

2025 年河北省职业院校学生技能大赛
高职组“智慧城市技术应用”赛项

样题

竞赛须知

一、竞赛要求

- 1、正确使用工具，操作安全规范；
- 2、竞赛过程中如有异议，可向现场考评人员反映，不得扰乱赛场秩序；
- 3、遵守赛场纪律，尊重考评人员，服从安排。

二、职业素养与安全意识

- 1、完成竞赛任务所有操作符合安全操作规范，注意用电安全；
- 2、操作台、工作台表面整洁，工具摆放等处理符合职业岗位要求；
- 3、遵守赛场纪律，尊重赛场工作人员；爱惜赛场设备、器材。

三、扣分项

- 1、在竞赛过程中，因操作不当导致设备破坏性损坏或造成事故，视情节扣10~20分，情况严重者取消比赛资格；
- 2、衣着不整、污染赛场环境、扰乱赛场秩序、干扰裁判工作等不符合职业规范的行为，视情节扣5~10分，情节严重者取消竞赛资格；

四、选手须知

- 1、任务书如出现缺页、字迹不清等问题，应及时向裁判示意，并进行更换；比赛结束后，比赛提供的所有纸质材料、U盘等不得带离赛场；
- 2、设备的安装配置需严格按照任务书的要求及工艺规范进行操作；
- 3、参赛团队应在规定时间内完成任务书要求的内容，任务实现过程中形成的文件资料必须按要求存储到指定位置，未存储到指定位置的文件均不得分；
- 4、比赛过程中，选手如怀疑设备问题，且有明确证据确认损坏由非选手因素造成，可向裁判提交书面说明，经技术人员判断和裁判长裁决认可，可更换设备，并由裁判长裁决是否补时和补时长度，没有明确证据确认损坏由非选手因素造成设备损坏的，不予更换设备和补时；
- 5、在裁判组宣布竞赛结束后，参赛选手应立即停止对竞赛设备与计算机的任何操作。

竞赛设备及注意事项

竞赛选手依照本竞赛项目的任务内容，完成任务书要求的相关操作与开发任务。

一、注意事项

1、检查硬件设备、电脑设备是否正常。检查竞赛所需的各项设备、软件和竞赛材料等；

2、竞赛任务中所使用的各类软件工具、软件安装文件等，都已拷贝至 U 盘上（如提供压缩包形式，需参赛选手将其复制到电脑的最后一个盘中并解压），自行根据竞赛任务要求使用；

3、竞赛过程中应严格按照竞赛任务中的描述，对各设备进行安装配置、操作使用，对于竞赛前竞赛工位上已经连接好的设备，可能与后续的竞赛任务有关，请勿变动；

4、竞赛任务完成后，需要保存设备配置，保持各模块的运行状态，不要关闭任何设备，不要拆动硬件的连接线，不要对设备随意加密。

二、硬件环境

序号	设备名称	单位	数量
1	智慧城市技术通用竞赛平台	套	1
2	个人计算机	台	2

第一部分 智慧城市中的图像识别与检测

任务一：智慧城市监控系统的人脸和微笑检测

场景描述：在智慧城市监控系统中，计算机视觉技术被广泛用于人脸检测和微笑检测，以提高城市的监控和安全性。您将创建一个 Python 项目，使用 OpenCV 分类器实现这些功能。

任务描述：

1、根据“竞赛资料\第一部分\任务一\1”中提供的图片数据集，使用 opencv 中的分类器，对给定图片数据集中的人脸进行检测，要求使用绿色框将人脸标出，红色框将眼睛标出，并将检测后图片进行保存到“提交资料\第一部分\任务一\1”目录下。

2、根据“竞赛资料\第一部分\任务一\2”中提供的视频数据集，使用 opencv 中的分类器，对视频中的面部微笑进行检测，要求检测到微笑时将字符‘smile’打印展示在视频中，并将检测后的视频保存到“提交资料\第一部分\任务一\2”目录下。

