

EMito-Metrix is a high-performance pipeline for analyzing mitochondrial morphology and ultrastructure in multiple tissues from low to high-resolution images acquired with Electron microscopy (EM). The application allows to compute a set of morphometrics and texture measurements, and provides a list of graphs for optimizing data visualization using dimensionality reduction (UMAP or PCA) and more conventional depiction of data distribution (density curves, histograms, violin plots or star plots). Additionally, a machine learning (ML) module with predictive analytic tools allows to determine how any given experimental condition would impact on mitochondrial morphology and ultrastructure

EMito-Metrix application was written by Mathieu Vigneau, Emmanuel Doumard and Jean-Philippe Pradère from the RESTORE Institute.

This section describes installation instructions for Windows, Linux and MacOS operating systems

## - SYSTEM REQUIREMENTS




---

Windows, Linux and MacOS are supported for running the EMito\_Metrix application. Before installing, we recommend to check the following requirements for each software and environment used by the application:

- Anaconda distribution requirements:  
<https://docs.anaconda.com/anaconda/install/>
- Fiji software requirements:  
<https://imagej.net/software/fiji/downloads#system-requirements>
- Cellpose software requirements:  
<https://github.com/MouseLand/cellpose?tab=readme-ov-file#system-requirements>

## - TABLE OF CONTENTS

---

• Emito-Metrix installation for Windows.....	
• Emito-Metrix installation for MacOS.....	
• Emito-Metrix installation for Linux.....	

## EMITO-METRIX INSTALLATION FOR WINDOWS

### 1- CELLOSE PYTHON ENVIRONMENT: INSTALLATION AND SETTING

#### - Anaconda distribution installation

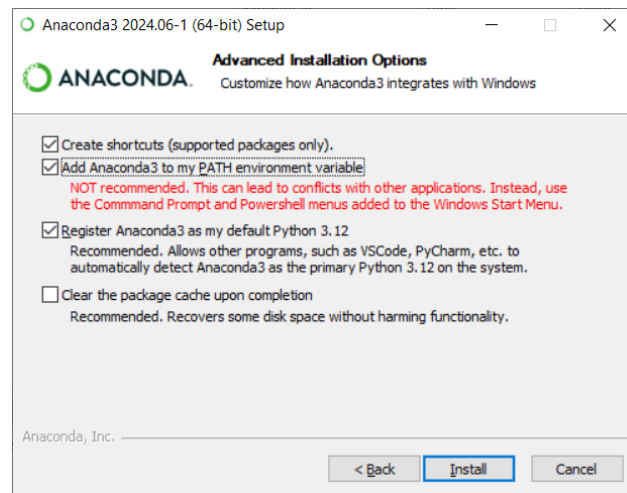
Install an Anaconda distribution for Python 3 using the following URL:

<https://www.anaconda.com/download/success>

Download the latest version of Anaconda installer for Windows 64bits. After downloading, go to your download folder and double-click the installer to launch Anaconda installation.

It is recommended that you install Anaconda using the *Just Me* option, which will install Anaconda Distribution to just the current user account (doesn't require Windows Administrator privileges).

During installation, add Anaconda to your PATH environment variable, as shown below:



See [here](#) for detailed instructions of Anaconda installation and settings

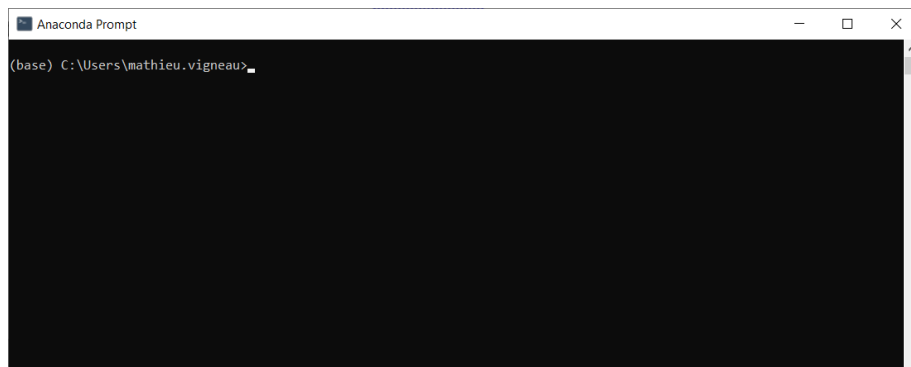
## - Cellpose installation & setting

---

Once you installed anaconda environment, you will need to create and set a python virtual environment for Cellpose application (which is essential for the EMito-Metrix application).

- **Open an anaconda prompt**

From the Windows start menu (🔍), type *anaconda prompt* in the search bar and click on the Anaconda application to start it. This will open a new command prompt window with Anaconda enabled, as shown below:



- **Create the Cellpose Python environment**

We are going to create a **Python virtual environment (venv) for Cellpose** application.

*A virtual environment is simply a Python subdirectory with its own file structure, python packages and settings. This Virtual environment let you have a stable and reproducible Cellpose environment, which is essential for the proper running of the EMito-Metrix application.*

To create the Cellpose venv, type the following python command line in your opened anaconda prompt, and press Enter:

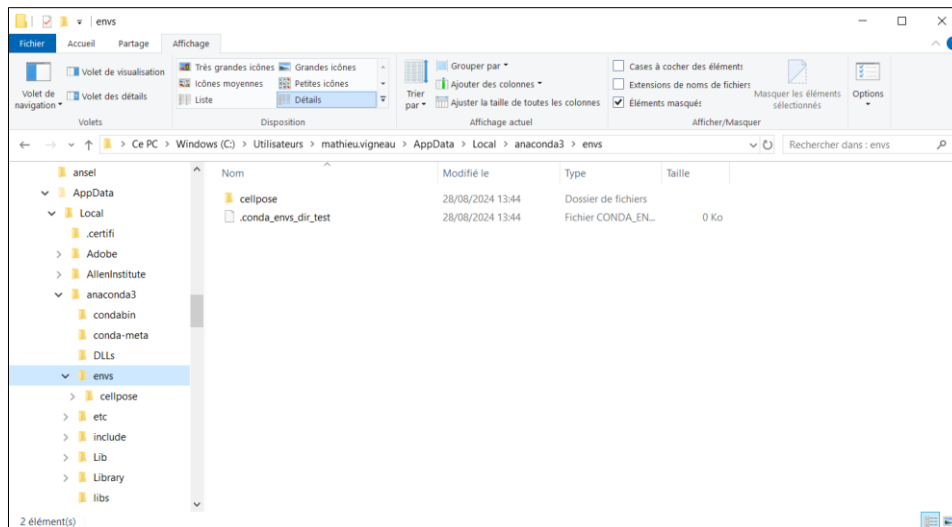
```
conda create --name cellpose python=3.9
```

This command line **will create a new python3.9 environment named *cellpose***.

This will create a new folder, containing your Cellpose python environment. This cellpose folder is found under the following Windows path:

*C:\Users\username\AppData\Local\anaconda3\envs\cellpose* or

*C:\Users\username\AppData\anaconda\envs\cellpose*



- **Cellpose environment activation**

To activate your Cellpose python environment, type the following python command line in your opened anaconda prompt, and press Enter:

```
conda activate cellpose
```

```

Anaconda Prompt - "C:\ProgramData\anaconda3\condabin\conda.bat" activate cellpose
vc                pkgs/main/win-64::vc-14.2-h2eaa2aa_1
vs2015_runtime    pkgs/main/win-64::vs2015_runtime-14.29.30133-h43f2093_3
wheel             pkgs/main/win-64::wheel-0.43.0-py311haa95532_0
xz               pkgs/main/win-64::xz-5.4.6-h8cc25b3_1
zlib             pkgs/main/win-64::zlib-1.2.13-h8cc25b3_1

Proceed ([y]/n)? y

Downloading and Extracting Packages
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#   $ conda activate cellpose
#
# To deactivate an active environment, use
#
#   $ conda deactivate

(base) C:\Users\mathieu.vigneau>conda create --name cellpose python=3.11
(base) C:\Users\mathieu.vigneau>conda activate cellpose
(cellpose) C:\Users\mathieu.vigneau>

```

- **Cellpose package installation**

Next, you need to install Cellpose application from your Cellpose python venv. In your opened anaconda prompt, type the following python command line, and press Enter:

```
python -m pip install cellpose[gui]
```

This command line will install Cellpose python packages in your python virtual environment, also including the **Cellpose Graphical user Interface**.

