Project-2 : Content Based Image Retrieval

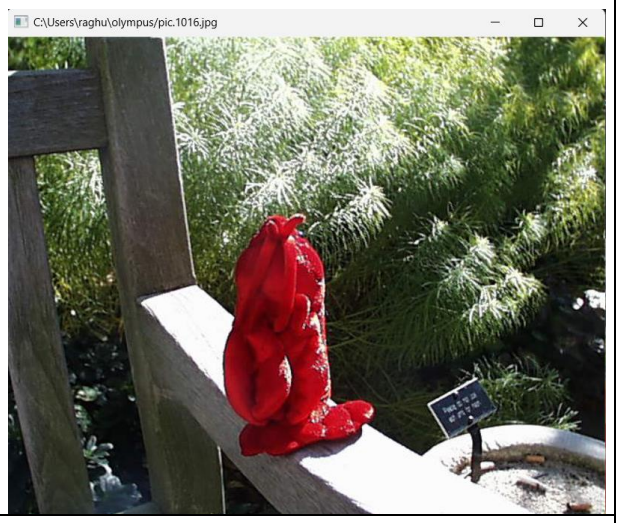
By: Sriram Pandi & Srinivas Peri

Task 1: Baseline Matching

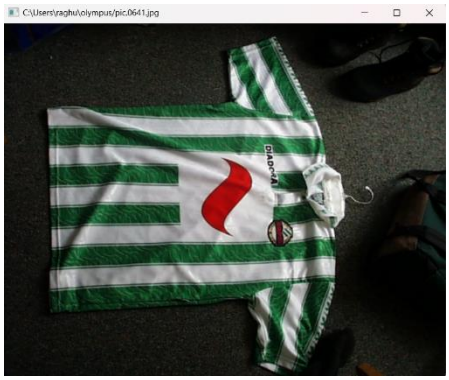
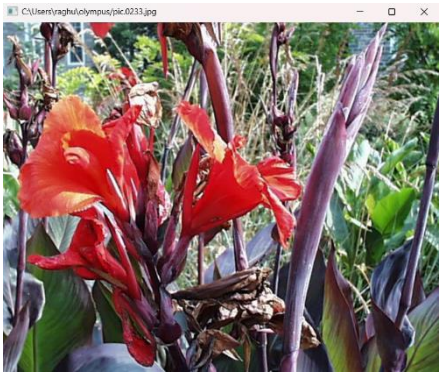
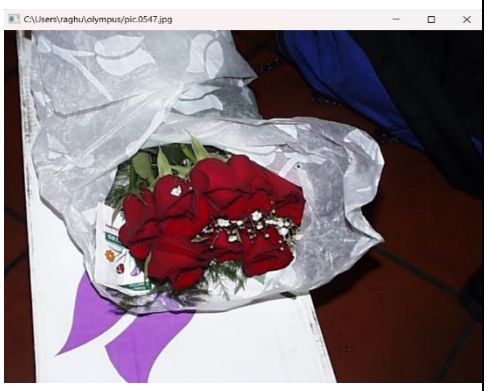
C:\Users\raghu\olympus/pic.1032.jpg	**Image path**
0.662848	**Sum of squared distances value of the closest image**

Requires result-1: *The top three matches for the target image pic.1016.jpg.*

Input Image:

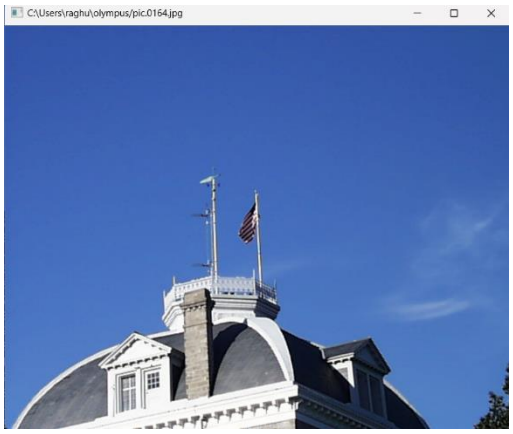


C:\Users\raghu\olympus/pic.1016.jpg 0

		
C:\Users\raghu\olympus/pic.0641.jpg 10170	C:\Users\raghu\olympus/pic.0233.jpg 12600	C:\Users\raghu\olympus/pic.0547.jpg 15660

