**Name – P.B.K. Sanjana Gimhani Kapukotuwa**

**Index Number – 200287L**

**EN3160 Assignment 2 on Fitting and Alignment**

GitHub repo: <https://github.com/sanjanakapukotuwa/EN3160-Image_Processing_and_Machine_Vision>

**Question 1**

Range of sigma used: 1-3

Radius of the largest circle: 25 , corresponding sigma value = 3



A screenshot of a computer code

Description automatically generated

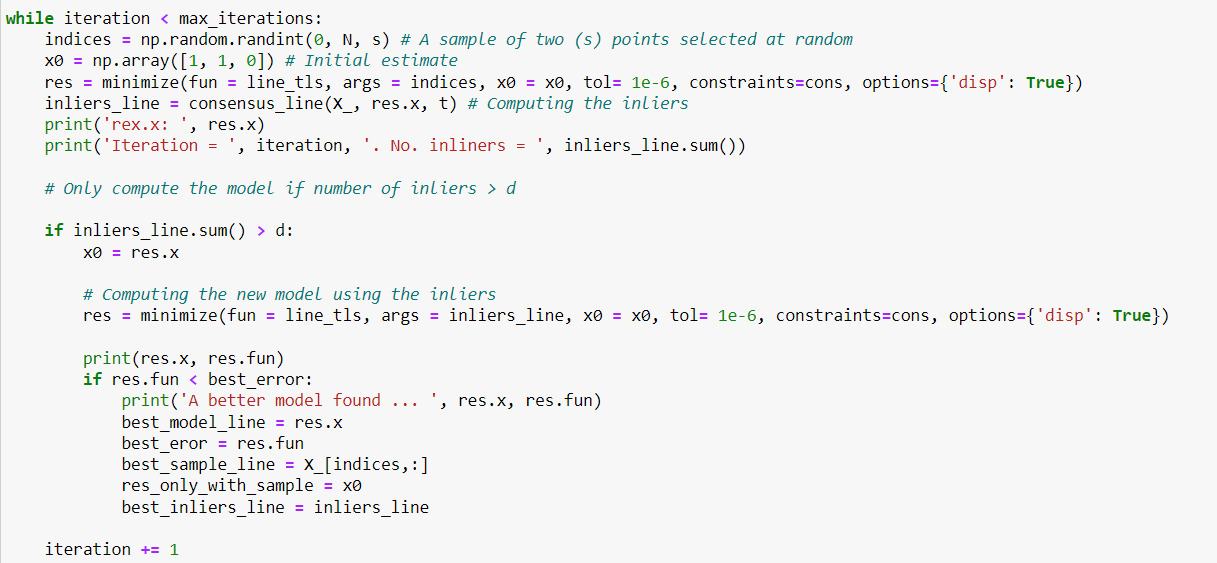
**Question 2**

The steps of estimating the best model for a line/circle using the RANSAC algorithm are,

* select s number of points at random
* fit the model to those s points
* find the inliers of that model that lie within the given threshold t
* if the number of inliers is greater than the minimum amount required, refit the model considering all the inliers
* repeat until we find the best model

A graph with colored lines and dots

Description automatically generated with medium confidenceBest fit line is estimated using the inliers corresponding to the best sample using scipy.optimize.minimize



A circle with colored dots and text

Description automatically generated with medium confidenceBest fit circle is estimated using the inliers and center point corresponding to the initial circle passing through the best 3 samples using scipy.optimize

A white background with black text

Description automatically generated

A screenshot of a computer code

Description automatically generated

A close-up of a computer screen

Description automatically generated

The ground truth line/circle, model passing through the sample points, best fitting model using the inliers of both the line and circle are plotted separately below.

