

# Bachelor Thesis

for

Mr. Wei Dai

Analysis of Face Recognition Capabilities by Example of a Mobile Web Game

Professor Dr. Kratzke

gefördert durch:



Date of handing out: 19<sup>th</sup> March 2018  
Date of handing in: 19<sup>th</sup> June 2018

A. Hanemann

(Professor Dr. rer. nat. Andreas Hanemann)  
Board of examinations



**Task Description:**

This thesis shall identify face recognition, face tracking, and face gesture recognition libraries,

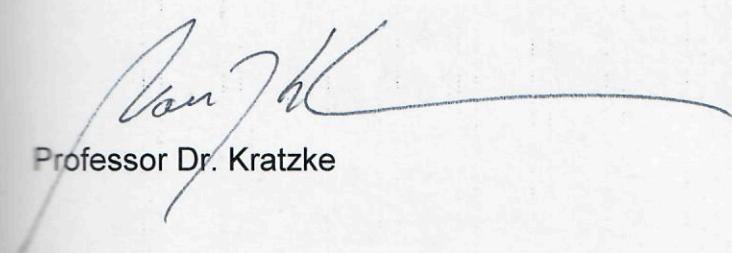
- summarize and categorize their features,
- assess and compare their efficiency,

to develop and realize an interesting game concept that demonstrates face recognition capabilities of smartphones. The resulting web game should be executable in a modern mobile HTML5 web browser. The game is intended to be presented on fairs to demonstrate student outcomes of computer science students. The game will likely involve technologies like JavaScript, HTML, DOM-Tree, CSS and so on. The game should be realized as a single page web app. No server component should be necessary to operate the game.

It is recommended to study and consider this template for thesis documentation from the very beginning.

**TASKS, DOCUMENTATION AND OUTCOMES**

- Analyze face recognition, face tracking, and face gesture recognition libraries,
- Compare performance and efficiency of such kind of libraries,
- Summarize features, opportunities and limitations of these libraries and derive an interesting game concept that is playable on a wide range of mobile browsers on SmartPhones,
- Implementation and documentation of the game
- Evaluation of the game including typical software testing and user acceptance strategies
- Documentation of the above mentioned steps. It is recommended to align the documentation according to this Template.



Professor Dr. Kratzke

**Statement on the bachelor thesis**

I assure that I have written the bachelor thesis independently, without outside help.

Only the given sources have been used for the writing of the bachelor thesis. Literally or in the sense after taken parts are marked as those.

I agree that my work is published, in particular that the work is presented to third parties for inspection or copies of the work are made to pass on to third parties.

18.06.2018

---

Date

DAI Wei

---

Signature

# Analysis of Face Recognition Capabilities by Example of a Mobile Web Game

# Analysis of Face Recognition Capabilities by Example of a Mobile Web Game

# Content

<b>1 Introduction .....</b>	<b>4</b>
1.1 Motivation .....	4
1.2 Structure .....	4
<b>2 Background and Related work .....</b>	<b>5</b>
2.1 Face recognition technology .....	5
2.2 Existing libraries .....	6
2.2.1 Research of relevant libraries.....	6
2.2.2 The summary of search results.....	10
2.3 Selection criteria.....	11
2.4 Exclusion and inclusion according to criteria .....	11
2.5 Assessment and comparison of the libraries.....	12
2.5.1 Clmtrackr.js.....	12
2.5.2 Tracking.js.....	12
2.5.3 CVV.....	12
<b>3 Game Concept.....</b>	<b>13</b>
3.1 Description .....	13
3.2 Product functions .....	14
3.3 User characteristics .....	14
<b>4 Software architecture .....</b>	<b>15</b>
4.1 Design of MVC model .....	15
4.1.1 Class diagram.....	15
4.2 Advantage of MVC model.....	18
<b>5 Implementation .....</b>	<b>19</b>
5.1 GUI Design .....	19
5.2 Game process.....	20
<b>6 Evaluation and verification.....</b>	<b>22</b>
6.1 Verification by boundary test.....	22
6.1.1 Normal illumination without glasses .....	22
6.1.2 Normal illumination with glasses.....	24
6.1.3 Weak illumination without glasses .....	25
6.1.3 Weak illumination with glasses .....	27
6.2 Evaluation by UI test.....	29
<b>7 Lesson learned .....</b>	<b>30</b>

