

# 说明文档

作者：思绪无限

公众号: AI技术与分享

CSDN: <https://wuxian.blog.csdn.net>

BiliBili: <https://space.bilibili.com/456667721>

博客园: <https://www.cnblogs.com/sixuwuxian/>

知乎: <https://www.zhihu.com/people/sixuwuxian>

原创代码，仅用于学习交流，  
受版权保护，禁止他人商用和盗卖，盗版追究法律责任！

.idea  
\_\_pycache\_\_  
datasets  
icons  
runs  
test\_media  
themes  
ultralytics  
weights  
\_\_init\_\_.py  
LoginForm.py  
LoginForm.ui  
LoginWindow.py  
Recognition\_UI.py  
Recognition\_UI.ui  
RecSystem.py  
RecSystem.qrc  
requirements.txt  
run\_main\_login.py  
run\_main\_noLogin.py  
run\_test\_camera.py  
run\_test\_image.py  
run\_test\_video.py  
run\_train\_model.py  
System\_login.py  
System\_noLogin.py  
UserDatabase.db  
YOLOv8v5Model.py  
环境配置.txt  
说明文档.docx

◆ 功能演示

◆ 软件安装

◆ 环境配置

◆ 项目介绍

◆ 修改项目

思绪无限

衣

(  
视

对  
贴

或  
“



# 环境配置

## 项目环境配置教程

见本人CSDN博客  
(**思绪无限**)、B站  
视频 (**思绪亦无限**)

对应资源博客下也  
贴有教程链接!

或者到主页搜索  
“环境配置”



思绪无限

码龄8年

3,454,855 总访问量 | 81 原创 | 73,285 排名 | 57,841 粉丝 | 1,512 铁粉 | 学习成就

个人简介: 人工智能博主, 机器学习及机器视觉爱好者, 专注专业知识整理与项目总结。

IP 属地: 上海市

查看详细资料

原力等级

成就 >

当前等级	当前总分	当月
9	10,050	44

2023博客之星TOP 20 榜单公布

查看榜单

个人成就

博客专家认证

获得 5,876 次点赞

内容获得 3,512 次评论

获得 30,922 次收藏

代码片获得 19,077 次分享

最近 | 文章 82 | 资源 15 | 问答 1 | 帖子 47 | 视频 | 课程 | 关注/订阅/互动 | 收藏

搜TA的内容

基于深度学习的人脸情感识别代码

发布资源 2023.06.28 · zip

基于OpenCV+DNN的灰度图像自动彩色化代码 (含Python代码+模型文件)

发布资源 2023.06.28 · zip

基于YOLOv5+DeepSort的车辆跟踪完整代码

发布资源 2023.06.28 · zip



SRGAN图像超分重建算法Python实现 (含数据集代码)

本文介绍深度学习的SRGAN图像超分重建算法, 使用Python以及Pytorch框架实现, 包含完整训练、测试代码, 以及训练数据集文件。博文介绍图像超分算法的原理, 包括生成对抗网络和SRGAN模型原理和实现的代码, 同时结合具体...

原创 发布博客 2023.06.27 · 8168 阅读 · 8 点赞 · 3 评论 · 47 收藏

SRGAN图像超分重建算法Python实现完整资源 (含数据集代码)