3、根据“竞赛资料\第一部分\任务一\3”中提供的视频文件使用 OpenCV 分类器检测视频中的行人并用绿色框进行标注，将框的中心点用红色点进行标注，将处理后的视频保存到“提交资料\第一部分\任务一\3”目录下。

完成后，保存本工程所有文件及代码到竞赛 U 盘“提交资料\第一部分\任务一”目录下。

任务二：图像轮廓检测与直方图计算

场景描述：在图像处理领域，轮廓检测和直方图计算是常见的技术，可以用于图像分析、特征提取等应用。您将创建一个 Python 项目，通过编写代码实现对给定数据集中图像的轮廓检测和直方图计算，并保存处理后的结果。

任务描述：

1. 根据“竞赛资料\第一部分\任务二”中提供的数据集和文件，对数据集中所有的图片进行轮廓检测，将轮廓检测处理后的图片保存到竞赛 U 盘“提交资料\第一部分\任务二\处理后的图片”目录下。
2. 根据“竞赛资料\第一部分\任务二”中提供的数据集和文件，使用 `calcHist` 函数计算所有图像的直方图，将计算后的直方图用 `matplotlib` 绘制出并保存在“竞赛资料\第一部分\任务二\2”目录下。

完成后，保存本工程所有代码到竞赛 U 盘“提交资料\第一部分\任务二”目录下。

第二部分 智慧城市中的图像预处理

任务一：OpenCV 图像处理在智慧城市监控系统中的应用

场景描述：智慧城市监控系统利用图像处理技术对城市中的各种情况进行监测和分析，以维护城市的安全和秩序。本项目旨在利用 OpenCV 对智慧城市监控系统中的图像进行处理，以实现多种操作。

任务描述：

1. 从路径“竞赛资料\第二部分\任务一”中提供的数据集中，将所有图片缩放到大小为 480*270 像素，并同名保存到指定目录“提交资料\第二部分\任务一\1”下。
 2. 从缩放后的数据集中任选 2 张猫狗图片进行图像融合操作。要求狗占比 0.8，猫占比 0.2，保存处理后的图片到指定目录“提交资料\第二部分\任务一\2”下。
 3. 将数据集中的所有图片进行水平镜像翻转，并同名保存到指定目录“提交资料\第二部分\任务一\3”下。
 4. 对提供的 mohu.jpg 进行中值滤波操作，并同名保存到指定目录“提交资料\第二部分\任务一\4”下。
 5. 将数据集中的所有图片进行颜色空间转换，将 RGB 图像转换为 HSV 图像，并同名保存到指定目录“提交资料\第二部分\任务一\5”下。
- 完成后，请保存所有工程代码到竞赛 U 盘“提交资料\第二部分\任务一”目录下。

任务二：智慧城市中的文本识别与处理

场景描述：在智慧城市的运行中，信息的快速处理和识别至关重要。

本任务旨在利用 OpenCV 技术处理智慧城市中的文本信息，以提高城市管理效率和智能化水平。

任务描述：

1. 对“竞赛资料\第二部分\任务二”中提供的文本图像，将图像读取到 Python 中，并使用 OpenCV 进行灰度处理和二值化，优化图像质量，将处理后的图像保存，并提交到“提交资料\第二部分\任务二\2”目录下。
2. 使用 OpenCV 进行文本识别来识别图像中的文字，在图像上将识别到的文字区域用红色框标出，并保存标注后的图片到“提交资料\第二部分\任务二\2”目录下。
3. 使用 OpenCV 进行文本识别来识别图像中的文字，在图像上将识别到的文字打印展示在对应区域的下方，并保存处理后的图片到“提交资料\第二部分\任务二\3”目录下。

完成后，保存本工程所有代码到竞赛 U 盘“提交资料\第二部分\任务二”目录中。

第三部分 智慧城市中的识别模型构建

任务一：模型参数设定与训练配置

场景描述：在构建和训练深度学习模型时，合理设置模型参数和训练配置是非常重要的。本任务旨在对提供的 TensorFlow 模型进行参数设定和训练配置，以确保模型的有效性和训练效率。

