

LogoCleaner User Manual

Introduction

LogoCleaner is an AI-powered tool that automatically detects and removes logos from images. Using advanced machine learning models, it identifies logos in your photos, generates precise masks around them, and intelligently fills in the background to create clean, logo-free images.



System Requirements

- **Operating System:** Windows, macOS
- **CPU:** Any modern multi-core processor (no GPU required)
- **RAM:** Minimum 8GB, recommend 16GB
- **Disk Space:** Minimum 2GB for application and models
- **VPN:** Needs VPN outside Hong Kong since the software uses OpenAI API do background filling

Installation

Setting up Conda Environment

Conda is recommended for managing the Python environment. Follow these steps to set up:

1. **Install Miniconda or Anaconda** (if not already installed):
 - Download from <https://docs.conda.io/en/latest/miniconda.html>

- Follow the installation instructions for your platform

2. **Create a new conda environment:**

```
1  conda create -n logocleaner python=3.10
```

3. **Activate the environment:**

```
1  conda activate logocleaner
```

4. **Install PyTorch (CPU version):**

```
1  pip3 install torch torchvision torchaudio
```

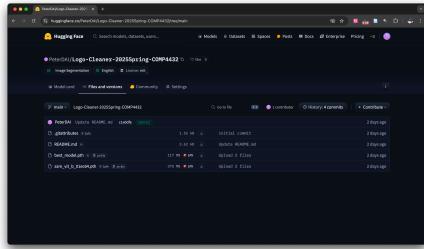
Downloading Model Files

The LogoCleaner application requires two model files:

1. **Logo Detection Model:** `best_model.pth`
2. **Segment Anything Model:** `sam_vit_b_01ec64.pth`

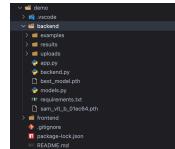
Follow these steps to download them:

1. **Visit the Hugging Face repository:**
 - Go to <https://huggingface.co/PeterDAI/LogoCleaner/tree/main>
2. **Download the model files:**
 - Click on `best_model.pth` and download it
 - Click on `sam_vit_b_01ec64.pth` and download it



3. Place the model files in the backend folder:

- o Move both downloaded files to the `LogoCleaner_github/demo/backend/` directory



Backend Setup

1. Navigate to the backend directory:

```
1 cd /path/to/LogoCleaner_github/demo/backend
```

2. Install required Python packages:

```
1 pip install -r requirements.txt
```

This will install all necessary packages including Flask, Pillow, PyTorch, and other dependencies.

Frontend Setup

1. Install Node.js (if not already installed):

- o Download from <https://nodejs.org/> (version 14+ recommended)

- o Follow the installation instructions for your platform

2. Navigate to the frontend directory:

```
1 cd /path/to/LogoCleaner_github/demo/frontend
```

3. Install Node.js dependencies:

```
1 npm install
```

This will install all necessary packages including Vue.js, Axios, and other frontend dependencies.

Running the Application

The application consists of two parts that need to run simultaneously:

Starting the Backend Server

1. Open a terminal and navigate to the backend directory:

```
1 cd /path/to/LogoCleaner_github/demo/backend
```

2. Start the Flask server:

```
1 python app.py
```

3. Verify the server is running:

- o You should see output indicating the server is running on <http://localhost:5050>



Starting the Frontend Application

1. Open a new terminal window and navigate to the frontend directory:

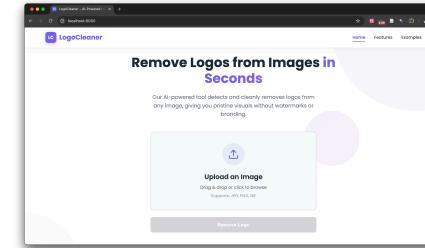
```
1 cd /path/to/LogoCleaner_github/demo/frontend
```

2. Start the development server:

```
1 npm run dev
```

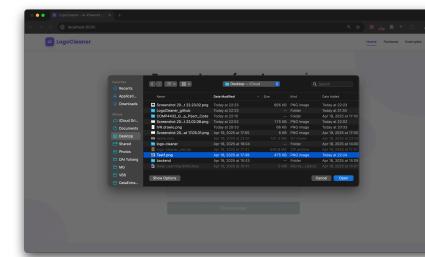
3. Verify the frontend is running and Access the application:

- The console will provide the URL where the application is accessible (<http://localhost:8080>)
- Open your web browser and navigate to the URL provided in the console (<http://localhost:8080>)



2. Select an image:

- Click on the upload area or drag and drop an image file and click the "Remove Logo" button
- Supported formats: JPG, PNG, GIF



Using LogoCleaner

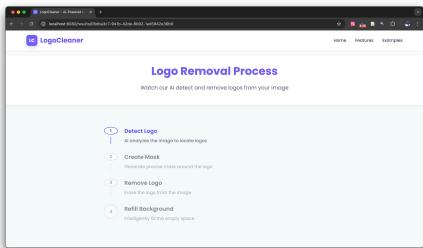
Uploading Images

1. Access the home page:

- Navigate to <http://localhost:8080> in your web browser
- You'll see the LogoCleaner interface with an upload area

Logo Detection Process

The logo removal process happens in several steps, which you can observe in real-time:



1. Upload:

- Your image is uploaded to the server
- A processing job is created

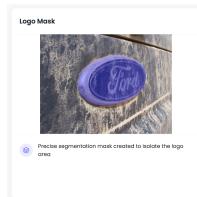
2. Logo Detection:

- The AI model analyzes the image to locate logos
- A bounding box is created around the detected logo



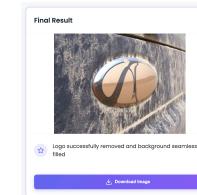
3. Mask Refinement:

- The Segment Anything Model (SAM) creates a precise mask of the logo
- This ensures accurate boundary detection



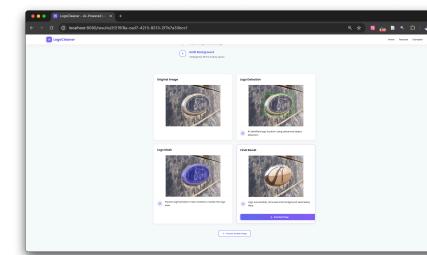
4. Background Filling:

- The logo area is intelligently filled with appropriate background content
- This creates a natural-looking result



Viewing Results

After processing completes, you'll be automatically redirected to the results page:



Downloading Processed Images

1. Download your logo-free image:

- Click the "Download" button on the results page
- The processed image will be saved to your computer

2. Process another image:

- Click "Process Another Image" to return to the upload page
- You can then select a new image to process

Troubleshooting

1. **Backend fails to start:**
 - o **Issue:** Error about missing model files
 - o **Solution:** Ensure both model files (`best_model.pth` and `sam_vit_b_01ec64.pth`) are in the backend directory
2. **"Connection refused" error in frontend:**
 - o **Issue:** The backend server is not running or is on a different port
 - o **Solution:** Ensure the backend server is running on port 5050