# 摄像头识别测温

目录

[摄像头识别测温 1](#__RefHeading___Toc1149_1733389947)

[1.项目概述 3](#__RefHeading___Toc1151_1733389947)

[1.1工作流程 3](#__RefHeading___Toc1153_1733389947)

[1.2主程序 4](#__RefHeading___Toc1155_1733389947)

[1.2.1 注意事项 4](#__RefHeading___Toc1157_1733389947)

[1.2.2 关联 5](#__RefHeading___Toc1159_1733389947)

[1.2.3 网页前端 6](#__RefHeading___Toc1161_1733389947)

[2 项目功能介绍 7](#__RefHeading___Toc1163_1733389947)

[2.1 摄像头 7](#__RefHeading___Toc1165_1733389947)

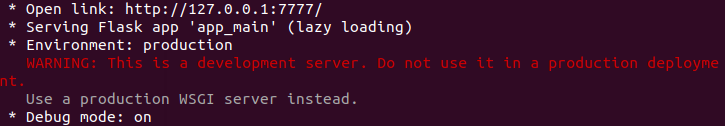
[2.1.1 设备登录 7](#__RefHeading___Toc1167_1733389947)

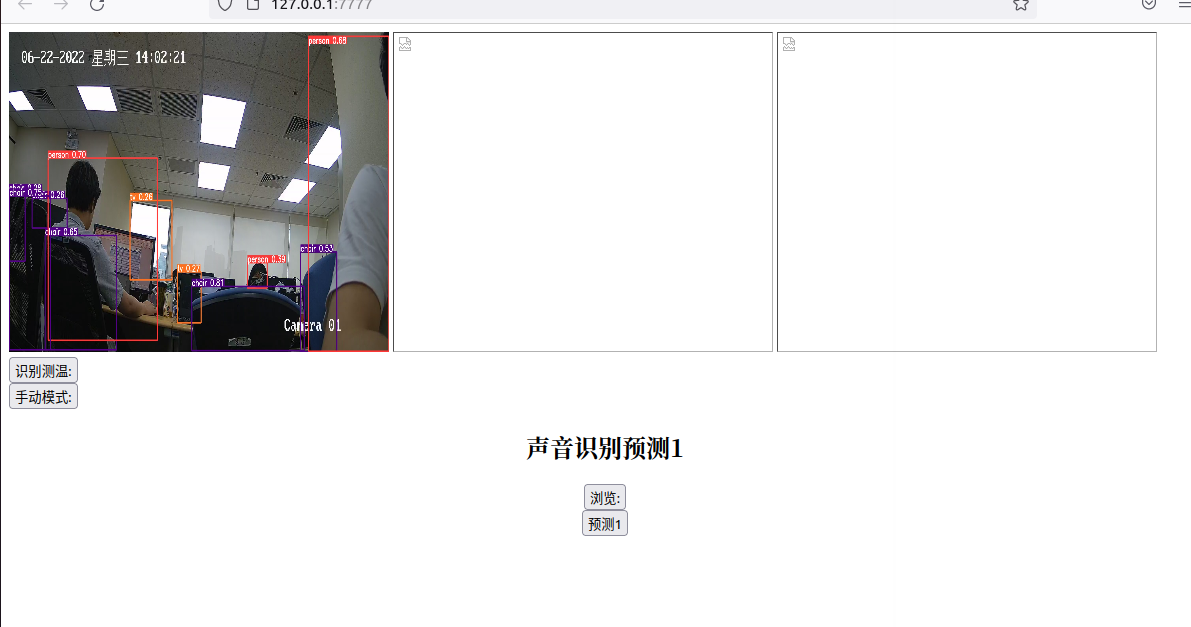
[2.1.2 摄像头控制和操作 7](#__RefHeading___Toc1169_1733389947)

# 1. 项目概述

## 1.1工作流程

通过rosrun启动脚本文件（如图1.1），浏览器打开设置好的端口（默认设置为http://127.0.0.1:7777/），打开后等待几秒模型加载完毕可以看到以下界面（如图1.2），页面中具有4个功能：