任务描述：

1. 设定填写 `input_shape` 的值，查看自己构建模型后的 Total parmas, 值。Total parmas 代表模型的训练参数个数，`input_shape` 代表图片的输入尺寸（要求最低为 224）。
2. 设定填写 `epochs` 的值。`epochs` 代表训练轮数，表示完成模型训练过程需要遍历整个训练数据集的次数。
3. 设定填写 `batch_size` 的值。`batch_size` 表示模型每次更新参数所需要的样本数量。
4. 设置训练的损失函数为交叉熵损失函数。
5. 将模型中的所有激活函数设置为“Rectified Linear Unit（修正线性单元）”。
6. 设置模型优化器为自适应学习率的优化器 Adaptive Moment Estimation。

完成后，将每一步的代码截图保存至竞赛 U 盘中答题卡的指定位置上，参数要求合理以确保能在比赛时间内完成模型训练。

任务二：智慧城市中的猫狗识别模型训练与优化

场景描述：在智慧城市的监控系统中，猫狗识别模型是一项重要的技术，可以帮助城市管理者更好地监控城市中的宠物活动，确保城市的环境安全。本任务旨在对提供的猫狗数据集进行模型训练和优化，以确保模型在测试集上的准确性。

任务描述：

1. 对提供的猫狗训练集图片进行随机翻转、平移、随机裁剪、随机缩放等数据增强操作，并截图对应代码块保存到竞赛 U 盘中答题卡的指定位置上。
2. 合理调整及优化代码（参考：网络结构、输入图片大小、优化器、batch_size 等），使得测试集上的得分(test_score)不低于 85%，并截图保存到竞赛 U 盘中答题卡的指定位置上。
3. 在模型训练中加入回调函数使得能在 tensorboard 中对训练过程进行可视化，运行 tensorboard，查看训练过程中的准确率和损失值的变化图像，并截图保存到竞赛 U 盘中答题卡的指定位置上。
4. 对于训练好的所有模型文件进行保存，要求指定模型格式为 hdf5，并保存到竞赛 U 盘“提交资料\第三部分\任务二\4”目录下。
5. 对提供的猫狗数据集重新进行迁移学习训练，模型要求使用 vgg19 网络，将 vgg19 网络结构图及训练过程结果对应截图保存到竞赛 U 盘中答题卡的指定位置上。
6. 舍弃 vgg19 原来的全连接层，自己添加新的全连接层训练，将对应的截图保存到竞赛 U 盘中答题卡的指定位置上。

7. 在 vgg19 网络训练中加入回调函数，要求在模型 ACC 达到 85%时停止训练并保存模型，将对应的截图保存到竞赛 U 盘中答题卡的指定位置上，模型保存位置为竞赛 U 盘“提交资料\第三部分\任务二\7”目录下。

完成后，保存本工程所有文件和代码到 U 盘“提交资料\第三部分\任务二”目录中。

任务三：智慧城市中的模型调用与预测

场景描述：在智慧城市监控系统中，模型的调用和预测是关键步骤，可以帮助识别和监测城市中的各种对象和活动。本任务旨在调用训练好的模型，对提供的图片数据集进行逐一预测，并将预测结果标注在验证集图片上，以便后续分析和应用。

任务描述：

1. 调用任务二中训练保存的 hdf5 模型或训练好的 vgg19 网络模型。
2. 选择准确率较高的模型对“竞赛资料\第三部分\任务三”中提供的图片数据集(待预测图片集-300)进行逐一进行预测并将预测结果用绿色字体打印保存在验证集图片上，将预测完毕的验证集保存到竞赛 U 盘“提交资料\第三部分\任务三”中。

