



Getting started with napari

# DEEP NAPARI

Herearii METUAREA, Engineer, Université d'Angers, France

David ROUSSEAU, Full professor, Université d'Angers, France

15th November 2023

Room PC4 - Belambra clubs & hôtels, BAS-MADRAGUE



**INRAe**





Expert

CO PRO S2 (Weighted Loss) Unet\_segmentation\_notebook.ipynb ☆

Fichier Modifier Affichage Insérer Exécution Outils Aide Dernière modification effectuée le 27 juillet

+ Code + Texte

Split train data into train and validation

```
[ ] from sklearn.model_selection import train_test_split
x_train, x_val, y_train, y_val = train_test_split(X_train, Y_train, test_size=0.2, random_state=42) #text_
[ ] print('Train data shape is:',x_train.shape)

print('validation data shape is:',x_val.shape)

Train data shape is: (32, 256, 256, 3)
validation data shape is: (8, 256, 256, 3)

[ ] # Fiting the model
results = model.fit(x_train, y_train,
                     validation_data=(x_val,y_val),
                     batch_size=2, epochs=50,
                     callbacks=[early_stop,Model_check])

Epoch 11/50
16/16 [=====] - ETA: 0s - loss: 5.5026e-04 - dice_coefficient: 0.2432
Epoch 11: val_loss improved from 0.00069 to 0.00065, saving model to /content/gdrive/My Drive/data/best_mo
16/16 [=====] - 1s 44ms/step - loss: 5.5026e-04 - dice_coefficient: 0.2432 - val_
Epoch 12/50
```



Biologist



NIS-element



Expert

jupyterlab<sup>TNG</sup>

```
[ ] S2 (Weighted Loss) Unet_segmentation_notebook.ipynb ☆  
PRO Fichier Modifier Affichage Insérer Exécution Outils Aide Dernière modification effectuée le 27 juillet  
+ Code + Texte  
Split train data into train and validation  
  
[ ] from sklearn.model_selection import train_test_split  
x_train, x_val, y_train, y_val = train_test_split(X_train, Y_train, test_size=0.2, random_state=42) #text_...  
  
[ ] print('Train data shape is:',x_train.shape)  
  
print('validation data shape is:',x_val.shape)  
  
Train data shape is: (32, 256, 256, 3)  
validation data shape is: (8, 256, 256, 3)  
  
[ ] # Fiting the model  
results = model.fit(x_train, y_train,  
                    validation_data=(x_val,y_val),  
                    batch_size=2, epochs=50,  
                    callbacks=[early_stop,Model_check])  
  
Epoch 11/50  
16/16 [=====] - ETA: 0s - loss: 5.5026e-04 - dice_coefficient: 0.2432  
Epoch 11: val_loss improved from 0.00069 to 0.00065, saving model to /content/gdrive/My Drive/data/best_mo...  
16/16 [=====] - 1s 44ms/step - loss: 5.5026e-04 - dice_coefficient: 0.2432 - val_...  
Epoch 47/50
```

?      ?      ?      ?



**interface that  
satisfies  
everyone**



Biologist



NIS-element





Expert



PRO S2 (Weighted Loss) Unet\_segmentation\_notebook.ipynb ☆

Fichier Modifier Affichage Insérer Exécution Outils Aide Dernière modification effectuée le 27 juillet

+ Code + Texte

Split train data into train and validation

```
[ ] from sklearn.model_selection import train_test_split
x_train, x_val, y_train, y_val = train_test_split(X_train, Y_train, test_size=0.2, random_state=42) #text_
[ ] print('Train data shape is:',x_train.shape)
print('validation data shape is:',x_val.shape)

Train data shape is: (32, 256, 256, 3)
validation data shape is: (8, 256, 256, 3)

[ ] # Fiting the model
results = model.fit(x_train, y_train,
                     validation_data=(x_val,y_val),
                     batch_size=2, epochs=50,
                     callbacks=[early_stop,Model_check])

Epoch 11/50
16/16 [=====] - ETA: 0s - loss: 5.5026e-04 - dice_coefficient: 0.2432
Epoch 11: val_loss improved from 0.00069 to 0.00065, saving model to /content/gdrive/My Drive/data/best_mo
16/16 [=====] - 1s 44ms/step - loss: 5.5026e-04 - dice_coefficient: 0.2432 - val_
```



