**《Python高级应用》实验报告**

实验名称 ： 实验5其他应用

实验日期 ： 　 2024/11/24

姓 名 ： 　高心阳

学 号 ： 　 084623237

班 级 ： 　 计算机232

成 绩 ：

**人工智能与信息技术学院**

**南京中医药大学**

|  |
| --- |
| **实验目的：** |
| 1. 熟悉OpenCV库和pyttsx3库的使用； 2. 了解python架构的使用 |
| **实验内容和要求** |
| （1）使用pyttsx3等库实现从文本到语音的转换，并查阅资料，尝试将语音转化为文本。  （**\*（选做）**使用百度API等接口，实现人机交互））  （2）使用OpenCV库在图像中检测特定物体（如人脸、车辆、猫狗等），判断检测图像中是否包含目标对象，并标注出来。可以从简单的人脸检测开始。  （**\*（选做）**  **人脸表情识别**：识别图像中的人脸表情（如微笑、惊讶等）。  **车牌识别**：从图片中识别车牌并提取车牌号。  **手势识别**：识别简单的手势（如数字手势、OK手势等））  （3）尝试使用Flask架构实现用户登录。  （**\*（选做）**用户注册**、**连接数据库）） |
| **运行结果（写清题号）** |
| |  | | --- | | 1 | | import pyttsx3  def text\_to\_speech(text):  engine = pyttsx3.init()  engine.say(text)  engine.runAndWait()  text\_to\_speech("你好！Hello World!") | | 无输出 | | 2 | | import cv2 def detect\_faces(image, output="output.jpg"):  face\_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade\_frontalface\_default.xml')   image = cv2.imread(image)  if image is None:  print("无法加载图像，请检查路径！")  return   gray = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)   faces = face\_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5, minSize=(160, 160))   print(f"检测到 {len(faces)} 张人脸。")   for (x, y, w, h) in faces:  cv2.rectangle(image, (x, y), (x + w, y + h), (0, 255, 0), 2)   cv2.imwrite(output, image)  print(f"标注后的图像已保存至 {output}")  input\_image = "../test\_image.jpg" detect\_faces(input\_image) | | output检测到 12 张人脸。  标注后的图像已保存至 output.jpg | | 3 | | from flask import Flask, render\_template, request, redirect, url\_for, flash, session  app = Flask(\_\_name\_\_) app.secret\_key = 'your\_secret\_key'  # 模拟用户数据库 users = {"test\_user": "test\_password"}  @app.route('/') def home():  return render\_template('login.html')  @app.route('/login', methods=['POST']) def login():  username = request.form['username']  password = request.form['password']   # 验证用户名和密码  if username in users and users[username] == password:  session['username'] = username  flash('登录成功！', 'success')  return redirect(url\_for('dashboard'))  else:  flash('用户名或密码错误！', 'danger')  return redirect(url\_for('home'))  @app.route('/dashboard') def dashboard():  if 'username' in session:  return f"欢迎，{session['username']}！<br><a href='/logout'>注销</a>"  else:  flash('请先登录！', 'warning')  return redirect(url\_for('home'))  @app.route('/logout') def logout():  session.pop('username', None)  flash('已注销成功！', 'info')  return redirect(url\_for('home'))  app.run(debug=True) | | 屏幕截图 2024-11-24 204422  \* Serving Flask app '3-app'  \* Debug mode: on  WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.  \* Running on http://127.0.0.1:5000  Press CTRL+C to quit  \* Restarting with stat  \* Debugger is active!  \* Debugger PIN: 420-532-995  127.0.0.1 - - [24/Nov/2024 20:43:55] "GET / HTTP/1.1" 200 -  127.0.0.1 - - [24/Nov/2024 20:43:55] "GET /favicon.ico HTTP/1.1" 404 -  127.0.0.1 - - [24/Nov/2024 20:43:59] "POST /login HTTP/1.1" 302 -  127.0.0.1 - - [24/Nov/2024 20:43:59] "GET / HTTP/1.1" 200 -  127.0.0.1 - - [24/Nov/2024 20:44:10] "POST /login HTTP/1.1" 302 -  127.0.0.1 - - [24/Nov/2024 20:44:10] "GET /dashboard HTTP/1.1" 200 - | |
| **实验的体会与建议** |
|  |