<b>8 Conclusion.....</b>	<b>31</b>
--------------------------	-----------

# **1 Introduction**

## **1.1 Motivation**

With the speedy development of technology, human recognition technology gradually from relying on fingerprints, iris to face now. Face recognition technology is widely used in several fields. In the field of social security, the American National Security Service uses face recognition technology and networks to hunt fugitives nationwide[1]. The face recognition technology is also applied to electronical passport and ID card, the ICAO has identified face recognition as the primary approach between its member countries and regions[1]. And in daily life, Facebook can automatically label people in photos based on user's previous annotations. Thanks to the efforts of software engineers to face recognition technology, there are numerous face recognition, face tracking and face gesture recognition libraries based on different languages such as Python, JavaScript, C++ and so on. Some of them are based on browser environment such as tracking.js, CVV and clmtrackr.js. They are lightweight, intuitive and based on JavaScript.

Although there are several JavaScript libraries which can realize face recognition, the summary and category of their features has not been researched. In this thesis, these libraries' features are summarized and their efficiency are assessed and compared. In order to demonstrate the face recognition capabilities clearly, one interesting game concept is developed in this paper. And the realized game can be executable in a modern mobile HTML5 web browser.

The motivation to create this game is to relax users' face through a mobile game. At present, people work in the office and look at computer screen at most time. After a long period of work, people need to have a relax. This game is created to relax users mentally and physically.

## **1.2 Structure**

The structure of this paper is listed, the background and related work are stated in chapter 2. It includes the face recognition technology, existing libraries and the inclusion and exclusion from these libraries and efficiency assessment. In chapter 3, the game concept are declared. Chapter 4 explains the software architecture and the advantage of this architecture. The mobile game is implemented in chapter 5 and tested in chapter 6. Then the lessons learned are listed in chapter 7 and chapter 8 is the conclusion.

## **2 Background and Related work**

### **2.1 Face recognition technology**

As the progress of technology, biometric-based recognition technology gradually replace the traditional recognition methods such as PIN, passwords or electric cards. The PINs, passwords and electric cards are more likely to be stolen and forgotten compared to biometric-based recognition[2]. In addition to face recognition, the biometric-based recognition technology consists of fingerprint, iris, signature recognition and so on. But compared to other biometric-based recognition methods, the face recognition has several significant advantages. Firstly the test subjects can hardly realize it when they are under inspection. On the contrary, the fingerprint, iris or other biometric-based recognition methods need test subjects' subjective action otherwise recognition can't be finished. The second advantage is that the face recognition technology relies comparatively less expensive equipment. Although the accuracy of face recognition technology is affected by the background light, test subjects and so on, it can be developed by better algorithms. The advantages of face recognition make it appropriate to be applied in public area.

The application of face recognition technology is roughly divided into two classes, the one-to-one recognition and one-to-many recognition. The one-to-one recognition means only one person can be tested every time. For example, in the area of security, the access control system of some companies is based on face recognition. Each time one employee is inspected by the machine and after finished, another employee continues to be inspected[3]. On the contrary, the one-to-many recognition mode is more suitable for public area such as airports, seaports[4].

Although the traditional recognition technology seems to be completely replaced by face recognition technology, there are many factors significantly decrease the recognition accuracy. These factors can be divided into test subjects' intrinsic variations and variations of external conditions. The first group includes the age changes, whether wearing glasses and facial emotions. And the second group is mainly lead by illumination[5].

With decades of work, the accuracy of face recognition has been markedly developed. And there are some face recognition methods which get satisfying results in the lab. But unfortunately, when they are applied into real application, the recognition accuracy reduces obviously[6]. According to the relevant research, the age, light and posture variations are three main sources which decrease the recognition accuracy[7].

In the real world, the face recognition technology varies from different product demand. Some application needs to analyze static images from camera but some of them need to process video streams[8].

