**EN2550 – Fundamentals of Image Processing and Machine Vision**

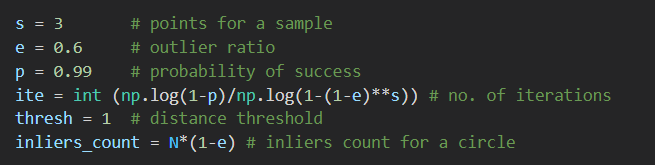
Assignment 02

Name: ADIKARI A.M.A.D.

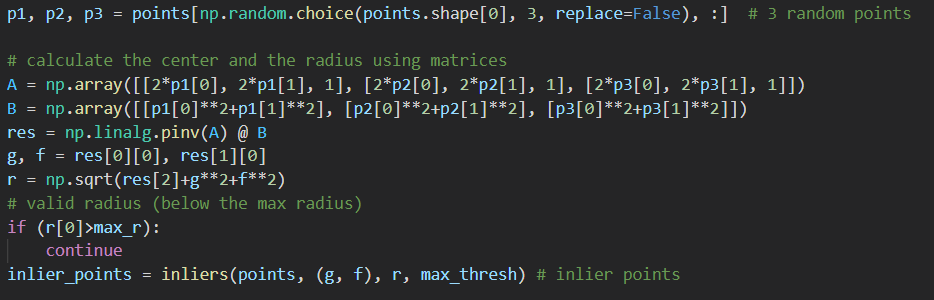
Index No: 190021A

* **Question 01**

- In this question, we must find the best-fitting circle using the RANSAC algorithm for given data points. One hundred data points have been generated with noise.

- We need three points for a sample and for parameter values in Figure, we can calculate the number of iterations that we need to run to get an accurate circle.

* **RANSAC algorithm**

-By using the RANSAC() function, first, we get three random sample points and then we determine the circle for the selected points. According to the following figure, the radius and center of the circle can be found by using matrices. Then the radius is checked whether it is a valid value as if the sample points are very closer to each, the radius would be higher. After that, we can find the inliers for the circle by using the inliers() function. Inliers are found in region radius ± threshold. For this example, 1 is given as the threshold.

Text

Description automatically generated - If the inliers count is greater or equal to the given threshold count, the above RANSAC process is repeated to find a more accurate circle with higher inliers count. All these circles are listed.

Chart

Description automatically generated- After all the iterations perform, the circle with the highest inliers count will be selected as the best circle by using the best\_RANSAC() function. If several circles have similar highest inliers count, then the circle with the minimum mean absolute error will be selected.

- This figure shows the results of the RANSAC circle fitting. For this example, the Inliers count is 48. + signs represent the centers of the circles. Two circles are the first and second results of running the RANSAC function. Circle points are got by using the circle\_points() function. Some of these data points are according to a line outside the circle.

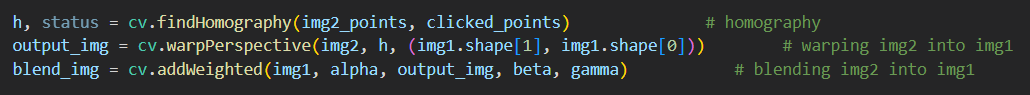
* If we use a method other than RANSAC, those points will be affected to the circle fitting due to the approximate linear spreading. Therefore, the RANSAC method is especially useful and valuable for fittings in data points with widespread outliers.
* **Question 02**

- In this question, we need to find a homography for image pairs and by using that, warp and blend one image to a plane of the other image.

* **Algorithm**

- First, the main image is shown using the OpenCV window. Then we can click 4 points (for the corner points of the warping image) using the mouse in a desired plane and the points are shown in red circles. The mouse\_click() and mousePoints() functions are used for this purpose.

- In the warpAndBlend() function, a homography is determined using the findHomography() built-in function for the image pair. Then by using that, we can warp and blend the warping image to the selected plane of the main image as shown in the following figure.



- Here, alpha and beta parameters in the addWeighted() function are the weights of the two images and gamma is the value that is added for the final image. Then the results are shown using matplotlib.

* **Building and the flag**

A red and blue logo

Description automatically generated with low confidenceA flag in front of a building

Description automatically generated with medium confidence

A building with a flag painted on it

Description automatically generated with low confidence

- This figure shows the final output image. The flag is superimposed on a wall of the building. For this example, the parameter values in the blending are as follows.

alpha = 1

beta = 0.8

gamma = 0

* **Car and the logo**

**A white sports car

Description automatically generated with medium confidence**- In this image, the logo of a horse is superimposed on the bonnet of the car. In this logo image, the horse is not rectangular and if we use the logo directly, a rectangular shape of the logo will be shown on the bonnet. Therefore, the outside region of the horse is converted to black color by using a mask. Then the black region will not affect when we add the weights. For this example, weight parameters are used as alpha=0.7, beta=0.3, and gamma=0 to get a good and some kind of dark output.