

ScopeSys Software Engineer Take Home Task

This assignment is meant to give you two simplified examples of the work that we do everyday, as well as give you a chance to demonstrate your approach to the technical challenges you might face.

You may use any programming language or library you're familiar with, and these challenges are fully open book.

Please share the code and your final output of these challenges as a ZIP.

Challenge #1: Microscope Image Stitching

In our experiments, we image a pit array where we trap DNA molecules for observation.

Our pit array is a grid that covers several hundred microns in either direction, but our camera only covers 80 x 80 microns at a time. In order to visualize the entire pit array, we take a snapshot of each part of the grid, with some overlap, and then stitch them back together into a single image.

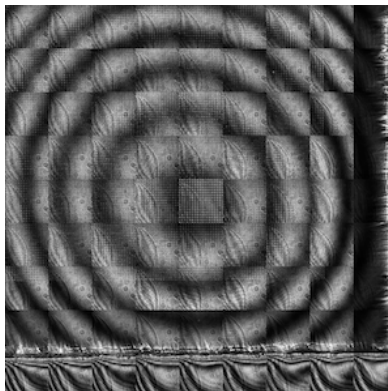
We've given you 81 grayscale TIFF images, representing the 9 x 9 grid of our pit array. Each grayscale image is named in the format:

test_row{ROW_NUMBER}_col{COLUMN_NUMBER}.tif

So, for instance, the image in the first row, first column is **test_row1_col1.tif**.

There is **24 pixels of overlap in both the x and y directions** of each image.

Your goal is to generate a stitched version of the image, given the microscope snapshots, using code. Your final output should look similar to this:

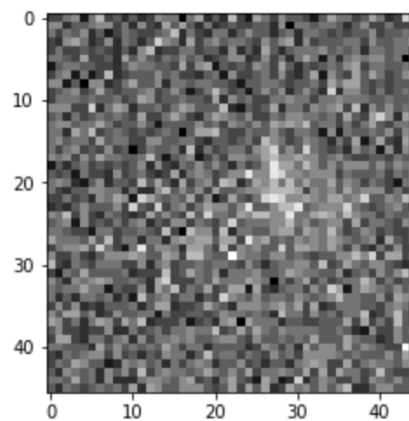


Challenge #2:

The molecules we observe often have fluorescence that is similar to the background noise in our flow cells.

We first often want to remove the background noise in these images to analyze our data.

Next, the molecular complex that we're interested will be lit up due to the fluorescence, and will look like a bright intensity spot bouncing around the pit:



In this image, you can see the complex at roughly by the (30, 20) XY coordinate.

For this challenge, we've provided with you with a set of 50 grayscale images that represent the frames of a video taking with our imaging system. Each image is in the format:

image_{TIMESTAMP}.tif

Where timestamp is the frame of the video in order, from 0 to 50.

For each frame of the video:

- Reduce the noise in the background in any way you can.
- Place a green pixel as close as you can to where the center of the complex would be.