

CHAPTER 5

IMPLEMENTATION AND TESTING

5.1. Implementation

In this project using the Python programming language using the Haar Cascade algorithm by detecting the presence of a face that uses a mask or does not use a mask. Here's how the program run :

```
1. bw_threshold = 80
2. gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
3. (thresh, black_and_white) = cv2.threshold(gray,
    bw_threshold, 255, cv2.THRESH_BINARY)
```

Line 1 is used for adjusting the light level for black and white. Line 2 is used for converting the image to gray to make it one color intensity. Line 3 is for converting images to black and white which aims to distinguish objects and backgrounds.

```
4. for (x, y, w, h) in faces:
5.     cv2.rectangle(img, (x, y), (x + w, y +h), (255, 255,
    255), 2)
```

In line 4 to detect faces, x, y, w, and h are given which function to mark the detected faces. The detected faces will be rectangle in a way like line 5.

x = coordinate x

y = coordinate y

w = width

h = height

```

6.     roi_gray = gray[y:y + h, x:x + w]
7.     roi_color = img[y:y + h, x:x + w]

8.     eye = eye_cascade.detectMultiScale(roi_gray)
9.     for (ex, ey, ew, eh) in eye:
10.         cv2.rectangle(roi_color, (ex, ey), (ex + ew, ey +
            eh), (0, 255,0), 2)

```

Roi_gray on line 6 to create a numpy variable gray array. While line 7 returns to the original color after getting the image coordination. In line 9, a rectangle is created if it detects an eye object.

```

11.         nose = nose_cascade.detectMultiScale(gray, 1.15, 35)
12.         mouth_rects = mouth_cascade.detectMultiScale(gray,
            1.18, 35)
13.         if (len (nose) == 0 and len(mouth_rects) == 0 ):
14.             cv2.putText(img, 'Mask On', (x, y),
                cv2.FONT_HERSHEY_SIMPLEX, 2,(0, 255, 0), 2)
15.         elif (len (nose) > 0 and len (mouth_rects) == 0 ):
16.             for (nx, ny, nw, nh) in nose:
17.                 cv2.rectangle(img, (nx, ny), (nx + nw, ny +
                    nh), (255, 0, 0),2)
18.                 cv2.putText(img, 'Wrong', (x, y),
                    cv2.FONT_HERSHEY_SIMPLEX, 2,(0, 0, 255), 2)
19.                 print("Image "+str(count)+"saved")
20.                 file='C:/Python/Python38/Foto/Simpan'
21.                 +str(count)+'.jpg'
22.                 cv2.imwrite(file, img)
23.                 count +=1
24.                 break

```

```

25.         else :
26.             for (mx, my, mw, mh) in mouth_rects:
27.                 if (y < my < y + h):
28.                     cv2.rectangle(img, (mx, my), (mx + mh,
my + mw), (0, 0, 255), 2)
29.                     break
30.             for (nx, ny, nw, nh) in nose:
31.                 if (y < ny < y + h):
32.                     cv2.rectangle(img, (nx, ny), (nx + nh,
ny + nw), (255, 0, 0), 2)
33.                     break
34.             cv2.putText(img, 'Mask Off', (x, y),
cv2.FONT_HERSHEY_SIMPLEX, 2, (0, 0, 255), 2)
35.             print("Image "+str(count)+"saved")
36.             file='C:/Python/Python38/Foto/Simpan'
37.             +str(count)+'.jpg'
38.             cv2.imwrite(file, img)
39.             count +=1
40.             break

```

In lines 11 and 12 to call the nose and mouth library functions. On line 13, marking begins with the text "Mask On" if the detection results when the nose is not detected, and the mouth is not detected. Giving the text "Wrong" to detect the presence of a nose but not detecting a mouth is listed on line 15. Giving the text "Mask Off" if the nose and mouth are detected is listed on line 25. In line 19 and 35 are given a print to capture photos if the text appears " Wrong" and "Mask Off", the image will be saved to the computer folder.



Figure 15 The Result of the Image Capture Not Using a Mask

5.2. Testing

The test is carried out as when someone wants to enter the place in public. Where the test uses one face first then with two faces with the following results.

Table 1 Positive and Negative Prediction

	Positive Prediction	Negative Prediction
Positive Class	True Positive (TP)	False Negative (FN)
Negative Class	False Positive (FP)	True Negative (TN)

True Positive if the program successfully detects using a mask if the person is indeed wearing a mask, while True Negative is when the program does not detect a mask if the person is not wearing a mask. False Positive if the program detects wearing a mask even though the person is not wearing a mask. While False Negative is if the program detects not wearing a mask even though the person is wearing a mask.

From the data obtained, there are 50 images using masks, 50 images not using masks, and 25 images not using masks correctly with 17 images not detection faces with the following results:



Figure 16 Video Data Used

Table 2 Result Positive and Negative Prediction

	Positive Prediction	Negative Prediction
Positive Class	31 Data	6 Data
Negative Class	11 Data	60 Data

$$\begin{aligned}
 \text{Precision} &= \frac{TP}{TP+FP} \\
 &= \frac{31}{31+11} \\
 &= \frac{31}{42} \\
 &= 0,73 = 73\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Recall} &= \frac{TP}{TP+FN} \\
 &= \frac{31}{31+6} \\
 &= \frac{31}{37} \\
 &= 0,83 = 83\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Accuracy} &= \frac{TP+TN}{TP+TN+FP+FN} \\
 &= \frac{31+60}{31+60+11+6} \\
 &= \frac{91}{108} \\
 &= 0,84 = 84\%
 \end{aligned}$$

