Computer Vision Assignment – 3 **Name: Ishu Goyal**

Github link: https://github.com/ishu98goyal/Computer-Vision-CSC8830/tree/main/Assignment%203

PART B

Ques 5

Code: GitHub repo → Part B → Ques 5

Steps:

1. Read the target image containing a cluttered scene.
2. Read the reference image containing the object of interest.
3. Detect feature points in both images.
4. Visualize the strongest feature points found in the target image.
5. Visualize the strongest feature points found in the target image.
6. Extract feature descriptors at the interest points in both images.
7. Match the features using their descriptors.
8. Display putatively matched features.
9. Locate the Object in the Scene Using Putative Matches
10. Get the bounding polygon of the reference image.
11. Transform the polygon into the coordinate system of the target image.
12. The transformed polygon indicates the location of the object in the scene.
13. Display the detected object.

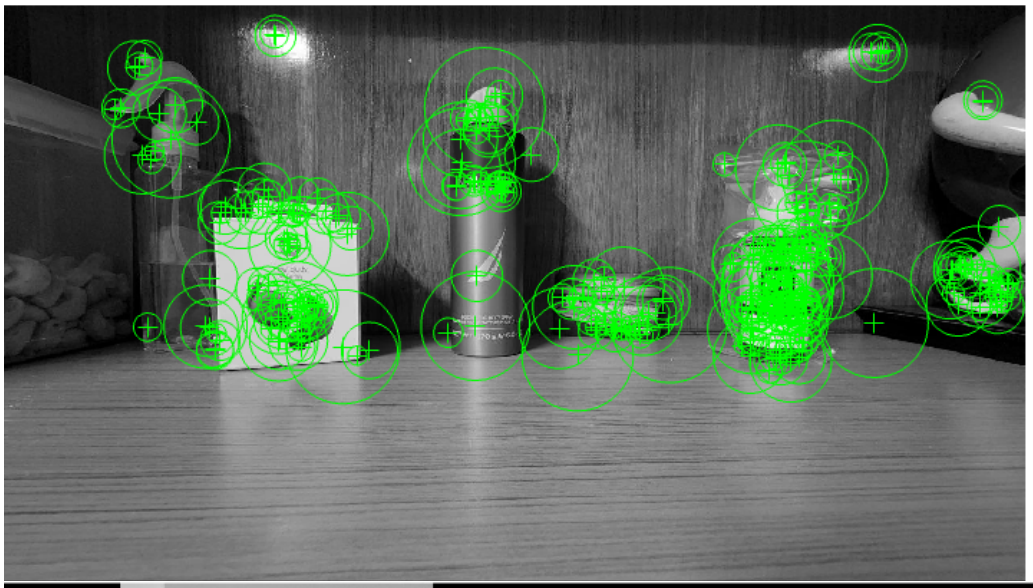
Output:

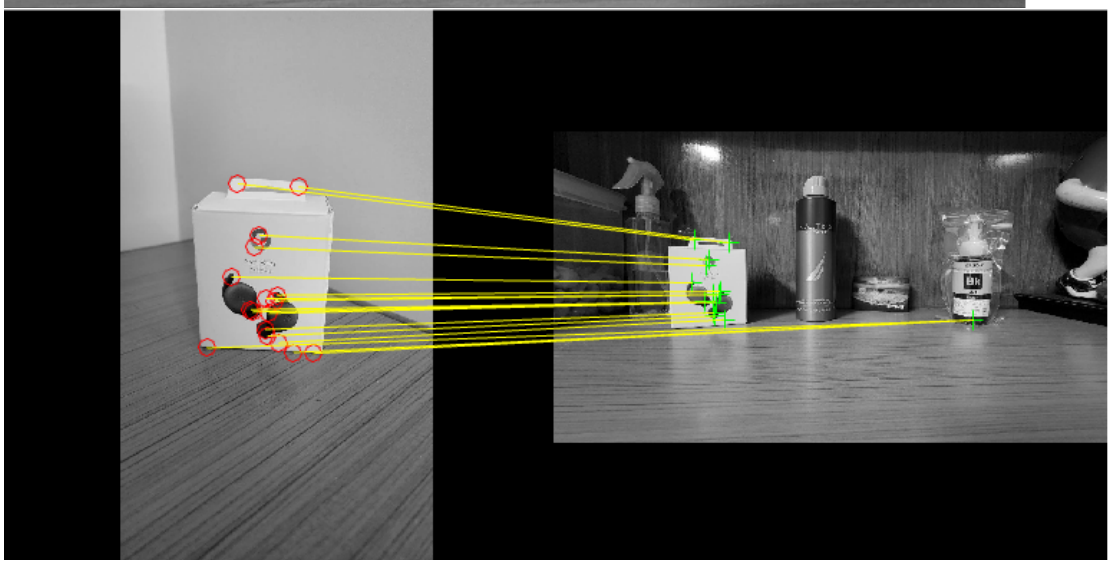
Ques 6
Code: GitHub repo → Part B → Ques 6

