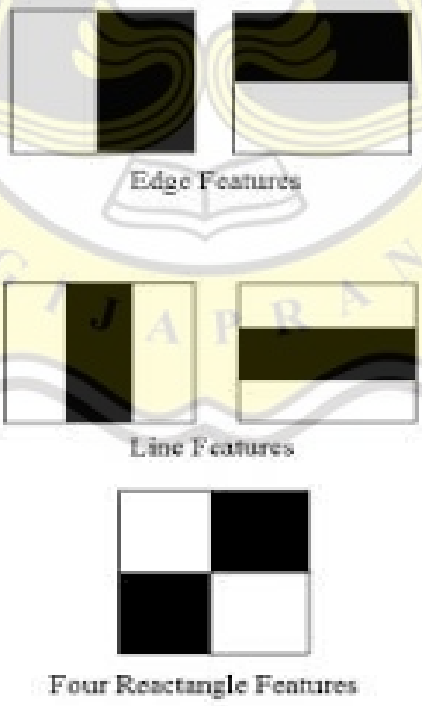
CHAPTER 4
ANALYSIS AND DESIGN

4.1. Analysis

This chapter provides a detailed explanation and discusses the troubleshooting methods that have been mentioned in several points in chapter I. The main objective of making this project is to be able to detect someone who is wearing glasses or not by using video input directly, and to apply the haarcascade classifier method in detection.

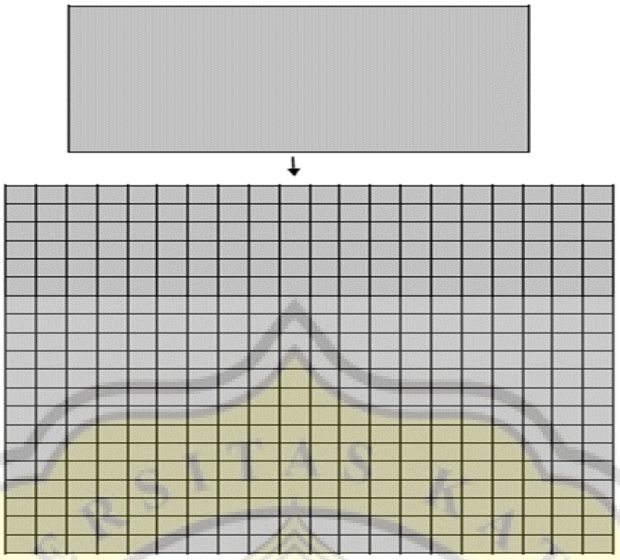
Based on the problem formulation that has been described, the first is how the haarcascade classifier algorithm works to detect someone who is wearing glasses or not. In that case, it is necessary to discuss how the haarcascade classifier algorithm works itself. As described below:

1. Convert video directly to grayscale.
2. After converting the video to grayscale, then use the edge feature, line feature, four rectangle feature which is used to detect features on the face and eyes.



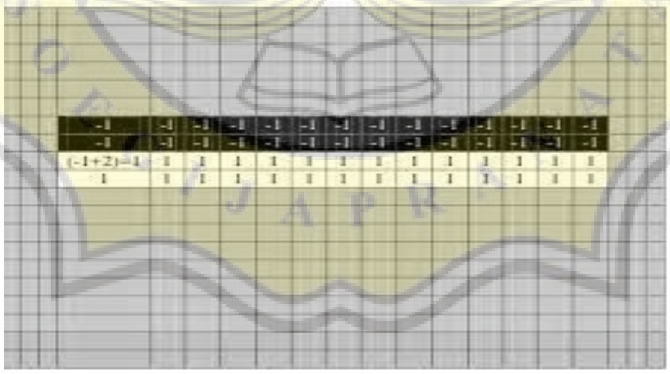
Gambar 4.1: Haar-Like Feature

3. The next stage is the formation of a 24x24 sub-window according to the contents of the haarcascade classifier dataset. As follows:



Gambar 4.2: Haarcascade Sub Window Formation

4. After the 20x20 sub-window has been created, the next step is to determine the haar features at stage 0 and tree 0. Black pixels are with feature value = -1, while white pixels are pixels with feature value = (-1+2) = 1. Below are examples of haar features at stage 0 and tree 0:

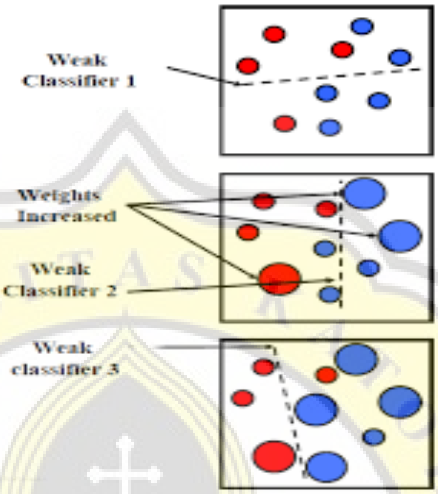


Gambar 4.3: Haar Features At Stage 0

5. After determining the haar feature at stage 0, the next step is to calculate the pixel value using an integral image and compare the pixel values in the light and dark areas.

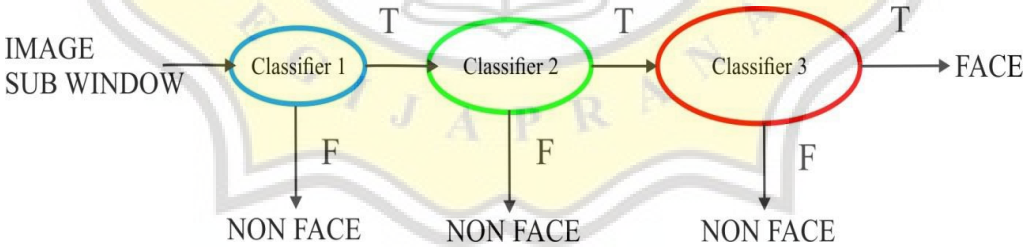
If there is a difference in pixel values in the light and dark areas, it is called the threshold value, the threshold value is declared to have features or objects in the video.

6. The next stage is to apply the adaptive boosting technique to combine many weak classifiers to form a good classifier combination, the results of this adaptive boosting produce strong classifiers and basic classifiers.



Gambar 4.4: Adaptive Boosting Classifier

7. After generating a strong classifier and a basic classifier, the next step is to develop a stratified classification that aims to determine whether there are really facial and eye features in the video.