完成后，保存本工程所有代码及模型到竞赛 U 盘“提交资料\第三部分\任务三”目录中。

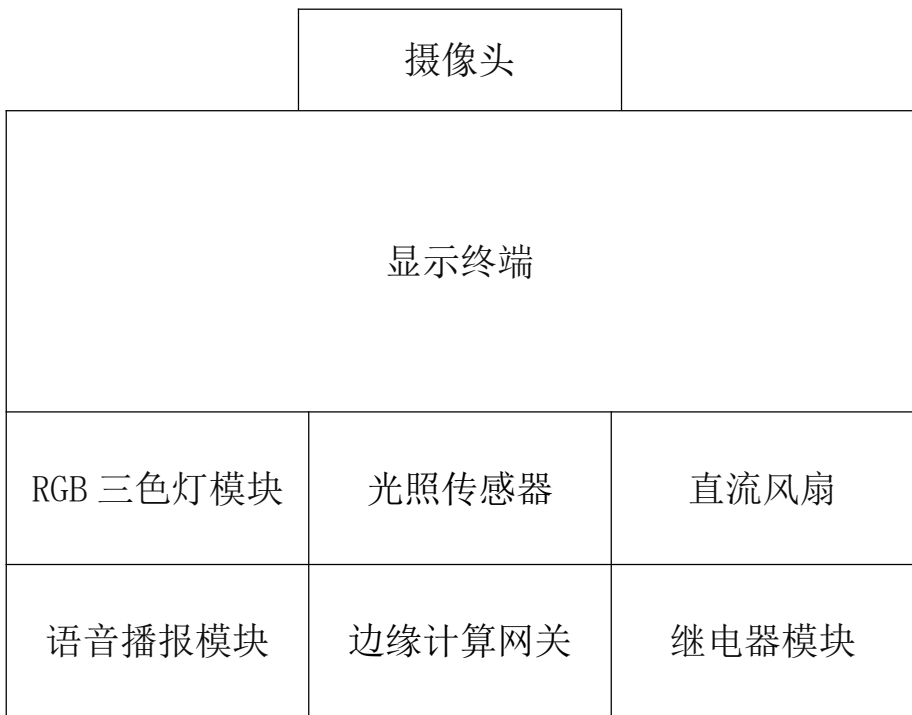
第四部分 智慧城市中的终端代码编辑与应用

任务一：智慧城市基础设施层技能考核

场景描述：在智慧城市建设中，基础设施层的设备选型、布局 and 安装连接至关重要。这些设备包括传感器和执行器，它们的正确安装对于城市的智能化运行至关重要。本任务要求按照指定的布局图，将传感器和执行器设备安装到智慧城市技术通用竞赛平台的对应位置上。安装过程需符合工艺标准，确保设备安装正确、位置工整、美观。

任务描述：

工位设备安装布局图如下：



任务二：智慧城市网络通信层设备配置

场景描述：在智慧城市的网络通信层中，设备配置是确保信息传输和通讯畅通的关键。参赛选手将在现场进行局域网的连接部署，并根据提供的配置要求对网络设备进行设置，以确保网络正常运行。

1、局域网的连接部署

路由器的管理地址为 `http://192.168.1.1` 或【指定地址】，如果无法进入路由器管理界面，参赛选手需自行将路由器重置成出厂设置，再访问管理地址并重新设定管理密码后，方可进入管理界面。

- 现场使用赛场中准备的网线将个人计算机与路由器相连，组成局域网，并确保整个网络畅通。网络连接好之后，参赛选手按照下表路由器的上网设置完成 WAN 口的配置。

序号	网络配置项	网络配置内容
1	WAN 口连接类型	固定 IP 地址
2	IP 地址	192.168.0.【工位号】
3	子网掩码	255.255.255.0
4	网关	192.168.0.254

- 按照下表的要求完成无线网络配置。

序号	网络配置项	网络配置内容
1	无线网络功能	关闭无线网络

- 按照下表的配置要求，通过对路由器 LAN 口 IP 设置，完成有线局域网的搭建。

序号	网络配置项	网络配置内容
1	LAN 口 IP 设置	手动
2	IP 地址	192.168.1.【工位号】
3	子网掩码	255.255.255.0

完成以上任务后做以下步骤：

- ◆ WAN 口配置完成后，将 WAN 口配置界面截图。要求截图中可以看到要求配置的信息，**将截图保存在答题卡任务二 WAN 口配置对应位置。**
- ◆ 无线配置完成后，将路由器关闭无线网络设置的界面截屏。要求截图中可以看到关闭了无线网络功能，**将截图保存在答题卡任务二无线配置对应位置。**
- ◆ LAN 口配置完成后，将路由器的 LAN 口配置界面截图。要求截图中可以看到要求配置的信息，**将截图保存在答题卡任务二 LAN 口配置对应位置。**

