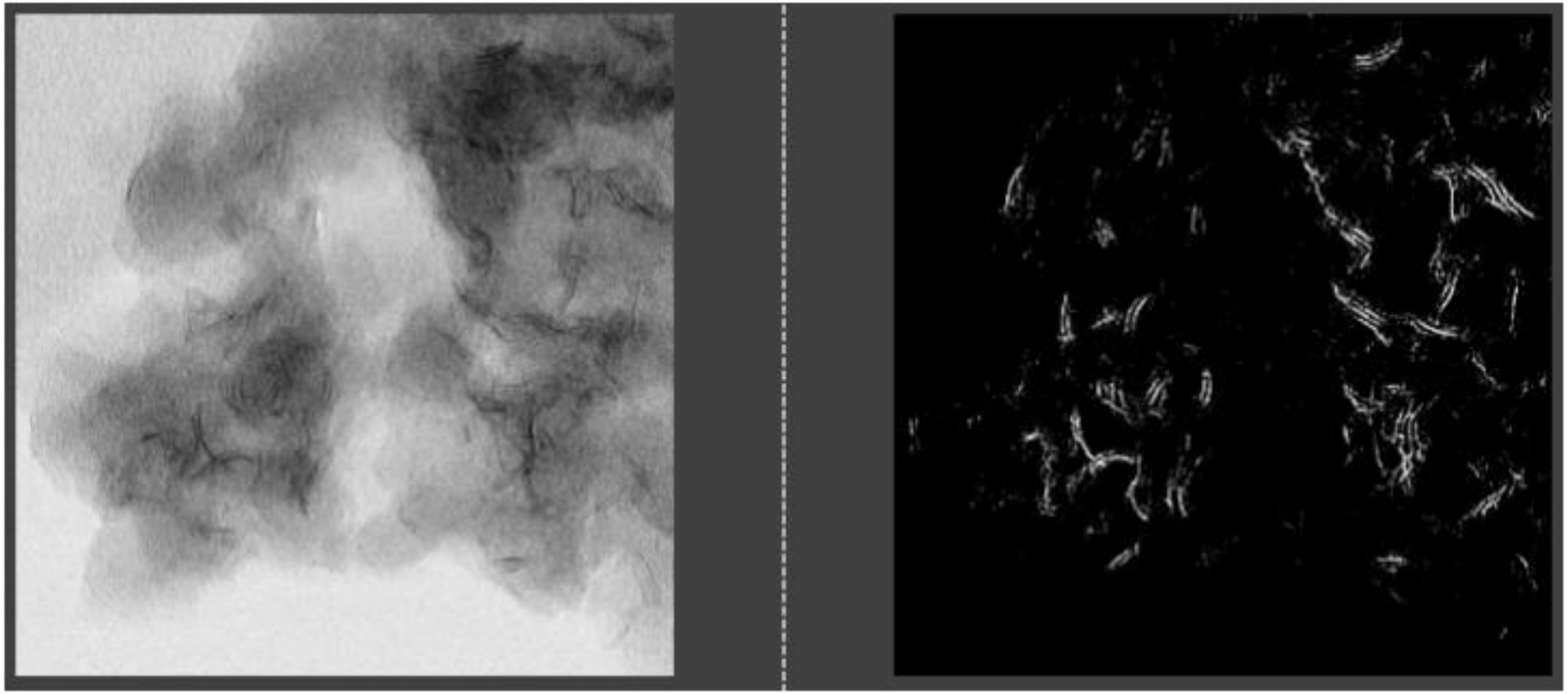
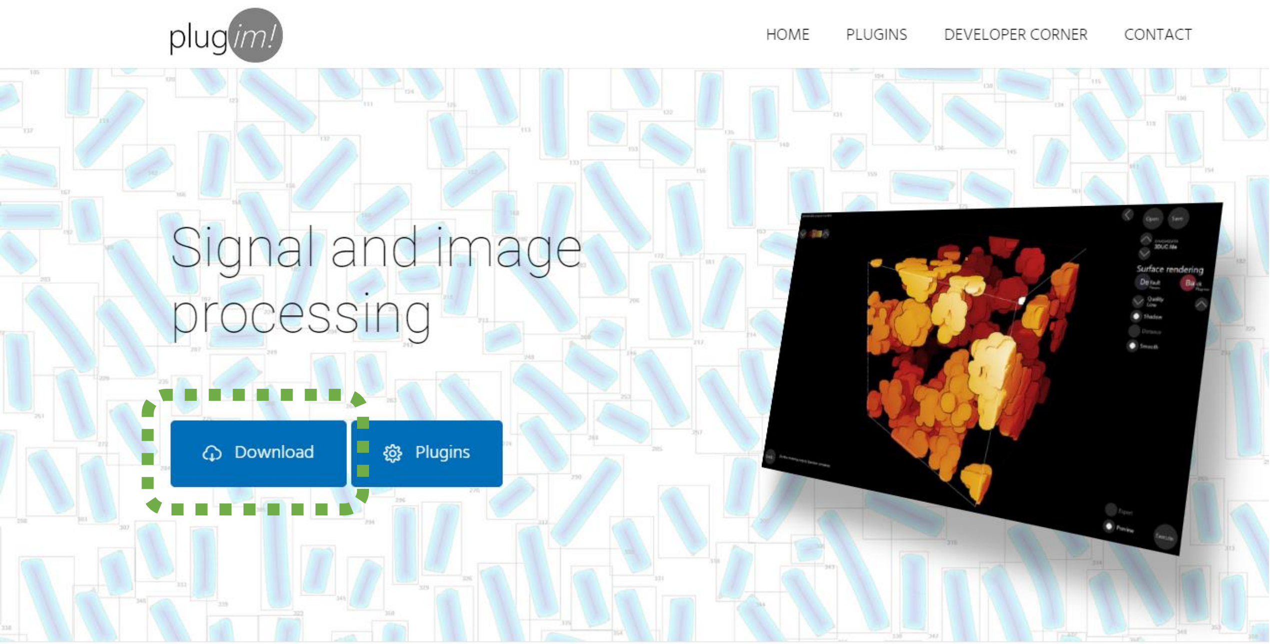
Semantic segmentation with deep learning, a comprehensive tutorial with plug im!

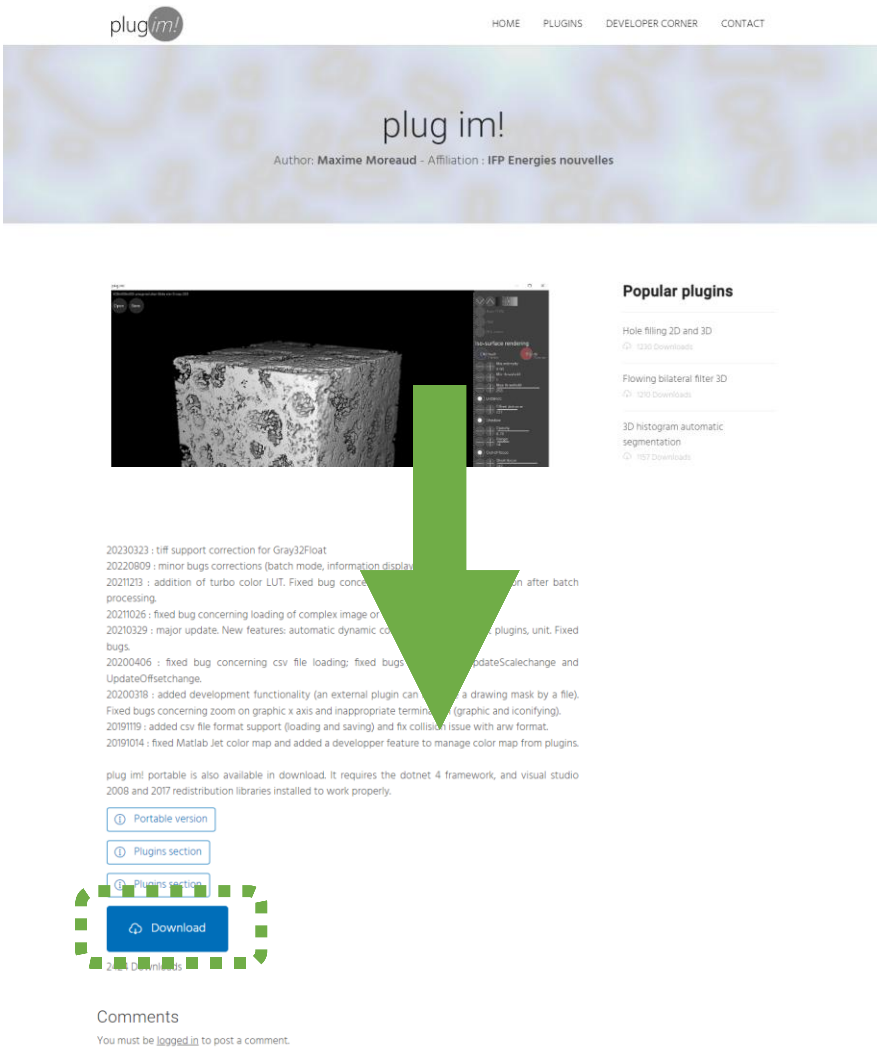


Download and install plug im software

- www.plugim.fr



Signal and image processing



plug im!