In the area of processing static images, there are two main approaches. The first one is feature-based method which means the recognition is based on feature points like eyes, mouth, nose and so on[9]. In this method, the most important processing step is how the feature points are extracted and the geometric relationship between them. Although this approach can lead to rapid

recognition, the accuracy of result is limited and significantly affected by the tolerance value. A high tolerance value means a lower accuracy and a low tolerance may result an omission of right image. In addition, the difficulty of this method is how to set different weight to these feature points. The second recognition method is from a total perspective and it includes statistics and AI. The statistics means comparing the similarities between two images[10]. When an image is input into the recognition system, it is compared to all images in database. As a result, the statistics approach leads to more expensive computation and the recognition result is sensitive to face orientation of test subject, illumination and so on. Next, the AI means neural network and machine learning[11]. But compared to feature-based approach, the holistic approach generates some unique advantages. For example, it has more accurate recognition result and only specific regions of image needs to be analyzed[12]. In the meanwhile, the holistic method also has some flaws. If all pixels of an image is set equally weight, it will be computation expensively[13]. In addition, the recognition accuracy is also significantly affected by illumination and the pose of test subjects[14].

When a software provides the real-time recognition, the dynamic face recognition approach is needed. The process of dynamic face recognition can be divided into three steps, the face detection, face tracking and face recognition[15]. Although this approach deals with the image streams, some good frames are filtered and then analyzed to recognize faces[16]. Compared to the static image recognition mode, the video-based recognition is more likely affected by poor quality images[17], the background variants[18] and multiple faces appearing at the same time[19]. But thanks to a greater number of input images, the approaches can choose the better frames[20].

## 2.2 Existing libraries

In this thesis, the webcam is used to get users' face and relevant software recognize and track users' faces. As a result, some technologies are used to deal with the video streams. Although there are many libraries about face recognition, in this paper, the game should run on mobile web browser. So the initial limitation is that the libraries should be written by JavaScript. And the face recognized, face tracking and emotion detected are realized on client side. After a search, the subsequent libraries are all based on JavaScript.

### 2.2.1 Research of relevant libraries

In the process of searching relevant libraries, the <https://github.com> is the main source. According to the task description of bachelor thesis, some functions such as face recognition, face tracking and face gesture recognition must be realized. And the search process is listed below.

Firstly, "face recognition" is set as key words and JavaScript is also set as the necessary condition. Then there are 493 results and they are sorted by "most stars". Because of the numerous result, only the libraries with more than ten stars are selected and analyzed.

After this search, 13 libraries meet the criteria. And they are FacebookBot, face-recognition.js, facematch, ardrone-webflight, get-me-through, face-recognition-api, Face-Recognition-Web-

Demo, CivisFace, facerec-server-node-sails, FCClientJS, face-recognition-brain, face-recognition-service and react-facial-feature-tracker.

Search results	Description	Appropriate
FacebookBot	It supports Voice Recognition, Natural Language Processing and features such as: search nearby restaurants, search trending news, transcribe and save memos to the cloud. It also save user data (with permissions, of course) such as favorite locations and can provide customized greetings. But it can't support the face recognition, face tracking and face gesture recognition.	No
Face-recognition.js	A simple node.js package which supports face detection and face recognition. Although it can provide face recognition function, it is based on Node.js.	No
Facematch	A Facebook face recognition wrapper. User can submit an image, and then query Facebook to recognize his friends' faces. But it can't deal with the video streams.	No
Ardrone-webflight	Needs the Node.js	No
Get-me-through	a free, offline, real-time, open-source web-app to assist organization of an event to allow only authorized/invited people using face recognition technology or QR code. But it needs Python3, Node.js, MongoDB and so on.	No
Face-recognition-api	An open-source built on top of python and Node.js that lets programmers build their own face recognition API.	No
Face-Recognition-Web-demo	A web demo of face recognition. And it uses the <b>OpenCV</b> as its library.	Yes
CivisFace	An iOS app for exploring facial recognition using react-native, AWS Recognition, and AWS Polly. It was made to playfully recognize employees around the office from their staff photos and to educate visitors on facial recognition technology. It is an application of IOS rather than based on web browser.	No
Facerec-server-node-sails	Sails application using <b>OpenCV</b> .	Yes

FCClientJS	A face detection and recognition JavaScript client library. It seems to be very good but it is not free. It costs at least 50 euro per month if all functions are realized.	No
Face-recognition-brain	A JSON library and it is not free to use	No
Face-recognition-service	It uses the Python, C++ as the library	No
React-facial-feature-tracker	A react component for facial feature recognition based on the <b>clmtracker</b> .	Yes

(Table 1: Result of the first search )

Secondly, “face tracking” is set as key words and JavaScript is also set as the necessary condition. Then there are 113 results and they are sorted by “most stars”. Because of the numerous result, only the libraries with more than ten stars are selected and analyzed.

