# Complete FLASH Installation Protocol

## Muscle Fiber Analyzer - Complete Installation

### System Requirements

**Minimum configuration required:**

* **Operating System:** Windows 10/11 (x64), macOS 10.15+, or Linux Ubuntu 18.04+
* **RAM:** 8 GB minimum (16 GB recommended)
* **Disk Space:** 5 GB free space
* **Graphics Card:** NVIDIA (optional, for GPU acceleration)
* **Administrator Rights:** Required only for GPU/CUDA installation

## STEP 0: GPU Verification (Windows only - optional)

### A. Graphics Card Identification

1. **Open Task Manager:**
   * Press Ctrl + Shift + Esc
   * Click on the "Performance" tab
   * Select "GPU" in the left panel
2. **Check your graphics card:**
   * **Required:** NVIDIA GTX 10xx, RTX series, or newer
   * **Recommended GPU RAM:** 6 GB or more
   * Note the exact card name (e.g., "NVIDIA GeForce RTX 3070")

**Compatibility check:**

* ✅ **Compatible:** GTX 1060, RTX 2060, RTX 3070, RTX 4080, etc.
* ❌ **Not compatible:** GTX 900 series or older, AMD cards

### B. CUDA Toolkit 11.8 Installation (Administrator rights required)

1. **Download CUDA Toolkit 11.8:**
   * Go to: https://developer.nvidia.com/cuda-11-8-0-download-archive
   * Select: Windows → x86\_64 → 10/11 → exe (local)
2. **CUDA Installation:**
   * Close all applications
   * Run the downloaded file as administrator
   * Choose "Custom installation" if offered
   * Restart computer after installation
3. **CUDA Verification:**
   * Open command prompt (cmd)
   * Type: nvcc --version
   * Verify that "Cuda compilation tools, release 11.8" is displayed

## STEP 1: FIJI/ImageJ Installation

### A. Download and installation

**If you already have FIJI, skip to step B.**

1. **Download FIJI:**
   * Website: https://imagej.net/software/fiji/downloads
   * Choose the "Stable download" for your OS
2. **Installation:**
   * Extract the archive to a folder (e.g., C:\Fiji or /Applications/Fiji.app)
   * Launch FIJI to verify installation

### B. Initial configuration

1. In FIJI: Help → Update...
2. Click "Manage update sites"
3. **Activate the following sites** (check boxes):
   * ✅ IBMP-CNRS
   * ✅ ImageScience
   * ✅ PTBIOP
4. Click Close → Apply changes
5. **Restart FIJI** after installation

## STEP 2: Python and Cellpose Installation

### A. Miniconda Installation (recommended)

1. **Download Miniconda:**
   * Website: https://repo.anaconda.com/miniconda/
   * **Windows:** Miniconda3-latest-Windows-x86\_64.exe
   * **macOS:** Miniconda3-latest-MacOSX-x86\_64.pkg
   * **Linux:** Miniconda3-latest-Linux-x86\_64.sh
2. **Installation:**
   * **Windows:** Double-click executable, check "Add to PATH"
   * **macOS:** Double-click the .pkg
   * **Linux:** bash Miniconda3-latest-Linux-x86\_64.sh
3. **Restart** your terminal/command prompt

### B. Conda Configuration and Cellpose Environment Creation

**Open a terminal/command prompt and execute in order:**

# Initial Conda configuration

conda config --set auto\_activate\_base false

conda config --set channel\_priority flexible

conda config --set solver libmamba

# Accept Conda terms

conda info

# Clean existing environments

conda env remove -n cellpose -y

conda clean --all -y

# Create cellpose environment with Python 3.8

conda create -n cellpose python=3.8 -y

# Activate environment

conda activate cellpose

**Installation according to your configuration:**

**For GPU (Windows with NVIDIA):**

pip install cellpose==3.1.1.2

pip install torch==1.13.1+cu118 torchvision==0.14.1+cu118 torchaudio==0.13.1 --index-url https://download.pytorch.org/whl/cu118 --no-cache-dir

**For CPU only (macOS/Linux or Windows without GPU):**

pip install cellpose==3.1.1.2

pip install torch==1.13.1+cpu torchvision torchaudio --index-url https://download.pytorch.org/whl/cpu --no-cache-dir

**Common dependencies:**

pip install numpy==1.24.3 opencv-python-headless scikit-image imageio matplotlib scipy numba

### C. Installation Verification

# Basic test

python -c "import cellpose; print('Cellpose successfully installed!')"

