ENPM 673 – P1

Name: Hrishikesh Tawade
UID: 116078092

Name: Krishna Bhatu
UID: 116304764

Name: Kapil Rawal
UID:116133357

1) Tag Detection

1.1) Tag corner detection

Our tag corners detection pipeline consists of the following steps:

1. Take frame input from video
2. Convert the frame into grayscale and then binary image
3. Then we find the contours in the binary image
4. We select only those contours which has no parent contour and minimum 2 children. Thus, we get all the white papers on which the tags are printed, and this rejects the external background and all the white objects at the end in the video.
5. We then select the child of the parent contour satisfying a perimeter condition which gives us the tag.
6. We then approximate the contour points using the approxPolyDP function and tuning its epsilon. This reduces all the contours points to just 4 points which are the corners of our tag.

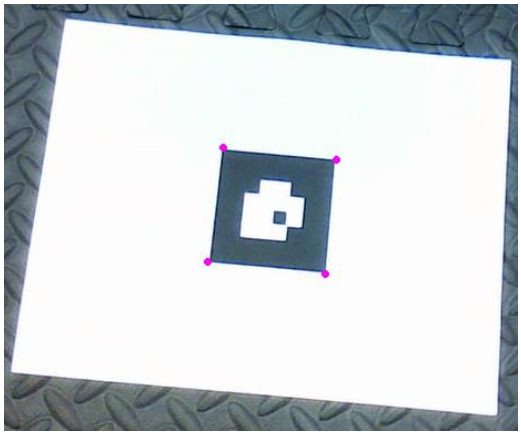

Figure 1. Corner detection

1.2) Tag orientation detection

Our tag orientation detection pipeline consists of the following steps:

1. Calculate the homography matrix between the corners of the tag found in previous step and an upright position calculated based on the width and height of the rectangle formed by the tag contour.
2. Pass this matrix to the warp function to apply the matrix to the image and make it upright.
3. Convert it to gray image and threshold it. Further we smoothen it using a blur and bilateral filter since we want to smoothen the edges but still characteristics
4. Find the contour of this processed image and by applying approxPolyDP we are left with 4 strongest corner points of the region containing encoding of the tag.
5. Find the distance of these 4 points from the 4 corners of the upright image and find the point with the shortest distance which is none other the corner of the white square which denotes orientation.
6. Rotate the tag based on the position of this point in the array and make the tag upright.

Figure 2. Tag rotated after orientation detection

Name: Hrishikesh Tawade
UID: 116078092

Name: Krishna Bhatu
UID: 116304764

Name: Kapil Rawal
UID:116133357

1.3) Tag ID detection

Our tag ID detection pipeline consists of the following steps:

1. From the binary image of the upright tag in previous step calculate 4 corners of the first square of the ID encoding region
2. Based on these 4 points calculate the intensity value of center pixel for all 4 squares and convert it into 1 or 0.
3. Then arrange this 1 or 0 from MSB to LSB and convert to decimal to calculate ID.
4. Then print the ID on the video.

