**Project 7 Coin Detection with OpenCV**

Name: \_\_\_\_\_Michael Fatemi\_\_\_\_\_\_ Period: \_\_\_7\_\_\_\_ Date: \_\_\_\_5/5/2021\_\_\_\_\_

Did you name your file l071.cpp (Lower case L, then 071)? \_\_Yes\_\_

Did you use OpenCV to detect coins? \_\_\_\_\_Yes\_\_\_\_\_\_

What functions/methods from OpenCV did you use?

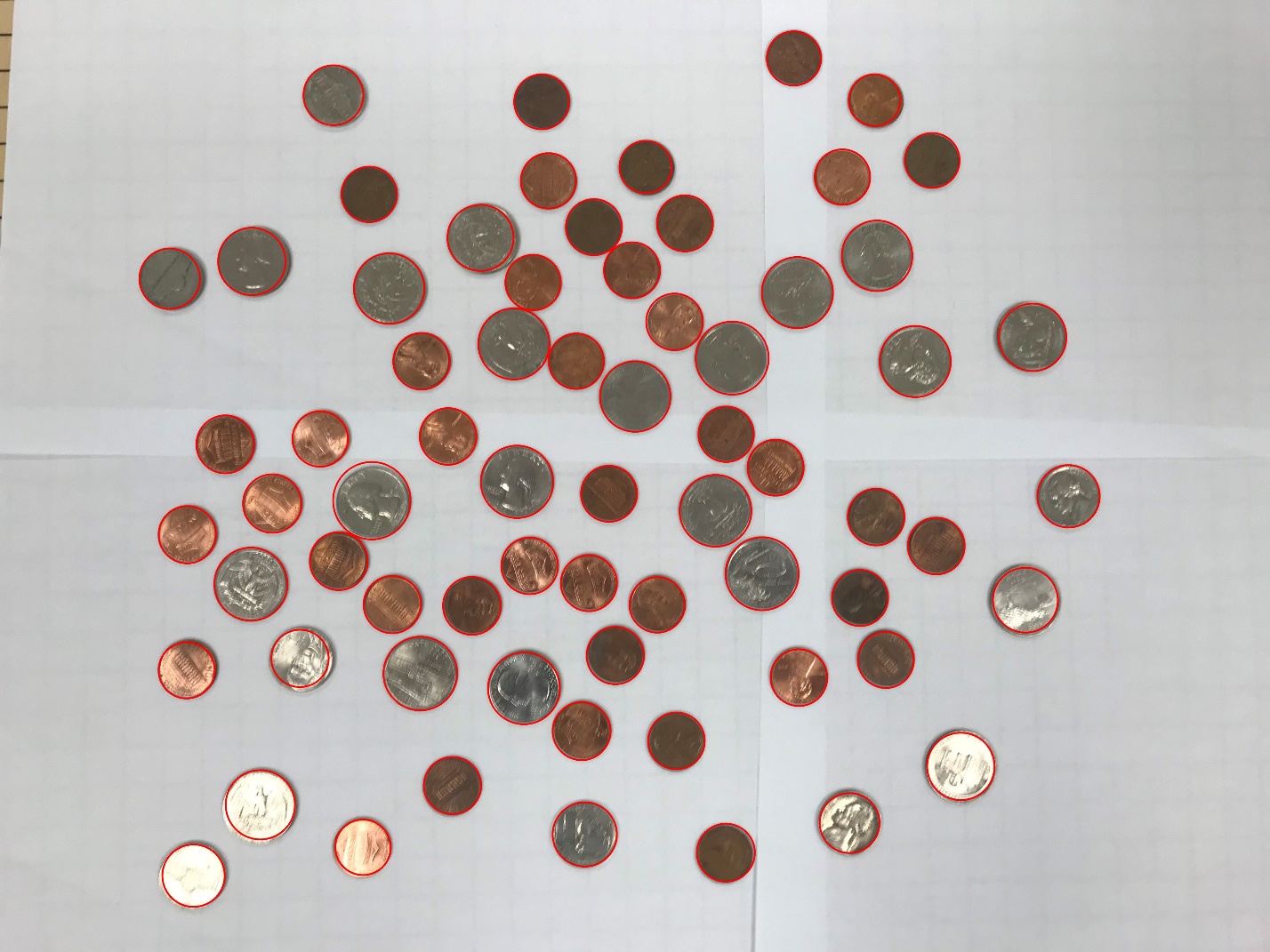
I ended up using Hough circles and median blur (because it was able to reduce noise in edge detection during Hough Circles). Surprisingly it was harder to get Hough circles to work with OpenCV because there was less flexibility in how I could score the circles

What functions/methods from OpenCV did you experiment with but ended not using?

