WD-Derainer V1.0

用户说明书

**1、引言**

**1.1 编写目的**

本文档旨在为“WD-Derainer”软件提供一份详细的用户说明书，帮助用户了解软件的功能、使用方法及安装步骤，同时为开发者和技术人员提供必要的背景信息和技术支持参考。通过本说明书，用户能够快速掌握软件的操作流程，充分发挥其在图像去雨处理中的作用；开发者则可以通过文档了解软件的设计理念和技术特点，以便进行后续维护或功能扩展。

**1.2 开发背景**

随着计算机视觉技术的快速发展，图像处理在日常生活和工业应用中的重要性日益凸显。然而，恶劣天气（如雨天）对图像质量的影响成为许多视觉任务（如自动驾驶、监控系统、摄影美化等）的瓶颈。雨滴会导致图像模糊、细节丢失，严重影响后续分析和使用。因此，开发一款高效、易用的图像去雨工具，不仅能满足图像预处理的实际需求，还能为数据集清洗、图像美化等领域提供支持。

基于此背景，我们开发了“WD-Derainer”软件。该软件利用深度学习技术，特别是创新的 WDNet 模型，结合小波变换、自适应部分卷积、自适应LeakyReLU激活函数以及变形采样技术，旨在实现高效的批量图像去雨滴处理。通过提供图形用户界面（GUI）和命令行工具，我们希望为不同层次的用户（从普通用户到专业开发者）提供便捷的使用体验。

**1.3 定义**

在本说明书中，以下术语具有特定含义：

**系统、本系统、该系统:** 指“WD-Derainer”软件，包括其核心去雨模型（WDNet）、图形用户界面以及相关脚本。

**用户:** 指使用“WD-Derainer”软件进行图像去雨处理的人员，包括普通用户（如摄影爱好者）、研究人员和开发者。

**去雨处理:** 指通过算法移除图像中的雨滴，恢复清晰图像的过程。

**WDNet:** 指本系统使用的核心深度学习模型，全称为“Wavelet-based Deformable Network”。

**2、系统概述**

**2.1 系统用途**

“WD-Derainer”是一款专为图像去雨滴处理设计的软件，集成了深度学习模型推理、批量图像处理和部分结果可视化功能。该系统旨在帮助用户高效移除图像中的雨滴，恢复清晰的图像内容，适用于多种场景，包括但不限于：

1. 图像美化: 提升雨天拍摄照片的视觉质量，适用于摄影爱好者和专业摄影师。
2. 数据集预处理: 为计算机视觉任务（如目标检测、图像分割）提供干净的训练数据。
3. 实时应用支持: 为需要高质量图像输入的系统（如监控、自动驾驶）提供预处理支持。
4. 通过提供命令行工具和图形用户界面（GUI），本系统满足了不同用户的需求，既适合快速批量处理，也支持直观的交互式操作。

**2.2 系统功能及特点**

WD-Derainer总工作流程如图1所示。系统的工作流程可分为三个阶段：

（1）初始化阶段：系统首先调用数据加载器加载训练和测试数据集（可能涉及数据预处理或分批次加载），随后实例化WDNet模型并检查是否有预训练模型，若有则直接加载，完成模型准备。