napari



Biologist

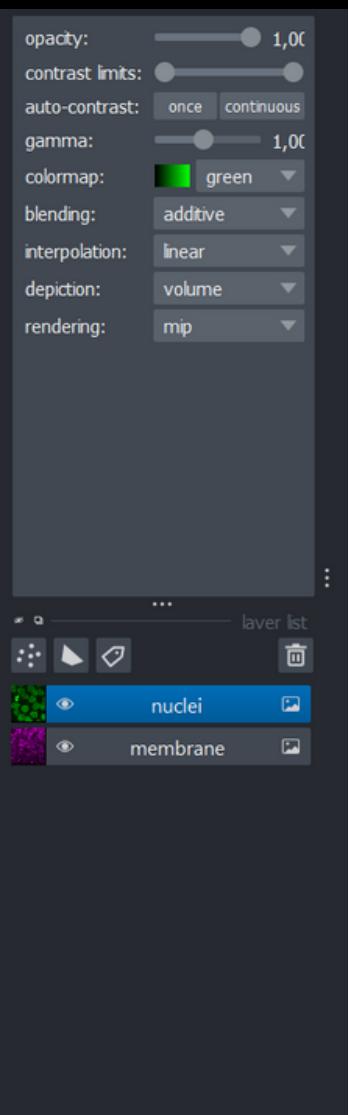


NIS-element





napari

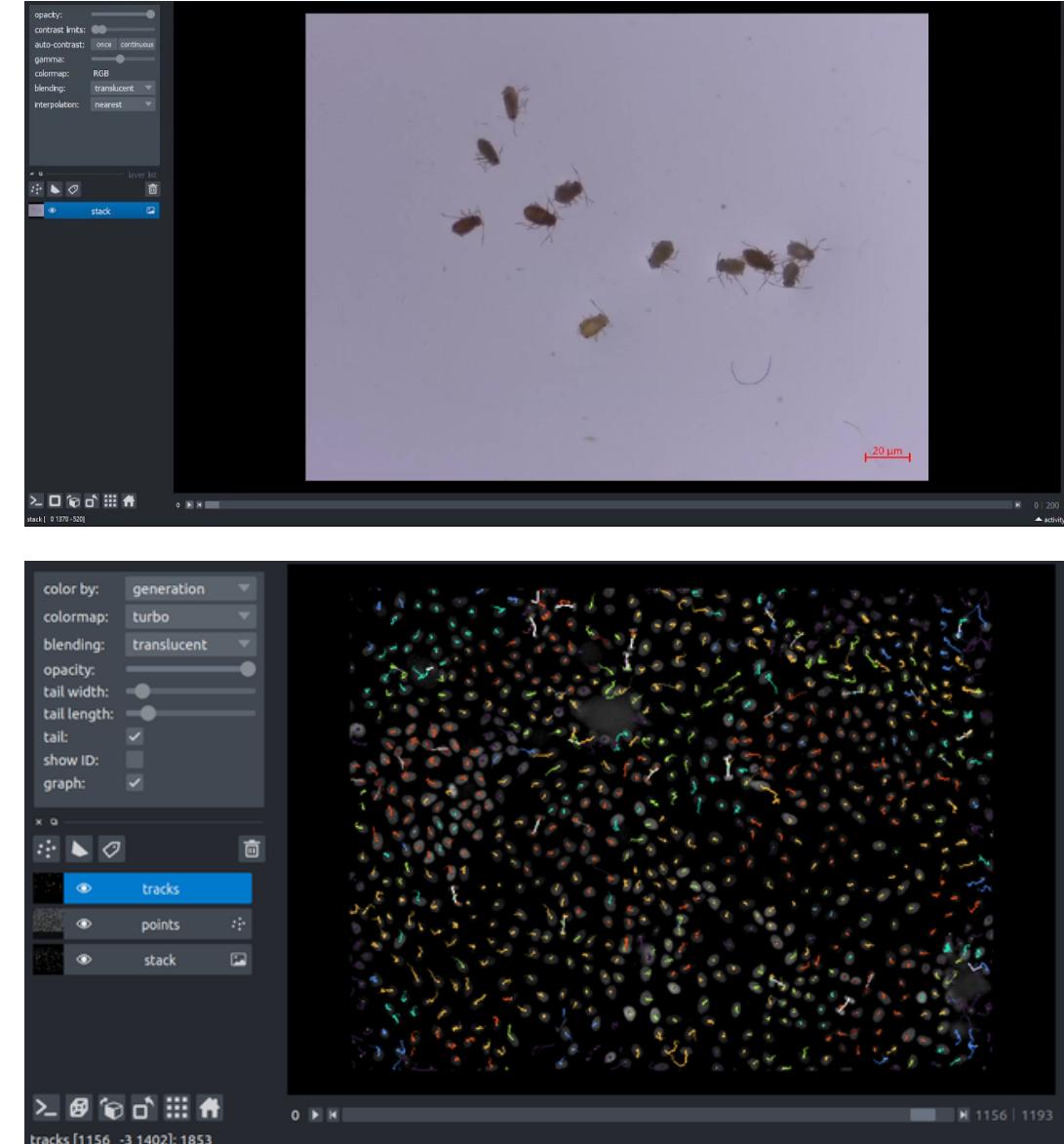
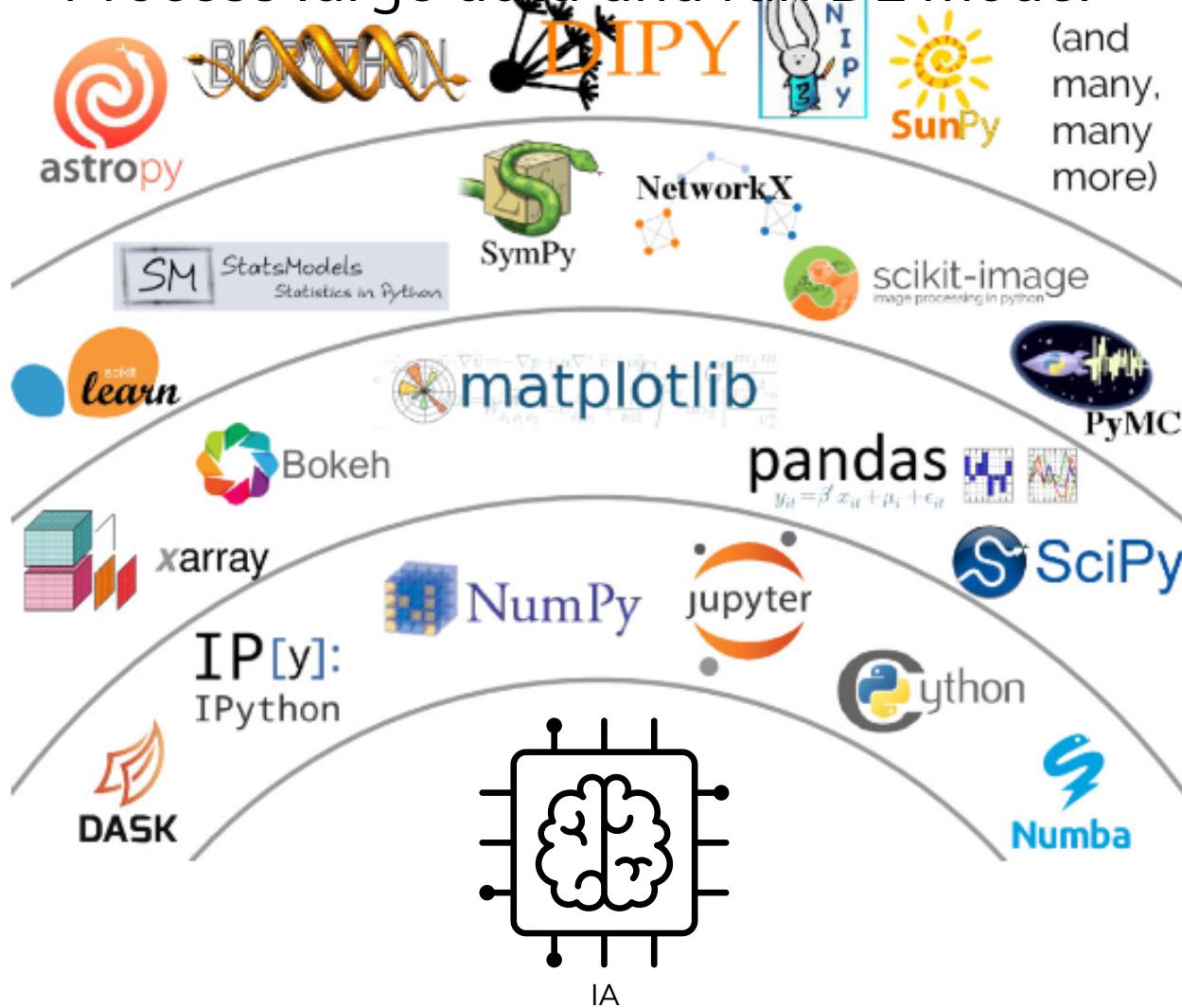