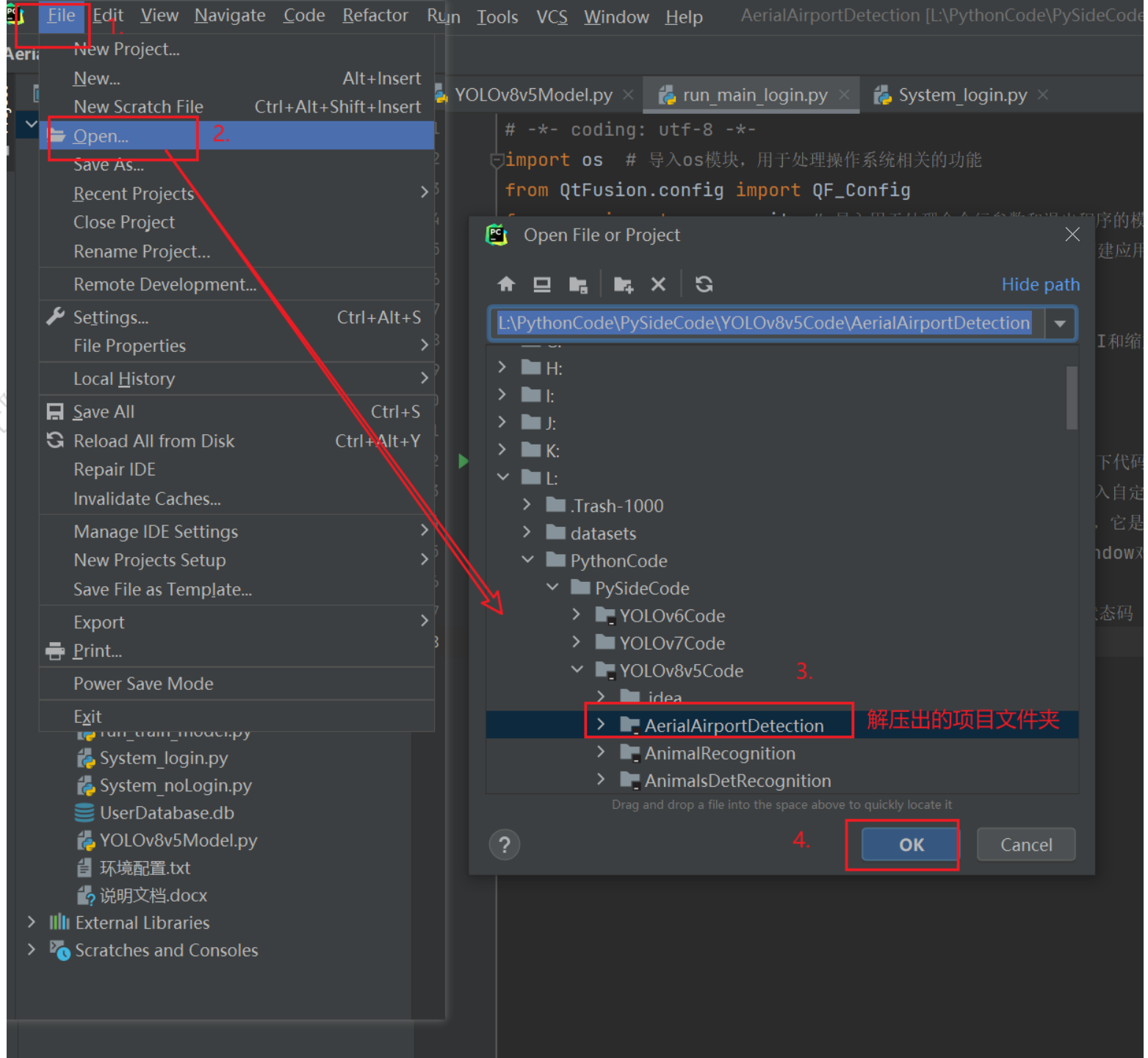


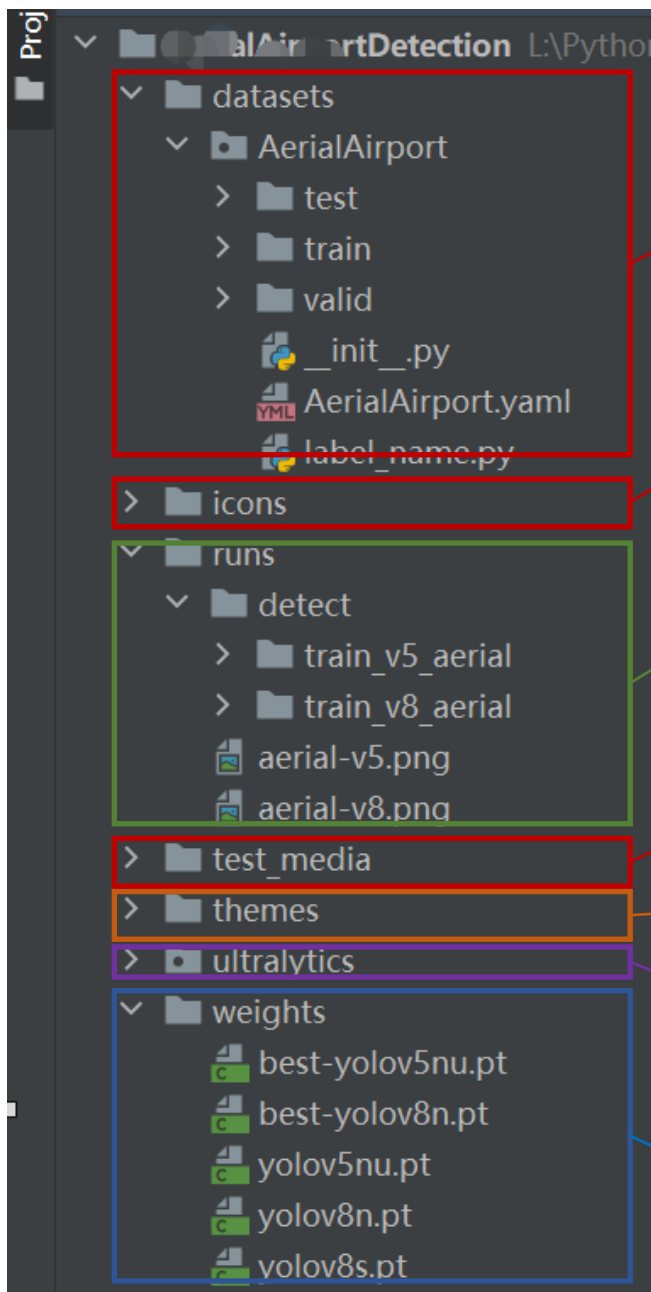
# 项目介绍

在安装好pycharm后打开项目

如右图，依次打开项目文件夹

(注意：解压出的项目文件夹有时可能嵌套了一个同名文件夹，打开项目时选择最里面一层文件夹)