（2）训练阶段：系统启动训练过程，训练从数据加载器获取数据并调用WDNet模型进行迭代学习，最终保存训练好的模型参数。

（3）测试阶段：系统调用测试过程，加载已保存的模型，从数据加载器获取独立测试数据集进行推理，WDNet返回预测结果，最终系统输出性能指标以验证泛化能力。

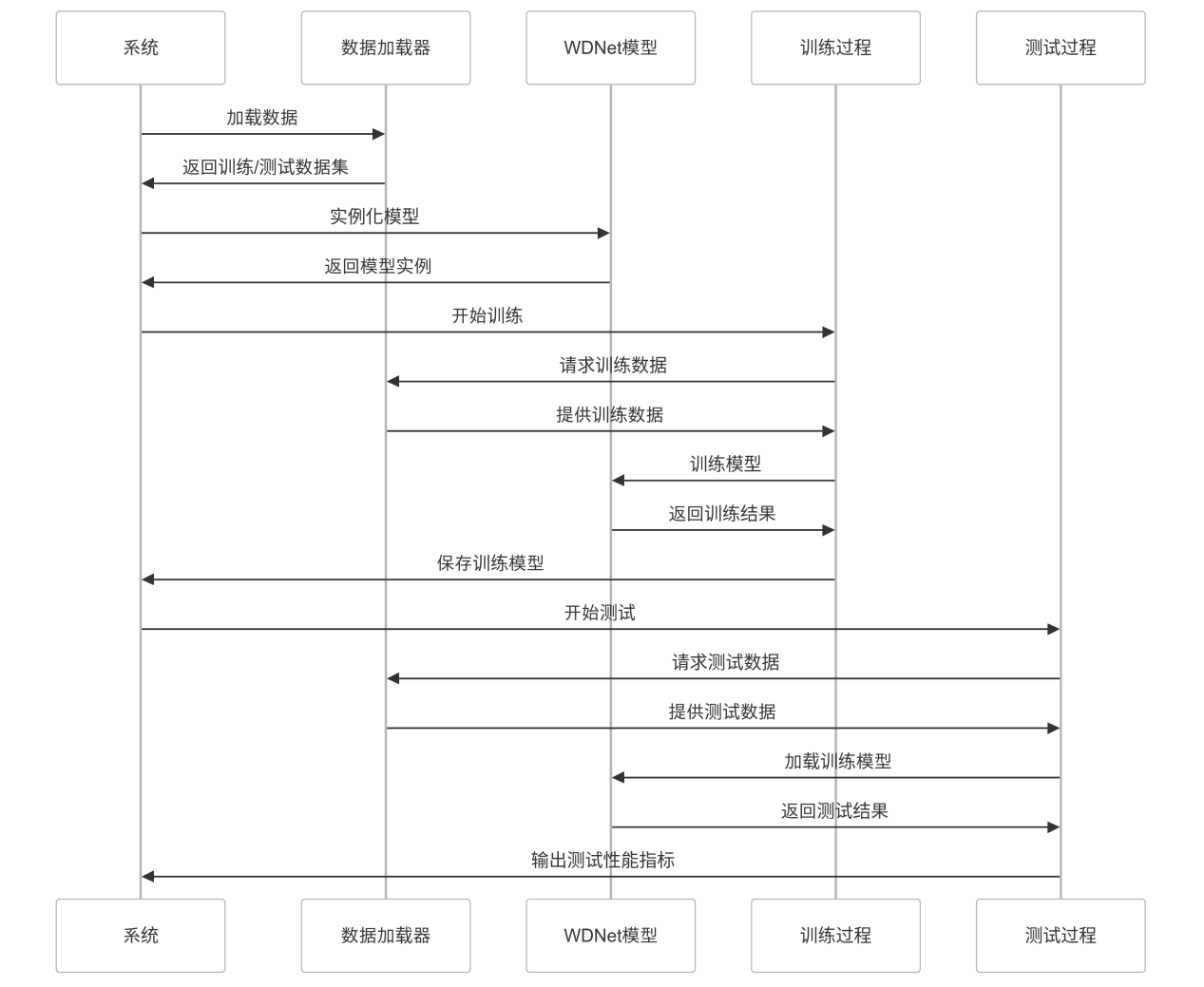


图 1 WD-Derainer总工作流程

“WD-Derainer”系统的主要功能包括：

1. **批量图像去雨处理:** 支持对指定文件夹中的多张图像进行自动去雨处理，结果保存到指定目录。
2. **去雨效果演示:** 提供 GUI 界面，随机展示单张图像的去雨前后对比，便于用户直观评估效果。
3. **进度监控:** 在批量处理过程中显示进度条和实时状态（如处理速度、剩余时间），提升用户体验。
4. **模型训练支持:** 提供训练脚本，允许用户基于自定义数据集优化去雨滴模型。
5. **去雨滴结果保存:** 处理后的图像以原始格式保存，便于后续使用。

“WD-Derainer”的核心模型 WDNet采用创新的编码器-瓶颈-解码器架构，结合小波变换、注意力机制和多尺度可变形采样技术，实现了高效、精确的图像去雨处理。其技术特点如下：

1. Haar Wavelet Separate Attention Block (HWSAB)

HWSAB 是编码器中的关键模块，通过 Haar 小波变换将输入特征图分解为低频分量（捕捉全局结构）和三个高频分量（水平、垂直、对角，捕捉雨滴的方向性特征）。

对低频分量，使用 1×1 卷积核结合通道注意力机制，保留全局背景信息。对高频分量，分别应用方向性卷积核（水平分量用 1×5，垂直分量用 5×1，对角分量用 3×3），通过空间注意力机制增强雨滴特征的提取。分解后的特征图经过独立处理后沿通道维度拼接，再通过 3×3 卷积层融合，输出分辨率减半、通道数加倍的特征图。相比传统方法直接融合所有频率分量，HWSAB 的分离处理方式能更精准地区分雨滴和背景细节，提升细节恢复能力。

1. Adaptive Partial Convolution-Adaptive Activation Mixer (APC-AAM)

瓶颈层通过 APC 和 AAM 模块优化特征表达，减少冗余并保留关键信息。自适应部分卷积APC根据通道特征图的方差排序，动态选择前 25%（可调参数 p=0.25）的活跃通道进行卷积，未选中通道保持不变，最终拼接输出。其可以减少对低信息量区域的计算，提升效率和表达能力。自适应激活混合器AAM通过空间权重生成器和通道权重生成器，计算注意力权重，融合原始输入和 ReLU 激活输出。其避免了传统 ReLU 对负值的硬性截断，动态保留有价值信息。APC-AAM协同工作，使瓶颈层在压缩特征的同时保持高质量的细节表达。

1. Multi-Scale Deformable Sampling Block (MS-DSB)

解码器中的 MS-DSB 通过多尺度偏移生成器和可变形采样，动态调整采样位置，恢复雨滴引起的失真区域。输入特征经过多尺度卷积（3×3、5×5 等），拼接后通过 TanH 激活和双线性插值生成双通道偏移图，两个通道分别表示x轴和y轴方向的偏移。然后，使用偏移图调整转置卷积的采样点，生成上采样特征图。最后，将采样结果与编码器对应层的跳跃连接特征拼接，进一步通过卷积层、批归一化层和ReLU激活层进行优化。相比传统固定采样，MS-DSB 的动态调整能力能够提升对复杂雨滴失真的恢复效果。

此外，它支持 GPU 加速，利用 PyTorch 深度学习框架实现高效推理，同时训练过程包含损失爆炸检测和自动恢复机制，确保模型的鲁棒性与稳定性；系统灵活性高，支持多种图像格式（PNG、JPG、JPEG），并通过预处理变换适配不同分辨率图像；此外，其用户友好的 GUI 界面操作简单，搭配适合开发者的命令行工具，兼顾了便捷性和专业性。

