Setting up environment for AI

# Training environment

Steps to install TensorFlow in anaconda prompt to run the training in GPU.

1. Install anaconda from - https://www.anaconda.com/products/distribution [Anaconda Link](https://repo.anaconda.com/archive/Anaconda3-2022.10-Windows-x86_64.exe)
2. Select installation directory as D:\Anaconda3

Graphical user interface, text, application, email

Description automatically generated

1. From start menu open anaconda navigator after the installation.

A picture containing graphical user interface

Description automatically generated

1. In anaconda go to Environments in sidebar.

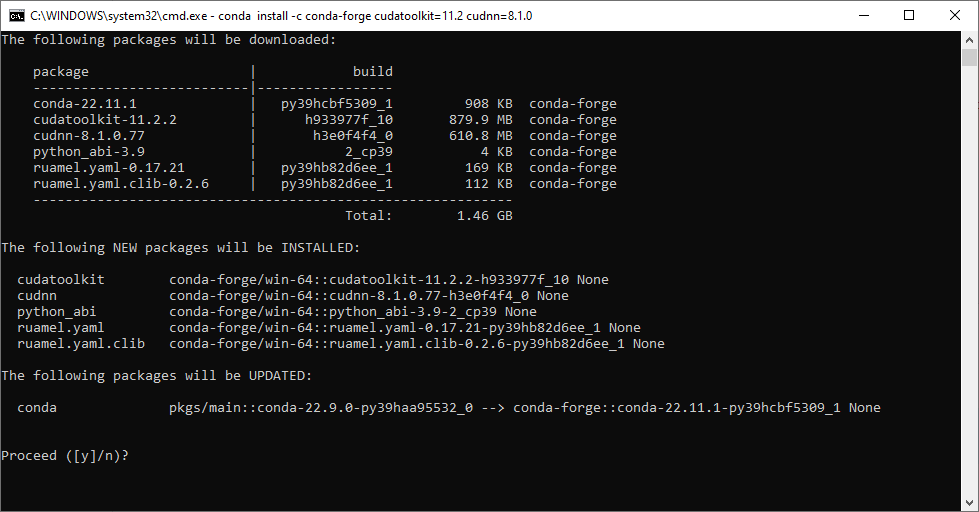
Graphical user interface, text, application, email

Description automatically generated

1. From base root, click and open terminal as shown in above screenshot.
2. Once command prompt opens up, run the below command, for more information visit <https://www.tensorflow.org/install/pip#windows-native>

conda install -c conda-forge cudatoolkit=11.2 cudnn=8.1.0

1. Enter y, when cmd prompts



1. Install tensorflow using pip3 in same prompt

pip3 install "tensorflow<2.11"

1. To test whether tensorflow detected GPU. Run this command.

python -c "import tensorflow as tf; print(tf.config.list\_physical\_devices('GPU'))"



# Labelling and data preparation environment

This section describes the procedure to setup labelling tool.

1. Open anaconda prompt as mentioned in previous section. Run below command to install labeling tool.

pip3 install django-labeller

for more information visit [GitHub - Britefury/django-labeller: An image labelling tool for creating segmentation data sets, for Django and Flask.](https://github.com/Britefury/django-labeller)

1. To run the web-based labeling tool. Use command

python -m image\_labelling\_tool.flask\_labeller --images\_dir=<images\_directory

Example:

python -m image\_labelling\_tool.flask\_labeller --images\_dir=D:/data/region\_of\_interest

After running the command, verify flask app is runnin. Visit <http://localhost:5000/>

You should also see Loaded ‘N’ images in command prompt.