After search, there are 11 results. And they are jeelizFaceFiltter, contain-facebook, brfv4\_javascript\_examples, JS-tracking-demo, ros\_robots\_projects, facebook, face-traking-p5js, everything-but-the-chat, face-detectify, pizzaCatcherJS, and FacebookTrackingRemoval.

Search results	Description	Appropriate
JeelizFaceFilter	A JavaScript lightweight face detection and tracking library to create snapchat or MSQRD like face filters. It unleashes the power of the webcam to interact with the browser in a new and fun way. And the outputs of the library are if the a face is detected or not, the position and the scale of the detected face and the rotation Euler angles. But it does not support face gesture capture or emotion detection.	No
Contain-facebook	It isolates users’ Facebook activity from the rest of their web activity in order to prevent Facebook from tracking users outside of the Facebook website via third party cookies.	No
BRFv4	A real time face detection and tracking SDK. Users put in image data (camera stream or single picture) and it outputs facial data. It seems to be appropriate but it needs a server.	No

JS-face-tracking-demo	A face recognition demo using <b>tracking.js</b> library.	Yes
Ros-robotics-projects	It is about learning Robotics using Python.	No
Facebook.tracking.exposed	A collaborative tool for algorithm investigation. It also needs the server side.	No
Face-tracking-p5js	It is used to create playful or critical web applications. But it is a set of online courses rather than a library.	No
Everything-but-the chat	It uses the <b>clmtrackr.js</b> as the library.	Yes
Face-detetcify	It uses the <b>tracking.js</b> as its library to realize the face recognition.	Yes
PizzaCatherJS	A color-tracking webcam game where users catch pizzas with their face. It uses the <b>tracking.js</b> as its library to realize face detection function.	Yes
FacebookTrackingRemoval	It removes Facebook's tracking from links and sets the proper link so they work like a normal link.	No

(Table 2: Result of the second search )

Thirdly, “face detect” is set as key words and javascript is also set as the necessary condition. Then there are 361 results and they are sorted by “most stars”. Because of the numerous result, only the libraries with more than 20 stars are selected and analyzed.

There are ten results and they are HTML5-Face-Detection, face-detection-node-opencv, web-based-face-detect, font-face-detect, font-detect.js, copterverface, face-detection-browser-opencv, detect-demo, simle-face-detector, weapp-face-detect.

Search results	Description	Appropriate
HTML5-Face-Detection	A HTML5 Video Face Detection with the <b>CCV</b> JavaScript Library.	Yes
Face-detecttion-node-opencv	A real-time face detection using <b>OpenCV</b> , Node.js, and Web-Sockets.	Yes

Web-based-face-detect	A web based interface for face detection with <b>OpenCV</b>	Yes
Font-face-detect	A feature detection for the CSS3 feature of font-face. This enables users to use a hosted font file as a font-family in their CSS. This is not relevant to human face recognition, face detection or face emotion recognition.	No
Font-detect	A font detection approach. It is not relevant to face recognition, face tracking or face emotion recognition.	No
Copterface	A script which attempts to position an ar-drone so that it centers on any face detected in the center of its field of vision. But it is based on node.js and needs a server side.	No
Face-detection-broswer-opnecv	A real-time face detection using WebRTC, <b>OpenCV</b> , Node.js, and Web-Sockets.	Yes
Detect-demo	A face detection demo application using Face++ API. But the Face++ interface is not free to use.	No
Smile-face-detector	A smile face detector and the it uses <b>OpenCV</b> .	Yes
Weapp-face-detector	A face detection interface which only can be realized on WeChat.	No

(Table 3: Result of the third search )

### 2.2.2 The summary of search results

From the three tables of search, there are some JavaScript libraries are filtered. There are OpenCV, tracking.js, clmtrackr.js , and CVV. The OpenCV supports many operation systems but not the modern HTML5 web browser. So OpenCV is excluded from this summary. Then the summary of tracking.js, clmtrackr.js and CVV are listed below.

**Tracking.js**. The tracking is an independent JavaScript library which is used to track data received from the camera in real time, and the tracked data can be either color or human face. The JavaScript events are triggered when specific colors or human faces exist or move. The tracking is also an open source and embeds algorithms into web pages. Thanks to its lightweight core(7 KB) and many methods, the tracking seems to be the best face recognition JavaScript library.