- **(Optional) Cellpose GPU package installation**

If you have a **CUDA supported GPU** installed on your computer (see [here](#) for CUDA GPU compatibility), you can install a GPU version of Cellpose.

Please see this [URL](#) for the install instructions of CUDA toolkit and GPU version of Cellpose.

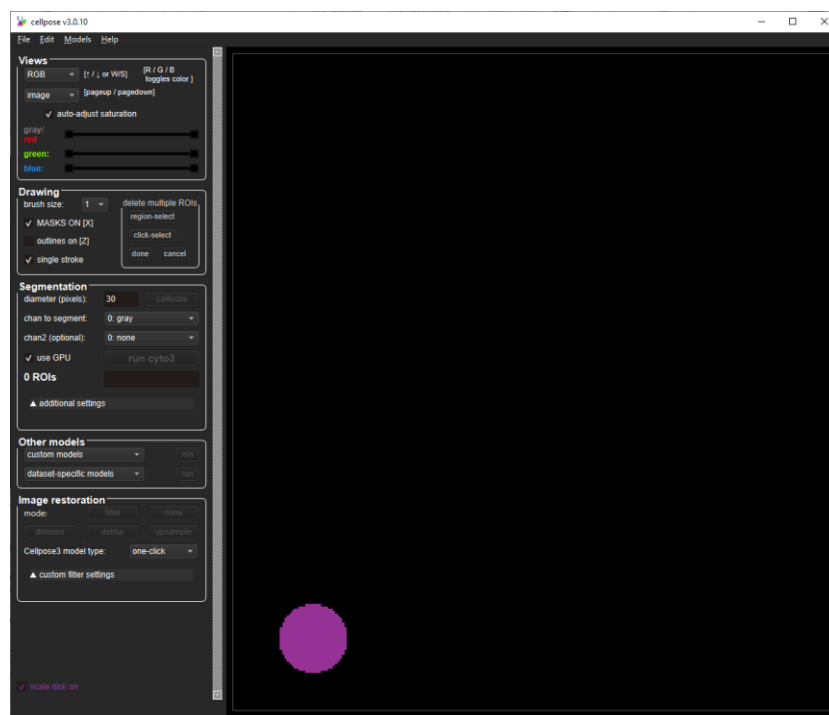
If you need more information about Cellpose detailed requirements, dependencies and installation instructions, see [here](#).

- **(Optional) Cellpose package checking**

Once you installed the Cellpose package, we recommend verifying the validity of the installation. From your Cellpose python environment, type the following python command lines and press Enter:

```
cellpose
```

If the package installation is correct, the following window will appear:



If you have issues with installation, see the cellpose [docs](#) for more details.

- **Cellpose environment deactivation**

Once the Cellpose installation is correct, **deactivate your Cellpose python environment** typing in your opened anaconda prompt the following python command line:

```
conda deactivate
```

## 2- FIJI INSTALLATION AND SETTING

---

### - **Fiji software installation**

---

Install Fiji software distribution using this following URL:

<https://imagej.net/software/fiji/downloads>

Download the latest version of Fiji for Windows 64-bit or Windows-32-bits using the *imagej.net* (USA) link. After downloading, go to your download folder and double-click the *Fiji-zip* file to uncompress it.

**It is highly recommended that you store your Fiji app directory in your user space (C:\Users\username\MyDocuments\FijiApp) rather than in the Program Files windows-system directory (write permission issues in these system-wide folders could prevent Fiji updating)**

See [here](#) for detailed installation instructions and system requirements of Fiji application

### - **FIJI plugin updating & setting**

---

EMito-Metrix plugin relies on the following Fiji plugins, which are not automatically installed during the Fiji software installation:

- Bio-Formats (<https://imagej.net/formats/bio-formats>)

*The Bio-Formats Importer is a plugin for reading data into Fiji. It can open many dozens of proprietary life sciences formats*

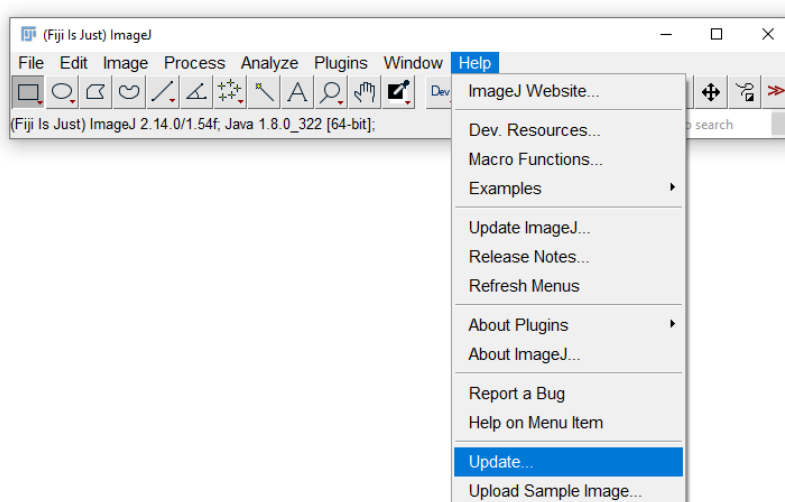
- ImageScience (<https://imagej.net/libs/imagescience>)

*ImageScience is a library of Java classes to facilitate scientific image computing.*

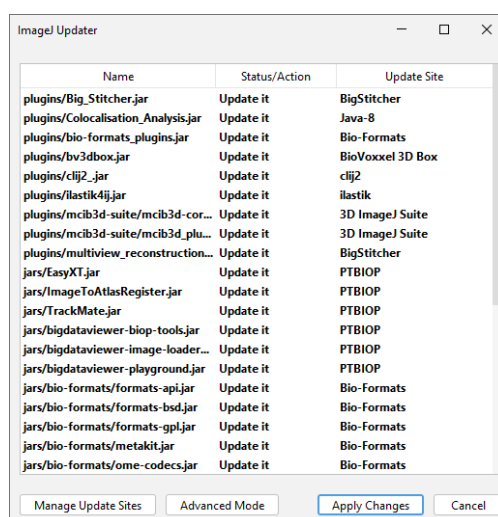
- PTBIOP (<https://wiki-biop.epfl.ch/en/ipa/fiji/update-site>)

*The EPFL - PTBIOP (BioImaging & Optics Platform) is a Fiji package including plugins, macros, ActionBars and Wrappers for external software calls (Cellpose, Elastix, Ilastix...)*

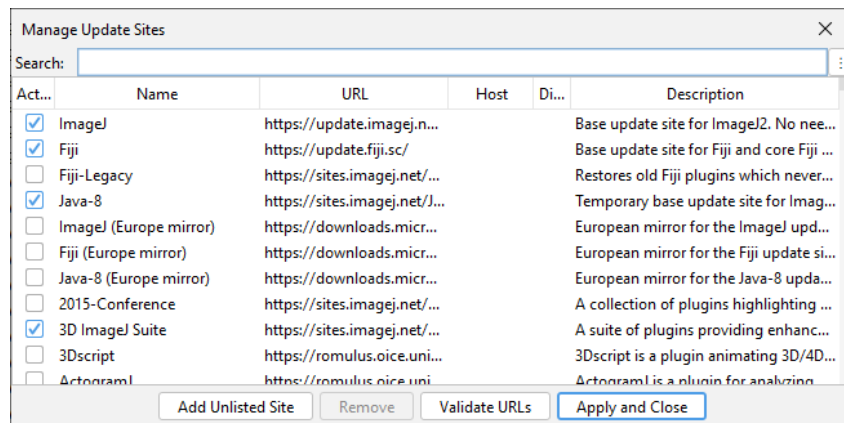
- In your Fiji App directory, double-click the *ImageJ-win64.exe* file to start Fiji software
- Once the software is opened, select *update* from the *Fiji Help menu* (as shown below) in order to update Fiji plugins



- In the *ImageJ Updater* window, select the *Manage update sites* button (see below).



- The *Manage Update Sites* window appears, which contains a list of Fiji plugins you can upload and install in your Fiji application (plugins already installed are checked).



