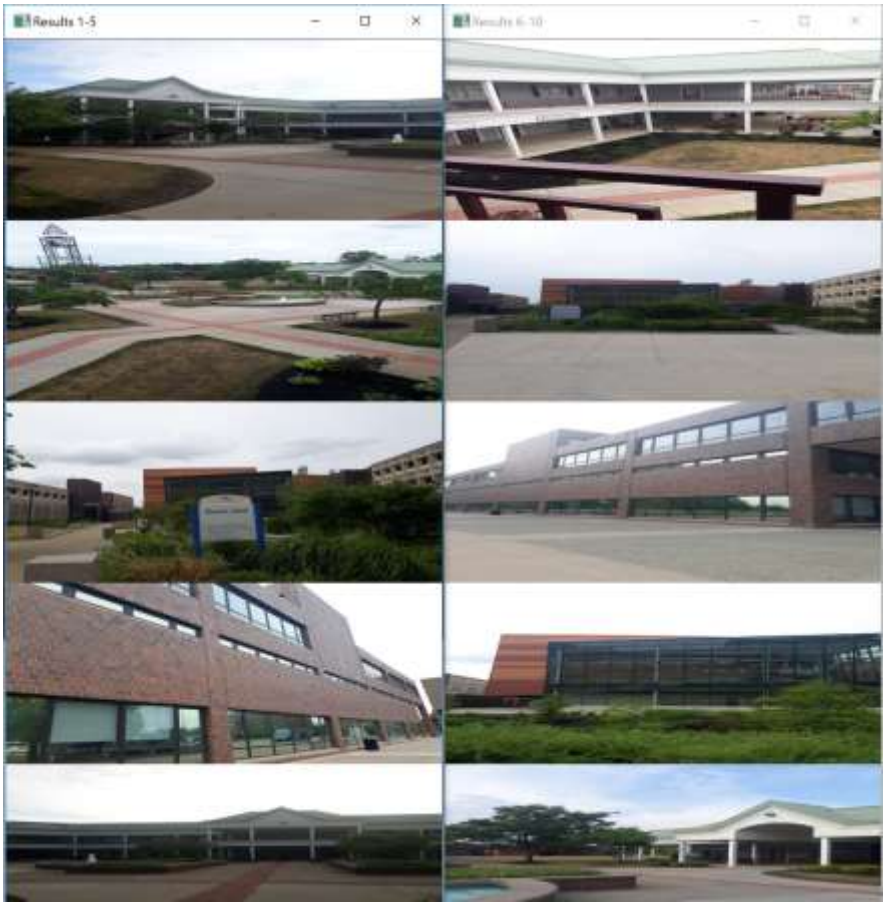
CSE473/573 SUMMER 2018

PROGRAMMING ASSIGNMENT #3

Submitted by

Keerthana Kannan – 50248873

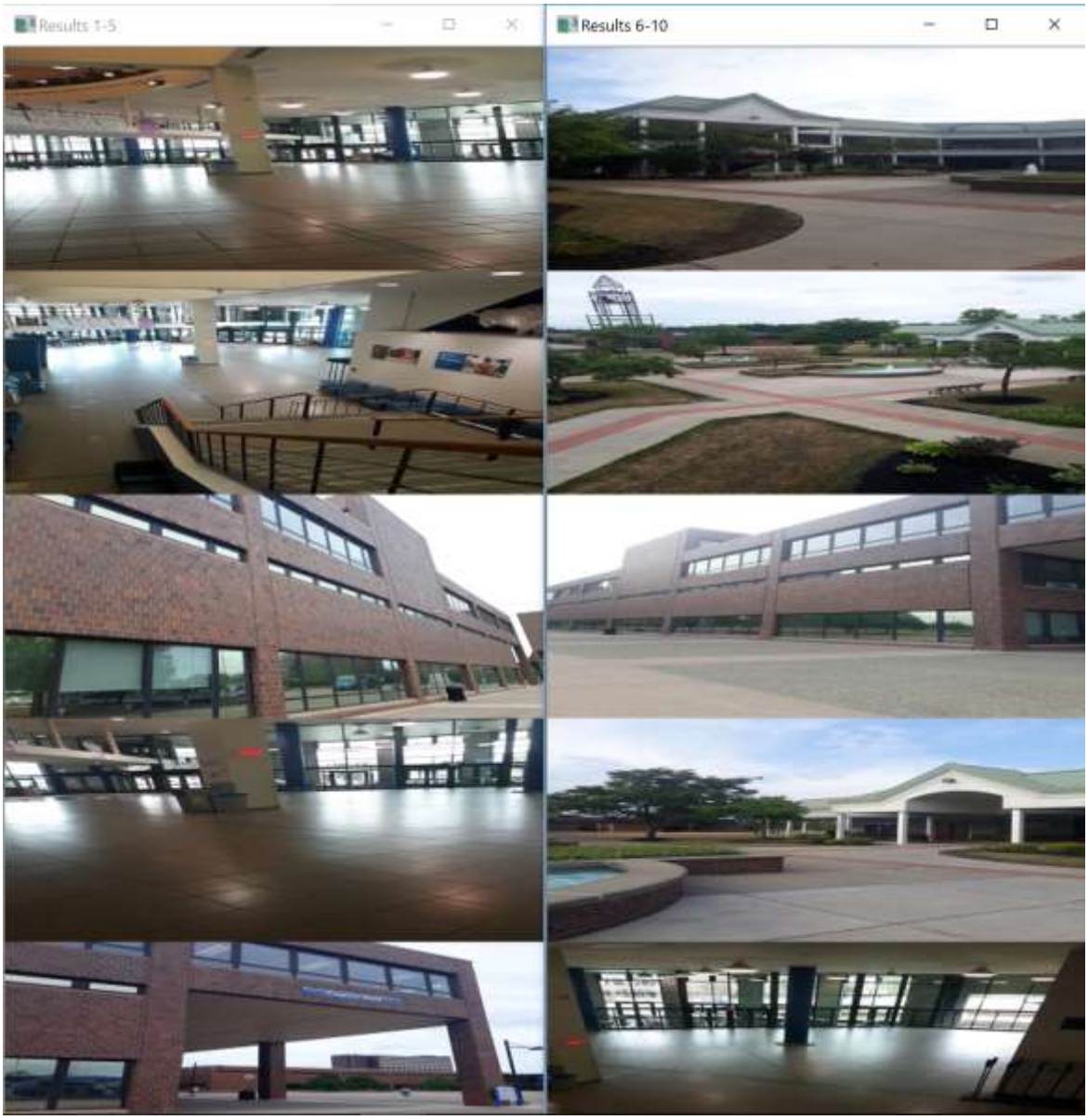
QUERY 1:



```
queries/commons-query.jpg
 1. commons-2.jpg : 0.403
 2. commons-3.jpg : 0.439
 3. davis-5.jpg : 0.644
 4. capen-1.jpg : 0.688
 5. commons-1.jpg : 0.748
 6. commons-4.jpg : 0.814
 7. davis-1.jpg : 0.836
 8. capen-3.jpg : 0.884
 9. davis-4.jpg : 0.896
 10. commons-5.jpg : 0.902
```

The accuracy for query 1 is: 3/5 = 60%

QUERY 2:




```
queries/SU-query.jpg
 1. SU-3.jpg : 0.482
 2. SU-1.jpg : 0.618
 3. capen-1.jpg : 0.846
 4. SU-5.jpg : 0.875
 5. capen-2.jpg : 0.943
 6. commons-2.jpg : 0.970
 7. commons-3.jpg : 1.024
 8. capen-3.jpg : 1.111
 9. commons-5.jpg : 1.136
 10. SU-2.jpg : 1.217
```

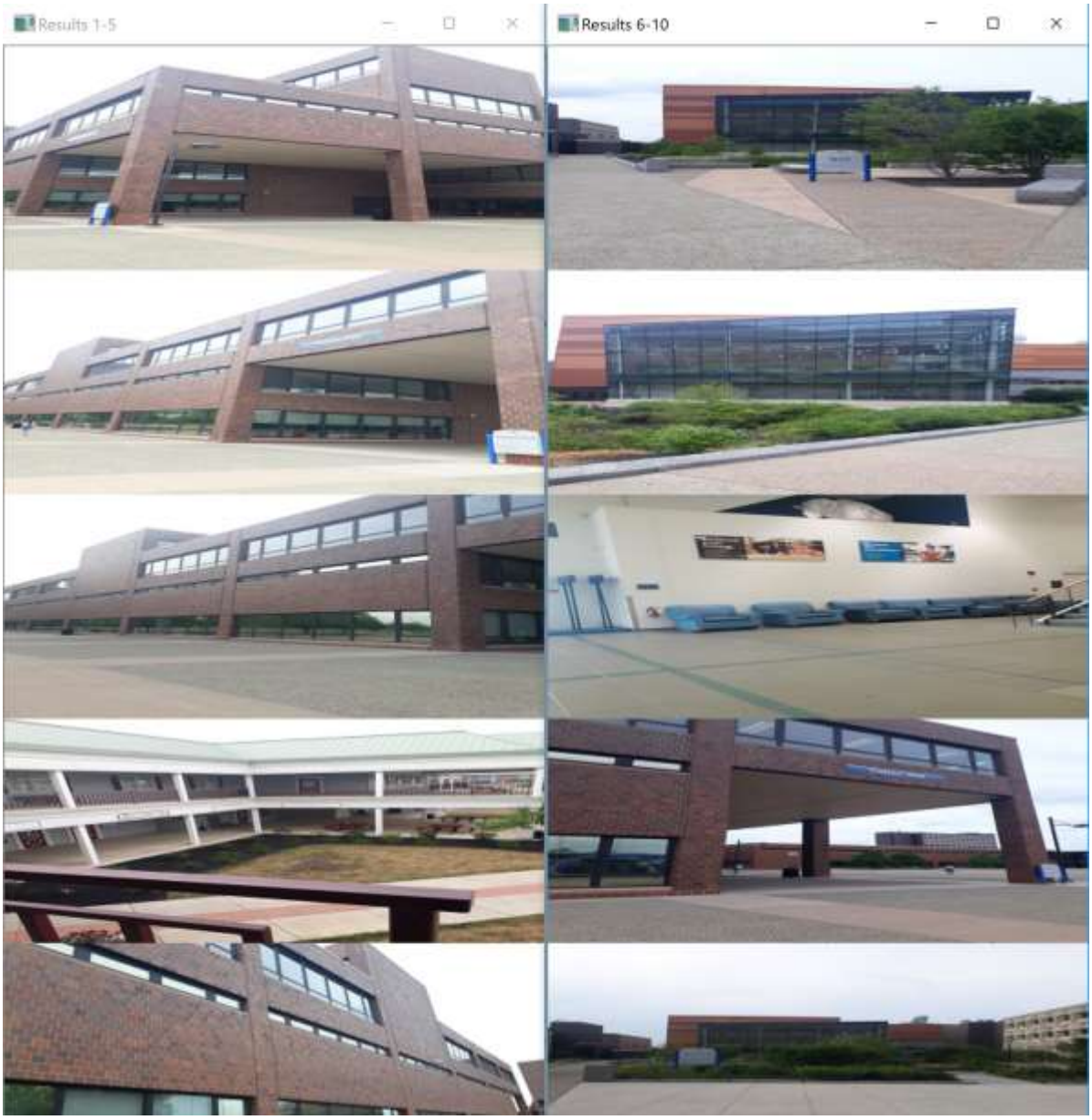
The accuracy for query 2 is: 3/5 = 60%

QUERY 3:

```
queries/lake-query.jpg
 1. lake-3.jpg : 0.714
 2. lake-4.jpg : 1.185
 3. lake-1.jpg : 1.346
 4. SU-2.jpg : 1.538
 5. SU-5.jpg : 1.621
 6. davis-5.jpg : 1.653
 7. lake-2.jpg : 1.682
 8. commons-1.jpg : 1.748
 9. commons-2.jpg : 1.795
 10. commons-3.jpg : 1.874
```

