

**CS512 Assignment 3: Report**  
**Arpit Hasmukhbhai Patel**  
**Department of Computer Science**  
**Illinois Institute of Technology**  
**October 16,2018**

## **1. Problem statement :**

1. Load and display two images contains similar content or capture by webcam.
2. Change all the images into grayscale before processing
3. Include the help key describing its functionality.
4. Estimate image gradients and apply Harris corner detection algorithm.
5. Obtain a better localization of each corner.
6. Compute a feature vector for each corner were detected.
7. Using the feature vectors, you computed match the feature points.  
Number corresponding points with identical numbers in the two images.
8. Interactively controlled parameters should include in functions.

## 2. Proposed solution :

**Harris corner detection:**

$$R = \det(M) - k(\text{trace}(M))^2$$

where

- $\det(M) = \lambda_1 \lambda_2$
- $\text{trace}(M) = \lambda_1 + \lambda_2$
- $\lambda_1$  and  $\lambda_2$  are the eigen values of M

If  $R > \text{threshold}$  , we mark it a corner.

**Compute a feature vector for each corner were detected:**

find the keypoints and descriptors with SIFT

SIFT keypoints of objects are first extracted from a set of reference images[2] and stored in a database. An object is recognized in a new image by individually comparing each feature from the new image to this database and finding candidate matching features based on Euclidean distance of their feature vectors. From the full set of matches, subsets of keypoints that agree on the object and its location, scale, and orientation in the new image are identified to filter out good matches.

Use BFmatcher to compare the point you detected and SIFT descriptor found. And number corresponding points with identical numbers in the two images. Brute-Force matcher is simple. It takes the descriptor of one feature in first set and is matched with all other features in second set using some distance calculation. And the closest one is returned.