图 1.1

图 1.1

（1）识别测温：即摄像头拍照抓取一张图片，将拍照的图片传入识别模型根据设置的标签进行识别（如设置识别‘person’标签即只识别人）,识别完成之后再将识别的数据集（包含一个或多个识别框的（左上角坐标，右下角坐标，标签名称））传入摄像头再对数据集中所给的坐标进行测温，返回测温的结果（默认为坐标范围内最高温），如果测温范围内没有可以识别的标签，则没有执行其他动作。

（2）浏览：即浏览一个文件路径，是用于声音识别，需要浏览的文件类型是wav格式的文件。

（3）预测：即正常或异常声音识别预测结果，先在浏览中选取了一个文件，如果是wav格式则进行识别预测，如果结果是[0]则说明该声音是异常声音，如果是[1]则说明该声音是正常声音。

（4）手动模式：是将测温的模式变成手动测温，点击“手动测温”后会弹出一个cv窗口可以使用鼠标手动进行测温，但是由于本台电脑的线程bug，无法和实时展示的另一个画面一起运行，会导致cv的线程卡死。

## 1.2主程序

### 1.2.1 注意事项

主程序是app\_main.py，使用flask框架部署预测模型，ros运行时需要在导入包的前面添加以下：

#!/usr/bin/env python3

import rospy

from std\_msgs.msg import String

并且在if \_\_name\_\_ == ‘\_\_main\_\_’执行部分添加以下，目的是开启ros节点，可以rosnode list查询到该脚本节点

pub = rospy.Publisher('chatter', String, queue\_size=10)

rospy.init\_node('app\_cam', anonymous=True, disable\_signals=True)

如果不使用rosrun运行脚本，可以将以上包含import部分全部注释也可以正常在python中运行。

### 1.2.2 关联

主程序主要关联了3个部分，图像识别，声音预测，摄像头。

（1）摄像头：app\_main.py中导入的是在子目录RGB\_TH文件夹下的hk\_main.py文件，是摄像头相关的主程序，其相关功能的使用包含了摄像头的登录，拍照，测温（机型相关），云台控制（机型相关）的功能。其他功能可以根据海康威视的SDK进行添加，具体其他的功能使用可以参考海康威视官网提供的开发文档。

（2）图像识别：app\_main.py中导入的是在子目录yolov5\_master文件夹下的detect\_app.py和detect\_video.py文件，是基于yolov5的detect.py修改，主要是用于该项目的文件，app是用于在网页前端点击“识别预测”按钮时执行的相关功能，而video是用于网页前端中画面的展示，使用的是rtsp视频流，会有几秒钟的延迟存在。

（3）声音预测：app\_main.py中导入的是在子目录sound文件夹下的sound.py文件，用于声音预测，根据选取的wav格式文件传入sound.py进行预测。

### 1.2.3 网页前端

在这个项目中使用的前端html文件在该文档目录下的templates文件夹中的app\_main.html文件，页面组件可以在这里更改，图片展示的大小也可在这更改。

# 2 项目功能介绍

## 2.1 摄像头

### 2.1.1 设备登录

登录摄像头主要的相关代码在RGB\_TH文件夹下的hk\_sdk.py文件中，在登录之前需要对摄像头进行初始化操作调用init函数即可，其中的login函数是登录模块，登录需要输入摄像头ip和用户信息（需要和摄像头在同一个局域网下）这些信息的修改是在app\_main.py文件的开头处，而端口port默认8000即可，不用更改。

### 2.1.2 摄像头控制和操作

摄像头的控制相关功能代码在RGB\_TH文件夹下的hk\_ctrl.py文件

（1）拍照：capture函数，在hk\_main中登录了信息后传入摄像头的信息，完成拍照RGB（frame\_rgb.jpg）和热成像（frame\_th.jpg）两个画面的图像，再对RGB图添加一个可测温范围的框保存为另一张图片（frame\_rgb\_addrect.jpg）

（2）连续拍照：cam\_rgb和cam\_thermal，一个是RGB画面的连续拍照，一个是热成像画面的连续拍照，这些功能是用于手动测温功能，优点是画面是实时的，延迟低；缺点是画面帧率低，是由于摄像头执行拍照功能需要花费一些时间。两个函数的功能是类似的。

