  
BACS2003 ARTIFICIAL INTELLIGENCE

**202105 Session, Year 2021/22**

**Assignment Documentation**

|  |  |
| --- | --- |
| **Full Name: Tang Xiao Zu** | |
| **Student ID: 19WMR11402** | |
| **Programme: RSF** | |
| **Tutorial Class: 1** | |
| **Project Title: Face & Object Recognition** | |
| **Module In-Charged: Face Recognition** | |
| **Other team members’ data**   |  |  |  | | --- | --- | --- | | **No** | **Student Name** | **Module In Charge** | | **1** | **Tang Xiao Zu** | **Face Recognition** | | **2** | **Phang Wai Hong** | **Object Recognition** | | **3** |  |  | | |
| **Lecturer: Dr. Goh Ching Pang** | **Tutor : Dr. Goh Ching Pang** |
| **Deadline: 12th Sept 2021 (Week 12, Sunday, turn in to Google Classroom before 11.59pm)** | |

Contents

[**1.** **Introduction** 2](#_Toc82374457)

[1.1. Problem Background 2](#_Toc82374458)

[1.2. Objectives/Aims 2](#_Toc82374459)

[1.3. Motivation 3](#_Toc82374460)

[1.4. Timeline/Milestone 4](#_Toc82374461)

[**2.** **Research Background** 5](#_Toc82374462)

[2.1. Background of the applications 5](#_Toc82374463)

[2.2. Analysis of selected tool with any other relevant tools 7](#_Toc82374464)

[2.3. Justify why the selected tool is suitable 9](#_Toc82374465)

[**3.** **Methodology** 10](#_Toc82374466)

[3.1. Description of dataset 10](#_Toc82374467)

[3.2. Applications of the algorithm(s) 11](#_Toc82374468)

[3.3. System flowchart/activity diagram 16](#_Toc82374469)

[3.4. Proposed test plan/hypothesis 16](#_Toc82374470)

[**4.** **Result** 17](#_Toc82374471)

[4.1. Results 17](#_Toc82374472)

[4.2. Discussion/Interpretation 18](#_Toc82374473)

[**5.** **Discussion and Conclusion** 19](#_Toc82374474)

[5.1. Achievements 19](#_Toc82374475)

[5.2. Limitations and Future Works 19](#_Toc82374476)

[**Reference & Source** 20](#_Toc82374477)

[**Appendix** 22](#_Toc82374478)

# **Introduction**

## Problem Background

In today's world, technologies that use computers to provide information are rapidly emerging into human’s daily life. As the digital world and the real-world merge more and more throughout the years, facial recognition has been one of the most important research topics that has been existing in the world. Government all over the world have been demanding on this issue, encouraging artificial intelligence expert to develop facial recognition. In the old times, traditional recognition technology was used. All of the methods of the recognition technology require human’s memory because they need to remember their own information such as their username and password. This causes the traditional recognition method to have a big con which is security risk, because humans can forget things easily. Original identity material takes a long time to retrieve back but the information inside can be required easily by a thief. As a result, the identity could be impersonated by other people and there will be serious consequences. For easier understanding, facial recognition is the use of computer vision technology and algorithms, from the picture, video, or real-life working environment through a camera, to analyze identity. Other than analyzing identity, there are additional attributes that may be conducted through facial recognition, which is the individual’s gender, age, emotion etc.

## Objectives/Aims

There are a few objectives to design the face detection system. The objective of the face detection system is :

* To design real time face detection system.
* To recognize up to 10 different faces.
* To utilize the face detection system based on Haar Cascade Classifier.
* To develop the face detection system using Visual Studio Code python 3.9.5 version.

## Motivation

The facial recognition system develop is a general system with normal identification function. The benefit that the system can bring to the working environment world is that it can help to find missing person. As in the world today, there are a lot of CCTVS on the road, we could implement the facial recognition system in those CCTV to help the police to find missing people and even identify criminals. Other than that, the facial recognition system can also help to reduce touchpoints which means that fewer human resources than other type of security measures, such as fingerprints. In other words, it doesn’t need any physical contact or any direct human interaction to trigger the facial recognition system, it is an automatic process.

## Timeline/Milestone

|  |  |  |  |
| --- | --- | --- | --- |
| Timeline/ Milestone | Days | Start Date | End Date |
| Planning | 7 | 5/7/2021 | 11/7/2021 |
| Background studies | 4 | 12/7/2021 | 15/7/2021 |
| Analysis of selected tool | 6 | 16/7/2021 | 21/7/2021 |
| Identify algorithm | 7 | 22/7/2021 | 28/7/2021 |
| Proposed test hypothesis | 5 | 29/7/2021 | 2/8/2021 |
| Coding | 20 | 3/8/2021 | 22/8/2021 |
| Prepare Dataset | 9 | 23/8/2021 | 1/9/2021 |
| Results | 8 | 2/9/2021 | 9/9/2021 |
| Conclusion | 2 | 10/9/2021 | 11/9/2021 |
| Submission of project | 1 | 12/9/2021 | 12/9/2021 |

# **Research Background**

## Background of the applications

Digital image processing is a process of manipulating digital images through a digital computer and it is also often called as digital picture processing. The very first image processing were found in the 1920s, which is called the Bartlane cable picture transmission system and the system is used to transfer the picture sent by submarine between London and New York in less than three hours’ time (July 2018, Sajjan Singh). In the 1960s, the image processing system is being enchanted at Bell Laboratories, the Jet Propulsion Laboratory, Massachusetts Institute of Technology, and other few research facilities to improve the images of the moon (Jan 2016, Wiki).

In today’s world, image processing is a very popular application among computer science developers. Fields which are traditionally used by analog imaging are now converting to digital imaging which are more affordable and more accurate. The examples are the medicine field, video production, photography, remote sensing and even security monitoring. These fields produce a huge amount of digital image daily, more than it can be examined manually by humans. Basically, image processing is a process of converting those digital images into a two-dimensional picture by a computer. The outcome of image processing could be an image or the features or characteristics or the image itself. There are a few common applications found using image processing systems, computer vision, remote sensing, facial recognition, and biomedical imaging.

* Computer Vision – Computer vision is one of the applications in image processing where computers can get meaningful information from digital images and videos. After that stage, the system will allow computers to act based on the information they obtain because computer vision helps them to see, observe and understand. For example, applying computer vision in an autonomous vehicle, this will allow the vehicle to have the feature of autopilot as the system itself can detect and observe what is happening in front of the vehicle and will make necessary action to prevent the vehicle from getting into an accident.
* Facial Recognition – Facial recognition is a method that allows the technology to detect important features of the person face and else is ignored. The technology collects a dataset of unique biometric data of each individual which includes their face and facial expression in order to verify or authenticate that particular person. For example, facial recognition system can help the police to detect missing people and even identify criminals through the CCTVs.
* Remote Sensing – Remote sensing is a process where the data collected is from the energy or signals of an object or by phenomenon. Which means that this process is using real-time sensing devices which are wireless and. To be more precise, it means that the wire does not need to have direct contact with the object. For example, remote sensing in space probes allow the space probes to detect and monitor physical characteristics of an area in Earth by collecting the energy emits from the surface of Earth.
* Biomedical Image Enhancement – In the field of medicals, this process is very important as this helps the original digital methods such as the Computed Tomography (CT) or Magnetic Resonance Imaging (MRI) to enhance the results, which also means that to get rid of dead pixels of the images to make the images clearer.

## Analysis of selected tool with any other relevant tools

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tools comparison** | **Remark** | **Visual Studio Code** | **Jupyter** | **Spyder** |
| Type of license and open-source license | State all types of licenses | MIT License, Propriety Software License | MIT License | MIT License |
| Year founded | When is this tool being introduced? | April 29, 2015 | February, 2015 | 18 October, 2009 |
| Founding company | Owner | Microsoft | Fernando Perez | Pierre Raybaut |
| License Pricing | Compare the prices if the license is used for development and business/commercialization | Free of Charge and Open Source | Free of Charge and Open source | Free of Charge and Open source |
| Supported features | What features that it offers? | - Language support, it can support a lot of common language such as python, C++, Java, and others.  - Integrated Command Line Interface, it allows user to open CLI in it.  - ESLint, a tool to help users spot errors in their code and fix their code during their coding stage.  -Code formatter, it is to format the style based on the language so that user interface can be nicer. | - Able to download some useful extension such as Hinterland, which is for code autocompletion.  - Jupyter widgets, this allows the user to build interactives GUIs in the notebook. | - Support multiple IPython consoles  - Explore and edit GUI  - A help pane to retrieve variety of functions, classes, and methods  - A debugger |
| Common applications | In what areas this tool is usually used? | - Debugger  - Editor | - Debugger | - Debugger |
| Customer support | How the customer support is given, e.g., proprietary, online community, etc. | -Online Community:  <https://dev.to/t/vscode>  -Visual Studio Code FAQ | - Online Community:  <https://gitter.im/jupyter/jupyter>  -Jupyter Community Guides | -Online Community:  <https://gitter.im/spyder-ide/home> |
| Limitations | The drawbacks of the software | - A lot of extension needed to be installed in the system to make it stable and those extension is hard to install.  - Some plugins will cause VSC to crash.  - | - No IDE integration  - No code style correction. | - User interface not attractive  - Unable to do collaborative works |

## 

## Justify why the selected tool is suitable

The integrated development environment used to develop the facial recognition system is the Visual Studio Code (VS Code). VS Code is a free open-source code editor that can support a lot of programming language and even python which are the main language we are going to use to develop the facial recognition system. It is available on many platforms such as Windows, Linux and MacOS. With some simple setup, users can even use VS Code on their mobile phone to do coding. VS Code is also an editor under Microsoft Company.

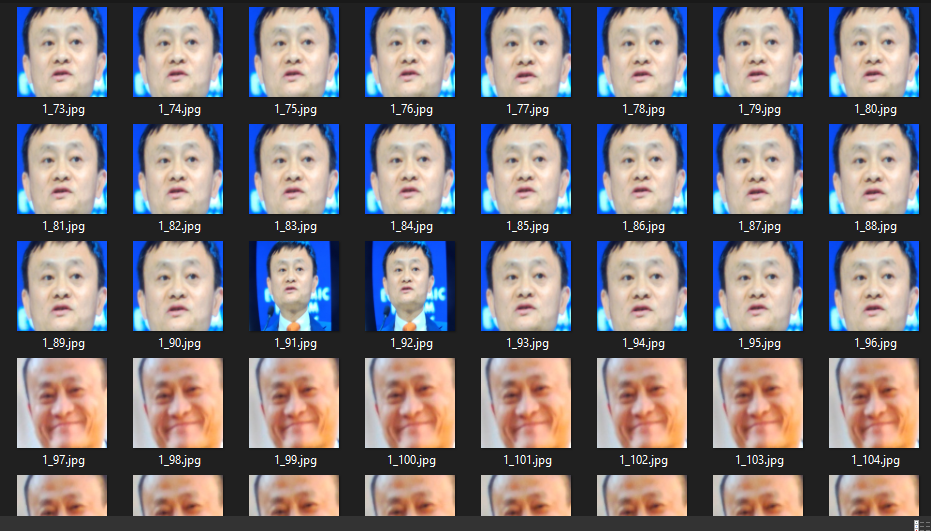
VS Code comes with a lot of useful features that will help a lot during the coding stage, this is also the reason why VS Code is chosen as the development tool for this assignment. VS Code includes built-in support for code completion, richer semantic code understanding and navigation, and code refactoring (Documentation for Visual Studio Code, 2016). Other than that, we can install multiple extensions to improve the efficiency and effectiveness during the coding session. The other useful extension is that when working with git repository, VS Code offers a neat way of comparing the versions users are working with the current one in the repository using Git Compare View. To be more precise, VS Code shows the code changes by anyone in the repository under Git Source Control. At the left-hand side of the editor, it will show the user the editing history on who and which editor make changes to the code or anything in the file. VS Code will also show the code that are being duplicated by several editors with merging conflict function, where it will require the users to choose which code is newer so that they can merge the conflict in the file.