Author: **Maxime Moreaud** - Affiliation : **IFP Energies nouvelles**

Popular plugins

Hole filling 2D and 3D

Flowing bilateral filter 3D

3D histogram automatic segmentation

20230323 : tiff support correction for Gray32Float
20220809 : minor bugs corrections (batch mode, information display [illegible]
20211213 : addition of turbo color LUT. Fixed bug conce[illegible]on after batch processing.
20211026 : fixed bug concerning loading of complex image or [illegible]
20210329 : major update. New features: automatic dynamic co[illegible] plugins, unit. Fixed bugs.
20200406 : fixed bug concerning csv file loading; fixed bugs [illegible]pdateScalechange and UpdateOffsetchange.
20200318 : added development functionality (an external plugin can [illegible] a drawing mask by a file). Fixed bugs concerning zoom on graphic x axis and inappropriate termina[illegible] (graphic and iconifying).
20191119 : added csv file format support (loading and saving) and fix collision issue with arw format.
20191014 : fixed Matlab Jet color map and added a developper feature to manage color map from plugins.

plug im! portable is also available in download. It requires the dotnet 4 framework, and visual studio 2008 and 2017 redistribution libraries installed to work properly.

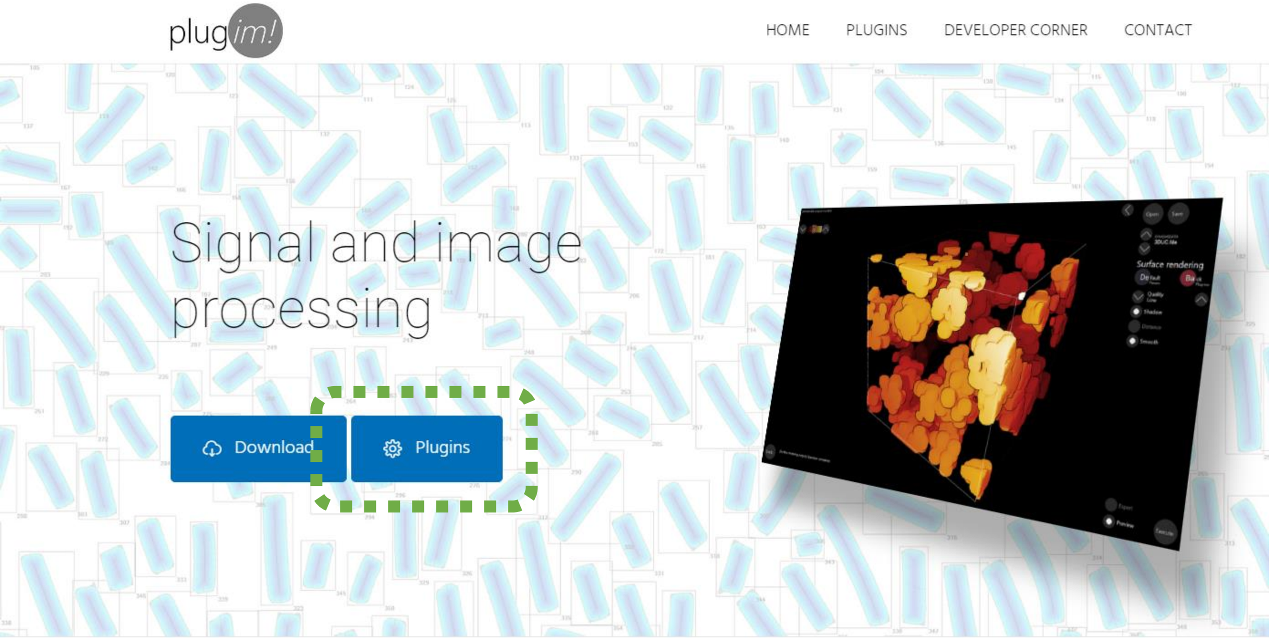
Comments

You must be logged in to post a comment.

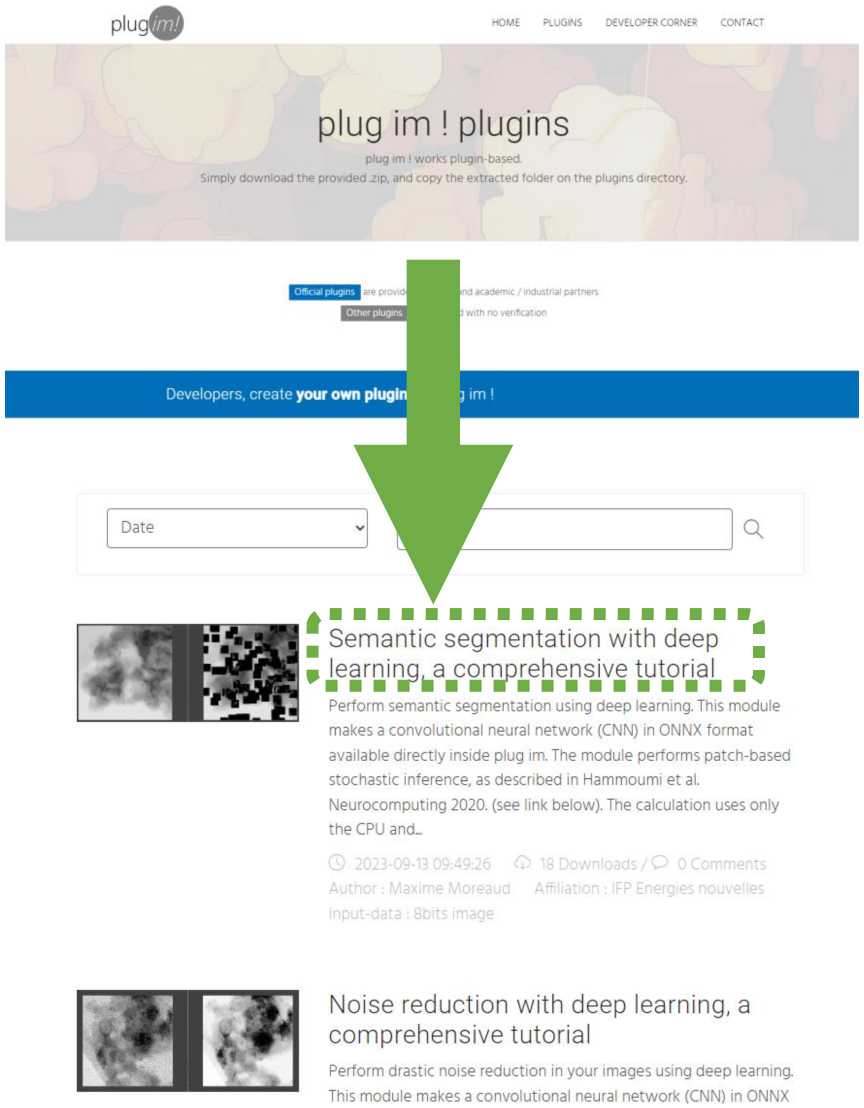
- Follow installation procedure

Download “Segmentation” Package (1/2)

- www.plugim.fr



Signal and image processing



plug im ! plugins

plug im ! works plugin-based.
Simply download the provided .zip, and copy the extracted folder on the plugins directory.

Official plugins are provid … and academic / industrial partners

Other plugins … d with no verification

Developers, create **your own plugin** … g im !

Semantic segmentation with deep learning, a comprehensive tutorial

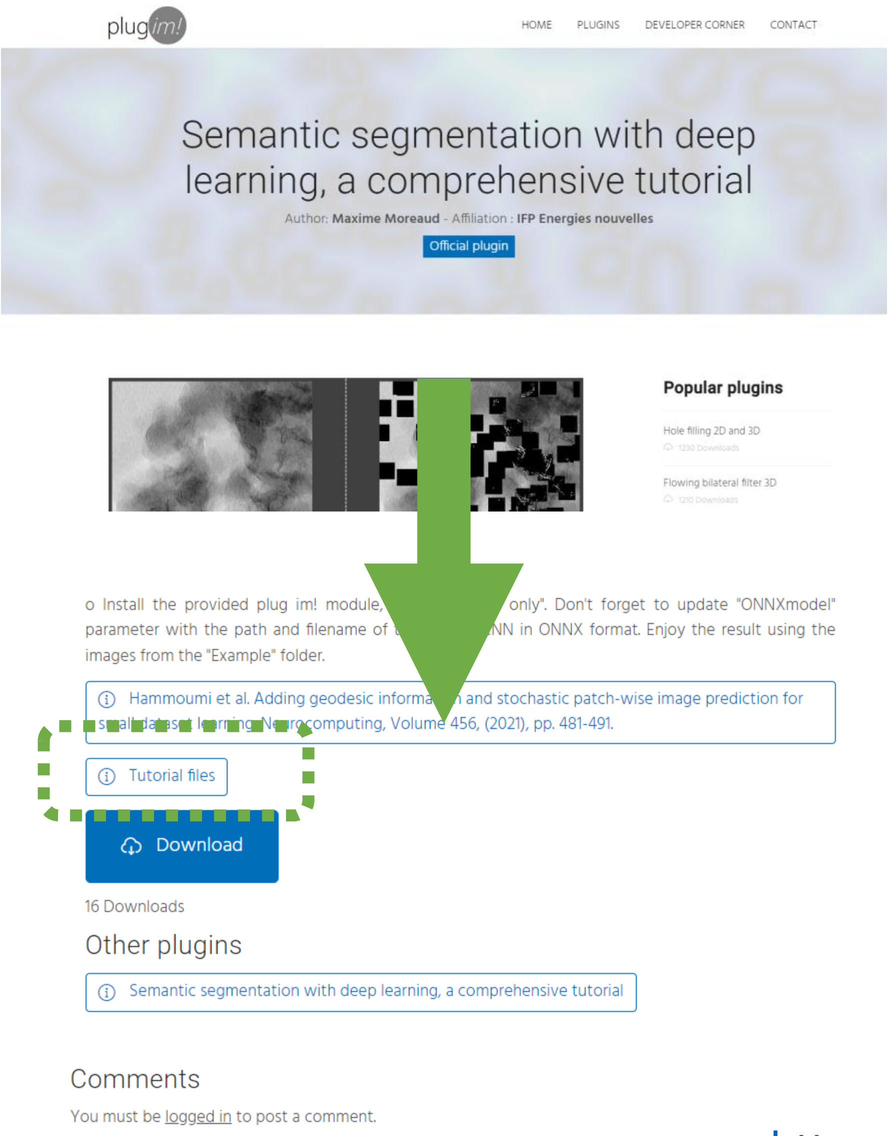
Perform semantic segmentation using deep learning. This module makes a convolutional neural network (CNN) in ONNX format available directly inside plug im. The module performs patch-based stochastic inference, as described in Hammoumi et al. Neurocomputing 2020. (see link below). The calculation uses only the CPU and...

