publicityWebApp tutorial

PuBliCiTy ships with a web app for 2D and 3D image visualization and annotation. It can be started by running

```
python publicityWebApp.py
```

on Terminal. Be mindful of upper/lower cases, since there's also a file named PuBliCiTyWebApp.py, which should not be called directly.

Before running it, however, we have to setup a *Server* folder. In the same directory as the source code, create the following subfolder structure:

```
Server
      User
            Data
            MachineLearning
                  Annotations
                  Models
            Scratch
```

The web app can only read .tif images of 2 to 4 dimensions (i.e. 2D and 3D images with 1 or more channels). To access the images, they should be placed inside the Data subfolder. Eventual annotations are saved by the app inside the Annotations subfolder. The app can also train basic machine learning models, which are saved in Models – however we do not recommend training machine learning models using the app.

1. Startup

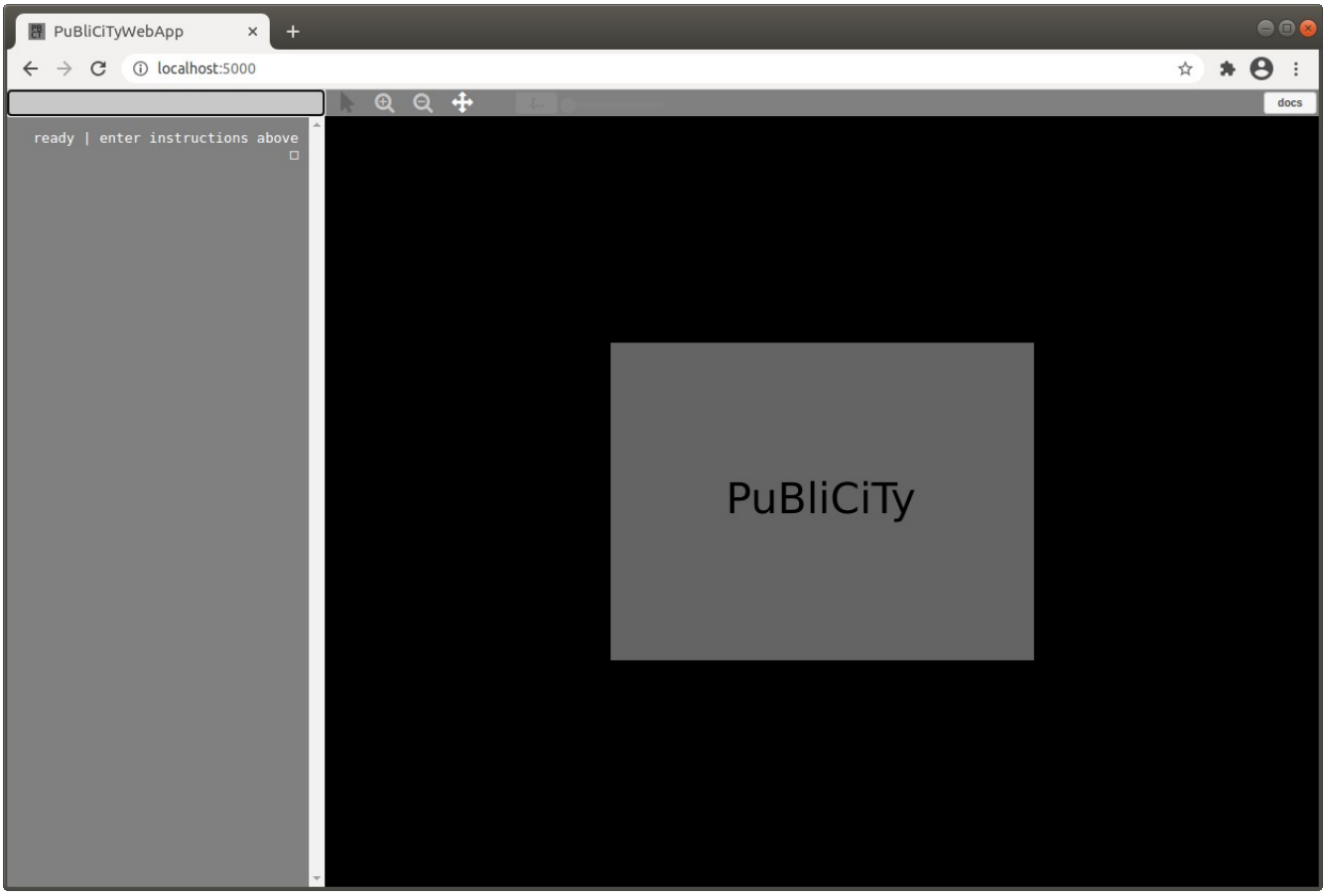
After running

```
python publicityWebApp.py
```

on Terminal, open a browser tab and navigate to

```
localhost:5000
```

The following is a screenshot of the app's 'home' screen, running on the Chrome browser:



The app's primary interaction mode is by entering text commands in the 'prompt box' at the top left of the interface. For a list of such commands, and other instructions, click on the 'docs' icon at the top right of the interface.