**数据集文件夹**，用于模型训练和评估的数据集，包含训练、测试、验证集图片和标注文件，可在此文件夹中添加自定义数据集进行训练（保持文件夹目录结构一致）

**图标文件夹**，存放用于项目界面中的图标文件，包含界面中所需的背景、图标文件，可自行替换修改

**训练结果文件夹**，用于存储模型训练过程中生成的日志、训练结果、评估图表，包含不同模型训练后的pt模型文件、训练结果、图表文件，可查看训练和评估的结果；

**测试图片和视频文件夹**，用于测试模型性能的媒体文件，包含用于测试的图片、视频文件；

**样式文件夹**，于用户界面美化的主题文件，包含界面样式表、图标文字设置的yaml文件；

**YOLO的包**，官方代码包，Ultralytics YOLO模型有关的脚本或模块，包含模型定义、配置、工具等文件；

**模型文件夹**，存储训练好的模型权重文件，包含预训练模型pt文件，以及在本项目数据集训练好的模型pt文件；

> datasets  
> icons  
> runs  
> test\_media  
> themes  
> ultralytics  
> weights

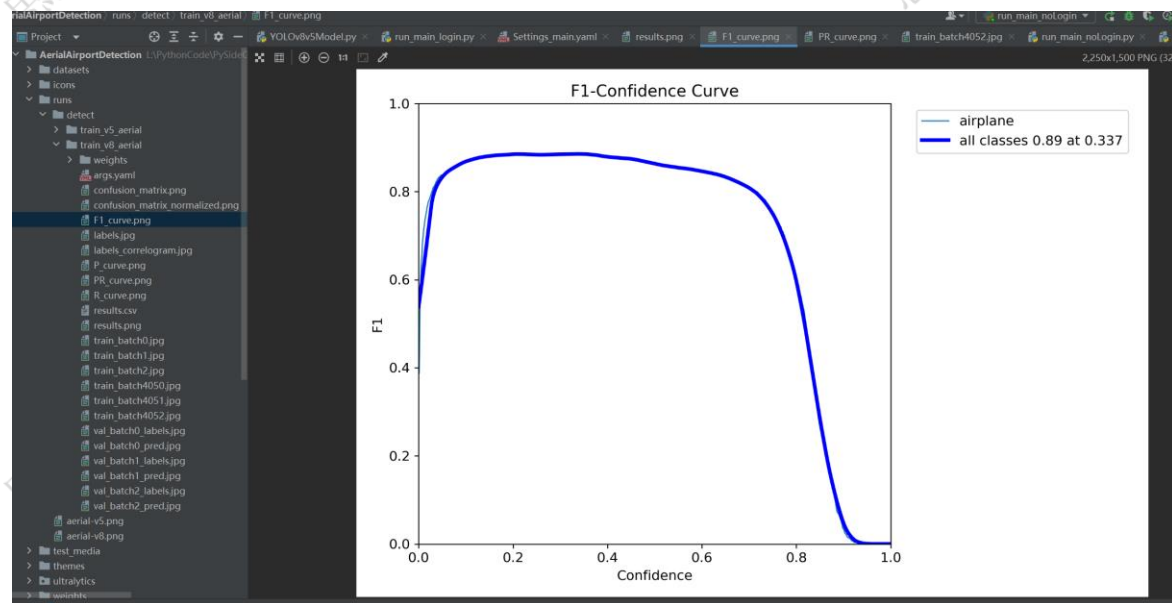