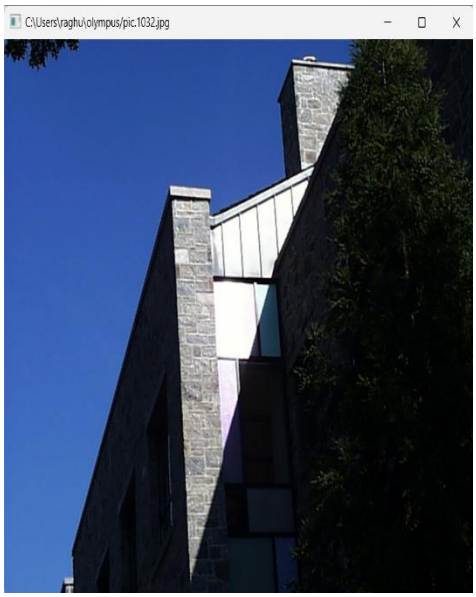
Task 2: Histogram Matching

Required results: *The top three matches for the target image pic.0164.jpg.*



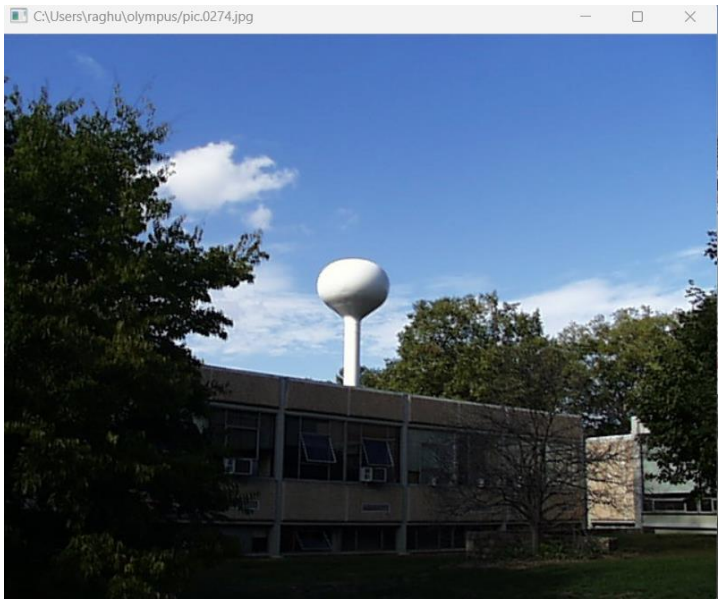
Input Image

C:\Users\raghu\olympus/pic.0164.jpg 9.99201e-16

		
C:\Users\raghu\olympus/pic.0110.jpg 0.611465	C:\Users\raghu\olympus/pic.0092.jpg 0.639352	C:\Users\raghu\olympus/pic.1032.jpg 0.662848

Task 3: Multi-histogram Matching

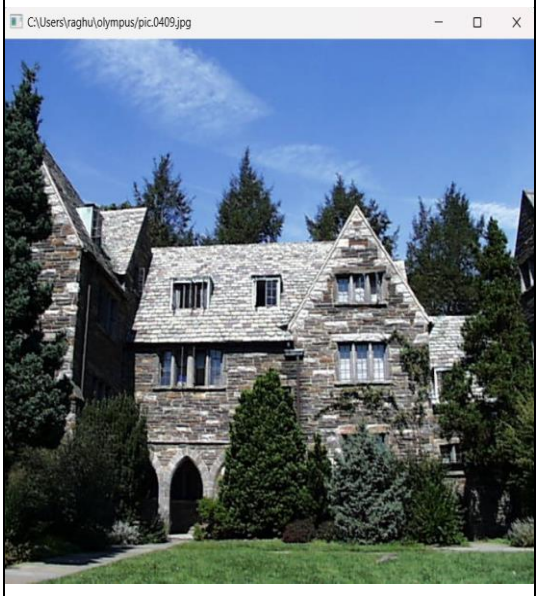
Required results: *The top three matches for the target image pic.0274.jpg.*



Input Image:

C:\Users\raghu\olympus/pic.0274.jpg 0

C:\Users\raghu\olympus/pic.0881.jpg 0.52002



C:\Users\raghu\olympus/pic.0409.jpg 0.549744

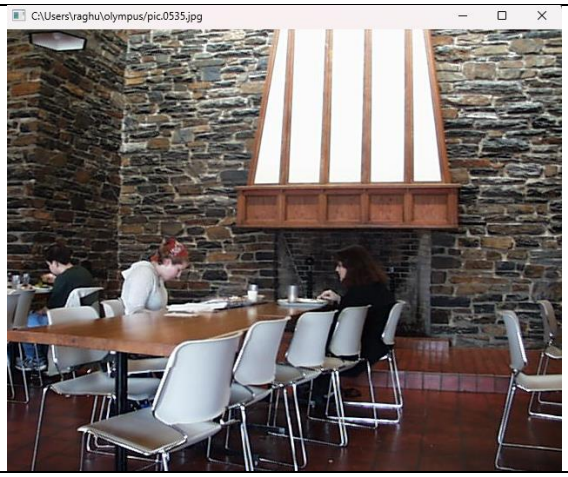
C:\Users\raghu\olympus/pic.1055.jpg 0.504388

