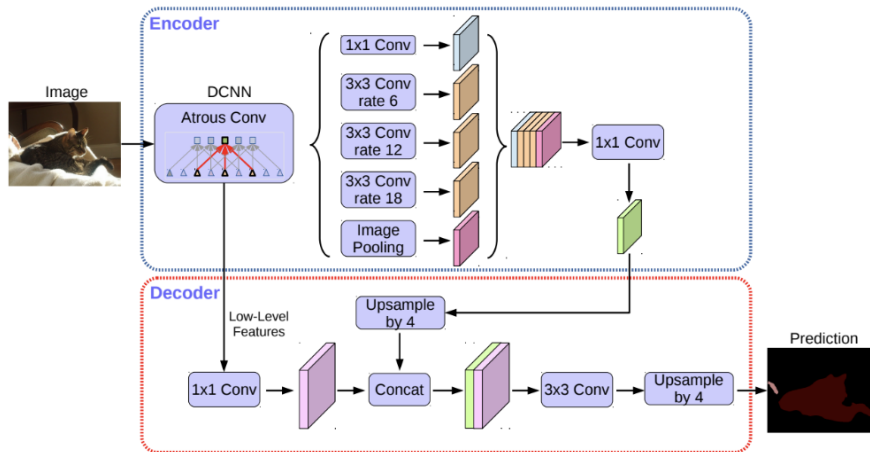


5.2.3 算法设计

我们目标提取算法使用的是官方发布的 baseline 项目（[【官方】第十一届“中国软件杯”百度遥感赛项：目标提取功能](#)）。运行 100epoch 后，最好的模型在验证集上 miou 为 0.77，测试集上指标：IoU 为 0.57，Acc 为 0.79，Kappa 系数为 0.71, F1 为 0.73。

1.模型架构

语义分割任务通常会用到空间金字塔模块和编解码模块。前者有利于提取多尺度上下文信息，后者更容易捕获边缘信息。deeplabv3+通过增加了简单并且高效的编码模块来改进分割结果。Backbone 我们选用了 ResNet50_vd。



2.数据划分和增强

我们对训练的数据进行了按照 9: 1 的比例进行随机划分，划分为训练集和验证集。

我们使用 PaddleRS 内置 transforms 对数据进行动态增强和预处理，以增强模型的泛化能力，包括训练时 RandomCrop、RandomHorizontalFlip、RandomVerticalFlip 和 Normalize，验证时 Normalize。

3.学习率下降策略

我们使用 Adam 优化器优化学习率，学习率设置为 0.001。

4.模型训练与预测

我们训练了 100 轮，每 1 个 epoch 保存一次模型权重并进行验证，验证的最优模型在验证集上的 mIOU 为 0.77。

使用训练时保存的最优模型进行预测，我们使用与验证集同样的 transforms，来让预测更贴近模型训练时的参数。

最终在测试集上的指标为：IoU 为 0.57，Acc 为 0.79，Kappa 系数为 0.71, F1 为 0.73

5.小结

由于时间关系，我们团队并未在目标提取上过多尝试，所以，模型泛化能力应该有很大的提升空间，可以尝试调整学习率策略、使用更多的图像增强策略等方式，增强模型拟合和泛化能力。

5.3 目标检测

5.3.1 任务背景

目前，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的需求也与日俱增。传统方式对高分辨率卫星遥感图像的对特征刻画能力差且依赖人工经验工作量巨大。随着人工智能技术的兴起，特别是基于深度学习的图像识别方法获得了极大的发展，相关技术也推动了遥感领域的变革。相对于传统基于人海战术的目视解译方法，基于深度学习的遥感图像识别技术可以自动分析图像中的地物类型，在准确率和效率方面展现出极大的潜力。此次赛题目标检测部分要求参赛者使用目标检测技术对卫星图像中指定对象完成检测。

5.3.2 数据集介绍

本项目使用 RSOD 遥感影像目标检测数据集。RSOD 数据集由武汉大学于 2017 年发布，包含 4 种类型的遥感地物目标（本项目仅使用其中的 playground 类）。影像提取自 Google Earth、天地图等平台。

数据集包括飞机、油罐、操场和立交桥。此数据集的格式为 PASCAL VOC，数据集包括 4 个文件，每个文件对应一种对象。飞机数据集，446 张图片中的 4993 架飞机；操场，189 张图片中的 191 个操场；立交桥，176 张图片中的 180 座立交桥；油罐，165 张图片中的 1586 个油罐。