**2.3 编程语言**

Python（版本 3.8 或更高）

**3、系统安装**

**3.1 运行环境**

**1）硬件环境**

**最低要求：**

处理器： Intel Core i5-12500H

内存： 512 MB RAM

硬盘： 256 MB 空闲硬盘空间

显示卡：英伟达 RTX 2050

**建议配置：**

处理器： Intel Xeon Platinum 8362

内存： 1GB RAM

硬盘： 512MB 空闲硬盘空间

显示卡：英伟达 RTX 3090

**2）软件环境**

**最低要求：**

操作系统：Windows 10，Ubuntu 22.04

英伟达显示卡支持的CUDA版本：CUDA 12.1

**建议配置：**

操作系统：Windows 10或更高版本，Ubuntu22.04 或更高版本

英伟达显示卡支持的CUDA版本：CUDA 12.3

**软件第三方依赖库及版本：**

|  |  |  |
| --- | --- | --- |
| 序号 | 第三方依赖库名称 | 版本号 |
| 1 | torch | 2.2.1 |
| 2 | pytorch-wavelets | 1.3.0 |
| 3 | PyWavelets | 1.5.0 |
| 4 | pillow | 10.2.0 |
| 5 | torchvision | 0.17.1 |
| 6 | tkinter | 0.1.0 |
| 7 | numpy | 1.26.4 |
| 8 | tqdm | 4.66.2 |
| 9 | scikit-image | 0.22.0 |

**3.2 系统安装**

首先，安装所有第三方依赖库。

然后，构建图2所示文件组织结构。其中，datas文件夹用于存放训练用数据集；imgs文件夹用于存放用户需要进行去雨滴的图像；results文件夹用于存放用户提供的图片去雨滴后的结果；model文件夹下的\_\_init\_\_.py文件内容为空，用于标记model文件夹为包，wdnet.py为WDNet深度学习模型框架代码文件；dataLoader.py为数据集加载器，在模型训练时使用；test.py为单图像去雨测试程序，提供函数调用接口；train.py为训练程序；main.py为GUI界面程序，其调用了test.py中的图像去雨滴的函数接口为用户提供图像去雨滴功能。