The accuracy for query 3 is: 3/5 = 60%

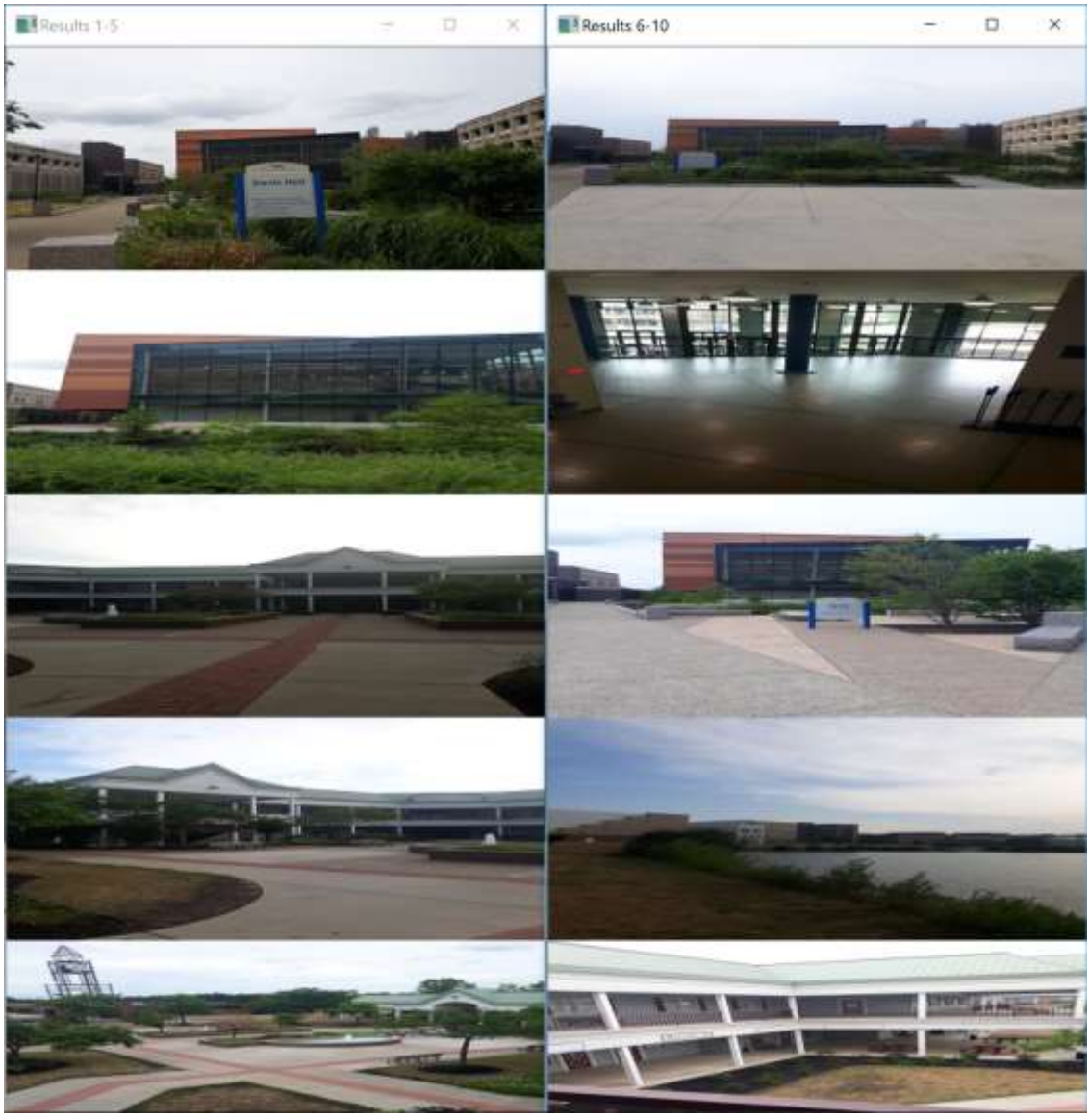
QUERY 4:

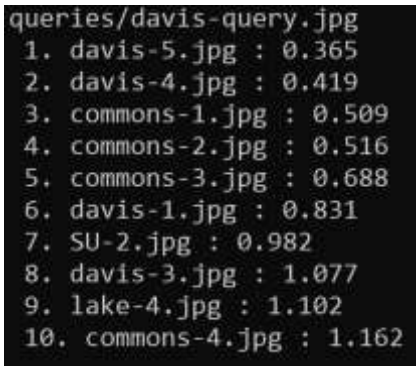



```
queries/capen-query.jpg
 1. capen-5.jpg : 0.335
 2. capen-4.jpg : 0.697
 3. capen-3.jpg : 0.743
 4. commons-4.jpg : 0.858
 5. capen-1.jpg : 0.969
 6. davis-3.jpg : 0.997
 7. davis-2.jpg : 1.053
 8. SU-4.jpg : 1.076
 9. capen-2.jpg : 1.159
 10. davis-1.jpg : 1.247
```

The accuracy for query 4 is: 4/5 = 80%

QUERY 5:





The accuracy for query 5 is: 2/5 = 40%

Overall Accuracy of the search engine:

It is given as the average accuracy of all the queries.

The accuracy = (60+60+60+80+40)/5 = 60%

Performance and improvements:

The runtime of the code was very less and the results were obtained very fast. There was no much memory space was required. But the performance of the system in terms of accuracy was not negligible. The accuracy of the results is not up to the expectations of the user. If the system is accuracy is just 60% for such a small dataset, then it would be even lesser for bigger datasets. Also, some of the relevant images can been seen in the results 6-10 which can be a degrading factor to the performance. Since the system takes only the color distribution in the image, any 2 buildings with same color are considered to be same by the search engine.

The improvements that can be made are:

- Linear color model can be used instead of RGB. This because the RGB histograms are sensitive to noise, brightness change and also to the source of the light.
- Other histograms like color correlogram can be used, along with color distribution it also considers spatial relationship among the pixels.
- RANSAC can be used to make the engine more susceptible to outliers. This is because any 2 image with lot of grass or trees are considered to be similar as their similarity metric is same.

Quality of the code and its organization:

The code was very well structured and well documented with appropriate explanation. There were comments for each line and also the commands for the implementation was clear. This helped in understanding the processes performed during an image search. Since it was developed using the concepts of object oriented programming, the time to execute the program was very less.

The basic steps of the image search were organized as 4 modules of the program:

First is to define the 3D color histogram in the RGB color space. Then second is to index the dataset, i.e. run the histogram on all the images in the dataset and store the values in a pickle file. Then third is to find the similarity metric between query image and the indexed dataset. Here, ci-squared distance metric is used. The final program was to search for the relevant images from the obtained similarity metric.

Take away from this assignment:

The assignment helped in understanding the process of visual search in general and also helped in understanding its implementation in the real world i.e. in large scale. Also it helped in understanding the loss of image quality and information loss due to resizing the image to 400x166. The drawbacks of using RGB histogram was also learnt and the improvisation on this was studied. The code helped in realizing that the common factors in images like sky, grass, tree also play a major role in the search as they give same similarity matrix in all the cases.