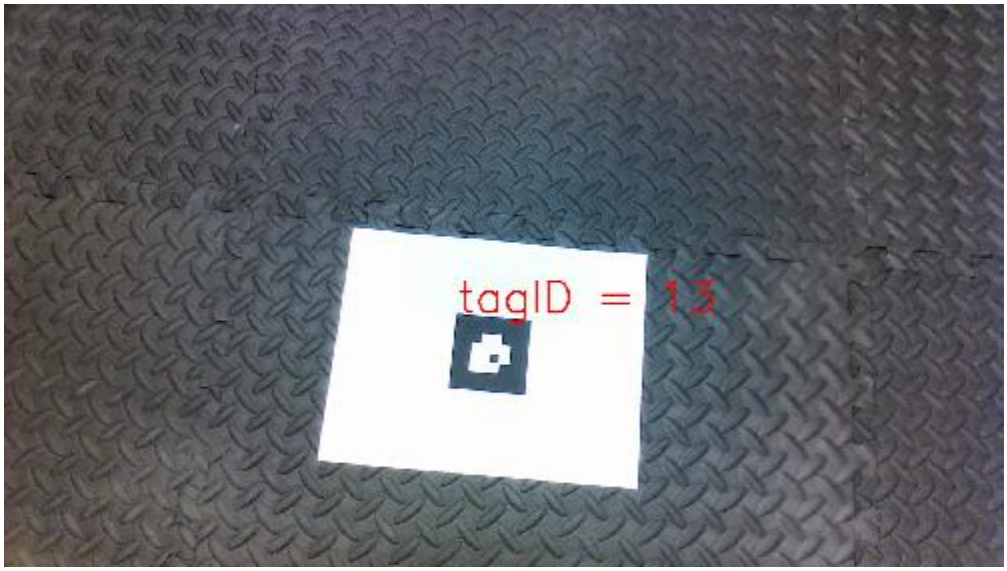

Figure3. Printing of Tag ID on the image

2) Superimposing Image on Tag

2.1) Calculate homography matrix

1. After the tag orientation is detected we calculate the homography matrix between the 4 points of Lena's image (template image) and the corners of the tag detected in step 1.1.
2. Then this homography matrix is passed to the warped perspective function which sticks the transformed Lena's image in a black background of the size of the frame.

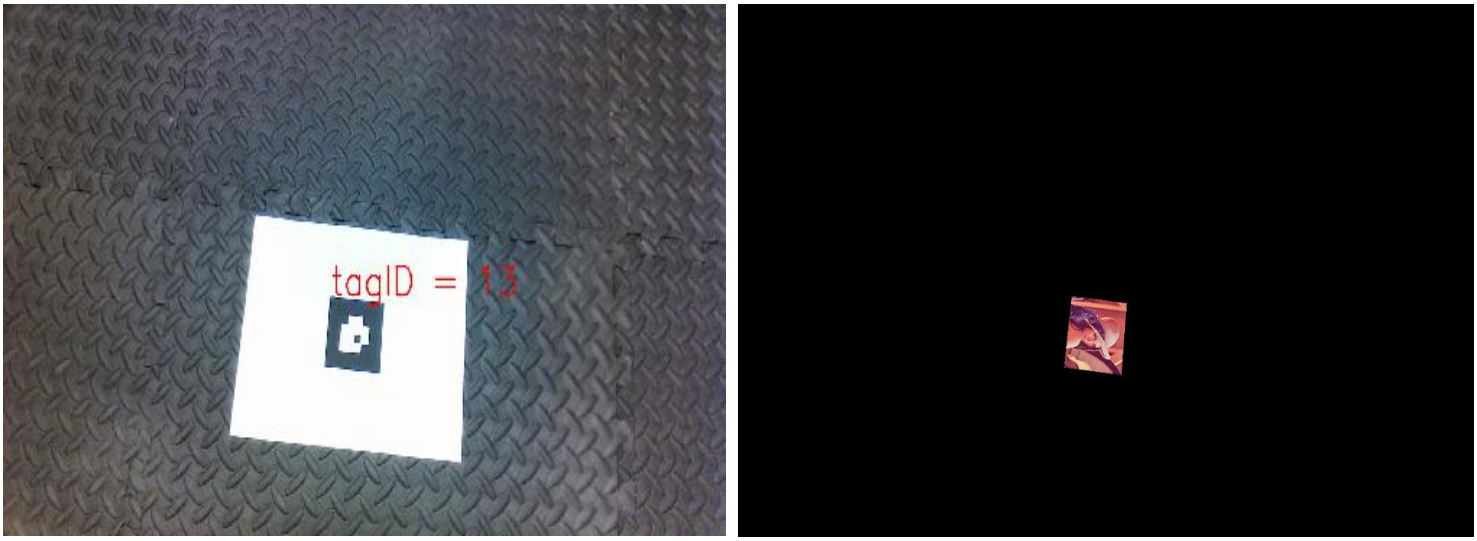

Figure4. Lena's image after application of homography and rotating

Name: Hrishikesh Tawade
UID: 116078092

Name: Krishna Bhatu
UID: 116304764

Name: Kapil Rawal
UID:116133357

2.2) Replace tag with lena

1. Create a mask using the transformed image obtained above by thresholding it.
2. Inverse the mask.
3. We then take bitwise_and of the input frame and the mask inverse.
4. Further we take the bitwise_and of the transformed lena image and the mask.
5. We then add both image and finally we have an image with the tag replaced by lena.

