Report 3

SELECTED ISSUES OF MACHINE VISION

Team 1C

1 Vasu Mangukiya
2 Harshit Verma
3 Wiktor Pelka

Instructor **Martinez Carranza Juan**
MSc. Eng

Date of exercise [23-03-2023]
Date of report submission [17-04-2023]

Total score _____ **/ 5.0**

1. Goal of the exercise

This exercise aims to familiarize us with the techniques for picture collecting and analysis that are helpful in the management of production processes. We'll introduce some fundamental machine vision object recognition methods and functions.

2. Lab run

2.1 Recognition of elements

- A method for locating particular trained objects inside the current image is provided by the Object Recognition module. Depending on the filtered parameters of confidence, size, rotation, etc., the module will detect those items inside the current image once it has been trained with example template images.
- A relatively common method of face recognition chooses a set of pixel comparisons that best captures the idea of a face using Haar-like filters. Although it has its uses, the Haar approach can also be used to recognize particular objects.
- Using a modified fast Harris feature detector, the Feature Points method will find noteworthy points in the template and compare them to points found in the current image
- Recognizing Street Signs for self-driving vehicles,
- Recognizing Logos for commerce.
- Fiducials for Medical purposes.

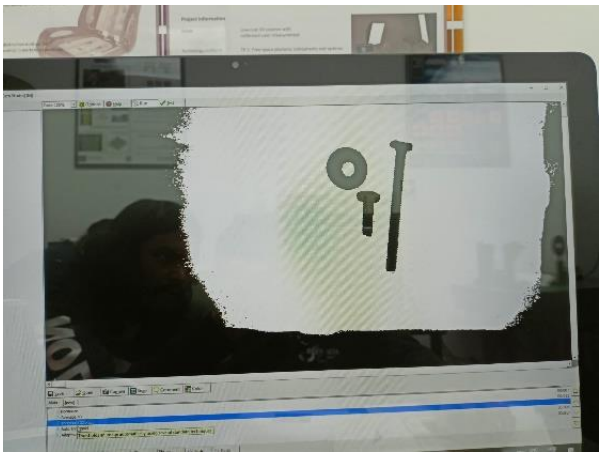

Figure 1.1 [Threshold]

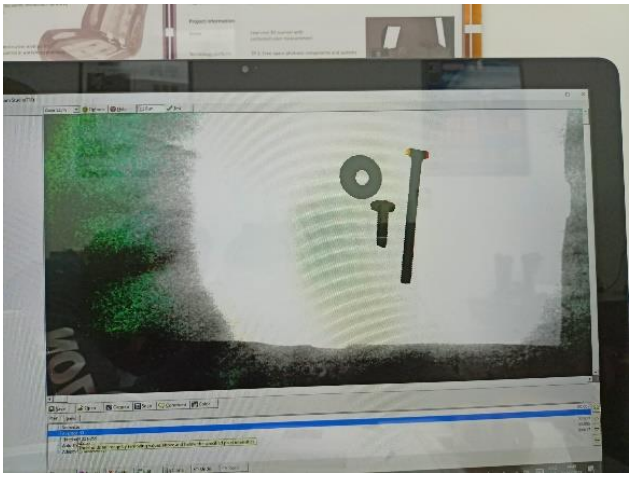

Figure 1.2 [Average Threshold]

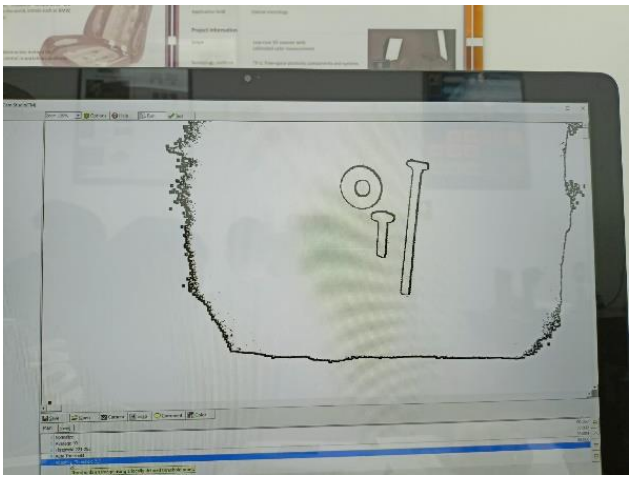

Figure 1.3 [Adaptive Threshold]