2023-09-13 09:49:26 18 Downloads / 0 Comments
Author : Maxime Moreaud Affiliation : IFP Energies nouvelles
Input-data : 8bits image

Noise reduction with deep learning, a comprehensive tutorial

Perform drastic noise reduction in your images using deep learning. This module makes a convolutional neural network (CNN) in ONNX

Download “Segmentation” Package (2/2)



Semantic segmentation with deep learning, a comprehensive tutorial

Author: **Maxime Moreaud** - Affiliation : **IFP Energies nouvelles**

Popular plugins

Hole filling 2D and 3D

1230 Downloads

Flowing bilateral filter 3D

1210 Downloads

o Install the provided plug im! module, only". Don't forget to update "ONNXmodel" parameter with the path and filename of NN in ONNX format. Enjoy the result using the images from the "Example" folder.

ⓘ Hammoumi et al. Adding geodesic informa n and stochastic patch-wise image prediction for small dataset learning, Neurocomputing, Volume 456, (2021), pp. 481-491.

16 Downloads

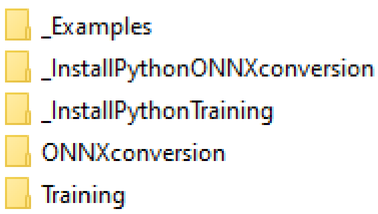
Other plugins

ⓘ Semantic segmentation with deep learning, a comprehensive tutorial

Comments

You must be logged in to post a comment.

- Package :



_Examples
_InstallPythonONNXconversion
_InstallPythonTraining
ONNXconversion
Training

https://www.plugim.fr/plugin/120

Tutorial steps

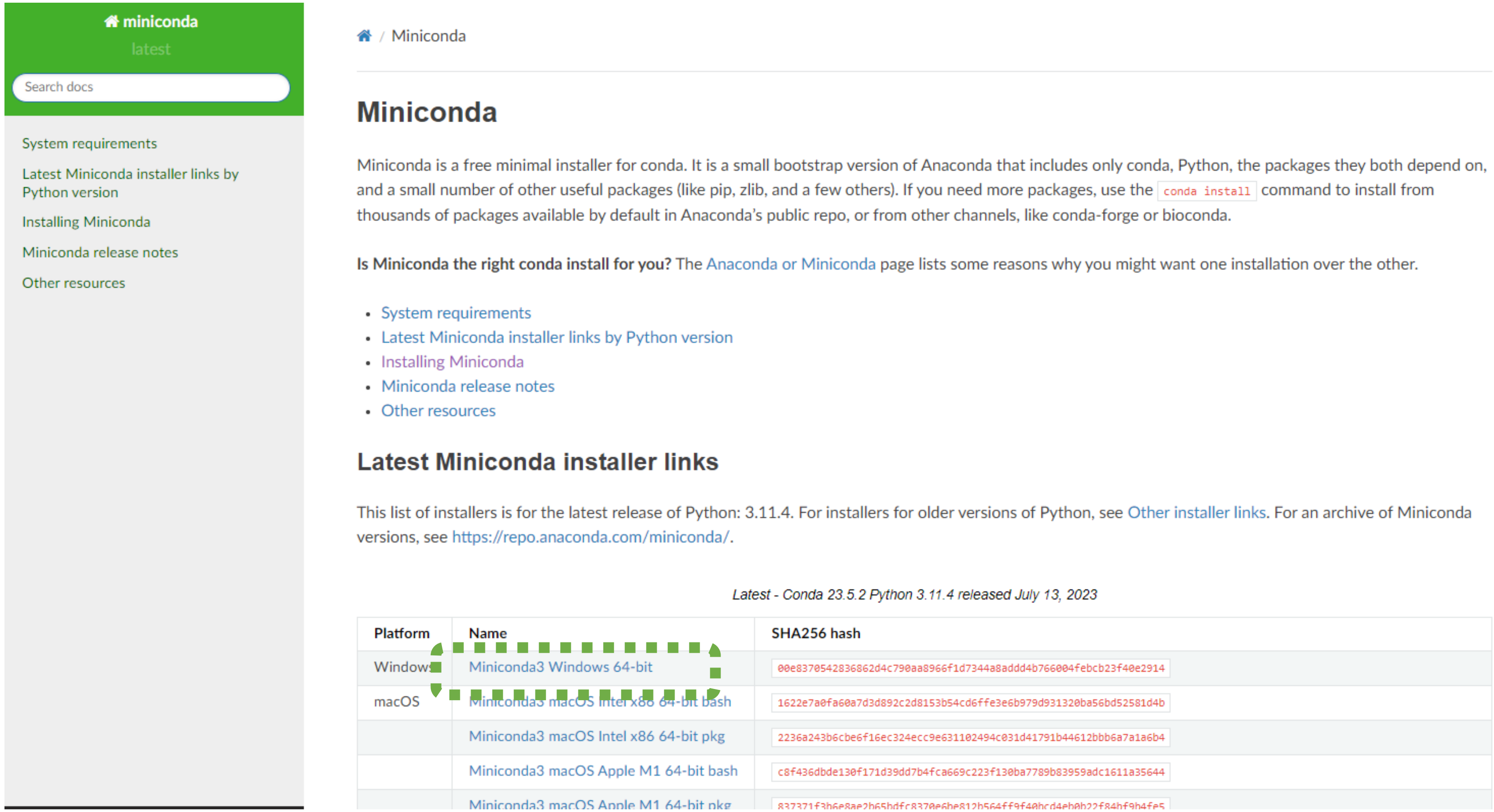
- Training
  - Miniconda installation
  - Python environnement installation
  - Jupyter Notebook

- ONNX conversion
  - Python environnement installation
  - Jupyter Notebook

- Inference with plug im!
  - Plugin installation

Training / Miniconda installation

- Install Miniconda3 for Windows 64-bit with default instructions
- https://docs.conda.io/projects/miniconda/en/latest/index.html



Miniconda

Miniconda is a free minimal installer for conda. It is a small bootstrap version of Anaconda that includes only conda, Python, the packages they both depend on, and a small number of other useful packages (like pip, zlib, and a few others). If you need more packages, use the `conda install` command to install from thousands of packages available by default in Anaconda's public repo, or from other channels, like conda-forge or bioconda.

Is Miniconda the right conda install for you? The Anaconda or Miniconda page lists some reasons why you might want one installation over the other.