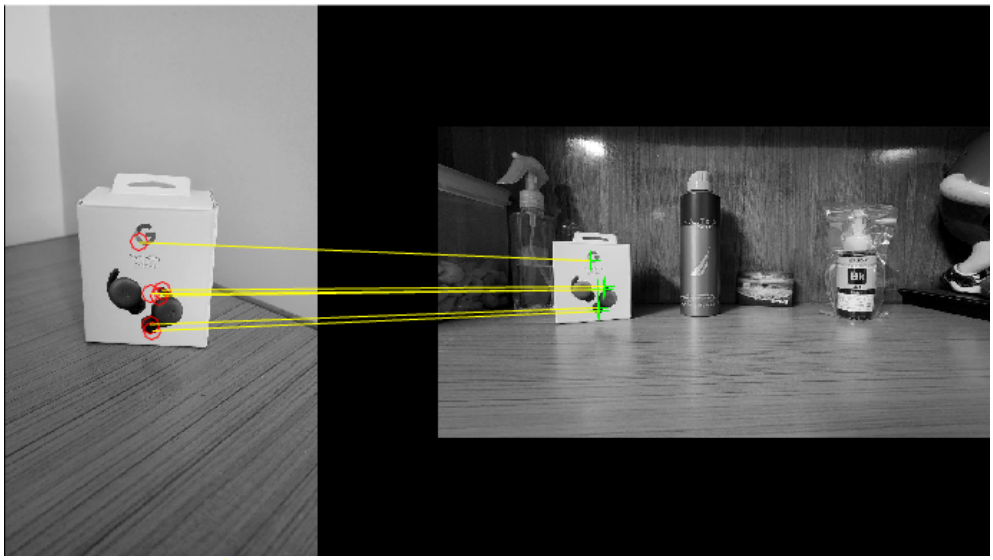
Steps:

1. Read the target image containing a cluttered scene.
2. Read the reference image containing the object of interest.
3. Detect feature points in both images.
4. Visualize the strongest feature points found in the target image.
5. Visualize the strongest feature points found in the target image.
6. Extract feature descriptors at the interest points in both images.
7. Match the features using their descriptors.
8. Locate the Object in the Scene Using Putative Matches
9. Get the bounding polygon of the reference image.
10. Transform the polygon into the coordinate system of the target image.
11. The transformed polygon indicates the location of the object in the scene.
12. Display the detected object.

Output:







Ques 7

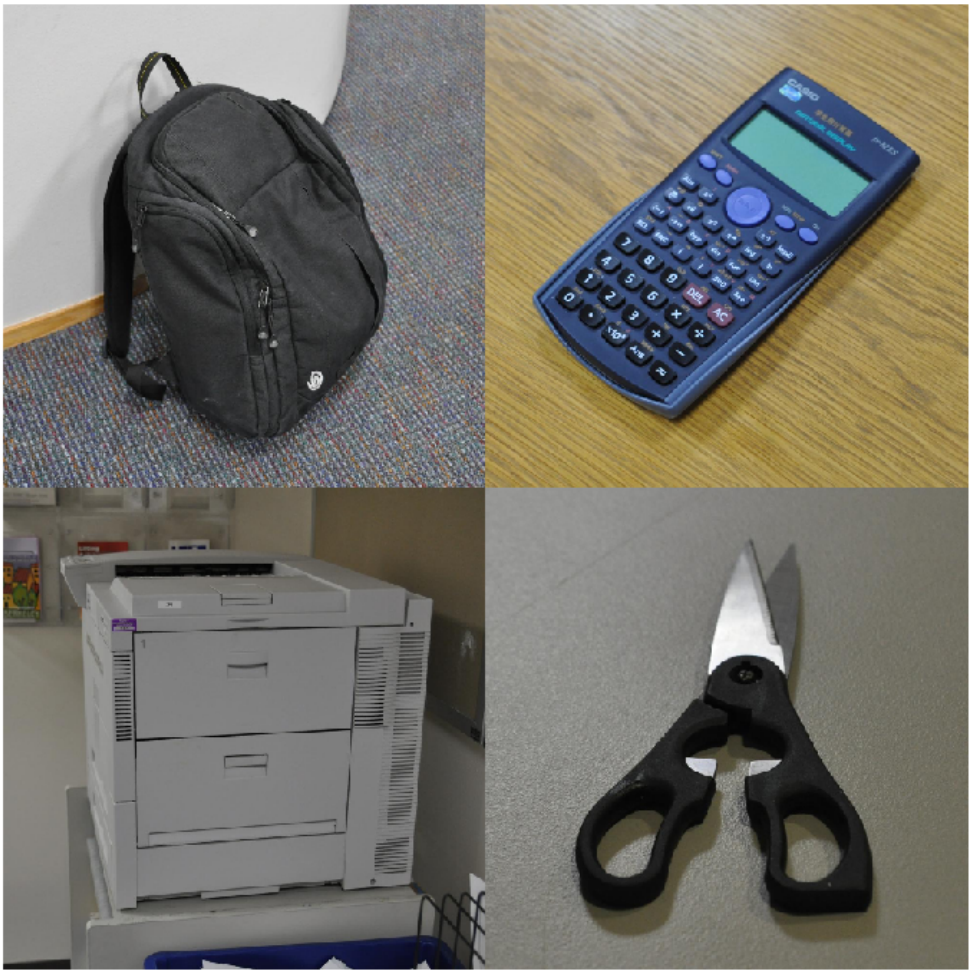
Code: GitHub repo → Part B → Ques 7
Steps:

1. Load Image Dataset
2. inspect the number of images per category, as well as category labels
3. Prepare Training and Validation Image Sets
4. Create a Visual Vocabulary and Train an Image Category Classifier
5. Creating Bag-Of-Features.
6. Encoding images using Bag-Of-Features.
7. Plot the histogram of visual word occurrences
8. Training an image category classifier for 5 categories.
9. Evaluate Classifier Performance on training set and on validation set
10. Run classification

Output:

`tbl = 5×2 table`

	Label	Count
1	backpack	10
2	calculator	10
3	pen	10
4	printer	10
5	scissors	10



```
Creating Bag-Of-Features.
-------------------------
* Image category 1: backpack
* Image category 2: calculator
* Image category 3: pen
* Image category 4: printer
* Image category 5: scissors
* Selecting feature point locations using the Grid method.
* Extracting SURF features from the selected feature point locations.
** The GridStep is [8 8] and the BlockWidth is [32 64 96 128].
```

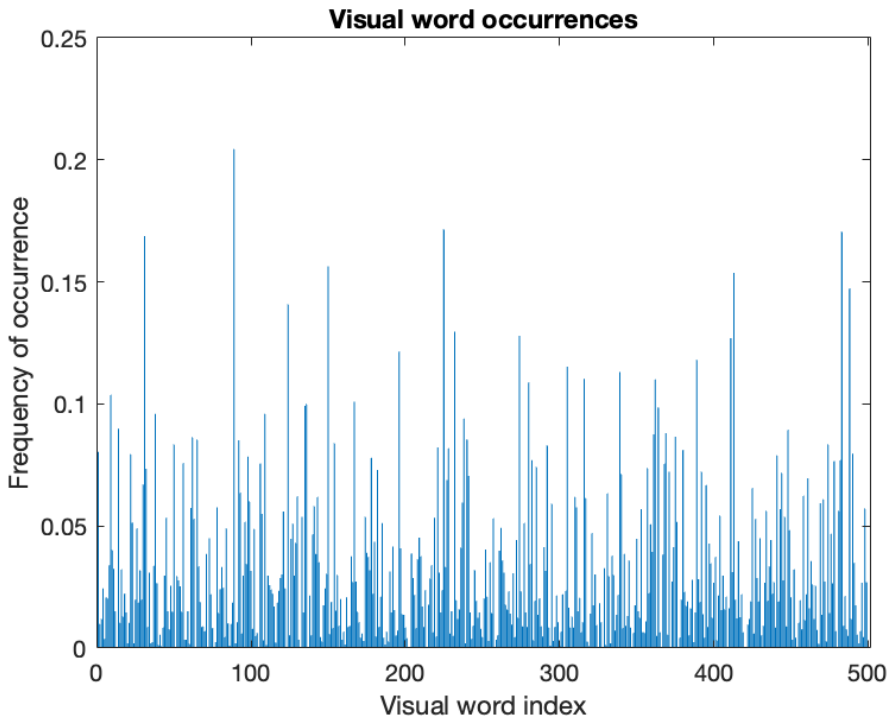
```
* Extracting features from 30 images...done. Extracted 1875000 features.

* Keeping 80 percent of the strongest features from each category.

* Creating a 500 word visual vocabulary.
* Number of levels: 1
* Branching factor: 500
* Number of clustering steps: 1

* [Step 1/1] Clustering vocabulary level 1.
* Number of features          : 1500000
* Number of clusters          : 500
* Initializing cluster centers...100.00%.
* Clustering...completed 42/100 iterations (~4.95
seconds/iteration)...converged in 42 iterations.

* Finished creating Bag-Of-Features
Encoding images using Bag-Of-Features.
--------------------------------------
* Encoding an image...done.
```



PART C

Ques 9
Code: GitHub repo → Part C

Output: