**2025 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：角点检测及图像分析**

**新建 Python 项目，通过编写代码运行实现以下内容**

1. 对“竞赛资料\第一部分\任务一”提供的图像文件执行中值滤波操作，以平滑图像并去除噪点。保存在“提交资料\第一部分\任务一\1”目录下。
2. 对上述去噪后的结果执行角点检测操作，并使用红色圆点标识图像中的关键角点。将标注后的图片保存到竞赛 U 盘“提交资料\第一部分\任务一\2”目录下。
3. 对“竞赛资料\第一部分\任务一”提供的图像文件使用calcHist 函数计算图像的直方图，分析图像的颜色分布情况。将计算后的直方图用 matplotlib 绘制出并保存在“提交资料\第一部分\任务一\3”目录下。
4. 对上述直方图检测后的图片执行直方图均值化操作，增强图像的对比度。保存所有处理后的图片到竞赛U盘“提交资料＼第一部分＼任务一＼4”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\任务一”目录下。**

**任务二：OpenCV 边界填充**

**新建 Python 项目，通过编写代码运行实现以下内容**

1. 将数据集中所有图片使用BORDER\_REPLICATE函数处理填充，将处理后的图片保存到“提交资料\第一部分\任务二\1”目录下。
2. 将数据集中所有图片使用BORDER\_REFLECT函数处理填充，将处理后的图片保存到“提交资料\第一部分\任务二\2”目录下。
3. 将数据集中所有图片使用BORDER\_REPLICATE函数处理填充，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。
4. 将数据集中所有图片使用BORDER\_WRAP函数处理填充，将处理后的图片保存到“提交资料\第一部分\任务二\4”目录下。
5. 将数据集中所有图片使用BORDER\_CONSTANT函数处理填充，将处理后的图片保存到“提交资料\第一部分\任务二\5”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 智能数据处理

**任务一：OpenCV 文本处理**

**新建 Python 项目，对“竞赛资料\第二部分\任务一”中提供的图进行以下操作：**

1. 对“竞赛资料\第二部分\任务**一**”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**一**\1”目录下。
2. 使用 OpenCV 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**一**\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**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”目录中。**

**任务二：OpenCV 视频处理**

**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\任务二”中提供的视频文件的以下操作：**

1. 加载竞赛提供的“竞赛资料\第二部分\任务二”中提供的视频文件“car．mp4”，使用MOG背景减除算法检测视频中运动的小车。并将结果保存到竞赛U盘“提交资料＼第二部分＼任务二＼1”目录下。
2. 对背景减除的结果使用开运算，去除检测结果中的噪声，将结果保存到竞赛U盘“提交资料＼第二部分＼任务二＼2”目录下。
3. 对去噪后的结果进行轮廓查找，使用蓝色矩形框对运动的小车进行标注，将处理后的视频保存到竞赛U盘“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”目录下。**

第三部分：智能模型构建

**任务一：构建车牌识别模型**

**任务说明：U盘\资料\第三部分\任务一（根据该目录给定资源完成以下任务）**

**任务描述：**

1. 调用给定数据集，对图片进行预处理操作，将尺寸调整为224\*244后，进行归一化、翻转、平移等操作。将编写的代码截图保存到“U盘\提交资料\答题卡.docx的指定位置上”。
2. 将数据集划分为训练集、验证集和测试集，保证划分的随机性和合理性。将编写的代码截图保存到“U盘\提交资料\答题卡.docx的指定位置上”。
3. 使用tensorflow构建VGG19模型对数据集进行构建，并建立适用于车牌识别的全连接层。将编写的代码截图保存到“U盘\提交资料\答题卡.docx的指定位置上”。
4. 使用交叉熵损失函数和Adam优化器进行训练。将编写的代码截图保存到“U盘\提交资料\答题卡.docx的指定位置上”。
5. 在测试集上评估模型的性能，计算准确率等指标。将编写的代码截图保存到“U盘\提交资料\答题卡.docx的指定位置上”。
6. 输出训练过程中的每一轮准确率值，并标注最高准确率值。将编写的代码截图保存到“U盘\提交资料\答题卡.docx的指定位置上”。
7. 确保模型在测试集上的准确率达到一定要求，不低于0.85。将编写的代码截图保存到“U盘\提交资料\答题卡.docx的指定位置上”。

**完成后，将训练后准确率最高的模型保存为model.h5文件。将编写的代码截图保存到“U盘\提交资料\答题卡.docx的指定位置上”。**

**任务二：预测车牌识别模型**

**任务说明：U盘\第三部分\任务二（根据该目录给定文件完成以下任务）**

**任务描述：**

1. 调用任务一训练的模型，将模型转换为tflite模型，并将模型上传到平台中。将编写的代码截图保存到“U盘\提交资料\答题卡.docx的指定位置上”。
2. 加载检测‘test.jpg’并用已上传的tflite模型对图片中的车牌进行标注。将编写的代码截图保存到“U盘\提交资料\答题卡.docx的指定位置上”。
3. 在预测后的手势图1片上添加预测的标签和对应的概率，并将图片保存到当前目录 的子文件夹finish文件夹中。并将编写的代码截图保存到“U盘\提交资料\答题卡.docx的指定位置上”。

**完成后，将完整的项目文件和代码保存到“U盘\提交资料\第三部分\任务二”中。**

第四部分：智能模型部署

**任务一：车牌检测控制系统部署**

**任务说明：U盘\第四部分\任务一（根据该目录给定文件完成以下任务）**

**任务描述：**

1. 调用“资料\第四部分\任务二”中的车牌识别模型，控制LCD1602显示器。
2. 开启摄像头对车牌进行检测，实现：

* 当检测到车牌时，识别出车牌的内容之后终端打印出识别的车牌
* 在LCD显示屏上显示出识别到的车牌信息。
* 对比车牌信息，是否为已知车牌，若为已知车牌亮起绿灯，打开门锁。若为未知车牌，则亮起红灯蜂鸣器响起。

1. 将运行结果以及代 码截图保存在“U盘\提交资料\答题卡.docx的指定位置上”。

**完成后，将代码文件保存到“U盘\提交资料\第四部分\任务二”中。**

**任务二：表情检测系统部署**

**任务说明：桌面\第四部分\任务二（根据该目录给定文件完成以下任务）**

**任务描述：**

1. 调用“资料\第四部分\任务一”中的表情识别模型，控制RGB三色灯。
2. 使用摄像头对手势进行实时监测，实现：

* 检测到愤怒时 RGB三色灯亮起红色。
* 检测到开心时，RGB三色灯亮起绿色。
* 检测到平静时，RGB三色灯亮起蓝色。
* 在检测到的图像中以对应颜色的框框出所检测到表情，并在框右上标出对应表情的标签

1. 将运行结果截图保存至“U盘\提交资料\答题卡.docx的指定位置上”。

**完成后，将代码文件保存到“U盘\提交资料\第四部分\任务一”中。**

第五部分 智能文档编写

完成对第四部分“车牌检测控制系统部署”和“表情检测系统部署”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“车牌识别系统使用说明.docx”和“表情检测系统.docx”，提交保存至“U盘\提交资料\第五部分\”