CS5330_Project2

A short description of the overall project in your own words. (200 words or less)

The purpose of this project is to implement a pipeline to achieve content-based image retrieval from a given database. We need to design and use different feature vectors and distance metrics to retrieve similar images as the target image. Especially, we need to design special features for a specific type of image.

This project focuses on manipulating and analyzing image at a pixel level, and for the feature vector, the focuses are on color, texture, spatial layout, etc.

Any required images along with a short description of the meaning of the image.

Required result 1: Baseline Matching

Feature Vector: 9 x 9 square in the middle of the image

Distance Metric: sum of square difference

Target: image 1016

Results:

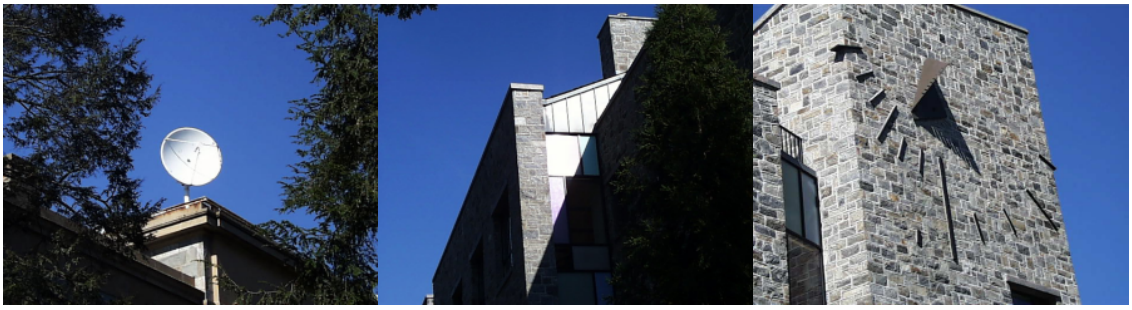
Required result 2: Histogram Matching

Feature Vector: 3D histogram of BGR color of the whole image. Use 8 bins for each color

Distance Metric: histogram intersection

Target: image 0164

If I do not normalize the histogram, the result is:



If I L2 normalized the histogram, the result is slightly different. The green bottle surprised me but the blue background does similar to the sky in the target image.

(For the tasks below, I am always L2 normalizing the histograms.)

Required result 3: Multi-histogram Matching

Feature Vector: Split the image into a 2 x 2 grid, calculate a 3D histogram of BGR color for each part. Use 8 bins for each color, and L2 normalize each histogram

Distance Metric: histogram intersection, each part is treated equally

Target: image 0274

Results:

Required result 4: Texture and Color

Feature Vector: The first feature is the 2D histogram of the Sobel gradient orientation and magnitude of the whole picture. The second feature is the 3D color histogram on the whole picture. Then the two feature vectors are concatenated to make sure they are treated equally when calculating histogram intersection.

Distance Metric: histogram intersection, each part is treated equally.

Target: image 0535

Result:

If I apply the feature vector and distance metric in task 3 on the target image, the results are:

If I apply the feature vector and distance metric in task 2 on the target image, the results are:

Required result 5: Custom Design

For task 5, I choose the blue trash bin as the specific type of image to execute CBIR.

I first apply the texture and color feature in task 4, then I also try to adjust the weight of texture and color and try to use different spatial. But the results are not satisfying. I believe it may be because the background can bring too much noise. Then I notice that in the given database, most of the blue trash bin objects are in the middle of the image. Therefore, I decided to focus on the middle part of the image. To achieve that, I split the image into a 3 x 3 grid and only calculate the texture and color features of the middle session of the grid. Also, I use histogram intersection as the distance metric.

The algorithm works better than calculating features on the whole image. For some input, the algorithm can find almost all the other same trash bins. It will also display some images containing blue gift boxes or blue mailboxes, which have similar color and shape as the trash bin.

The major limitation of this algorithm is that it only works on images in which the main object is located in the middle of the image. Also, texture analysis works better on horizontal and vertical lines. For objects with irregular edges(e.g. banana), it seems like that basic texture analysis doesn't work well.

Feature Vector: The first feature is the 2D histogram of the Sobel gradient orientation and magnitude of the middle part of the image. The second feature is the 3D color histogram on the middle part of the image. Then the two feature vectors are concatenated to make sure they are treated equally when calculating histogram intersection.

Distance Metric: Histogram Intersection

Target 1: image 0287

Results 1:

Target 2: image 0289

Results 2:

A description and example images of any extensions.

I choose to implement additional texture analysis using Gabor filters. And I will also combine it with the color information and spatial layout.

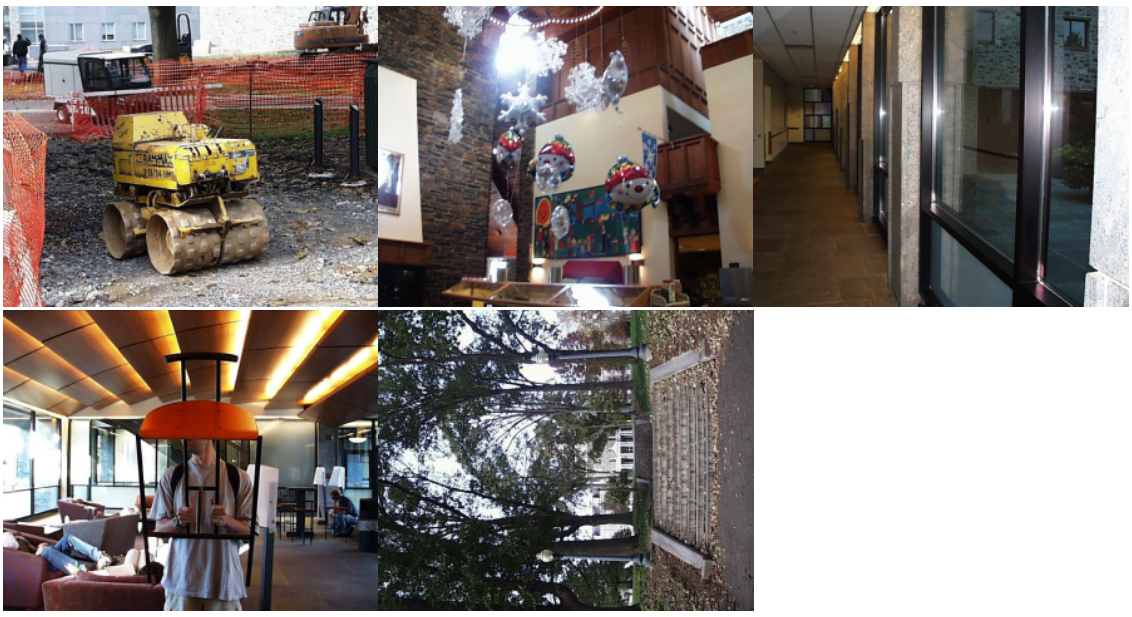
For the Gabor texture, I used the getGaborKernel() function from OpenCV to apply 48 different filters(in the combination of 5 scales and 16 orientations) to an image. Then for each filtered image, I calculate the mean and standard deviation of it. Finally, I put all the values in a feature vector and L2 normalized it to get the final feature vector.

Target: image 0535

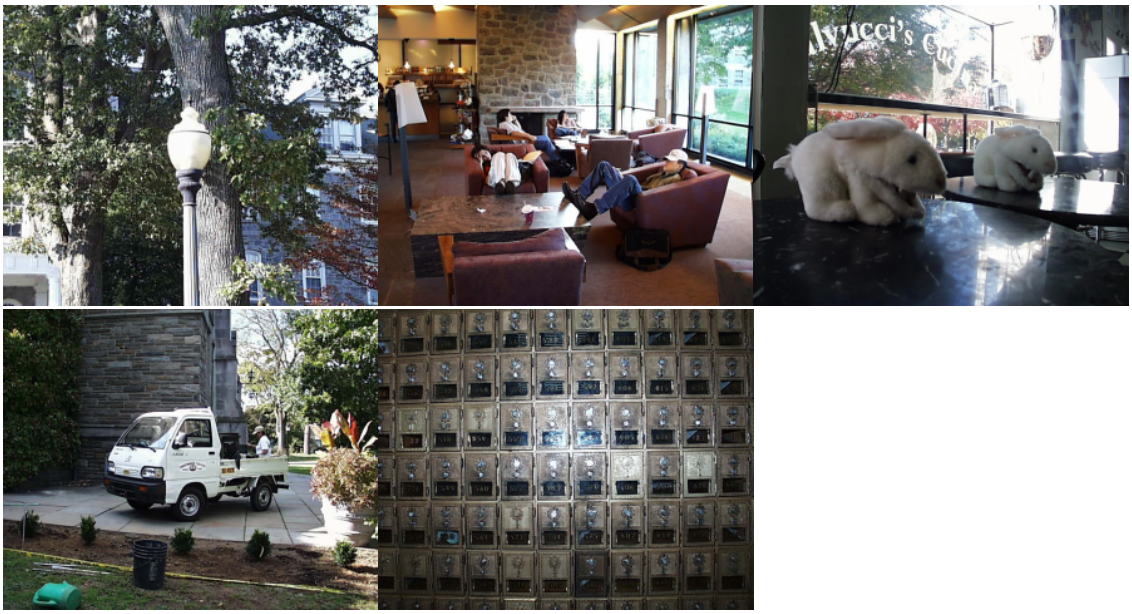
I apply the Gabor texture and sum-of-square-difference as the distance metric on the target image and get the following result.

Then I combined Gabor texture with the 3D BGR color histogram(used in task 2), weight them equally. I still use sum-of-square-difference as the distance metric.

The top 3 result is similar to what we get in task 4 using Sobel texture and color. However, the speed of running Gabor texture is much slower since it applies multiple filters to the image.



I also split the image into a 2 x 2 grid and apply the combination of Gabor texture and color histogram in each part, with sum-of-square-difference as the distance metric.



A short reflection of what you learned.

I learned how to design and implement different feature histograms as well as how to use the built-in feature extractors from OpenCV.

Besides, combining different features and modifying the weight of each feature provided me with a better understanding of how different features can influence the result of content-based image retrieval.

I also get a new understanding of texture analysis. For example, when I used texture-based features to find images similar to image 0535, the road roller always showed up. I was confused at the beginning, then after carefully comparing the road roller and my target image, I realized that both of the two pictures had several horizontal lines and the ground in the road roller image was similar to the wall in my target image. That helped me to see texture analysis in a new view.

Acknowledgment of any materials or people you consulted for the assignment.

- OpenCV Documentation: https://docs.opencv.org/3.4/index.html
- Understand 3D Mat: https://joebartelmo.com/computer_vision/opencv-accessors/
- Get ROI: https://stackoverflow.com/questions/6566295/opencv-c-getting-region-of-interest-roi-using-cvmat
- Using Gabor Filter for CBIR: https://cs.pomona.edu/~dkauchak/ir_project/whitepapers/Image.pdf
- Using Gabor Filter for CBIR: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.63.8420&rep=rep1&type=pdf