Help scientist to access Python's scientific ecosystem, with no prior coding experience

Multi-dimensional data viewer in Python  
open-source, community-developed

# Napari

Process large data and run DL model



# Deep learning in Napari

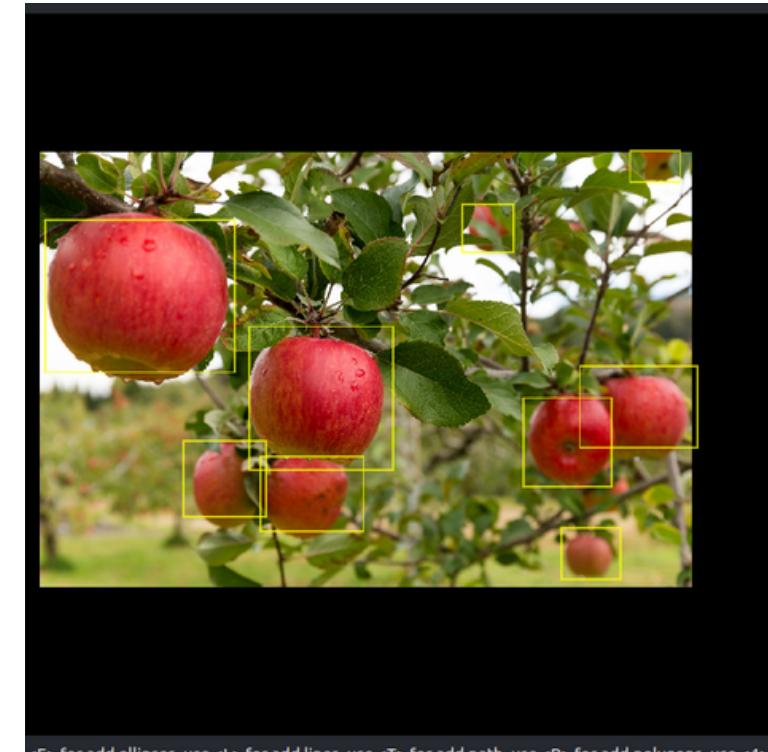
Apple flower detection



Plugin: napari-blossom

A screenshot of the Napari software interface showing the 'napari-blossom' plugin. On the left is a dark gray sidebar with a 'layer' dropdown menu set to 'IMG\_3730 (data)' and a 'Run' button below it. The main canvas area is black, indicating no active analysis or visualization.

Apple detection in orchard



Plugin: napari-apple

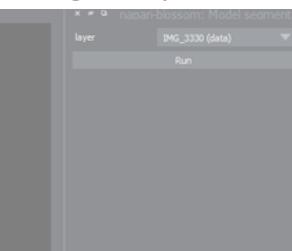
A screenshot of the Napari software interface showing the 'Image detection (napari-apple)' plugin. On the left is a dark gray sidebar with a 'Pick a file:' dropdown set to 'ubscription\_XL-1.jpg' and a 'Select file' button, followed by a 'string' input field containing 'e/g-laris89/Documents/darknet' and a 'Run' button. The main canvas area shows a photograph of an apple branch with several yellow bounding boxes highlighting detected apples.

# Deep learning in Napari

Apple flower detection



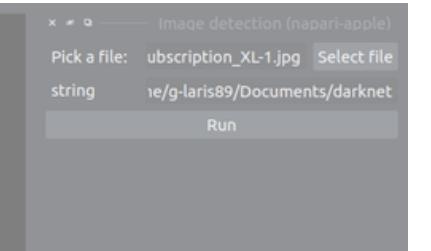
Plugin: napari-blossom



Apple detection in orchard



Plugin: napari-apple



And many more !

# Topic of this workshop

## Objective

Create a napari plugin from image processing code including DL model

## What we will do

Review a DL code

Create a napari plugin in local

Integrate DL model (tensorflow) in plugin

## What we will not do

Teach Python programming and the base of deep learning (requirements)

Deploy the plugin in napari-hub plateform (napari library plugin)

# Practical session

Let's suppose we have



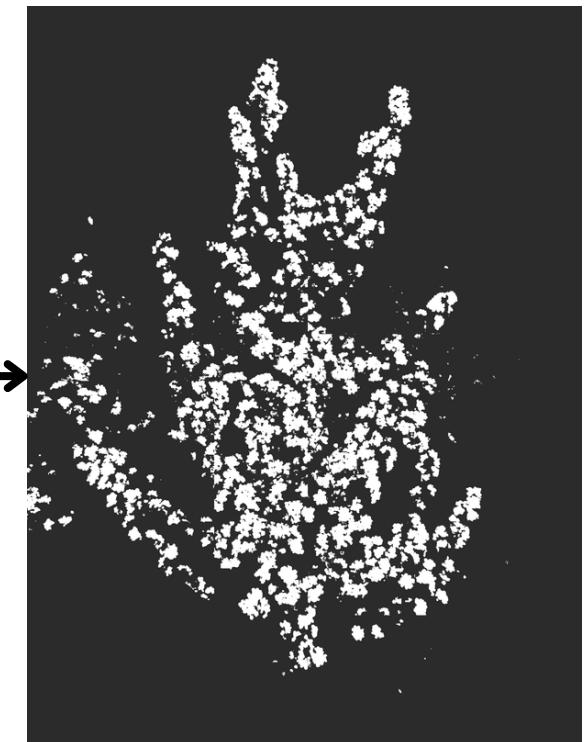
CO PRO S2 (Weighted Loss) Unet\_segmentation\_notebook.ipynb ☆