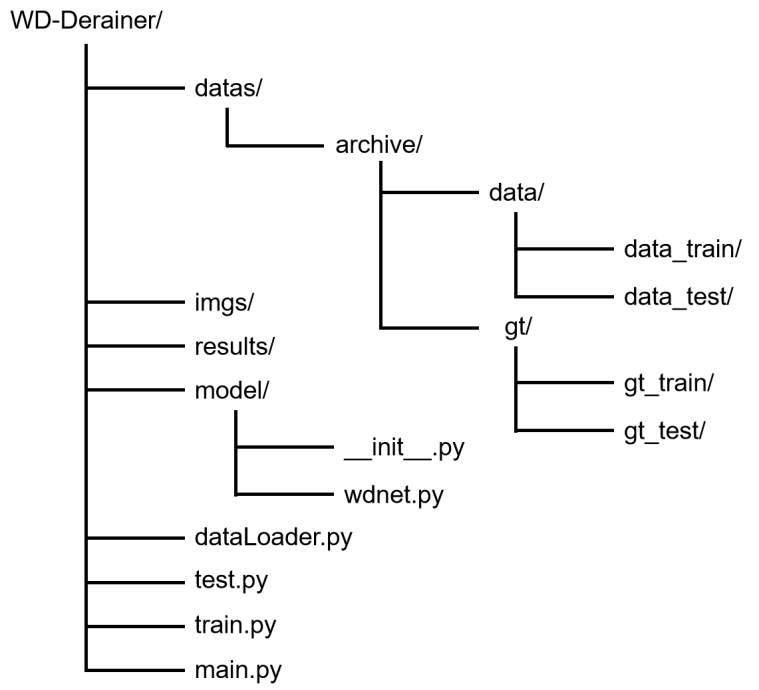


图 2 WD-Derainer代码文件组织结构

**4、软件功能**

**4.1 图形用户界面（GUI）**

首先，进入WD-Derainer目录，在命令行输入python main.py即可启动程序，出现如图2所示的界面。

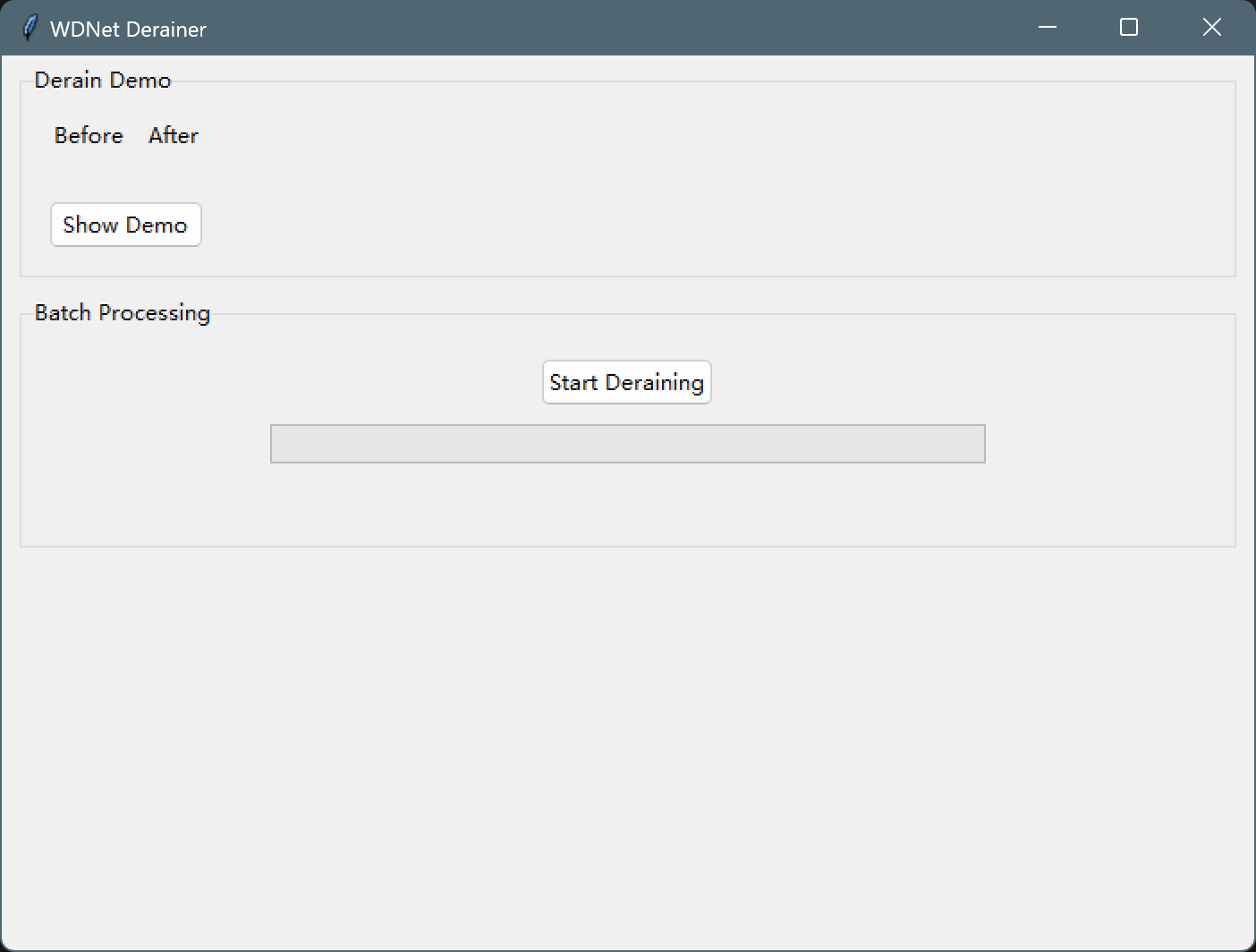


图 3 初始界面

**4.2 图像去雨滴单图测试功能**

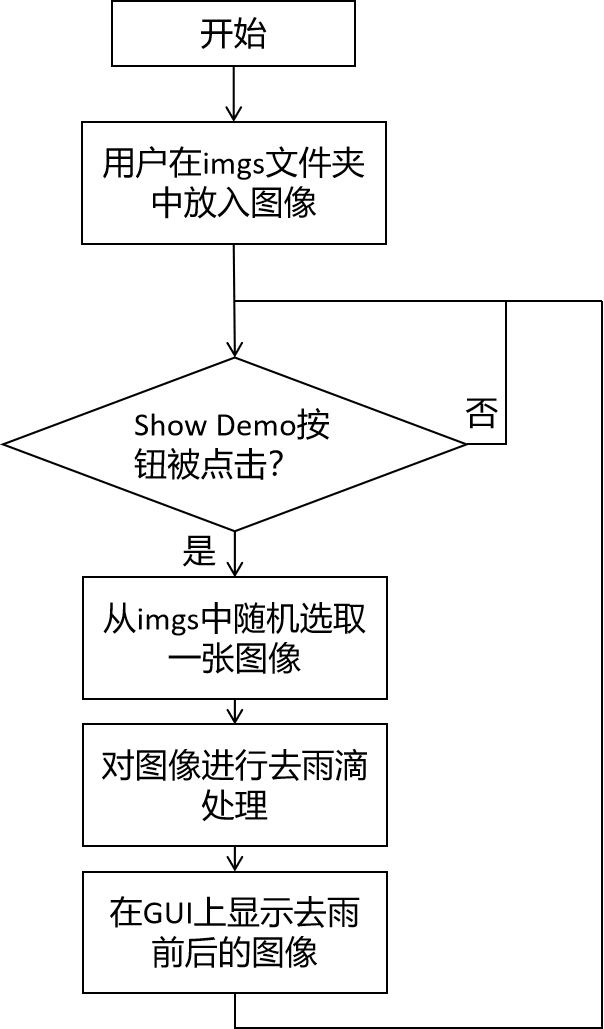


图 4 图像去雨滴单图测试流程图

系统的图像去雨滴单图测试功能流程如图4所示。用户在imgs文件夹中放入需要进行去雨滴处理的图像后，可以先点击“Derain Demo”区域中的“Show Demo”按钮。然后，程序就会从imgs文件夹中随机选取一张图像进行去雨并可视化在“Before”和“After”下方显示，如图3所示。如果想要重新选取一张图进行测试，可以继续点击“Show Demo”按钮，程序将重新随机在imgs文件夹选取一张图像进行雨滴去除并可视化在GUI上。注意，图像去雨滴测试并不会将测试图像保存至results文件夹中。