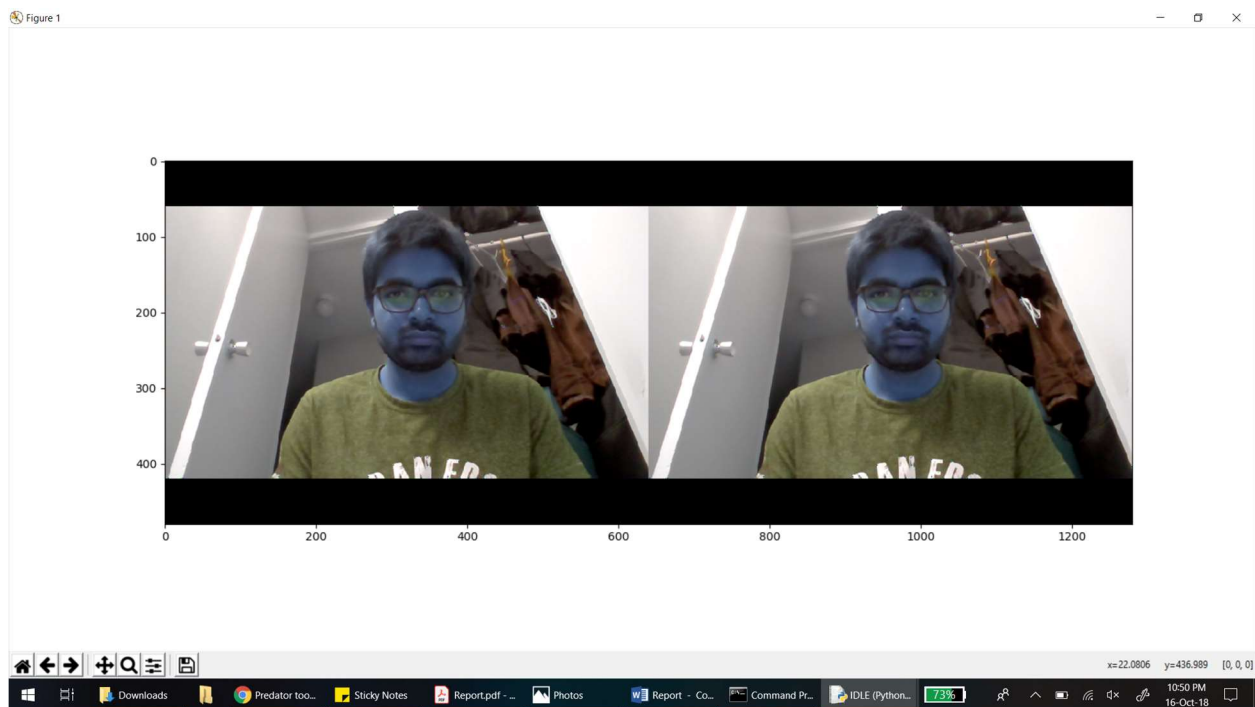
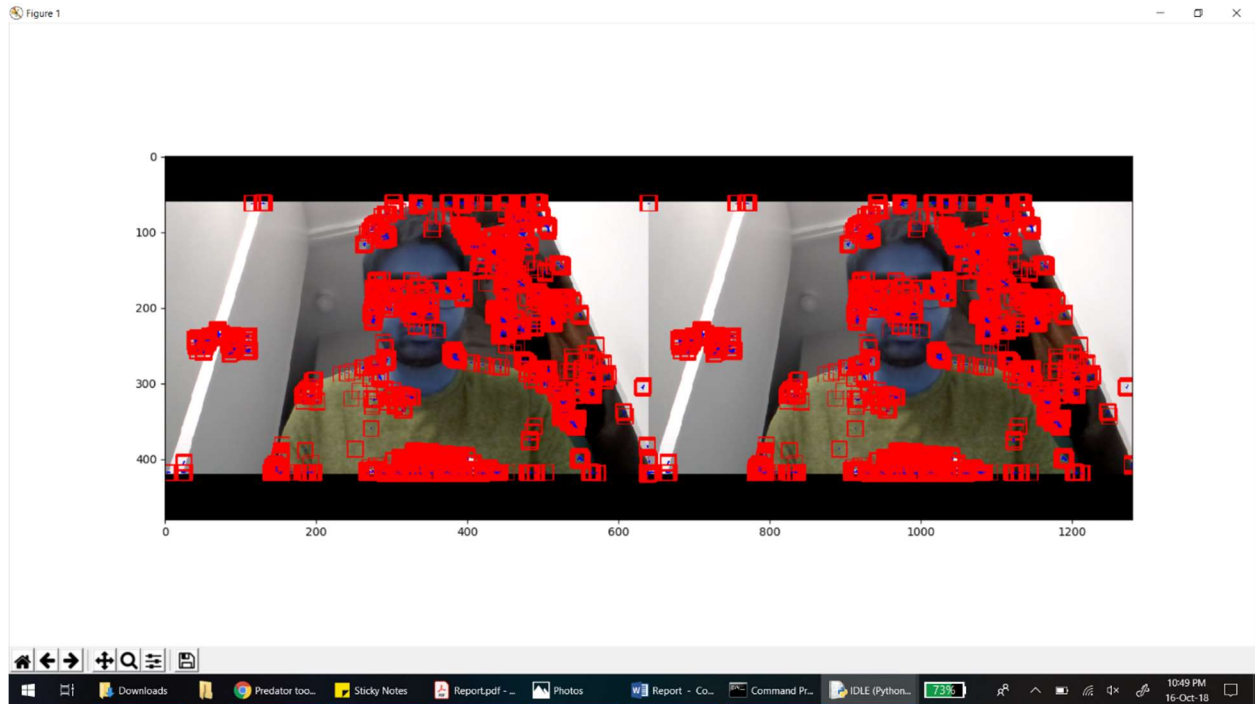
**Better Localization:**

