NORTHEASTERN UNIVERSITY

CS 5330: Pattern Recognition and Computer Vision (Spring 2022)

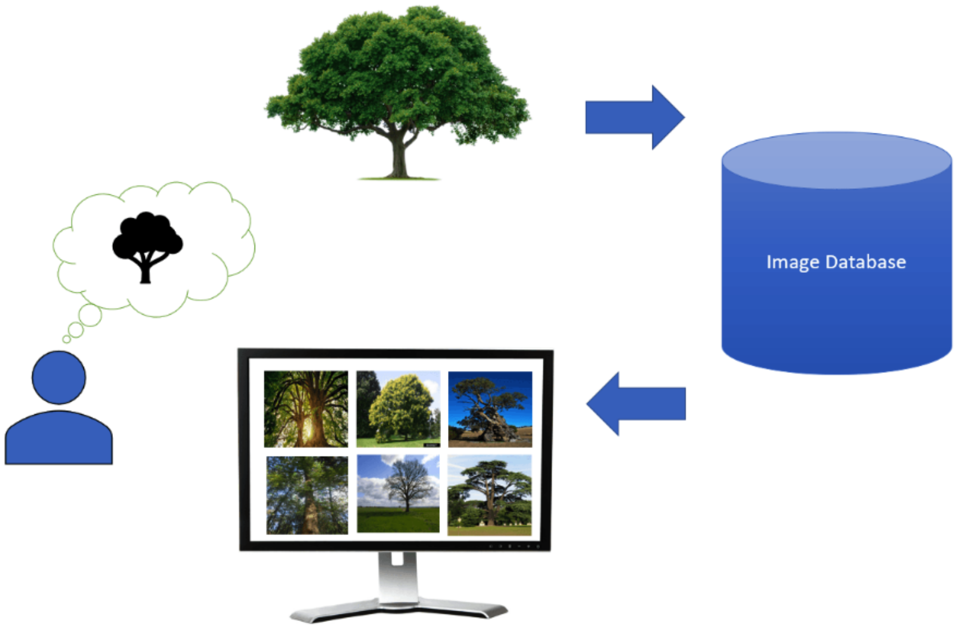
PROJECT - 2

CONTENT-BASED IMAGE RETRIEVAL

Submitted By:
TARUN SAXENA (002979327)

I. Description

Content-based Image Retrieval is an important area of computer vision. This project's goal is to continue learning how to modify and analyze photos at the pixel level. Furthermore, in this project, we are using matching or pattern recognition.



Content-Based Image Retrieval

This project's challenges included matching two images based on RGB pixels, matching images based on histograms, matching images using multiple histograms, comparing images based on a full-color image histogram and a full-texture histogram, and constructing a custom design. The entire process is implemented as a *command line program* that accepts a target filename for T, an image database directory as B, the feature type, the matching method, and the number of images N to return. The filenames of the top N matching images are printed by the program.

The above processing tasks are performed using OpenCV libraries in C++ programming language in Xcode IDE *(Version 14.2)* on MacBook Pro (M1 Chip).

II. Tasks

1. Baseline Matching:

In this task, we used the sum of squared distance metrics to match a center 9x9 image region of interest. The sum of squared is always 0 for the same image. All of the parameters are available as command-line arguments.

The top three matches for ***pic.1016.jpg*** are as follows:
1: pic.0986.jpg has a value of 354
2: pic.0641.jpg has a value of 518
3: pic.0233.jpg has a value of 1062

Fig.1 Target Image (pic.1016.jpg)

Fig.2 Match Image (pic.0986.jpg)

Fig.4 Match Image (pic.0233.jpg)

Fig.3 Match Image (pic.0641.jpg)

2. Histogram Matching:

To calculate the distance between the target image and other images, we used histogram intersection as a distance metric. In this case, we're making a 2D histogram with 8 bins. The feature vectors of all the images were saved and then used to calculate the distance between the target image and the other images.

The top three images matching with ***pic.0164.jpg*** found by sorting by distance are as follows:
1: pic.0110.jpg has a value of 0.575369
2: pic.1032.jpg has a value of 0.576556
3: pic.0092.jpg has a value of 0.610901

Fig.5 Target Image (pic.0164.jpg)

Fig.6 Match Image (pic.0110.jpg)

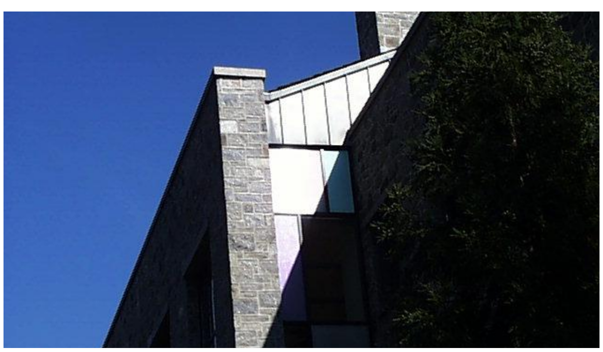


Fig.7 Match Image (pic.1032.jpg)

Fig.8 Match Image (pic.0092.jpg)

3. Multi-Histogram Matching:

In this task, we began by dividing the image into two halves: top and bottom. We calculated the RGB histogram using to generate a 2D histogram with 8 bins for each half. We concatenated the features of both halves and calculated their weighted intersection. We calculated the distance between the target image and other images using histogram intersection as a distance metric. All of the images' feature vectors were saved and then used to calculate the distance between the target image and the other images.