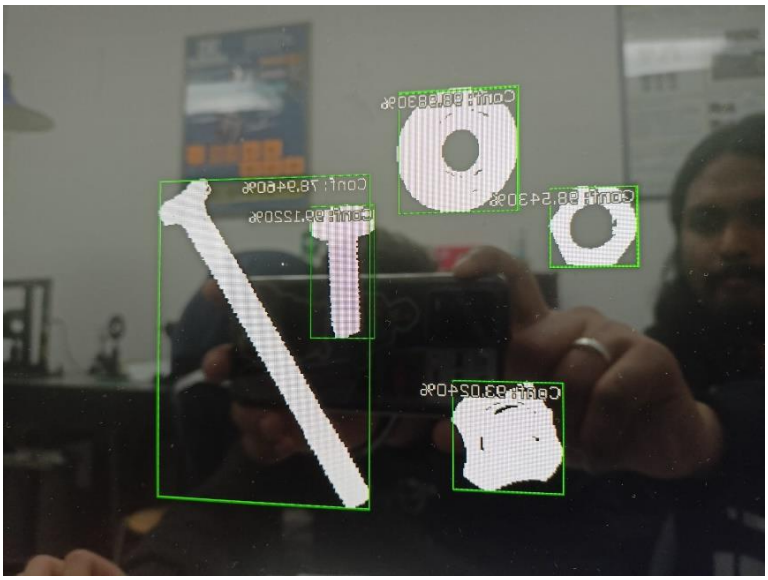

Figure 1.4 [Object Recognition]

Conclusions

- We were successfully able to complete the task.
- RoboRealm could detect the Objects and tag them.
- Major Difficulties are managing the lighting of the setup, incorrect lighting could produce shadows or reflections.
- Adaptive threshold doesn't respond well to the reflections, through that we could get a better outline of the object, but in this case, we needed whole object silhouette recognition.
- Best way to achieve the desired form of the Object is to use a Normal Threshold Filter and manage the Filter manually.

2.2 Part 2 – Recognizing hand gestures

- The attributes of the centre of gravity are employed by the hand gesture system to determine the distance between the object's centre and the index finger, indicating the direction in which the robot should travel.
- Using particular hand gestures for each piano note, hand gesture recognition can be used to play a virtual piano. It can also be used to play interactive games, allowing two players to control the game using their hands instead of a joystick.
- Simulating the mouse as a visual input device with all of its functions, including left and right clicks, double clicks, dragging and dropping, and scrolling, can be a significant application of technology.
- Face Recognition.

Figure 2.1 [Setup for the Experiment]

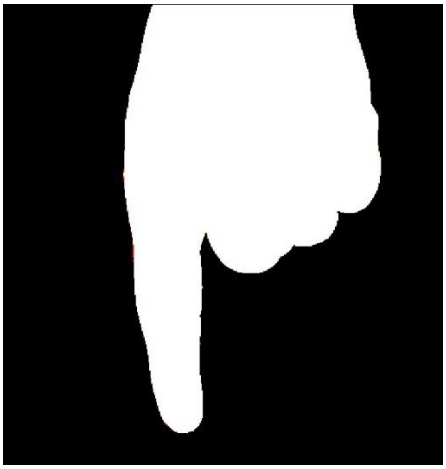

Figure 2.2 [No. 1]

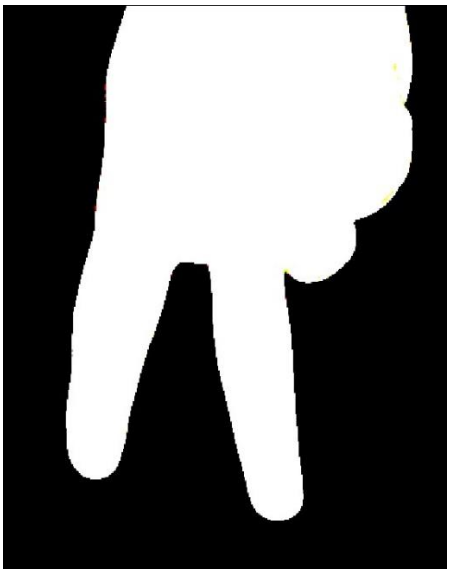

Figure2.3 [No.2]