- System requirements
- Latest Miniconda installer links by Python version
- Installing Miniconda
- Miniconda release notes
- Other resources

Latest Miniconda installer links

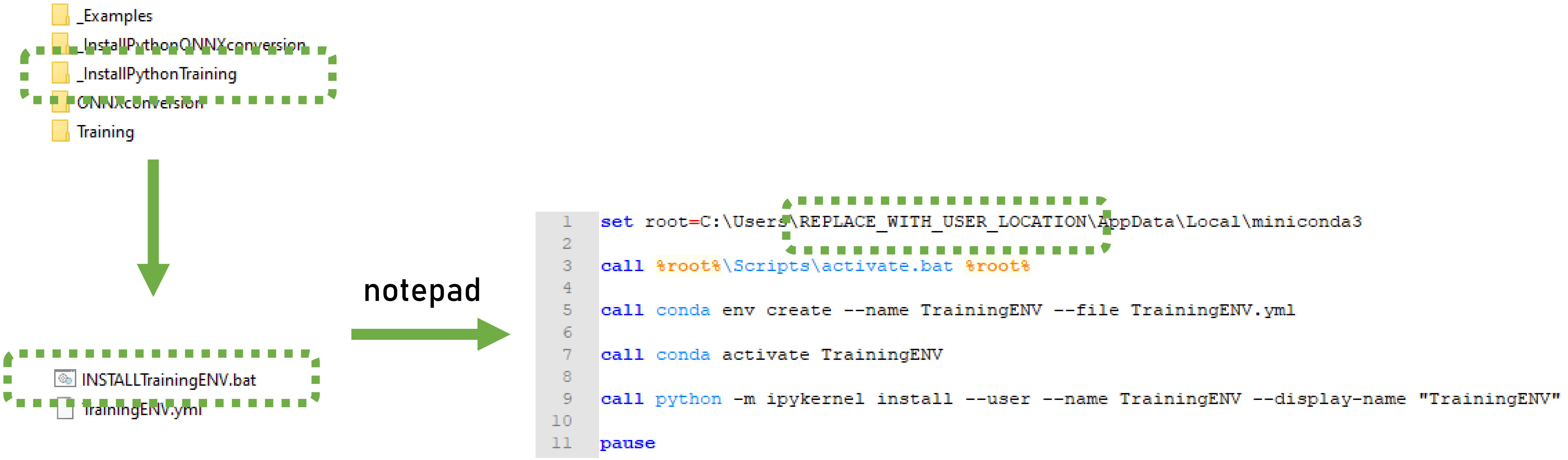
This list of installers is for the latest release of Python: 3.11.4. For installers for older versions of Python, see Other installer links. For an archive of Miniconda versions, see https://repo.anaconda.com/miniconda/.

Latest - Conda 23.5.2 Python 3.11.4 released July 13, 2023

Platform	Name	SHA256 hash
Windows	Miniconda3 Windows 64-bit	`00e8370542836862d4c790aa8966f1d7344a8addd4b766004febcb23f40e2914`
macOS	Miniconda3 macOS Intel x86 64-bit bash	`1622e7a0fa60a7d3d892c2d8153b54cd6ffe3e6b979d931320ba56bd52581d4b`
	Miniconda3 macOS Intel x86 64-bit pkg	`2236a243b6cbe6f16ec324ecc9e631102494c031d41791b44612bbb6a7a1a6b4`
	Miniconda3 macOS Apple M1 64-bit bash	`c8f436dbde130f171d39dd7b4fca669c223f130ba7789b83959adc1611a35644`
	Miniconda3 macOS Apple M1 64-bit pkg	`837371f3b6e8ae2b65bdfc8370e6be812b564ff9f40bcd4eb0b22f84bf9b4fe5`

Training / Python environment installation

- Update _InstallPythonTraining / INSTALLTrainingENV.bat

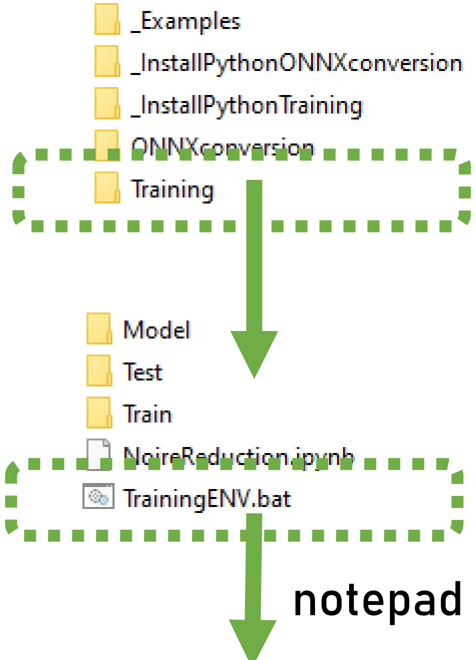


```
set root=C:\Users\REPLACE_WITH_USER_LOCATION\AppData\Local\miniconda3

call %root%\Scripts\activate.bat %root%

call conda env create --name TrainingENV --file TrainingENV.yml

call conda activate TrainingENV

call python -m ipykernel install --user --name TrainingENV --display-name "TrainingENV"

pause
```

- Double click on INSTALLTrainingENV.bat

Training / Jupyter notebook (1/2)

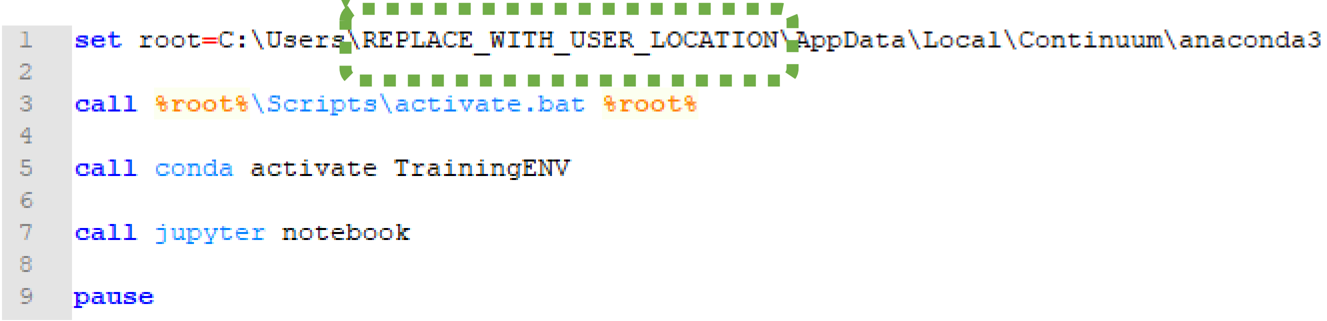
- Update Training / TrainingENV.bat



_Examples
_InstallPythonONNXconversion
_InstallPythonTraining
ONNXconversion
Training

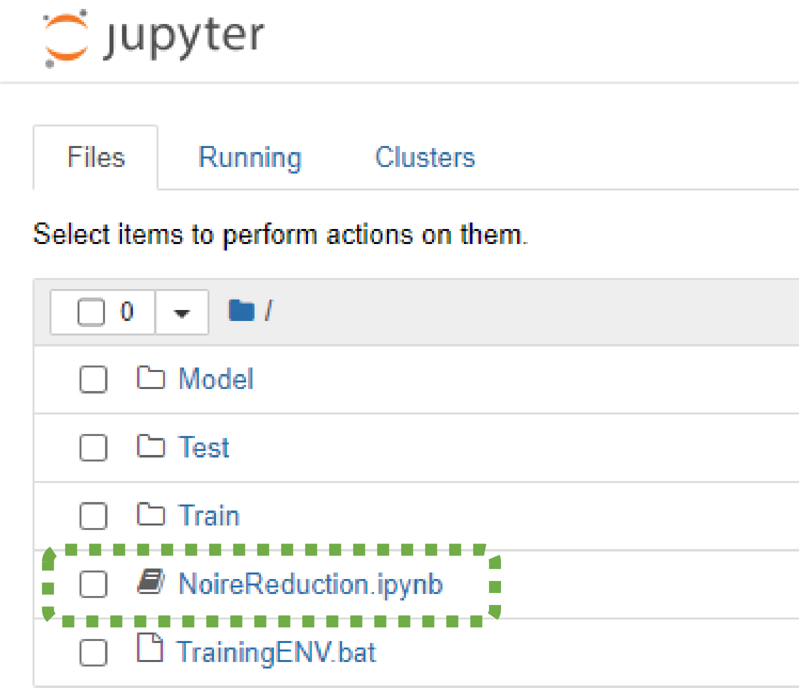
Model
Test
Train
NoireReduction.ipynb
TrainingENV.bat

notepad



```
set root=C:\Users\REPLACE_WITH_USER_LOCATION\AppData\Local\Continuum\anaconda3

call %root%\Scripts\activate.bat %root%

call conda activate TrainingENV

call jupyter notebook

pause
```

- Double click on TrainingENV.bat

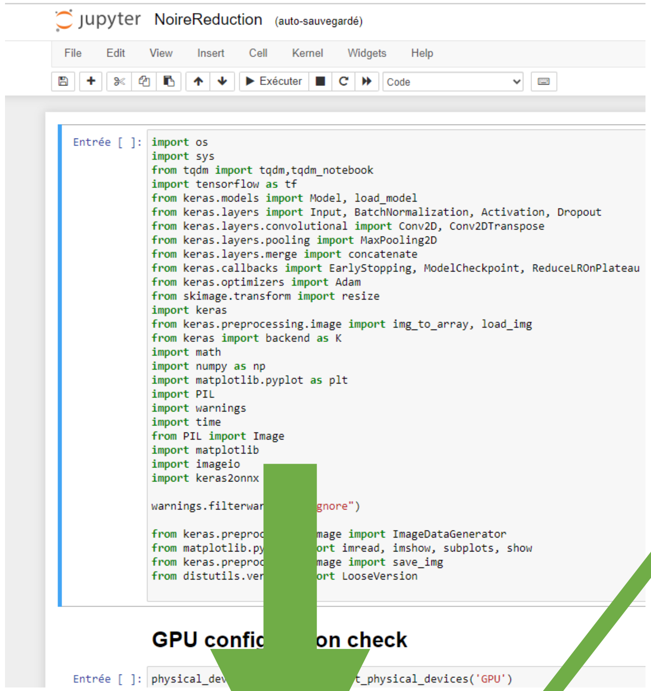


Select items to perform actions on them.