(c) Final models

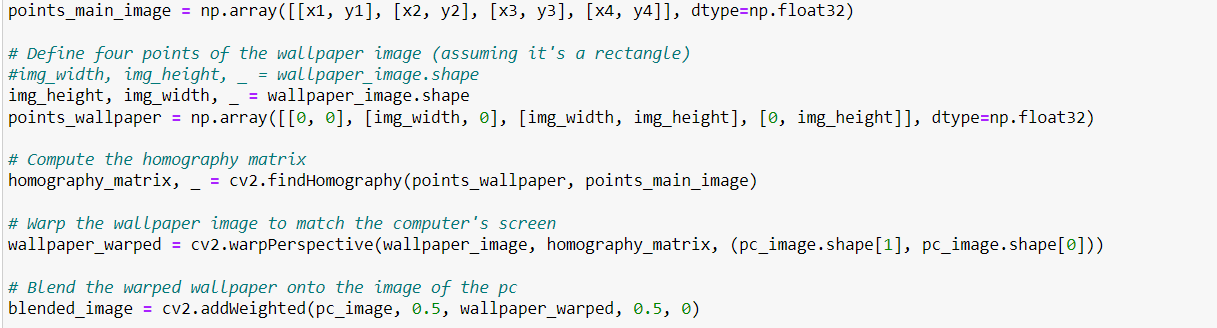
A graph of a circle with colored dots

Description automatically generated

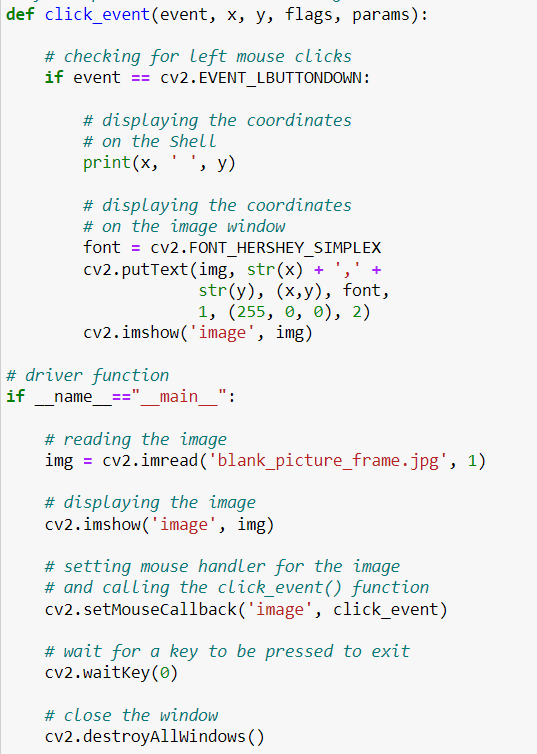
(d) If we try to fit the circle at first, the model will consider the points corresponding to the line to be points on the radius of the circle as well and estimate a circle with a very large radius passing through those points. This will lead to an incorrect model.

**Question 3**

The below code computes a homography that maps the flag image to this plane, and warps the flag, and blends it on to the source image.

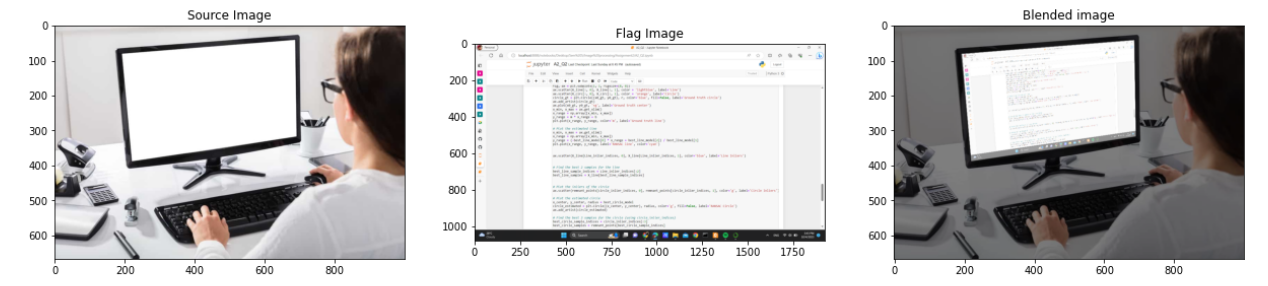
****

The below code snippet was used to identify the 4 points in the source image on which the flag image must be superimposed.

