**人工智能训练师（三级）操作技能考核**

**试题单**

准考证号：

试题代码：

试题名称：花朵智能识别系统交互流程设计

考核时间：20min

**1.**场地设备要求

（1）人工智能训练师主机 1 台；   
（2）Python 编译环境；   
（3）Pytorch框架。

**2.**工作任务

花朵智能识别系统在现代城市绿化管理中起着越来越重要的作用，其利用先进的计算机视觉技术，如花朵检测与识别，实现了对花朵种类的实时监控与管理。本系统要求开发一个基于已训练模型的花朵检测与分类系统，能够准确识别出不同类别的花朵。  
AI模型说明：提供的模型“flower-detection.onnx”是使用 Pytorch 框架和基于深度卷积神经网络训练得到的，专门用于进行花朵识别。对应的标签文件为“labels.txt”。 该模型的使用交互流程为：  
1)加载模型“flower-detection.onnx”和加载类别标签“labels.txt”；  
2)加载一张本地花朵图片“flower\_test.png”，并预处理图像；  
3)使用flower-detection模型对花朵图片进行识别；  
4)输出花朵的预测类型和识别的准确率。  
你作为一名人工智能训练师，请完成以下工作任务：  
（1）补全该模型的使用交互流程对应的Python代码（3.2.4.ipynb），实现本地测试图片“flower \_test.png”的识别，将其识别结果截图保存为jpg格式文件，命名为3.2.4-1.jpg。  
（2）在上面的使用交互流程基础上，给出在花朵智能识别系统中使用“flower-detection.onnx”模型的一种人机交互的最优流程，将其保存为docx文件，命名为3.2.4.docx。  
所有结果文件储存在桌面新建的考生文件夹中，文件夹命名为“准考证号+身份证号后六位”。

**3.**技能要求

（1）能确保模型在单一场景下稳定运行；   
（2）能通过分析，设计单一场景下人工和智能交互的最优流程。

**4.**质量指标

（1）模型运行稳定，使用正常；   
（2）单一场景下人工和智能交互的最优流程切实可行。

**人工智能训练师（三级）操作技能考核**

**试题评分表**

准考证号：

试题代码：

试题名称：花朵智能识别系统交互流程设计

考核时间：20min

测量分评分表

| 细则编号 | 配分 | 评分细则描述 | 规定或  标称值 | 结果或  实际值 | 得分 |
| --- | --- | --- | --- | --- | --- |
| M1 | 2 | 模型加载代码正确得2分 | 根据数据 |  |  |
| M2 | 2 | 类别标签加载代码正确得2分 | 根据数据 |  |  |
| M3 | 2 | 从指定路径加载图片代码正确得2分； | 根据数据 |  |  |
| M4 | 2 | 预处理图片代码正确得2分； | 根据数据 |  |  |
| M5 | 2 | 使用模型对图片进行识别代码正确得2分； | 根据数据 |  |  |
| M6 | 2 | 获取识别分类后的准确率代码正确得2分； | 根据数据 |  |  |
| M7 | 2 | 获取预测的类别索引代码正确得2分； | 根据数据 |  |  |
| M8 | 2 | 获取预测的准确值（转换为百分比）代码正确得2分； | 根据数据 |  |  |
| M9 | 2 | 获取预测的类别标签代码正确得2分； | 根据数据 |  |  |
| M10 | 1 | 截图3.2.4-1.jpg中类别和准确率正确得1分； | 根据数据 |  |  |
| M11 | 1 | 人机交互最优流程（不少于3条）正确得1分； | 根据数据 |  |  |
| 合计配分 | 20 | 合计得分 | |  |  |

**参考答案**：

源代码（3.2.4.ipynb）：  
import onnxruntime as ort

import numpy as np

import scipy.special

from PIL import Image

# 预处理图像

def preprocess\_image(image, resize\_size=256, crop\_size=224, mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225]):

image = image.resize((resize\_size, resize\_size), Image.BILINEAR)

w, h = image.size

left = (w - crop\_size) / 2

top = (h - crop\_size) / 2

image = image.crop((left, top, left + crop\_size, top + crop\_size))

image = np.array(image).astype(np.float32)

image = image / 255.0

image = (image - mean) / std

image = np.transpose(image, (2, 0, 1))

image = image.reshape((1,) + image.shape)

return image

# 加载模型 2分

session = ort.InferenceSession('flower-detection.onnx')

# 加载类别标签 2分

with open('labels.txt') as f:

labels = [line.strip() for line in f.readlines()]

# 获取模型输入和输出的名称

input\_name = session.get\_inputs()[0].name

output\_name = session.get\_outputs()[0].name

# 加载图片 2分

image = Image.open('flower\_test.png').convert('RGB')

# 预处理图片 2分

processed\_image = preprocess\_image(image)

# 确保输入数据是 float32 类型

processed\_image = processed\_image.astype(np.float32)

# 进行图片识别 2分

output = session.run([output\_name], {input\_name: processed\_image})[0]

# 应用 softmax 函数获取识别分类后的准确率 2分

accuracy = scipy.special.softmax(output, axis=-1)

# 获取预测的类别索引

predicted\_idx = np.argmax(accuracy)

# 获取预测的准确值（转换为百分比）

prob\_percentage = accuracy[0, predicted\_idx] \* 100

# 获取预测的类别标签

predicted\_label = labels[predicted\_idx]

# 输出预测结果，包含百分比形式的概率

print(f"Predicted class: {predicted\_label}, Accuracy: {prob\_percentage:.2f}%")

**3.2.4-1.jpg： （注意考试时应按要求保存文件到指定文件夹，这里只是参考答案截图）**



**3.2.4.docx：（（注意考试时应按要求保存文件到指定文件夹，这里只是参考答案截图）**

花朵智能识别系统人机交互最优流程：

1. 一键启动：用户打开系统，自动加载模型与标签文件。
2. 图片上传：支持拖拽或点击上传本地花朵图片（如flower\_test.png）。

3.智能处理：后台自动完成图像预处理与模型推理。

4.即时反馈：界面直接显示识别结果（类别+百分比置信度）。

5.容错提示：非图片/无效文件时，引导用户重新上传。