2、局域网各设备 IP 配置

- 按照下表的内容完成对局域网中各个网络设备 IP 地址、子网掩码、网关地址等的设定，并保证各个网络设备的通畅。各设备网络接口方式自行设定。

序号	设备名称	配置内容
1	个人计算机 1	IP 地址：192.168.1. 【工位号+1】
2	个人计算机 2	IP 地址：192.168.1. 【工位号+2】
3	边缘计算网关	IP 地址：192.168.1. 【工位号+3】

- 利用竞赛资料中提供的 IP 扫描工具，扫描检查局域网中各终端 IP 地址。

完成以上任务后做以下步骤：

- ◆ 将 IP 扫描结果截图，**将截图保存在答题卡任务二 IP 扫描结果对应位置。**

任务三：智慧城市中间件平台的使用和配置

场景描述：智慧城市中间件平台是智慧城市建设中的重要组成部分，用于管理和监控城市中的各种设备和传感器。本任务旨在使用浏览器登录智慧城市中间件平台，并完成项目创建、设备添加、终端设备绑定和 API 接口生成等操作。

任务描述：

使用浏览器输入 `http://192.168.1.100:8080/wziot/`，进入智慧城市中间件平台，进行如下操作：

登录平台：

使用用户名：student，密码“123456”，完成用户登录；

创建项目：自行创建项目类型为“智慧城市”项目；

添加设备：添加设备类型为“第三代网关”、序列号在边缘计算网关上的物联网关应用系统上方；

绑定终端设备：在第 3 步设备下，添加节点，选择 UUID 为 30001、30002、30003、30004 的节点，完成添加；

生成 API 接口：完成终端设备绑定后回到主页面进行刷新，可以看到设备数量以及传感器数量，确保设备为在线状态下生成 API 接口，使用 API 接口完成模块二智慧城市应用层开发；

选手完成平台配置后将平台首页（包含已创建项目）、设备管理页（包含已添加设备）、传感器管理页（包含已绑定传感器）、API 接口页（包含设备详情接口）分别截图，将截图分别保存在答题卡任务二智

慧城市中间件平台首页、设备管理页、传感器管理页、API 接口页对应位置。

任务四：智慧城市终端示例程序下载

场景描述：

在智慧城市的发展中，光照传感器被广泛应用于各类智能设备中，用于实时监测环境中的光照强度。通过对光照强度的监测，城市管理者可以及时调整路灯亮度、优化建筑设计、提升能源利用效率等。在本任务中，您将使用智慧城市技术通用竞赛平台上的光照传感器模块节点，以模拟监测城市环境中的光照强度。

任务描述：

在竞赛提供的智慧城市技术通用竞赛平台上找到“光照传感器”模块节点，并在个人计算机上执行以下操作：

使用竞赛提供的终端开发环境，利用 U 盘竞赛资料中提供的光照传感器工程代码对工程进行编译。

将编译后的程序下载到光照传感器对应的终端节点中。

确保程序正确编译并下载到相应模块后，加电保持运行状态，等待裁判评判。

任务五：智慧城市终端应用代码编辑

场景描述：在智慧城市中，手势控制系统在智慧城市中具有广泛的应用前景，可以提升城市设施的智能化水平，改善市民生活体验，促进城市可持续发展。本任务旨在根据提供的终端节点使用教程和基础示例代码，完成手势识别终端功能的配置。