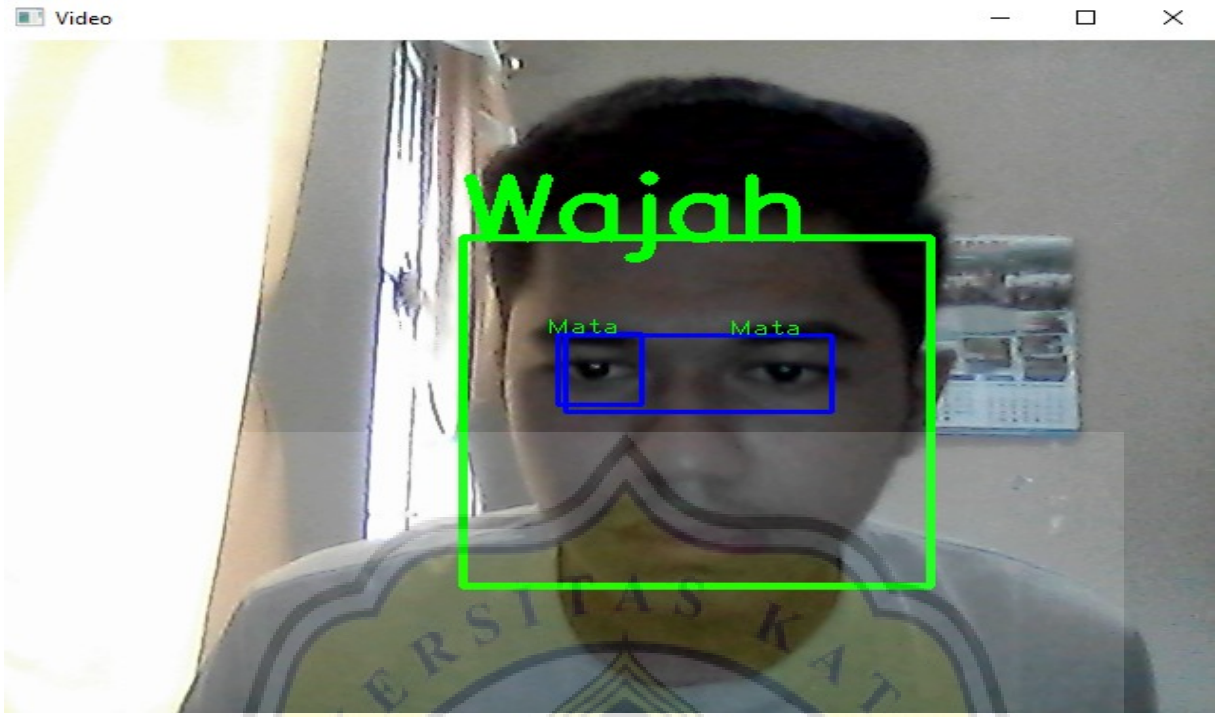
Gambar 4.5: Cascade Classifier

8. If the face and eye detection stage in the cascade classifier has been carried out, then the next step will determine the results of a person using glasses or not, if the face and eyes are detected, it can be ascertained that the person is not wearing glasses.

Meanwhile, if only the face is detected then the person is wearing glasses. The following are the results of face and eye detection that have been carried out:

Gambar 4.6: Face Detect

Figure 4.6, the results of detecting a person's face have been carried out with a haar-like feature process that functions to determine black pixels and white pixels on each face object, then processed using integral image techniques accompanied by adaptive boosting. In the last stage, a cascade classifier technique is applied which functions to determine face objects in the video directly. In determining the object of a person's face, the program will form a green square column around the face area along with face writing at the top of the square column.



Gambar 4.7: Detect Face And Eyes

Figure 4.7, the results of detecting a person's face and eyes have been carried out through a haar-like feature process that functions to determine black pixels and white pixels on each face object or eye object. Furthermore, it is processed using an integral image technique along with an adaptive boost technique. In the last stage, applying the cascade classifier technique which serves to determine the face and eye objects in the video directly. In determining a person's face and eyes, the program will form a green square column around the face area and a blue square column around the eye area, along with face and eye writing at the top of the square column.

The second problem formulation discusses the advantages of the haarcascade classifier algorithm or method in its use in real time video. In the case of detecting objects as well as faces and the human eye using the haarcascade classifier method in real time video, it has advantages as described below:

1. The haarcascade classifier algorithm is able to detect quickly because it only depends on the number of pixels in the square of an image or video.
2. The haarcascade classifier algorithm can detect objects in real time and accurately because real time videos are processed using the haar-like feature to compartmentalize

