CHAPTER 5

IMPLEMENTATION AND TESTING

5.1 Implementation

This Project using Python Programming Language. First step this project use color picker for analyze yellow color of jaundice, to gained right information of color we gonna test for this program to detect jaundice

```
1.  def masking(event,x,y,flags,param):# fungsi mouse callback
2.     if event == cv2.EVENT_LBUTTONDOWN:
3.           pixel = imageHSV[y,x]


4.     atas =  np.array([pixel[0] + 20, pixel[1] + 20, pixel[2] +
    40])
5.     bawah =  np.array([pixel[0] - 20, pixel[1] - 20, pixel[2] -
    40])
6.           print(pixel)


7.           mask = cv2.inRange(imageHSV,bawah,atas)
8.           cv2.imshow("mask",mask)
```

In the first step this program try to finding yellow jaundice color and get information on which areas will be masked through the color we choose. The program uses the mouse callback function, to get the color values for a specific area. In line 3 pixel = imageHSV which is the program will work if image already convert to hsv first so we can get the right information,line 4 and 5 is to set of area where to mask after we pick the specific area. Line 6 is the area that we select, then this line going to print the color values of the area, in line 7 is using threshold to get mask area that we chose, line 8 is showing the result

```
9.  def main():
10. global imageHSV, pixel #supaya kita bisa menggunakan mouse
11. callbackgambar = cv2.imread('4.jpg')

12.   if gambar is None:
                print ("tidak ada gambar")
                exit(1)

13.   elif gambar is not None:
          w, h = gambar.shape[:2];#melakukan scaling pada gambar

                x = int(w * 4);
                y = int(h * 4);

                xScale = x/(w-1);
                yScale = y/(h-1);

14. gambarbaru = np.zeros([x, y, 3]);#mengisi array baru dengan 0

15.         for i in range(x-1):
                for j in range(y-1):
                        gambarbaru[i + 1, j + 1]= gambar[1 +
int(i / xScale),
                                                                1
+ int(j / yScale)]
16. imageHSV = cv2.cvtColor(gambar,cv2.COLOR_BGR2HSV)
```

In processing we call the image, if the image is none then the program will print("tidak ada gambar") then exit, if image is exist then image going through scaling process (line 13 until 15) after that we convert the image into hsv so we can back to pre processing to get information about value of yellow jaundice, after we got the yellow jaundice color then we set range color of yellow jaundice like below

```
17. min = np.array([15,150,50],np.uint8)
18. max = np.array([30,255,255],np.uint8)

19.    areaKulit = cv2.inRange(imageHSV,min,max)

20. kernel = np.array([[0, -1, 0],
                       [-1, 5, -1],
                       [0, -1, 0]])

21. contours,hierarchy = cv2.findContours(areaKulit, cv2.RETR_TREE,
    cv2.CHAIN_APPROX_SIMPLE)

            for i, c in enumerate(contours):
                  kontur = cv2.contourArea(c)
                  if kontur > 1000:
            cv2.drawContours(gambar, contours, i, (255, 255, 0), 3)
```

After we found range color of yellow jaundice skin. The image will enter the thresholding process where the value range of the yellow disease color is compared to the original image, after that image will be process into convolution kernel, then if the program detects yellow skin areas, the image will enter the next process (line 21) where the area exposed to jaundice will be drawn contour lines, we set the contour> 1000 so that the area around it can be drawn properly, After that the result will be shown and print.

5.2 Testing

Based on the test that have we done as much as 8 times to our 100 data set where is a mix between jaundice, healthy people, and a combination of the two, as well as of various races and ages the result work the as like below.

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

Table 5.1:table positive and negative predicition

True positive is classified if the program is successful in detecting jaundice if the data is indeed a jaundice patient, while **true negative** is when the program does not detect jaundice to the data that it is not a jaundice patient or a healthy person. **False positives** are classified if the program detects jaundice even though the data being tested are healthy people, while **false negatives** are the opposite of false positives, so it is classified if the program does not detect jaundice even though the data is a jaundice patient.