2. Image Visualization

For this tutorial we will use a 2D image form the t01_pixelclassifier tutorial, renamed as Sample2D.tif, and a 3D image from the t02_voxelclassifier tutorial, renamed as Sample3D.tif. Please refer to those tutorials for how to generate them. Remember to place them under the Server/User/Data subfolder.

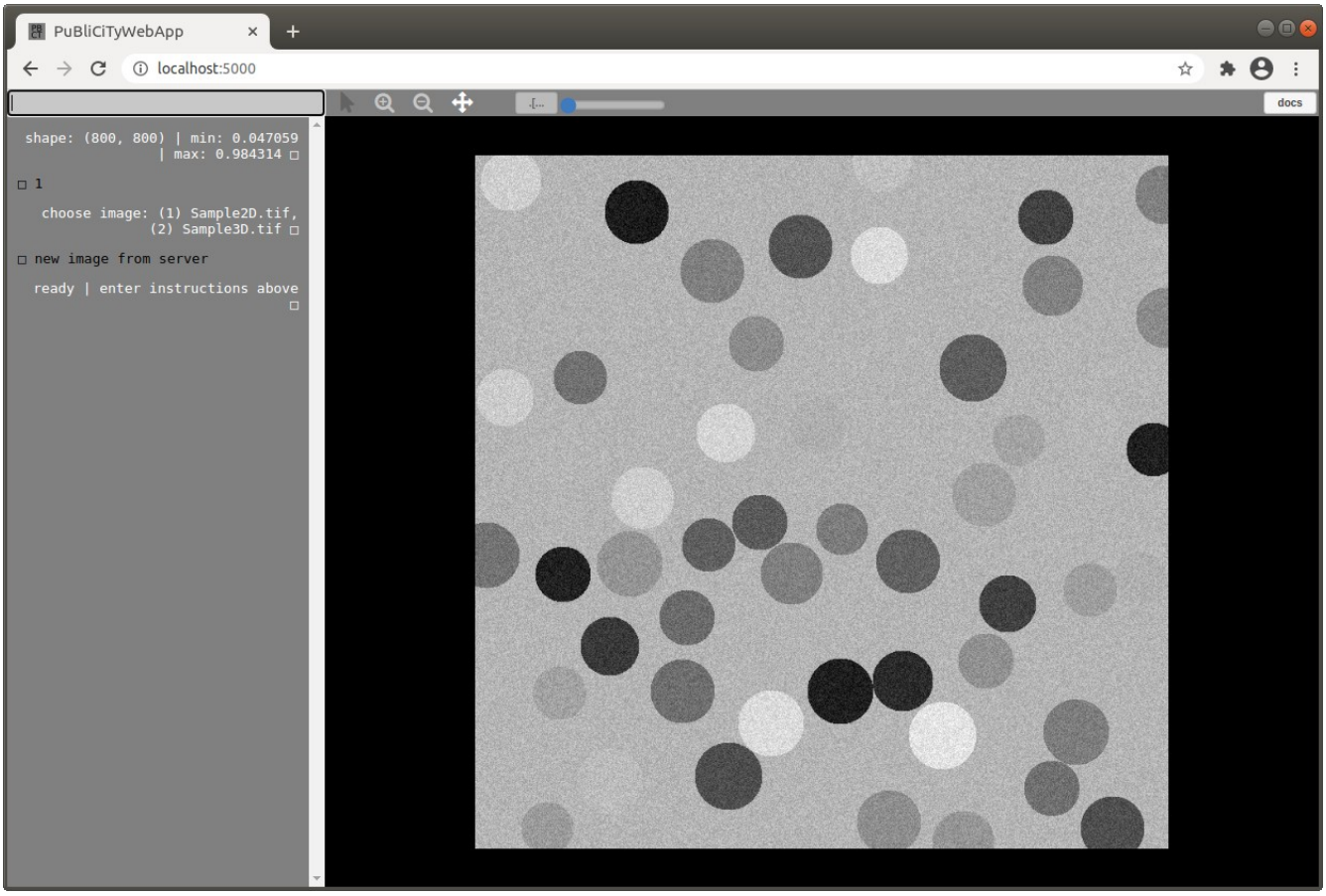
In the prompt box, type

```
new image from server
```

The app will respond with

```
choose image: (1) Sample2D.tif, (2) Sample3D.tif
```

We then type the index of the image we want to open – which is 1 for now.



Above the image area there are some basic image manipulation icons for zoom, pan, drawing, and contrast adjustment. Zoom, draw, and pan icons alter the ‘state’ of the mouse actions accordingly, and have to be ‘activated’ (clicked on) to be used. The draw icon is only enabled in certain circumstances (e.g. when the annotation tool is active).

