

Image Classification

Yiren Shi

May 1, 2016

0.1 Approach

In this project, I used a new programming library OpenCV to do image processing. Firstly, the user should the file path of the image that need to be test. Then, depend on the data exists, I built up a template which is one of the images in the data. Since in this program I used supervised learning to do classification, the template can be considered as training set and the image inputted is considered as test set. After building the template, I changed both test image and the template into the same size to make sure the only thing the program compares is the shape of the figure. After that, using the function `cv2.matchTemplate` to compare how much does the test image match the template. I set a threshold for the percentage of match and a temporary max match percentage for each class. While testing by each class, if the percentage of match is greater than threshold, this percentage will be set as the temporary max match percentage for that class. Since the more similar two images are, the higher match percentage will get. If the match percentage is lower than the threshold, those two images are not similar enough to be considered as same class. Then the program will choose the next image in the data as a new template and compare with the test image again to check if there is another data matches more with the test image. If there exist a higher match percentage, then the max will be changed into the higher percentage. The program will repeat compare, match and get highest match percentage by each class until all the data in all 5 classes has been compared to the test image. After checked all the data, the program will find the highest match percentage from five max match percentage for the five classes. Then the program will determine the class of the test image as the same with the one get the highest match percentage after all. At the end, I checked if the test image is the same class as the program determined to verify if it is correct.

0.2 Threshold Determination and Improvement

For the threshold, first I set it as 0.98 which is close to 1 means the test image should be 98 percent same with the template. If the test image is exactly same with one of the image in the data, it can give me 100 percent correct answers. However, since the threshold is high, the program cannot

determine the class of image which is not in data at all. In order to test more images which are not in my data, I decided to reduce the threshold. In this case, I test how much percentage two different images will similar to each other. Then I get if two different images are in different class, the highest percentage they match each other is near 45 percent. However, if two different images are in the same class, the highest percentage can reach 65 percent to 70 percent. So I changed my threshold into 0.65, which can probably determine the class of the image not in the data and provide correct answer. But the reduction of the threshold will result in some incorrect class determination by two reasons. The first is there may exist two images that in different class which are similar enough (which I set is 65 percent) to each other to be determined as the same class by the program. The second reason is there may exist an image that does not reach the 65 percent similarity with any other images in its class as well. In order to solve this problem, I changed program to let it able to find the max match percentage for each class. Then, it will compare the max match percentage for each class and find the highest one. In this case, the program can determine the class of the test image by checking which class the test image matches the most. In this way, it can provide me correct answers.

0.3 Accuracy

Firstly, I train the program with all the data in all 5 classes and test it with images in the data. As long as the test image is inside my data, the accuracy will be 100 percent with the threshold 0.98. Since the test image is from my data which has been trained before, the program can definitely determine the class of the test image. But if I test it with images that is not in my data, the accuracy will be 0 percent. Then, I changed my threshold into 0.65 to test more images which are not in my data, the accuracy will reduce into 90 percent. To increase the accuracy, I modified my program to make it find the highest match percentage which is higher than my threshold 0.65. In this way, my accuracy increase to near 100 percent. Since if there exist an image which cannot reach threshold with any class I have, the program still cannot recognize it.

0.4 Conclusion

The OpenCV is a really useful library to do image processing and matching two images. In this program, it can recognize images which is not just limited to the data by using a lower threshold. But as a result of having a lower threshold, the accuracy will reduce a little bit as well but still acceptable. The program cannot ensure it can reach 100 percent accuracy while testing with images out of the data but still try to make it as accurate as possible.