From this *Manage Update Sites* window, check the following plugins to install it:

- Bio-Formats
- IBMP-CNRS
- ImageScience
- PTBIOP

- Once these plugins are checked, click on *Apply and Close* button from the *Manage Update Sites* window. Then, click on *Apply Changes* button from the *ImageJ Updater* window
- Once the plugin are installed, restart Fiji application

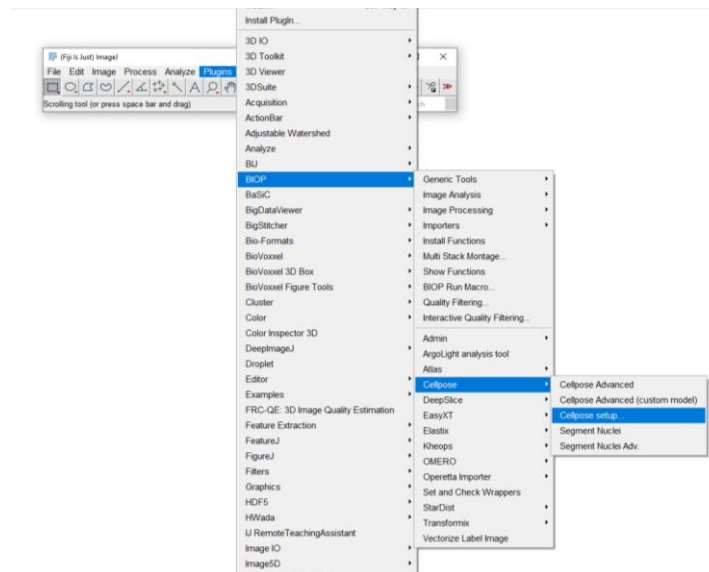
#### - **Cellpose wrapper settings**

---

Cellpose wrapper is a command from PTBIOP plugin that enables performing Cellpose object segmentation through Fiji application, using a python virtual environment with Cellpose package installed (see [here](#) to create Cellpose virtual environment).

- In your Fiji App, select *BIOP* from the *Fiji Plugins* menu. Then, select *Cellpose setup* in the *Cellpose* menu, as shown below :

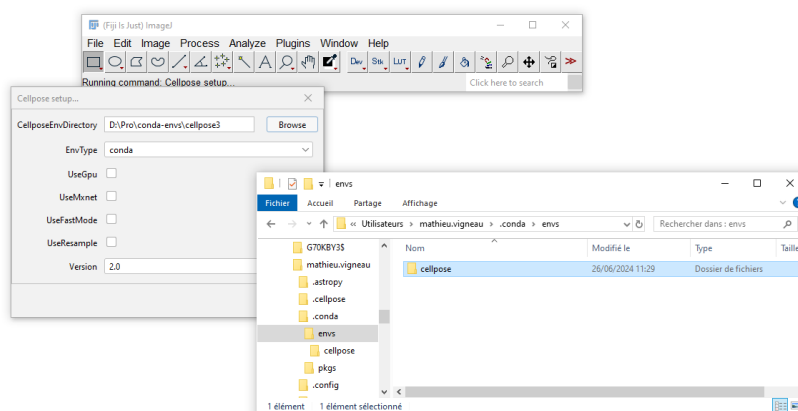




- The *Cellpose setup* window appears which allows setting Cellpose wrapper. From this window, modify the following settings:

- CellposeEnvDirectory: select your own Cellpose Python environment folder

*C:\Users\username\AppData\Local\anaconda3\envs\cellpose*      or      *C:\Users\username\AppData\anaconda\envs\cellpose*



- EnvType: python package chosen for your Cellpose environment

*Choose **conda** if you have installed Anaconda distribution for Python*

- UseGpu: allows using Cellpose with GPU computing (faster than CPU computing)

*Activate the checkbox if you have installed a Cellpose GPU version*

- Version: version of Cellpose launch from the Fiji wrapper

*Choose 2.0*

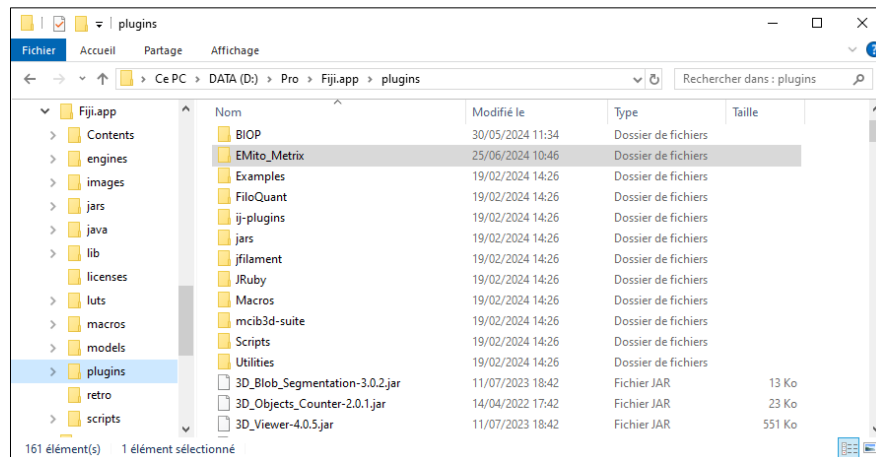
- Then press OK

## - **EMito-Metrix plugin installation & setting**

---

- **EMito Metrix folder creation**

In your Fiji plugins folder (*yourFijiFolder\plugins\*), create a new folder named **EMito\_Metrix**



- **EMito Metrix file upload**

**Download and save the EMito\_Metrix files** in your *EMito\_Metrix Fiji folder*, using the following GitHub repository:

<https://github.com/Restore-Lab/EMito-Metrix>

**You must upload these following files from the GitHub repository (!! Keep the same file names !!) :**

- EMito\_Metrix\_Fiji\_.ijm (Fiji plugin file)
- dataComputation.py (python file for data display)
- mlComputation.py (python file for data computation and prediction)
- requirements.txt (python package dependencies)
- GeneralistModel\_GM\_EM.zip (Cellpose GM model)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

- SpecialistModel\_SM\_FLY\_EM.zip (Cellpose SM model for Fly)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

- SpecialistModel\_SM\_HUMAN\_EM.zip (Cellpose SM model for Human)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

- SpecialistModel\_SM\_MOUSE\_EM.zip (Cellpose SM model for Mouse)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

- SpecialistModel\_SM\_ZEBRAFISH\_EM.zip (Cellpose SM model for Zebrafish)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

### 3- EMITOMETRIX PYTHON ENVIRONMENT: INSTALLATION & SETTING

---

- **Create the Emitometrix Python environment**

We are going to create a **Python virtual environment (venv)** for data computation plugins.

*A virtual environment is simply a Python subdirectory with its own file structure, python packages and settings. This Virtual environment let you have a stable and reproducible environment for data visualization and computation plugins, which are essential for the proper running of the EMito-Metrix application.*

To create the emitometrix venv, type the following python command line in your anaconda prompt, and press Enter:

```
conda create --name emitometrix python=3.11
```

This command line **will create a new python3.11 environment named *emitometrix***.

This will create a new folder containing your Emito-Metrix python environment. This folder is found under the following Windows path:

*C:\Users\username\AppData\Local\anaconda3\envs\emitometrix or*

*C:\Users\username\AppData\anaconda\envs\emitometrix*

- **Emitometrix environment activation**

To **activate your *emitometrix* python environment**, type the following python command line in your opened anaconda prompt, and press Enter:

```
conda activate emitometrix
```

- **Python dependencies installation**

EMito-Metrix plugin relies on the following Python packages, which are not automatically installed during emitometrix python environment installation:

- cloudpickle==3.0.0	- matplotlib==3.9.0	- pyparsing==3.1.2	- six==1.16.0
- contourpy==1.2.1	- numba==0.60.0	- python-dateutil==2.9.0.post0	- slicer==0.0.8
- cycler==0.12.1	- numpy==1.26.4	- pytz==2024.1	- threadpoolctl==3.5.0
- fonttools==4.53.0	- packaging==24.1	- scikit-learn==1.5.1	- tqdm==4.66.4
- joblib==1.4.2	- pandas==2.2.2	- scipy==1.14.0	- tzdata==2024.1
- kiwisolver==1.4.5	- pillow==10.4.0	- seaborn==0.13.2	- umap-learn==0.5.6
- llvmlite==0.43.0	- pynndescent==0.5.13	- shap==0.46.0	- xgboost==2.1.0

The list of these packages is included in the *requirements.txt* file that is found under your *EMito\_Metrix Fiji* folder

*example of EMitometrix folder : C:\Users\username\Documents\Fiji.app\plugins\EMito\_Metrix*

**To install python dependencies in your emitometrix environment**, type the following python command line from your opened anaconda prompt, and press Enter:

```
pip install -r yourFijiFolfer\plugins\EMito_Metrix\requirements.txt
```