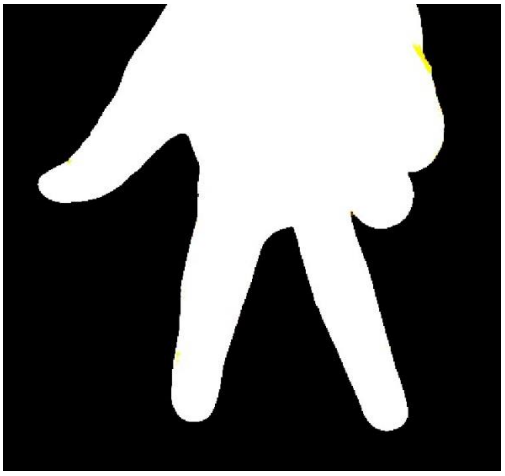

Figure 2.4 [No.3]

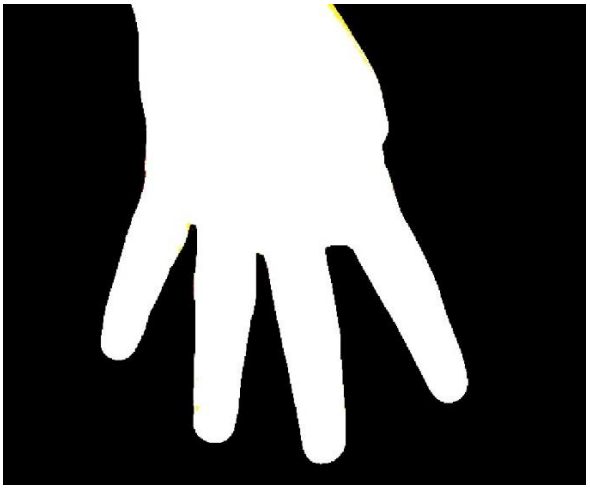

Figure 2.5 [No.4]

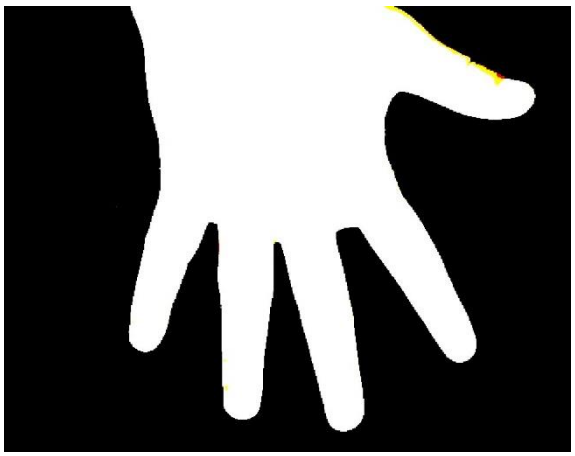

Figure 2.6 [No. 5]

Figure2.7 [Normal hand with the Wrist]

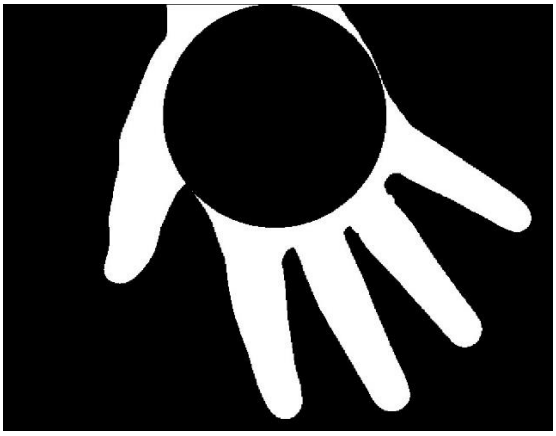

Figure 2.8 [Blob Filter]

a. Conclusion

- Task was difficult to achieve with the camera always turning off and taking much time to calibrate.
- It was difficult for RoboRealm to recognize the no.s sequentially because of the camera.
- It was difficult to maintain the reflections and shadows, We had to maintain the appropriate light for best setup.
- We manually added the Objects(Hand gesture) for the Software to recognize.
- We understood the concept and were able to perform the task.

