CS 5330 - Pattern Recognition and Computer Vision

Project 2 : Content-based Image Retrieval

Group Members : Shirish Kishore Kumar (002980081), Soorya Vijayaragavan (002921909)

Introduction:

The main aim of this project is to analyze and understand the images at pixel level and use it to perform matching and pattern recognition tasks. This is done by matching a target image with a database of images based on its characteristics like color distributions, texture and histogram. Then, the feature vectors of both target image and the database of images are generated based on different characteristics listed above for various tasks. Finally, the distance metric is calculated between the feature vectors of the target image and the database of images and the results are listed, as the database image that has the least distance with the target image is more similar.

Tasks:

1. **Baseline Matching:**
   For this task we used the center 9*9 pixels as the feature points and used the sum of squared difference as the distance metric. The first image is the target image and the other three images are images retrieved arranged in increasing order of distance from the target.




```
You have entered 3 arguments:
./main
./olympus
./olympus/pic.1016.jpg
<-------------------------------------------------------------------------------------------------------------------------->

Enter the type of feature extraction process you want
A : Center 50*50 as a first Histogram and whole image RGB as second histogram
B : Multi Histogram Matching
C : Texture and color
D : Extension (3 Histogram)
E : Baseline Matching
F : Histogram Matching
e
Extracting the features :
<------------------------------Done--------------------------->
Writing to CSV...
Creating the CSV file Completed ...
Enter the Number of closest image you want
4


1.              :       pic.1016.jpg
2.              :       pic.0986.jpg
3.              :       pic.0641.jpg
4.              :       pic.0233.jpg
(base) warrior@pop-os:~/CV_ASSIGNMENTS/cbir-cv-assignmnet-2$
```

2. **Histogram Matching:**

   For this task we created a RGB 3d histogram with 8 bins and used histogram intersection as the distance metric. The first image is the target image and the other three images are images retrieved arranged in increasing order of distance from the target.

```
You have entered 3 arguments:
./main
./olympus
./olympus/pic.0164.jpg
<-------------------------------------------------------------------------------------------------------------------->

Enter the type of feature extraction process you want
A : Center 50*50 as a first Histogram and whole image RGB as second histogram
B : Multi Histogram Matching
C : Texture and color
D : Extension (3 Histogram)
E : Baseline Matching
F : Histogram Matching
f
Extracting the features :
<------------------------------Done-------------------------->
Writing to CSV...
Creating the CSV file Completed ...
Enter the Number of closest image you want
4


1.              :       pic.0164.jpg
2.              :       pic.0110.jpg
3.              :       pic.1032.jpg
4.              :       pic.0092.jpg
(base) warrior@pop-os:~/CV_ASSIGNMENTS/cbir-cv-assignmnet-2$
```

3. **Multi-histogram Matching:**

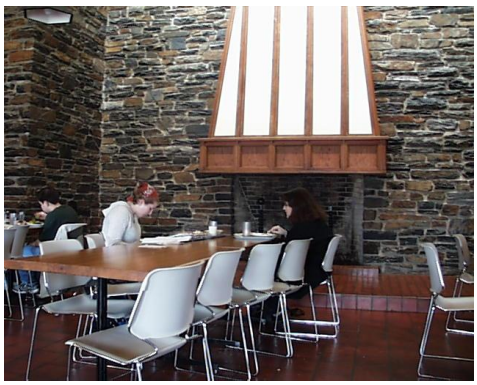
   For this task we created a two histogram where the first histogram represents the **top half** of the image and the second histogram represents the **bottom half** of the image.
   Used 3d Histogram intersection as the distance metric.
   The first image is the target image and the other three images are images retrieved arranged in increasing order of distance from the target.