3. 2D Annotations

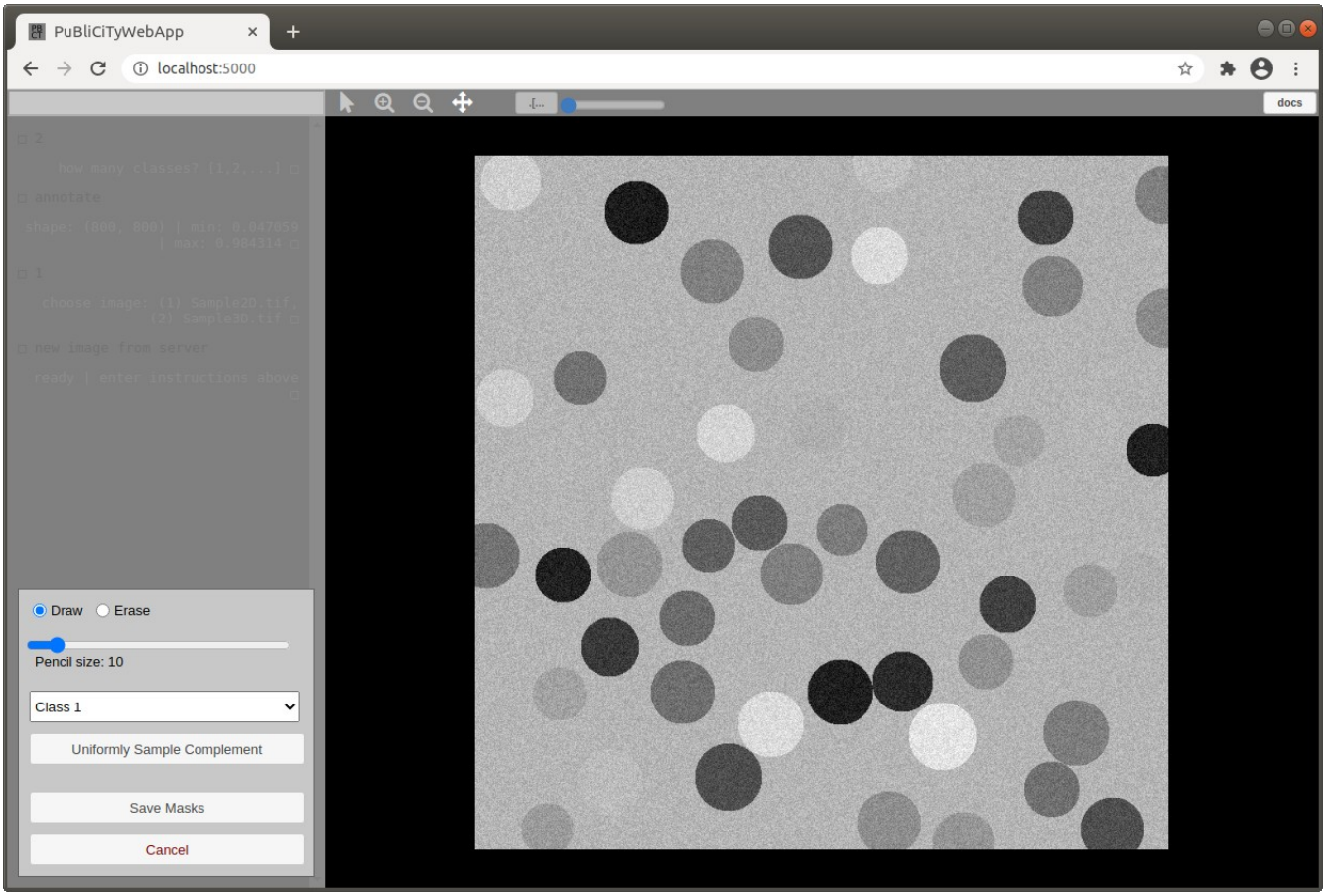
The main reason for this web-app is to gather annotations for machine learning models. With the 2D sample image open, type the following on the prompt box:

```
annotate
```

The app then asks:

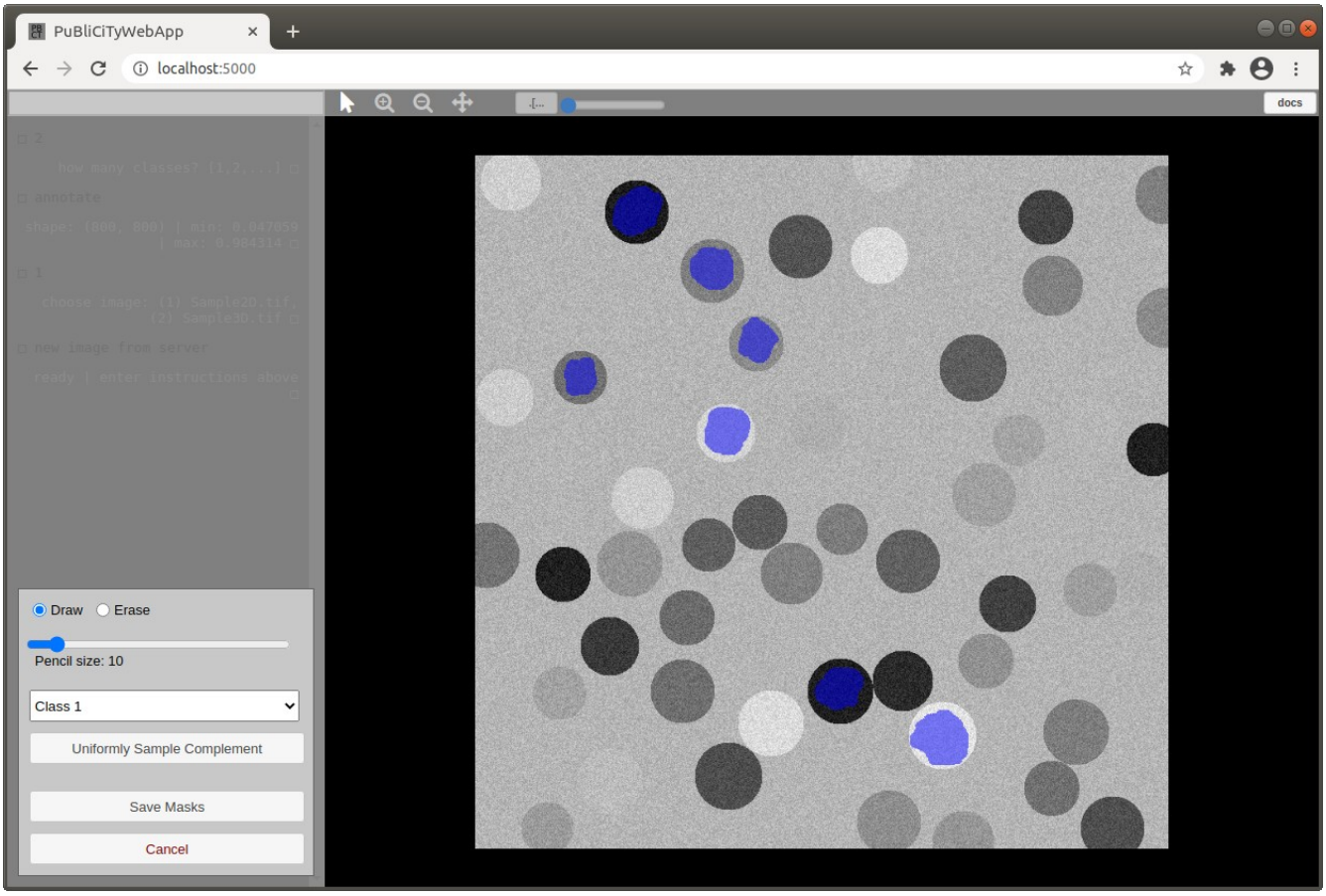
```
how many classes? [1, 2, …]
```

For this exercise, enter 2, but any integer 1, 2, 3, … is valid and depends on your application. After entering the number of classes, the ‘annotation tool’ is displayed on the bottom left of the interface:



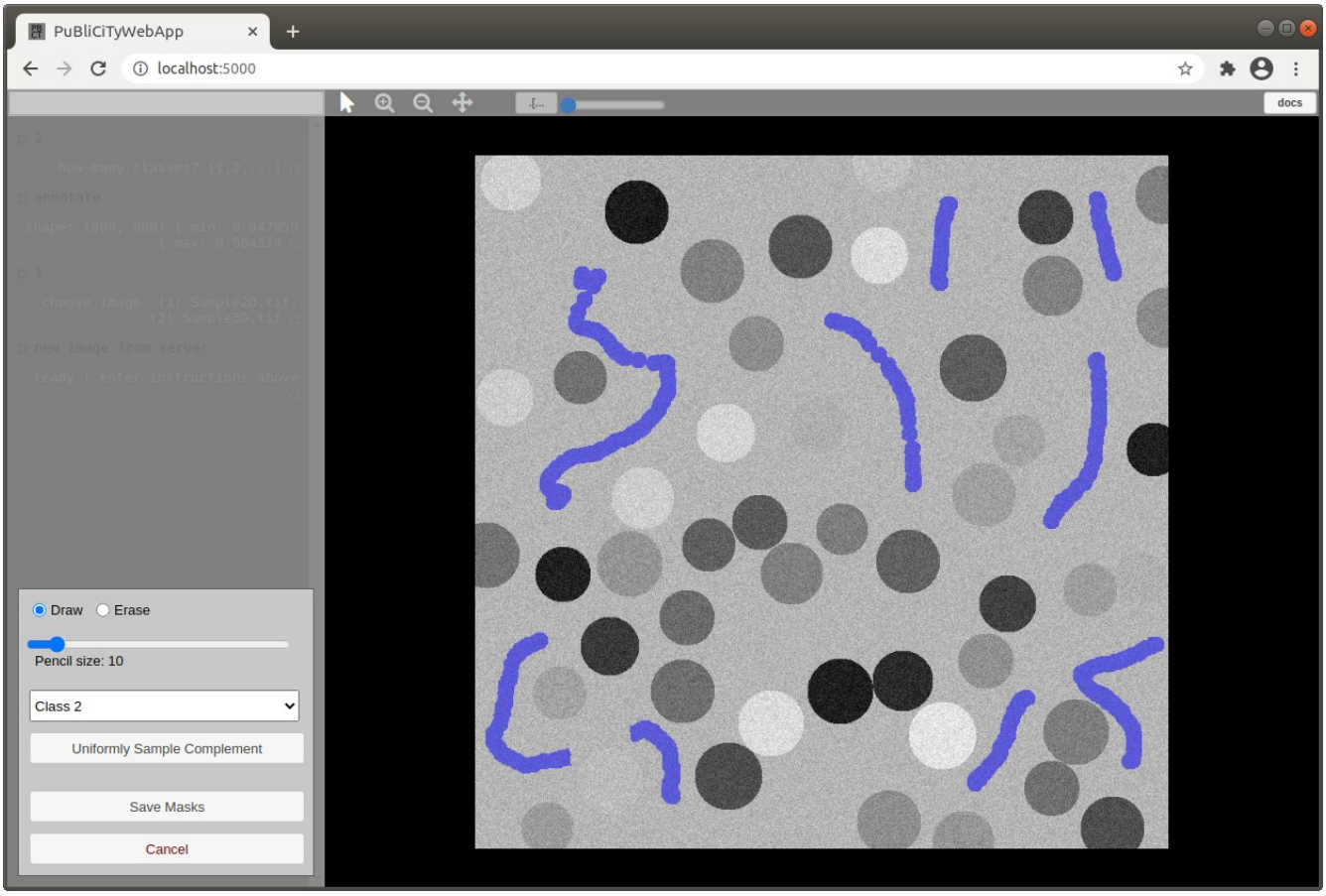
The commands here should be self explanatory. Check in particular the pencil size slider and the Class popup menu. We will draw cells as Class 1, and background as Class 2. To draw, click on the pointer icon (to the left of the zoom + icon).