本次数据集只抽取了操场进行训练。

数据集链接：<https://aistudio.baidu.com/aistudio/datasetdetail/52980>

Test Results
Image



Detection



5.3.3 算法设计过程

1.模型架构

对于一个目标检测算法而言,通常可以将其划分为 4 个通用的模块,具体包括:输入端、基准网络、Neck 网络与 Head 输出端,PPYOLO 骨架—YOLO 骨架是一个卷积神经网络,它将图像像素合并以形成不同粒度的特征。主干通常在分类数据集(通常为 ImageNet)上进行预训练。PPYOLO 脖子—YOLO 脖子(上面选择了 FPN)在传递到预测头之前对 ConvNet 图层表示进行组合和混合。PPYOLO 头部—这是网络中进行边界框和类预测的部分。它由关于类,框和对象的三个 YOLO 损失函数指导。

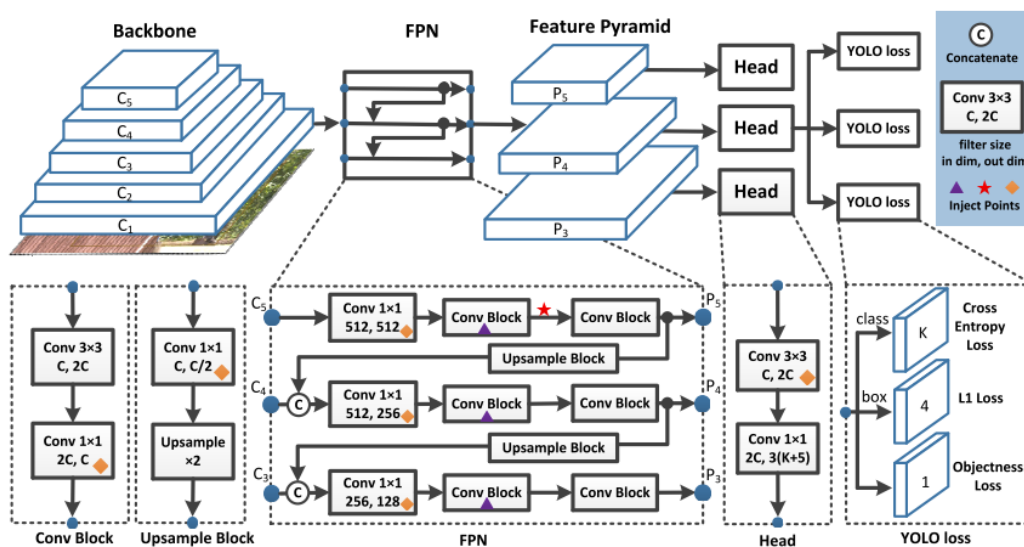


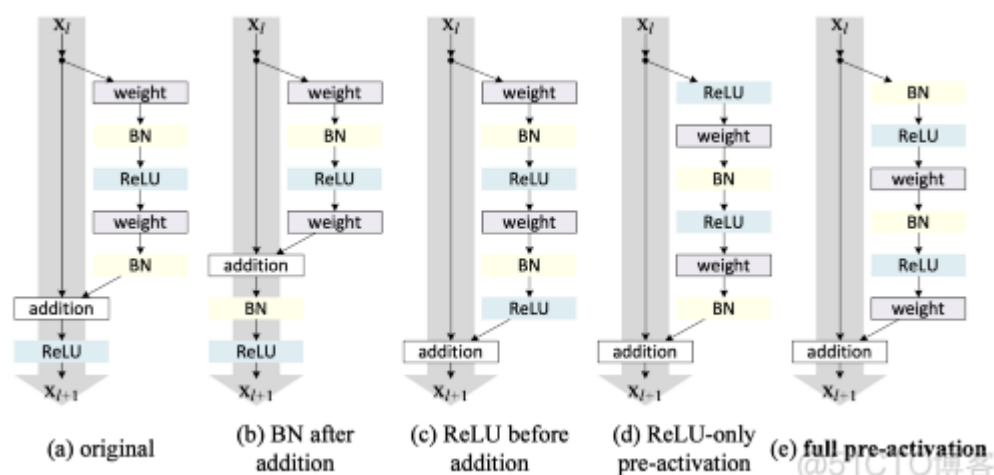
Figure 2. The network architecture of YOLOv3 and inject points for PP-YOLO. Activation layers are omitted for brevity. Details are described in Section 3.1 and Section 3.2.