- **Cellpose environment deactivation**

Once the Cellpose installation is correct, **deactivate your Emitometrix python environment** typing in your opened anaconda prompt the following python command line:

```
conda deactivate
```

**Once you correctly installed these software and environments, the EMito-Metrix application is ready for running. See the *EMitoMetrix tutorial* for running instructions.**

## EMITO-METRIX INSTALLATION FOR MACOS

### 1- CELLOSE PYTHON ENVIRONMENT: INSTALLATION AND SETTING

#### - Anaconda distribution installation

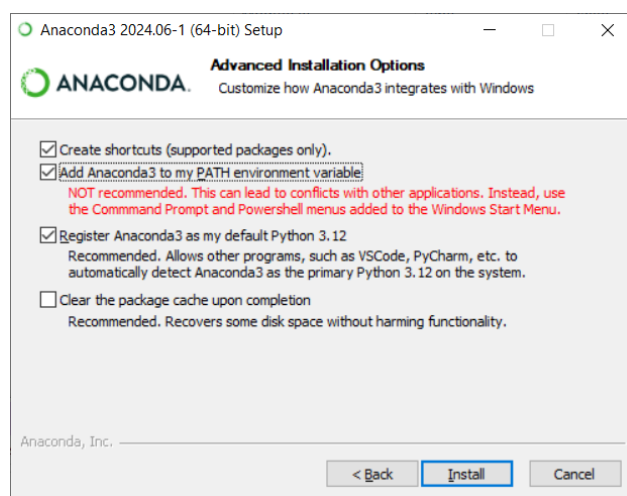
Install an Anaconda distribution for Python 3 using the following URL:

<https://www.anaconda.com/download/success>

Download the latest version of Anaconda installer for Windows 64bits. After downloading, go to your download folder and double-click the installer to launch Anaconda installation.

It is recommended that you install Anaconda using the *Just Me* option, which will install Anaconda Distribution to just the current user account (doesn't require Windows Administrator privileges).

During installation, add Anaconda to your PATH environment variable, as shown below:



See [here](#) for detailed instructions of Anaconda installation and settings

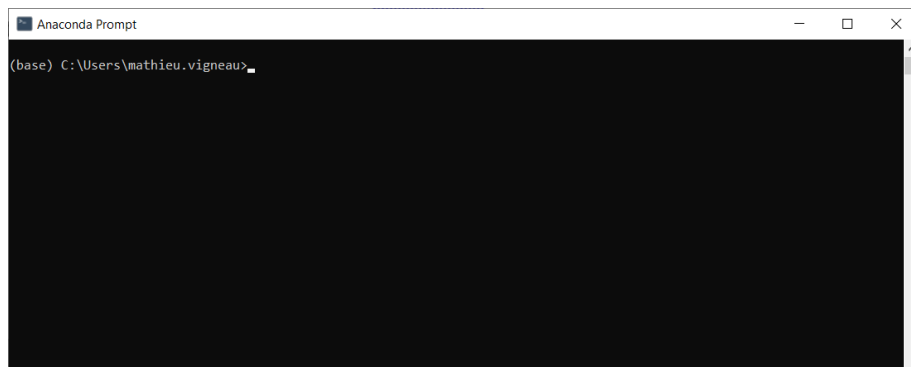
## - Cellpose installation & setting

---

Once you installed anaconda environment, you will need to create and set a python virtual environment for Cellpose application (which is essential for the EMito-Metrix application).

- **Open an anaconda prompt**

From the Windows start menu (🔍), type *anaconda prompt* in the search bar and click on the Anaconda application to start it. This will open a new command prompt window with Anaconda enabled, as shown below:



- **Create the Cellpose Python environment**

We are going to create a **Python virtual environment (venv)** for Cellpose application.

*A virtual environment is simply a Python subdirectory with its own file structure, python packages and settings. This Virtual environment let you have a stable and reproducible Cellpose environment, which is essential for the proper running of the EMito-Metrix application.*

To create the Cellpose venv, type the following python command line in your opened anaconda prompt, and press Enter:

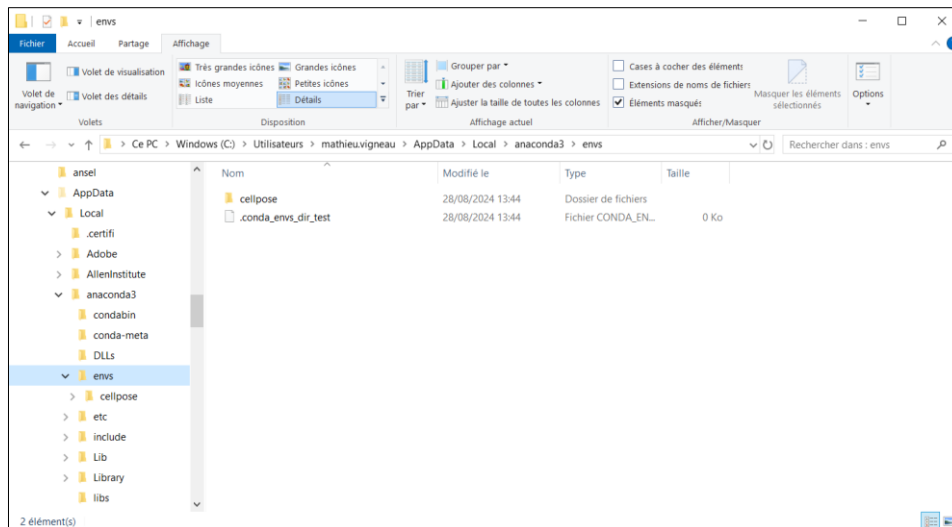
```
conda create --name cellpose python=3.9
```

This command line **will create a new python3.9 environment named *cellpose***.

This will create a new folder, containing your Cellpose python environment. This cellpose folder is found under the following Windows path:

*C:\Users\username\AppData\Local\anaconda3\envs\cellpose* or

*C:\Users\username\AppData\anaconda\envs\cellpose*



- **Cellpose environment activation**

To activate your Cellpose python environment, type the following python command line in your opened anaconda prompt, and press Enter:

```
conda activate cellpose
```

```

Anaconda Prompt - "C:\ProgramData\anaconda3\condabin\conda.bat" activate cellpose
vc                pkgs/main/win-64::vc-14.2-h2eaa2aa_1
vs2015_runtime    pkgs/main/win-64::vs2015_runtime-14.29.30133-h43f2093_3
wheel             pkgs/main/win-64::wheel-0.43.0-py311haa95532_0
xz                pkgs/main/win-64::xz-5.4.6-h8cc25b3_1
zlib              pkgs/main/win-64::zlib-1.2.13-h8cc25b3_1

Proceed ([y]/n)? y

Downloading and Extracting Packages
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#   $ conda activate cellpose
#
# To deactivate an active environment, use
#
#   $ conda deactivate

(base) C:\Users\mathieu.vigneau>conda create --name cellpose python=3.11
(base) C:\Users\mathieu.vigneau>conda activate cellpose
(cellpose) C:\Users\mathieu.vigneau>

```

- **Cellpose package installation**

Next, you need to install Cellpose application from your Cellpose python venv. In your opened anaconda prompt, type the following python command line, and press Enter:

```
python -m pip install cellpose[gui]
```

This command line will install Cellpose python packages in your python virtual environment, also including the **Cellpose Graphical user Interface**.

- **(Optional) Cellpose GPU package installation**

If you have a **CUDA supported GPU** installed on your computer (see [here](#) for CUDA GPU compatibility), you can install a GPU version of Cellpose.

Please see this [URL](#) for the install instructions of CUDA toolkit and GPU version of Cellpose.