2.3 Part 3 – Counting dice roll result

- Dice values can be counted using a vision system after they have been rolled and come into frame with a camera.
- Now that we have the Blob Count module, we can count the amount of blobs. In order to display that count into the image, we will finally use the Display Variables module.
- To assist the color in the image blend smoothly, we can add a Mean Module with a value of 2. The filter helps to soften the hue, however we won't notice much of a difference to the unmagnified eye. The RGB Filter module is then used to isolate the image's red only.

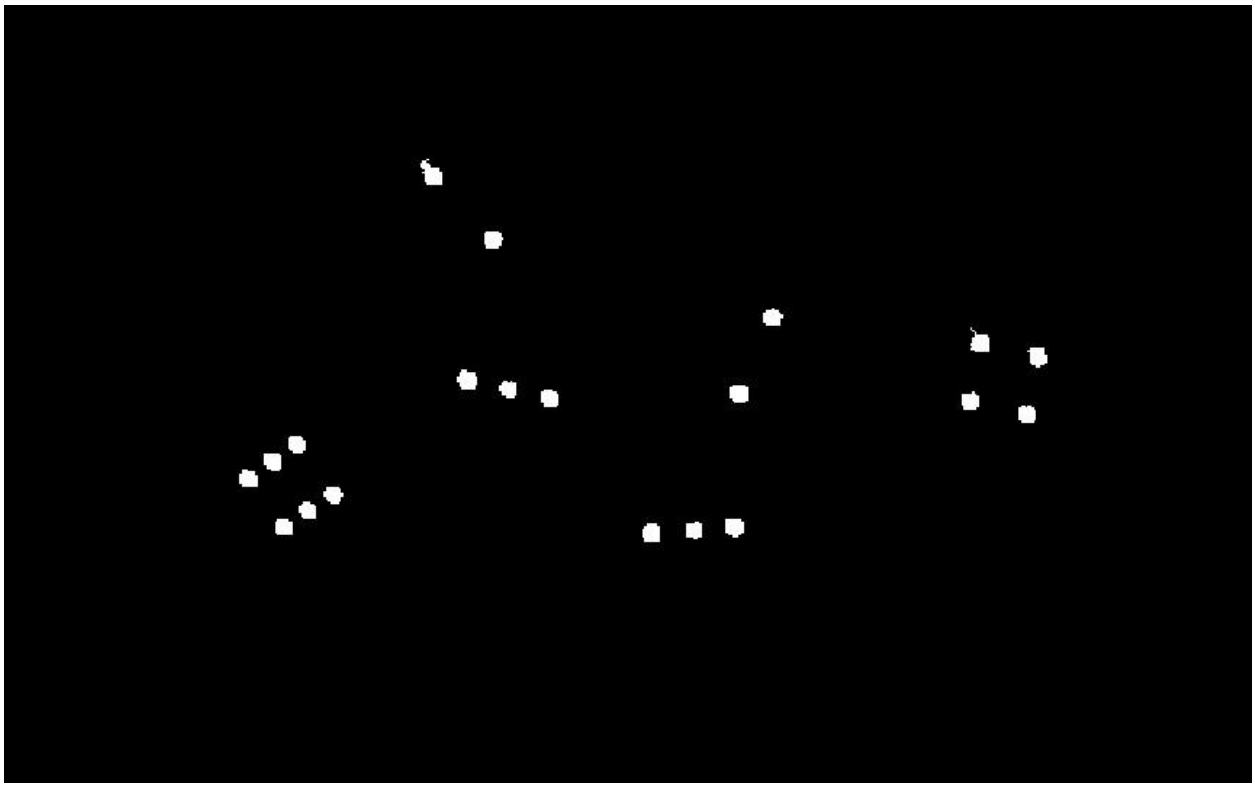

Figure 3.1 [Thresholding Result]