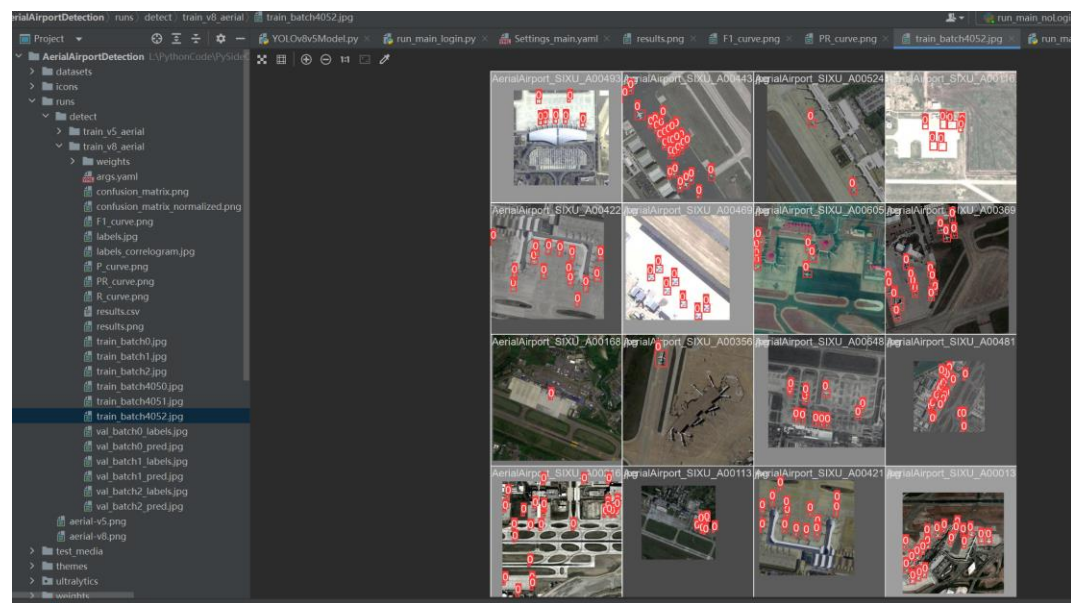
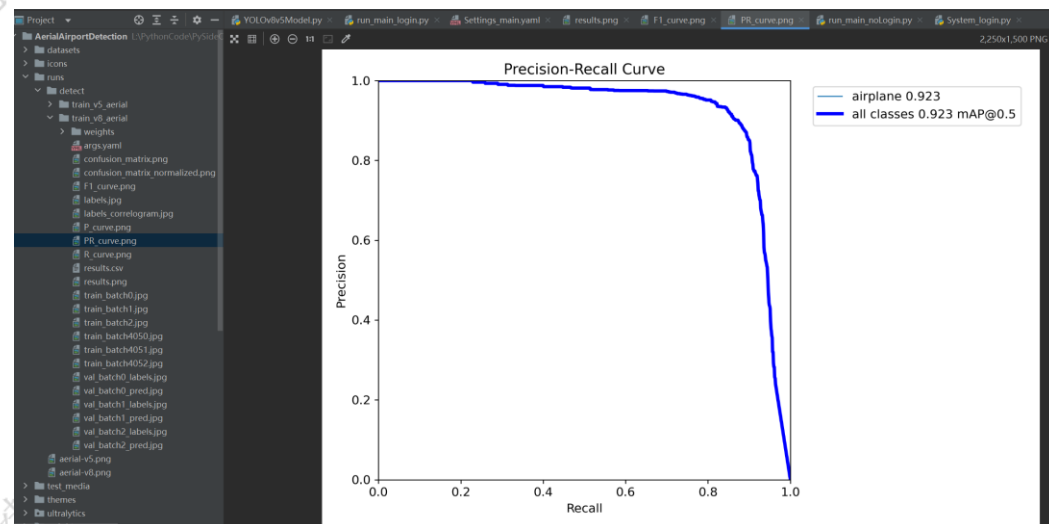
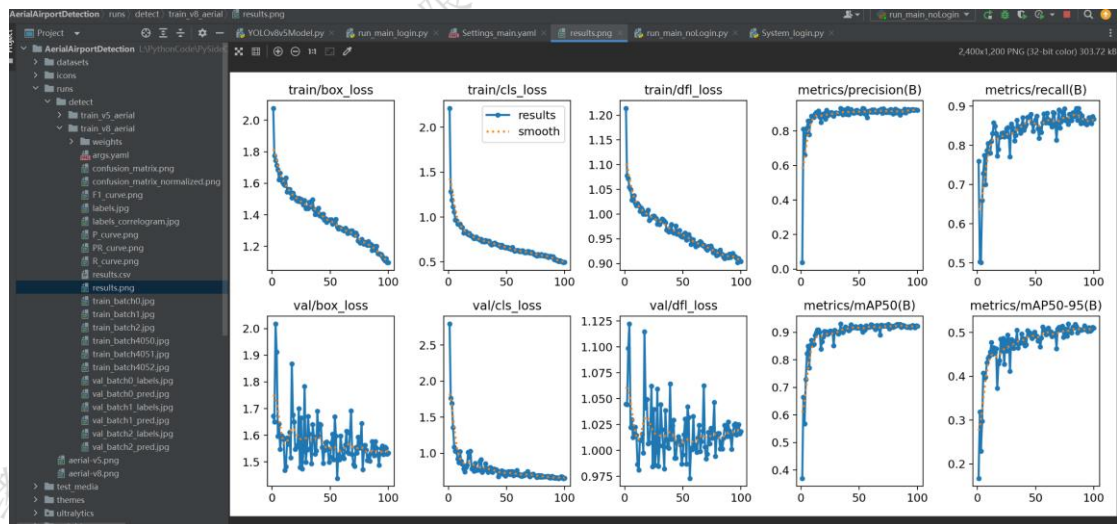
• \_\_init\_\_.py  
• LoginForm.py  
• LoginForm.ui  
• LoginWindow.py  
• Recognition\_UI.py  
• Recognition\_UI.ui  
• RecSystem.py  
• RecSystem.qrc  
• requirements.txt  
• run\_main\_login.py  
• run\_main\_noLogin.py  
• run\_test\_camera.py  
• run\_test\_image.py  
• run\_test\_video.py  
• run\_train\_model.py  
• System\_login.py  
• System\_noLogin.py  
• UserDatabase.db  
• YOLOv8v5Model.py  
• 环境配置.txt  
• 说明文档.docx