Text

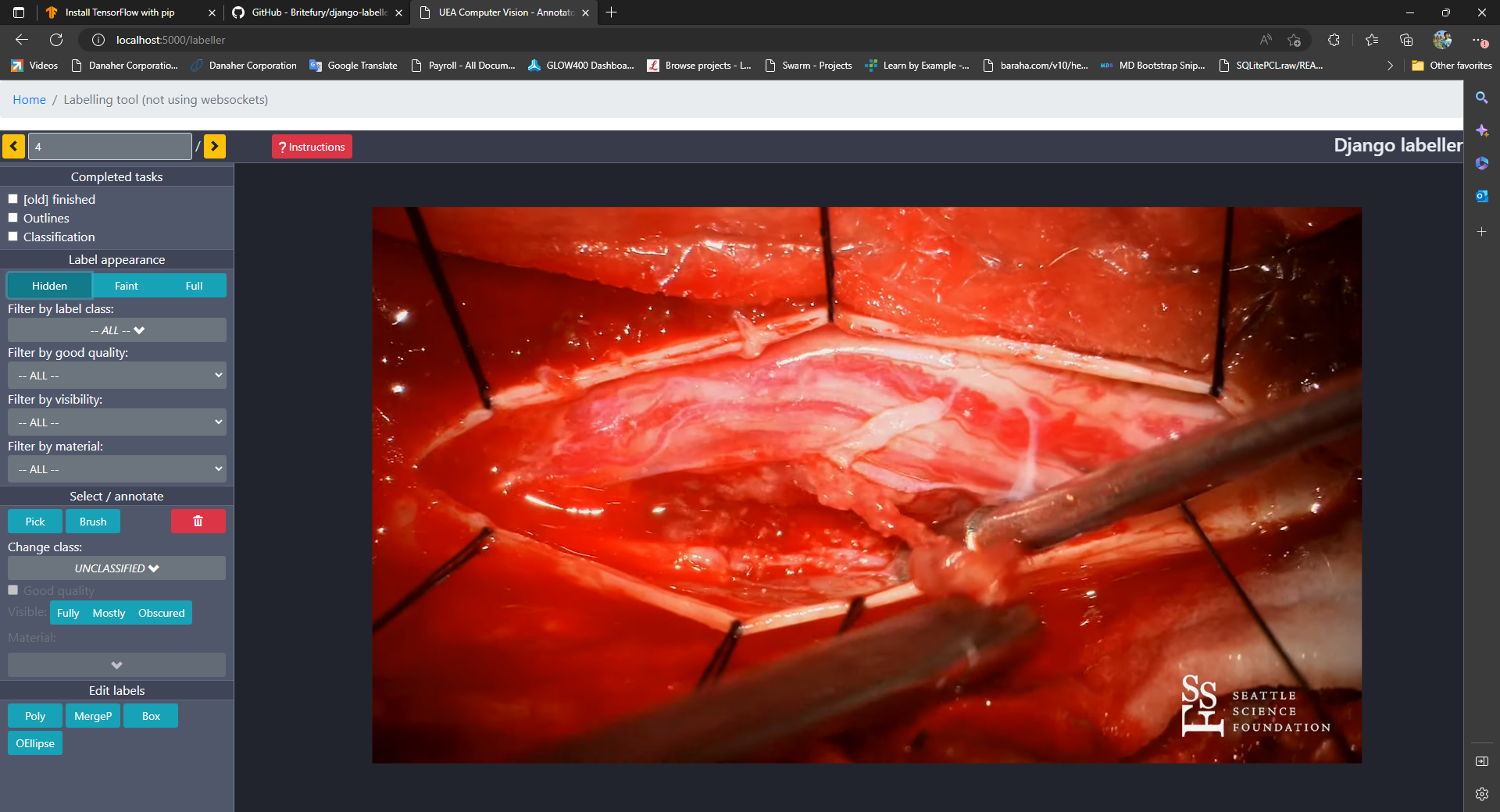
Description automatically generated

Web app would appear as shown below.