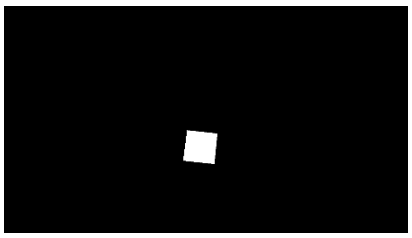

Figure 5: Mask

Figure 6: Inverse Mask

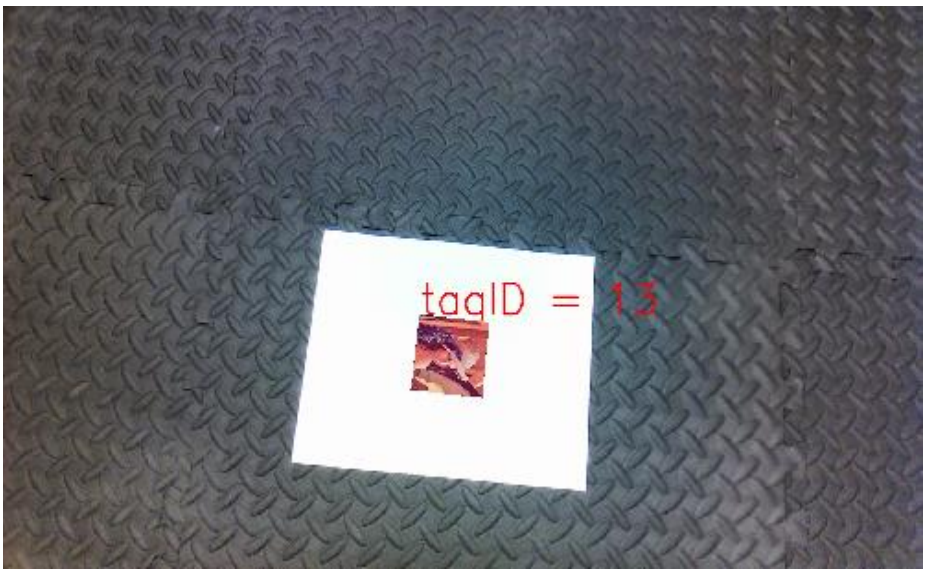

Figure 7. Tag replaced with Lena's image

3) Superimposing Cube on Tag

3.1) Calculate projection matrix

1. We calculate the homography matrix between the 4 points of reference AR marker and the corners of the tag detected in step 1.1.
2. Then from this homography matrix and calibration matrix we calculate r1, r2 and t. Furthermore, r3 = r1Xr2.
3. We append r1, r2, r3 and t together and form the projection matrix.

3.2) Forming Cube

1. We calculate the 4 corners of the top plane of the cube and then multiply it with the projection matrix and find the 4 corners in the image plane.
2. We then form a contour for the bottom 4 points and the top 4 points and draw lines between them.