Fichier Modifier Affichage Insérer Exécution Outils Aide Dernière modification effectuée le 27 juillet

+ Code + Texte

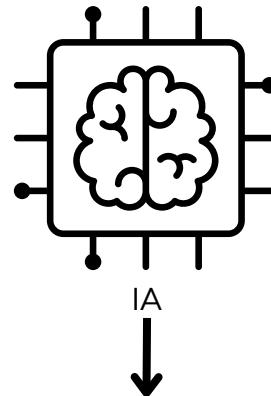
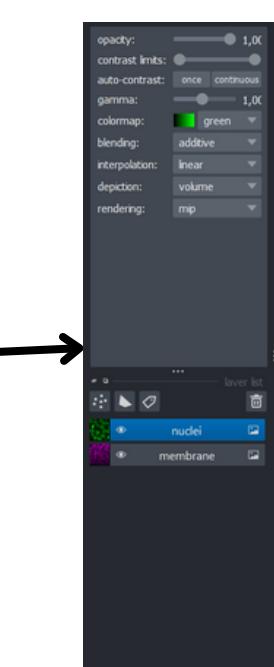
Split train data into train and validation

```
[ ] from sklearn.model_selection import train_test_split  
x_train, x_val, y_train, y_val = train_test_split(X_train, Y_train, test_size=0.2, random_state=42) #text_size=0.3  
  
[ ] print('Train data shape is:',x_train.shape)  
  
print('validation data shape is:',x_val.shape)  
  
Train data shape is: (32, 256, 256, 3)  
validation data shape is: (8, 256, 256, 3)  
  
[ ] # Fiting the model  
results = model.fit(x_train, y_train,  
                    validation_data=(x_val,y_val),  
                    batch_size=2, epochs=50,  
                    callbacks=[early_stop,Model_check])  
  
Epoch 11/50  
16/16 [=====] - ETA: 0s - loss: 5.5026e-04 - dice_coefficient: 0.2432  
Epoch 11: val_loss improved from 0.00069 to 0.00065, saving model to /content/gdrive/My Drive/data/best_model.h5  
16/16 [=====] - 1s 44ms/step - loss: 5.5026e-04 - dice_coefficient: 0.2432 - val_loss: 6.48  
Epoch 12/50
```

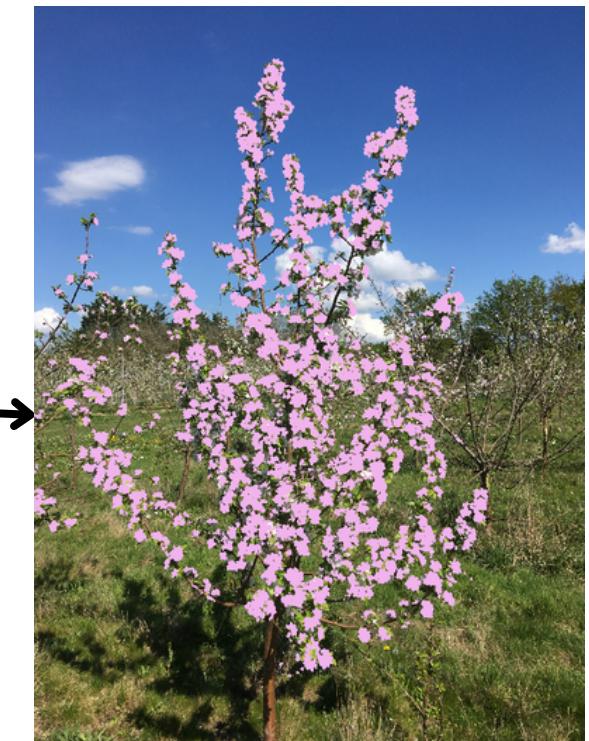
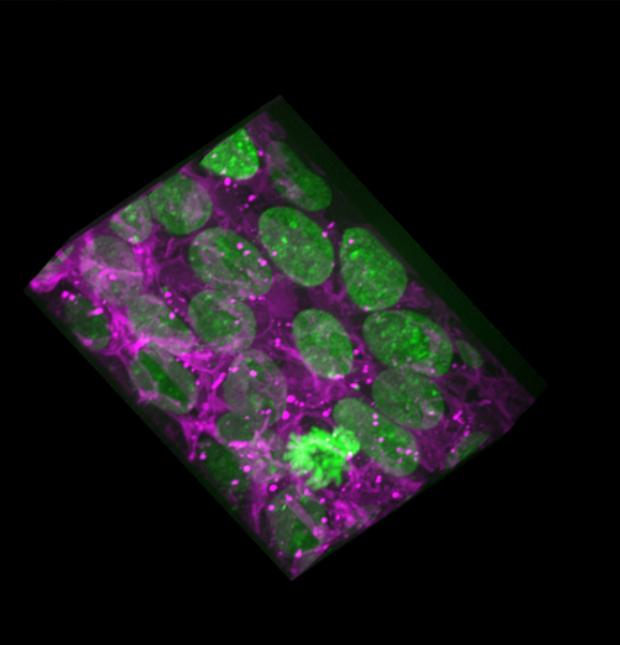