https://blog.csdn.net/qq_41375609

2. 算法设计

1. 第一种 PP YOLO 技术

运用 Resnet50-vd-dcn ConvNet 主干替换 YOLOv3 Darknet53 主干。Resnet 是一个更流行的主干，它的执行优化了更多的框架，并且其参数少于 Darknet53。交换此主干可以看到 mAP 的改进，这对 PP YOLO 来说是一个巨大的胜利。



2. 模型参数的 EMA

PP YOLO 跟踪网络参数的指数移动平均值，以维持预测时间的模型权重的阴影。已经证明这可以提高推理精度。

3.批量更大

PP-YOLO 的批量大小从 64 增加到 192。当然，如果您有 GPU 内存限制，这很难实现。

4.DropBlock 正则化

PPYOLO 在 FPN 脖子上实现 DropBlock 正则化（过去，这通常发生在主干中）。DropBlock 在网络中的给定步骤中随机删除了一部分训练功能，以指示模型不依赖于关键特征进行检测。

5.小结

在训练过程中我们不断进行尝试，但并没有过多的改动其模型网络本身，后续将会在此进行改进，也会更多的学习相应的知识。

5.4 地物分类

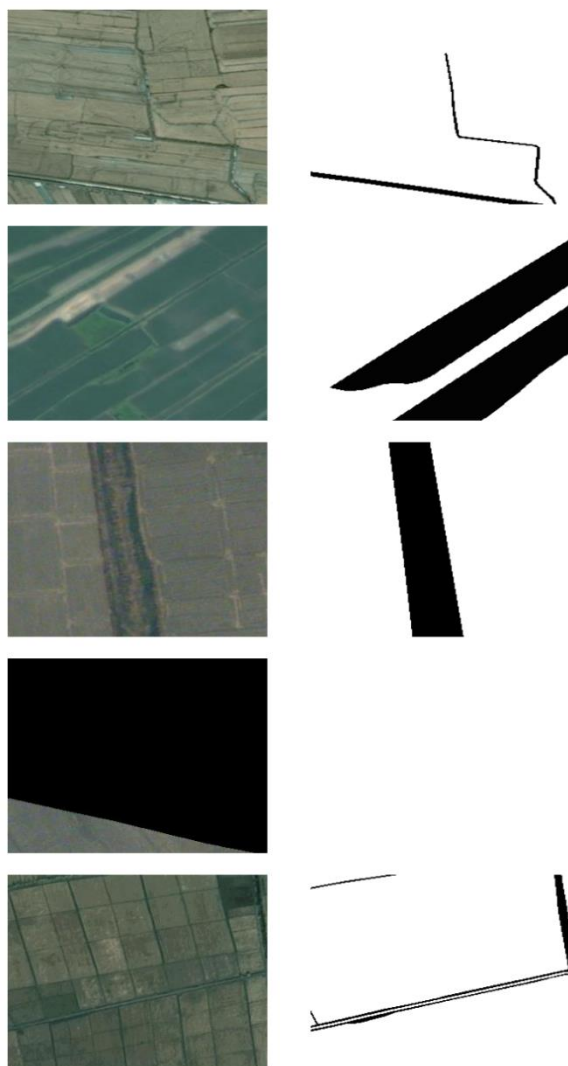
5.4.1 任务背景

地物分类，旨在对遥感影像进行像素级内容解析，对遥感影像中感兴趣的类别进行提取和分类，在城乡规划、防汛救灾等领域具有很高的实用价值，在工业界也受到了广泛关注。现有的遥感影像地块分割数据处理方法局限于特定的场景和特定的数据来源，且精度无法满足需求。因此在实际应用中，仍然大量依赖于人工处理，需要消耗大量的人力、物力、财力。本赛题旨在衡量遥感影像地块分割模型在多个类别（如建筑、道路、林地等）上的效果，利用人工智能技术，对多来源、多场景的异构遥感影像数据进行充分挖掘，打造高效、实用的算法，提高遥感影像的分析提取能力。

5.4.2 数据集介绍

地物分类数据集包含训练图像和标签，其中包含 66652 张原始图像和 66652 张标签图像。原始图像每张图片分辨率为 2m/pixel，尺寸为 256 * 256，标签图像全部 PNG 图片共包括 4 种分类，像素值分别为 0、1、2、3，此外，像素值 255 为未标注区域，表示对应区域的所属类别并不确定。