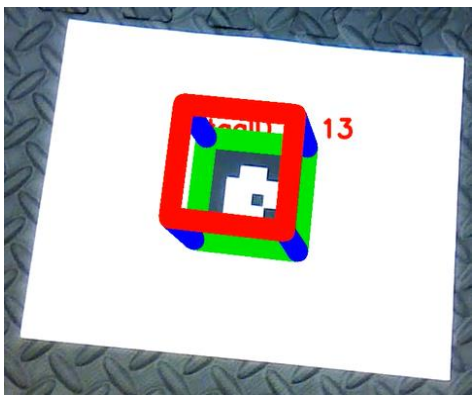

Figure 8. Cube on Tag

Name: Hrishikesh Tawade
UID: 116078092

Name: Krishna Bhatu
UID: 116304764

Name: Kapil Rawal
UID:116133357

4) Test Outputs

4.1) Tag 0 Video

Figure 9. Input Image

Figure 10. Lena on Tag

Figure 11. Cube on Tag

Name: Hrishikesh Tawade
UID: 116078092

Name: Krishna Bhatu
UID: 116304764

Name: Kapil Rawal
UID:116133357

4.2) Tag 1 Video

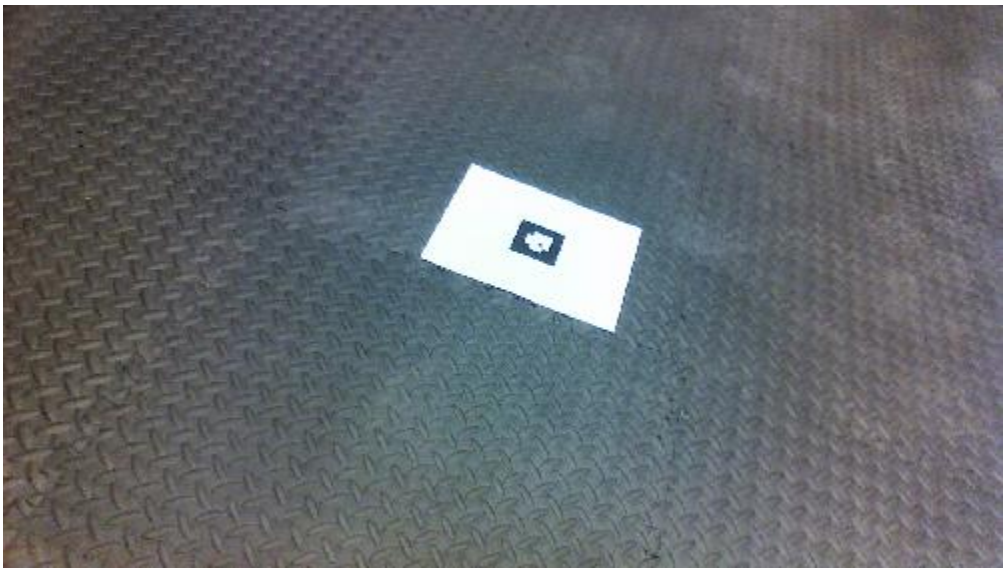

Figure 12. Input Image

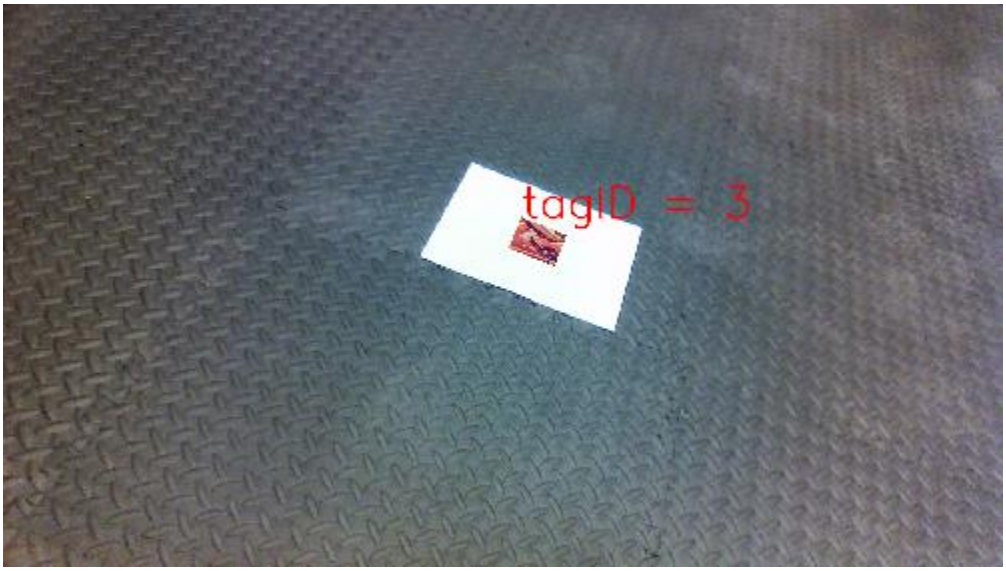

Figure 13. Lena on Tag

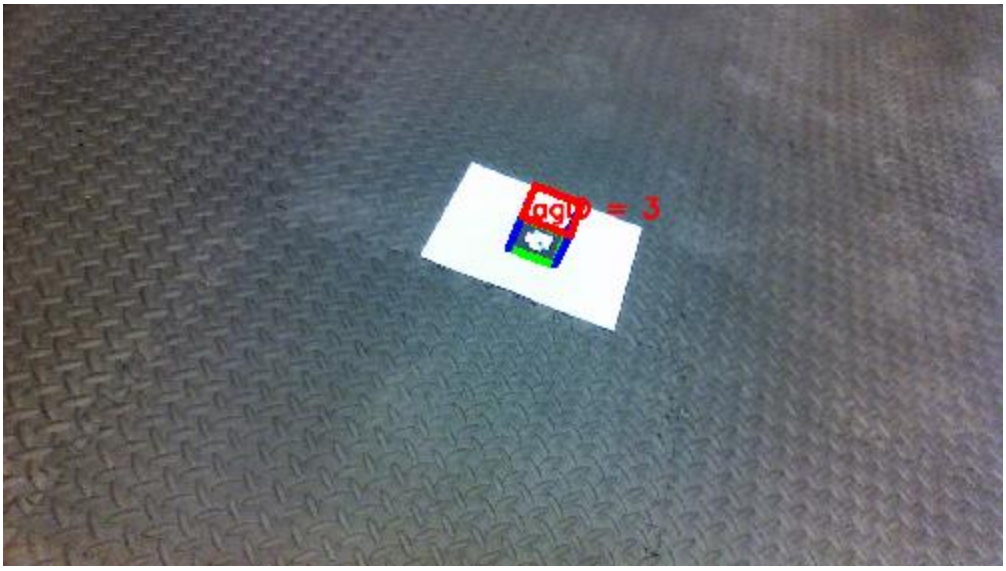

Figure 14. Cube on Tag

Name: Hrishikesh Tawade
UID: 116078092

Name: Krishna Bhatu
UID: 116304764

Name: Kapil Rawal
UID:116133357

4.3) Tag 2 Video

Figure 15. Input Image

