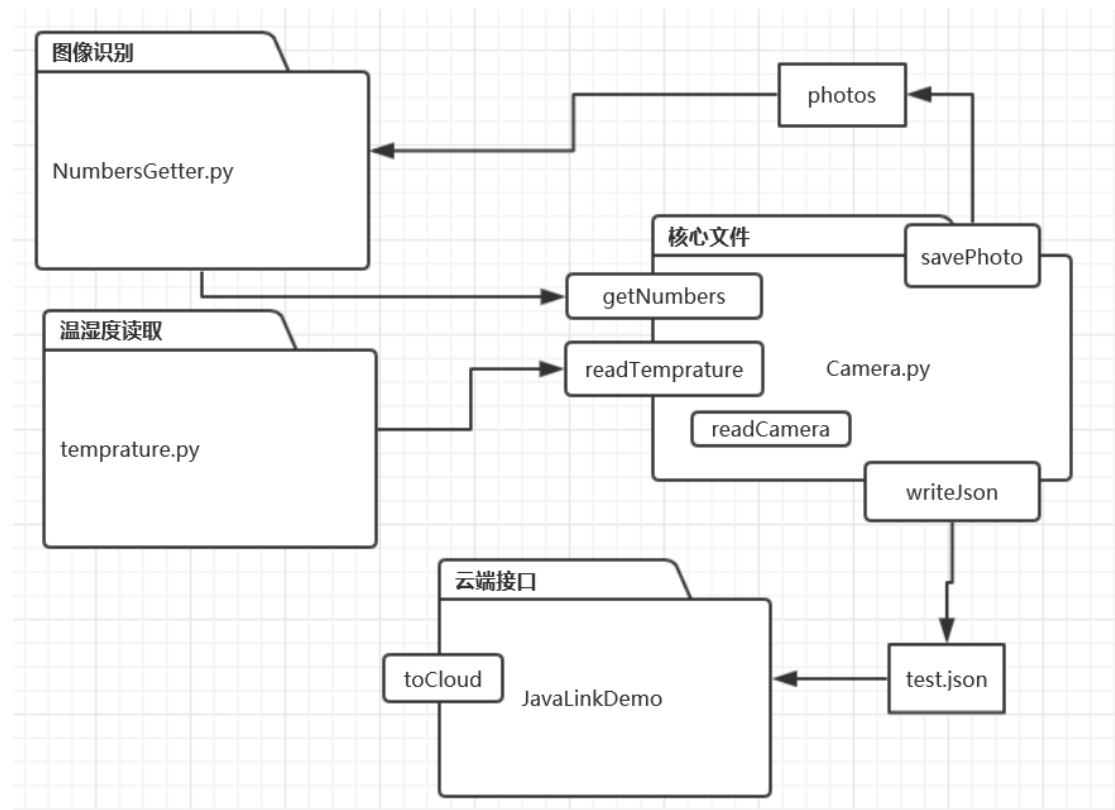


《边缘计算程序文档》

项目结构：



模块介绍:

核心文件 Camera.py, 需要 root 权限运行在 cv 环境里

Camera.py 为项目的核心模块, 主要功能分为图像、温湿度与数据三个部分。

```
53         imgbase64 = readImg(img_name)
54         resJson = sendImg(imgbase64)
55         res,lis = getMessage(resJson)
```

图像部分引入 cv2 库来调用摄像头获取图像, 并将图片以 jpg 格式储存。图像部分还负责通过 readImg 接口与图片路径名称 img_name 获取对应的 base64 编码图片, 调用 sendImg 接口生成与图片对应的 json 信息, 再通过 getMessage 接口与上述 json 信息获取 NumbersGetter.py 对对应图像进行处理后得到的编号信息。

```
61         tmp_data, hum_data = get_data(ser)
```

温湿度部分则通过 init 接口初始化 temprature 模块的串口与波特率, 再通过 get_data 接口获取 temprature.py 读取的温湿度信息。

```
57         f = open('./test_case.json','w')

63         out_json = detail(flag, tmp_data, hum_data)
64         f.write(out_json)
65         f.close()
```

