Cairo University
Faculty of Engineering
Systems and Biomedical Engineering

Computer Vision

Task 3 Report

Submitted to:

Dr. Ahmed M. Badawi

TA/ Eng Laila Abbas.

TA/ Eng. Peter Salah.

Prepared by: Team 8.

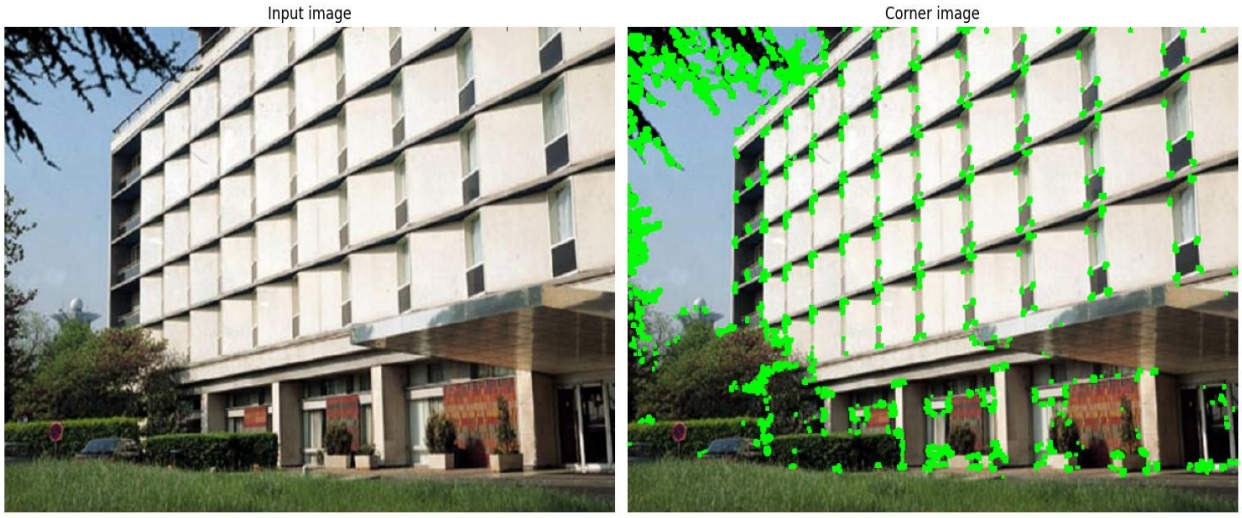
Algorithms Implemented:

Harris operator Algorithm

The Harris corner detection function is used for identifying corners and inferring features of an image. Corners are the important features in the image, which are invariant to translation, rotation, and illumination. Harris work by computing a score for each pixel in an image, based on the amount of variation in intensity that occurs when the image is shifted by a small distance in any direction. Harris function take the image and lambda and first computes the gradient of the image in both the x and y directions using a Sobel filter. Then compute the products of the gradients at each pixel, as well as their sum of squares. Then calculate a "structure tensor" for each pixel. Applying Gaussian filter to reduce noise. Then The corniness value" the corner response function" can be defined as:

R = det (M) - k (trace (M)) ^2

Where det (M) is the determinant of the structure tensor, trace (M) is the trace of the structure tensor, and k is a constant parameter after applying thresholding to identify corners.



- Execution Time

```
Execution time for harris detector:  8.638997793197632
```

SIFT Algorithm

SIFT Scale-Invariant Feature Transform (SIFT) is another technique helps locate the local features in an image, commonly known as the '*keypoints*'of the image. These keypoints are scale & rotation invariants that can be used for various computer vision applications. The entire process used to identify these keypoints can be divided into 4 parts:

- Constructing a Scale Space**:** To make sure that features are scale-independent.
- Keypoint Localisation: Identifying the suitable features or keypoints.

- Orientation Assignment: Ensure the keypoints are rotation invariant.
- Keypoint Descriptor: Assign a unique fingerprint to each key point.

Main difference between Harris and SIFT:

The Harris Detector, shown above, is rotation-invariant, which means that the detector can still distinguish the corners even if the image is rotated. However, the Harris Detector cannot perform well if the image is scaled differently.The SIFT detector is rotation-invariant and scale-invariant.

- Execution Time

```
Execution time of SIFT is 41.53581357002258 sec
```

Feature Matching Algorithm

An algorithm that calculates how similar the features of the two images are using two techniques which are Sum of Squared Distance (SSD) and Normalized Cross Correlation (NCC) from the key-points and descriptors calculated from the previous SIFT algorithm. The higher the score the more the two images match. Then we decide if each score between two features is strong or not based on a threshold related to the key points. And finally, draws the matched lines.

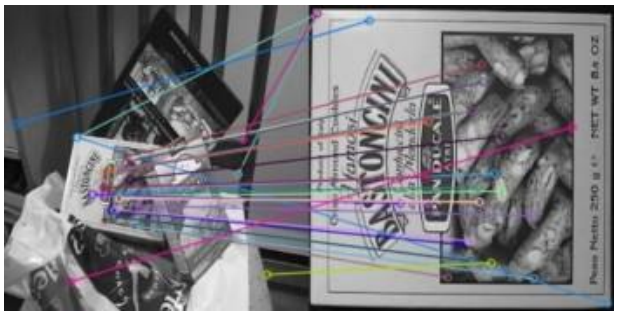
- SSD is calculated from calculates algorithm that return the sdd_score in negative value since as sdd_score decrease the matching increase. And ssd_score is calculated from the following formula:

```
ssd = - (np.sqrt (np.sum ((des1 - des2)**2)))
```

- NCC is calculated from calculate_ncc algorithm that return the ncc_score where, as ncc_score increase the matching increase. And ncc_score is calculated from the following formula:

```
normlized_output1 = (des1 - np.mean (des1)) / (np.std (des1))
normlized_output2 = (des2 - np.mean (des2)) / (np.std (des2))
correlation_vector = np.multiply (normlized_output1, normlized_output2)
ncc = float (np.mean (correlation_vector))
```

SSD Output



NCC Output

- Execution Time

```
Execution time of SIFT is 98.38849949836731 sec
SSD computation time:  1.901520013809204
NCC computation time:  22.264068603515625
```