## 软件说明书：火焰检测系统软件说明书

### （一）、软件概述

本火焰检测系统是一个基于深度学习技术的应用程序，旨在通过计算机视觉算法对图像和视频中的火焰进行自动检测。系统结合了 YOLOv5 目标检测算法、PyTorch 深度学习框架、OpenCV 图像处理库和 PyQt5 图形用户界面库，为用户提供了一个直观、易用的平台，可方便地对各种来源的图像和视频进行火焰检测，帮助用户在不同场景下实现对火焰的识别和监测，可广泛应用于消防安全、工业监控、森林保护等领域。

### （二）、系统要求

**① 硬件要求**

* **处理器**：建议使用至少为 Intel Core i5 或 AMD 同等性能的处理器，对于大规模图像和视频处理或使用 GPU 加速，推荐使用更高性能的处理器，如 Intel Core i7 或 AMD Ryzen 7 及以上。
* **内存**：最低 4GB 内存，为保证系统流畅运行，尤其是处理高分辨率图像和视频时，建议使用 8GB 或更多内存。
* **显卡**：对于使用 GPU 加速的情况，需要 NVIDIA 或 AMD 的独立显卡，并安装相应的 GPU 驱动程序。NVIDIA 显卡建议使用 CUDA 兼容的 GPU，以支持 PyTorch 的 GPU 加速功能。对于仅使用 CPU 进行计算的情况，集成显卡也可满足基本需求，但处理速度可能较慢。
* **存储**：至少有 2GB 的可用硬盘空间用于存储软件、模型文件和可能的检测结果。如果需要存储大量的图像、视频和检测数据，建议预留更多空间。

**② 软件要求**

* **操作系统**：支持 Windows 7 及以上版本。

### (三)、软件使用

**（一）启动软件**

打开命令行终端，切换到存储 GUI.py 文件的目录。

输入以下命令启动软件：python GUI.py

等待片刻，系统将启动图形用户界面（GUI），用户将看到一个标题为 “火焰识别检测” 的窗口。

用户也可以直接打开打包好的“GUI.exe”应用程序，直接开始使用。

**（二）界面操作**

* **主界面**：
  + 软件启动后，用户将看到一个窗口，其大小约为 800x600 像素，包含以下主要组件：
    - 一个大的显示区域，占据窗口的大部分空间，用于显示用户上传的图像或视频的原始内容和检测结果。该区域将显示一个占位信息，如 “请上传文件进行火焰检测”，直到用户上传文件并完成检测。
    - 一个 “上传图片” 按钮，位于窗口底部或侧边，用于触发文件选择操作。

**（三） 文件上传**

1. 点击 “图片检测” 按钮，将弹出文件选择对话框。
2. 在文件选择对话框中，用户可以浏览本地文件系统，选择需要检测的文件。支持的文件类型包括图像文件（如 .jpg, .png）和视频文件（如 .mp4, .avi）。
3. 对于图像文件，用户可以选择单张图像进行检测；对于视频文件，系统将对视频中的每一帧进行火焰检测。

### （四）火焰检测

1. 当用户选择文件并确认后，系统将自动开始火焰检测过程。
   * 对于图像文件：
     + 系统会将图像加载到内存中，并将其传递给 YOLOv5 模型进行检测。
     + 检测过程中，用户可能会看到鼠标指针变为加载状态（根据操作系统不同可能有所不同），表示系统正在处理图像。
     + 经过模型推理和后处理后，检测到的火焰区域将在显示区域中显示，用绿色矩形框标记火焰的边界，同时在矩形框附近显示该检测结果的置信度，如 Conf: 0.85 表示置信度为 85%。
   * 对于视频文件：
     + 系统将逐帧读取视频，并对每一帧进行火焰检测。
     + 显示区域将实时显示视频帧，并在帧上绘制检测到的火焰边界框和置信度信息。
     + 系统将按照视频的原始帧率播放处理后的视频，用户可以看到带有火焰检测结果的视频播放，就像正常播放视频一样，但会有火焰检测的标记。

### （五）结果查看和保存

* **查看结果**：
  + 对于图像文件，检测结果将直接显示在显示区域，用户可以看到图像上标记出的火焰区域。
  + 对于视频文件，用户可以看到视频的播放，并观察到火焰在视频中的出现位置和相应的置信度信息。
* **保存结果**：
  + 目前系统暂不提供自动保存结果的功能，但用户可以使用操作系统的截图或录屏功能保存检测结果。
  + 对于开发者或高级用户，可以在代码中添加保存功能，例如使用 OpenCV 的 cv2.imwrite 函数保存检测后的图像，或使用 cv2.VideoWriter 保存带有检测结果的视频。

### （六）、功能说明

#### （一）图像检测

* **操作步骤**：
  1. 点击 “图片检测” 按钮。
  2. 选择所需的图像文件。
  3. 等待系统完成检测，查看显示区域中的结果。

#### （二）视频检测

* **操作步骤**：
  1. 点击 “视频检测” 按钮。
  2. 选择所需的视频文件。
  3. 系统将自动播放视频并实时显示火焰检测结果。

#### （三）性能调整

* **模型选择**：
  + 在代码中，用户可以通过修改加载的 YOLOv5 模型的权重文件来选择不同的模型，例如使用 yolov5s.pt （速度较快但精度稍低）、yolov5m.pt、yolov5l.pt 或 yolov5x.pt （精度更高但速度较慢）。修改以下代码行：model = DetectMultiBackend('path/to/your\_preferred\_weight.pt', device=device)
  + 将 path/to/your\_preferred\_weight.pt 替换为所需的权重文件路径。
* **参数调整**：
  + 对于开发者或高级用户，可以在代码中修改 YOLOv5 的参数，如置信度阈值和交并比（IoU）阈值，以调整检测的精度和召回率。这些参数可以在调用 non\_max\_suppression 函数时修改。

### （七）、常见问题及解决方法

#### （一）软件无法启动

* **问题描述**：输入启动命令后，系统无响应或报错。
* **可能原因及解决方法**：
  + 未安装所需的依赖库：检查是否已正确安装 PyTorch、OpenCV、PyQt5 等依赖库，可重新运行相应的 pip 安装命令。
  + Python 版本问题：确保使用的是 Python 3.6 或更高版本，可通过 python --version 命令检查 Python 版本。
  + 缺少 YOLOv5 代码或权重文件：确认已从官方仓库下载 YOLOv5 代码和所需的权重文件，并将其放在正确的目录中，确保代码中文件路径的设置正确。

#### （二）文件无法上传

* **问题描述**：点击 “上传图片” 按钮后，文件选择对话框不弹出或无法选择文件。
* **可能原因及解决方法**：
  + 权限问题：检查是否具有对文件系统的访问权限，尤其是在某些安全限制的环境下，可能需要管理员权限。
  + 系统错误：可能是 PyQt5 的 QFileDialog 出现异常，尝试重新启动软件或检查 PyQt5 安装是否完整。

#### （三）检测结果不准确

* **问题描述**：火焰未被检测到或误将非火焰区域检测为火焰。
* **可能原因及解决方法**：
  + 模型训练不足：考虑使用更多的训练数据对 YOLOv5 模型进行微调，或尝试不同的预训练权重。
  + 环境因素：在复杂的环境中（如低光照、强噪声、遮挡），模型可能表现不佳。可以考虑使用图像增强技术对输入图像进行预处理，或在不同的环境条件下收集更多数据进行训练。

#### （四）系统性能不佳

* **问题描述**：检测速度慢或界面卡顿。
* **可能原因及解决方法**：
  + 硬件性能限制：使用低性能的 CPU 或未启用 GPU 加速。确保 GPU 驱动已安装，并在代码中正确设置使用 GPU。
  + 高分辨率文件：对于高分辨率的图像和视频，处理时间会较长。可在代码中对输入图像和视频进行降采样处理，降低分辨率，例如在图像读取时使用较小的尺寸：

### 七、注意事项

1. 在使用 GPU 加速时，确保 GPU 设备有足够的显存，否则可能会出现显存不足的错误，导致程序崩溃或性能下降。
2. 对于大规模的数据处理或长时间运行，建议定期保存检测结果，防止因意外情况（如程序崩溃）导致数据丢失。
3. 对于不同的操作系统，文件路径的表示方式可能不同，在代码中使用 os.path 或 pathlib.Path 进行路径操作，避免使用硬编码的路径，确保跨平台兼容性。
4. 本系统目前仅为一个基础的火焰检测系统，对于复杂的实际应用场景，可能需要进一步的定制和开发，例如集成更多的功能、优化性能和提高检测精度。

通过这份软件说明书，用户可以清晰地了解火焰检测系统的安装、使用方法，以及可能遇到的问题和相应的解决办法，帮助用户更好地使用该软件完成火焰检测任务，同时为开发人员对系统的进一步开发和优化提供了参考。