VS Code also have an interesting built-in feature which is called quick fixes. This feature helps a lot during the coding phase as it can identify problems in the code and provide several options to fix the code. For example, by typing a package name in the editor such as “path.”, if the user has not imported any package in the first place, the “path,” code will have error. At this time, code action will be available to automatically help the user to import the path package.

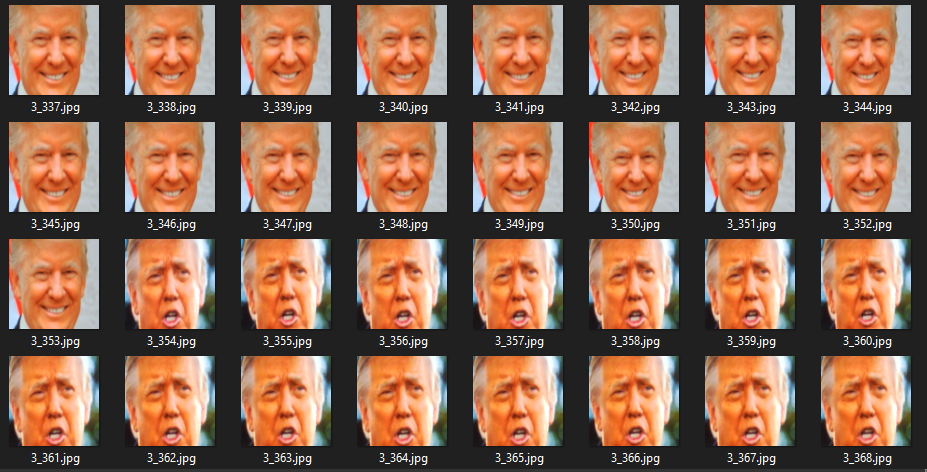
In conclusion, VS Code is a good IDE for collaborative work as our assignment team have 2 person and it also provide features that will help us a lot during the coding phase.

# **Methodology**

## Description of dataset



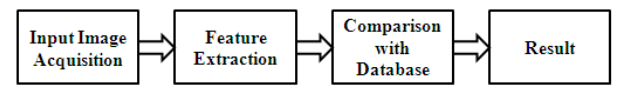
*Figure 3.1.1: Dataset example 1*



*Figure 3.1.2: Dataset example 2*

The dataset that we used for this facial recognition system is all images. All the images are capture from the photo that existed on the internet and even our own face. The dataset required is used for training in order for the system to increase its accuracy in detecting different faces. To ensure the accuracy of the detection, we collected images with different position from every particular person.

## Applications of the algorithm(s)



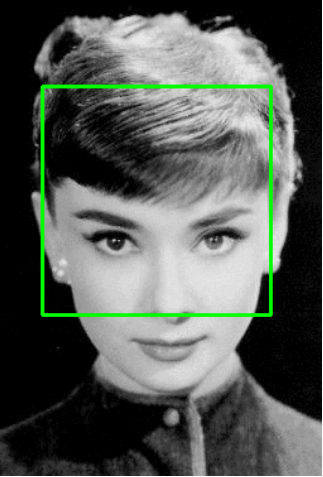
*Diagram 3.2: Digital Image Process*

In digital image process analysis, it can be divided into few phases. The diagram above shows the stage of digital image processing.

* Image Acquisition – This is the first important step of digital image processing. In this stage, the image is being require in the format of digital. After requiring the image, preprocessing such as scaling must be made to the image. Images in this stage can be require through digital cameras or aerial camera. To have proper analysis, the image required must be in good conditions.
* Preprocessing – After the first stage, second stage which is the preprocessing operations are required to be performed on the image require. Preprocessing helps to enhance the features of the image. High resolution image requires longer time to process, so the size of the image need to be reduced to solve this problem. The images are then converted into gray scale image because lesser information will be processed.
* Edge Detection & Segmentation – In image edge detection, boundaries of objects inside the image are being process. While in image segmentation, image is being identified into multiple segments in order for the image to be easier to analyze. Only the important part of the segments is being process.
* Image Restoration – In this stage, the image’s appearance is improved which also means that the image is being recovered from a degraded version. This process is based on mathematical algorithm and probabilistic analysis of an image.
* Result – After an image have been through all those stages and process, the image is used as a dataset to be recognized.