From the 100 datasets where 53 data are people with jaundice, 40 data are normal people, while the remaining 7 are combined data between the two datasets. I have tested the entire data set 3 times and the program shows the same results. The results are as follows

	Positive Prediction	Negative Prediction
Positive Class	(TP) = 50 data	(FN) = 3 data
Negative Class	(FP) = 2 data	(TN) = 38 data

Table 5.2: Table result positive and negative predicition

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$= \frac{50}{50 + 2}$$

$$= \frac{50}{52}$$

$$= 0.9615 = 96\%$$

$$\text{Recall} = \frac{TP}{TP+FN}$$

$$= \frac{50}{50 + 3}$$

$$= \frac{50}{53}$$

$$= 0.9433 = 94\%$$

$$\text{Accuracy} = \frac{(2 \text{ x Precision x Recall})}{(\text{Precision + Recall })}$$

$$= \frac{(2 \text{ x } 0.9615 \text{ x } 0.9433)}{(0.9615 + 0.9433)}$$

$$= \frac{1.8141}{1.9049}$$

$$0.9523 = 95\%$$

The test above was carried with 8 type data so as to produce test results accuracy, precision, and recall as in Table below:

Test	Data Type	Accuration	Precision	Recall
Test 1	Mix	95%	96%	94%
Test 2	Age (Child)	94%	94%	94%
Test 3	Age (Adolescent)	89%	88%	91%
Test 4	Age (Adult)	93%	100%	88%
Test 5	Age(Elder)	88%	80%	100%
Test 6	Race(Asia)	90%	90%	90%
Test 7	Race(Negro)	93%	100%	87%
Test 8	Race(Caucasian)	95%	97%	94%

Table 5.3:Result table of Fmeasure

Table 5.4:Result sort by age

Then we tested 5 times from the age type data with a different sample image with same amount of image in each of the five trials , it can be concluded that the first trial to the fifth trial the children had an average success rate of about 94%, while adolescents in the first trial got 90% and in the 2nd and 3rd trials 94 % and the fifth trial got 95%, while adults in the first trial got 89% and the

second trial got 94% then in the third trial it fell to 93% and returned to 94%, then the last data is parents in the first and second trials it got 94% then in the 3rd to 5th trials it succeeded in being at 100%, while the average 1 to 5 trials using age data types got 94.6%

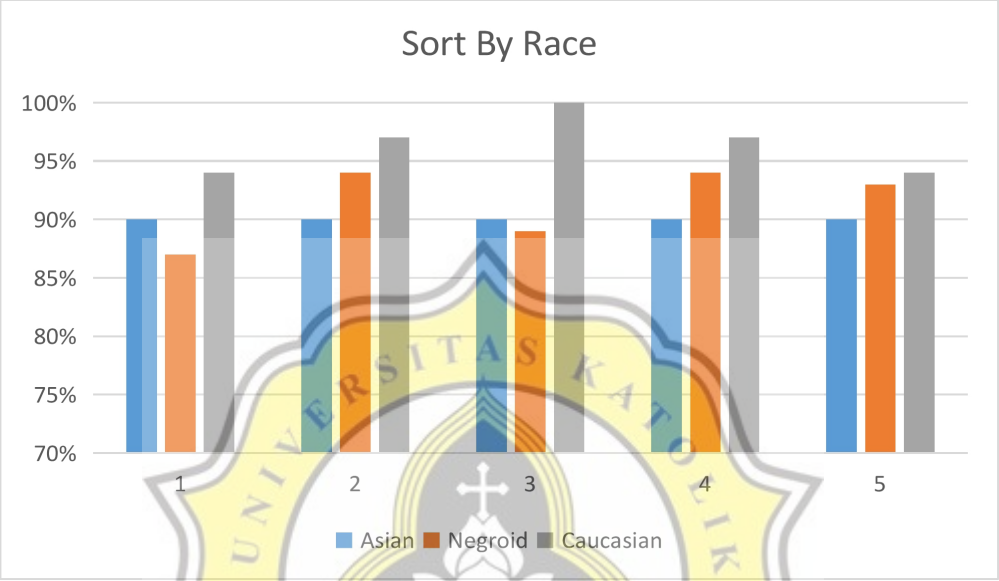

Table 5.5:Sort by Race

After we tried the age data type, then we tried the race data type 5 times as well with a different sample image in each of the five trials,the following results the program managed to consistently detect Asian skin around 90% while in the Negroid race, the program was quite difficult to test first with 87% accuracy then increased in the second trial about 94% then fell again in the third trial 89% and managed to survive around 94% and 93% in the 4th and 5th trials, while for the Caucasian race in the first try it was around 94% then it increased in the 2nd trial around 97% even in the third trial the program was able to detect about 100% and returned to 97% in the 4th and 5th trials. The result is that the average success rate in race data types is around 92%

We also test program by 2 categories (Disease and non Disease) with 7 data type using Sensitivity (Non Disease) and Specificity (Disease) which is show in the table and graphic below:

$$\text{Specificity: } \frac{TN}{\text{Total non Disease}} \times 100 \qquad \text{Sensitivity: } \frac{TP}{\text{Total Disease}} \times 100$$