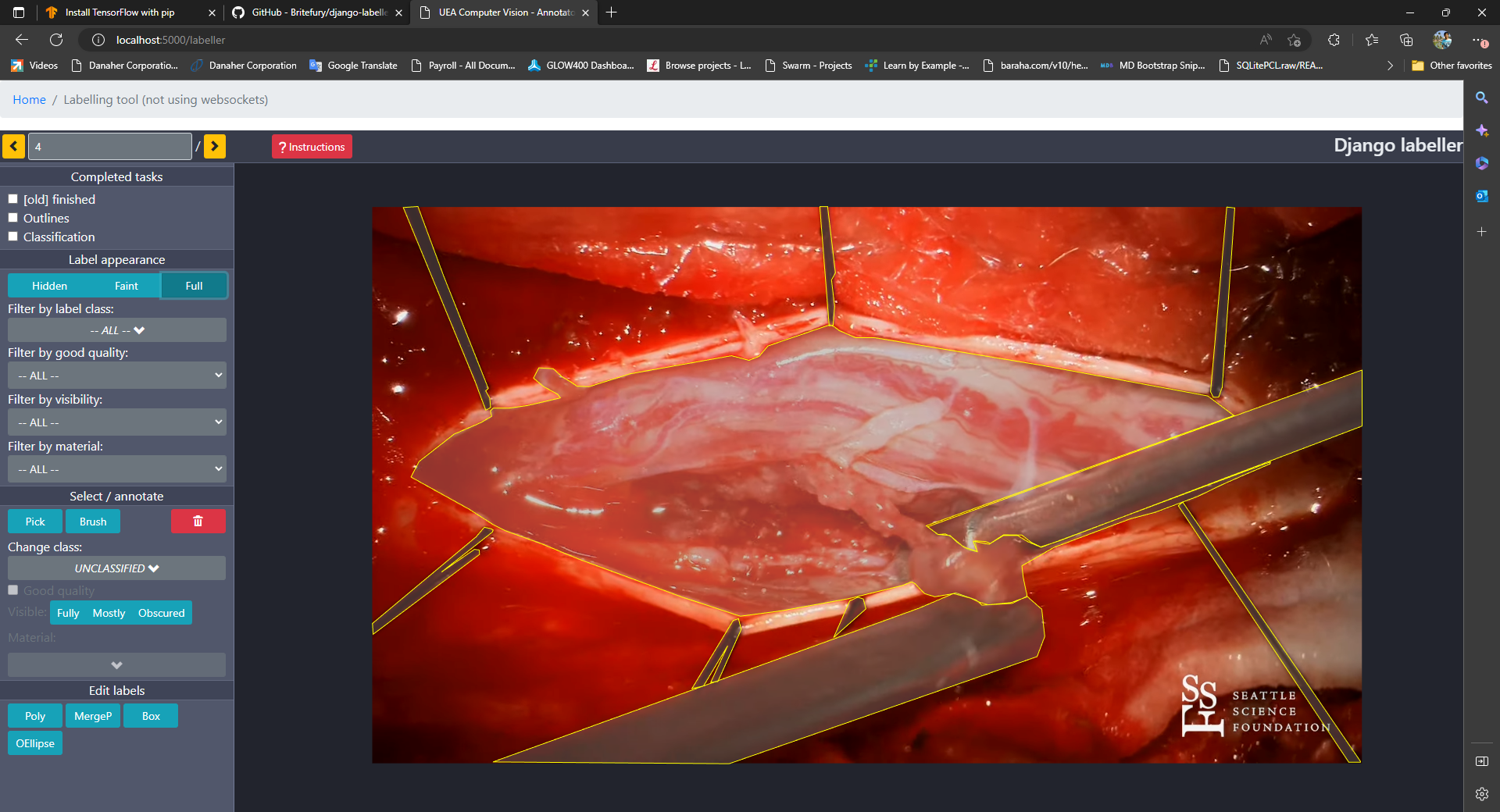
Graphical user interface, text, application

Description automatically generated

Go to schema editor and add some classes that you want to segment. This is used for generating the ground truth data.

One pressing labelling tool button, you would see 

After annotation this is how annotated data would look like.



Annotate the ground truth using polygon. You would see json files being created in the same folder as the image directory path you have set as a parameter to labeling tool.

# Install additional utilities

1. Opencv, image augmenttion for image manipulation

pip3 install opencv-python

pip3 install imgaug

1. Jupyterlab set working directory

Run command

jupyter notebook --generate-config

edit jupyter\_notebook\_config.py after running the command, this command returns the location of the file. Uncomment below line & change the directory of your wish.

# c.NotebookApp.notebook\_dir = ''

Text

Description automatically generated