数据部分通过改写 test_case.json 文件, 以此来储存并向 JavaLinkDemo 传送信息。其中 detail 函数为 Camera.py 中定义的数据整合函数, 负责根据传入信息生成写入 json 文件的数据。

```
44     while cam.isOpened():
```

Camera.py 的主逻辑在调用的摄像头对象打开时进行循环执行上述功能。

图像识别 NumbersGetter.py

NumbersGetter.py 是图像识别模块。该模块内提供了 getAccessToken、readImg、sendImg、getMessage 接口。

```
13     appkey = "LZ7XNHWV8XekN0XsE94Po0zt"
14     secretkey = "T5H3C7ZoClU8FRzzSnIAH6Z9GdmUAQB6"
15     host = 'https://aip.baidubce.com/oauth/2.0/token?grant_type=client_credentials&client_id='+ appkey + '&client_secret=' + secret
16     response = requests.get(host)
```

getAccessToken 是百度接口调用的配置, 负责记录百度接口调用的 key 与 host 等信息, 同时负责调用百度接口。

```
25         img = open(path, "rb")
26         imgBase64 = base64.b64encode(img.read())
27         img.close()
28         return imgBase64
```

readImg 的功能是通过提供的图片路径名称打开对应图片, 并将图片改为使用 base64 编码后返回。

```

32     url = 'https://aip.baidubce.com/rest/2.0/ocr/v1/numbers?access_token='+getAccessToken()
33     headers = {
34         'Content-Type': "application/x-www-form-urlencoded",
35     }
36     body = {
37         "image": imgBase64Code,
38     }
39     response = requests.post(url, body, headers = headers)
40     return response.text

```

sendImg 负责将提供的 base64 编码图片信息发送到百度 api, 并将百度 api 提供的对应 json 信息返回。

```

44     message = json.loads(numberJson)
45     resultNum = message.get('words_result_num')
46     resultList = message.get('words_result')
47     for i in range(len(resultList)):
48         resultList[i] = resultList[i]['words']
49     return resultNum, resultList

```

getMessage 则是负责处理提供的 json 数据获取其中的信息, 并将这些信息返回。

温湿度读取 temprature.py

temprature.py 是温湿度读取模块。temprature.py 引入 serial 库, 并提供 init 与 get_data 两个接口。

```

11     ser = serial.Serial("/dev/ttyUSB0", 9600) # 选择串口, 并设置波特率

```

Init 接口是初始化 serial 调用的接口。可以通过更改参数选择串口与设置波特率。

```

15     if ser.is_open:
16         # hex(16进制)转换为bytes(2进制), 应注意Python2.7与Python3.7此处转换的不同
17         send_data = '010300000002C40B'
18         send_data = bytes.fromhex(send_data) # 发送数据转换为b'\xff\x01\x00U\x00\x00V'
19         ser.write(send_data) # 发送命令
20         time.sleep(0.1) # 延时, 否则len_return_data将返回0, 此处易忽视!!!
21         len_return_data = ser.inWaiting() # 获取缓冲数据(接收数据)长度
22         if len_return_data:
23             return_data = ser.read(len_return_data) # 读取缓冲数据
24             # bytes(2进制)转换为hex(16进制), 应注意Python2.7与Python3.7此处转换的不同, 并转为字符串后截取所需数据字段, 再转为10进制
25             str_return_data = return_data.hex()
26             feedback_data = int(str_return_data, 16)
27             tmp_data = int(str_return_data[7:10], 16)/10
28             hum_data = int(str_return_data[11:14], 16)/10
29             print(tmp_data)
30             print(hum_data)
31             return tmp_data, hum_data
32     else:
33         print("port open failed")
34         return '0', '0'

```

get_data 接口负责从提供的 serial 中读取温湿度信息, 并将温湿度信息处理成可用格式并返回。

云端接口 JavaLinkDemo

JavaLinkDemo 是云端接口模块, 由云平台提供的接口文件包 JavaLinkKitDemo-1.0.0-jar-with-dependencies.jar 构成, 其功能是每 5s 读取一次 test.json 文件的内容并上传到云端。