数据集链接: <https://aistudio.baidu.com/aistudio/datasetdetail/77571>



5.4.3 算法设计

1. OCRNet 模型结构

OCR 方法的主要思想是显式地把像素分类问题转化成物体区域分类问题，这与语义分割问题的原始定义是一致的，即每一个像素的类别就是该像素属于的物体的类别

OCRNet 方法的实现主要包括 3 个阶段：

(1) 从骨干网络得到特征表示，并估测一个简单粗略的语义分割结果作为 OCR 方法的一个输入，即软物体区域（Soft Object Regions），矩阵表示为 $b \times c \times h \times w$ ；

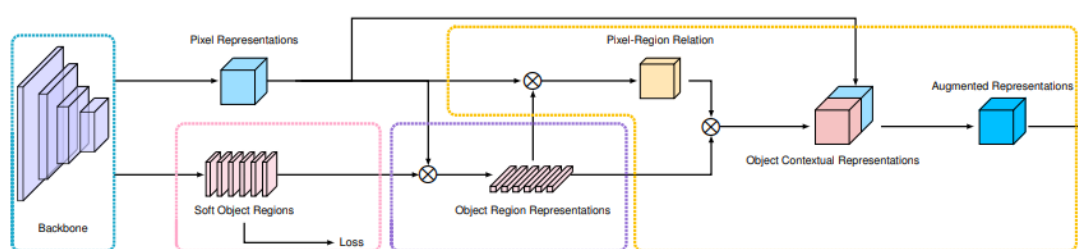
(2) 根据软物体区域 ($b \times c \times h \times w$) 和网络最深层输入的特征表示 ($b \times c \times h \times w$) 表示计算出 K 组向量, 即物体区域表示 (Object Region Representations), 其中每一个向量对应一个语义类别的特征表示, 矩阵表示为 $b \times c \times k \times l$

(3) 计算网络最深层输出的像素特征表示 (Pixel Representations) 与计算得到的物体区域特征表示 (Object Region Representation) 之间的关系矩阵 $b \times (h \times w) \times k$, 然后根据每个像素和物体区域特征表示在关系矩阵中的数值把物体区域特征加权求和, 得到最后的物体上下文特征表示 OCR (Object Contextual Representation), 矩阵表示为 $b \times k \times h \times w$ 。

当把物体上下文特征表示 OCR ($b \times k \times h \times w$) 与网络最深层输入的特征表示 ($b \times k \times h \times w$) 拼接之后作为上下文信息增强的特征表示 (Augmented Representation) ($b \times 2k \times h \times w$), 可以基于增强后的特征表示预测每个像素的语义类别, 也可拼接上 ASPP 特征, 即 OCR+Features+ASPP

OCRNet 可计算一组物体区域的特征表达, 然后根据物体区域特征表示与像素特征表示之间的相似度将这些物体区域特征表示传播给每一个像素。

Segmentation Transformer: OCR for Semantic Segmentation



2.数据划分和增强

我们将数据集的 97%划分为训练集, 3%划分为验证集, 使用 PaddleSeg 的 api 进行数据增强, 训练时包括 ResizeStepScaling、RandomPaddingCrop、RandomHorizontalFlip、Normalize 等, 测试时包括 Normalize 操作。

3.学习率下降策略

优化器选用 SGD 随机梯度下降法, 衰减策略使用 0.9power 的 poly 下降策略。

4.模型训练与预测

模型 `iters` 设置为 1600000 次，每 10000 次保存一下模型权重并进行验证，验证的最优效果 `mIOU` 为我们训练了 1000 次，每 10 个 `epoch` 保存一次模型权重并进行验证，验证的最优模型在验证集上的 `mIoU` 为 0.4681。

模型使用训练时保存的最优模型进行预测分割，在测试集的 `mIoU` 为 0.58838，`score` 为 58.83834。

5.小结

基于 `PaddleSeg` 进行地物分类语义分割任务，完成了数据准备、模型训练和预测的流程。

在数据集的分析上，我们从单幅图像本身进行数据增强，训练过程中考虑类别分布不均等特征。

在模型选择和构建上，我们选用 `OCRNet` 作为地物分类的分割模型，其引入注意力机制能缓解异物同谱对地块分割的负面影响，对于分割效果有提升。由于地物分类具有尺度效应，即不同地物的最佳分割尺度不同，模型之后会考虑将尺度因素引入模型中去以提高效果。

