Odd Semester (2022)



**BINUS UNIVERSITY**

**BINUS INTERNATIONAL**

**Assignment Cover Letter**

**(Individual Work****)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | |  | |  | |
| **Student Information**: **Surname** | | | | | **Given Names**  Hengky | | **Student ID Number**  2201852492 | |
| 1. | | Sanjaya |  | |
|  |  |
| **Course Code** | **:** COMP6056 |  |  | | **Course Name** | | **:** Program Design Methods | |
| **Class** | **:** L1CC |  |  | | **Name of Lecturer(s)** | | **:** Jude Joseph Lamug Martinez | |
|  |  |  |  | |  | |  | |
| **Major** | **:** CS |  |  | |  | |  | |
| **Title of Assignment** | : FaceRecognition Attendance | |  |  | |  | |  | |
| **Type of Assignment**    **Submission Pattern** | **:** Final Project |  |  | |  | |  | |
| **Due Date** | **:** 21-11-2018 |  |  | | **Submission Date** | | **:** | |

The assignment should meet the below requirements.

1. Assignment (hard copy) is required to be submitted on clean paper, and (soft copy) as per lecturer’s instructions.
2. Soft copy assignment also requires the signed (hardcopy) submission of this form, which automatically validates the softcopy submission.
3. The above information is complete and legible.
4. Compiled pages are firmly stapled.
5. Assignment has been copied (soft copy and hard copy) for each student ahead of the submission.

# Plagiarism/Cheating

BiNus International seriously regards all forms of plagiarism, cheating and collusion as academic offenses which may result in severe penalties, including loss/drop of marks, course/class discontinuity and other possible penalties executed by the university. Please refer to the related course syllabus for further information.

# Declaration of Originality

By signing this assignment, I understand, accept and consent to BiNus International terms and policy on plagiarism. Herewith I declare that the work contained in this assignment is my own work and has not been submitted for the use of assessment in another course or class, except where this has been notified and accepted in advance.

Signature of Student: (Name of Student)

Hengky Sanjaya

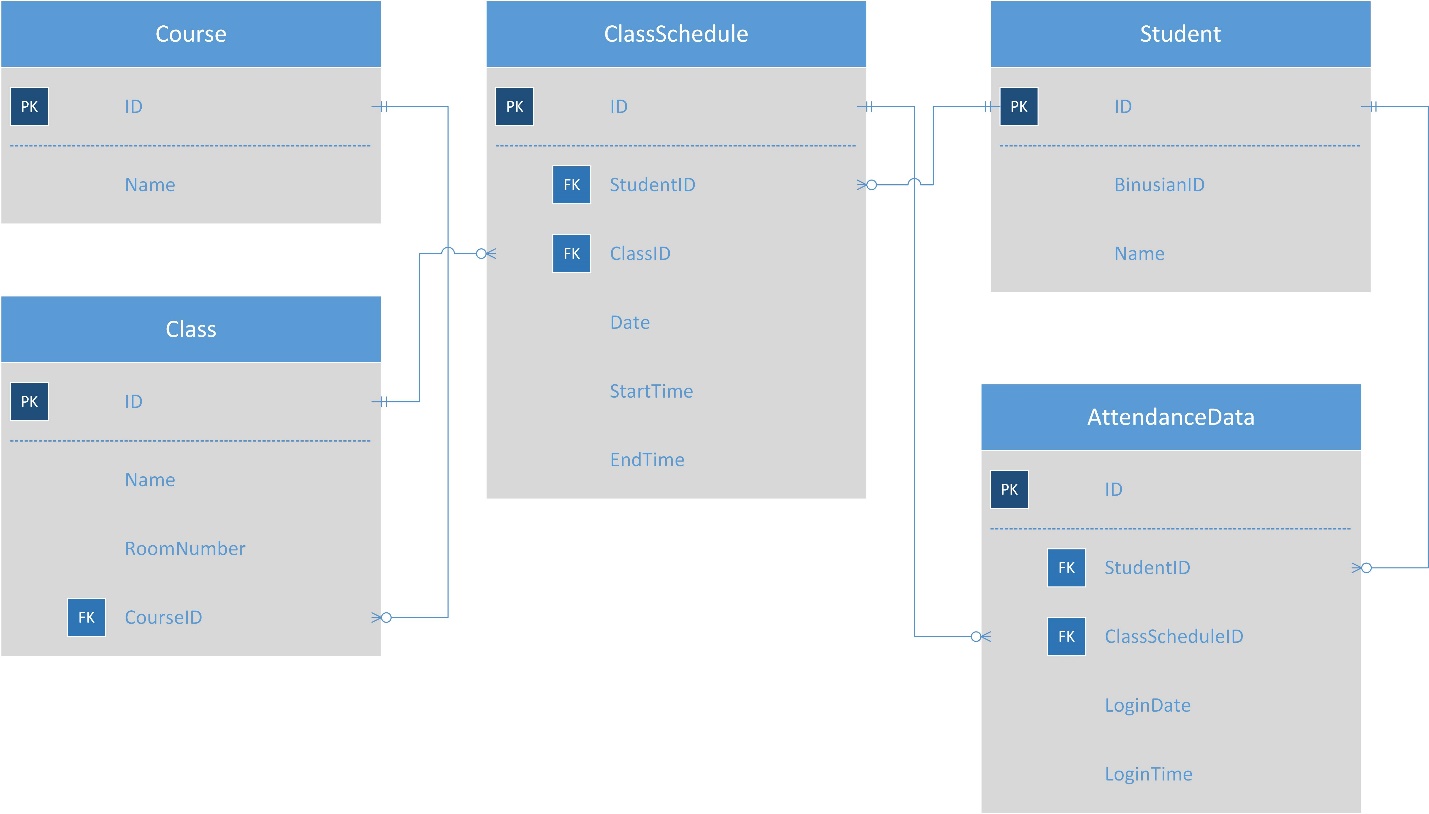
**“Face Recognition Attendance”**

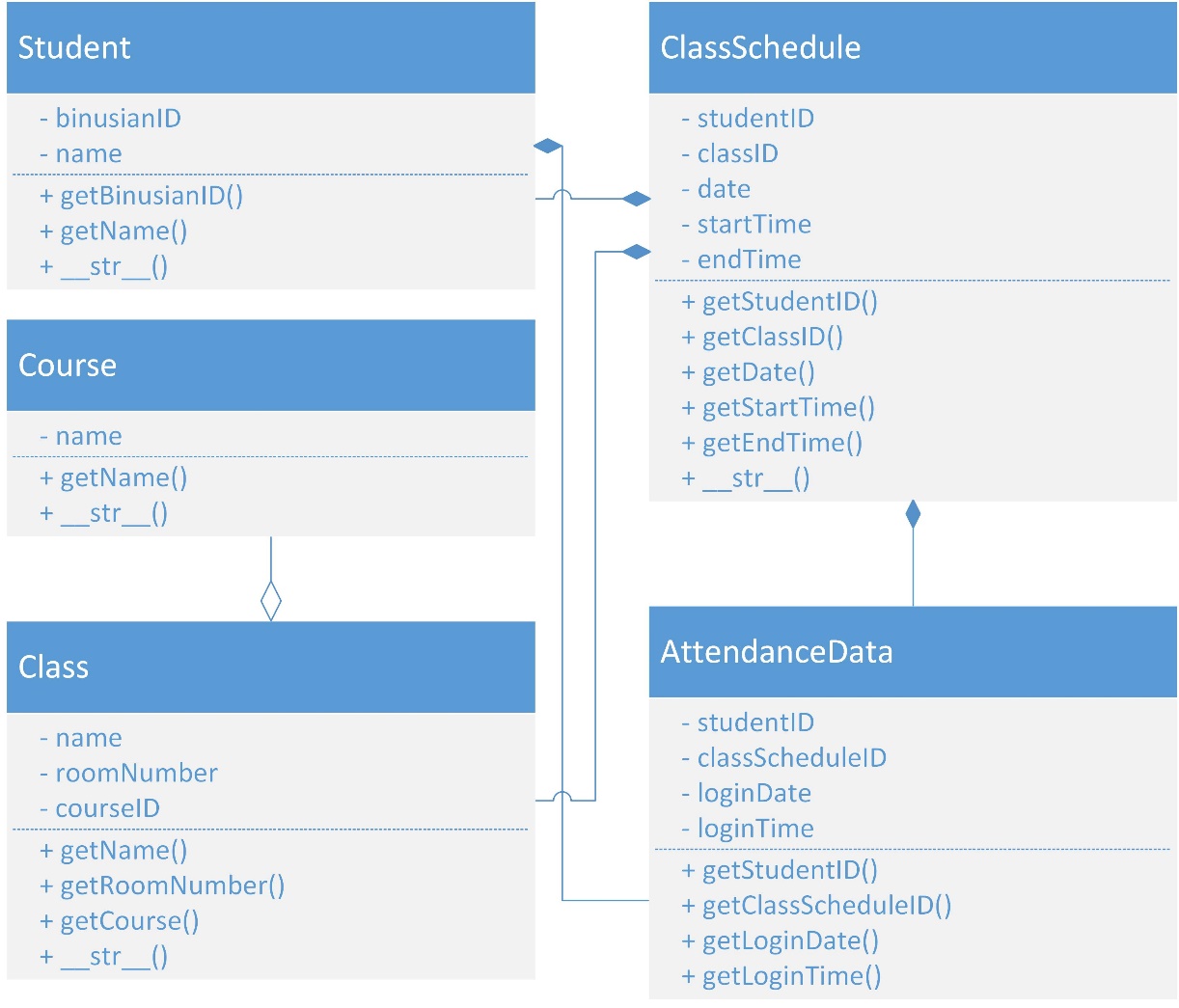
**Name : Hengky Sanjaya**

**ID : 2201852492**

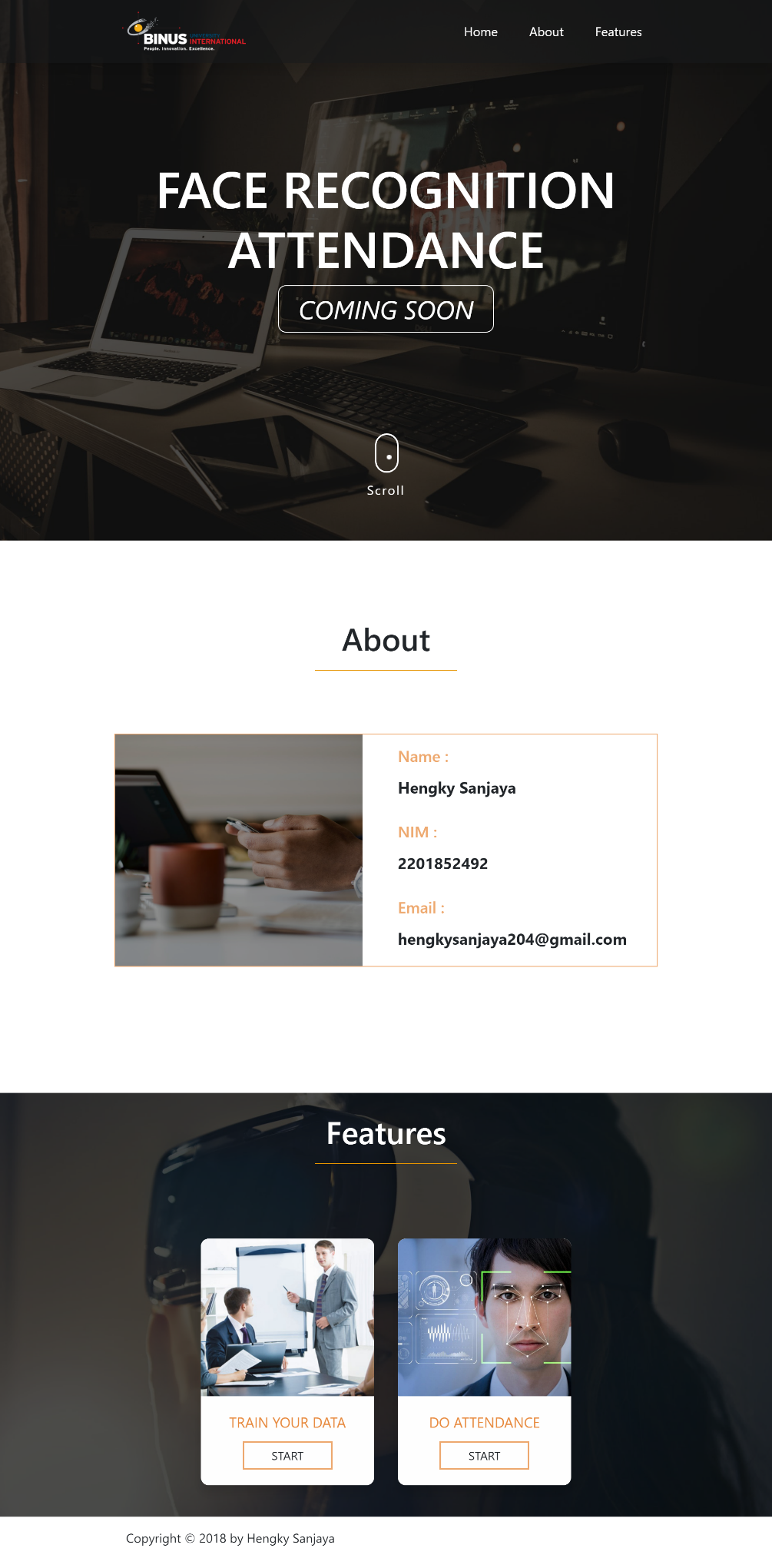
1. **Program Description**

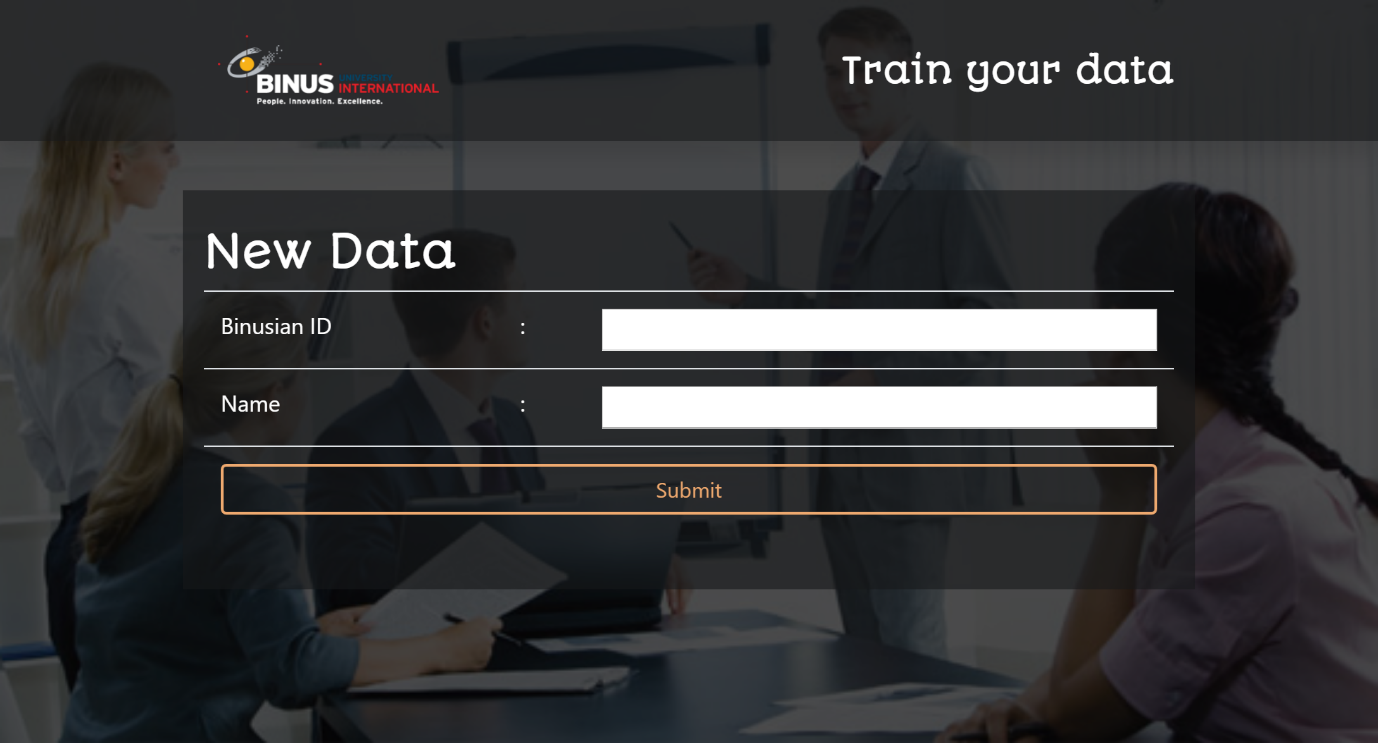
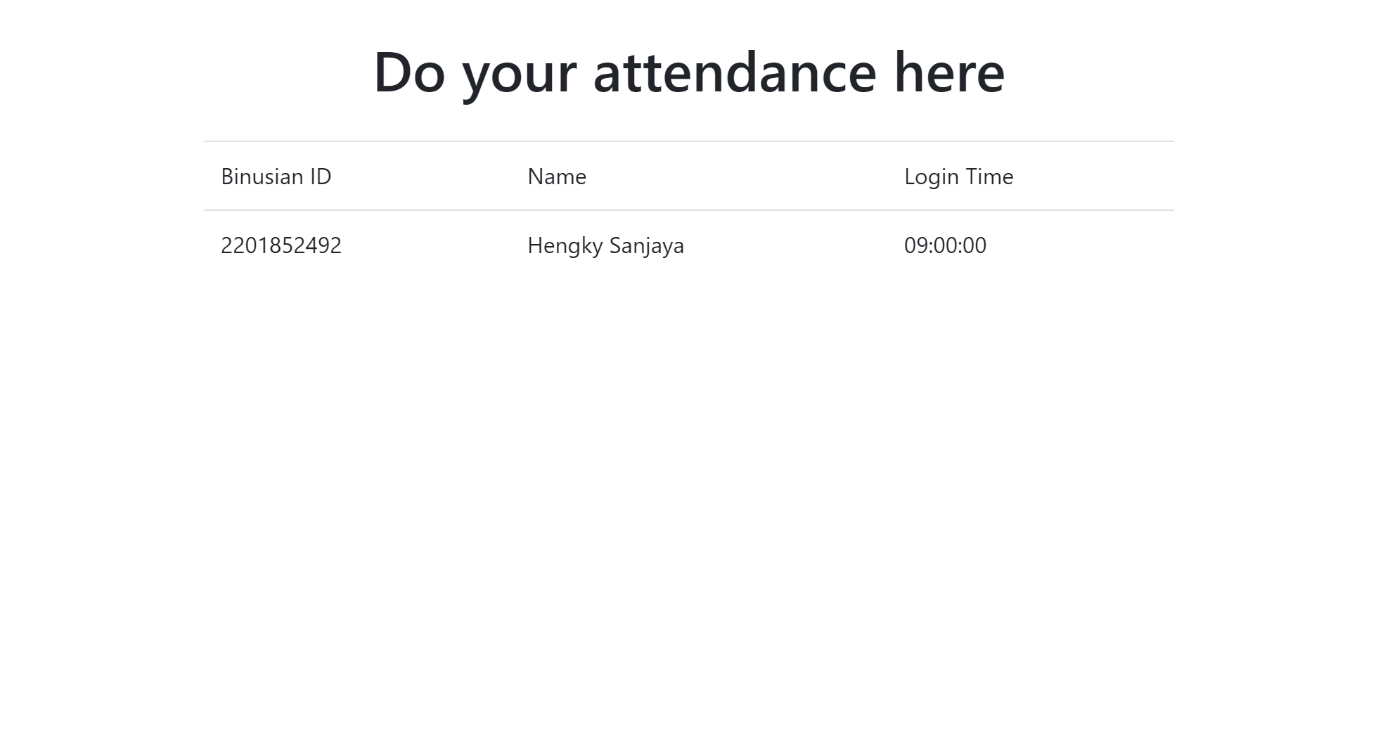
This application is a web-based application to help people do their attendance easier using face recognition. This application is built using python version 3.7, Django framework version 2.1.2, SQLite for the database management, bootstrap for the user interface design. This program is meant to help people who always forget to bring their identity card.

1. **ERD & Class Diagram**



1. **Application Interface**
2. Homepage



1. Train Data Page
2. Do Attendance Page
3. **Library used**

There are some library that I used to build my “Face Recognition Attendance” project:

* Face-recognition version 1.2.3

To do the face detection and recognition.

* OpenCV-python 3.4.3.18

This library is used to open camera / webcam.

* OS

To use some built-in function from python like getting the list of directories from specific path

* Datetime

To get current datetime value and also parsing function

1. **Lessons that Have Been Learned**

Since this is the first time I am learning python language so this final project makes me learn and get so many new things and it’s really fun, and I also learn new framework called Django to build my final project in web based.

1. **Project Technical Description**

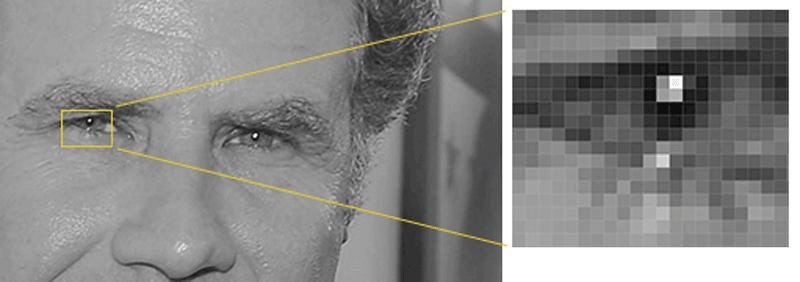
* Finding faces using OpenCV

There are basically two primary ways to find faces using OpenCV:

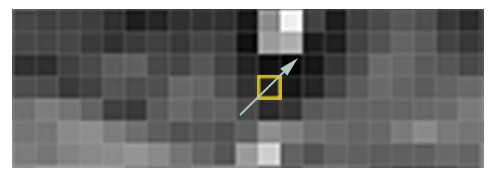
* + Haar Classifier
  + LBP Cascade Classifier

And I am using Haar Classifier because it is more accurate than LBP but it is also much slower, that’s why I have this file “haarcascade\_frontalface\_default.xml” to do the face detection.

* Finding faces using face-recognition library

To find faces in an image, I start to make the image in black and white because we don’t need color data to find faces.

Then look at every single pixel in the image one at a time. Our goal is to figure out how dark the current pixel is compared to the pixels directly surrounding it. Then draw an arrow showing in which direction the image is getting darker.

These arrows are called *gradients* and they show the flow from light to dark across the entire image.

1. **References**

* <https://www.blog.pythonlibrary.org/2018/08/15/face-detection-using-python-and-opencv/>
* <https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78>
* https://github.com/ageitgey/face\_recognition