This task is accomplished by splitting the image into left and right halves and comparing the hist of each value with the split images in folder and the results came out as shown above.

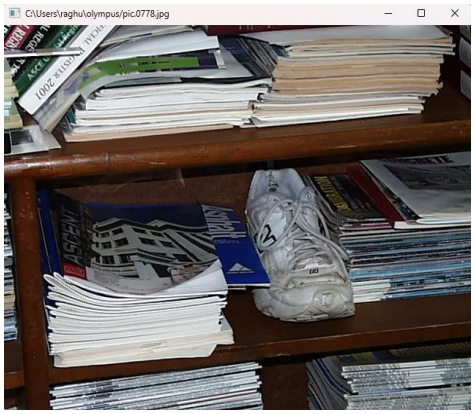
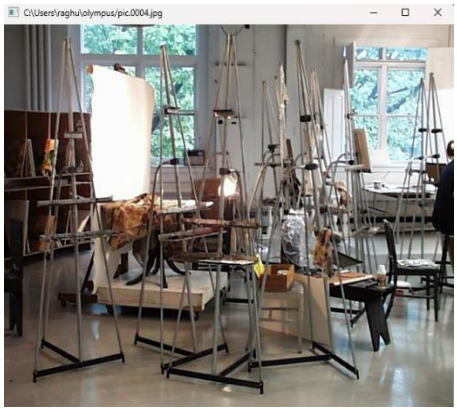
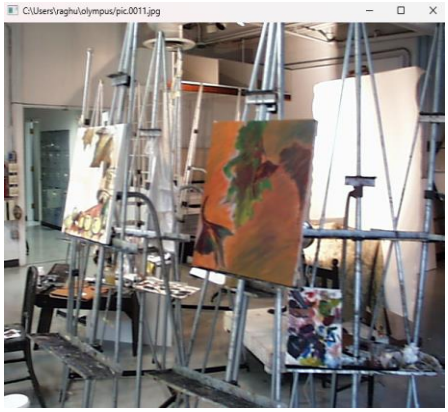
Task 4: Texture and Color

Required results 4:

1. *The top three matches for the target image pic.0535.jpg*

2. 

C:\Users\raghu\olympus/pic.0535.jpg 7.43849e-15

		
C:\Users\raghu\olympus/pic.0778.jpg 0.0852203	C:\Users\raghu\olympus/pic.0004.jpg 0.105835	C:\Users\raghu\olympus/pic.0011.jpg 0.124023

2. *How they differ when compared to tasks 2 and 3.*

The task 4 uses Sobel filter to extract textures, followed by finding the gradient magnitude. Then the histogram is calculated and converted in to floating point vector. The query image is then compared with all the images in the data set using the intersection distance metric. Where-as in the task2 and task3 for feature vector was obtained by calculating the histograms of either the total image (single histogram) or by considering multiple portions of the image and then finding the histograms(multi-histogram).

Task 5: Custom Design :

Required results 5: For two target images of your choice, show the top ten results. It's also helpful to show some of the least similar results.

For this task we have taken a subset of the Olympus data as training dataset with around 80 images.

The following are the query images from the training dataset :

	
Target-1	**Target-2**
"C:\Users\18573\Desktop\data_set\pic.0013.jpg"	"C:\Users\18573\Desktop\data_set\pic.0016.jpg"

Output:

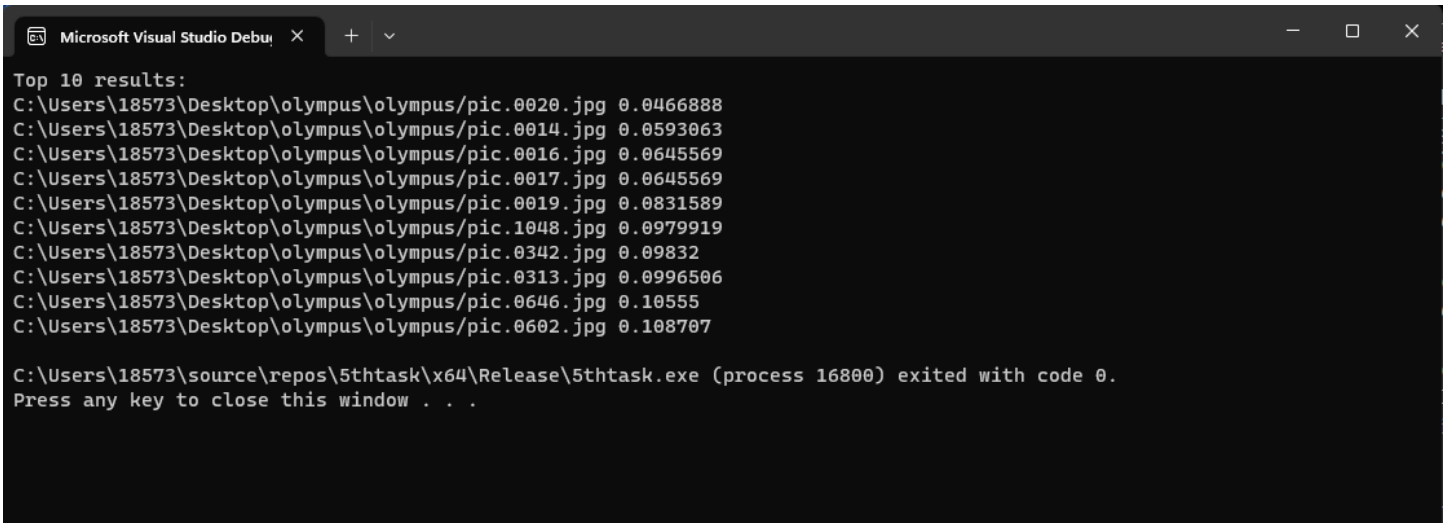
The following are the output 6 images for the query images:

```
Microsoft Visual Studio Debu
Top 6 results:
C:\Users\18573\Desktop\data_set/pic.0020.jpg 0.0466888
C:\Users\18573\Desktop\data_set/pic.0014.jpg 0.0593063
C:\Users\18573\Desktop\data_set/pic.0016.jpg 0.0645569
C:\Users\18573\Desktop\data_set/pic.0017.jpg 0.0645569
C:\Users\18573\Desktop\data_set/pic.0019.jpg 0.0831589
C:\Users\18573\Desktop\data_set/pic.0025.jpg 0.163895

C:\Users\18573\source\repos\5thtask\x64\Release\5thtask.exe (process 14872) exited with code 0.
Press any key to close this window . . .
```

When executed on the total Olympus dataset for the same input query images the outputs are as follows:

```
Microsoft Visual Studio Debu
Top 6 results:
C:\Users\18573\Desktop\olympus\olympus/pic.0020.jpg 0.0466888
C:\Users\18573\Desktop\olympus\olympus/pic.0014.jpg 0.0593063
C:\Users\18573\Desktop\olympus\olympus/pic.0016.jpg 0.0645569
C:\Users\18573\Desktop\olympus\olympus/pic.0017.jpg 0.0645569
C:\Users\18573\Desktop\olympus\olympus/pic.0019.jpg 0.0831589
C:\Users\18573\Desktop\olympus\olympus/pic.1048.jpg 0.0979919

C:\Users\18573\source\repos\5thtask\x64\Release\5thtask.exe (process 25580) exited with code 0.
Press any key to close this window . . .
```

Showing the top 10 results:



Observations:

The first 5 images appear to be more similar with the query image and that can be observed with the intersection values as well, all the remaining images after starting from sixth image seem to be more dissimilar and that's observable from the intersection values in the above figure.

Explanation:

The programs loads the two input images specified by the command-line arguments(the two query Images), calculates the magnitude of their Sobel gradients, computes the histogram of the magnitude, and converts the histogram to a vector of floats. It then loads all the images in the directory specified by the third command-line argument(data-set) and computes their histogram of magnitudes, converts it to a vector of floats. Finally computes the intersection(distance metric) between the two vectors using the **intersection** function.

The program stores the filename and intersection of each image in a **ImageData** struct and adds it to a vector. It then sorts the vector based on the intersection using the **comp** comparison function and outputs the sorted list of filenames.

In the first case when the program was evaluated on the training data set (a subset of Olympus dataset with 80 images), the 6 output images show a very reasonable relevance to the query images and the intersection values seem to be very low (observable in the command window)