- \_\_init\_\_.py - Python包初始化文件，允许Python知道该目录可以被视为一个包；
- LoginForm.py - 登录界面布局的py代码，包含用户登录界面创建、表单逻辑；
- LoginForm.ui - 用户登录界面的UI布局文件，用Qt Designer创建，可以重新编辑界面布局
- LoginWindow.py - 实现登录窗口逻辑的Python脚本；
- Recognition\_UI.py - 主界面的布局py代码，包含本项目主界面的创建、生成、布局逻辑
- Recognition\_UI.ui - 本项目主界面布局的ui文件。
- RecSystem.py - 资源集合文件的py文件，包含用于Qt应用程序的资源。
- RecSystem.qrc - 资源集合文件，用于Qt应用程序，管理应用程序资源。
- requirements.txt - 包含项目依赖项及其版本的文本文件，用于设置开发环境。
- **run\_main\_login.py** - 启动登录功能的主运行脚本，登录注册后进入主界面。
- **run\_main\_noLogin.py** - 启动不带登录界面功能的主运行脚本，直接进入主界面。
- **run\_test\_camera.py** - 用于测试摄像头输入的脚本，不带界面。
- **run\_test\_image.py** - 用于测试图像识别功能的脚本，不带界面。
- **run\_test\_video.py** - 用于测试视频流识别功能的脚本，不带界面。
- **run\_train\_model.py** - 启动模型训练过程的脚本，可重新训练。
- System\_login.py - 包含登录系统后端逻辑的脚本。
- UserDatabase.db - 用户信息数据库文件，用于认证和授权，存储用户的账号和密码。
- YOLOv8v5Model.py - 包含YOLO模型相关代码的Python脚本。
- 环境配置.txt - 包含环境配置说明的文本文件。
- 说明文档 - 项目的详细说明文档。