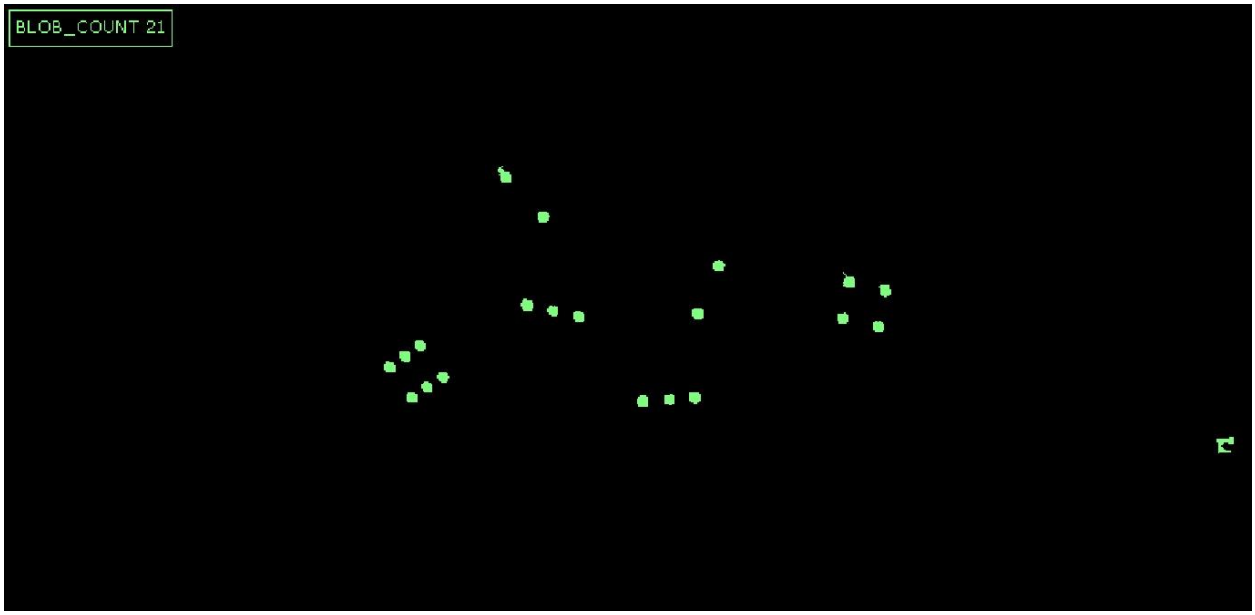

Figure 3.2 [Blob Count Result]

Conclusion

- Due to the Camera not working properly, We could not get live results, but in Theory as well as Static Scenes(images), RoboRealm identified the blobs and Counted them correctly.
- We understood the Concept and were able to complete the task.

4. Summary

- Machine Vision has a variety of opportunities and applications. Machine vision Uses Pixels and Arithmetic Applications to manipulate the Image/Scene and obtain the required data.
- With many issues that could occur from capturing to processing the data, We Focused on a few selected issues of Machine Vision.
- We learned the projectability of RoboRealm. We were able to create scenes and learn how to use the software to identify shapes and blobs and perform Arithmetics.
- We learned a few Limitations and difficulties of Software which depended on the computing and Scene to the scene.

5. Questions

1. This exercise demonstrates the operation of filtering using a temporal frame averaging operation and convolution filtering using a Gauss filter. Explain the differences between these two approaches when minimizing noise in an image. Justify which of the mentioned filtering methods can be recommended for noise filtering in a dynamic scene and which in a static scene.

- Temporal frame averaging is used to get better signal to noise ratio. The current frame is continuously displayed as an image that is created by averaging one or the prior frames of the image data leading to the reduction or decrease in the image noise. Greater reduction in the image noise is obtained by averaging more frames together. - The Gaussian filter which is a type of linear filter is used to blur the image to reduce image noise and detail. It attenuates high frequencies more than low frequencies making the image blurred while sharpening the edges. The gaussian output is the weighted average of neighbouring pixels, and the average is weighted more towards the centre of the pixel preserving better edge information. - Frame averaging is recommended for dynamic scene as the output is the average of the frames over time and reduces the noise (During the experiment it was shown by moving the hand

between the table and camera. The hand appeared slowly over an average of 10 frames). - As for the gaussian filter which is convolution filtering, it requires the original image matrix and the kernel (matrix h) to calculate the convolution to make the resultant image blur. As for a dynamic scene we will have multiple frames changing over time, and if we use gaussian filter for the dynamic scene, the calculations of convolution might not be fully completed for one frame and the original will adapt the next frame, hence missing out on calculations. Thus, static scene is recommended for Gaussian filtering.

