

2023 年河北省职业院校技能大赛

多信息智能识别控制的技术与应用（高职组）

竞赛样题

第一部分 智能识别应用

任务一：轮廓识别和角点检测

新建 Python 项目，通过编写代码运行实现以下内容

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。
3. 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图像特征-harris 角点检测，将角点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。

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。

任务二：OpenCV 图像识别

新建 Python 项目，通过编写代码运行实现以下内容

1. 加载竞赛提供的“竞赛资料\第一部分\任务二”中提供的 2 张图片，使用 OpenCV 颜色转换算法将图像转换为灰度图像。使用二值化算法将图像转换为二值图像，并检测出图像中的所有直线。将直线绘制到原始图像上，并将结果保存到“提交资料\第一部分\任务二\1”目录下。
2. 给定一个有噪声的黑白图像，将图像读取到 Python 中，并将其转换为灰度图像，将处理后的图像保存到“提交资料\第一部分\任务二\2”目录下。
3. 使用礼帽操作进行去噪处理，并将处理后的图片保存到“提交资料\第一部分\任务二\3”目录下。请在这个目录中提供处理前和处理后的图片。
4. 尝试调整礼帽操作的核大小并比较不同参数下的处理结果。将最终的核大小和输出结果保存到“提交资料\第一部分\任务二\4”目录下。请在文件夹中提供至少两种不同核大小的处理结果。

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 智能数据处理

任务一：OpenCV 图像处理

新建 Python 项目，编写代码调用 OpenCV 相关接口运行实现对“竞赛资料\第二部分\任务一”中提供的数据集的以下操作：

1. 加载竞赛提供的“竞赛资料\第一部分\任务二”中提供的视频，对其视频帧进行长宽缩放，要求将其调整为长宽为 640x360 像素的大小，并同名保存到“提交资料\第二部分\任务一\1”目录下。
2. 加载竞赛提供的“竞赛资料\第一部分\任务二”中提供的视频，根据视频帧中每个像素点的色调值进行像素颜色的实时转换，要求将原有的 RGB 颜色通道转换为 BGR 颜色通道，并将处理后的视频帧保存到“提交资料\第二部分\任务一\2”目录下。
3. 加载竞赛提供的“竞赛资料\第一部分\任务二”中提供的视频，使用 OpenCV 中的背景减除算法，以及 GMG 算法识别像素运动，将运动物体的轨迹在视频中用绿色线条绘制出，并同名保存到“提交资料\第二部分\任务一\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\任务一”目录下。

任务二：OpenCV 特征匹配

新建 Python 项目，对“竞赛资料\第二部分\任务二”中提供的图片进行以下操作：

1. 对“竞赛资料\第二部分\任务二”中提供的 2 张图像进行二值化操作，将二值化处理后的图片保存，并提交到“提交资料\第二部分\任务二\1”目录下。
2. 对“竞赛资料\第二部分\任务二”中提供的 2 张图像使用 sift 进行 1 对 1 特征匹配操作，将处理后的 2 张图片的特征点匹配并连线保存，并提交到“提交资料\第二部分\任务二\2”目录下。
3. 对“竞赛资料\第二部分\任务二”中提供的 2 张图像使用 sift 进行 K 对最佳匹配操作，将处理后的 2 张图片的特征点匹配并连线保存，并提交到“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 智能模型构建

任务一：手势识别模型参数设定

在“竞赛资料\第三部分\任务一”提供的 TensorFlow 模型构建代码中找到并修改设定以下参数：

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。

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：手势识别模型训练与优化

根据任务一修改完毕的代码，合理优化模型完成模型训练，训练要求如下：

1. 对提供的手势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. 对提供的手势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. 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\6”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：模型调用与预测

新建 Python 项目，通过编写代码实现以下功能

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”中提供的不同手势图片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 智能模型部署

任务一：表情灯控系统部署

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。

1. 在局域网内完成路由器网络的配置，设定无线网络名称为“JNDS-XX”，XX 为你的工位号（如 JNDS-01），八位密码自行设定；设定无线网络 DHCP 服务的开始地址为 192.168.x.100，结束地址为 192.168.x.199，x 为你的工位号（如 192.168.1.100–192.168.1.199）；完成后根据答题卡要求将 WiFi 名称、密码和设置截图填写在答题卡的指定位置上。
2. 根据“AI 实训平台使用教程”，查看 AI 实训平台 IP 地址并修改 `wz_biaoqing.py` 中的 HOST 的 IP 为 AI 实训平台 IP。
3. 根据“AI 实训平台使用教程”，使用指定软件将修改后的 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 实训平台上运行修改好的 `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。
6. 修改终端节点代码中的 SERVER 及 WiFi 名称和密码，其中 SERVER 为 AI 实训平台 IP, WiFi 名称和密码为当前 WiFi 信息。
7. 根据“终端节点使用教程”，将修改后的代码下载到 RGB 三色灯模块对应的终端节点上并运行代码。

完成上述步骤后，实现表情灯控系统联动，将所有修改过的代码保存到 U 盘“提交资料\第四部分\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：手势控制系统终端配置

根据任务一中提供的“终端节点使用教程”及基础示例代码，完成“手势控制系统”的终端功能实现，具体功能如下。

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“开风扇”、“关风扇”、“开灯”、“关灯”，当节点底板上按键，按下奇数次时，语音播报“开关灯”；按下偶数次时，语音播报“开关风扇”模拟灯光和风扇控制；
2. 在竞赛平台上找到点阵显示模块，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：要求终端可以点阵显示“开风扇”、“关风扇”、“开灯”、“关灯”，当节点底板上按键，按下奇数次

时，点阵显示“开关灯”；按下偶数次时，点阵显示“开关风扇”模拟灯光和风扇控制；

完成上述功能后将程序代码烧写到对应的终端节点上，实现手势控制终端系统的配置，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\任务二”目录中。

任务三：手势控制系统部署

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，并使用点阵显示模块和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\第四部分\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“test3”目录下运行。

2. 在竞赛平台上找到语音播报模块、RGB 三色灯模块和点阵显示模块，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，点阵显示屏显示“开灯”；

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，点阵显示屏显示“关灯”。

3. 在竞赛平台上找到语音播报模块、风扇继电器模块和点阵显示模块，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，点阵显示屏显示“开风扇”；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，点阵显示屏显示“关风扇”。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\第四部分\任务三”的指定目录中。

第五部分 智能文档编辑

完成对第四部分“表情灯控系统”和“手势控制系统”的使用说明编写，内容需包括“系统运行要求”、“系统功能介绍”、“系统运行步骤”，文档格式要求统一美观，字数不低于 100 字，完成后将文档命名保存为“表情灯控系统使用说明.docx”和“手势控制系统使用说明.docx”，提交保存至竞赛 U 盘“提交资料\第五部分”的指定目录中。