The top three images matching with ***pic.0274.jpg*** found by sorting by distance are as follows:
1: pic.0273.jpg has a value of 0.173669
2: pic.1031.jpg has a value of 0.187555
3: pic.0409.jpg has a value of 0.189761

	
Fig.9 Target Image (pic.0274.jpg)	**Fig.10 Match Image (pic.0273.jpg)**

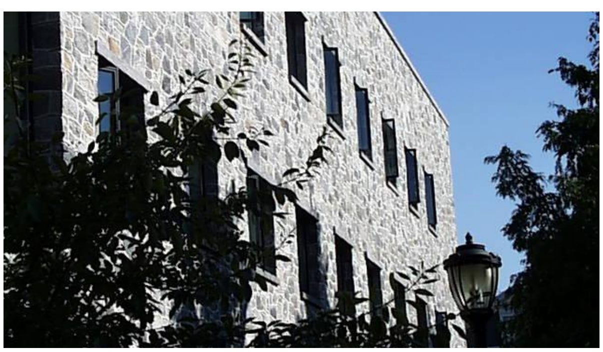



Fig.11 Match Image (pic.1031.jpg)	**Fig.12 Match Image (pic.0409.jpg)**

4. Texture and Color:

We used a combination of color and texture as features in this task. The color feature is an RGB histogram using 8 bins, and the texture feature is a magnitude of sobel filters histogram. We obtain the Sx and Sy images first, and then compute the magnitude image from these. The magnitude image was then histogrammed. Both of these features were combined were saved for future use. A weighted histogram intersection is used as the matching metric. The intersection weights are parameters that can be changed. We used equally weighted histogram intersections to achieve the following results.

The top three images matching with ***pic.0535.jpg*** found by color and texture matching are as follows:
1: pic.0171.jpg has a value of 0.169987
2: pic.0004.jpg has a value of 0.177052
3: pic.0605.jpg has a value of 0.179577

Fig.13 Target Image (pic.0535.jpg)

Fig.14 Match Image (pic.0171.jpg)

Fig.15 Match Image (pic.0004.jpg)

Fig.16 Match Image (pic.0605.jpg)

The texture analysis image displays photos with the same number of edges and similar color. As pic.0004 is not visually similar, it is the best match to the given input image as the image has lots of edges and have similar pixel values.

The top three images matching with ***pic.0535.jpg*** found by Histogram Matching are as follows:

	
Fig.17 Target Image (pic.0535.jpg)	**Fig.18 Match Image (pic.0285.jpg)**
	
Fig.19 Match Image (pic.0628.jpg)	**Fig.20 Match Image (pic.0952.jpg)**

The histogram matching returns images with similar wall texture, and because the chairs, sky, and lights in the room are all white, these images are returned as the top results.

The top three images matching with ***pic.0535.jpg*** found by Multi-Histogram Matching are as follows:

	
Fig.21 Target Image (pic.0535.jpg)	**Fig.22 Match Image (pic.0698.jpg)**
	
Fig.23 Match Image (pic.0628.jpg)	**Fig.24 Match Image (pic.1105.jpg)**

In the case of the multi histogram, the results are produced by matching the top and bottom halves of the photos. The best result for this method is a ceiling that the algorithm believes is a similar image.

To conclude, the result obtained from this texture and color matching show the images above seem to have similar strong lines and colors overall. In addition, this comparison is a lot different from the ones above. In histogram matching, only color comparisons were done. In multi-histogram matching, spatial comparisons were done. In matching by texture and color, the textures and colors of the strong lines were compared.

5. Custom Design:

We used a combination of color and texture as features in this task. We generated the color feature using a region of interest and focused on the center of the image. Because most of the green bins will be kept outside, with trees and a road visible, the texture feature is calculated for the entire image to capture the surroundings as well. We obtain the Sx and Sy images first, and then compute the magnitude image from these. The magnitude image was then histogrammed. Both of these features were combined.

A weighted histogram intersection is used as the matching metric. The intersection weights are parameters that can be changed. We used 0.5*color feature and 0.5*texture feature weighted histogram intersections for the results shown below.

We can see in the results below that when we used an image with green bin, the results we got using the green bins feature were good, and we were able to recover 4 similar images out of 10.