（3）测温功能：temp函数，主要用于“识别测温”功能中的测温，在通过识别出来的标签和坐标信息，整合成数据集传入temp\_data参数，数据集需要注意的是：数据形式为[[(x1, y1), (x2, y2), 标签名], [[(), (), \_]], …]，主要是由测温左上角点和右下角点，以及标签名称组成。传入的坐标需要做预处理，坐标转换，使用的是RGB图像中在可测温的范围坐标，不超出测温范围，如果超出则转换成测温边界的坐标。传入temp后坐标会进一步处理，将坐标转换成热成像画面的坐标再转换成测温坐标，做了2次转换，在RGB坐标转换热成像坐标时会因为左右相机视场不一样直接转换会导致在热成像画面的物体并不能在准确的位置上，因此需要设置误差值，这个误差值会根据距离变化的，越进的物体误差值会越大，代码中设置的误差值是3米的误差，可以较准确的在3-5米范围内进行坐标转换。误差公式的设置可以在hk\_ctrl.py文件的开头参数定义中进行设置修改。

测温模块最主要的功能在hk\_thermal.py中，其中主要有3个函数，第一个是在可测温范围内测温所有坐标的最高温度get\_temperature\_all函数，第二个是在标定坐标内进行测温选择最高温（默认）get\_temperature\_max函数，还有最低温和平均温参数，第三个则是获取测温的温度get\_temperature0函数。其中主要使用的是第二个函数模块，可以灵活使用，可根据传入的热成像画面坐标进行测温。

（4）鼠标事件：是用于控制摄像头的操作，如云台控制，画面放大缩小，手动框选测温。手动框选测温是与测温功能是类似的，同样需要坐标转换。有event\_rgb和event\_th两个函数，一个是在RGB画面上控制，另一个是在热成像画面上控制。

## 2.2 识别检测

### 2.2.1 视频检测

视频检测主要使用的py文件是yolov5\_master文件夹下的detect\_video.py文件，是基于yolov5的源脚本detect.py更改的，减少其他功能的使用，专门用于视频流检测，根据摄像头的rtsp视频流的地址进行实时检测，当app\_main.py脚本运行时打开的端口页面会等待一段时间，这段时间是在加载模型的过程，加载完毕即显示检测的画面在html界面上。

### 2.2.2 图像识别测温

图像识别主要使用的是detect\_app.py文件，也是由detect更改而来的，只用于传入图片进行识别检测的，当在html界面点击了“识别测温”时，会先根据拍照的图片先进行识别和检测，后保存为识别后的图片（pred\_PIC.jpg），再根据是否超出可测温范围以及根据所选的标签进行筛选再对目标的坐标进行转换，将超出测温范围的部分坐标转换成测温范围以内的坐标，最后将这些坐标和标签整合成数据集传入RGB\_TH文件夹下的hk\_main.py的模式1（mode=1），即识别测温，传入数据后对识别转换后的坐标进行测温并将可见光画面和热成像画面分别保存图片为识别测温图（pred\_temp.jpg）和识别测温热成像图（pred\_temp\_thermal.jpg）。

## 2.3 声音预测

声音预测所使用的脚本是在sound文件夹下的sound.py文件。

首先点击“浏览”选择一个wav格式的音频文件，如果无误，则会加载声音识别模型并检测（使用的是正常的声音模型），返回的结果为[0]和[1]，如果是[0]则说明该声音是异常声音，如果是[1]则说明该声音是正常声音。

注：“浏览”按钮的设计不要使用html的浏览按钮，html的浏览按钮只能显示文件名称，不会显示全路径名称，因此使用的浏览功能是python的tkinter的浏览模块。

## 2.4 模型训练

训练自定义模型使用yolov5\_master文件夹下的train.py文件训练，可以参考以下网址：

<https://blog.csdn.net/weixin_55073640/article/details/122874005>

<https://www.cnblogs.com/zhaoyingjie/p/14620817.html>

<https://blog.csdn.net/a_cheng_/article/details/111401500>

<https://blog.csdn.net/qq_48639918/article/details/119865900?spm=1001.2014.3001.5501>