# Practical session

Deploy DL model in Napari plugin



IA

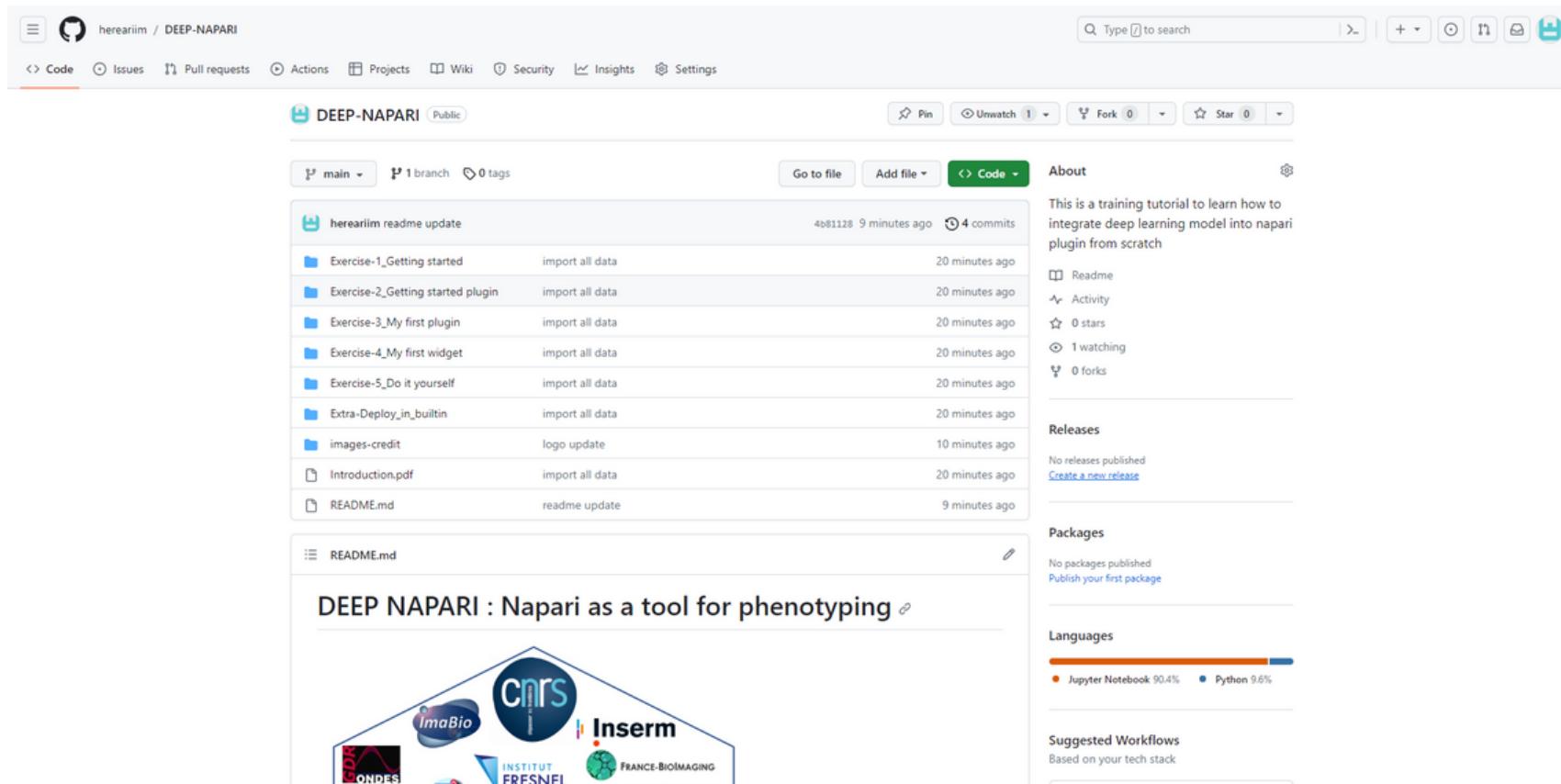


# Practical session



napari

Go to this page: <https://github.com/hereariim/DEEP-NAPARI>



The screenshot shows the GitHub repository page for "DEEP-NAPARI" owned by "hereariim". The repository is public and contains 4 commits. The main branch is "main" with 1 branch and 0 tags. The repository has an "About" section describing it as a training tutorial to learn how to integrate a deep learning model into napari from scratch. It includes sections for "Readme", "Activity", "Stars" (0), "Watching" (1), and "Forks" (0). There are no releases or packages published. The "Languages" section shows Jupyter Notebook at 90.4% and Python at 9.6%. Suggested workflows are based on the tech stack. The README.md file contains the following text:

```
DEEP NAPARI : Napari as a tool for phenotyping
```

Logos for various institutions are displayed at the bottom of the README:

- GDR FRONDES
- ImaBio
- CNRS
- INSTITUT FRESNEL
- Inserm
- FRANCE-BIOIMAGING

# To go further in this exercise

Tutorial on the presentation and design of a plugin

ImHorPhen Bio imaging research group  
217 abonnés S'ABONNER

ACCUEIL VIDÉOS PLAYLISTS CHAÎNES À PROPOS

SEARCH

The screenshot shows the YouTube channel page for 'ImHorPhen Bio imaging research group'. The channel has 217 subscribers and a red 'S'ABONNER' button. The navigation bar includes 'ACCUEIL', 'VIDÉOS' (which is underlined), 'PLAYLISTS', 'CHAÎNES', and 'À PROPOS'. A search icon is also present. Below the navigation, there are five main video thumbnails:

- Self-supervision applied to plants** (13:38) - Description: 'Self-supervised Learning' (Annotate User Data). Shows a flowchart of the process.
- Introduction to reinforcement learning part 1** (14:03) - Description: 'Deep Reinforcement Learning' (Learning in Dynamic Environments). Shows a robot arm interacting with a hand.
- Introduction to Self-Supervision** (10:36) - Description: 'What kind of transformations?' (Input Image -> Input Image + AD). Shows a diagram of the self-supervision process.
- 3D point Cloud annotation with Cloud Compare** (19:50) - Description: '3D point Cloud annotation with Cloud Compare'. Shows a 3D point cloud visualization.
- Intégrer un "deep model" dans plugin NAPARI** (14:53) - Description: 'Intégrer un "deep model" dans plugin NAPARI'. Shows a 3D plant structure with blue segmentation.

Below these are two rows of smaller video thumbnails:

- Installer NAPARI et faire son premier plugin** (13:14)
- Qu'est-ce qu'un plugin NAPARI ?** (2:48)
- Introduction au logiciel NAPARI** (5:14)
- PHENOME\_5\_AVRII\_2022** (16:06)
- ANF Deepscopie** (4:29)

Each thumbnail includes its title, duration, and a small preview image. The channel also features a logo with a stylized plant inside a house-like shape and the acronym 'HORP'.