2. Explain the differences between the three approaches to thresholding operations: manual (Threshold), automatic (Auto_Threshold), and adaptive (Adaptive_Threshold). What in the analyzed scene affects thresholding errors? How can thresholding operations be improved by selecting scene parameters? How can you numerically compensate for some of the scene setup errors? (2 pts)

Binarization, which is a low order arithmetic operation, involves the assignment of different logical values (0 or 1) to the pixel based on their values. One of the ways of achieving the binarization is thresholding operation that is choosing the pixel intensity threshold above which the pixel will be assigned (0 or 1). - Thresholding operation can be achieved manually, automatically or adaptively. In the manual thresholding operation, we can manually, based on our evaluation of image and choosing what is best for the image, specify the lower and higher thresholding pixel intensity for an image and its thresholds the image by removing values above and below those specified pixel intensities. - As for automatic thresholding, it uses several standard and predefined techniques to set the pixel intensity threshold in a way that the user obtains the best output. There are various automatic thresholding methods such as Kapur and Otsu (Used during the lab). - Adaptative thresholding automatically sets the cut off threshold adapting the inputs of window size and mean offset parameters, thus separating the objects from the background. Its thresholds an image using a locally defined threshold point. - Thresholding errors can be greatly impacted by the scene parameters such as shadows, reflections which are dark and close to the objects making the information of the outer counter less readable for the software, and lightening (Too strong or poorly lit). - There are different ways of improving the scene parameters, and here I will discuss the parameters related to our laboratory. During the lab, initially a piece of white paper was placed to replace the granite background of the worst texture, making the binarization of the image easier. - Later, two lamps were used in order to lit the objects to minimize the impact of the shadows. Also, diffusion of the light was introduced using a paper to decrease the impact of extra pair of shadows that were created by heavily strong light. At this point we had lesser amount of shadows but higher reflections. Finally, the best scene was obtained when the objects were lit from beneath the white paper which introduced neither or very few shadows and reflections and highlighting the outer contours of the object, and thus impacting less on thresholding. - Other errors can be minimized by changing the field of view using the crop option in the software, we can crop the image to show only the background where our objects are. - Temporal averaging, Gaussian blur or median filters can be used to numerically compensate for some scene setup errors by denoising the image.

3. Using the information gained during the exercise, propose a set of operations in RoboRealm program, which will result in displaying on

the screen, in real time, the current result of the dice throw. Justify the choice of each operation and describe the effect you expect from each of them. (2 pts)

- Firstly, we start by preparing the scene by setting the illumination to obtain optimal parameters of the scene. We also make sure the dots on the dice have a good visibility. We prepare the scene to minimize the thresholding errors. - Apply Temporal averaging (Temporal – Average) to denoise the dynamic image. This process will stabilize the flickering of the image. - Apply auto thresholding (Thresholding – Auto_Thresholding) to binarize the image as it will set for us the best cut-off threshold pixel intensity. Here, if the dice pips blob touch other blobs, apply erosion (Morphology – Erode) technique. At this stage we will have dice pips as blobs and the background as blob as well. - Use Blobs – Blob_Filter to delete all the blobs with circular radius of the pips. This procedure will only consider the pips on the dice as the visible blobs (white region(pip) surrounded by the black background (Dice)). - Use Blobs – Blob_Count to count all the visible blobs on the image. To display the information about total number of blobs choose the parameter Display – Display_Variables – Blob_Count and choose the colour of the displaying numbers. Finally, The number of pips or the number on the dices will be displayed on the corner.

6. References

https://www.roborealm.com/help/Object_Recognition.php

https://www.sciencedirect.com/science/article/pii/S187770581202646X

https://repository.najah.edu/items/382ed505-5ffa-4b41-a2ad-dd2d20b1615b

http://www.roborealm.com/tutorial/Counting_Dice/index.php

http://www.roborealm.com

https://www.cs.auckland.ac.nz/courses/compsci773s1c/lectures/ImageProcessing-html/topic4.htm

https://chat.openai.com