CHAPTER 5
IMPLEMENTATION AND TESTING

5.1 Implementation

This chapter five describes implementation and testing where the implementation would discover about the code and the explanation and the testing would discover about trial result of algorithms and data structures.

```
1. bounding_boxes, _ = detect_face.detect_face(frame, minsize, pnet,
   rnet, onet, threshold, factor)
2. nrof_faces = bounding_boxes.shape[0]
3. print('Detected_FaceNum: %d' % nrof_faces)
```

As we see on the list above, first line means the code is use to detect the face by going to detect_face file and detect_face class to start the face recognition. After they got the face then it counted how many face are detected in the frame.

5.2 Testing

In this sub-chapter, the project need about 1 month until this project get the last result. First week is the beginning of the step that need to learn what is Convolutional Neural Network and all the part's that useful for face detection. After done by research and learning about CNN, the next day I buy a CCTV cause the main CCTV at the house are not compatible and no one can operate to extract the video from the CCTV. After buy the CCTV, there's a problem because it can't be used for the first installation. At the same day I go to the store to get the warranty, the staff said that we need to change the location into China so the CCTV can be detected by the apps.

The next step after the CCTV can be used, we wait for 3 days until the CCTV and also looking for the references code for face detection using CNN as the method. After get the video extraction, then start to implement the code that already get from the github and many other

source. After done with the 3 days of trial, then I try to extend the CCTV result until 2 weeks. Here the result of the CCTV

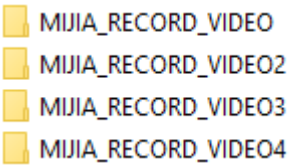


Illustration 5.1: Dataset from CCTV

After having some analysis and implementation, I do some testing which the purpose is to answer the problem of chapter one. Here are some testing that done by me as the researcher. First testing is how I can detect the face from full picture until become only face that seen or become the output of the dataset.

Here are some picture of the dataset that I use for this research:

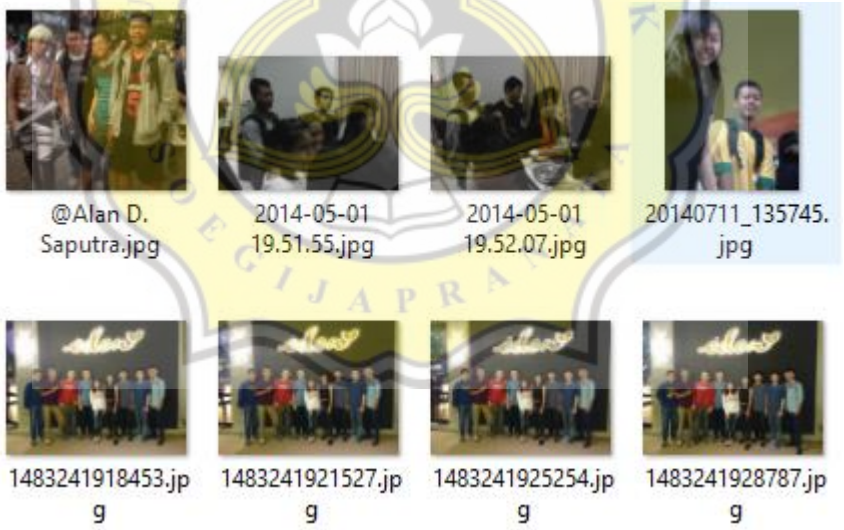
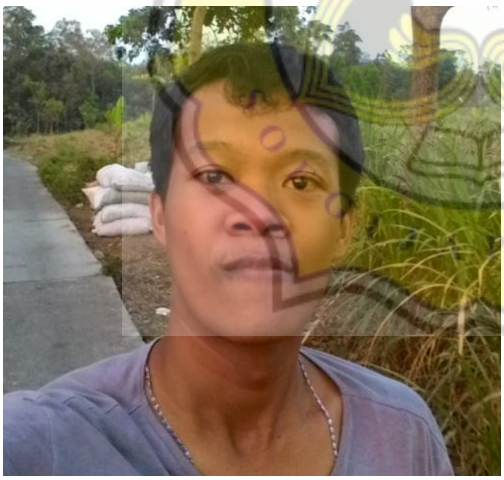


Illustration 5.1: Dataset for Training the face

Illustration 5.1: Dataset for training the face

Table 5.1: Table of Detecting Faces from the Datasets Using CNN

Input Datasets	Output Datasets
	






As we can see from the result above, there is one result that the position of the input dataset is different from the output dataset. This result can be done because of the further refinement and facial landmarks positions with onet function that can be found in David Sanberg code.

Table 5.1: Table of Detecting Faces from the Datasets Using Pimeyes

From the picture above, pimeyes is used to detect many faces in one picture. This project use pimeyes cause this project need to save the face after detecting with the name of the picture.

5.2.1 Diagram of Analysis

This sub-chapter give the result from the testing that already done after three weeks of testing and collecting the video for testing. As we can see from the picture below, this project can get the result of the diagram after detect the video that already collected from 14 days straight. In 14 days, the face that detected in the video only 88 minutes and only get 186 frame that consists of faces.

As we can see in this diagram blue color is for family member, orange for stranger, and grey is for the result that can't detect face from the frame. It gets 160 for family, 19 for stranger and 7 can't detect the face.

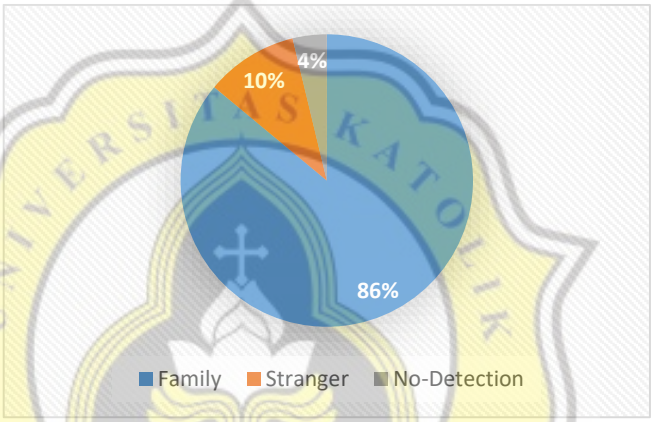


Illustration 5.2: Diagram of the Result

TP	159
FP	19
FN	8
recall	0,89325843
precision	0,95209581

Illustration 5.2: Result of Precision and Recall Analysis

For the final count, this project using Precision and Recall analysis to measuring the performance whether this project is good or not.

Precision = TruePositives / (TruePositives + FalsePositives)

Recall = TruePositives / (TruePositives + FalseNegatives)