Model
Test
Train
NoireReduction.ipynb
TrainingENV.bat

Training / Jupyter notebook (2/2)

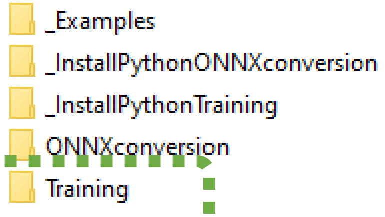
- Update Notebook



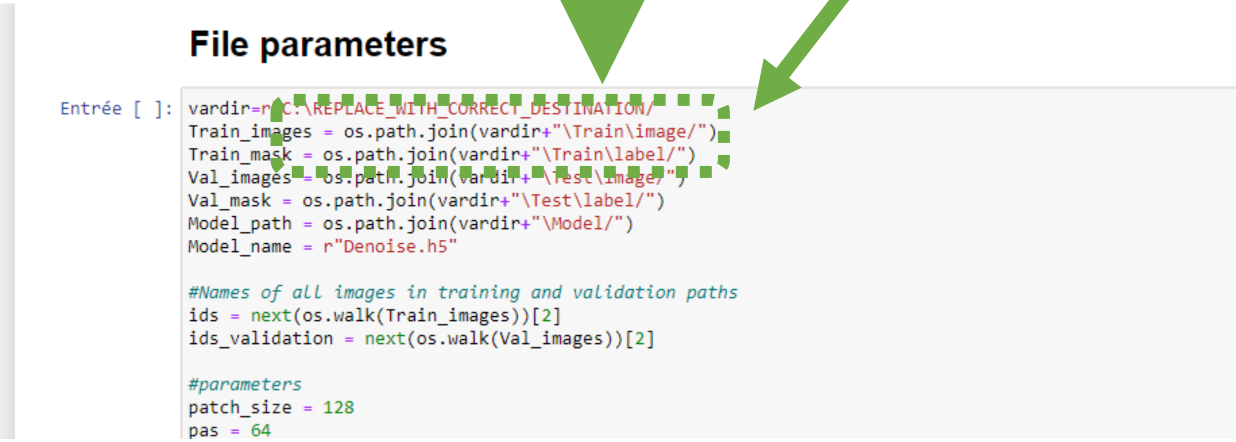
```
Entrée [ ]: import os
            import sys
            from tqdm import tqdm,tqdm_notebook
            import tensorflow as tf
            from keras.models import Model, load_model
            from keras.layers import Input, BatchNormalization, Activation, Dropout
            from keras.layers.convolutional import Conv2D, Conv2DTranspose
            from keras.layers.pooling import MaxPooling2D
            from keras.layers.merge import concatenate
            from keras.callbacks import EarlyStopping, ModelCheckpoint, ReduceLROnPlateau
            from keras.optimizers import Adam
            from skimage.transform import resize
            import keras
            from keras.preprocessing.image import img_to_array, load_img
            from keras import backend as K
            import math
            import numpy as np
            import matplotlib.pyplot as plt
            import PIL
            import warnings
            import time
            from PIL import Image
            import matplotlib
            import imageio
            import keras2onnx

            warnings.filterwar      gnore")

            from keras.preproc      mage import ImageDataGenerator
            from matplotlib.py      ort imread, imshow, subplots, show
            from keras.preproc      mage import save_img
            from distutils.ver      ort LooseVersion
```

GPU confic   on check

```
Entrée [ ]: physical_dev                t_physical_devices('GPU')
```



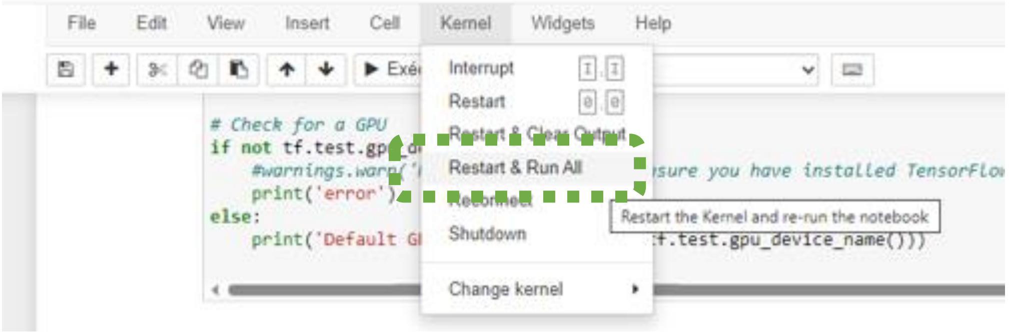
_Examples
_InstallPythonONNXconversion
_InstallPythonTraining
ONNXconversion
Training



File parameters

```
Entrée [ ]: vardir=r"C:\REPLACE_WITH_CORRECT_DESTINATION/"
            Train_images = os.path.join(vardir+"\Train\image/")
            Train_mask = os.path.join(vardir+"\Train\label/")
            Val_images = os.path.join(vardir+"\Test\image/")
            Val_mask = os.path.join(vardir+"\Test\label/")
            Model_path = os.path.join(vardir+"\Model/")
            Model_name = r"Denoise.h5"

            #Names of all images in training and validation paths
            ids = next(os.walk(Train_images))[2]
            ids_validation = next(os.walk(Val_images))[2]

            #parameters
            patch_size = 128
            pas = 64
```

- Execute Notebook



```
# Check for a GPU
if not tf.test.gp
    #warnings.warn('                                    sure you have installed TensorFlo
    print('error')
else:
    print('Default G                                   f.test.gpu_device_name()))
```

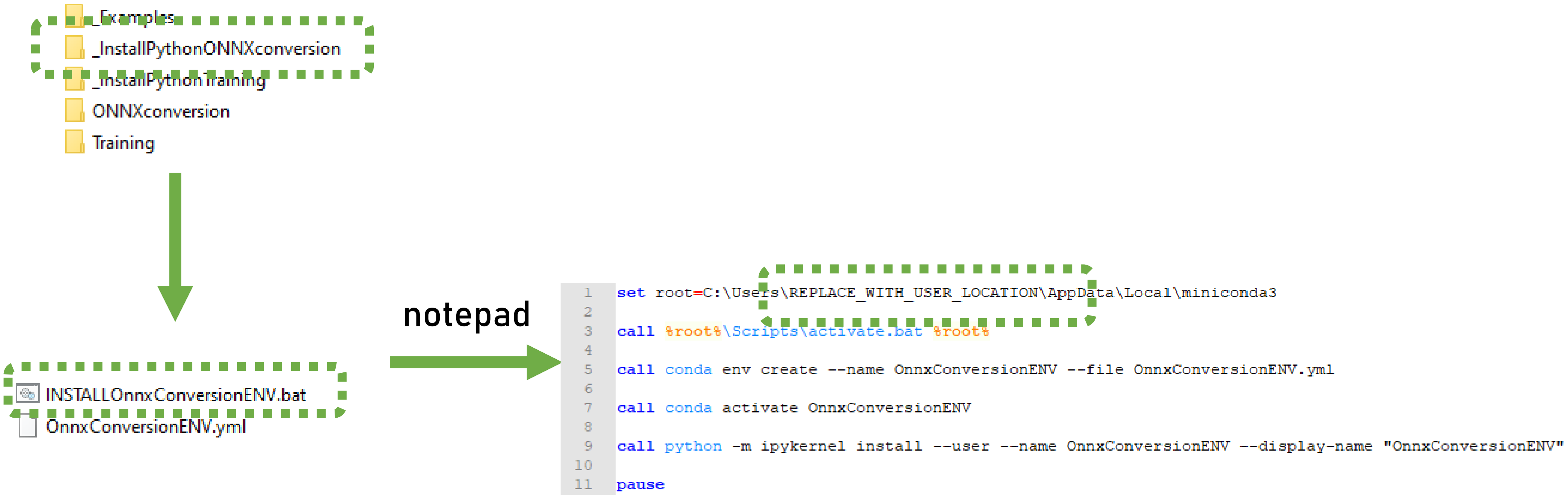
Restart the Kernel and re-run the notebook

Tutorial steps

- Training
  - Miniconda installation
  - Python environnement installation
  - Jupyter Notebook
- ONNX conversion
  - Python environnement installation
  - Jupyter Notebook
- Inference with plug im!
  - Plugin installation

ONNX conversion / Python environment installation

- Update _InstallPythonONNXconversion / _INSTALLOnnxConversionENV.bat

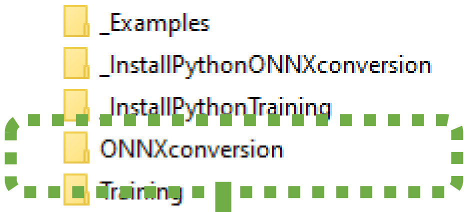


```
set root=C:\Users\REPLACE_WITH_USER_LOCATION\AppData\Local\miniconda3

call %root%\Scripts\activate.bat %root%

call conda env create --name OnnxConversionENV --file OnnxConversionENV.yml

call conda activate OnnxConversionENV

call python -m ipykernel install --user --name OnnxConversionENV --display-name "OnnxConversionENV"

pause
```

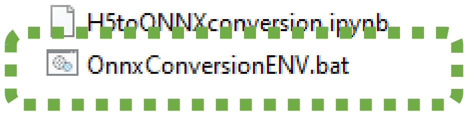
- Double click on INSTALLOnnxConversionENV.bat