# Conclusion



What we learnt ?

What we learnt ?

What we learnt ?

What we learnt ?

We learnt a method to use easily your DL model with GUI aka Napari

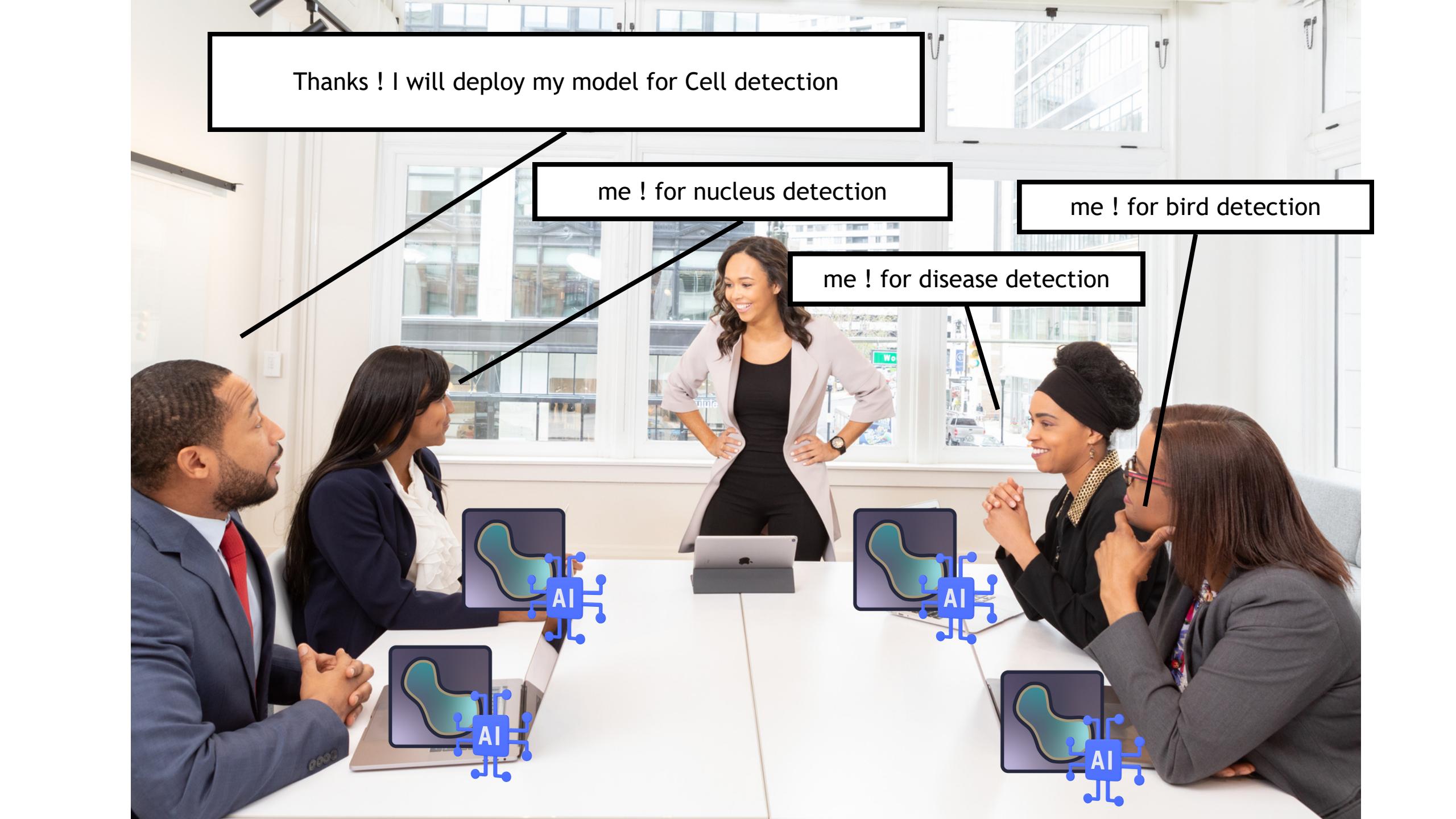


We learnt a method to use easily your DL model with GUI aka Napari



I hope this method we let you deploy your DL model to help experimental scientist



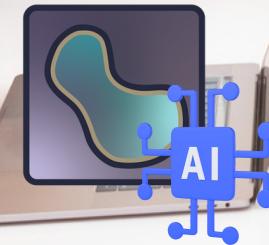
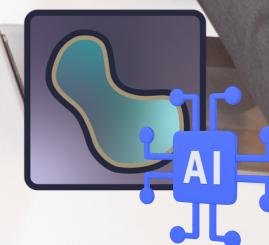
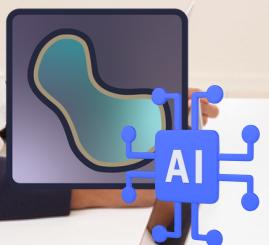


