The report of mini-project N3 program

by Ulvi Bajarani

SID 20539914

All retrieved results and processings are saved in the log file created by the program (if the log files don’t exist, the program creates it. Otherwise, it appends): **logFile.txt**

The program processes all images that must be in the **/imagesToProcess/Images** folder located in the same folder with the **/python** folder. In other words, **/imagesToProcess/Images** folder must be one level up than the code file. The templates are saved in **/imagesToProcess/Templates** folder located in the same folder with the **/python** folder. In other words, **/imagesToProcess/Templates** folder must be one level up than the code file. All templated named as **image\_name\_template.extension** format. While the image dimensions are 512x512, template dimensions are 32x32 and the templates made from the images resized to 256x256 (included in **/imagesToProcess/resizedImages** folder)

The .py file could be found in **/python** folder. The image processing results are saved in the **/resultsOfProcessing** folder. The **/python** folder contains the code files.

The function **picture\_writer(images\_list, folder\_path\_to\_process, folder\_path\_for\_templates, folder\_path\_to\_write, templates\_list, no\_of\_levels, metric)** takes the list of images **images\_list**, the folder path to get image names **folder\_path\_to\_process**, the folder path to get image template names **folder\_path\_for\_templates**, the folder name to write **folder\_path\_to\_write**, the list of image templates **templates\_list**, number of pyramid levels **no\_of\_levels**, and the correlation metric **metric**. The metrics were used:

Normalized Sum of Squared Differences – the variable name is **cv2.TM\_SQDIFF\_NORMED** In the processed file names, the files are named with additional ‘1’

Normalized Cross Correlation – the variable name is **cv2.TM\_CCORR\_NORMED**

In the processed file names, the files are named with additional ‘3’

The template matching is done by **cv2.matchTemplate(layer,template,metric)** function, while the pyramids are created by **cv2.pyrDown(layer)** function. The locations are got by **cv2.minMaxLoc(result)** function. All results for one image are concatenated in one image. The images with lower resolutions are written to the black image with the same number of rows as the highest layer (*512*). The concatenation is done horizontally by **cv2.hconcat([image1, image2])** function.

The result image contains, from left to right:

1) Original image, undetected

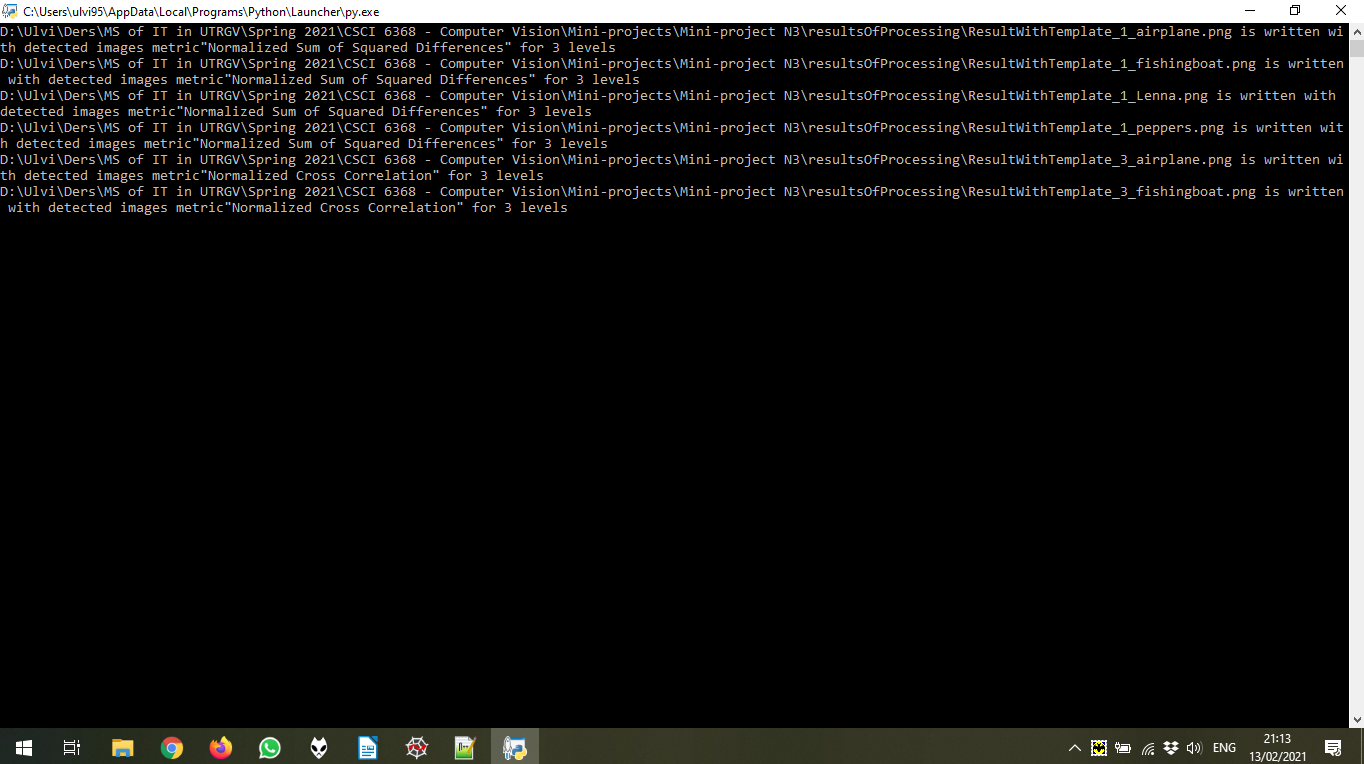
2) Layer 0 (Original image), detected (with black square)

3) Layer 1, detected (with black square)

4) Layer 2, detected (with black square)

5) The template used

The program describes the actual state of processing. In the screenshot below, it can be seen that what processes ended.



Below the result of processing for 1 image (out of 4) chosen randomly. As mentioned above, all results could be found in the **/resultsOfProcessing** folder.

With Normalized Sum of Squared Differences



With Normalized Cross Correlation