I tried using Gaussian blur, but it resulted in it being more difficult for the Hough transform to detect the edges of the circles accurately enough.

Below do the following for each of the 3 images I provided paste 2 clear pictures of the initial picture, the picture with the original image and the circles you detected drawn in red (bolded and you may use more colors if you wish) and also the content of your results.txt file

1. For the easy image: (paste in next line)



Result:

39 pennies

8 nickels

3 dimes

18 quarters

0 silver dollars

For a grand total of $5.59!

1. For the medium image: (paste in next line)



Result:

66 pennies

10 nickels

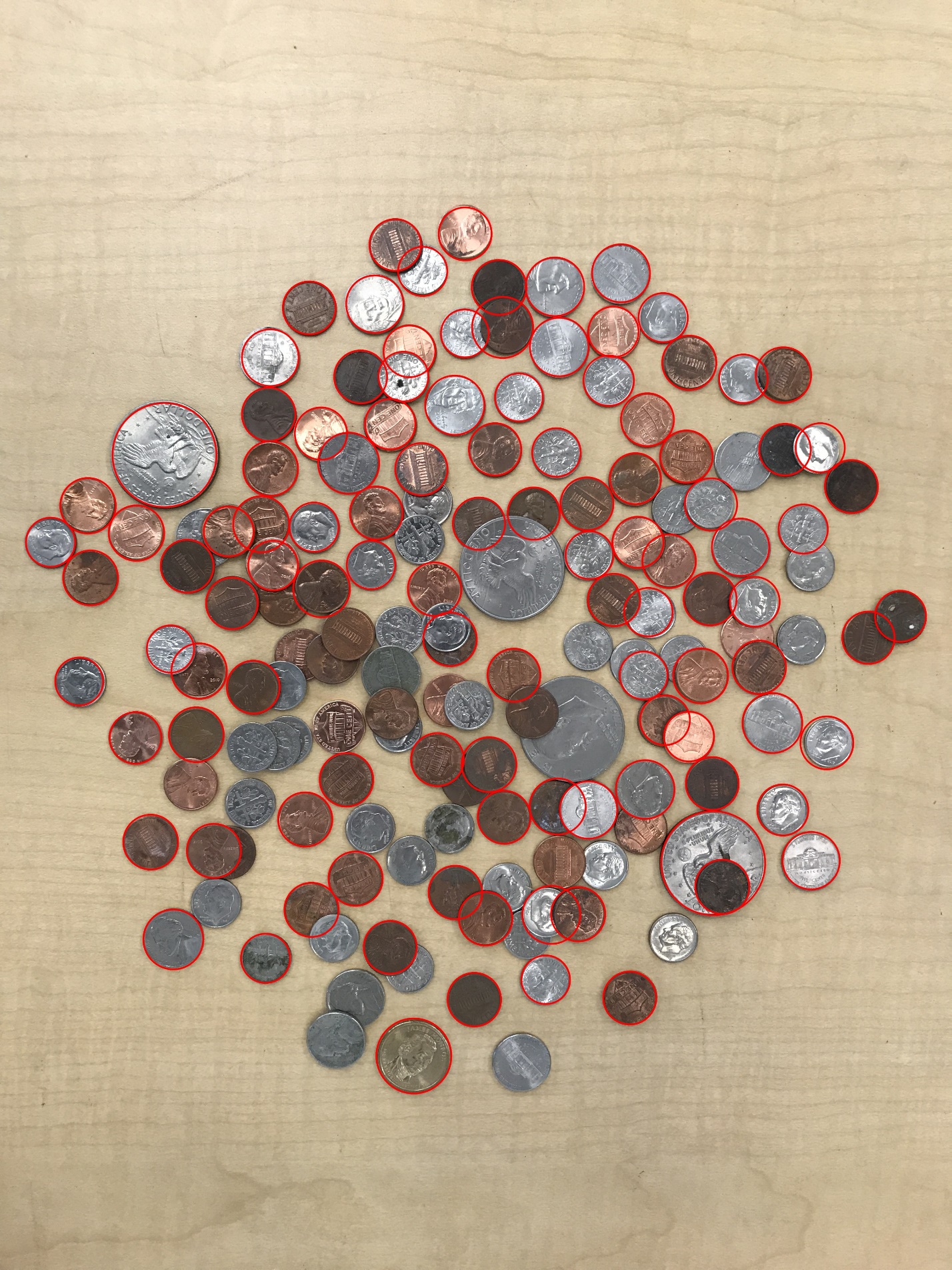
7 dimes

32 quarters

0 silver dollars

For a grand total of $10.36!

1. For the hard image: (paste in next line)



Result:

48 pennies

14 nickels

45 dimes

1 quarters

2 silver dollars

For a grand total of $7.93!