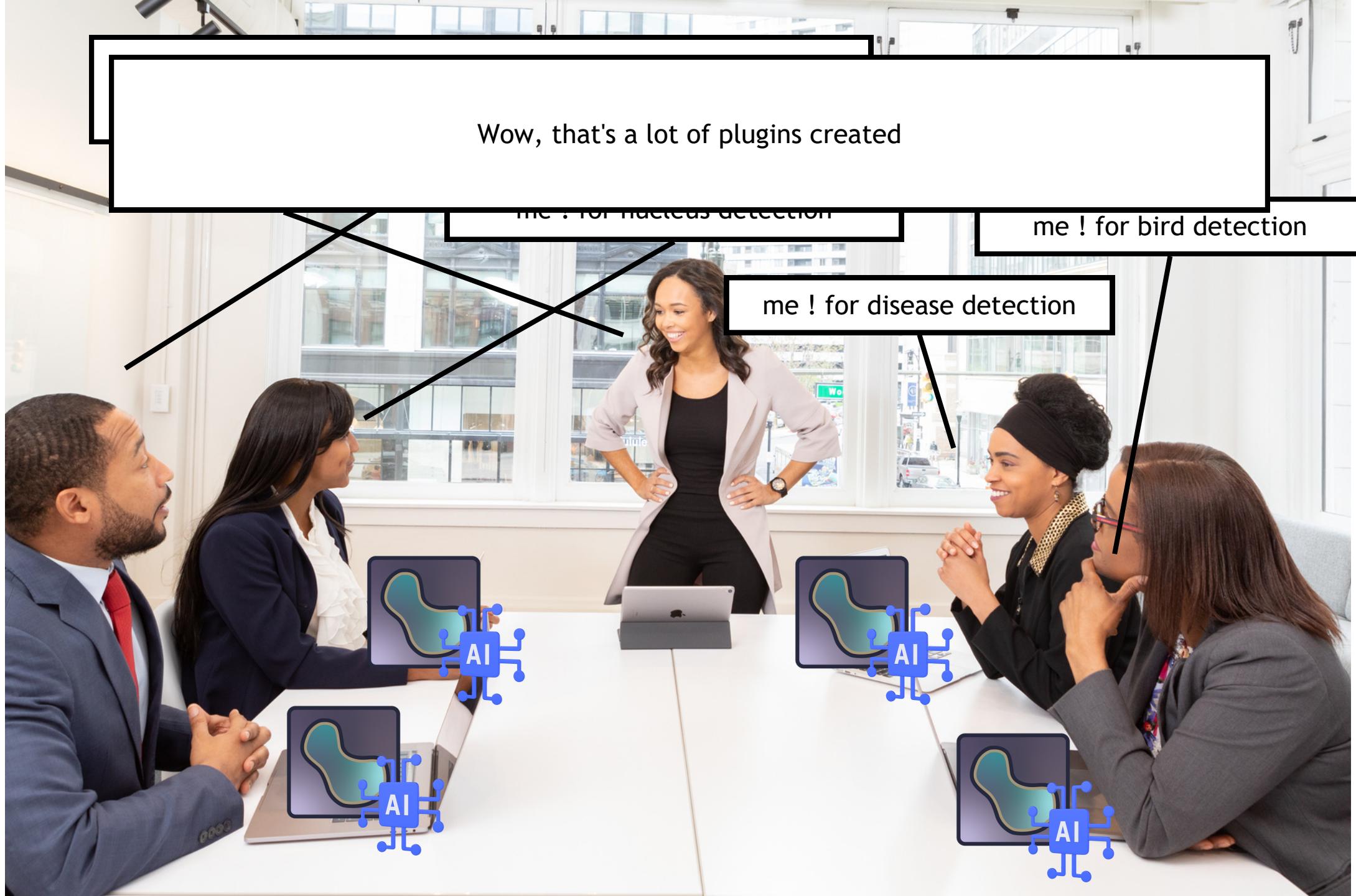
Thanks ! I will deploy my model for Cell detection

me ! for nucleus detection

me ! for bird detection

me ! for disease detection





Wow, that's a lot of plugins created

me : for nucleus detection

me ! for bird detection

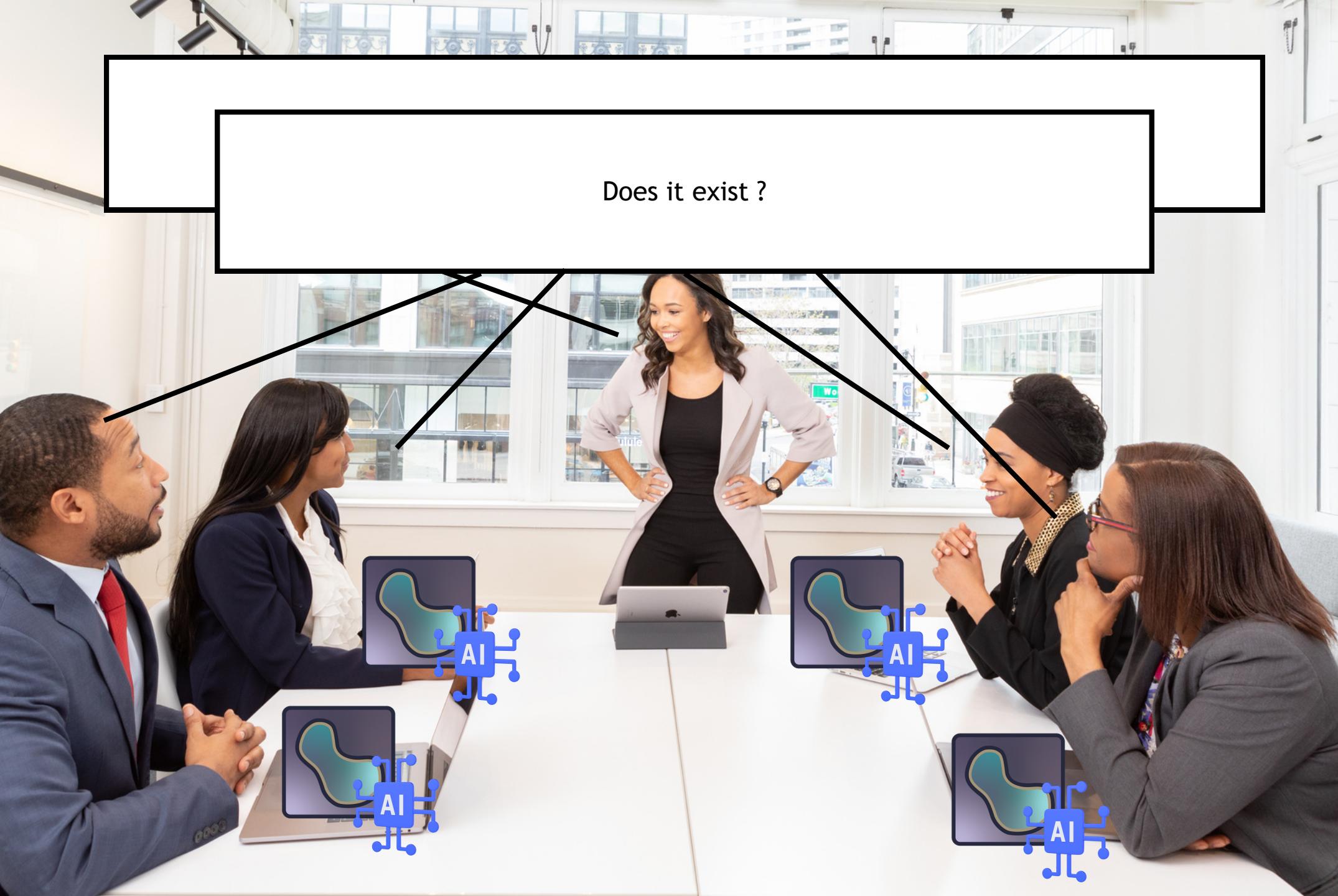
me ! for disease detection

Imagine, we have a single plugin to read all your DL model.