For the Target Images pic.0747.jpg and pic.0748.jpg, the custom matching result is as follows:

Fig.25 Target Image (pic.0747.jpg)

Fig.26 Target Image (pic.0748.jpg)

Fig.27 Match Image (pic.0747.jpg)

Fig.28 Match Image (pic.0748.jpg)

Fig.29 Match Image (pic.0746.jpg)

Fig.30 Match Image (pic.1102.jpg)

Fig.31 Match Image (pic.0749.jpg)

Fig.32 Match Image (pic.0707.jpg)

Fig.33 Match Image (pic.0691.jpg)	**Fig.34 Match Image (pic.0427.jpg)**
	
Fig.35 Match Image (pic.0391.jpg)	**Fig.36 Match Image (pic.0736.jpg)**

Similarly, we can see in the results below that when we used an image with green plants, the results we got using the green plants feature were good, and we were able to recover 7 similar images out of 10. For the Target Images pic.0067.jpg and pic.0068.jpg, the custom matching result is as follows:

	
Fig.37 Target Image (pic.0067.jpg)	**Fig.38 Target Image (pic.0068.jpg)**

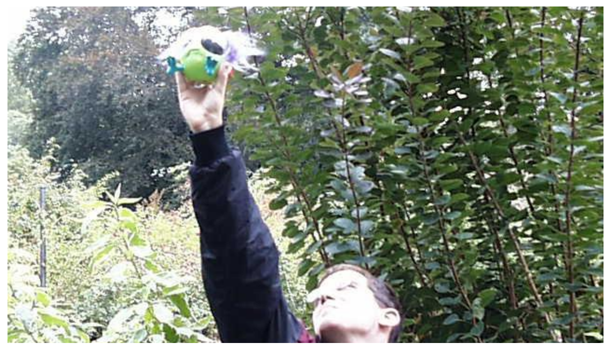
Fig.37 Match Image (pic.0067.jpg)	**Fig.37 Match Image (pic.0067.jpg)**
	
Fig.38 Match Image (pic.0254.jpg)	**Fig.39 Match Image (pic.0700.jpg)**
	
Fig.40 Match Image (pic.0399.jpg)	**Fig.41 Match Image (pic.0673.jpg)**
	
Fig.42 Match Image (pic.0112.jpg)	**Fig.43 Match Image (pic.0256.jpg)**

Fig.44 Match Image (pic.0130.jpg) | **Fig.45 Match Image (pic.0065.jpg)**

6. Extensions:

6.1. Blue Bin Detection

For this task the image is converted into the HSV color space and filtered to blue. This produces a binary image of one channel. The histogram of this binary image is computed using the knowledge gained in Task 5 - Custom Matching. Because the blue bins are plain and lack texture, computing magnitude could have produced a variety of results. Because the bin is in a single location, the spatial variance of the histogram will result in small values. As a result, I computed the distance metric using the sum of squared differences in the color histogram and the spatial variance. This resulted in 8 blue bins appearing in the top 10 results that gives 80% accuracy for our dataset.

Fig.46 Target Image (pic.0287.jpg) | **Fig.47 Match Image (pic.0289.jpg)**

Fig.48 Match Image (pic.0969.jpg)

Fig.49 Match Image (pic.0665.jpg)

Fig.50 Match Image (pic.0666.jpg)

Fig.51 Match Image (pic.0291.jpg)

Fig.52 Match Image (pic.0214.jpg)

Fig.53 Match Image (pic.0920.jpg)

Fig.54 Match Image (pic.0288.jpg)	Fig.55 Match Image (pic.0764.jpg)
	
Fig.56 Match Image (pic.0288.jpg)	

6.2. Sobel Comparison:

For this extension, the original's sobel x and y filters were compared to the sobel x and y filters of other images. The images returned by the sobel Y filter are similar to the original pic.0706, with strong edges. The images of the Sobel X filter produced the same result.

For Sobel X filter Matching

	
Fig.57 Target Image (pic.0706.jpg)	**Fig.58 Match Image (pic.0864.jpg)**
	
Fig.59 Match Image (pic.0919.jpg)	**Fig.60 Match Image (pic.0087.jpg)**

For Sobel Y filter Matching

	
Fig.61 Target Image (pic.1025.jpg)	**Fig.62 Match Image (pic.0087.jpg)**
	
Fig.63 Match Image (pic.0864.jpg)	**Fig.64 Match Image (pic.0919.jpg)**

III. Learning Outcomes

This project on Content-based Image Retrieval which focuses on filtering real time images was both interesting and challenging to work. I learned in this project how to use several features of an image to locate images that are similar to it. I saw how a minor change in a characteristic feature can result in the retrieval of various images, such as a small difference in weightage for color and texture features, or applying color and texture functions to slightly different areas of the image. The various matching strategies aided us in understanding Content Based Image Retrieval. This project has allowed me to broaden my knowledge of histograms, texture analysis, and the essential parts required for various images. I discovered that the content-based image retrieval system is very dependent on the type of image we input, and that we must create a separate filter for each image to achieve the best results.

IV. Acknowledgments

I am thankful to Prof. Bruce Maxwell for designing and executing a wonderful learning experience in Computer Vision. I would also like to extend my gratitude to this course's TA's for clearing the doubts whenever required. The materials that have helped in the completion of this project are:

1. CS 5330 Course Materials
2. https://www.baeldung.com/cs/cbir-tbir
3. OpenCV documentation: https://docs.opencv.org/3.4/index.html
4. StackOverflow