Detection of Objects with Simple Shape-and-Color Based Features

Nhat-Minh Nguyen, Nam-Thanh Le

2021 January

1 Abstract

In the project, attempts are made to detect and recognize objects based on their own specific shape including basic ones: squares, circles, triangles, rectangles, pentagons, and hexagons. Previously, objects might be divided into groups of colors which are determined by their variables in HSV Color Channel. The main image processing used knowledge for shape division is about contour finding in an image. Image Sources is fed mainly from the Internet, while some are selfmade.
Index Terms: Color Filters, Shape Detection, Contours.

2 Introduction:

Because of lots of people who are color-blind in this world, so detect object's color is really necessary. With the help of modern technology, color-blind patients can be supported by image processing. People used to be unable to recognize the color now can do so.
There are a lot methodologies to detect objects based on their shape. For example, template matching is a fine method to detect the objects based on the predicted template. Then, the application of Machine Learning is great as it is on the tendency of Image Processing and Artificial Intelligence now. With an enough amount of database, Machine Learning or even Deep Learning is accredited to be the most accurate and the fastest method. However, for objects which can be randomly rotated, applying Machine Learning or Deep Learning is obviously a waste of data and time for training. Therefore, in order to recognize the objects, we will use image contour method which is a process of identifying structural outlines of objects. Then, the shape of the object will be identified relatively fast and accurate.
The program is supposed to be able to detect objects based on their shapes. Besides, the program can detect color depend on colors with many shades.

3 Methodologies:

3.1 Image Thresholding:

For a gray converted image, in case the intensity of a pixel in the image is greater than the preset threshold, the corresponding output pixel is marked as white (foreground), and if the input pixels intensity is less then or equal to the threshold, the output pixel is marked as black(background). The used function in threshold has the first parameter is a gray image whereas the second one is a threshold, the third value is the max-value which is assigned that if the intensity pixels are larger than threshold, the fourth is the category of threshold. Depending on the sort of threshold, the intensity pixels will be assigned by different values which are of these five categories:
1) Thresh Binary: If value of pixels are larger than threshold, then it will be assigned by maxval. Otherwise, it will be assigned by 0.
2) Thresh Binary Inverted: If value of pixels are larger than threshold, then it will be assigned by 0. Otherwise, it will be assigned by maxval.
3) Thresh Trunc: If value of pixels are larger than threshold, then it will be assigned by equal to threshold. Otherwise, it will be saved.
4) Thresh Tozero: If value of pixels are larger than threshold, then it will be saved .Otherwise, it will be assigned by 0.
5) Thresh Tozero Inverted: If value of pixels are larger than threshold, then it will be assigned by 0. Otherwise, it will be saved.

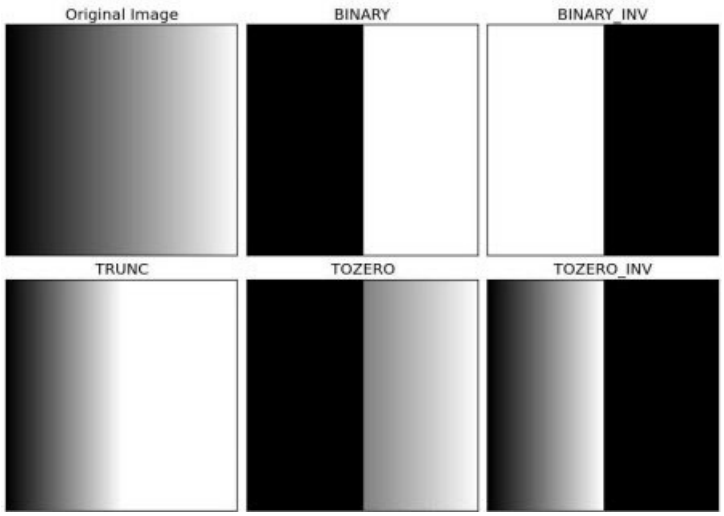

Figure 1: Thresholding Examples

In the system, we applied Thresh Binary threshold method since it is the most simple yet effective to clarify the image sources.

3.2 Color Detection:

For a long time, HSV Color Channel has been applied as a alternative solution for problems related to colors. Basically, if RGB Color Channel is generated as the result of 3 filters : Red, Green, and Blue, as matrixes, HSV Color Channel is generated as the combination of the 3 variables: Hue, Saturation, and Value in which Hue corresponds to mask color, Saturation to the mixture of that color with White, and Value is the mixture with Black. In our system, a mask created in HSV channel is applied as a color filter to detect only the color which users want. For each range of variables chosen, the mask works as a barrier which only allows the chosen color to pass through while blocking the other colors.
About the reason why people choose HSV instead of RGB, it's quite simple since noise reduction works better on HSV whose color is separated from light intensity, colors are seperated easier without too much optical noise. After this, a gray color conversion is performed so as to threshold the picture and to be a supportive step to contours finding.
There are many ways to convert an image from RGB to HSV Color Channel. However, in this project, we applied only two of the popular ones. The first one is determining HSV color range based on the formula in Figure 2. In this method, the variables are determined from V, S, to H respectively based on the maximum and the minimum variables among R, G, and B. Being academic, this method is proved to give the most exactness. However, since it relates too much to craft calculation, it is difficult to avoid mistakes while performing it.