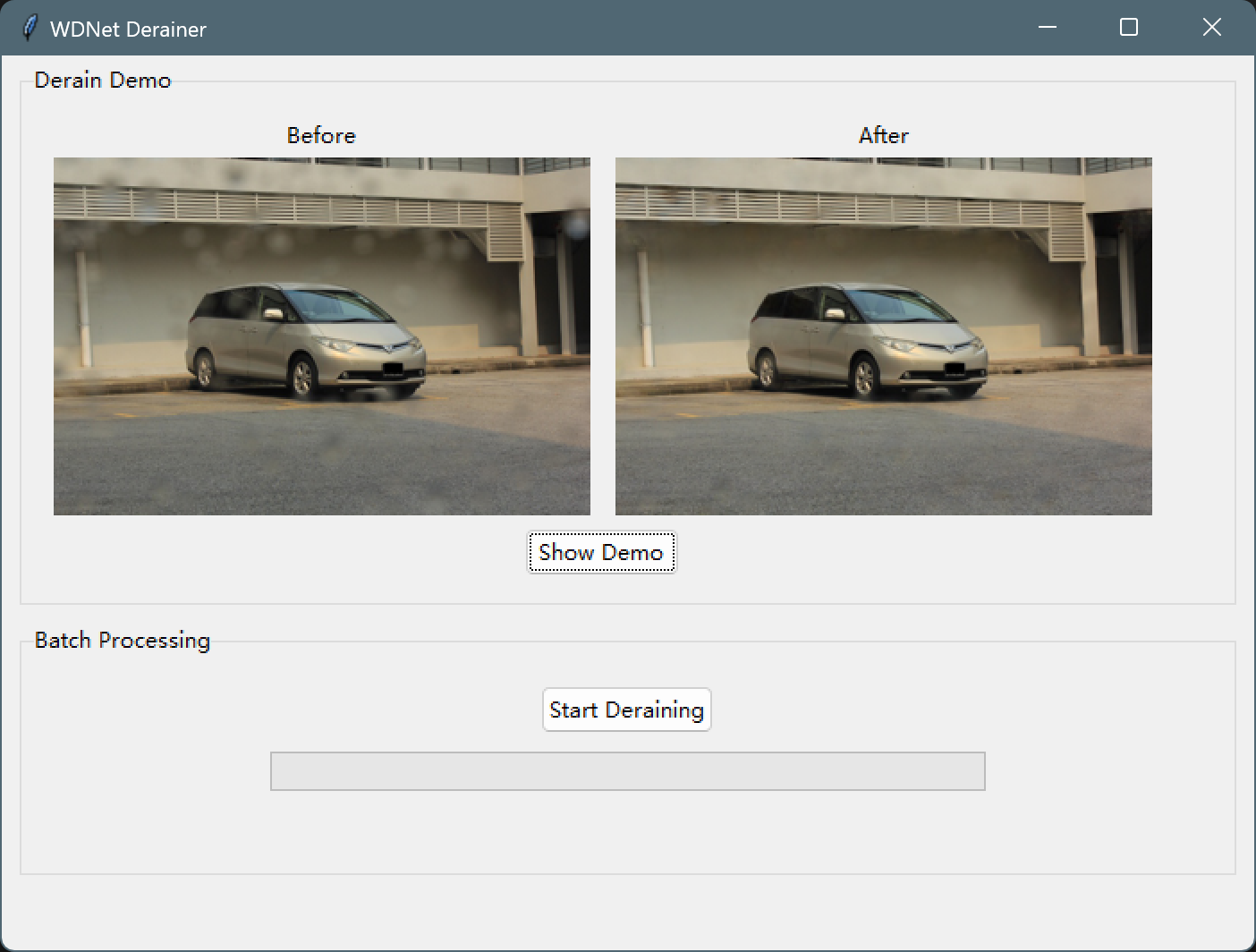
****

图 5 图像去雨滴测试功能

**4.3 图像批量去雨滴功能**

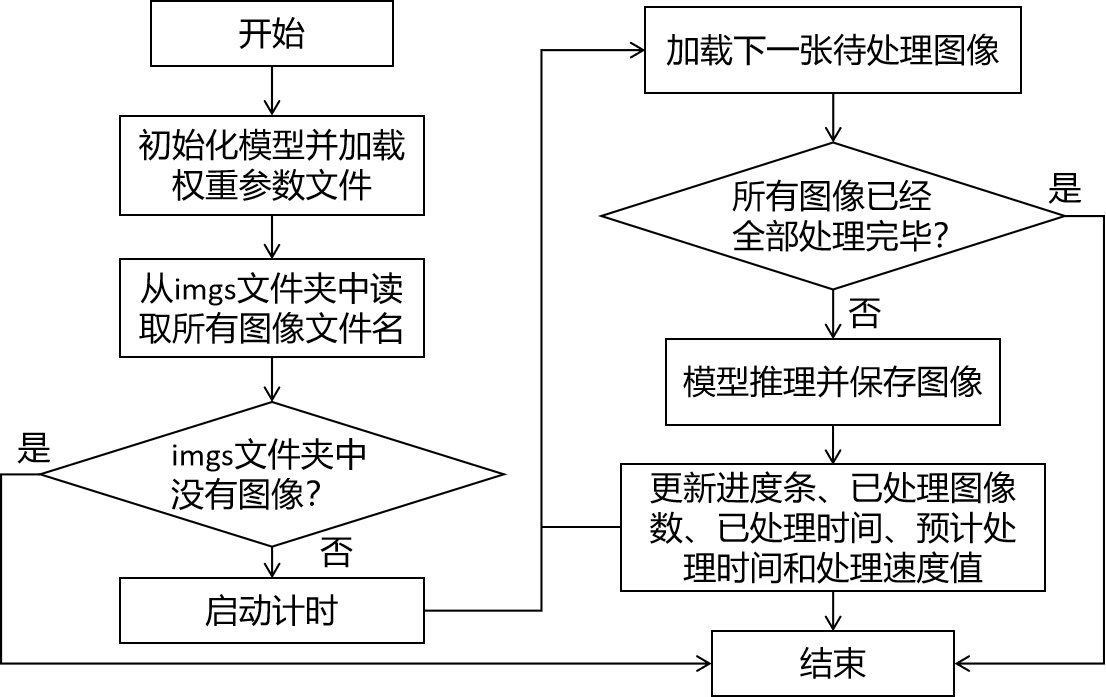


图 6 图像批量去雨滴功能

用户在进行去雨滴测试并观察结果且对测试结果满意，则可以进行imgs文件夹下所有图像文件的批量雨滴去除。要使用此功能，用户在确认imgs文件夹中存放了所有需要进行雨滴去除的图像后，即可点击“Start Deraining”按钮，开始去雨。

去雨过程中，进度条下方会显示Processed，Elapsed，Remaining和Speed四个数值。其中：

1. Processed提示用户当前已经处理的图像数量和需要处理的图像总数。例如图4中，Processed: 5/11表示当前已经处理了5张图像，一共有11张图像。
2. Elapsed提示用户当前已经处理耗费的时间。例如图4中，Elapsed: 1.5s表示程序当前已经处理了1.5秒。
3. Remaining提示用户根据当前的平均处理速度计算出的预计剩余处理时间。例如图4中，Remaining: 1.8s表示预计还剩1.8秒图像将全部处理完毕。
4. Speed提示用户程序处理图像的平均速度，单位为张每秒。例如图4中，Speed: 3.30 img/s表示程序处理图像的平均速度为每秒处理3.3张。

