**CMPE 181 Final Project**

**Student Check-In**

**(The Modern Examination Check-in System)**

**Author:**

**Yang Li, En-Ping Shih**

**Professor:**

**Dr. Kaikai Liu**

**Introduction**

Because of the COVID-19 and the shelter-in-place situation, students need to take their classes online as well as taking their online exams via Canvas or any other online platforms. Cheating in exams have never been solved in decades, and professors would not be able to notify any cheating behaviors during the quarantine while students taking the exams. We create a student check-in system that would detect cheating behaviors such as paid exam-taker and using other electric devices. The purpose of this application is to ask students to open the camera, and use their student ID to match if the picture of the student ID match the student. If it match, we would check-in the students for the exam.

**System Design**

**Overall system diagram**

一張含有 螢幕擷取畫面 的圖片

自動產生的描述

The system design of our application combined two part:

1) Back-end training model

2) Front-end detecting interface

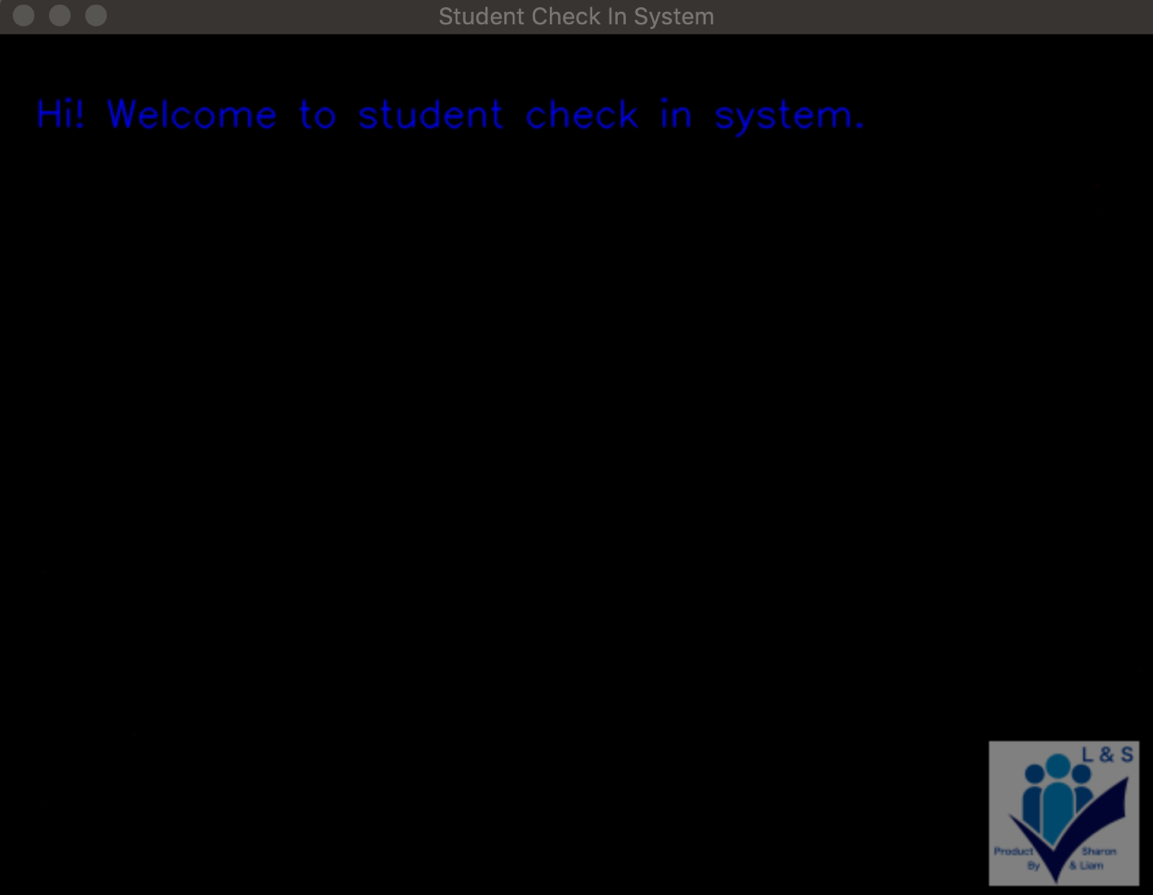
**Back-end training Model:**

**A screenshot of a cell phone

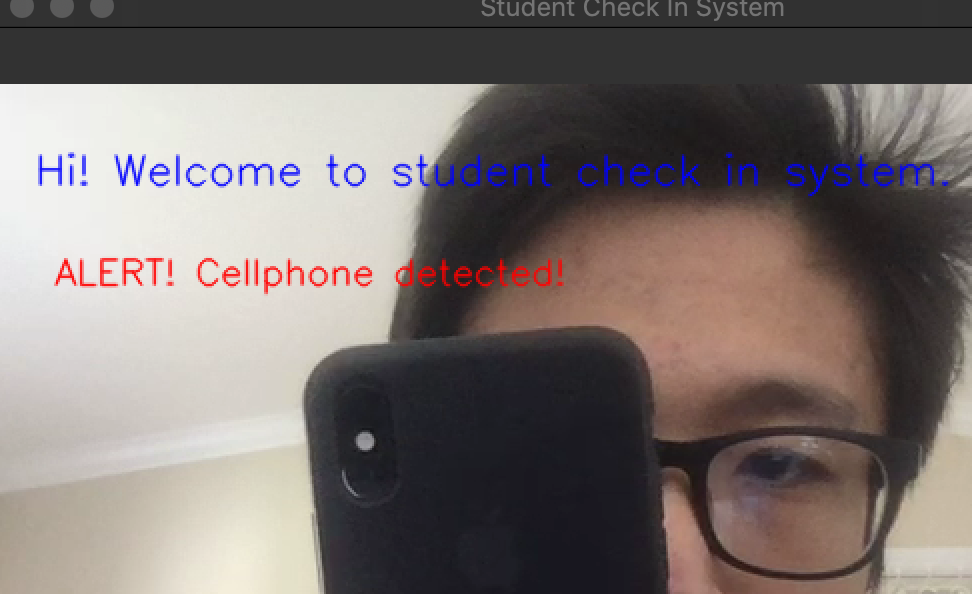
Description automatically generated**

We used PIL (the biggest python imaging library) and OpenCV (a library of programming functions mainly aimed at real-time computer vision.) to build up the model that can recognizes the faces while opening the camera.

**Front-end Detecting Interface**

****

We used another module for the front-end interface. The title is called Student Check-in System. We have the welcome message, and as well as the L & S logo on the right bottom corner that represent the authors: Yang(Liam) & En-Ping(Sharon). Whenever the student have their electric devices on the camera, the system would shows up the red message and alert the professor.