The test is carried out with 4 types of data so that the results for testing accuracy, precision, and recall are as in the table below:

Table 3 Result from the Test

Test	Data	Accuration	Precision	Recall
Test 1	person using a mask	79%	69%	80%
Test 2	person not using a mask	79%	78%	86%
Test 3	More than 1 person using a mask	100%	100%	100%

Test 4	More than 1 person not using a mask	100%	100%	100%
--------	-------------------------------------	------	------	------

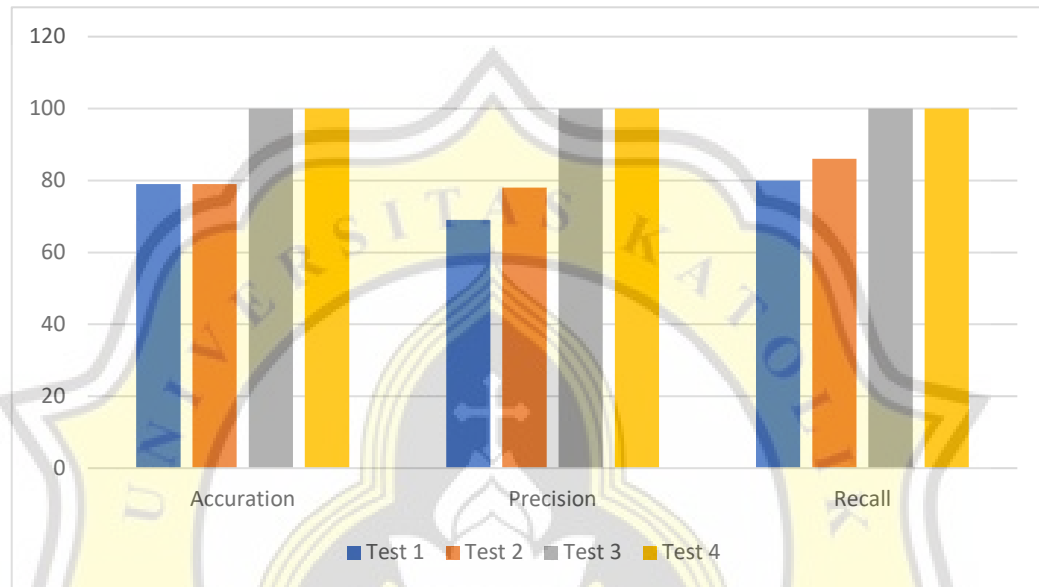


Figure 17 Chart Results for Accuracy, Precision, and Recall

From testing the 4 types of data, again the values of accuracy, precision, and recall are obtained. The average accuracy is 89,5%, the average precision is 86,7% with the highest precision 100%, and the recall average is 91,5% with the highest recall 100%. The highest precision and recall are not necessarily 100% because some images are not detected by the face.

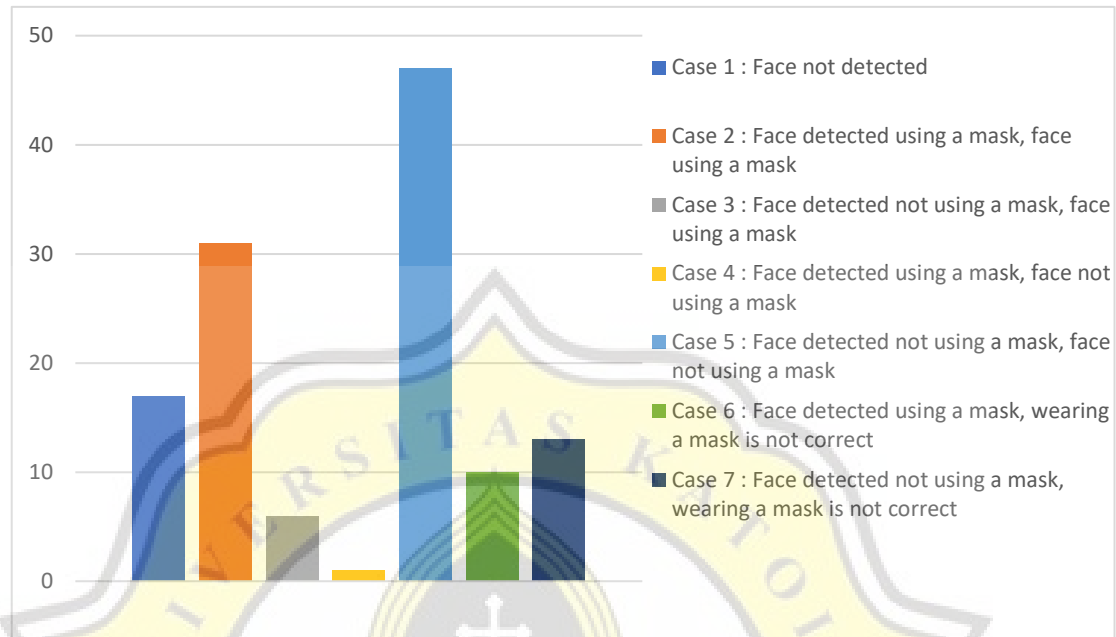


Figure 18 The Results of the Test

From the test above with 125 data, 7 cases were obtained. Case 1 with no face detected there are 17 data. Case 2 faces using masks and detected using masks there are 31 data. Case 3 faces using masks and not detected using masks, there are 6 data. Case 4 faces do not use masks but are detected using masks, there is 1 data. Case 5 faces do not use masks and are detected not using masks, there are 47 data. Case 6 faces do not use masks correctly but are detected using masks, there are 10 data. Case 7 faces do not use masks correctly and are detected not using masks, there are 13 data.