The following screenshot shows partially annotated cells:



To annotate other classes, simply chose that class on the popup menu in the annotation tool.

The following shows partially annotated background:



You can switch between classes during annotation. To save them, click on Save Masks. This will cause the annotation tool to disappear, and the dialog box to display

```
annotations uploaded
```

Behind the scenes, what happened is the app saved an [image, annotation] pair under Server/User/MachineLearning/Annotations:

Image_002_000_Img.tif
Image_002_000_Ant.tif

The two numbers in the name indicate the number of classes and the index of the annotated image (starting with 0). You can use these annotations as you please. If you were to annotate another image with two classes, and save the masks, they'd be saved as

Image_002_001_Img.tif
Image_002_001_Ant.tif

These images will disappear every time you re-load or re-start the web-app. To save them permanently, type the following on the prompt box:

```
save annotations to server
```

Then enter a folder name for your annotations. If we use, say, SampleAnnotations, the app will create a subfolder with that name and put the annotations there:
Server/User/MachineLearning/Annotations/SampleAnnotations

These are saved permanently (i.e. they don't disappear if you re-load or re-start the app). However, you can of course remove/delete that folder 'manually' if you like.

If you want to load and edit annotations, type in the prompt box:

```
load annotations from server
```

Then the index corresponding to the name you gave to your set of annotations. Now enter on the prompt box:

```
edit annotations
```

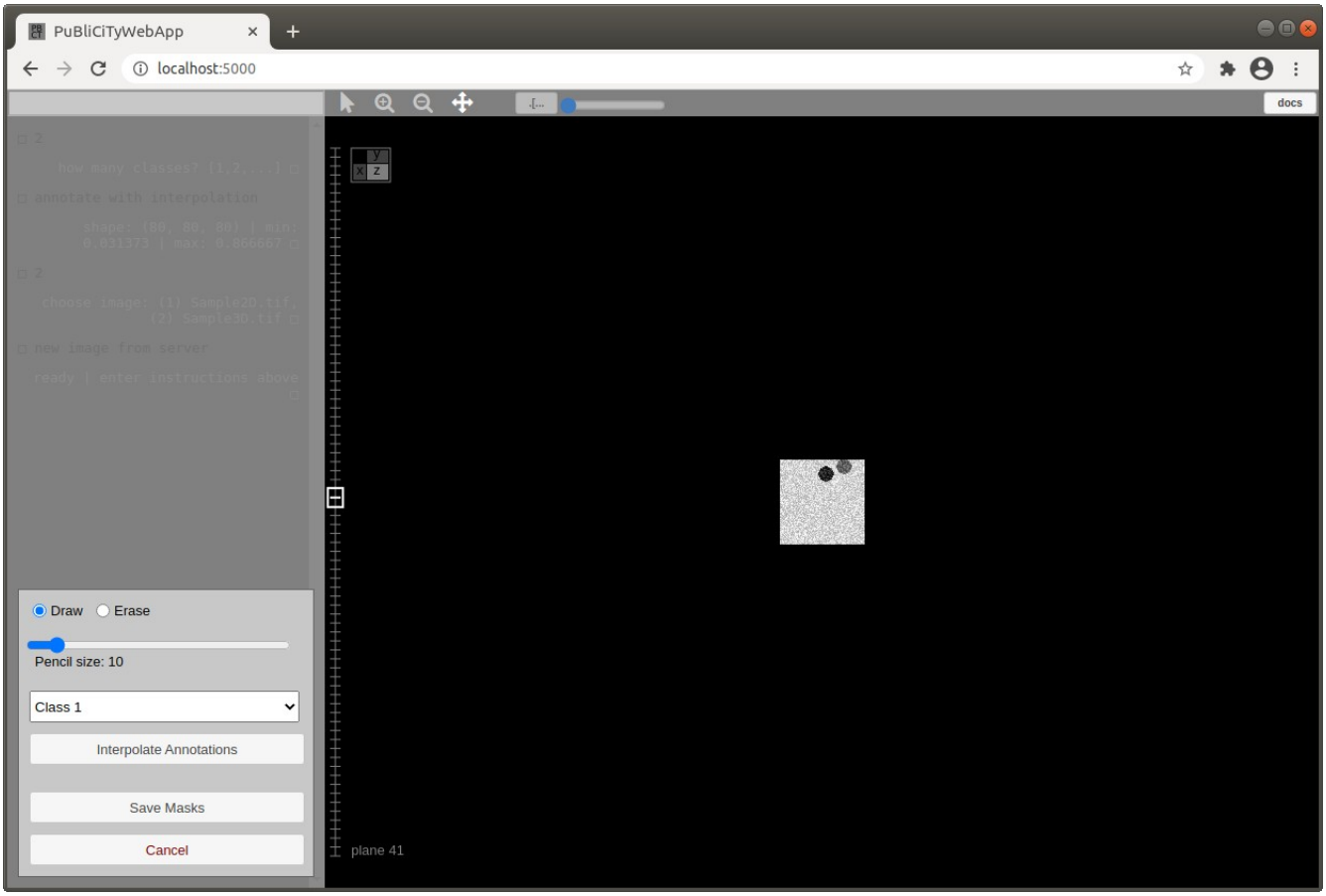
Then enter the index of the image whose annotations you want to edit, and the app will open them. From here you can alter the annotations and re-save them. Pay attention to the files in the Annotations subfolder to understand what happens behind the scenes during these editing steps.

4. 3D Annotations

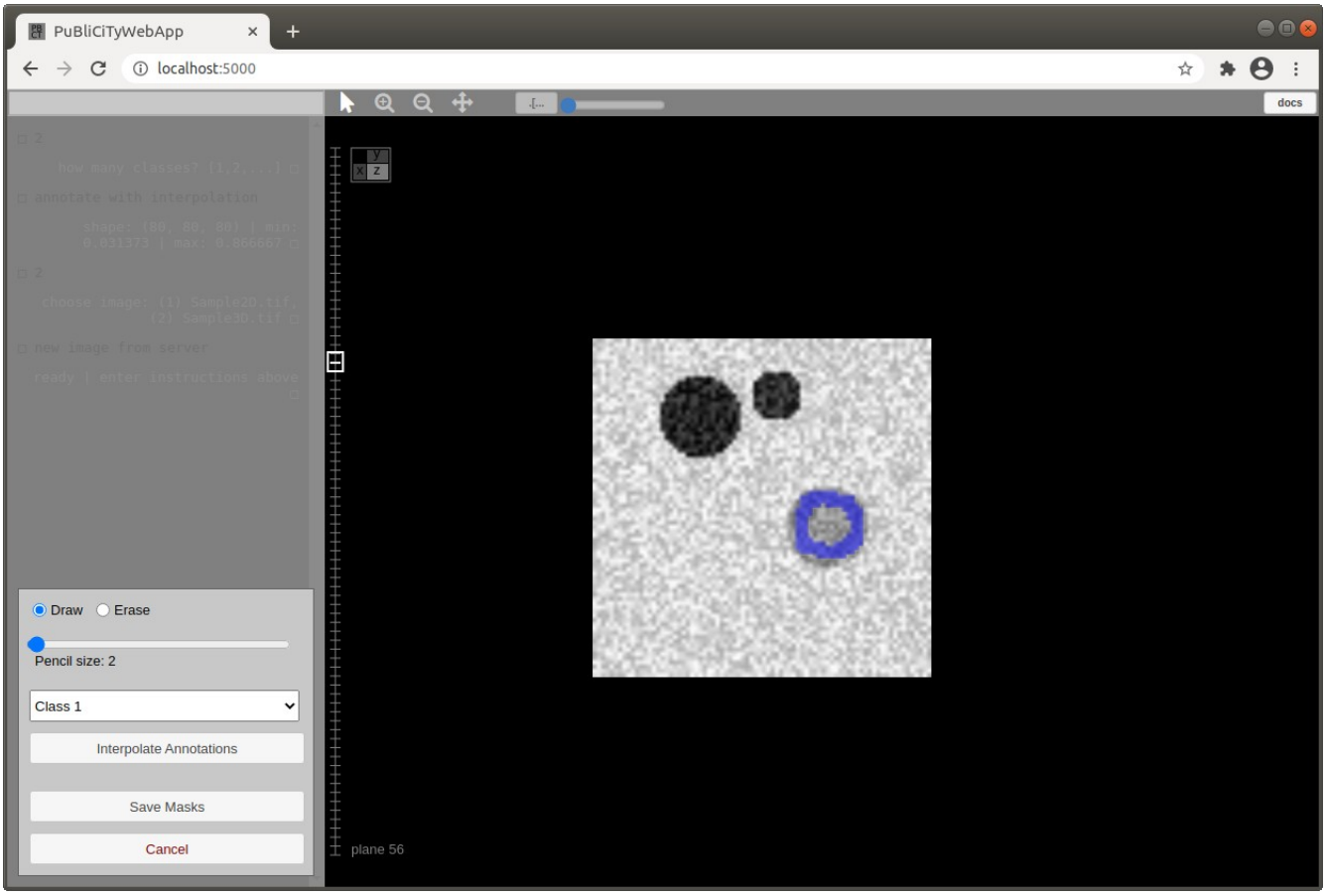
The steps for 3D annotations using the 'annotate' command are the same as above. There's a better 3D annotation mode though, which is accessed via the command

```
annotate with interpolation
```