ONNX conversion / Jupyter notebook (1/2)

- Update ONNXconversion / OnnxConversion.bat

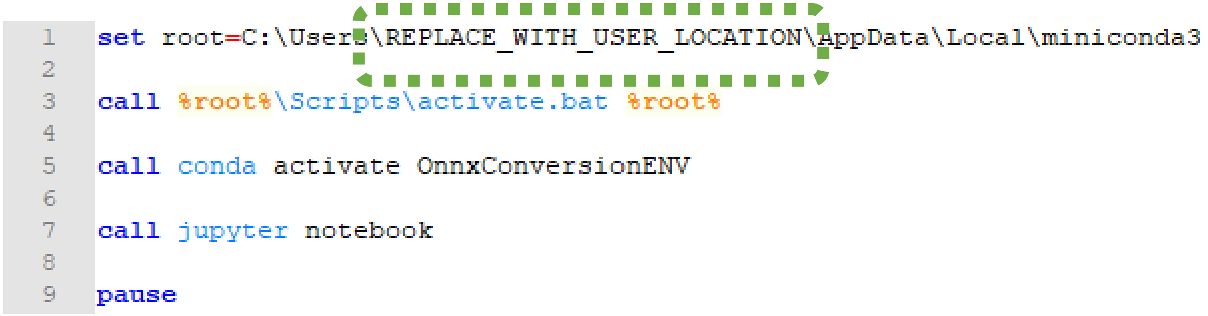


_Examples
_InstallPythonONNXconversion
_InstallPythonTraining
ONNXconversion
Training



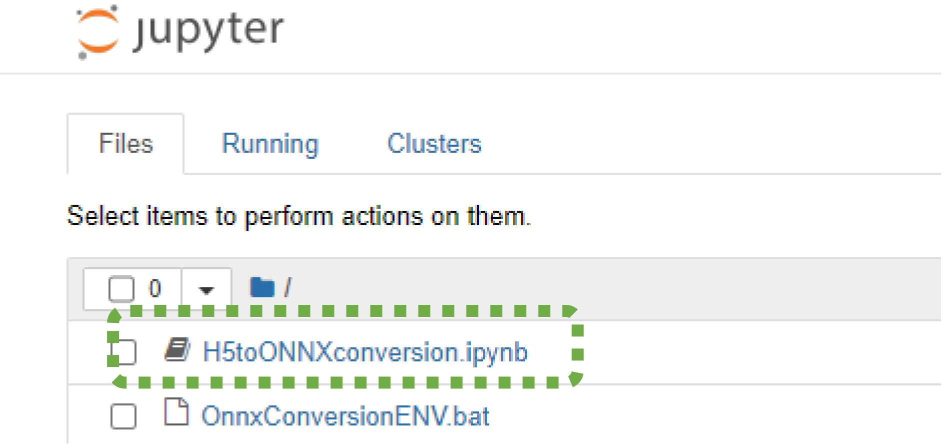
H5toONNXconversion.ipynb
OnnxConversionENV.bat

notepad



```
set root=C:\Users\REPLACE_WITH_USER_LOCATION\AppData\Local\miniconda3

call %root%\Scripts\activate.bat %root%

call conda activate OnnxConversionENV

call jupyter notebook

pause
```

- Double click on OnnxConvertionENV.bat



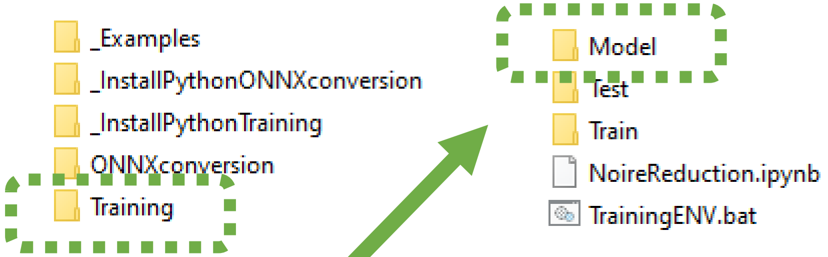
Select items to perform actions on them.

☐ 0 /

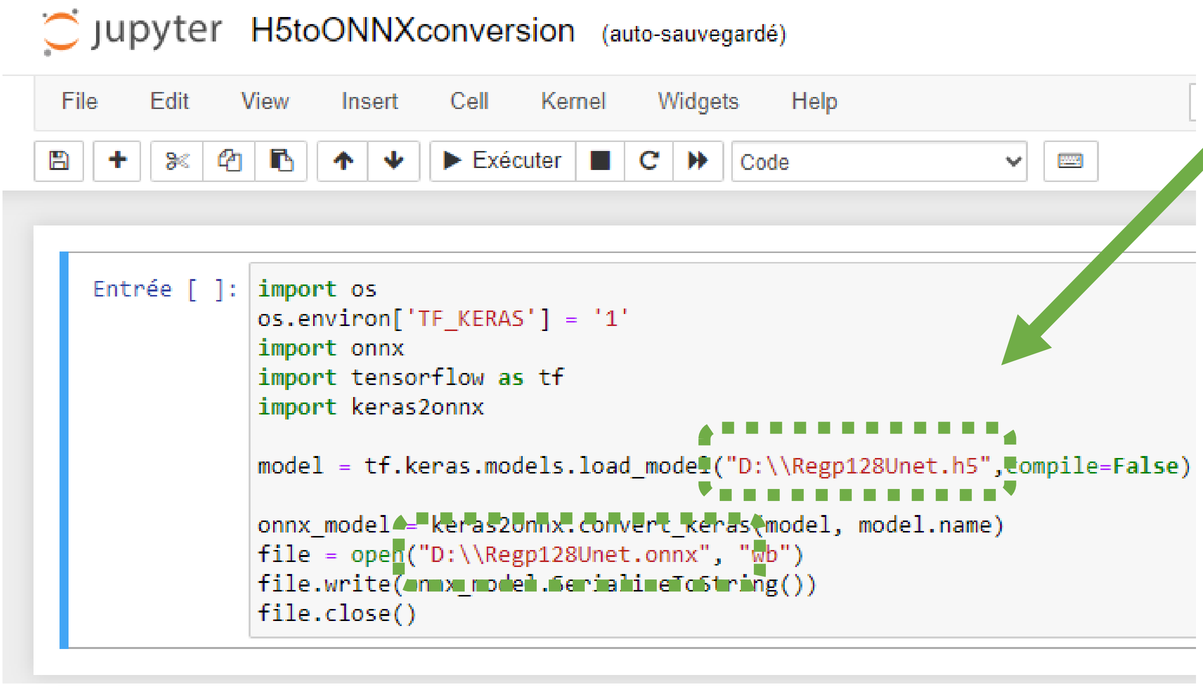
☐ H5toONNXconversion.ipynb

☐ OnnxConversionENV.bat

ONNX conversion / Jupyter notebook (2/2)

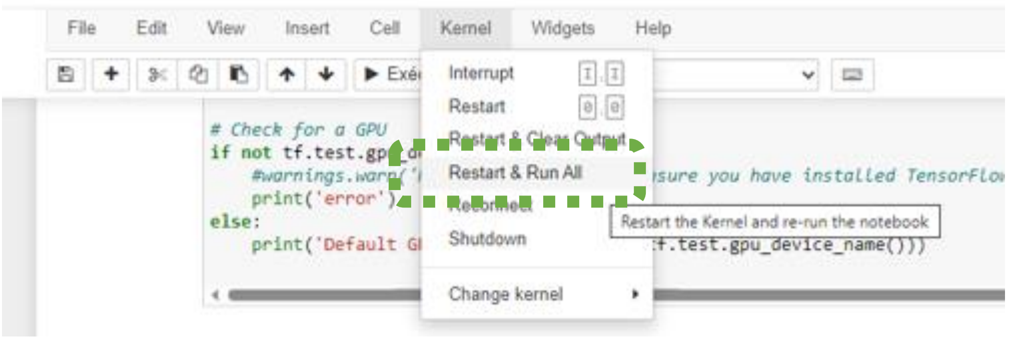


Update Notebook



```
import os
os.environ['TF_KERAS'] = '1'
import onnx
import tensorflow as tf
import keras2onnx

model = tf.keras.models.load_model("D:\\Regp128Unet.h5",compile=False)

onnx_model = keras2onnx.convert_keras(model, model.name)
file = open("D:\\Regp128Unet.onnx", "wb")
file.write(onnx_model.SerializeToString())
file.close()
```

Execute Notebook



```
# Check for a GPU
if not tf.test.gpu_d
    #warnings.warn('                          sure you have installed TensorFlow
    print('error')
else:
    print('Default G                          f.test.gpu_device_name()))
```