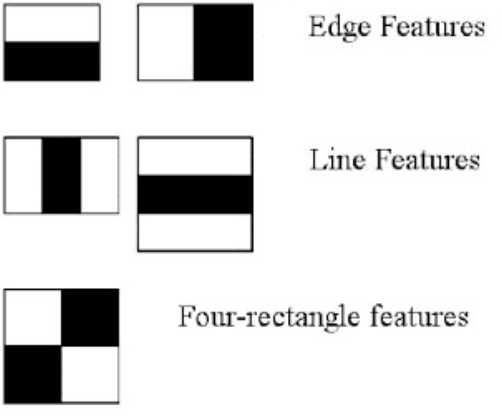
In this facial recognition application, there are a few algorithms and techniques used in order to get the result which is, Haar Cascade, OpenCV, os, Pandas and Numpy. OpenCV is a huge open-source library for image processing, and it supports programming language python (Mar 2021, Geeks). In OpenCV, it allows the system to identify faces which we will be doing in this assignment. To implement OpenCV in our code, we must first import the cv2 package into our code. The example code of the OpenCV is “cv2.VideoCapture(0)”, the meaning of this code of line is to open our webcam through the system. Other than that, the os module included in the code provides function for interacting with the operating system that is currently running for the facial recognition system. For example, the os module code used in this implementation is “os.listdir(“dataset”)”, this line of code means that the function will interact with the operating system to find the file name “dataset” and return a list of names in the directory given by path. Furthermore, pandas which is another open-source package used to help implement the system in this assignment. This package is built on top of another package which is called Numpy. Numpy stands for ‘Numerical Python’, it is a module that allows mathematical computation on arrays and matrices. In this assignment, Numpy array is used because arrays are an essential part of the facial recognition machine learning, and the package pandas is also used to support the Numpy package. This is very important to include in the facial recognition machine learning system as it requires to find, capture, store and analyze facial in order to match them with images of individuals stored inside an existing database or dataset.

Haar Cascade algorithm will be used in this facial recognition system to identify faces in an image and real time video which is using the webcam. The algorithm uses edge or line detection features proposed by Viola and Jones in their seminal 2001 publication ‘Rapid Object Detection using a Boosted Cascade of Simple Features’ (Apr 2021, Adrian Rosebrock). No matter what location and what scale the image is in, this algorithm is capable of detecting object in it, this algorithm can even run in real-time, making it possible to detect face through live cam.



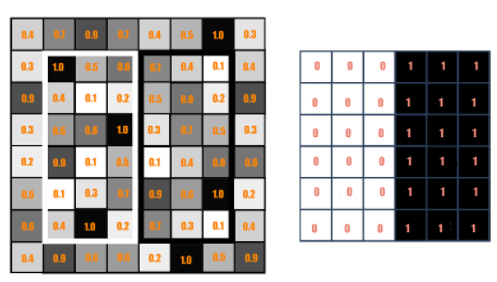
*Figure 3.2.1: Green sliding window box on image*

In the above figure, we can see that a fixed size green window is applied on the image. The green window will slide from left-to-right and top-to-bottom until the whole image is being process. At each stop from the figure above, the algorithm will compute some features by classifying the region as Yes if that region has faces in it, and No if the region do not have faces in it. By doing that, we can categorize the data into positive and negative data where the positive data supports the Yes region, and the negative data supports the No region. Throughout the whole process, we will get the total of positive and negative data and from these data, we can identify the faces that exist in the image.

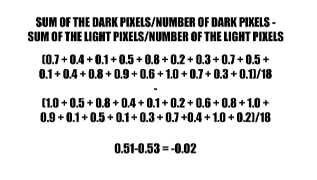


*Figure 3.2.2: 5 different types of Haar-like features extracted from an image path*

From the sliding green box, when it stops, five of these rectangular features are computed. These Haar features makes it easier to find out the edges or the lines in the image, or to pick areas where there is a sudden change in the intensities of the pixels (Dec 2020, Girija Shankar Behera). Depending on the features in figure 3.2.2, the first set of two rectangle is to find the edge in a horizontal and vertical direction. While the second set of three rectangle is to find the lighter region surrounded by darker regions, and in the third set of four rectangle is to find the change of pixel.



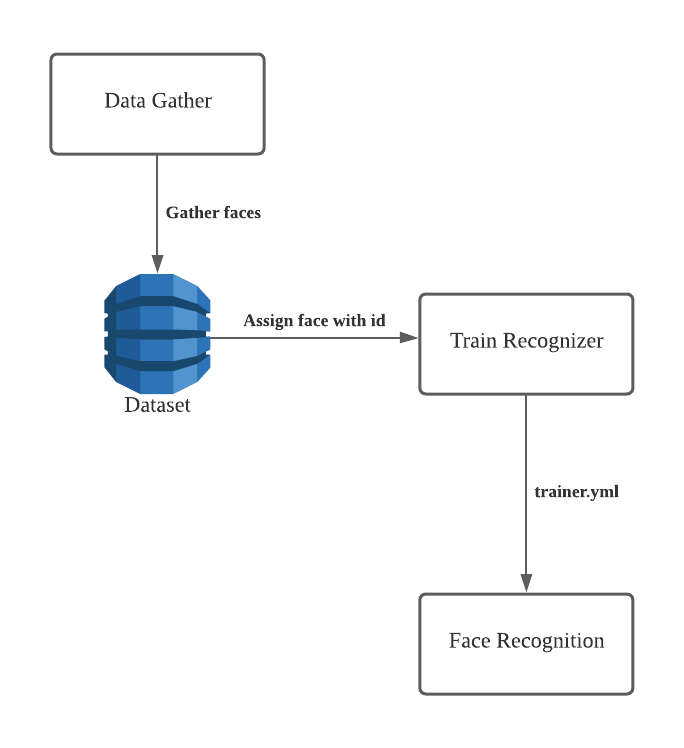
*Figure 3.2.3: Example of Haar calculation*



*Figure 3.2.4: Calculation*

Calculation of Haar value from one of the Haar features rectangle is shown on figure 3.2.3. The darker area shown are the haar features with pixel value 1 and, the lighter area is the haar features with pixel value 0. In order to identify the face in an image, the rectangle is responsible for finding out particular features such as edge, line, and any structure in the image when the haar features run through the image. If the total calculated value of the haar value is closer to 1, we can conclude that there is an edge exist in the rectangle. Based on the calculation in figure 3.2.4, since the haar value is not close to 1, we can say that there are no edges exists.

## System flowchart/activity diagram



*Diagram 3.3: System Flow of Facial Recognition*

## Proposed test plan/hypothesis

Based on the objective that had mentioned above, here is the hypothesis and test plan that we are going to carry out:

1. Able to detect the face of the user.
2. Able to capture photo of the user.
3. Able to detect up to 10 different faces.
4. Able to detect the face based on different conditions.

# **Result**

## Results

|  |  |  |  |
| --- | --- | --- | --- |
| No | Input | Condition | Output/ result |
| 1 |  | Wear Spectacle, No cap |  |
| 2 |  | No wear spectacle, No cap |  |
| 3 |  | Wear Cap, Wear spectacle |  |
| 4 |  | Wear cap, no wear spectacle |  |
| 5 |  | Cover half of the face |  |

## Discussion/Interpretation

In order to calculate the accuracy of the facial recognition system, the calculation to calculate the accuracy can be calculate by using this mathematical formula:

(Success / Total Runs) \* 100%

1. For the first result, 385 has succeeded out of the 500 runs so that the accuracy for the first result is: (385/500) \* 100% = 77%
2. For the second result, 373 has succeeded out of the 500 runs so that the accuracy for the second result is: (373/500) \* 100% = 75%
3. For the third result, 391 has succeeded out of the 500 runs so that the accuracy for the third result is: (391/500) \* 100% = 78%
4. For the fourth result, 420 has succeeded out of the 500 runs so that the accuracy for the fourth result is: (420/500) \* 100% = 84%
5. For the fifth result, 408 has succeeded out of the 500 runs, so that the accuracy for the fifth result is: (408/500) \* 100% = 82%

Since the result is not close to perfect, there are a few steps we can do in the future to make it more accurate, which is to get more data from the same person and do more training to the facial recognition system.

# **Discussion and Conclusion**

## Achievements

We proposed to develop a high performance, scalable, agile, and low-cost image processing application. We divided the proposed system into 2 modules which are the facial recognition and object recognition module. First, we studied how image processing works and how to get dataset and train the model. Then we examine and compare the with the source dataset. The model used will take do some processing in the image to find out the edges, line, and pixel changes in order to identify the faces in the images and even on real-time which is the webcam. In the end, the facial recognition system is able to recognize up to 10 different faces whether if it is in digital image form or in real-time camera form.