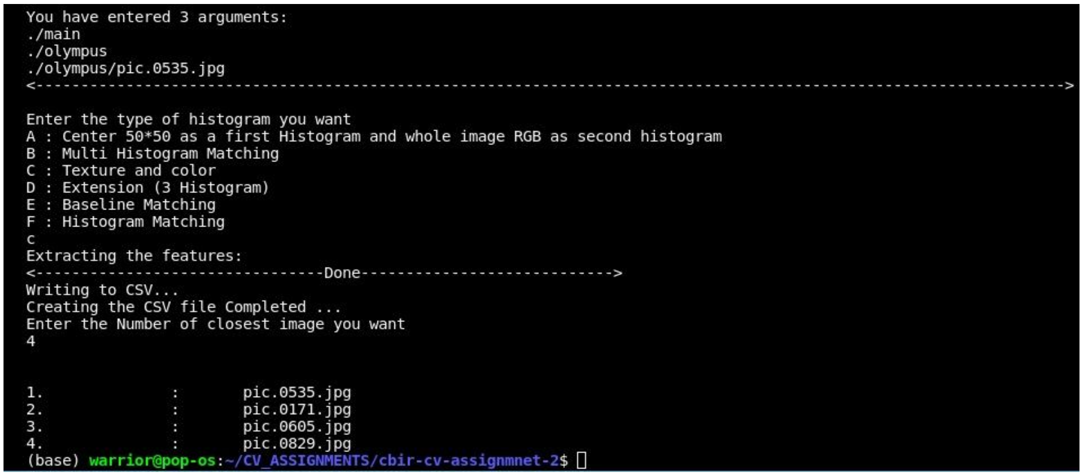

```
You have entered 3 arguments:
./main
./olympus
./olympus/pic.0274.jpg
<-------------------------------------------------------------------------------------------------------------->

Enter the type of feature extraction process you want
A : Center 50*50 as a first Histogram and whole image RGB as second histogram
B : Multi Histogram Matching
C : Texture and color
D : Extension (3 Histogram)
E : Baseline Matching
F : Histogram Matching
b
Extracting the features:
<-------------------------------Done--------------------------->
Writing to CSV...
Creating the CSV file Completed ...
Enter the Number of closest image you want
4


1.              :       pic.0274.jpg
2.              :       pic.0273.jpg
3.              :       pic.1031.jpg
4.              :       pic.0409.jpg
(base) warrior@pop-os:~/CV_ASSIGNMENTS/cbir-cv-assignmnet-2$
```

4. Texture and Color

For this task we created two histograms where the first histogram represents the RGB of the whole image and the second histogram represents the gradient magnitude of the whole image. Then we used the histogram intersection with equal weights as the distance metric. The first image is the target image and the other three images are images retrieved arranged in increasing order of distance from the target.



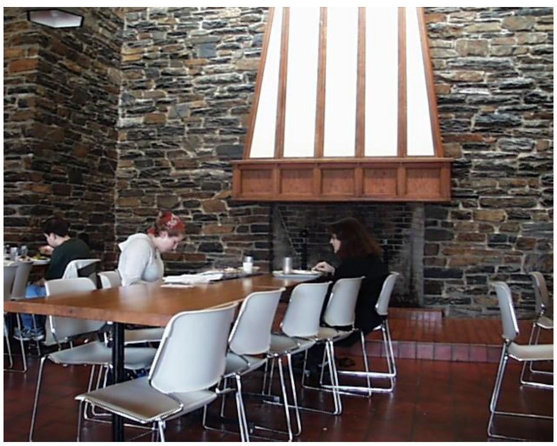


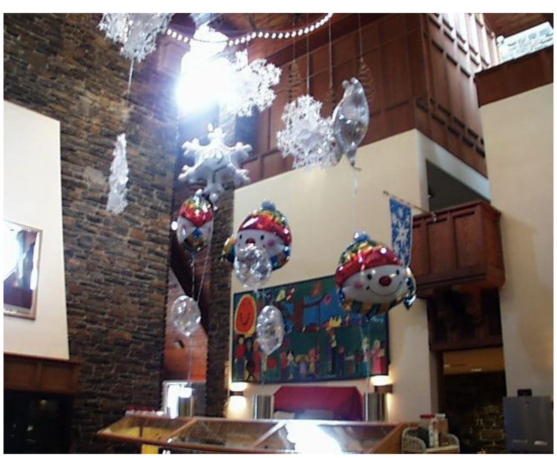


```
You have entered 3 arguments:
./main
./olympus
./olympus/pic.0535.jpg
<---------------------------------------------------------------------------------------------------->

Enter the type of histogram you want
A : Center 50*50 as a first Histogram and whole image RGB as second histogram
B : Multi Histogram Matching
C : Texture and color
D : Extension (3 Histogram)
E : Baseline Matching
F : Histogram Matching
c
Extracting the features:
<-------------------------------Done--------------------------->
Writing to CSV...
Creating the CSV file Completed ...
Enter the Number of closest image you want
4


1.              :       pic.0535.jpg
2.              :       pic.0171.jpg
3.              :       pic.0605.jpg
4.              :       pic.0829.jpg
(base) warrior@pop-os:~/CV_ASSIGNMENTS/cbir-cv-assignmnet-2$
```

