① Find a clever way to import this (links to python:

LIVECell-wide train and evaluate

Annotation set	URL
Training set	link
Validation set	link
Test set	link

→ Create 3 files:
1 for training
" " testing
" " validation
(Hint: They should be .json files)

② Open one of the files that you have created and observe the structure.

It contains 2 nested dictionaries

1st "images"
contains info about each image
(e.g. name, width, ...)

2nd "annotations"
contains info about the masking process

⇒ So find a way to go, for each image, from ① → ②

③ You must find the keys in ② that contain the info to segment the image

(Reminder: Dictionaries contain keys & values)

④ Find x,y coordinates:

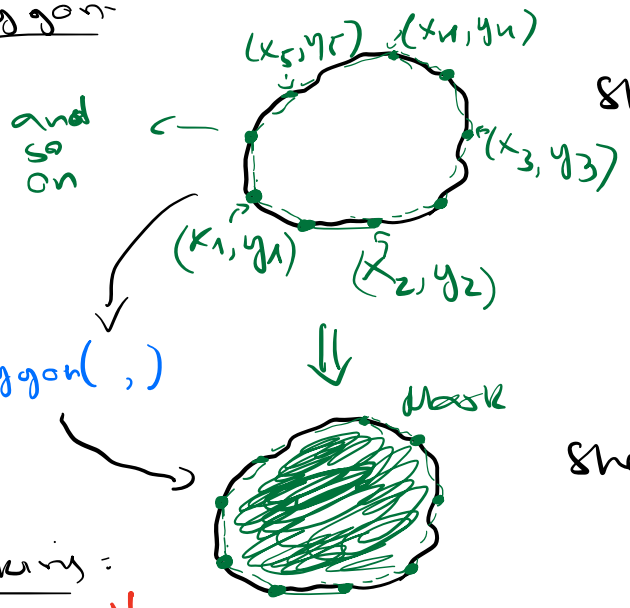
Take from the now acquired list every second value and use them as your x-coordinate
& take every first value and use it as your y-coordinate

! Initialize the mask to be of the same shape as the image

Exp: [0, 1, 2, 3, 4, 5 ...]

(x_1, y_1) (x_2, y_2) (x_3, y_3)

(5) Polygon:

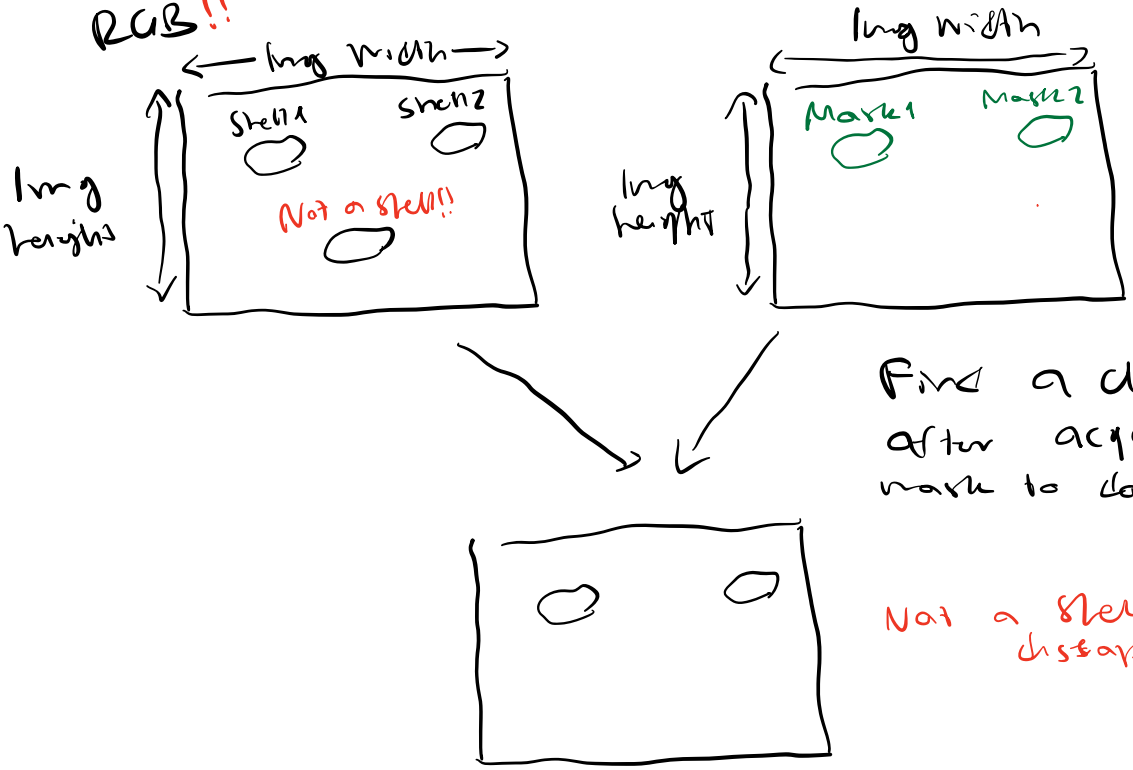


! Very rough approx of how the mask is created

!! Find a python function that creates a polygon

(6) Masking:

RGB!!



Find a clever masking after acquiring the mask to do this filtering!

Not a shell disappeared