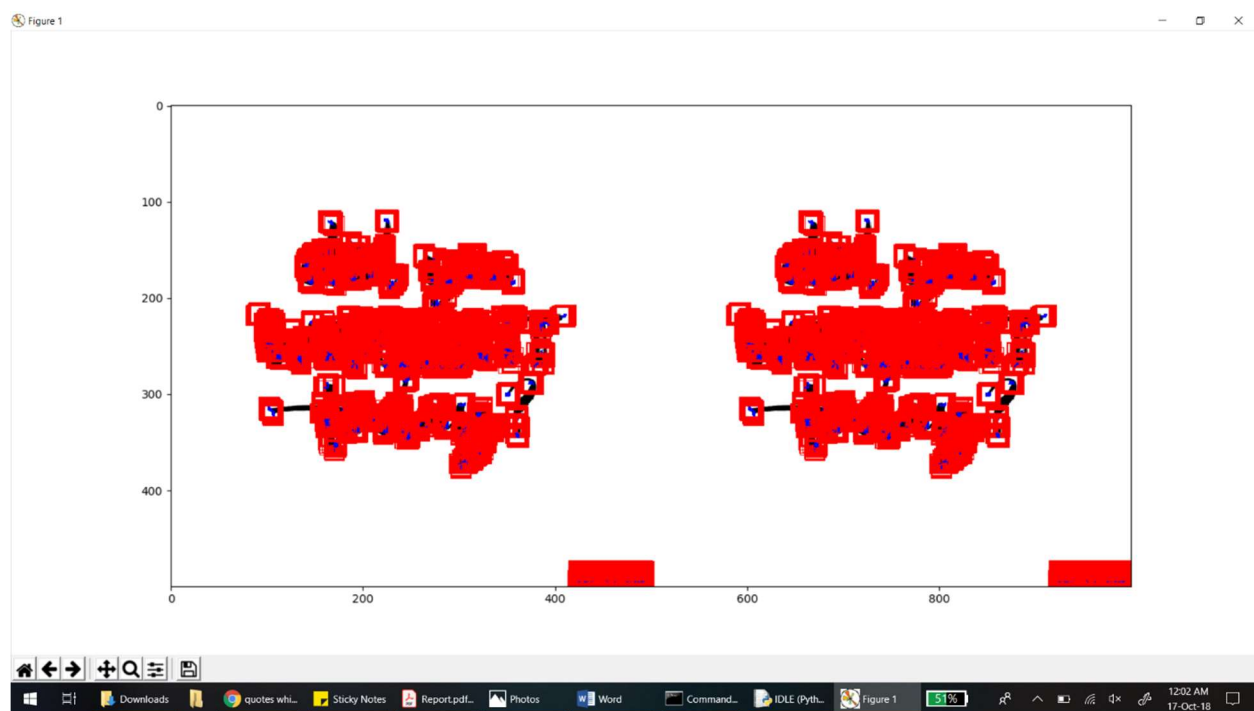
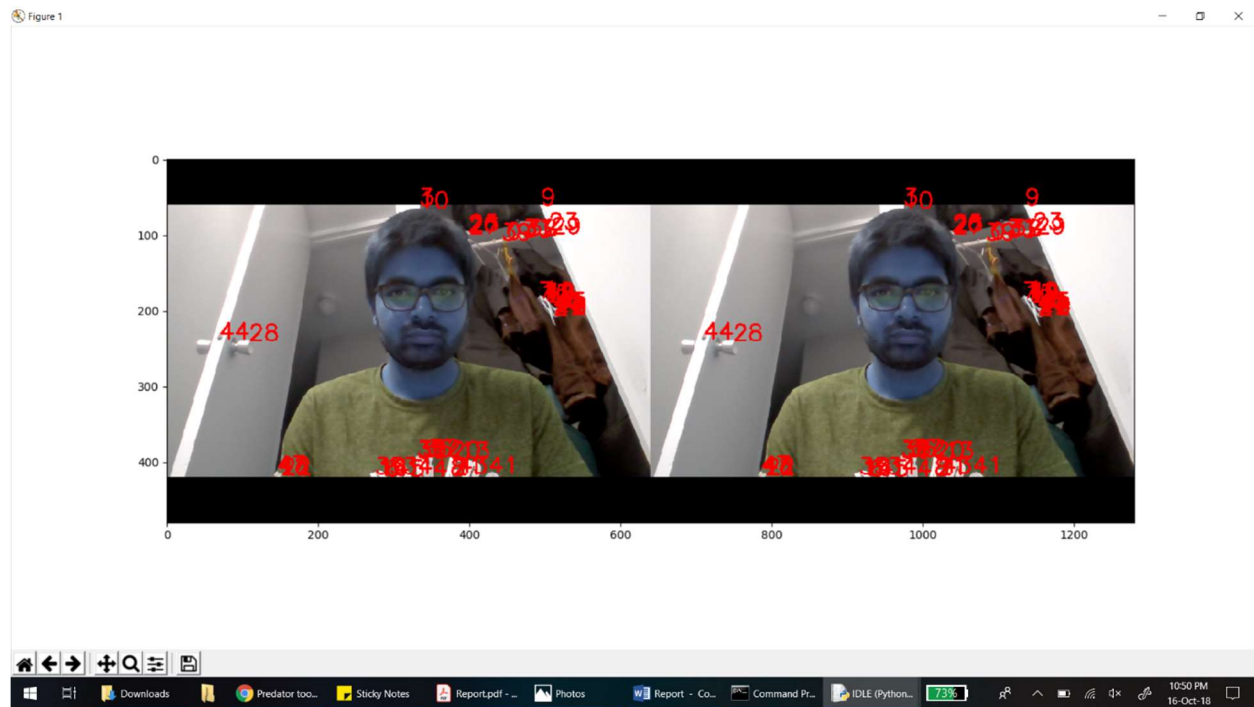
Project gradients onto edge hypothesis and choose  $p$  with minimal projection.