**Clmtrackr.js** . Clmtrackr.js is a library which can fit facial models to faces in videos or images. It tracks a face and output the outline of the face. However, the efficiency of clmtrackr.js is

limited when there are some faces in one image or video, it always can't detect many faces simultaneously. And if detecting a video, the browser is recommended to be supported with WebGL. This library also provides some general face models and self-annotated images.

CCV. CCV can be used to recognize and track face.

### 2.3 Selection criteria

Considering the aim of this game is to relax users, the JavaScript library should realize the face recognition, face track and face gesture recognition. In addition, the library need to be used in a mobile web game. Furthermore, the game is a single page web without server. As a result, the libraries should only be realized by JavaScript. The game is designed to run on the mobile web browser and considering the limitation of mobile CPU computing speed, the libraries should also be lightweight. In addition, the libraries should also have clear documentation. Finally, the library should be free to use without charges.

As a result, the selected library should

- 1) Realize the face detection, face tracking and face gesture recognition in a video stream.
- 2) Only be written by JavaScript and lightweight.
- 3) Have clear documentation and is free to use without any charge.

### 2.4 Exclusion and inclusion according to criteria

After the research, there are six libraries which are included in this bachelor thesis. They are tracking.js, clmtrackr.js , and CCV.

According to the first criteria, the tracking.js and CCV can't support the face gesture recognition or emotion detection.

As a result, the clmtrackr.js is selected as the library in this bachelor thesis and used to realize the basic functions of the mobile web game.

Library name	Face detection	Face tracking	Face gesture recognition	Written by JS and Lightweight	Clear documentation	No charge
Tracking.js	Y	Y	N	Y	Y	Y
CVV	Y	Y	N	Y	Y	Y
Clmtrackr.js	Y	Y	Y	Y	Y	Y

(Table 4: Assessment of result libraries)

## 2.5 Assessment and comparison of the libraries

### 2.5.1 Clmtrackr.js

Clmtrackr.js is based on the Constrained Local Models, also calls CLM approach[21] and the “ Face Alignment through Subspace Constrained Mean-shifts” algorithm[22]. It detects the face as 70 points and each point is filtered by unique special classifier. But due to the numerous classifiers, the efficiency is very low when this algorithm is used to recognize face. However, this drawback is well resolved by GPU, which can be used by riWebGL[21].

Although the efficiency has been improved, clmtrackr.js has two main problems. Firstly, it can't recognize well when user's posture deviates from the mean shape[21]. Secondly, this library applies 3D model rather than a 2D model. It leads to a higher processor resource occupancy and lower operating speed[21].

### 2.5.2 Tracking.js

Tracking.js is a lightweight JavaScript library. The core code is about 7KB[23]. It provides an implementation of “Features from Accelerated Segment Test” for features detection, which is also called as FAST[23]. It finds corners from part of the image. Although it is much faster than many other approaches, it can't detect the emotions.

### 2.5.3 CVV

The CVV realizes the state-of-art algorithms, includes the image classifier, frontal face detector and some object detectors. It can runs on many operation systems and is easy to be integrated to servers[24].. But when there is no server side in an application, the CVV is no longer useful.

Name of library	Used approach	Comparison of efficiency	Limitations	Opportunities
Clmtrackr.js	Face Alignment through Subspace Constrained Mean-shifts	Compared to tracking.js, it provides less efficiency and more accuracy. But with the usage of GPU, it is still more efficient than other libraries.	Less efficiency	Accurate face detection and emotion detection is required.
Tracking.js	Features from Accelerated Segment Test	In the area of face detection, it is the most efficient.	Less accuracy	Mass face recognition in public area
CVV	Image classifier, frontal face detector and some object detectors	If there is a server side in application, CVV is most efficient and easier to be integrated. Otherwise, it is less efficient compared to other libraries.	Useless in web application without server side.	Server side is mandatory in an application

(Table 5: Comparison of library efficiency)

# 3 Game Concept

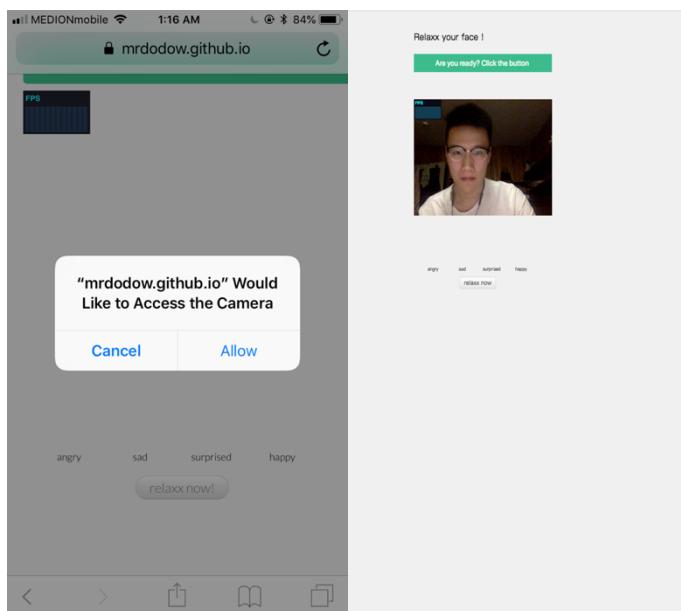
## 3.1 Description

The operation procedures are listed below. Firstly, users open this mobile game by a phone with HTML5 web browser and input the website (<https://mrdodow.github.io/>). Then a box popups and asks for user's permission of using front camera. After clicking the "Allow", user can enter the game. The game is a single-page game and the page is divided into three parts.

In the upper part, the text line shows different commands orderly. User can show any emotions after read the command and will get different reactions. For example, the text line shows "Show your happiness to endless work". And if user shows a "happy" expression, the command will change to "Good! What a positive attitude.". Then the command line shows "Show your surprise for overtime today" and if users shows "surprise" emotion, the command line will show "Relax, it is normal". But if user does not follow the order, the command line will not change.

The middle window shows the user's face and tries to detect it by the dynamic green line which dynamically recognizes and tracks user's face. On the other hand, the accuracy of recognition and tracking is affected by illumination, gestures, the expressions of test subjects and so on.

In the bottom part, there is an emotion table which has four emotion categories. And they are anger, sadness, surprise and happiness. The software can dynamically detect the user's emotions and classifies them into the four categories. When the software detects the user's emotion successfully, a corresponding emotion image will appear on the emotion table. Under the emotion table, there is a begin button (relax now!). User can't be recognized or detected until he click the begin button.



(Figure 1: The pop-up box) (Figure 2: The game page)

### **3.2 Product functions**

Several functions are realized in this game. And four of them are the main functions, they are recognizing the user's face, tracking the user's face, user's emotion detection and response to user's input.

<b>Recognize face</b>	User's face are obtained by the mobile front camera. This software use algorithms to recognize the user's face and also can outline it.
<b>Track face</b>	When playing the game, user can shake his head, blink the eyes or open month. The software can recognize face dynamically which means tracking face. No matter where the user's face is, the software can track and outline it by the green lines.
<b>Detect emotion</b>	User can show any emotions after read the command. The software can use the algorithm to identify the address of key nodes such as mouth, eyes and nose. Through analyzing these key nodes, it can detect the emotion and use four categories to describe the emotion. The four emotion categories are anger, sadness, surprise and happiness.
<b>Response to user's input</b>	When one order shows on the command line and user makes the right emotion, the game can responds to user's input and automatically shows the next command. In this game, then user's most input is through mobile front camera.

**(Table 6: Product functions)**

### **3.3 User characteristics**

This game is particularly suitable for people who have worked long time, it can provide mentally and facial relax for them. And there are some user characteristics:

- 1) The users have some basic computer or mobile knowledge.
- 2) The users have a mobile installed with modern HTML5 web browser and connected to Internet.
- 3) The users tend to have a relax through electronic products such as mobile phone.
- 4) Mobile game is an important entertainment choice for users in their fragmentation time.
- 5) The users are willing to interact with software and follow the commands in the game.

## 4 Software architecture

### 4.1 Design of MVC model

MVC model is a design pattern[25]. And it divides the information display from users interaction[26]. In a MVC model, it is separated into three parts and each of them has the specific task.