Implementing task 2 using pic.0535,

The first image is the target image and the other three images are images retrieved arranged in increasing order of distance from the target.






```
You have entered 3 arguments:
./main
./olympus
./olympus/pic.0535.jpg
<------------------------------------------------------------------------------------------------------------------>

Enter the type of histogram you want
A : Center 50*50 as a first Histogram and whole image RGB as second histogram
B : Multi Histogram Matching
C : Texture and color
D : Extension (3 Histogram)
E : Baseline Matching
F : Histogram Matching
f
Extracting the features :
<--------------------------------Done--------------------------->
Writing to CSV...
Creating the CSV file Completed ...
Enter the Number of closest image you want
4


1.              :       pic.0535.jpg
2.              :       pic.0285.jpg
3.              :       pic.0628.jpg
4.              :       pic.0952.jpg
(base) warrior@pop-os:~/CV_ASSIGNMENTS/cbir-cv-assignmnet-2$ ./main ./olympus ./olympus/pic.0164.jpg
```

Implementing task 3 using pic.0535,

The first image is the target image and the other three images are images retrieved arranged in increasing order of distance from the target.

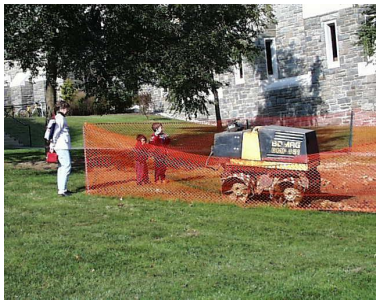
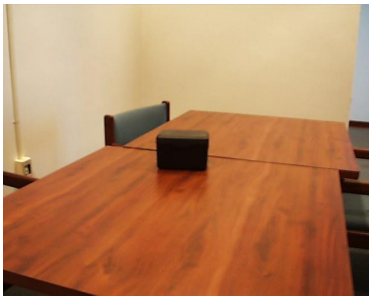
```
You have entered 3 arguments:
./main
./olympus
./olympus/pic.0535.jpg
<-------------------------------------------------------------------------------------------------------------------->

Enter the type of histogram you want
A : Center 50*50 as a first Histogram and whole image RGB as second histogram
B : Multi Histogram Matching
C : Texture and color
D : Extension (3 Histogram)
E : Baseline Matching
F : Histogram Matching
b
Extracting the features:
<--------------------------------Done--------------------------->
Writing to CSV...
Creating the CSV file Completed ...
Enter the Number of closest image you want
4


1.              :       pic.0535.jpg
2.              :       pic.0285.jpg
3.              :       pic.0628.jpg
4.              :       pic.0952.jpg
(base) warrior@pop-os:~/CV_ASSIGNMENTS/cbir-cv-assignmnet-2$
```

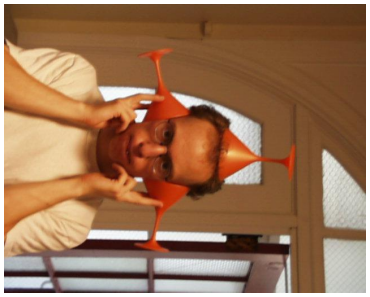
5. **Custom Design:**

   For this custom design we used a two 3d histograms where the first histogram represents the RGB of the whole image and the second histogram contains the 2500 pixels from the center and uses those rgb values to obtain a 8 bin histogram and gave 70 percent weightage to the center histogram, as we can see that it found all the masks from the olympus data set.

Image Used : **pic.0326**

The first image is the target image and the other images are images retrieved arranged in increasing order of distance from the target.