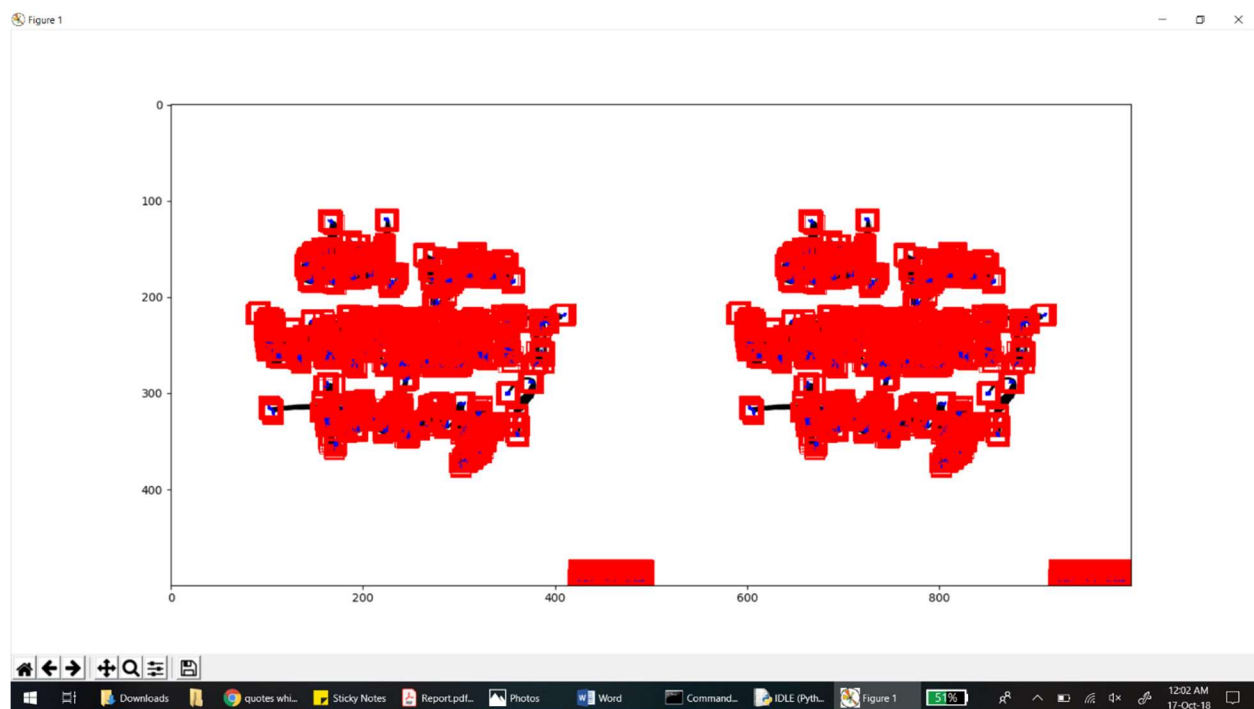
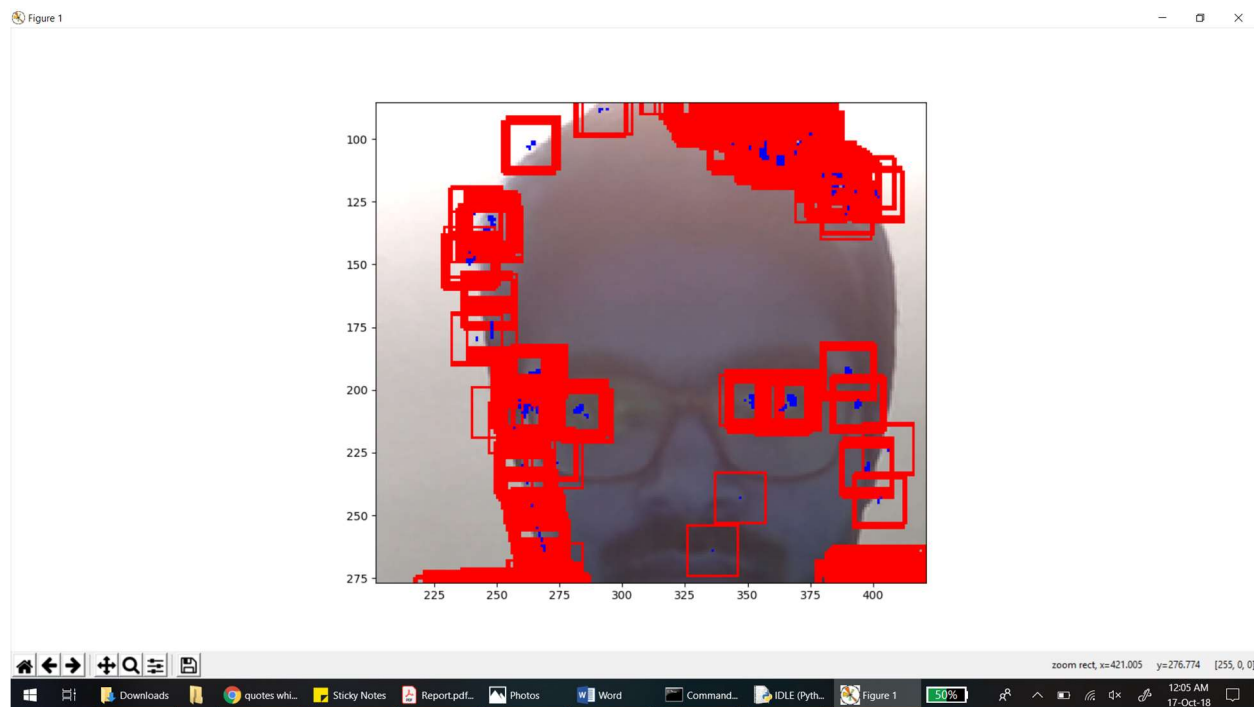
### 3. Implement details :

- Use webcam take two photos also give you two similar pictures.
- Use an isolate display function to display result for every function.
- When take parameter into some function, we need to know the type and change it into right type to compute.
- If we use trackbar to implement the real-time parameter changing we have to change the whole function. So, I choose input parameter and not trackbar.
- Parameters are depending on the pictures you choose; every picture should pick parameter by judge the texture or complexity of picture.
- It's possible to set different threshold at real-time, by store the 'r' and judge it every time you change the threshold. I left it in the code, but use the input way, because I want all the parameter can be changed at same time.
- Press 'H' to get help of this program. Press 'f' to compute a feature vector. Press 'b' to obtain a better localization.
- It required input parameter to get the result of Harris function. The required parameter are variance of gaussian scale, window size, weight of trace, threshold,

## 4. Results and Screenshots :







## **5. Reference:**

Opencv.org

Stackoverflow

Google Search