Tutorial steps

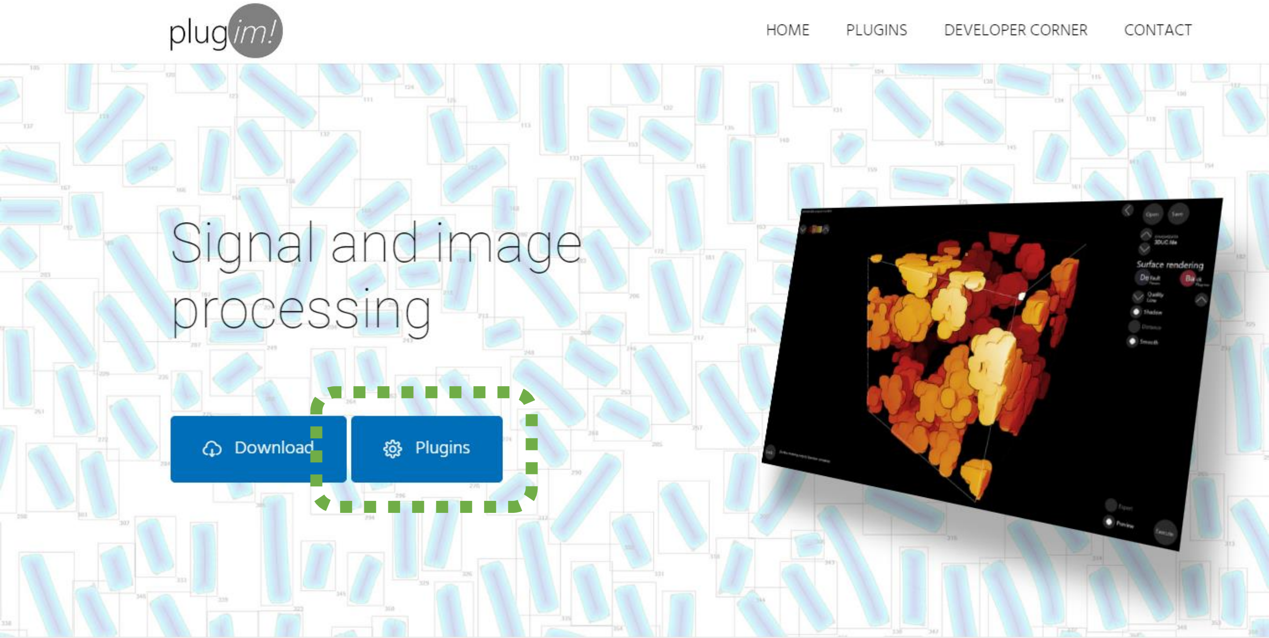
- Training
  - Miniconda installation
  - Python environnement installation
  - Jupyter Notebook

- ONNX conversion
  - Python environnement installation
  - Jupyter Notebook

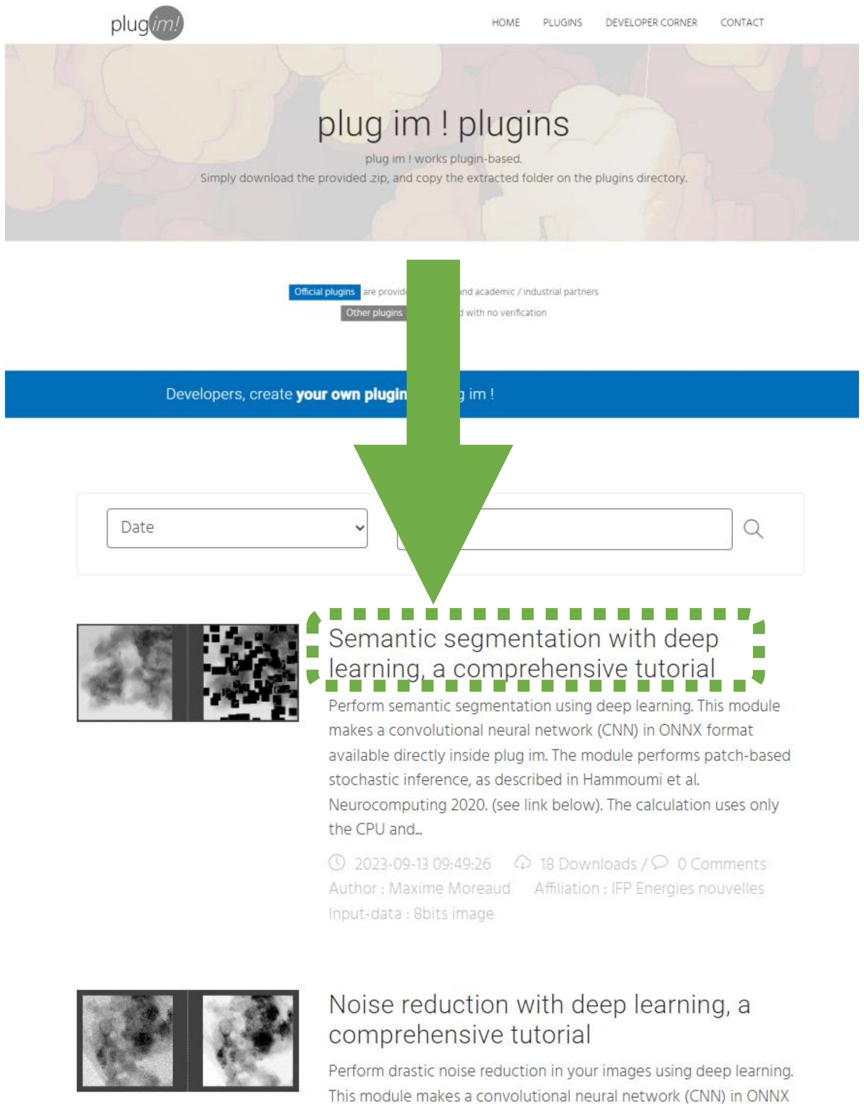
- Inference with plug im!
  - Plugin installation

Download “Segmentation” plugin

- www.plugim.fr



Signal and image processing



plug im ! plugins

plug im ! works plugin-based.
Simply download the provided .zip, and copy the extracted folder on the plugins directory.

Official plugins are provid nd academic / industrial partners
Other plugins d with no verification

Developers, create **your own plugin** g im !

Semantic segmentation with deep learning, a comprehensive tutorial

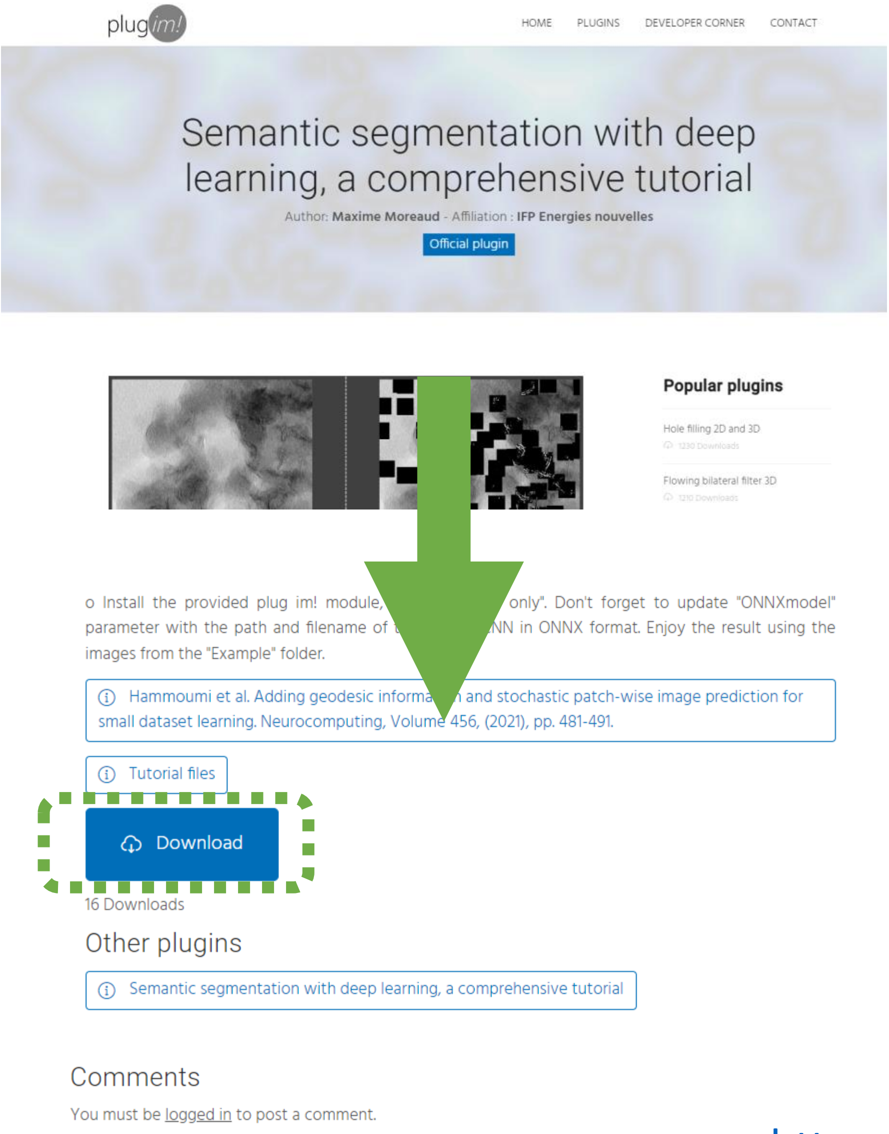
Perform semantic segmentation using deep learning. This module makes a convolutional neural network (CNN) in ONNX format available directly inside plug im. The module performs patch-based stochastic inference, as described in Hammoumi et al. Neurocomputing 2020. (see link below). The calculation uses only the CPU and...