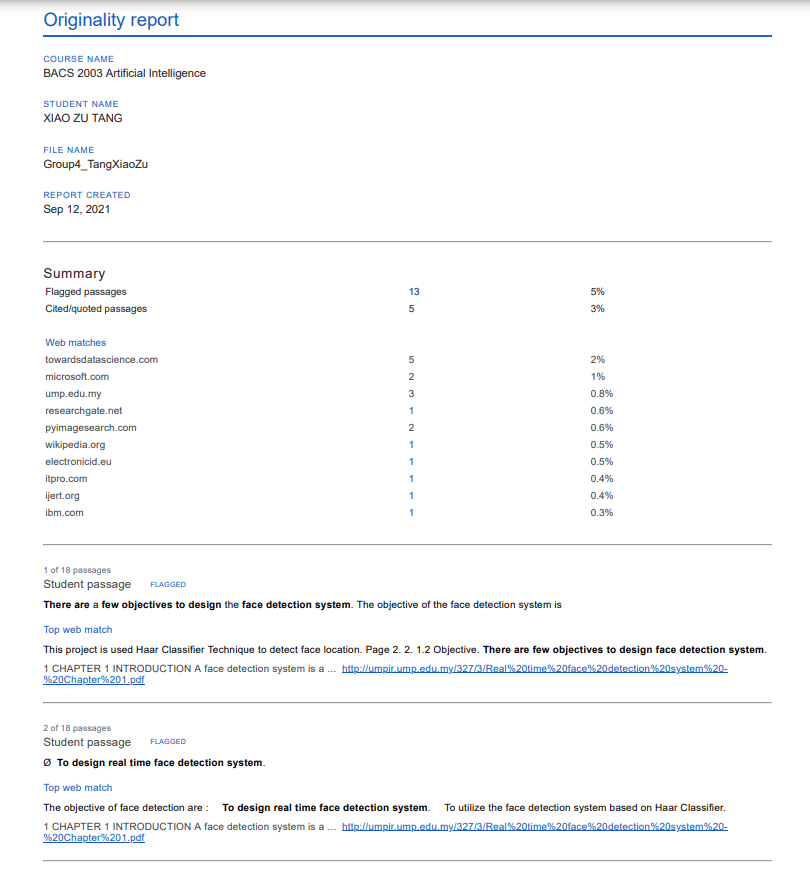
## Limitations and Future Works

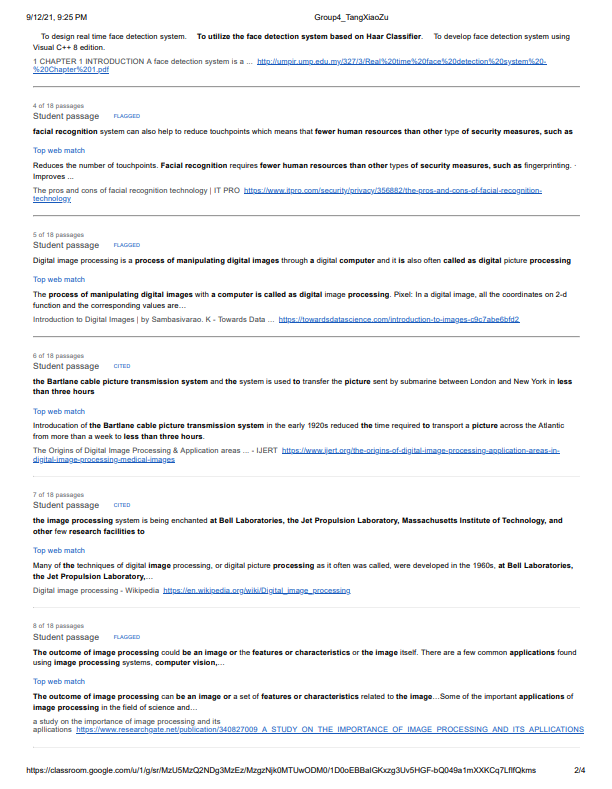
When we are developing the image processing application, there are still a lot of datasets obtain which is not that accurate based on certain environment. Due to the lack of knowledge on the current image processing application, we are not able to make the application more accurate and more success. For future works, we can research more on this field and re-train the application using a more specific dataset under variety of conditions. In the future, we wish to increase the accuracy of the facial recognition system by collecting more data and train our system more, so that our system can recognize everyone’s face easily.

# **Reference & Source**

1. Technology, 0., 2021. *U.K. Outlines Pros and Cons of Facial Recognition Technology | Data Privacy Philippines*. [online] Data Privacy Philippines. Available at: <https://www.privacy.com.ph/u-k-outlines-pros-cons-facial-recognition-technology/>.
2. En.wikipedia.org. 2021. *Digital image processing - Wikipedia*. [online] Available at: <https://en.wikipedia.org/wiki/Digital_image_processing>.
3. Slideshare.net. 2021. *Chapter 3 Cells*. [online] Available at: <https://www.slideshare.net/sgossett5757/chapter-3-cells-8702447>.
4. Singh, S., 2021. *The Origins of Digital Image Processing & Application areas in Digital Image Processing Medical Images*. [online] Ijert.org. Available at: <https://www.ijert.org/the-origins-of-digital-image-processing-application-areas-in-digital-image-processing-medical-images>.
5. Ibm.com. 2021. *What is Computer Vision? | IBM*. [online] Available at: <https://www.ibm.com/topics/computer-vision>.
6. Face Recognition: how it works and its safety. 2021. *Face Recognition: how it works and its safety*. [online] Available at: <https://www.electronicid.eu/en/blog/post/face-recognition/en>.
7. Oceanservice.noaa.gov. 2021. *What is remote sensing?*. [online] Available at: <https://oceanservice.noaa.gov/facts/remotesensing.html>.
8. Code.visualstudio.com. 2016. Documentation for Visual Studio Code. [online] Available at: <<https://code.visualstudio.com/docs>>.
9. Code.visualstudio.com. 2021. *Editing Python Code in Visual Studio Code*. [online] Available at: <https://code.visualstudio.com/docs/python/editing>.
10. Mathworks.com. 2021. *Edge Detection*. [online] Available at: <https://www.mathworks.com/discovery/edge-detection.html>.
11. Analytics Vidhya. 2021. *Image Segmentation | Types Of Image Segmentation*. [online] Available at: <https://www.analyticsvidhya.com/blog/2019/04/introduction-image-segmentation-techniques-python/>.
12. Rosebrock, A., 2021. *OpenCV Haar Cascades - PyImageSearch*. [online] PyImageSearch. Available at: <https://www.pyimagesearch.com/2021/04/12/opencv-haar-cascades/>.
13. Medium. 2021. *Face Detection For Beginners*. [online] Available at: <https://towardsdatascience.com/face-detection-for-beginners-e58e8f21aad9>.

# **Appendix**







# 