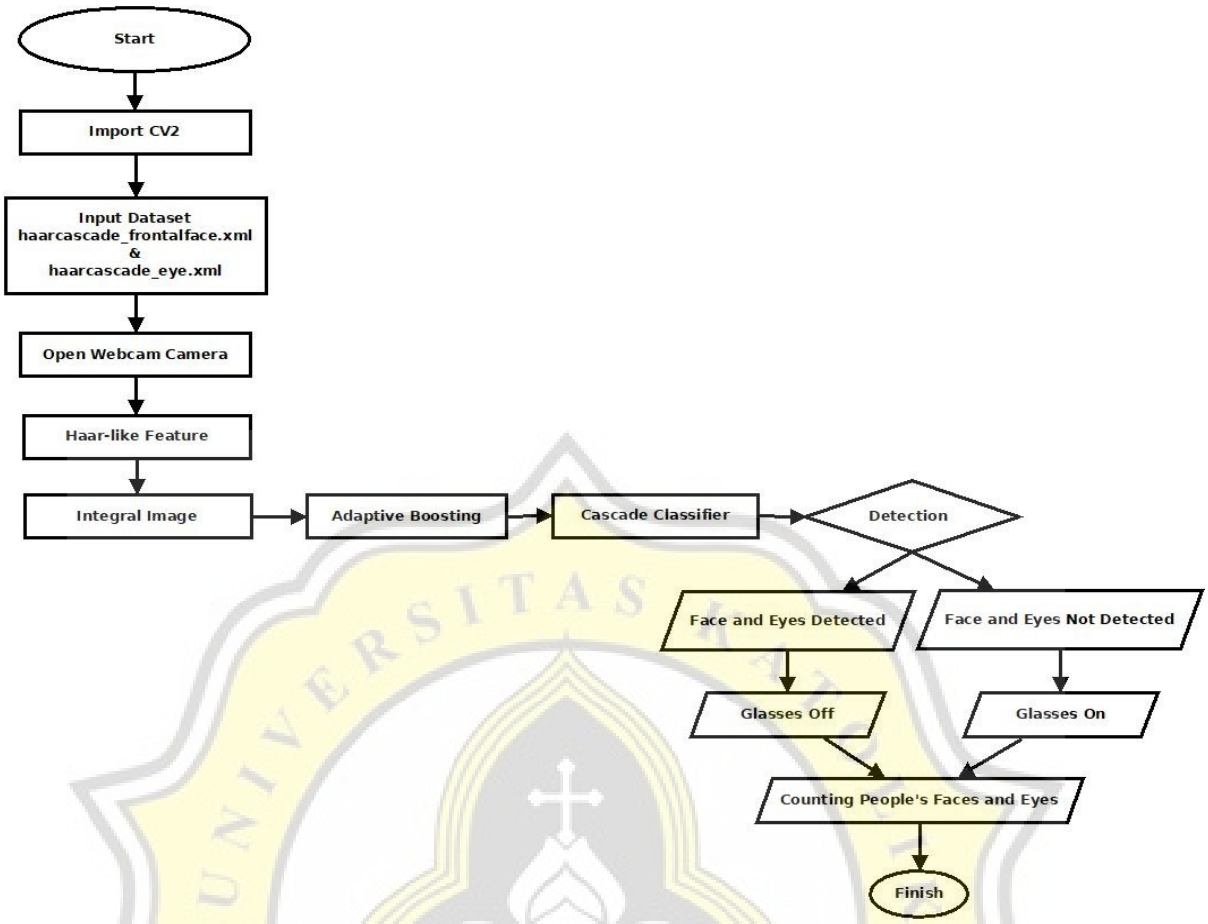
each area on the video which aims to find whether there are objects in the video, after going through the haar-like feature, it is processed using integral image technique that aims to calculate the pixel value in the video, then applies an adaptive boosting technique that aims to produce a strong classifier, and the last is processed using a cascade classifier technique which aims to measure the accuracy of object detection on video.

3. The haarcascade classifier algorithm can detect objects in real time using a pixel camera that is not very good, for example a laptop webcam camera.

The last problem formulation discusses the success rate of detecting someone who uses glasses using live video by applying the haarcascade classifier method. In this project, the success rate can be stated as good because it can detect a person's face and eyes using the laptop's built-in webcam which has less high pixels. And in doing the detection it must be with bright lighting conditions so that the program can run smoothly, as well as detecting someone who uses glasses can only be detected from the front view of the person's face while from the side it can't detect glasses.

4.2. Desain

The use of flowcharts aims to determine the process or procedure of a program so that it can make it easier to understand the program to be run. The system flowchart can be described as follows:



Gambar 4.8: Flowchart Project

Flowchart 4.8 is used as an illustration or flow of this project system. In general, the glasses detection system using the haarcascade classifier algorithm does not directly detect the glasses but goes through the stages of detecting the face and eyes first as below:

1. **Import CV2** : OpenCV is used to process data in the form of images or videos, while CV2 is used for face detection.
2. **Input Dataset haarcascade_frontalface.xml and haarcascade_eye.xml** : haarcascade_frontalface.xml is used to detect someone's face, while haarcascade_eye.xml is used to detect someone's eyes.
3. **Open Camera Webcam** : opens the laptop's webcam camera to detect the person's face and eye objects.

4. **Haar-like feature** : the video will change the scale to gray and select facial features and eye features and then determine the pixel value in each black and white box even though the process is not visible.
5. **Integral image** : calculate the difference in pixel values in each black box and white box and get a threshold value which can be interpreted as an object in the video.
6. **Adaptive Boosting** : combining weak classifiers to make a better classifier, and the adaptive boosting process produces a strong classifier.
7. **Cascade Classifier** : multilevel classification that aims to detect the accuracy of a person's face and eye objects in the video until it is successful.
8. **Face And Eyes Detected** : the object of a person's face and eyes is detected, it is certain that the person is not wearing glasses, and the output produced is glasses off.
9. **Face And Eyes Not Detected** : only the object of a person's face is detected while the eyes are not detected, it is certain that the person is wearing glasses, and the output produced by glasses is on.
10. **Counting People Face And Eyes** : count the number of faces and eyes of the person in the video.