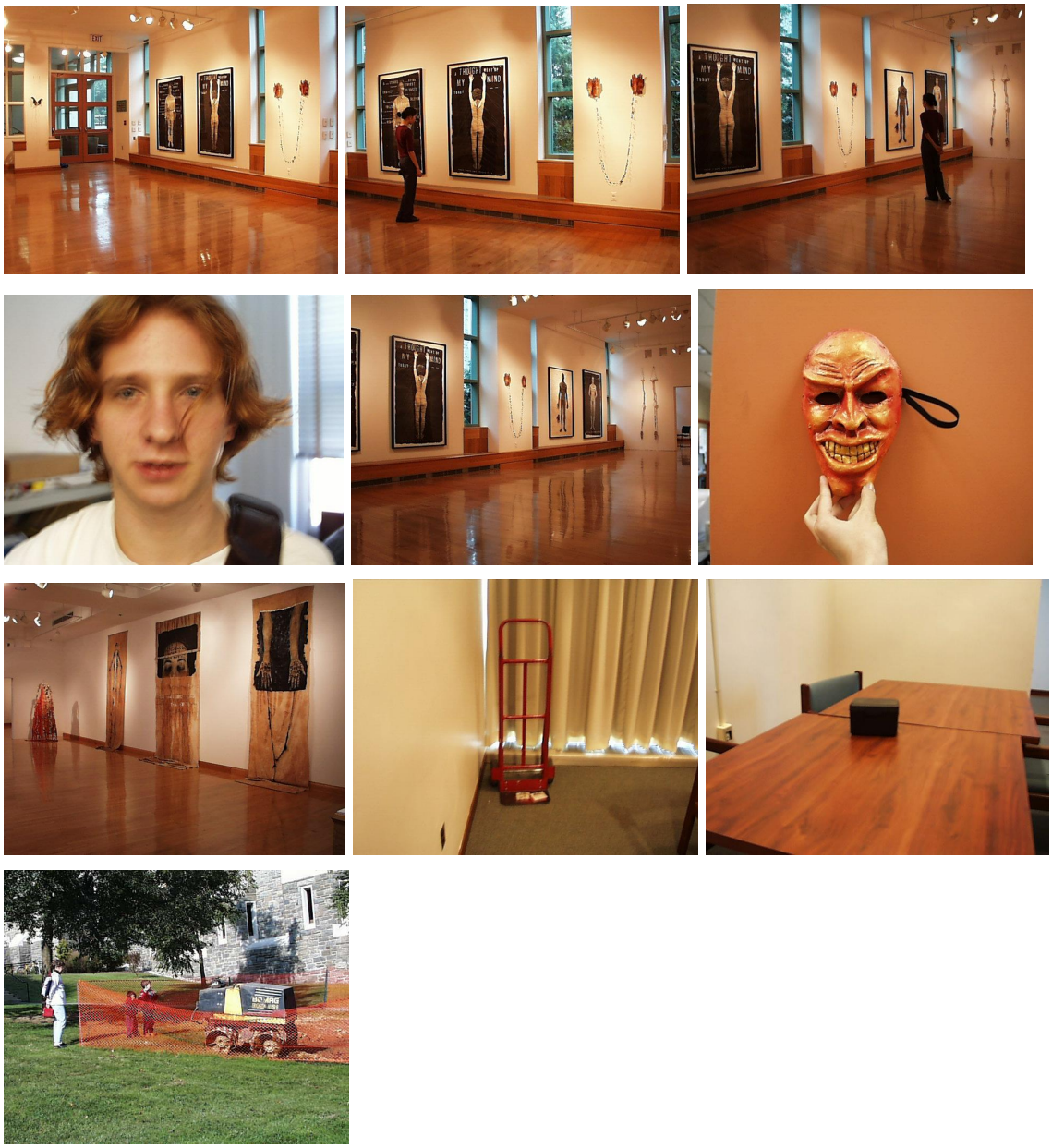





```
You have entered 3 arguments:
./main
./olympus
./olympus/pic.0326.jpg
<-------------------------------------------------------------------------------------------------------------------------->

Enter the type of histogram you want
A : Center 50*50 as a first Histogram and whole image RGB as second histogram
B : Multi Histogram Matching
C : Texture and color
D : Extension (3 Histogram)
E : Baseline Matching
F : Histogram Matching
a
Extracting the features:
<--------------------------------Done--------------------------->
Writing to CSV...
Creating the CSV file Completed ...
Enter the Number of closest image you want
10


1.              :       pic.0326.jpg
2.              :       pic.0323.jpg
3.              :       pic.0327.jpg
4.              :       pic.0324.jpg
5.              :       pic.0016.jpg
6.              :       pic.0325.jpg
7.              :       pic.0875.jpg
8.              :       pic.0377.jpg
9.              :       pic.0379.jpg
10.             :       pic.0183.jpg
(base) warrior@pop-os:~/CV_ASSIGNMENTS/cbir-cv-assignmnet-2$
```