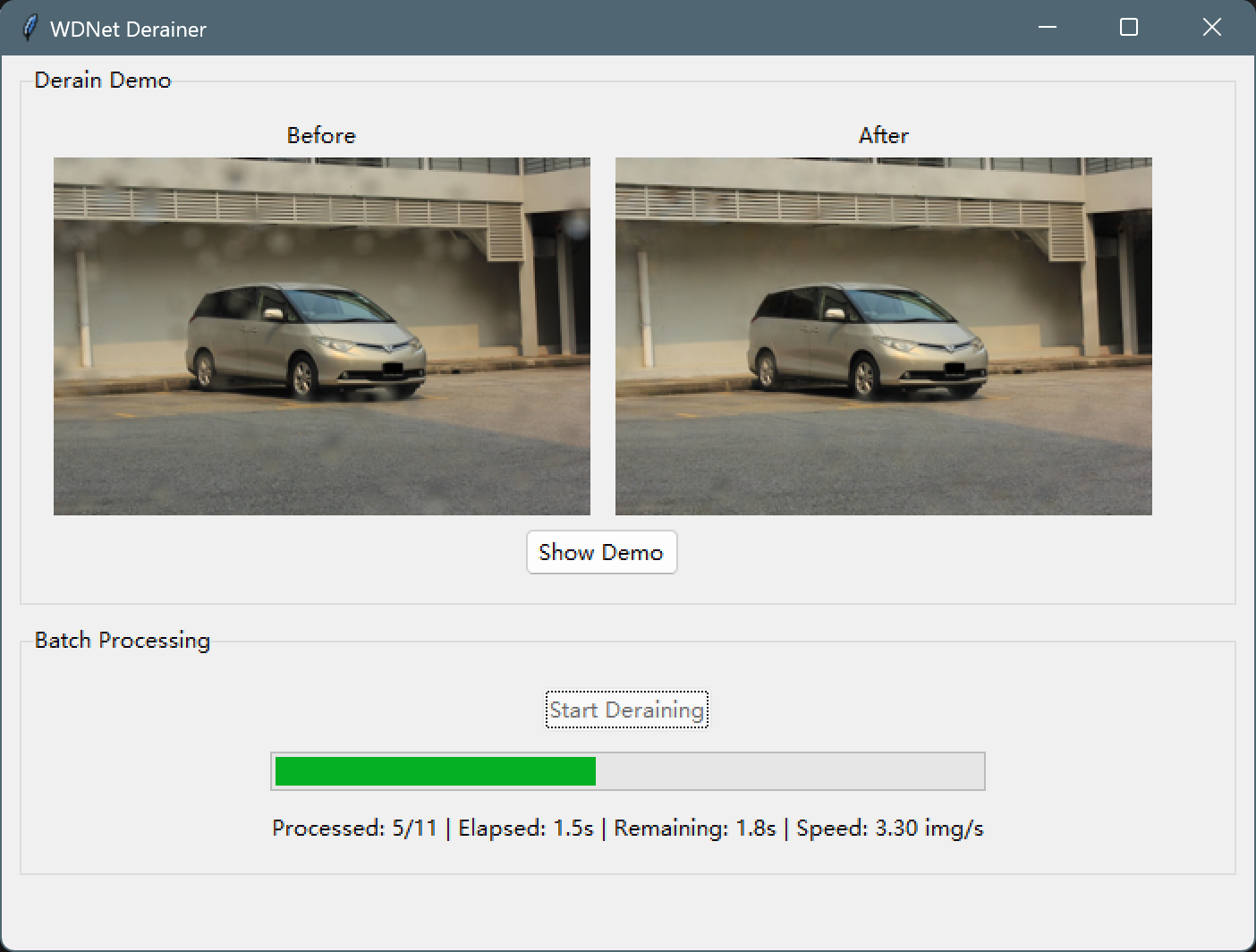


图 7 程序去雨过程

在所有图像的去雨滴处理都完毕后，程序将在进度条下显示”Processing complete”字样，提示用户所有图像均已处理完毕。

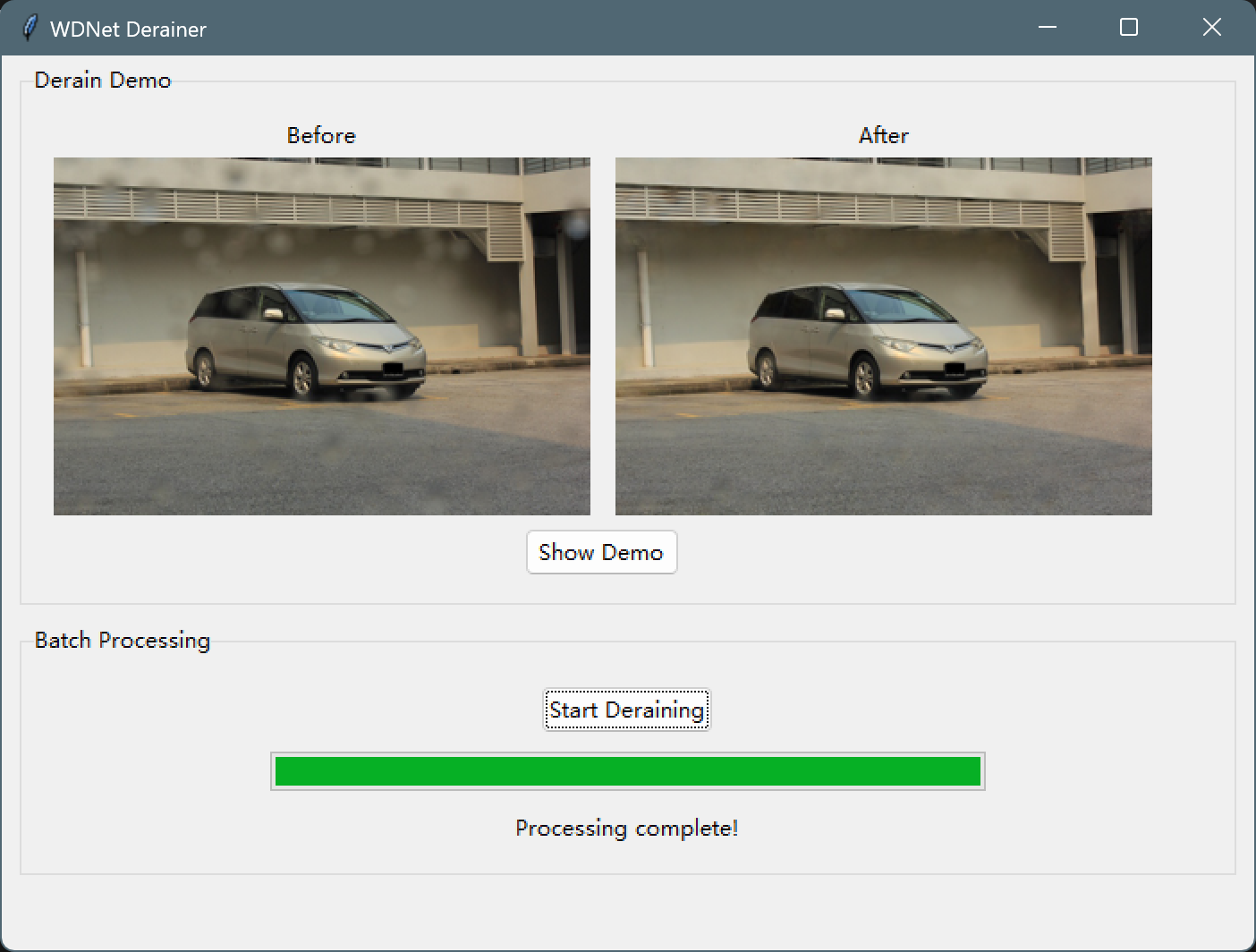


图 8 去雨完毕

**4.4 模型训练功能**

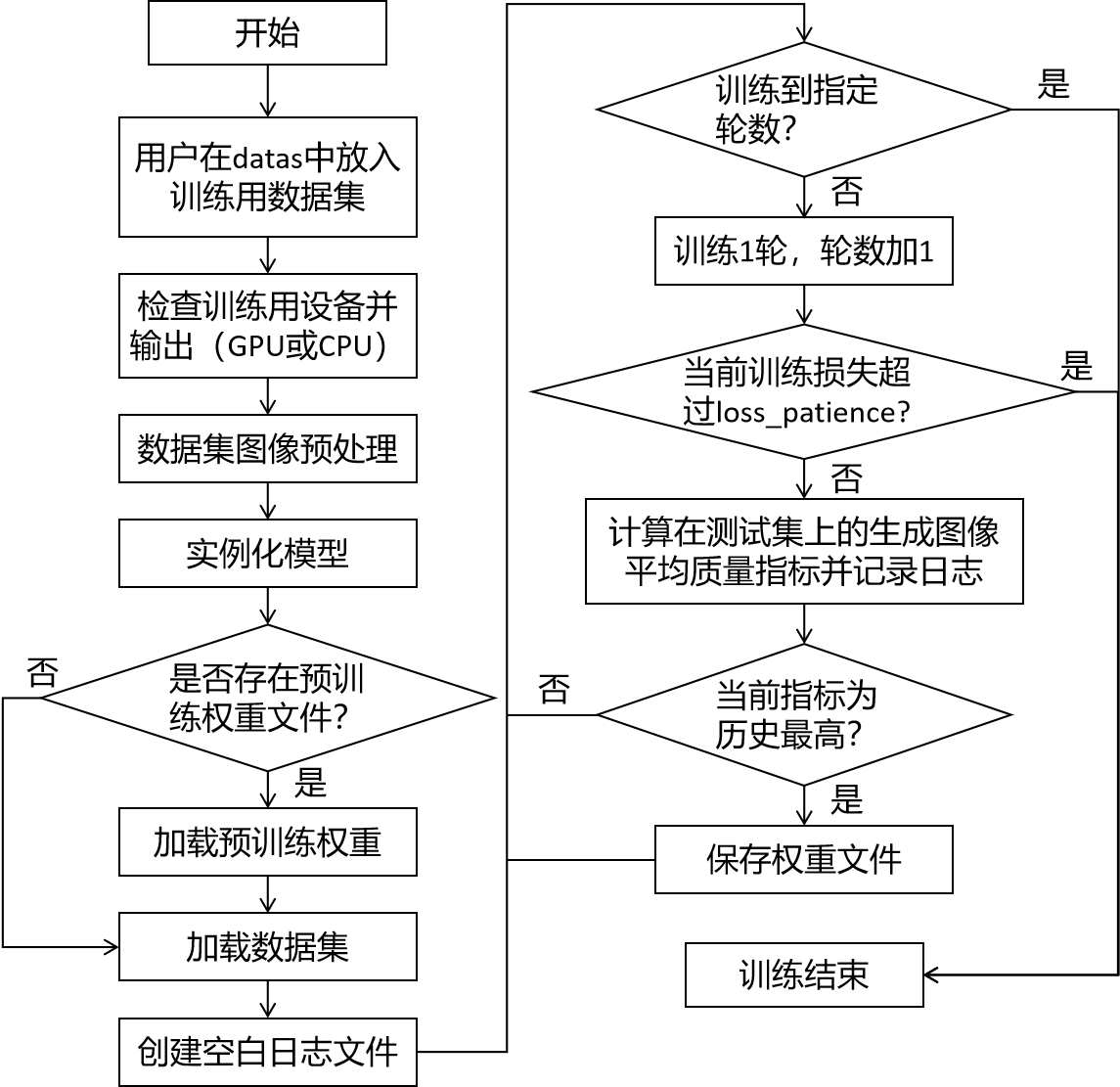


图 9 模型训练过程流程图

若用户需要训练自己的图像去雨滴模型参数，则可以使用模型训练功能。系统的模型训练功能流程图如图8所示。

首先，用户需要先将图像数据集数据集分为四个部分，分别是训练集有雨滴图像部分，训练集无雨滴图像部分，测试集有雨滴图像部分和测试集无雨滴图像部分。然后，按照表1所示将图像存入相应的文件夹中。

表 1 训练用数据集图像集合及相应存放位置

|  |  |
| --- | --- |
| 图像集合 | 应当存入的文件夹 |
| 训练集有雨滴图像 | WD-Derainer/datas/archive/data/data\_train |