Figure 16. Lena on Tag

Figure 17. Cube on tag

Name: Hrishikesh Tawade
UID: 116078092

Name: Krishna Bhatu
UID: 116304764

Name: Kapil Rawal
UID:116133357

4.4) Multiple tags Video

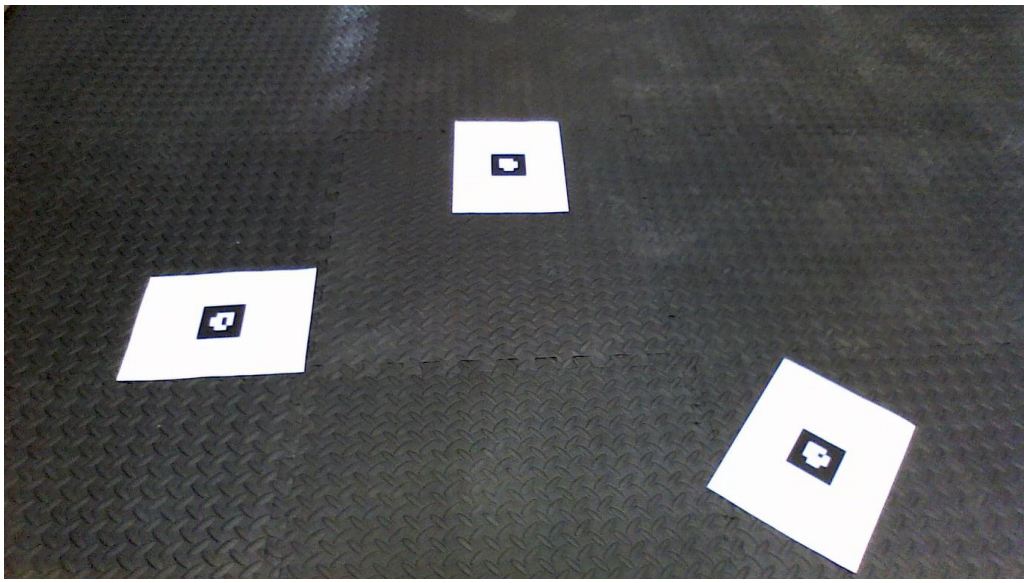

Figure 18: Input Image

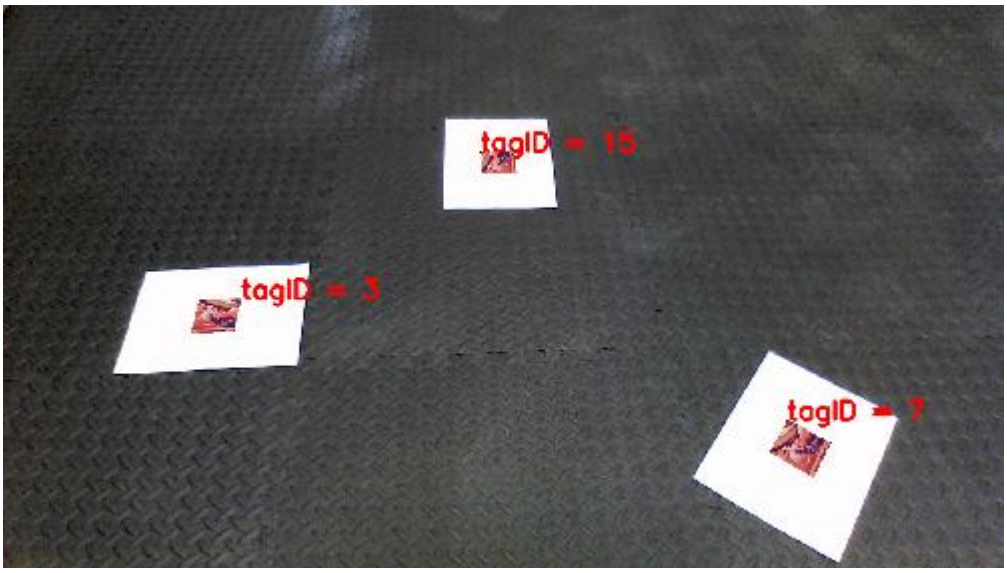

Figure 19. Lena on tag

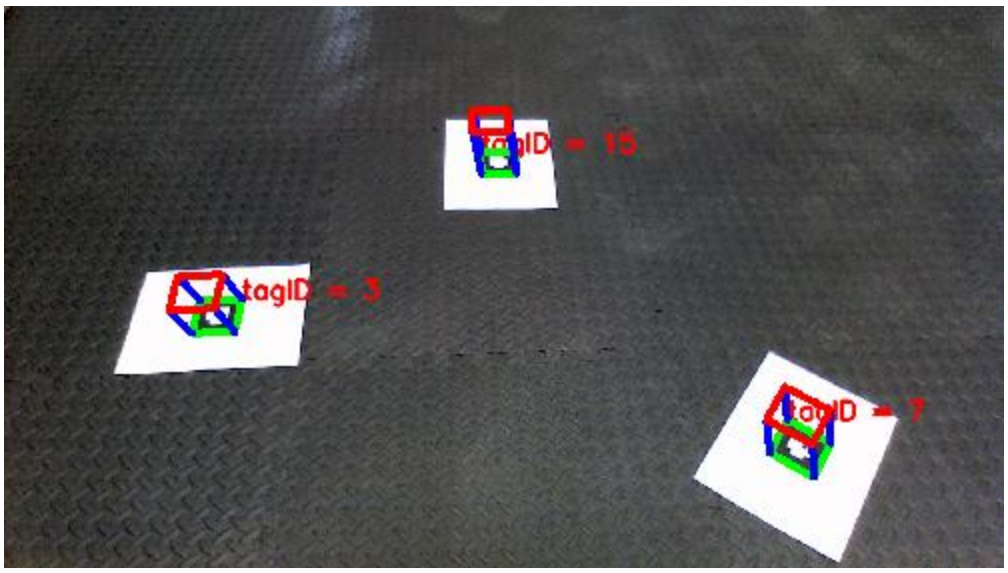

Figure 20: Cube on tag