$$V \leftarrow max(R,G,B)$$

$$S \leftarrow \begin{cases} \frac{V-min(R,G,B)}{V} & \text{if } V \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$H \leftarrow \begin{cases} 60(G-B)/(V-min(R,G,B)) & \text{if } V = R \\ 120+60(B-R)/(V-min(R,G,B)) & \text{if } V = G \\ 240+60(R-G)/(V-min(R,G,B)) & \text{if } V = B \end{cases}$$

Figure 2: RGB to HSV Formula

Therefore, we managed to find out another method to determine HSV range color. The range of variables are chosen based on Figure 3, in which V value will be bracket from 0 to 200 where colors still keep their features, S value is taken from 150 to 255 which all colors start to show their feature, the lower S generates much noises since all colors look the same, the H variable is chosen at last based on the X-axis as a clear color devision.
For example, the color range of orange is [0,150,255] to [25,255,255], of purple is [131,150,255] to [155,255,255], etc.

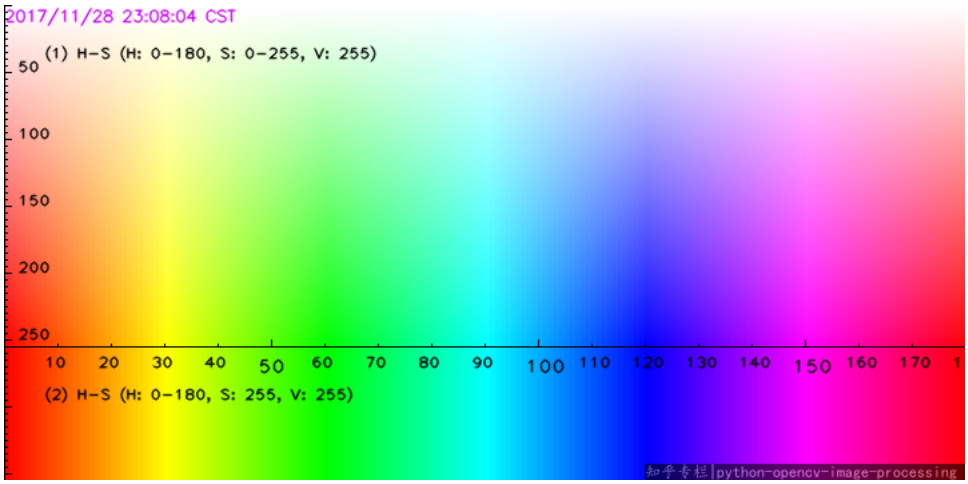

Figure 3: HSV Color Range

3.3 Shape Detection:

Applied to binary images thanks to Thresholding methods, Contours are defined as lines connecting all of the pixels, which are described as points, along the boundary of a shape whose pixels are relatively in the same intensity or color level. Contours is a helpful assistant in analysis of size, shape, etc.
In our project, the contour are classified into 6 states: contours with 3 lines are recognized as Triangle, 4 lines are classified into Squares which have the ratio of height and width between 0.95 and 1.05, Rectangles are the left of 4 lines, 5 lines are recognized as Pentagons, 6 lines are recognized as Hexagons and all of other states will be recognized as Circles. There are a lot of modes and methods in storing the information of contours. However, as we used Tree mode which stores fully all the information of contours and hierarchies of them, and Approximation none method which stores all points producing curves into a numpy array. Besides, there are many other modes and methods such as External mode which only receives the outmost contour, CComp mode which takes all contours and divides them into two hierarchies: the outernal and the internal. For method, there is also a Simple method which only stocks the coordinates of vertexes instead of all points on contours. However, through experiments, the Tree mode and the Approximation method turns out to be the most efficient selection for this kind of project that they produce the result as Figure 4. Then, an approximation is performed to connect separated points. Obviously, there is also a threshold of distance for any two points to be connected. If they are too far from each another, the connection is impossible.

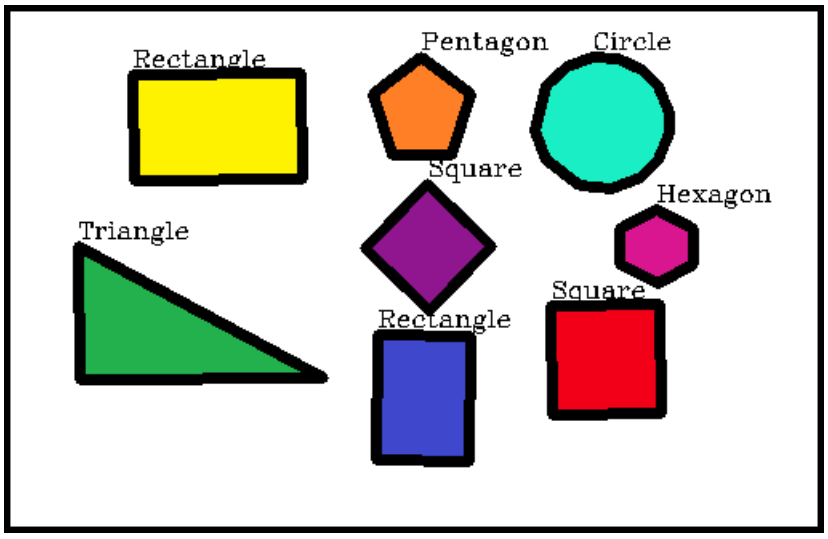

Figure 4: Shape Detection

4 Final Project:

This is our results while experimenting with different requirements of colors:

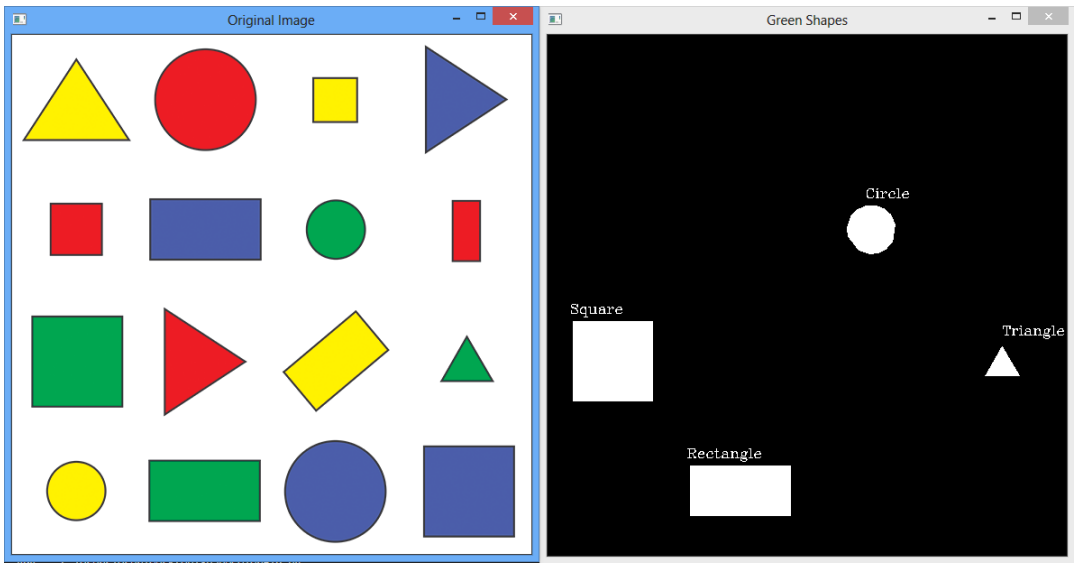

Figure 5: Green Shapes Classifier

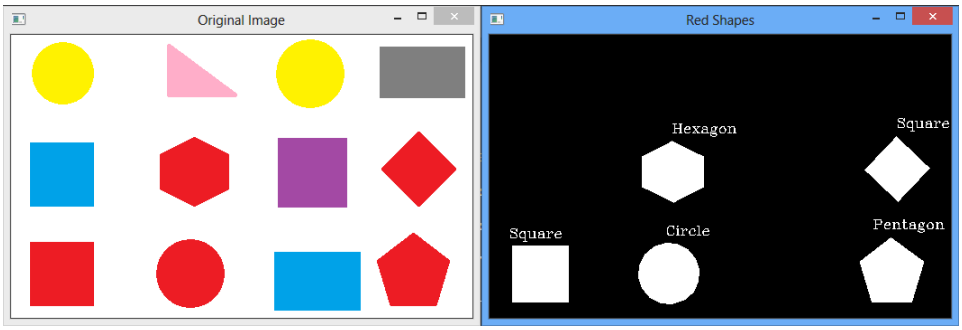

Figure 6: Red Shapes Classifier

5 Conclusion:

For many experiments, it turns out that the system works properly for basic shapes. By converting to HSV channel, it reduces the noises of light which will be a big problem in RGB color channel. Then, by pre-thresholding the picture, contours are found easily. However, for complex shapes such as stars, thunders, shapes with many vertexes or unaligned lines, the program will gets difficult due to the lack of pre-trained recognition. This problem can be obviously solve by more detailed input division of shapes.
About color, since it uses the threshold of color to classify objects, there will be a colored problem as the background has the other colors apart from white, or has the color which is almost the same as objects.

References

[1] Ruibing Jin and Jianliang Wang. *A Vision Tracking System Via Color Detection*. IEEE, 2016.

[2] OpenCV. *Color Conversions*.

[3] Steven L.Eddins Rafael C.Gonzalez, Richard E.Woods. *Digital Image Processing Using MATLAB*. 2016.

[4] Smruti Saurav Shasani, Ramiya Ranjan Meher, and Manoj Kumar Patra. *Color and Shape Recognition*. IEEE, 2016.

[5] Richard Szeliski. *Computer Vision: Algorithms and Applications*. Springer, 2011.
[1] [4] [3] [5] [2]