六、项目创意及特色

1、支持多用户使用。本项目可以满足企业级用户的需求规模，可多用户同时访问使用，每个用户都可以同时上传图片进行图片检测与翻译；

2、充分进行功能优化，在完成用户的四大基本要求的基础上，添加附加功能对于图片的变化情况进行深入分析以提升用户的体验感，在“目标提取”、“目标检测”、“变化检测”功能中，对图像检测的同时，依次显示道路所占比例、检测目标所占比例以及变化范围，在“地物分类”分类功能中，将目标划分为“建筑”、“耕地”、“林地”、“其他”，并且使用不同的颜色进行标记，充分展示分析结果。

3、界面设计简洁、用户交互性强。本项目的界面设计清爽明了，功能齐全且目的明确，提示信息和注释为用户保驾护航，用户易上手操作。

4、在安全方面，为企业用户的信息安全着想，增加相应的安全机制。例如用户登录认

证，日志记录等。

七、系统测评

基于百度飞桨的遥感图像智能解译平台测试前后超过 1 个月的时间，系统总共发布 13 个测试版本，其中 B1-B7 为本地迭代版本，B8-B10 为云服务器测试版本。B10-B13 计划外回归测试版本。

主要测试回顾如下：

表 7-1 测试回顾表

版本	主要内容	时间
B4	测试图片导入功能的可用性	5月14日
B5	测试变化检测相关功能	6月15日
B6	测试其他检测功能	6月19日
B8	系统兼容性测试	6月26日
B10	系统疲劳强度测试	6月30日
B11	用户操作安全测试	6月30日

主要测试表现如下：

表 7-2 技术测试表现表

技术点	测试表现
响应时间	平均响应时间为 90ms
服务器响应时间	平均响应时间为 600ms
部署方便性	部署方式：局域网+自建服务器+公网 IP；支持 Linux、macOS、Windows7 及以上操作系统

系统测试表现如下：

易用性	前端页面风格一致 系统界面组件提示完备，方便指导用户使用
可靠性	在用户错误操作时系统给出相应的提示信息

可维护性	提供详尽的开发文档 系统支持二次开发 系统各模块操作均记录日志
安全性	提供账户密码登录功能 系统权限设置合理

八、总结

人工智能已经在许多领域得到快速发展和广泛应用,在遥感影像自动解译方面也取得重要研究进展,并有部分场景得到应用。但是,遥感影像智能解译的规模化业务化应用还不够成熟,目前遥感智能解译存在的问题,包括功能不完善、缺乏标准规范等。通过人工智能深度学习,本团队设计并开发的基于百度飞桨的遥感图像智能解译平台,包括变化检测、目标提取、目标检测、地物分类四大功能,经过不断地模型优化调参,各项任务的精准度都得到显著提高,同时结合 `Vue+paddlers+django` 前后端开发,在网页客户端实现对于遥感图像的分类、检索、辨识等工作。

除完成基本遥感图像解释,本团队开发的系统还具备用户登录与权限管理,简洁清晰的界面,在得到检测结果后提供具体数据回溯功能,具备实用性与丰富性,为用户提供便捷服务的同时兼顾安全性与可靠性。

整个遥感图像智能解译系统,具有精准度高、安全性强、可维护性好、功能完备的特点,可广泛应用于智慧城市、气象预测、环境保护和防灾减灾、农林业监测等场景的重要研究领域,促进产业生产,具有广泛的应用价值与商业潜力。

参考引用

[1] Hao Chen, Zipeng Qi, and Zhenwei Shi. Remote Sensing Image Change Detection with Transformers. IEEE Transactions on Geoscience and Remote Sensing.

[2] Z Xiao, Q Liu, G Tang, X Zhai, "Elliptic Fourier transformation-based histograms of oriented gradients for rotationally invariant object detection in remote-sensing images", International Journal of Remote Sensing, vol. 36, no. 2, 2015

[3] Mnih, V. (2013). Machine Learning for Aerial Image Labeling [Phdthesis]. University of Toronto.

附录

为便于用户体验和版本升级迭代，我们秉持开源共享精神，将代码开源至 GitHub。如在操作中遇到问题，欢迎提问 Issue，我们将会根据您的 Issue 更新我们的版本，您的问题就是对我们的贡献，您的喜欢就是对我们作品最好的回报。

GitHub 库链接: <https://github.com/Shelly111111/CSC2022>

在线 Demo: <http://120.48.87.7:9527/>

致谢

感谢 PaddleRS 团队开源的代码

感谢 vue-element-admin 团队开源的代码

感谢 Django 团队开源的代码