| 测试集有雨滴图像 | WD-Derainer/datas/archive/data/data\_test |
| 训练集无雨滴图像 | WD-Derainer/datas/archive/gt/gt\_train |
| 测试集无雨滴图像 | WD-Derainer/datas/archive/gt/gt\_test |

接下来，用户可通过命令行参数自定义训练过程，参数如表2所示。

表 2 训练超参数定义

| 参数名 | 类型 | 默认值 | 描述 |
| --- | --- | --- | --- |
| --data\_dir | str | datas/archive | 数据集路径 |
| --image\_size | str | 480,720 | 输入图像尺寸（高度,宽度） |
| --batch\_size | int | 1 | 训练批次大小 |
| --num\_epochs | int | 1000 | 训练周期数 |
| --lr | float | 1e-3 | 初始学习率 |
| --weight\_decay | float | 1e-5 | 优化器权重衰减 |
| --model\_path | str | wdnet.pth | 模型保存路径 |
| --alpha | float | 0.5 | L1 损失与感知损失的权重 |
| --vgg\_model | str | vgg16 | 感知损失使用的 VGG 模型 |
| --max\_loss | float | 10.0 | 损失爆炸检测阈值 |
| --loss\_patience | int | 50 | 损失异常监控的步数 |
| --log\_file | str | 自动生成 | 日志文件路径 |
| --device | str | cuda | 训练设备（cuda 或 cpu） |

要想启动训练，进入WD-Derainer目录，在命令行输入python train.py即可按照默认参数启动训练程序。如果用户先前有训练好的模型权重文件（.pth格式）位于参数“--model\_path”的指定位置，模型则会自动加载该参数文件并继续训练；如果程序未在该位置找到权重文件，模型则将自动从头开始训练。

**5、技术支持**

如果在系统使用过程中出现任何问题，可将所出现问题由邮件形式进行描述，发送至邮箱，我们将尽力为您解决所有问题。如有任何建议，我们也非常愿意共同进行探讨。