CS-512 Assignment 3: Report

Russi Sinha
Department of Computer Science
Illinois Institute of Technology

Abstract

This is a report for Assignment 3 for CS512. The task given at hand is to perform corner detection using the Harris Corner Detection Algorithm, localize the corner and compute and match feature vectors amongst two images.

1. Problem Statement

The program must read an image either from an input file or must allow images to be captures from the camera to be processed. The following tasks must be performed on the captured images:

- Convert image to grayscale
- Perform Harris Corner detection using your own implementation of the function
- Interactively control the corner detection by parameters Gaussian scale, neighborhood size, weight of the trace and threshold value.
- Localize the detected corners
- Compute and match the feature vectors between the two captured images

2. Proposed Solution

For Harris Corner detection the approach used was as follows,

- In a .pyx file the image size and the window size are passed as arguments along with the image gradients and the threshold value
- To find corners, the image is looped over by the window using a double for loop
- In the double loop, at each position of the window, the X an Y gradients are calculated and summed
- The summation of the gradients is what forms the correlation matrix, which is defined as follows,

$$M = \begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix}$$

- Once the matrix is formed, we calculate the eigenvalues of the matrix M to measure the cornerness of the pixels as follows,

$$C(M) = \det(M) - k \times (trace(M))^2 = (\lambda_1 \lambda_2) - k \times (\lambda_1 + \lambda_2)^2$$

- The function in the .pyx file returns the cornerList consisting of the all the corners detected (which includes the corners measure and the pixel location)
- Now, in the ‘assign3.py’ file, the returned cornerList is sorted and the highest 10 pixels are chosen and marked by a blue rectangle.

3. Implementation Details

Before running the program install cython and make sure to place the python file and the setup.py in the working directory and run the following command,

```
python setup.py build_ext --inplace
```

Then run the python program using the command

```
python assignment3.py
```

Steps to run the program:

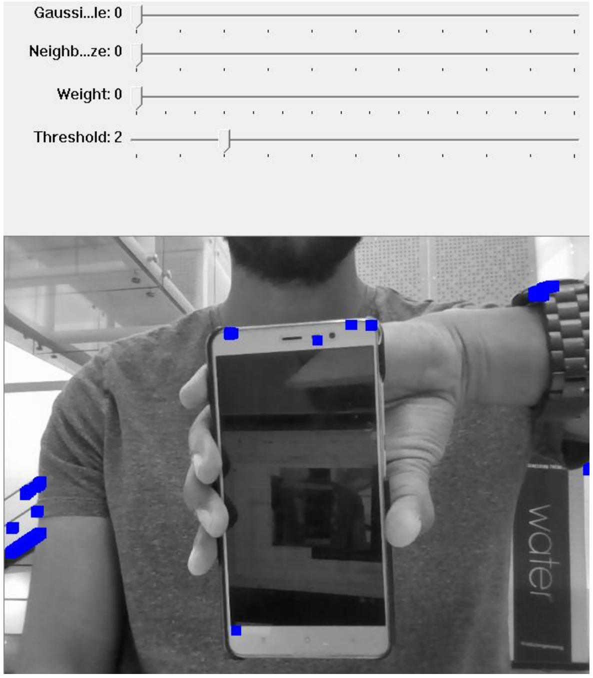
1. The program begins with a keyboard event loop, if the user has not entered the input file in the command line, the program uses the camera to capture images.
2. The key ‘c’ when pressed on the keyboard captures images and writes them into the working directory. (Note: 2 images need to be taken for the program to work as desired)
3. Hitting ‘Esc’ exits the camera mode and displays the image taken in the previous step
4. Control the amount of corners detected by the slider parameters
5. To exit the program press key ‘esc’.

Once the images intended to be processed are captured the following is done:

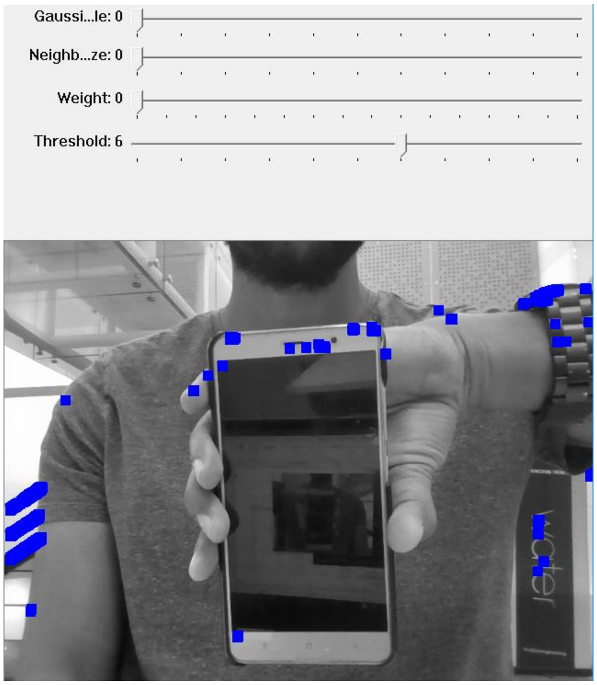
- The images are first converted into grayscale images for easier processing
- The slider is created for each parameter that is, Gaussian scale, neighborhood size, weight of the trace and threshold value.
- Each slider is connected to a function called ‘cornerSlider’ that captures the slider positions and then calls the ‘defineCorners’ function which uses the corners.pyx to perform our own implementation of the Harris Corner Detection algorithm to detect corners.
- The corners are detected dynamically by the slider on the window.

5. Results and Discussion

Below are the corners detected on an image captured by the webcam.



Gaussi...le: 0
Neighb...ze: 0
Weight: 0
Threshold: 2



Gaussi...le: 0
Neighb...ze: 0
Weight: 0
Threshold: 6

6. References

The following weblink were used:
http://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_tutorials.html
https://docs.scipy.org/doc/
http://cython.readthedocs.io/en/latest/index.html