# 项目介绍

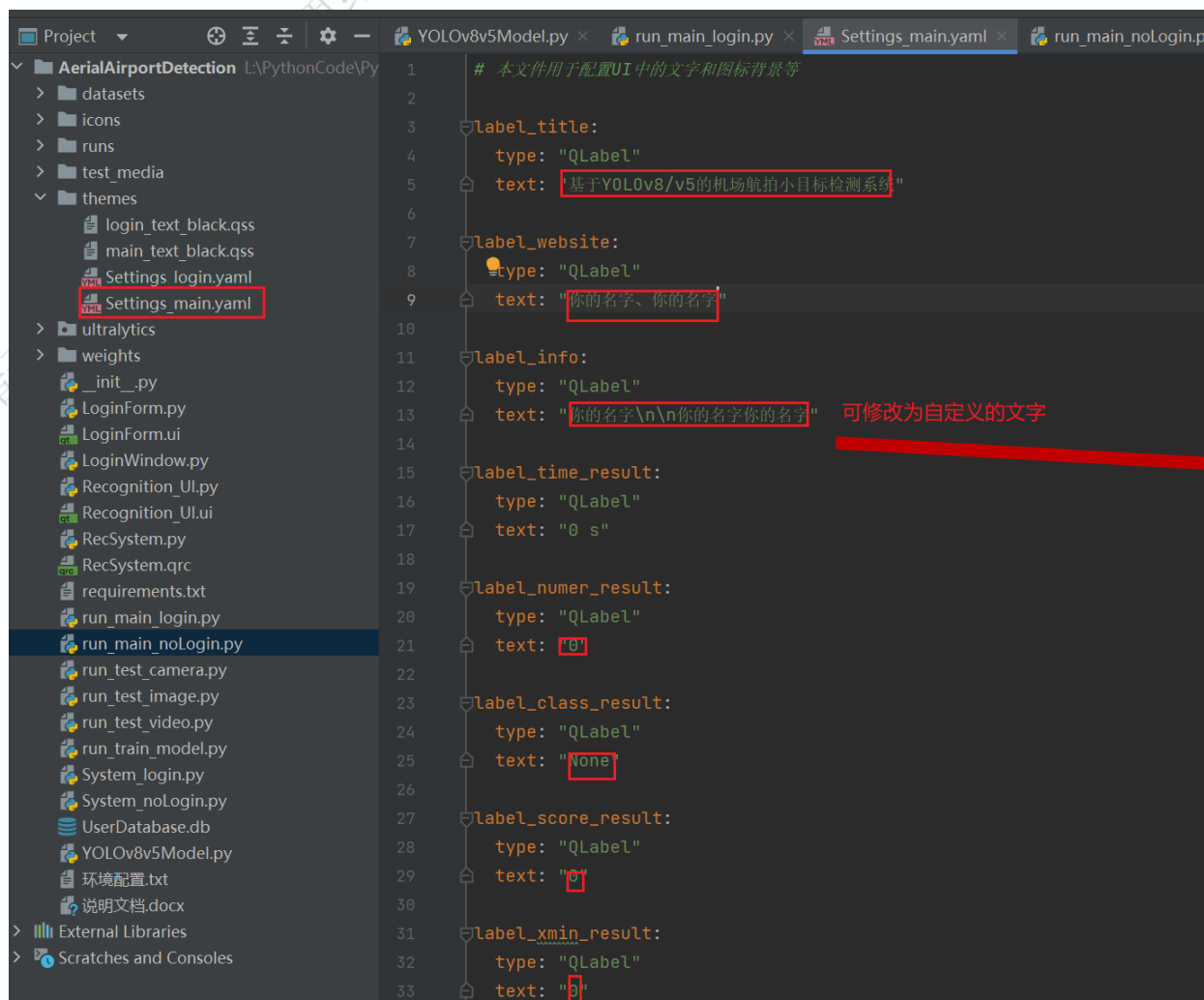
在项目文件夹下的runs文件夹中，可以找到训练和评估的结果图表





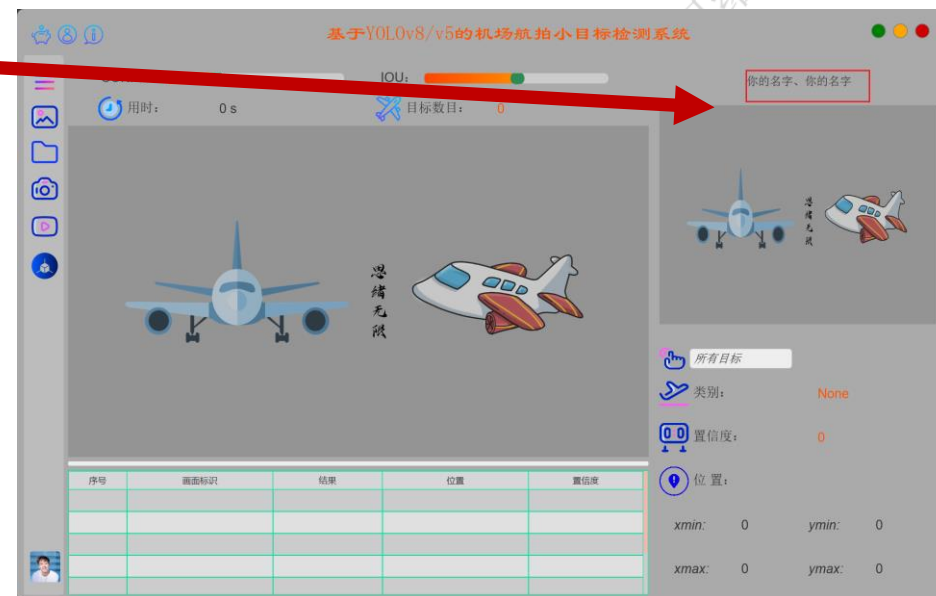
# 修改项目

修改界面上的图标背景等png，可直接用自己的图替换icons文件夹中的对应图片（不改变文件名），也可在themes文件夹的yaml文件中修改设置