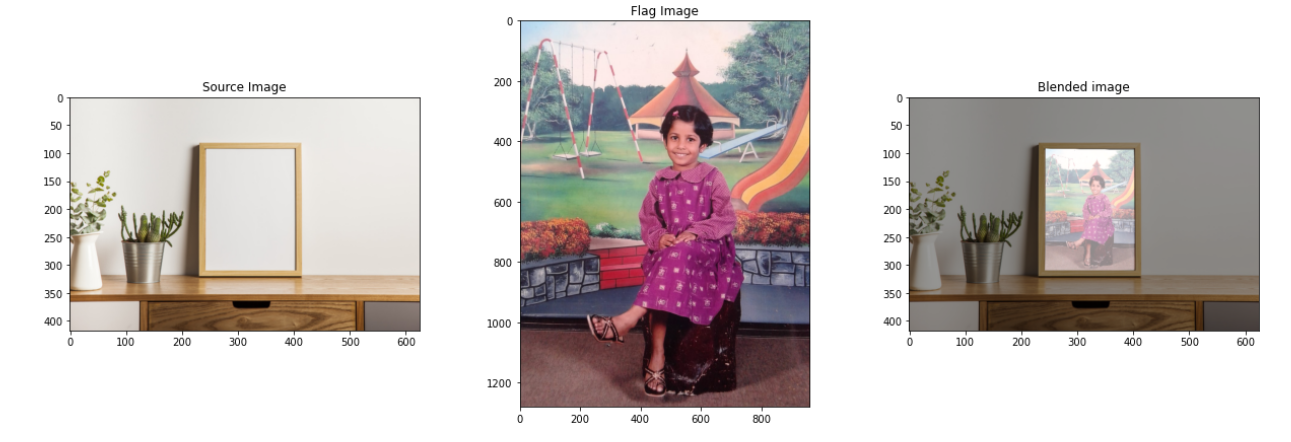
** A screenshot of a computer program

Description automatically generated**

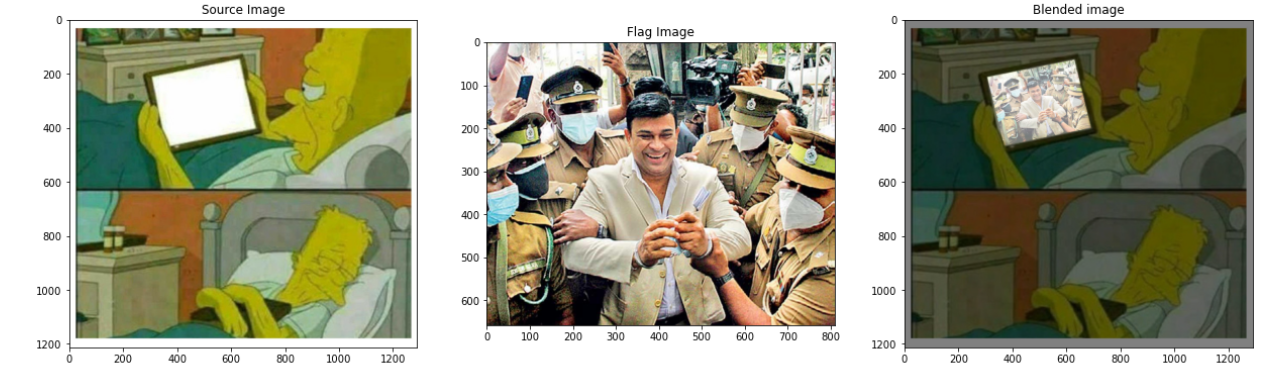
I clicked a screenshot of my laptop screen and superimposed with the source image so that it seems that the woman in working on some code on her computer.



A childhood image of me superimposed with an image of an empty picture frame kept on a table.



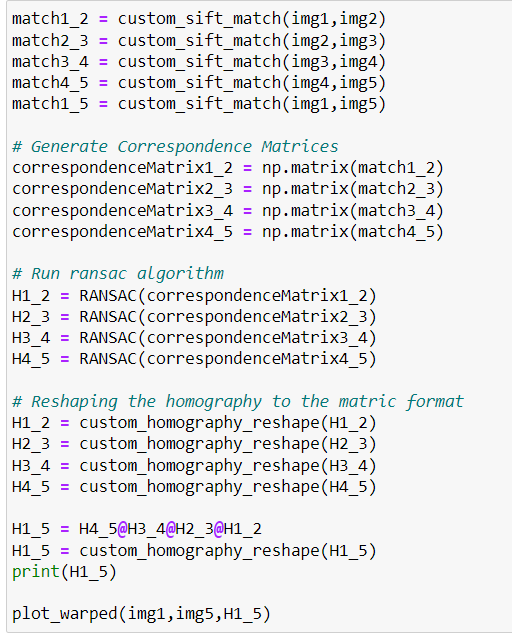
At last but not least, why not use this method to generate some meme material. 😉



**Question 4**

Few functions were defined to compute and match sift features and to compute the homography.

The below code snippet was used to call these functions and to stitch the two graffiti images. Output is also displayed below.



Homography obtained from the calculation.

A number and numbers on a white background

Description automatically generated