

# VR Assignment 1

Siddharth Reddy Maramreddy  
IMT2022031

February 2025

## 1 Part 1

I selected Fig. 1 for this part. First I applied Gaussian blur to smoothen the image as the coins contained unnecessary details (see Fig. 2). Then I converted the image to greyscale. I applied Canny edge detection with thresholds 50 and 100 on the greyscale image. Fig. 4 shows the output.



Figure 1: Image containing Indian coins.

Then I used Otsu's thresholding on the greyscale image to segment the coins (see Fig. 5). Otsu's thresholding is a technique used to automatically



Figure 2: Output after applying Gaussian blur on Fig. 1



Figure 3: Output after converting Fig. 2 to greyscale

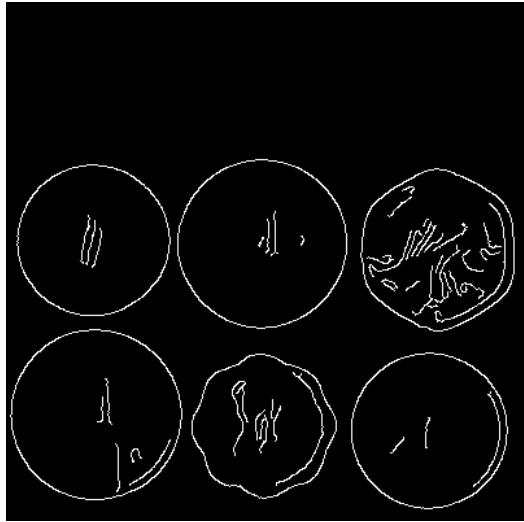


Figure 4: Canny edges

separate pixels into two classes in an image. Then I found contours in the thresholded image and selected contours with an area greater than 150. This gives the bounding circle for each of the coins. I counted the contours to get the number of coins in the image. Fig. 6 shows the contours drawn on the original image. I used these contours to create masks and used the mask to extract each coin from the image. Figs. 7a, 7b, 7c, 8a, 8b and 8c show the segmented output for each coin.

## 2 Part 2

I selected Figs. 9b and 9a for this part. I used SIFT on these images to get key points and descriptors. But the number of key points I got for each image was nearly 200000. So, I sorted the key points by response values and selected the top 10000 key points. Figs. 10a and 10b show the selected keypoints for each image. I then used the KNN matcher ( $k = 2$ ) to find matching key points in both images. I then filtered the matches using Lowe's ratio test eliminating matches with the ratio of distances less than 0.75. The filtered matches are shown in Fig. 11.

I used these matches to find the homography matrix. I then used the homography to warp the second image into the dimension of the first image.

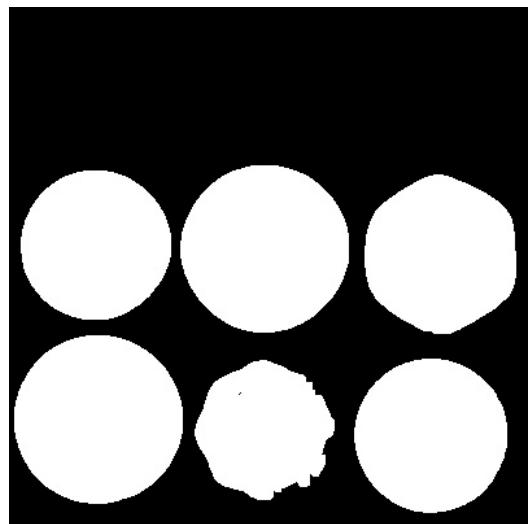


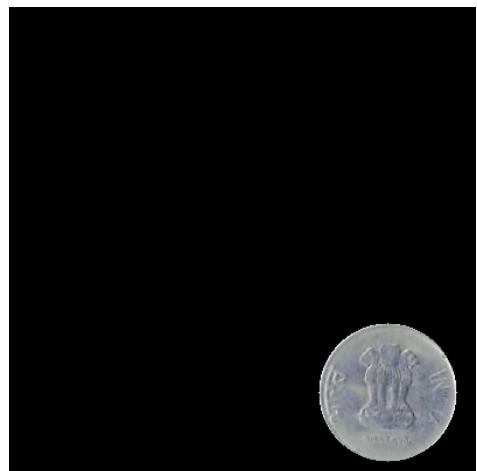
Figure 5: Thresholded output



Figure 6: Contours drawn on Fig. 1



(a)



(b)



(c)

Figure 7: Segmented outputs



(a)



(b)



(c)

Figure 8: Segmented outputs

Finally, I overlaid the first image to get the panorama. Fig. 12 shows the final output.



(a)



(b)

Github repo link: <https://github.com/siddharthmaram/VR-Assignment-1>



(a)



(b)

Figure 10: Best 10000 key points detected in each image



Figure 11: Keypoint matches

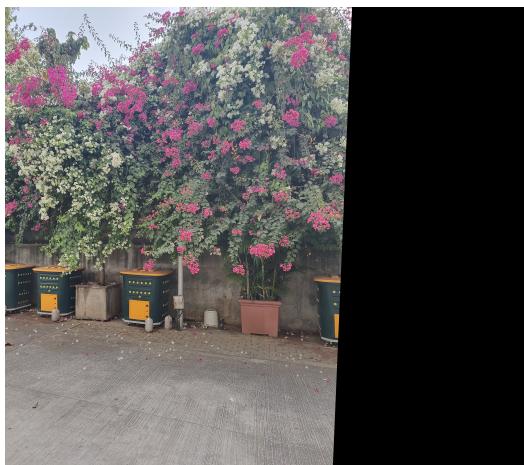


Figure 12: Final Panorama