Image Used: **pic.0016**

The first image is the target image and the other images are images retrieved arranged in increasing order of distance from the target.



We implemented the same program for the other image, as we can see that it did a great job in finding the closest image.

```
You have entered 3 arguments:
./main
./olympus
./olympus/pic.0016.jpg
<-------------------------------------------------------------------------------------------------------------------------------->

Enter the type of histogram you want
A : Center 50*50 as a first Histogram and whole image RGB as second histogram
B : Multi Histogram Matching
C : Texture and color
D : Extension (3 Histogram)
E : Baseline Matching
F : Histogram Matching
a
Extracting the features:
<--------------------------------Done---------------------------->
Writing to CSV...
Creating the CSV file Completed ...
Enter the Number of closest image you want
10


1.              :       pic.0016.jpg
2.              :       pic.0020.jpg
3.              :       pic.0017.jpg
4.              :       pic.0088.jpg
5.              :       pic.0014.jpg
6.              :       pic.0326.jpg
7.              :       pic.0013.jpg
8.              :       pic.0379.jpg
9.              :       pic.0377.jpg
10.             :       pic.0875.jpg
(base) warrior@pop-os:~/CV_ASSIGNMENTS/cbir-cv-assignmnet-2$
```

Extension:

Image Used: **pic.0537**

The first image is the target image and the 10 images are images retrieved arranged in increasing order of distance from the target.

For the extension part we created 3 3d histogram where the first histogram represents the RGB of the whole image, second histogram shows represents the gradient orientation and the third represents the center 2500 pixels of rgb image, we can see that it found the target image guy in the top ten search results and also from from the whole image dataset. When compared to the previous method this method produced excellent results since it considered three main features from the image.

```
You have entered 3 arguments:
./main
./olympus
./olympus/pic.0537.jpg
<------------------------------------------------------------------------------------------------------------------------>

Enter the type of histogram you want
A : Center 50*50 as a first Histogram and whole image RGB as second histogram
B : Multi Histogram Matching
C : Texture and color
D : Extension (3 Histogram)
E : Baseline Matching
F : Histogram Matching
d
Extracting the features:
<-------------------------------Done--------------------------->
Writing to CSV...
Creating the CSV file Completed ...
Enter the Number of closest image you want
10


1.              :       pic.0537.jpg
2.              :       pic.0538.jpg
3.              :       pic.0536.jpg
4.              :       pic.0551.jpg
5.              :       pic.0550.jpg
6.              :       pic.0533.jpg
7.              :       pic.0546.jpg
8.              :       pic.0554.jpg
9.              :       pic.0846.jpg
10.             :       pic.0980.jpg
```

Learnings:

In this project we learned about feature extraction methods using histograms and matching methods using sum of squared difference and histogram intersection. In addition to expanding my knowledge of histograms in different dimensions and color spaces, this project helped me develop my intuition for extracting feature representations from images in order to cluster and categorize the image. Learning about various distance metrics and how different feature vectors are weighted differently during matching was another important learning from this project.

References:

- https://stackoverflow.com/questions/26499307/transfer-a-2d-array-to-a-csv-file
- https://stackoverflow.com/questions/49444856/transpose-of-a-2d-matrix-using-vectors-in-c
- https://docs.opencv.org/3.4/files.html
- http://opencv-tutorials-hub.blogspot.com/