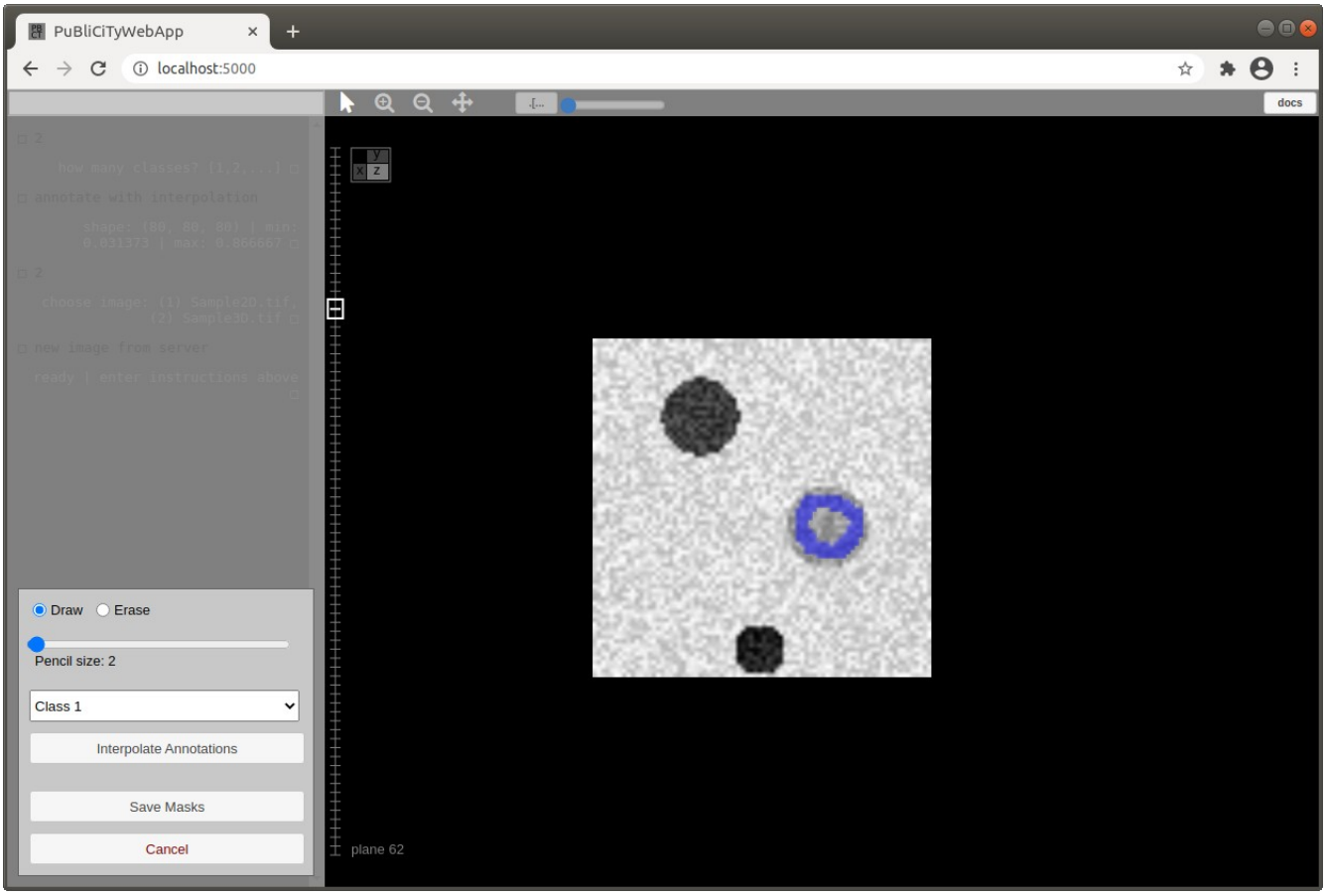
Load the 3D sample image and enter the above command. Then, when prompted, enter 2 as the number of classes to annotate.



