

基于 YOLOv11 的动物识别系统

1. 代码实现目标

本系统旨在提供一个基于 PyQt5 和 YOLOv11 的动物识别解决方案，主要实现以下目标：

1. 加载预训练的 YOLOv11 模型（yolo11n.pt）进行动物检测
2. 提供用户友好的图形界面，支持多种检测模式
3. 实现图像检测功能，显示原始图像和检测结果
4. 实现视频文件检测功能，实时显示检测结果
5. 实现摄像头实时检测功能，支持多摄像头选择
6. 提供检测参数调整功能（置信度阈值）
7. 支持检测结果保存功能
8. 显示检测统计信息和性能指标

2. 设计思路

系统采用模块化设计，主要分为以下几个模块：

1. **主界面模块**：基于 PyQt5 的 QMainWindow 构建，采用选项卡布局管理不同功能
2. **图像处理模块**：负责图像加载、处理和检测结果显示
3. **视频处理模块**：使用多线程技术实现视频文件的实时检测和显示
4. **摄像头模块**：使用多线程技术实现摄像头实时画面捕获和检测
5. **YOLO 模型模块**：使用 PyTorch Hub 加载预训练的 YOLOv5 模型进行检测
6. **UI 美化模块**：使用 QSS 样式表美化界面

系统采用 MVC（Model-View-Controller）设计模式：

- Model: YOLOv5 模型和检测逻辑
- View: PyQt5 构建的用户界面
- Controller: 事件处理函数和信号槽机制

3. 主要功能

3.1 图像检测功能

- 支持打开常见图像格式 (JPG、PNG、BMP)
- 显示原始图像和检测结果对比
- 可调整置信度阈值
- 显示检测统计信息 (检测到的动物种类和数量)
- 显示检测耗时
- 支持保存检测结果

3.2 视频检测功能

- 支持打开常见视频格式 (MP4、AVI、MOV)
- 播放/暂停/停止控制
- 实时显示检测结果
- 显示播放进度条
- 显示帧率信息
- 可启用/禁用实时检测
- 支持保存检测结果 (概念实现)

3.3 实时摄像头检测

- 支持多摄像头选择
- 实时显示摄像头画面
- 实时检测画面中的动物
- 可保存当前画面快照
- 显示实时帧率

3.4 通用功能

- 置信度阈值调整 (所有检测模式)
- 检测统计信息显示
- 检测性能指标显示
- 响应式界面设计

4. 技术特点

1. **多线程处理**: 使用 QThread 处理视频和摄像头数据, 避免界面冻结
2. **实时性能**: 通过优化图像处理流程, 实现实时检测
3. **模块化设计**: 各功能模块独立, 便于维护和扩展
4. **用户友好界面**: 采用现代 UI 设计, 提供直观的操作体验
5. **跨平台支持**: 可在 Windows、macOS 和 Linux 系统上运行
6. **参数可配置**: 置信度阈值可实时调整, 满足不同检测需求

5. 使用说明

1. 确保已安装所需依赖: PyQt5, OpenCV, PyTorch
2. 将训练好的 yolo11n.pt 模型文件放在程序同目录下
3. 运行程序: bash
3. python animal_detection.py
4. 使用图像检测功能:
 - 切换到"图像检测"选项卡
 - 点击"打开图像"按钮选择图片
 - 点击"检测图像"进行检测
 - 点击"保存结果"保存检测后的图片
5. 使用视频检测功能:
 - 切换到"视频检测"选项卡
 - 点击"打开视频"按钮选择视频文件
 - 点击"播放"开始检测
 - 使用滑块调整置信度阈值
6. 使用实时检测功能:
 - 切换到"实时检测"选项卡
 - 选择摄像头设备
 - 点击"开始摄像头"开启实时检测
 - 点击"保存快照"保存当前画面

6. 系统依赖环境

- Python 3.8
- PyTorch 2.1.0
- OpenCV-Python 4.1.0
- PyQt5 5.15
- CUDA 12.1

本系统提供了一个完整的动物检测解决方案，结合了先进的 YOLOv5 目标检测算法和用户友好的 PyQt5 界面，能够满足各种动物识别应用场景的需求。