任务描述：

1. 在竞赛平台上找到语音播报模块，根据提供的基础示例代码，编写并修改终端节点代码实现以下功能：要求终端可以播报“开灯”、“关灯”，当节点底板上按键，按下奇数次时，语音播报“开灯”；按下偶数次时，语音播报“关灯”；当智慧城市中间件平台通道 1 发送变量 1 时，语音播报“开灯”；当智慧城市中间件平台通道 1 发送变量 2 时，语音播报“关灯”；当智慧城市中间件平台通道 1 发送变量 3 时，语音播报“开风扇”；当智慧城市中间件平台通道 1 发送变量 4 时，语音播报“关风扇”。
2. 在竞赛平台上找到三色灯模块，根据提供的基础示例代码，编写并修改终端节点代码实现以下功能：要求上电后 RGB 灯散发白光，每按下一次节点底板上按键后，RGB 灯改变一次颜色，颜色在红色、黄色、绿色中轮换。当智慧城市中间件平台通道 1 发送变量 0 时，RGB 灯关闭；当智慧城市中间件平台通道 1 发送变量 1 时，RGB 灯散发红光；当智慧城市中间件平台通道 1 发送变量 2 时，RGB 灯散发绿光；当智慧城市中间件平台通道 1 发送变量 3 时，RGB 灯散发黄光；当智慧城市中间件平台通道 1 发送变量 4 时，RGB 灯散发蓝光；当智慧城市中

中间件平台通道 1 发送变量 5 时，RGB 灯散发紫光；当智慧城市中间件平台通道 1 发送变量 6 时，RGB 灯散发蓝绿光；当智慧城市中间件平台通道 1 发送变量 7 时，RGB 灯散发白光。

3. 在竞赛平台上找到继电器模块，根据提供的基础示例代码，编写并修改终端节点代码实现以下功能：要求当节点底板上按键，按下奇数次时，风扇被打开；按下偶数次时，风扇被关闭；当智慧城市中间件平台发送键值 1 时，风扇被打开；当智慧城市中间件平台发送键值 2 时，风扇被关闭。

完成上述功能后将程序代码烧写到对应的终端节点上，实现手势识别终端系统的配置，将所有编写修改的代码保存到竞赛 U 盘“提交资料\第四部分\任务五”目录中。

第五部分 智慧城市中的识别模型应用部署

任务一：智慧城市表情识别系统部署与优化

场景描述：表情识别系统是智慧城市中的一种智能化识别系统，它能够根据人脸，识别出人当前的表情的以此判断每个人当前的心情状态。此外外界环境的光照因素也实时影响着我们识别的准确率，本任务要求在竞赛设备上完成表情灯控系统的项目部署与优化。

任务描述：

根据“竞赛资料\第五部分\任务一”中提供的文件，按照下述步骤在竞赛设备上完成表情灯控系统的项目部署与优化。

1. 根据“AI 实训平台使用教程”，查看 AI 实训平台 IP 地址。
3. 根据“AI 实训平台使用教程”，使用指定软件将 AI 实训平台代码传输至 AI 实训平台，创建并保存在桌面的 test1 文件夹中。
4. 根据“AI 实训平台使用教程”，在 AI 实训平台上运行使用 rknn_transfer.py 将提供的 hdf5 模型转换为 rknn 模型，模型同样保存在桌面的 test1 文件夹中。
5. 根据“AI 实训平台使用教程”，在 AI 实训平台上运行 wz_biaoqing.py 调用转换好的 rknn 模型，在竞赛设备上实现摄像头检测人脸表情识别，并记录实验现象。
6. 优化修改代码，要求在检测人脸的同时右上角显示“实时光照：xxx”，xxx 为当前的实际光照值，要求数值能根据环境编号而变化。

完成上述步骤后，将所有修改过的代码保存到 U 盘“提交资料\第五部分\任务一”目录中，分析代码并将系统的效果现象记录在答题卡的指定位置上。

任务二：智慧城市手势控制系统部署

场景描述：手势控制系统是智慧城市中一种便捷的控制方式，能够通过手势识别实现对灯光和风扇等设备的开关控制。本任务要求在竞赛平台上完成手势控制系统的部署，包括对摄像头检测手势进行识别、使用语音播报模块进行提示、RGB 灯模块显示状态、点阵显示模块展示信息等功能的实现。