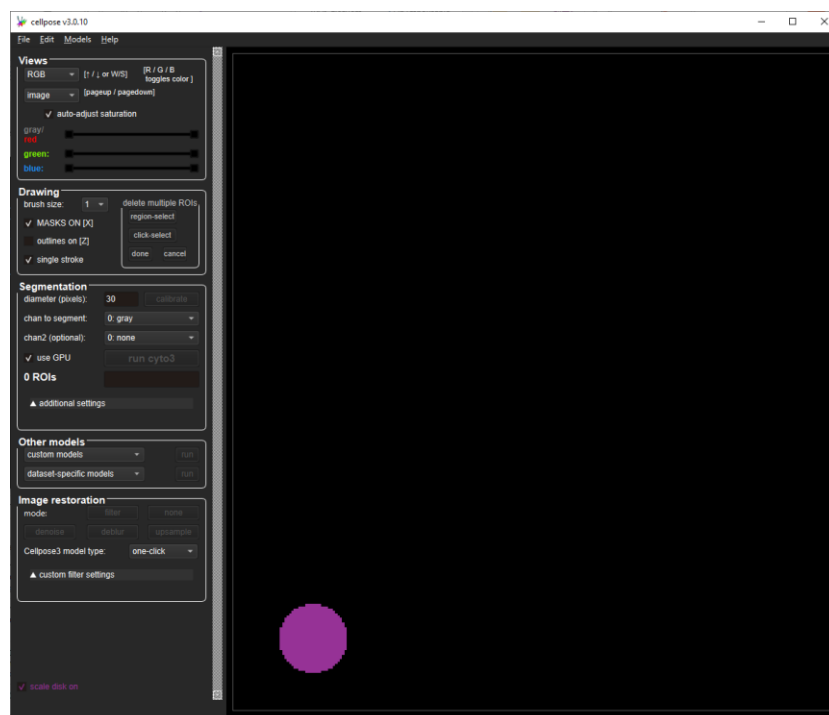
If you need more information about Cellpose detailed requirements, dependencies and installation instructions, see [here](#).

- **(Optional) Cellpose package checking**

Once you installed the Cellpose package, we recommend verifying the validity of the installation. From your Cellpose python environment, type the following python command lines and press Enter:

```
cellpose
```

If the package installation is correct, the following window will appear:





If you have issues with installation, see the cellpose [docs](#) for more details.

- **Cellpose environment deactivation**

Once the Cellpose installation is correct, **deactivate your Cellpose python environment** typing in your opened anaconda prompt the following python command line:

```
conda deactivate
```

## 2- FIJI INSTALLATION AND SETTING

---

### - **Fiji software installation**

---

Install Fiji software distribution using this following URL:

<https://imagej.net/software/fiji/downloads>

Download the latest version of Fiji for Windows 64-bit or Windows-32-bits using the *imagej.net* (USA) link. After downloading, go to your download folder and double-click the *Fiji-zip* file to uncompress it.

**It is highly recommended that you store your Fiji app directory in your user space (C:\Users\username\MyDocuments\FijiApp) rather than in the Program Files windows-system directory (write permission issues in these system-wide folders could prevent Fiji updating)**

See [here](#) for detailed installation instructions and system requirements of Fiji application

### - **FIJI plugin updating & setting**

---

EMito-Metrix plugin relies on the following Fiji plugins, which are not automatically installed during the Fiji software installation:

- Bio-Formats (<https://imagej.net/formats/bio-formats>)

*The Bio-Formats Importer is a plugin for reading data into Fiji. It can open many dozens of proprietary life sciences formats*

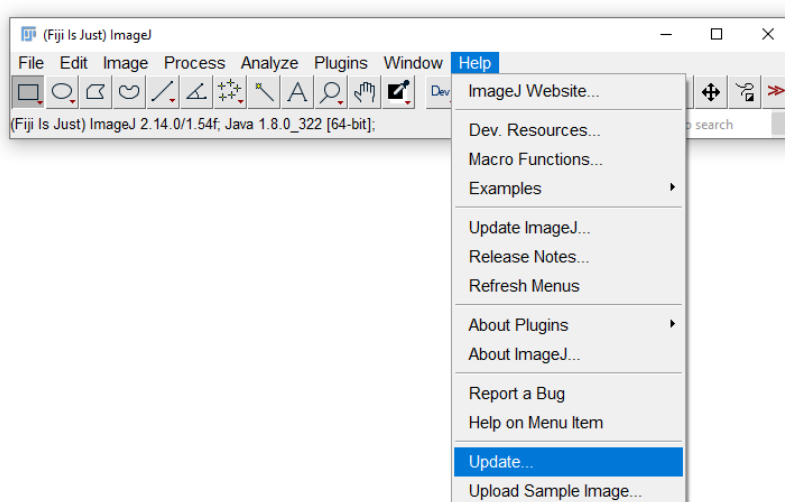
- ImageScience (<https://imagej.net/libs/imagescience>)

*ImageScience is a library of Java classes to facilitate scientific image computing.*

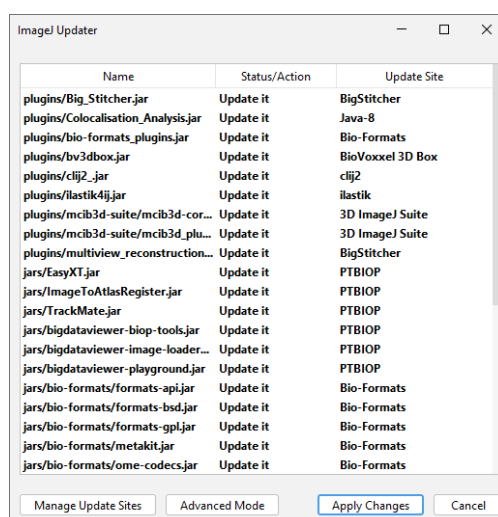
- PTBIOP (<https://wiki-biop.epfl.ch/en/ipa/fiji/update-site>)

*The EPFL - PTBIOP (BioImaging & Optics Platform) is a Fiji package including plugins, macros, ActionBars and Wrappers for external software calls (Cellpose, Elastix, Ilastix...)*

- In your Fiji App directory, double-click the *ImageJ-win64.exe* file to the start Fiji software
- Once the software is opened, select *update* from the *Fiji Help menu* (as shown below) in order to update Fiji plugins

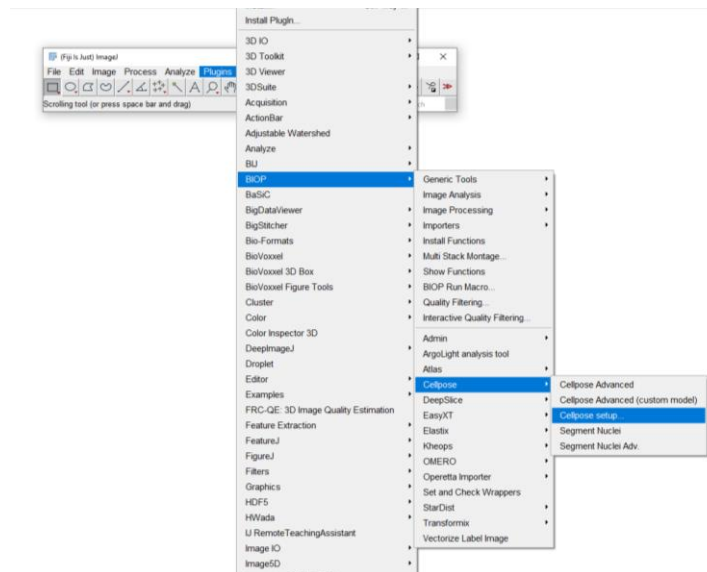


- In the *ImageJ Updater* window, select the *Manage update sites* button (see below).



- The *Manage Update Sites* window appears, which contains a list of Fiji plugins you can upload and install in your Fiji application (plugins already installed are checked).

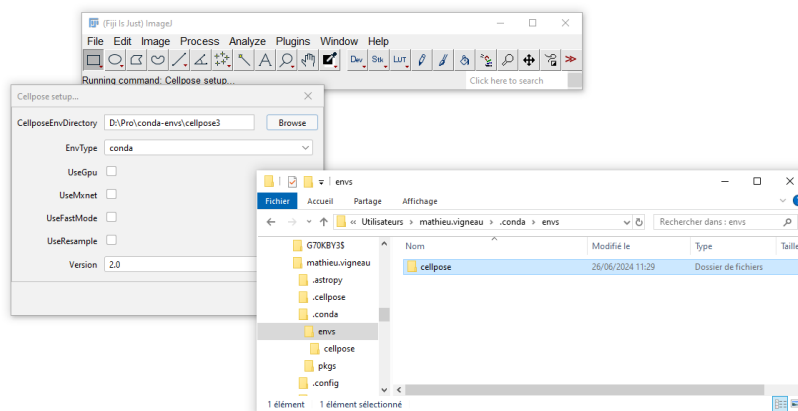




- The *Cellpose setup* window appears which allows setting Cellpose wrapper. From this window, modify the following settings:

- CellposeEnvDirectory: select your own Cellpose Python environment folder

*C:\Users\username\AppData\Local\anaconda3\envs\cellpose* or *C:\Users\username\AppData\anaconda\envs\cellpose*