Data Type	Sensitivity	Specificity
Age (Child)	94.7	100
Age (Adolescent)	91.6	100
Age (Adult)	100	90.9
Age(Elder)	100	100
Race(Asia)	90.90	96.42
Race(Negro)	87.5	100
Race(Caucasian)	100	100

Table 5.6:Sensivity and Specificity

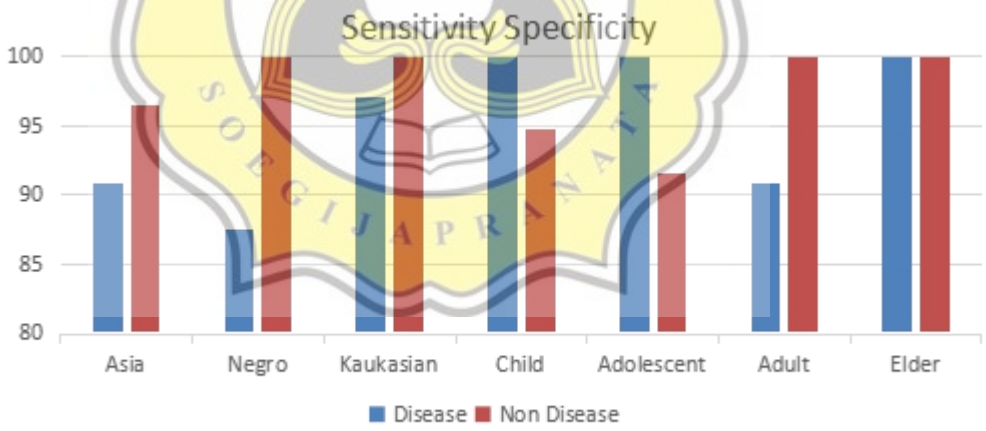

Table 5.7:Graph of Specificity and Sensivity

From the data table above, it can be concluded that this program has succeeded in detecting the elderly with a 100% accuracy rate for both disease and non-disease, compared to adolescents whose accuracy rate is 91.6% in detecting

jaundice, 100% for non-disease, while in the category of children the level of accuracy is equal to 94.7% disease and 85.7% non disease, and the last category is adult with an accuracy level of around 100% disease and 90.9% non disease, Background and image quality affect program performance,This program has also succeeded in detecting white people (Caucasians) with a 97% accuracy rate for disease and 100% non-disease, compared to Asia where the accuracy rate is 90.90% disease, 96.42% non-disease, while in the Negroid category (Black skin) the accuracy level is equal to 87.5% disease and 100% non-disease, Background and image quality affect program performance in detecting data.

5.3 Comparison with existing project

There is existing project Detecting Jaundice by Ashish Sardana. This project using his proposed algorithm with using YCbCr as color space and logistic regression as classification method to detect jaundice,while in my project using color detection with HSV as color space and Kernel Method as classification method

Figure 5.8:flowchart of proposed algorithm by Ashish Sardana

This project does not have approach to find the yellow color value from jaundice, so it has a huge impact on the final detection result, which results in the accuracy rate of this project being only about 68% claim in their journal, compared to the program I made with an average accuracy rate of 92% due to this project set the color range with the average value of yellow color in general,and only hold on to the removal of Y compound on YCbCr when the yellow color of human skin and yellow color is very different value.

then this algorithm will be tested based on the data set that I have used in my project by using 100 data with 8 type data set where the data contains jaundice patients, healthy people, and a combination of the two, with the following results

	Positive Prediction	Negative Prediction
Positive Class	(TP) = 27 data	(FN) = 40 data
Negative Class	(FP) = 33 data	(TN) = 0 data

Table 5.8:Result of positive and Negative prediction existing project

Then we calculate with f measure the result is like below:

Test	Data Type	Accuration	Precision	Recall
Test 1	Mix	42%	45%	40%
Test 2	Age (Child)	49%	47%	53%
Test 3	Age (Adolescent)	46%	46%	62%
Test 4	Age (Adult)	26%	37%	21%
Test 5	Age(Elder)	54%	60%	50%
Test 6	Race(Asia)	22%	30%	18%
Test 7	Race(Negro)	70%	75%	66%
Test 8	Race(Caucasian)	46%	38%	63%

Table 5.9:Result of Fmeasure existing project

From the results of the tests that have been carried out, it can be concluded that this existing project created by Ashish Sardana is not really able to detect jaundice, but only detects areas of skin we can see through first table where this program cant able to see different sample of jaundice nor negative sample where true negative is 0 ,after we count with f measure this program only got an average result below 50%.