# GPU test (Windows with NVIDIA only)

python -c "import torch; print('CUDA available:', torch.cuda.is\_available()); print('GPU detected:', torch.cuda.get\_device\_name(0) if torch.cuda.is\_available() else 'None')"

# Cellpose GPU test

python -c "from cellpose import models; model = models.Cellpose(gpu=True); print('Cellpose GPU: OK' if model.gpu else 'GPU not available')"

## STEP 3: Cellpose Plugin Configuration in FIJI

### Plugin configuration

1. In FIJI: open any image
2. Go to Plugins → BIOP → Cellpose/Omnipose → Cellpose...
3. **Check in "to add more parameters:"**
   * ✅ That --use\_gpu is written (if not, add it)
   * ❌ That --do\_3d is NOT present (if yes, delete it)

## STEP 4: FLASH Macro Installation

### A. Macro download

1. Save the macro code in a file named FLASH.ijm
2. **Place the file in FIJI's plugins folder:**
   * **Windows:** C:\Fiji\plugins\
   * **macOS:** /Applications/Fiji.app/plugins/
   * **Linux:** /path/to/fiji/plugins/

### B. Installation in FIJI

1. **Restart FIJI**
2. The macro appears in Plugins → FLASH
3. Alternative: Plugins → Macros → Run... → select FLASH.ijm

## STEP 5: Image Preparation

### A. Required format

* **Format:** .tif or .tiff only
* **Type:** Multi-channel images (3-5 channels recommended)
* **Resolution:** Minimum 1024×1024 pixels
* **Depth:** 8-bit or 16-bit

### B. Channel organization

**Standard organization example:**

1. **Channel 1:** Type IIb (MYH4) - optional
2. **Channel 2:** Laminin (membranes) - **required for segmentation**
3. **Channel 3:** Type I (MYH7) - optional
4. **Channel 4:** SDH (Oxidative) - optional
5. **Channel 5:** Type IIa (MYH2) - optional

### C. Folder preparation

1. Create a folder for analysis
2. Copy all images to this folder
3. Verify all files are in .tif format
4. **Ensure channel organization is identical for all images**

## STEP 6: Using FLASH

### A. Analysis launch

1. Open FIJI
2. Launch macro: Plugins → FLASH
3. **Fill parameters:**
   * **Total number of channels:** 4 (example)
   * **Laminin channel:** 2 (example)
   * **Fiber type channels** according to your staining
   * ✅ Check "Auto-calibrate diameter" (recommended)

### B. Advanced parameters (optional)

* **Cellpose sensitivity:** 1.0 (default)
* **Exclusion threshold:** 200 pixels
* **Adaptive threshold factor:** 0.3
* **GPU:** Check if NVIDIA card available and CUDA installed

### C. Folder selection and processing

1. Select folder containing your images
2. Wait for processing completion (progress bar)
3. Check results in source folder

## STEP 7: Results Interpretation

### A. Automatically generated files (per image)

* **ImageName\_Final.tif** - Image with colored overlay
* **ImageName\_Classified\_Results.csv** - Classification results
* **ImageName\_ROI\_Set.zip** - Regions of interest
* **ImageName\_cellposeMask.tif** - Segmentation mask

### B. Results analysis

* Open CSV file in Excel/LibreOffice
* **Important columns:**
  + **ROI:** Fiber number
  + **CSA/MinFeret:** Cross-sectional area/MinFeret
  + **Classification:** Identified fiber type
  + **Fiber type statistics** at bottom of file

## Troubleshooting

### Common problems and solutions

**CUDA Error:**

* Check graphics card compatibility
* Reinstall CUDA Toolkit 11.8
* Restart computer

**Cellpose not working:**

* Reinstall cellpose if necessary

**Images not detected:**

* Check .tif/.tiff format
* Verify channel organization
* Ensure laminin channel is correct

**Memory issues:**

* Close unnecessary applications
* Use smaller image
* Consider increasing virtual memory

## Additional Notes

### Performance optimization tips

* Use GPU acceleration when available for faster processing
* Process images in smaller batches if memory is limited

### Best practices

* Always backup original images before processing
* Document channel organization for each experiment
* Validate results on a subset of images before batch processing
* Keep detailed records of analysis parameters used

**Protocol tested on:** Windows 10/11, macOS 12+, Ubuntu 20.04+  
**Version:** FLASH v3.1 – July 2025