- EnvType: python package chosen for your Cellpose environment

*Choose **conda** if you have installed Anaconda distribution for Python*

- UseGpu: allows using Cellpose with GPU computing (faster than CPU computing)

*Activate the checkbox if you have installed a Cellpose GPU version*

- Version: version of Cellpose launch from the Fiji wrapper

*Choose 2.0*

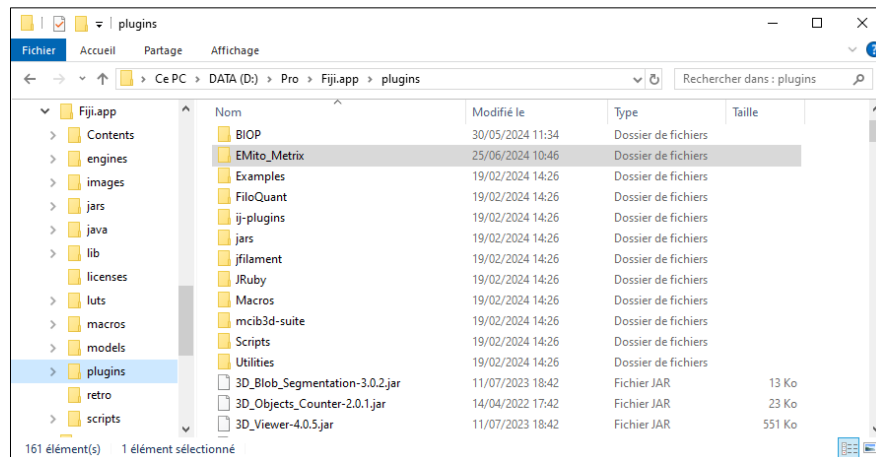
- Then press OK

## - **EMito-Metrix plugin installation & setting**

---

- **EMito Metrix folder creation**

In your Fiji plugins folder (*yourFijiFolder\plugins\*), create a new folder named **EMito\_Metrix**



- **EMito Metrix file upload**

**Download and save the EMito\_Metrix files** in your *EMito\_Metrix Fiji folder*, using the following GitHub repository:

<https://github.com/Restore-Lab/EMito-Metrix>

**You must upload these following files from the GitHub repository (!! Keep the same file names !!) :**

- EMito\_Metrix\_Fiji.ijm (Fiji plugin file)
- dataComputation.py (python file for data display)
- mlComputation.py (python file for data computation and prediction)
- requirements.txt (python package dependencies)
- GeneralistModel\_GM\_EM.zip (Cellpose GM model)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

- SpecialistModel\_SM\_FLY\_EM.zip (Cellpose SM model for Fly)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

- SpecialistModel\_SM\_HUMAN\_EM.zip (Cellpose SM model for Human)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

- SpecialistModel\_SM\_MOUSE\_EM.zip (Cellpose SM model for Mouse)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

- SpecialistModel\_SM\_ZEBRAFISH\_EM.zip (Cellpose SM model for Zebrafish)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

### 3- EMITOMETRIX PYTHON ENVIRONMENT: INSTALLATION & SETTING

---

- **Create the Emitometrix Python environment**

We are going to create a **Python virtual environment (venv)** for data computation plugins.

*A virtual environment is simply a Python subdirectory with its own file structure, python packages and settings. This Virtual environment let you have a stable and reproducible environment for data visualization and computation plugins, which are essential for the proper running of the EMito-Metrix application.*

To create the emitometrix venv, type the following python command line in your anaconda prompt, and press Enter:

```
conda create --name emitometrix python=3.11
```

This command line **will create a new python3.11 environment named *emitometrix***.

This will create a new folder containing your Emito-Metrix python environment. This folder is found under the following Windows path:

*C:\Users\username\AppData\Local\anaconda3\envs\emitometrix or*

*C:\Users\username\AppData\anaconda\envs\emitometrix*

- **Emitometrix environment activation**

To **activate your *emitometrix* python environment**, type the following python command line in your opened anaconda prompt, and press Enter:

```
conda activate emitometrix
```

- **Python dependencies installation**

EMito-Metrix plugin relies on the following Python packages, which are not automatically installed during emitometrix python environment installation:

- cloudpickle==3.0.0	- matplotlib==3.9.0	- pyparsing==3.1.2	- six==1.16.0
- contourpy==1.2.1	- numba==0.60.0	- python-dateutil==2.9.0.post0	- slicer==0.0.8
- cycler==0.12.1	- numpy==1.26.4	- pytz==2024.1	- threadpoolctl==3.5.0
- fonttools==4.53.0	- packaging==24.1	- scikit-learn==1.5.1	- tqdm==4.66.4
- joblib==1.4.2	- pandas==2.2.2	- scipy==1.14.0	- tzdata==2024.1
- kiwisolver==1.4.5	- pillow==10.4.0	- seaborn==0.13.2	- umap-learn==0.5.6
- llvmlite==0.43.0	- pynndescent==0.5.13	- shap==0.46.0	- xgboost==2.1.0

The list of these packages is included in the *requirements.txt* file that is found under your *EMito\_Metrix Fiji* folder

*example of EMitometrix folder : C:\Users\username\Documents\Fiji.app\plugins\EMito\_Metrix*

**To install python dependencies in your emitometrix environment**, type the following python command line from your opened anaconda prompt, and press Enter:

```
pip install -r yourFijiFolfer\plugins\EMito_Metrix\requirements.txt
```

- **Cellpose environment deactivation**

Once the Cellpose installation is correct, **deactivate your Emitometrix python environment** typing in your opened anaconda prompt the following python command line:

```
conda deactivate
```

**Once you correctly installed these software and environments, the EMito-Metrix application is ready for running. See the *EMitoMetrix tutorial* for running instructions.**

## EMITO-METRIX INSTALLATION FOR LINUX

### 1- CELLOSE PYTHON ENVIRONMENT: INSTALLATION AND SETTING

---

#### - **Anaconda distribution installation**

---

Install an Anaconda distribution for Python 3 using the following URL:

<https://www.anaconda.com/download/success>

Download the latest version of Anaconda installer, depending on your Linux architecture.

See [here](#) for detailed instructions of Anaconda installation and settings for Linux operating system

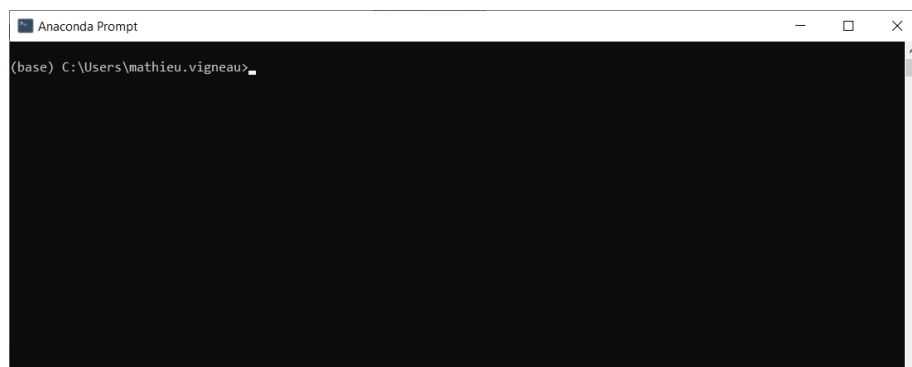
#### - **Cellpose installation & setting**

---

Once you installed anaconda environment, you will need to create and set a python virtual environment for Cellpose application (which is essential for the EMito-Metrix application).

- **Open an anaconda prompt**

Open a terminal window. This will open a new command prompt window with Anaconda enabled (as shown below).





- **Create the Cellpose Python environment**

We are going to create a **Python virtual environment (venv) for Cellpose** application.

*A virtual environment is simply a Python subdirectory with its own file structure, python packages and settings. This Virtual environment let you have a stable and reproducible Cellpose environment, which is essential for the proper running of the EMito-Metrix application.*

To create the Cellpose venv, type the following python command line in your opened anaconda prompt, and press Enter:

```
conda create --name cellpose python=3.9
```

This command line **will create a new python3.9 environment named *cellpose***.