No need to create a set of plugins and your DL model remains confidential



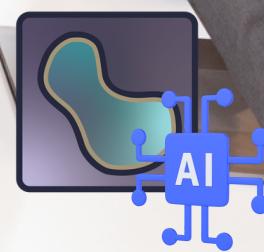
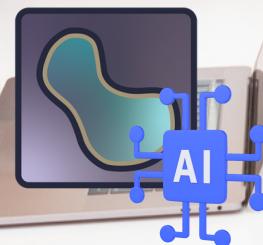
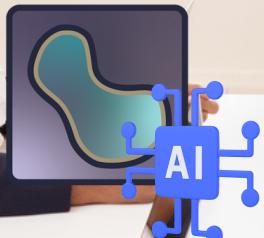


Does it exist ?



Does it exist ?

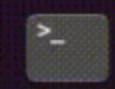
Sure !!!!!



Activities

napari

janv. 24 14:01



napari

# Manini

File View Window Plugins Help

layer controls



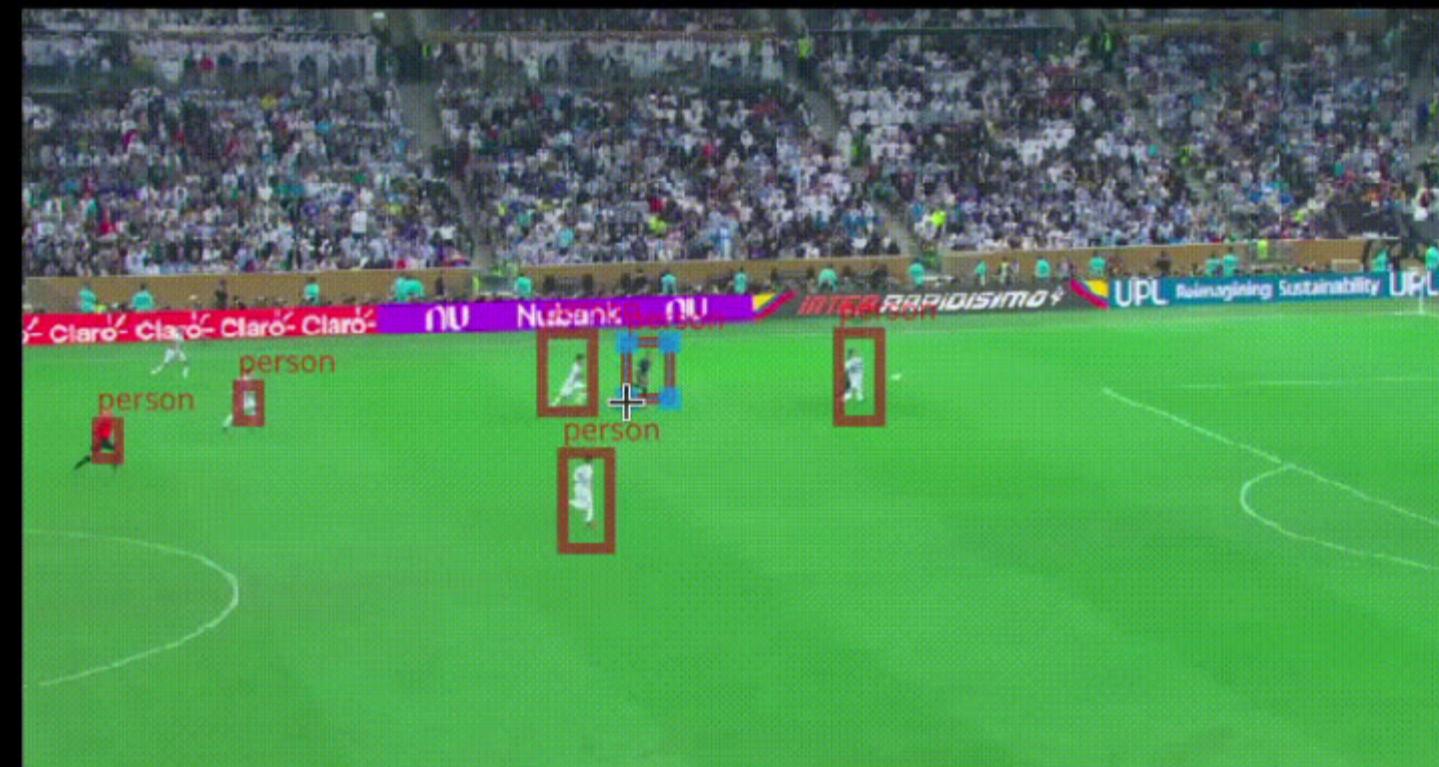
opacity: 0,70

edge width: 5

blending: translucent

face color: #00000000

edge color: brown

display text: 

layer list



bounding box



WC



bounding box [192 306]: 5, 4 use &lt;E&gt; for add ellipses, use &lt;L&gt; for add lines, use &lt;T&gt; for add path, use &lt;P&gt; for add polygons, use &lt;4&gt; for select vertices, use &lt;5&gt; for selec...

activity

Manini (manini)

Image segmentation

Image classification

Object Detection

Run

Save

images

commerce\_international\_3.j

Hôtel-de-la-Plage-Bordagair

RESTAURATION-R.jpg

road.jpg

wc.png

Class

person

bicycle

car

motorbike



Thanks you for your attention