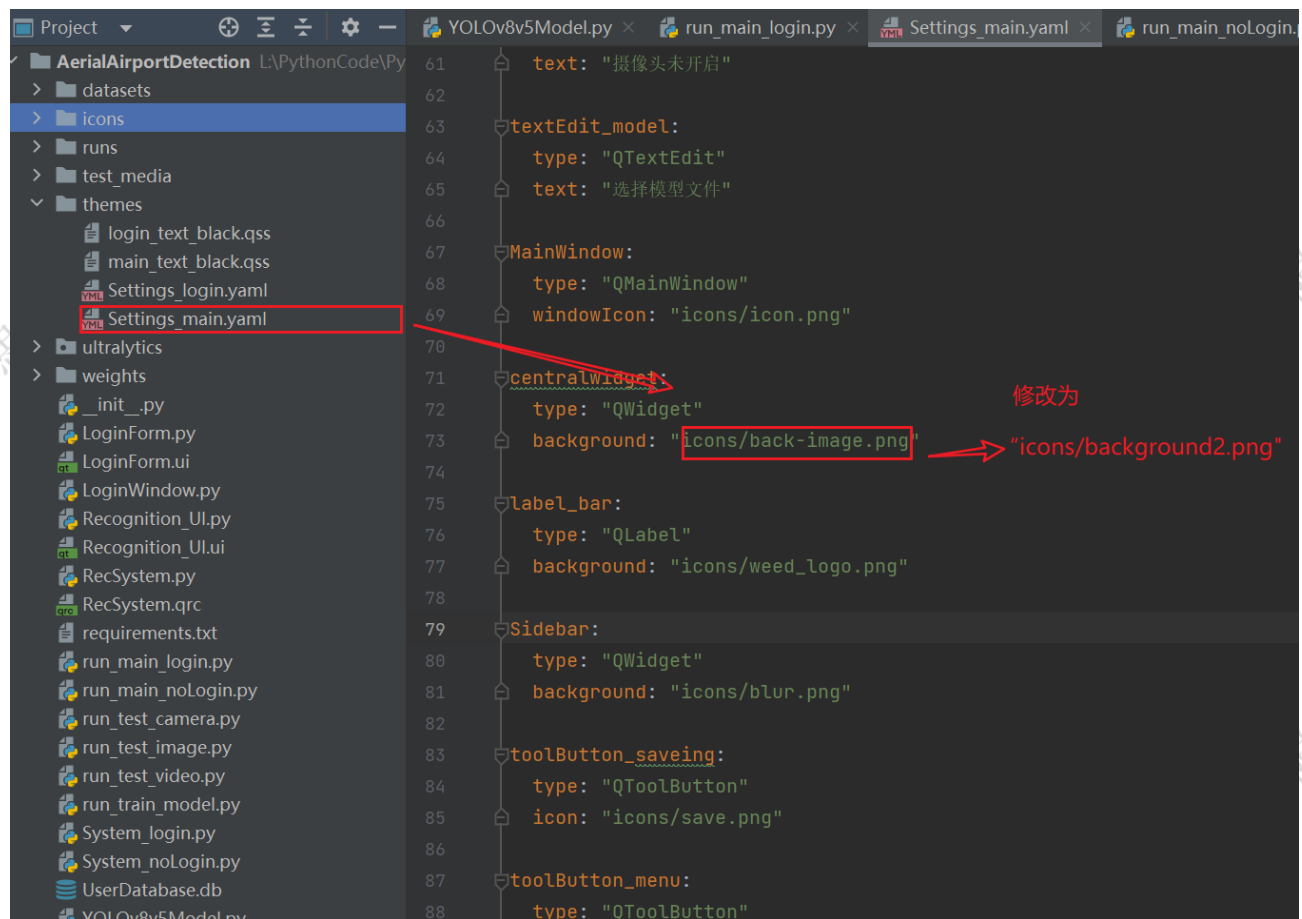


在themes文件夹中的yaml文件中可以修改界面上的文字、图标、背景等

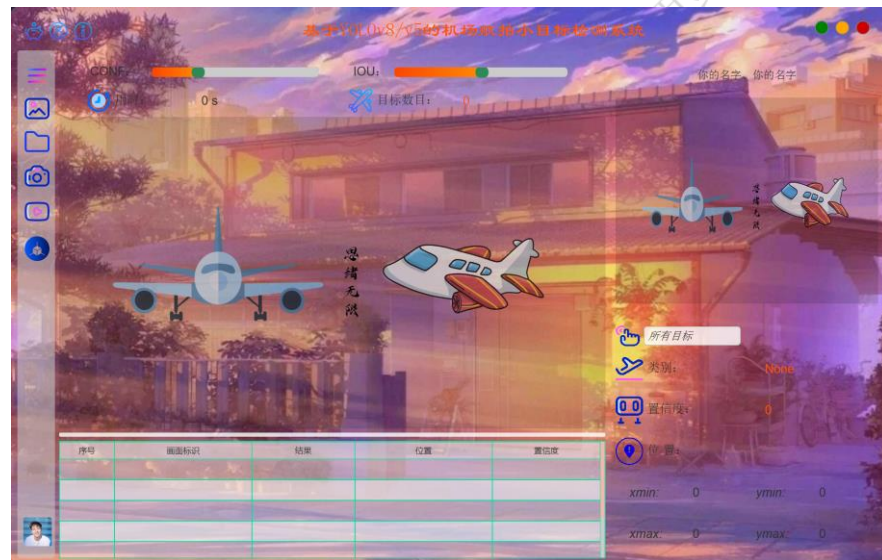
比如，将label\_website下的text改成“你的名字”，再次运行界面如下