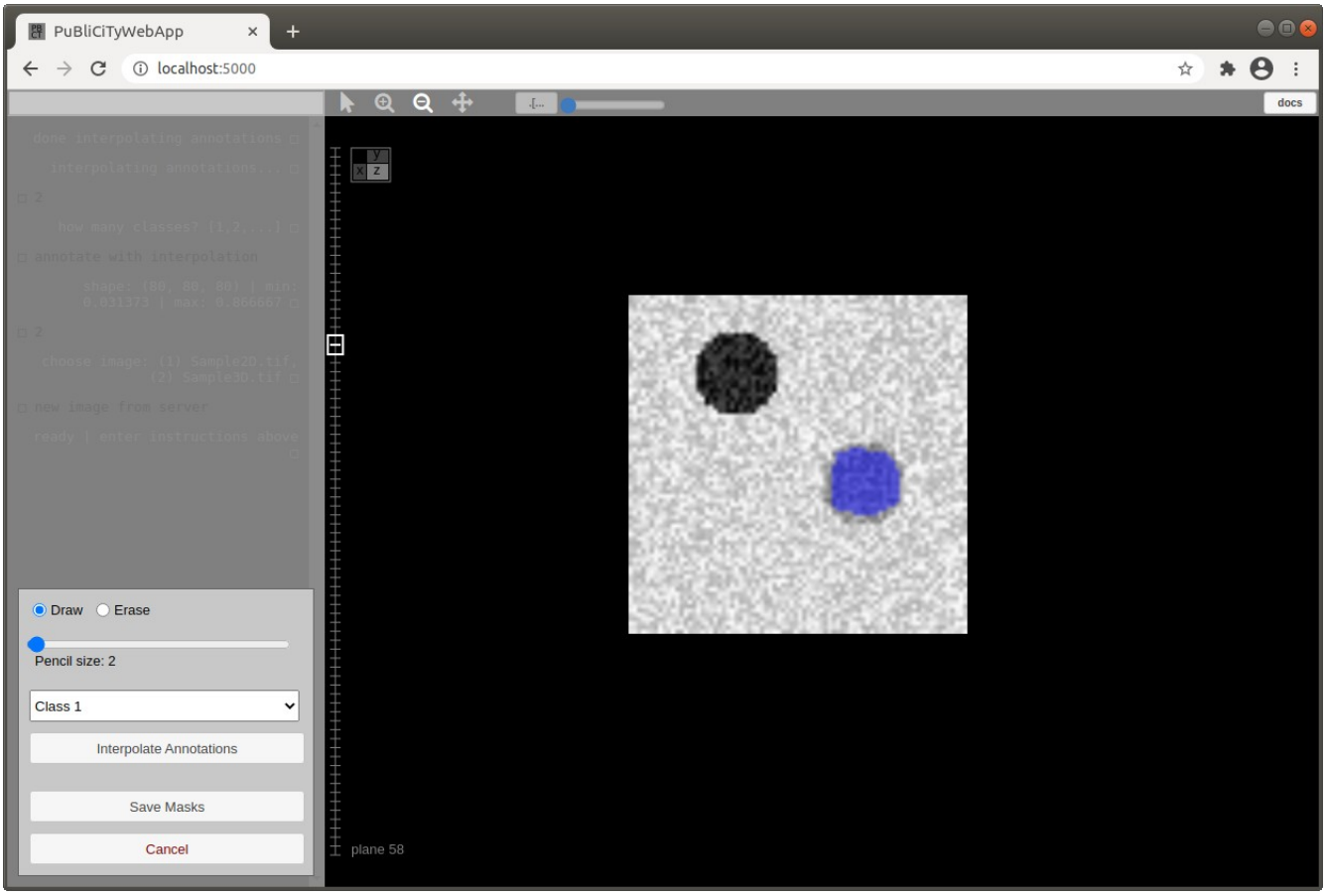
Notice that now the annotation tool has a new option: Interpolate Annotations. This will interpolate annotations to intermediate planes. To use it properly, you have to annotate *closed* contours for the same shape in different planes, and the app will interpolate the annotations for in-between planes. For example, if we annotate a sphere in plane 56 like so...



...then the same sphere in plane 62 like so...



...we can now press Interpolate Annotations, and the app will (1) fill in the annotate contours for planes 56 and 62, and (2) interpolate annotations for intermediate planes 57, 58, …, 61. In our example, the annotation for plane 58 is interpolated like so:



You can draw contours for more than 2 planes before pressing Interpolate Annotations, but you can only draw one contour per plane, and they have to correspond to the same object.

As in the other annotation mode, you can change classes and save annotations.