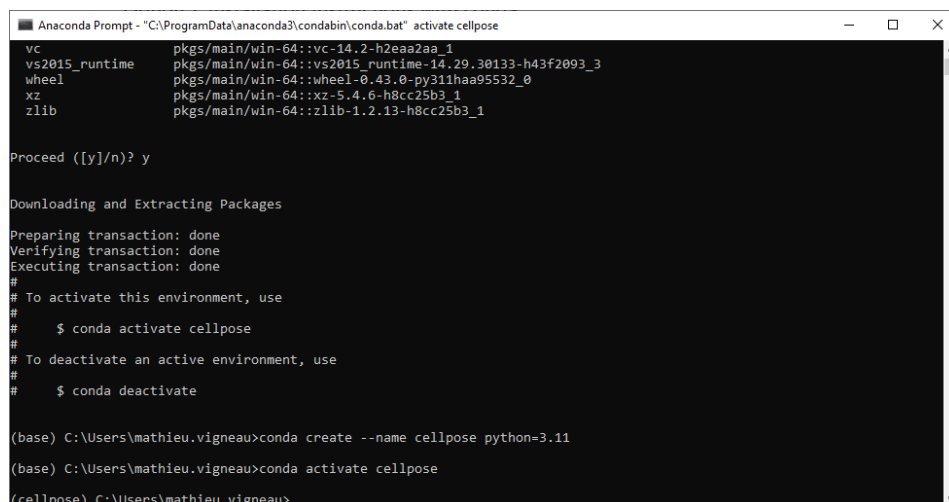
This will create a new folder, containing your Cellpose python environment. This cellpose folder is found under the following Linux path:

*/home/username/anaconda3/envs/cellpose*

- **Cellpose environment activation**

To **activate your Cellpose python environment**, type the following python command line in your opened anaconda prompt, and press Enter:

```
conda activate cellpose
```



```
Anaconda Prompt - "C:\ProgramData\anaconda3\condabin\conda.bat" activate cellpose
vc                pkgs/main/win-64::vc-14.2-h2eaa2aa_1
vs2015_runtime    pkgs/main/win-64::vs2015_runtime-14.29.30133-h43f2093_3
wheel             pkgs/main/win-64::wheel-0.43.0-py311haa95532_0
xz                pkgs/main/win-64::xz-5.4.6-h8cc25b3_1
zlib              pkgs/main/win-64::zlib-1.2.13-h8cc25b3_1

Proceed ([y]/n)? y

Downloading and Extracting Packages
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate cellpose
#
# To deactivate an active environment, use
#
#     $ conda deactivate
#

(base) C:\Users\mathieu.vigneau>conda create --name cellpose python=3.11
(base) C:\Users\mathieu.vigneau>conda activate cellpose
(cellpose) C:\Users\mathieu.vigneau>
```

- **Cellpose package installation**

Next, you need to install Cellpose application from your Cellpose python venv. In your opened anaconda prompt, type the following python command line, and press Enter:

```
python -m pip install cellpose[gui]
```

This command line will install Cellpose python packages in your python virtual environment, also including the **Cellpose Graphical user Interface**.

- **(Optional) Cellpose GPU package installation**

If you have a **CUDA supported GPU** installed on your computer (see [here](#) for CUDA GPU compatibility), you can install a GPU version of Cellpose.

Please see this [URL](#) for the install instructions of CUDA toolkit and GPU version of Cellpose.

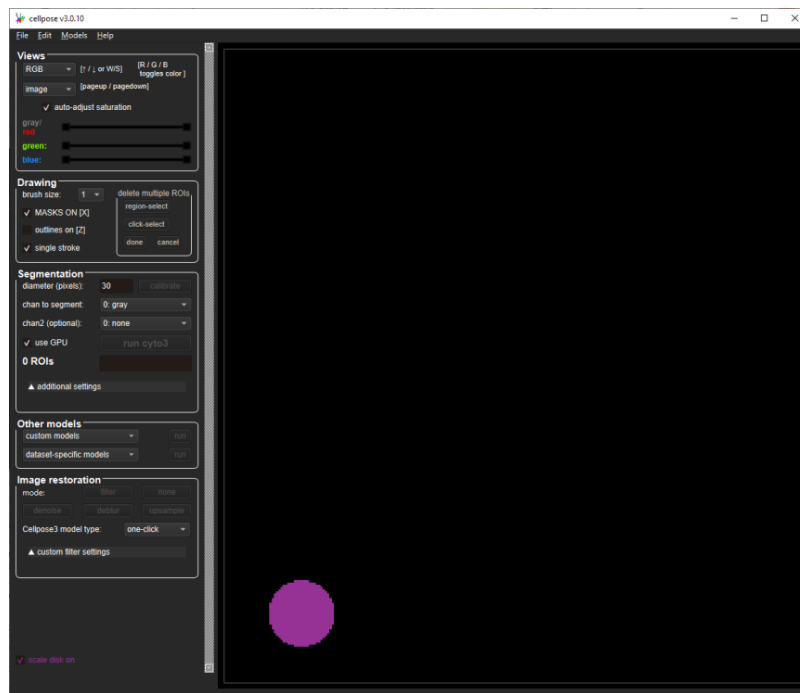
If you need more information about Cellpose detailed requirements, dependencies and installation instructions, see [here](#).

- **(Optional) Cellpose package checking**

Once you installed the Cellpose package, we recommend verifying the validity of the installation. From your Cellpose python environment, type the following python command lines and press Enter:

```
cellpose
```

If the package installation is correct, the following window will appear:



If you have issues with installation, see the cellpose [docs](#) for more details.

- **Cellpose environment deactivation**

Once the Cellpose installation is correct, **deactivate your Cellpose python environment** typing in your opened anaconda prompt the following python command line:

```
conda deactivate
```

## 2- FIJI INSTALLATION AND SETTING

### - Fiji software installation

Install Fiji software distribution using this following URL:

<https://imagej.net/software/fiji/downloads>

Download the latest version of Fiji for Linux 64-bit using the *imagej.net (USA)* link. After downloading, go to your download folder and double-click the *Fiji-zip* file to uncompress it.

See [here](#) for detailed installation instructions and system requirements of Fiji application

## - FIJI plugin updating & setting

---

EMito-Metrix plugin relies on the following Fiji plugins, which are not automatically installed during the Fiji software installation:

- Bio-Formats (<https://imagej.net/formats/bio-formats>)

*The Bio-Formats Importer is a plugin for reading data into Fiji. It can open many dozens of proprietary life sciences formats*

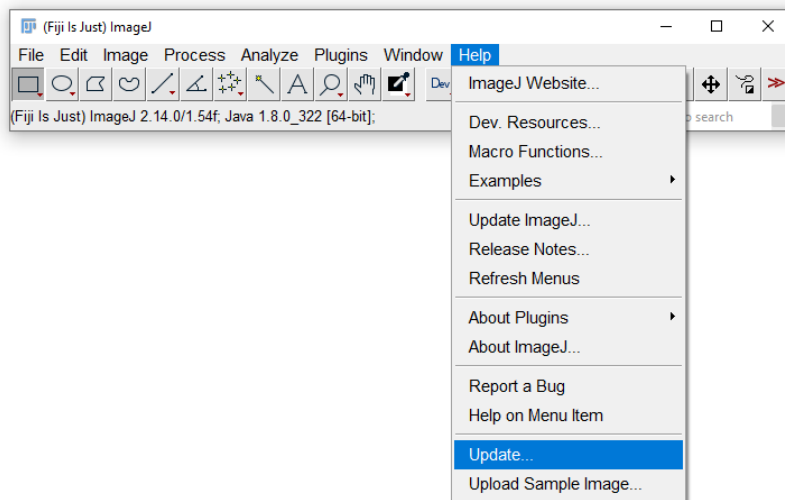
- ImageScience (<https://imagej.net/libs/imagescience>)

*ImageScience is a library of Java classes to facilitate scientific image computing.*

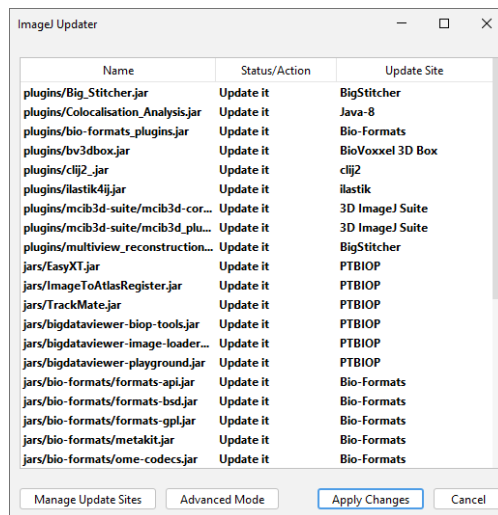
- PTBIOP (<https://wiki-biop.epfl.ch/en/ipa/fiji/update-site>)