**Key Techniques**

* Implementing the OpenCV cascade classifier would be the first step of our application.
  + Cascade Classifier <https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html>
* Used PIL Library and we created a folder and each sub folder contains photo of one person, to train the model that recognize the faces.

A picture containing phone

Description automatically generated

* Used Dr. Kaikai Liu’s example code of TensorFlow to adding the feature of detecting if the student use any other electric devices.

**Set-up Guide**

* Download from <https://github.com/SharonShih/student-check-in>
* Download the OpenCV Library
  + MAC: <https://www.codingforentrepreneurs.com/blog/install-opencv-3-for-python-on-mac/>
  + WINDOWS: <https://www.codingforentrepreneurs.com/blog/install-opencv-3-for-python-on-windows>
* Download the PIL Library
  + $ pip install pillow
  + <https://pypi.org/project/Pillow/2.2.1/>
* Download the TensorFlow Library
  + <https://www.tensorflow.org/lite/guide/python>

**Task Distribution**

|  |  |
| --- | --- |
| **Team Member** | **Task Distribution** |
| Yang Li | OpenCV, Face Detection, Project Report |
| En-Ping Shih | OpenCV, Face Detection, TensorFlow, Object Detection |

**References**

* Face recognition:
  + <https://www.youtube.com/playlist?list=PLEsfXFp6DpzRyxnU-vfs3vk-61Wpt7bOS>
  + <https://github.com/codingforentrepreneurs/OpenCV-Python-Series>
* Object detection: <https://github.com/lkk688/GoogleCloudIoT/tree/master/iotpython>