# 修改项目

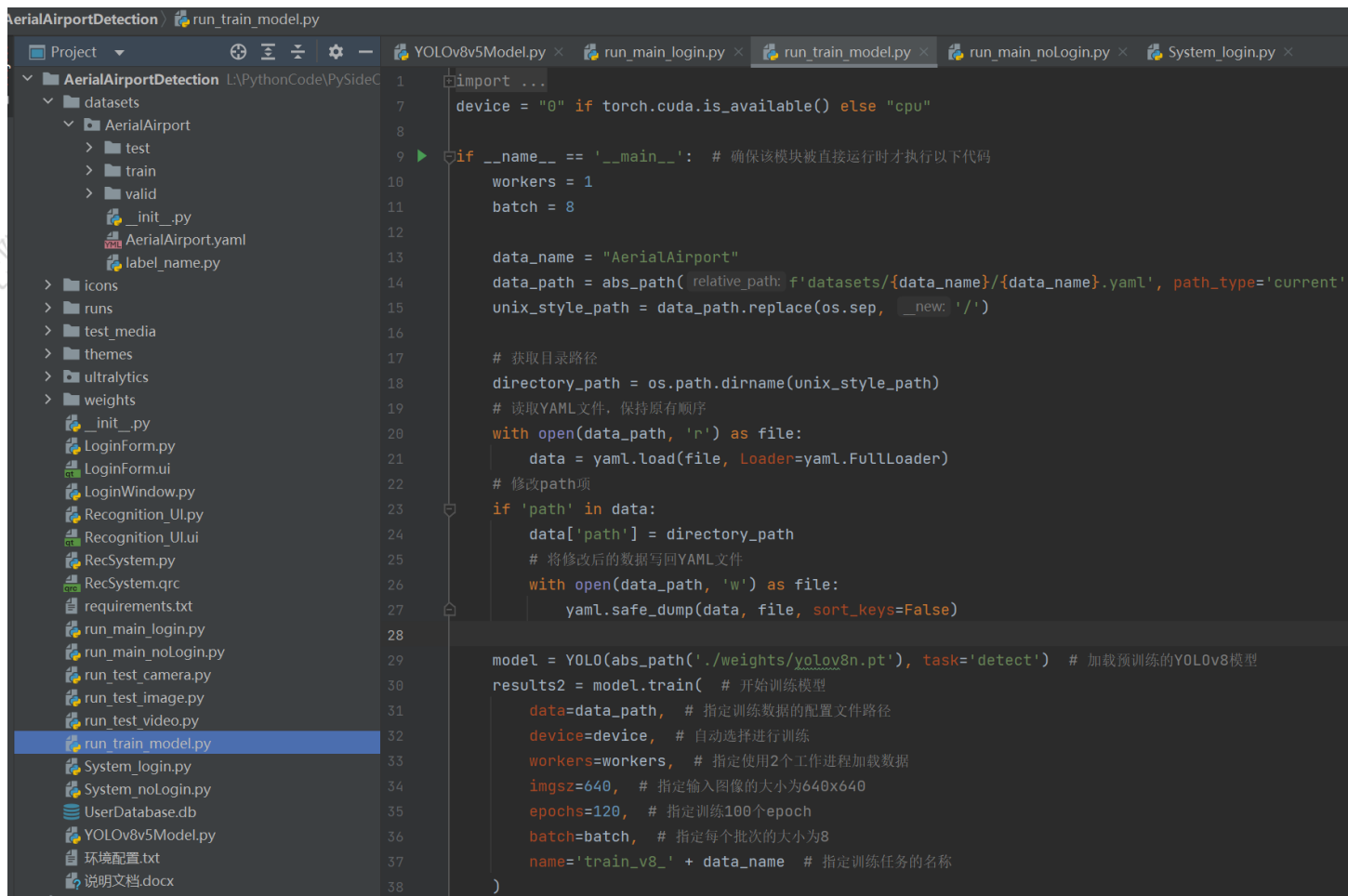


也可以修改背景和图标的文件路径，比如修改centralwidget的background为“icons/background2.png”，界面背景变为如下的，图标的路径也可以修改替换：



# 重新训练

运行run\_train\_model.py文件可以重新进行训练，其中如果安装的是gpu版本的pytorch会自动启动gpu训练，如果是cpu版本的pytorch则使用cpu运行；本项目中已经保存了训练好的模型和训练结果，可以不用再次训练而直接使用；



```
1 import ...
2
3 device = "0" if torch.cuda.is_available() else "cpu"
4
5
6
7
8
9 if __name__ == '__main__': # 确保该模块被直接运行时才执行以下代码
10     workers = 1
11     batch = 8
12
13
14     data_name = "AerialAirport"
15     data_path = abs_path( relative_path( f'datasets/{data_name}/{data_name}.yaml', path_type='current' ) )
16     unix_style_path = data_path.replace(os.sep, '/')
17
18     # 获取目录路径
19     directory_path = os.path.dirname(unix_style_path)
20     # 读取YAML文件，保持原有顺序
21     with open(data_path, 'r') as file:
22         data = yaml.load(file, Loader=yaml.FullLoader)
23     # 修改path项
24     if 'path' in data:
25         data['path'] = directory_path
26         # 将修改后的数据写回YAML文件
27         with open(data_path, 'w') as file:
28             yaml.safe_dump(data, file, sort_keys=False)
29
30     model = YOLO(abs_path('./weights/yolov8n.pt'), task='detect') # 加载预训练的YOLOv8模型
31     results2 = model.train( # 开始训练模型
32         data=data_path, # 指定训练数据的配置文件路径
33         device=device, # 自动选择进行训练
34         workers=workers, # 指定使用2个工作进程加载数据
35         imgsz=640, # 指定输入图像的大小为640x640
36         epochs=120, # 指定训练100个epoch
37         batch=batch, # 指定每个批次的大小为8
38         name='train_v8_' + data_name # 指定训练任务的名称
39     )
```

1. 内存较小的电脑可以将batch的数值设置更小一些防止内存不够报错；
2. 训练采用的数据集存放在datasets文件夹下，训练过程中的结果保存在runs文件夹下的子文件夹中；
3. 每次运行训练会自动添加一个文件夹，最终训练得到的模型pt文件也会在该文件夹中；