2023-09-13 09:49:26 18 Downloads / 0 Comments
Author : Maxime Moreaud Affiliation : IFP Energies nouvelles
Input-data : 8bits image

Noise reduction with deep learning, a comprehensive tutorial

Perform drastic noise reduction in your images using deep learning. This module makes a convolutional neural network (CNN) in ONNX

Download "Segmentation" plugin



plug im!

HOME PLUGINS DEVELOPER CORNER CONTACT

Semantic segmentation with deep learning, a comprehensive tutorial

Author: **Maxime Moreaud** - Affiliation : **IFP Energies nouvelles**

Official plugin

Popular plugins

Hole filling 2D and 3D

1230 Downloads

Flowing bilateral filter 3D

1210 Downloads

o Install the provided plug im! module, only". Don't forget to update "ONNXmodel" parameter with the path and filename of t NN in ONNX format. Enjoy the result using the images from the "Example" folder.

ⓘ Hammoumi et al. Adding geodesic informa n and stochastic patch-wise image prediction for small dataset learning. Neurocomputing, Volume 456, (2021), pp. 481-491.

ⓘ Tutorial files

Download

16 Downloads

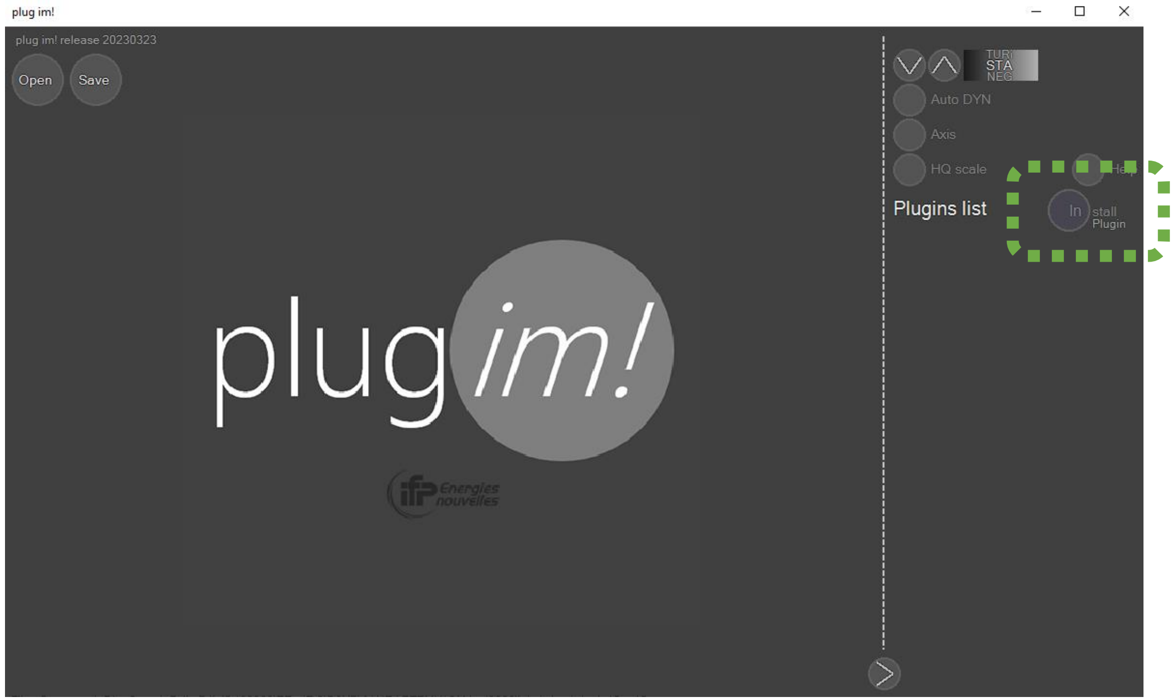
Other plugins

ⓘ Semantic segmentation with deep learning, a comprehensive tutorial

Comments

You must be logged in to post a comment.

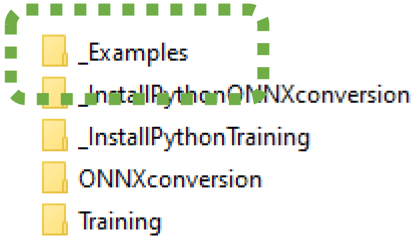
- Install downloaded zip file inside plug im!



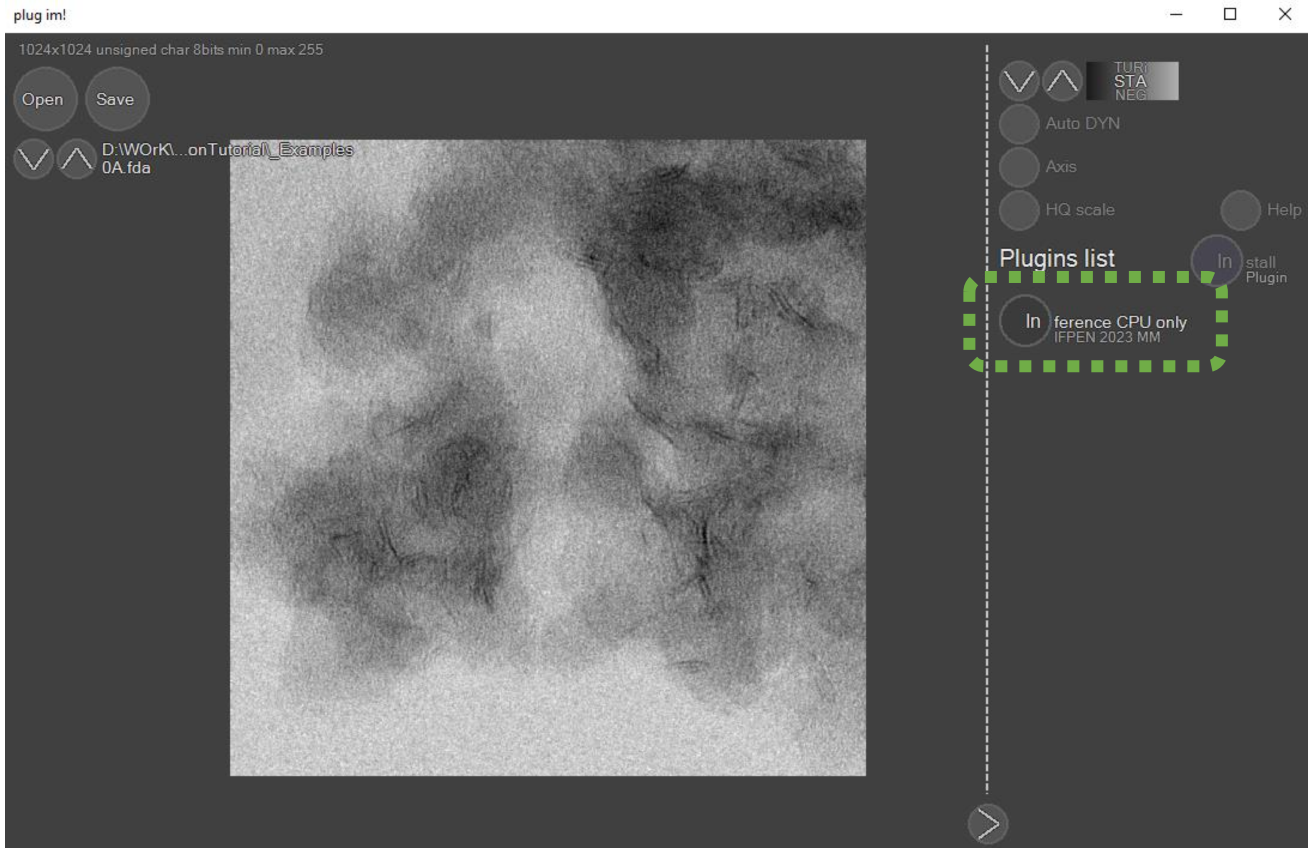
https://www.plugim.fr/plugin/120

Try semantic segmentation

- Load an image from _Examples
- Update ONNX model file parameter
- Enjoy!



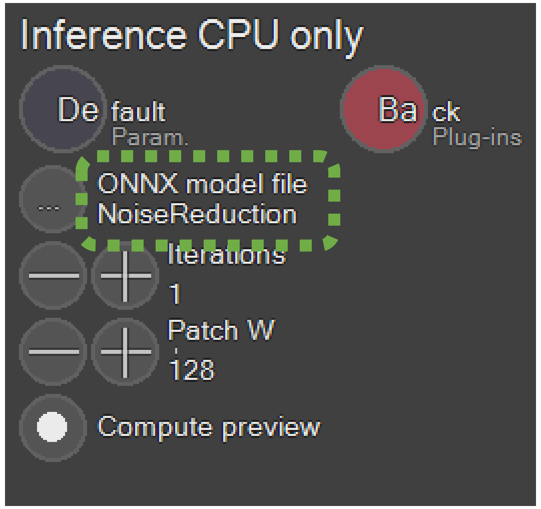
_Examples
_InstallPythonONNXconversion
_InstallPythonTraining
ONNXconversion
Training



plug im!

1024x1024 unsigned char 8bits min 0 max 255

D:\WOrK\...onTutorial_Examples
0A.fda



Inference CPU only

ONNX model file
NoiseReduction

Iterations
1

Patch W
128