任务描述：

根据任务一中提供的示例代码，在竞赛平台上完成“手势控制系统”的功能实现及项目部署，实现对调用摄像头检测手势控制灯光及风扇的开关，并使用语音播报模块进行播报显示。

1. 参考任务一中 AI 实训平台的代码及模型转换代码，修改代码实现将“竞赛资料\第五部分\任务二”中提供的 Tensorflow 模型转换为 rknn 模型；编写代码调用 rknn 模型进行检测，实现在竞赛设备上调用摄像头对 0-9 数字手势的识别（允许一定识别误差）；在摄像头画面上方用黄色字体显示“手势控制系统”，在摄像头画面右上角，用

红色字体显示手势数字，代码传输至 AI 实训平台的桌面自行创建“test2”目录下运行。

2. 在竞赛平台上找到语音播报模块和 RGB 三色灯模块，在任务二的基础上，修改终端节点代码及 AI 平台手势检测代码实现以下功能的增加：当 AI 实训平台程序检测到手势 1 时，使用语音播报模块播报“开灯”，RGB 灯显示白灯；当 AI 实训平台程序检测到手势 2 时，使用语音播报模块播报“关灯”，RGB 灯显示熄灭。

3. 在竞赛平台上找到语音播报模块、风扇继电器模块，在任务二的基础上，修改终端节点代码及 AI 平台手势检测代码实现以下功能的增加：当 AI 实训平台程序检测到手势 3 时，使用语音播报模块播报“开风扇”，继电器控制风扇打开；当 AI 实训平台程序检测到手势 4 时，使用语音播报模块播报“关风扇”，继电器控制风扇关闭。

完成后，请保持手势控制系统程序的正常运行，以便竞赛结束后裁判进行评分，将所有编写修改的代码保存到竞赛 U 盘“提交资料\第五部分\任务二”的指定目录中。

第六部分 智慧城市中的文档设计

智慧城市的技术应用需要清晰的文档来指导使用和维护。本任务要求参赛者撰写关于智慧城市中系统的各类文档，以使用户能够理解和操作系统。

任务一：智慧城市终端设备配置说明文档设计

设计一份智慧城市终端设备配置说明文档，用于指导智慧城市项目中各种终端设备的配置和使用。文档应包含以下内容：

设备概述：对智慧城市项目中涉及的各种终端设备进行介绍，包括摄像头、显示终端、传感器节点等。

配置步骤：详细说明每种终端设备的配置步骤，包括网络连接、软件安装和参数设置等。

功能说明：说明每种终端设备的功能和用途，包括摄像头的监控功能、显示终端的信息展示和传感器节点的数据采集等。

使用注意事项：提供使用终端设备时需要注意的事项和常见问题解答，包括设备维护、安全防护和故障排除等。

示例应用：给出一些智慧城市项目中典型的终端设备应用示例，以帮助用户理解设备的实际应用场景。

任务二：智慧城市中间件平台操作手册设计

设计一份智慧城市中间件平台操作手册，用于指导智慧城市项目中间件平台的使用和配置。文档应包含以下内容：

登录与注册：说明用户如何注册账号并登录智慧城市中间件平台。

项目创建：指导用户如何创建智慧城市项目，并设置项目的类型、设备信息和接口规范等。

设备管理：详细介绍如何添加和管理各种设备，包括网关、节点和传感器设备等。

接口生成：提供生成 API 接口的方法和步骤，包括设备接口和传感器数据接口的生成与配置。

数据监控：说明如何监控和管理智慧城市项目中的数据流量和设备状态，以及如何进行分析数据和故障诊断等。

安全设置：提供设置平台账号和项目权限的方法和建议，以确保数据和设备的安全。

常见问题解答：收集用户常见问题，并提供相应的解答和解决方法，以便用户在使用平台时快速解决问题。