*The EPFL - PTBIOP (BioImaging & Optics Platform) is a Fiji package including plugins, macros, ActionBars and Wrappers for external software calls (Cellpose, Elastix, Ilastix...)*

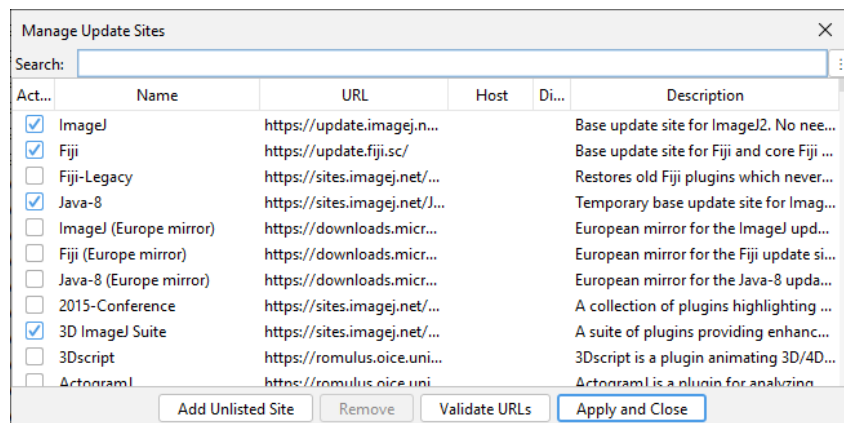
- In your Fiji App directory, double-click the *ImageJ-win64.exe* file to start Fiji software
- Once the software is opened, select *update* from the *Fiji Help menu* (as shown below) in order to update Fiji plugins



- In the *ImageJ Updater* window, select the *Manage update sites* button (see below).



- The *Manage Update Sites* window appears, which contains a list of Fiji plugins you can upload and install in your Fiji application (plugins already installed are checked).



From this *Manage Update Sites* window, check the following plugins to install it:

- Bio-Formats
- IBMP-CNRS
- ImageScience
- PTBIOP

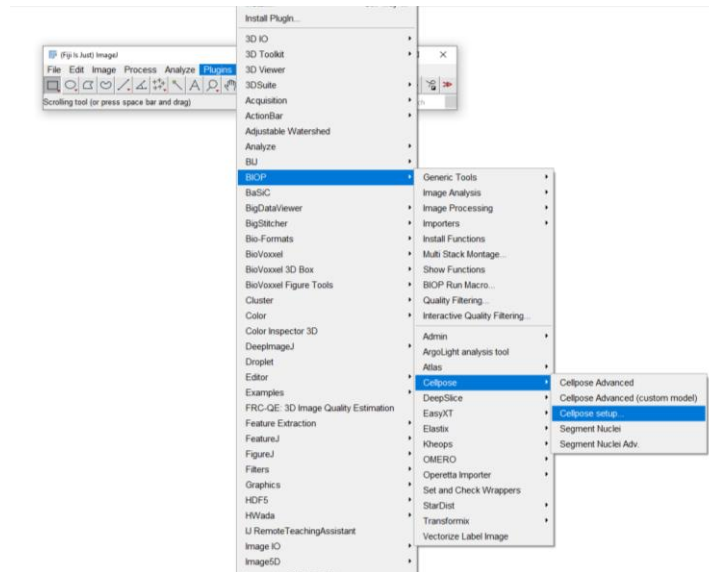
- Once these plugins are checked, click on *Apply and Close* button from the *Manage Update Sites* window. Then, click on *Apply Changes* button from the *ImageJ Updater* window
- Once the plugin are installed, restart Fiji application

## - Cellpose wrapper settings

---

Cellpose wrapper is a command from PTBIOP plugin that enables performing Cellpose object segmentation through Fiji application, using a python virtual environment with Cellpose package installed (see [here](#) to create Cellpose virtual environment).

- In your Fiji App, select *BIOP* from the Fiji *Plugins* menu. Then, select *Cellpose setup* in the *Cellpose* menu, as shown below :



- The *Cellpose setup* window appears which allows setting Cellpose wrapper. From this window, modify the following settings:

- CellposeEnvDirectory: select your own Cellpose Python environment folder

*/home/username/anaconda3/envs/cellpose/*

- EnvType: python package chosen for your Cellpose environment

*Choose **conda** if you have installed Anaconda distribution for Python*

- UseGpu: allows using Cellpose with GPU computing (faster than CPU computing)

*Activate the checkbox if you have installed a Cellpose GPU version*

- Version: version of Cellpose launch from the Fiji wrapper

*Choose 2.0*

- Then press OK

## - EMito-Metrix plugin installation & setting

---

- **EMito\_Metrix folder creation**

In your Fiji plugins folder (*yourFijiFolder\plugins\*), create a new folder named **EMito\_Metrix**

- **EMito\_Metrix file upload**

**Download and save the EMito\_Metrix files** in your *EMito\_Metrix Fiji folder*, using the following GitHub repository:

<https://github.com/Restore-Lab/EMito-Metrix>

**You must upload these following files from the GitHub repository (!! Keep the same file names !!) :**

- EMito\_Metrix\_Fiji\_.ijm (Fiji plugin file)
- dataComputation.py (python file for data display)
- mlComputation.py (python file for data computation and prediction)
- requirements.txt (python package dependencies)
- GeneralistModel\_GM\_EM.zip (Cellpose GM model)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

- SpecialistModel\_SM\_FLY\_EM.zip (Cellpose SM model for Fly)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

- SpecialistModel\_SM\_HUMAN\_EM.zip (Cellpose SM model for Human)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

- SpecialistModel\_SM\_MOUSE\_EM.zip (Cellpose SM model for Mouse)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

- SpecialistModel\_SM\_ZEBRAFISH\_EM.zip (Cellpose SM model for Zebrafish)

***Once copied, unzip the file in the EMito\_Metrix folder and delete the zip archive***

### 3- EMITOMETRIX PYTHON ENVIRONMENT: INSTALLATION & SETTING

---

- **Create the Emitometrix Python environment**

We are going to create a **Python virtual environment (venv)** for data computation plugins.

*A virtual environment is simply a Python subdirectory with its own file structure, python packages and settings. This Virtual environment let you have a stable and reproducible environment for data visualization and computation plugins, which are essential for the proper running of the EMito-Metrix application.*

To create the emitometrix venv, type the following python command line in your anaconda prompt, and press Enter:

```
conda create --name emitometrix python=3.11
```

This command line **will create a new python3.11 environment named *emitometrix***.

This will create a new folder containing your Emito-Metrix python environment. This folder is found under the following Linux path:

*/home/username/anaconda3/envs/emitometrix/*

- **Emitometrix environment activation**

To activate your ***emitometrix* python environment**, type the following python command line in your opened anaconda prompt, and press Enter:

```
conda activate emitometrix
```



- **Python dependencies installation**

EMito-Metrix plugin relies on the following Python packages, which are not automatically installed during emitometrix python environment installation:

- cloudpickle==3.0.0	- matplotlib==3.9.0	- pyparsing==3.1.2	- six==1.16.0
- contourpy==1.2.1	- numba==0.60.0	- python-dateutil==2.9.0.post0	- slicer==0.0.8
- cycler==0.12.1	- numpy==1.26.4	- pytz==2024.1	- threadpoolctl==3.5.0
- fonttools==4.53.0	- packaging==24.1	- scikit-learn==1.5.1	- tqdm==4.66.4
- joblib==1.4.2	- pandas==2.2.2	- scipy==1.14.0	- tzdata==2024.1
- kiwisolver==1.4.5	- pillow==10.4.0	- seaborn==0.13.2	- umap-learn==0.5.6
- llvmlite==0.43.0	- pynndescent==0.5.13	- shap==0.46.0	- xgboost==2.1.0

The list of these packages is included in the *requirements.txt* file that is found under your *EMito\_Metrix Fiji* folder

*example of EMitometrix folder : /home/username/Fiji.app/plugins/EMito\_Metrix*

**To install python dependencies in your emitometrix environment**, type the following python command line from your opened anaconda prompt, and press Enter:

```
pip install -r yourFijiFolfer\plugins\EMito_Metrix\requirements.txt
```

- **Cellpose environment deactivation**

Once the Cellpose installation is correct, **deactivate your Emitometrix python environment** typing in your opened anaconda prompt the following python command line:

```
conda deactivate
```

**Once you correctly installed these software and environments, the EMito-Metrix application is ready for running. See the *EMitoMetrix tutorial* for running instructions.**