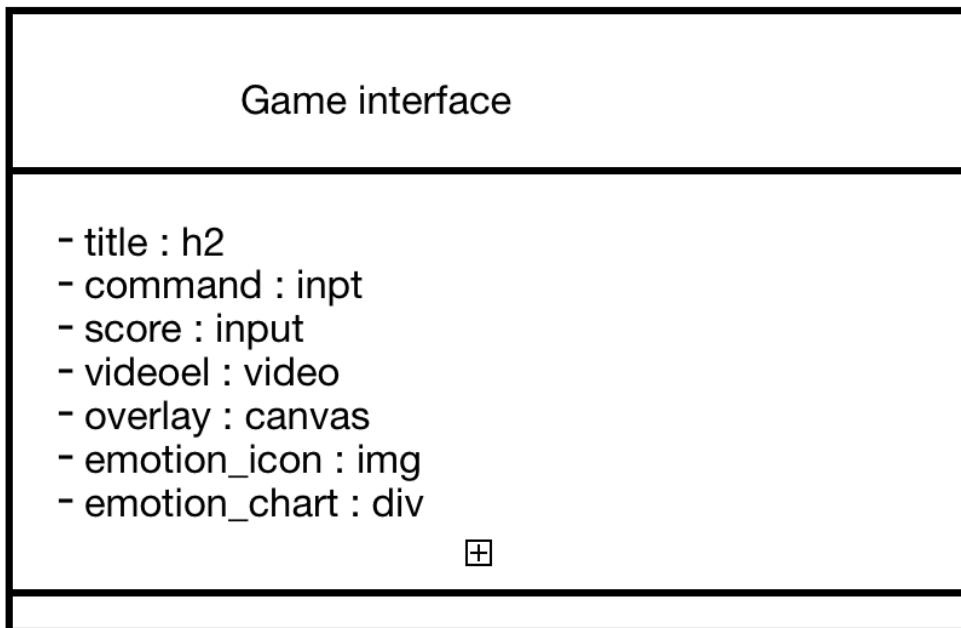
The Model in the architecture is related to business logic and date management. It provides methods of data access and validation. In most case, the Model layer handles the data interaction in the application[27].

The View layer in this architecture represent the GUI components in this application, it includes the buttons, text box, diagrams and all other HTML elements[28]. Through the View layer, the work can be separated into aesthetic design and date logic pattern. In the implementation phase of the application, HTML design can be adapted frequently without logic modifying.

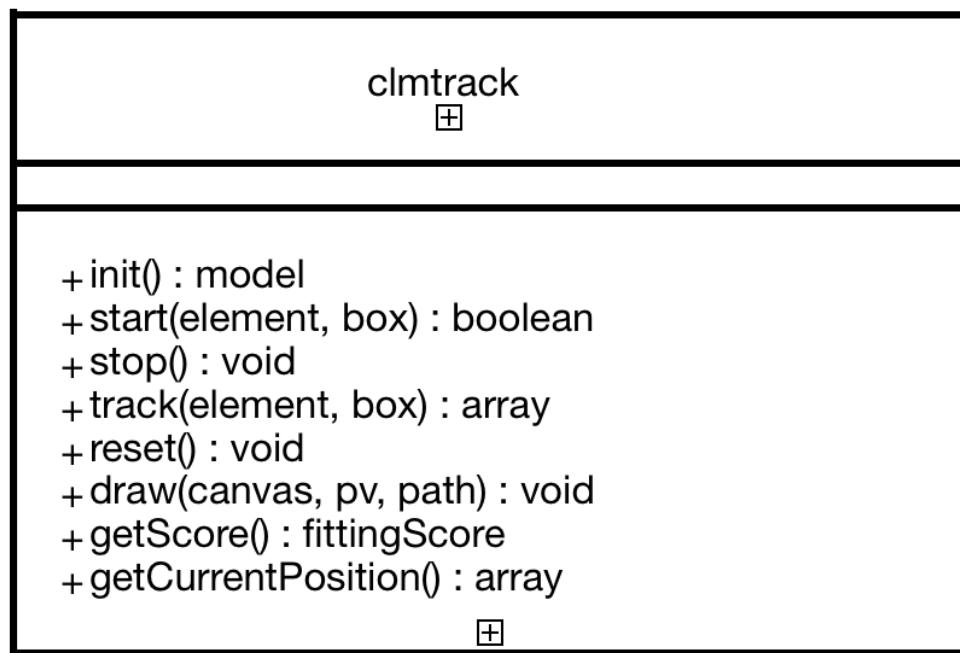
The Control layer is the logic layer in this architecture. It manages the relationship between the View layer and Model layer. The controller can be triggered by user's operations on View layer. In addition, it can get data from Model layer and return the relevant result to View Layer[28].

#### 4.1.1 Class diagram

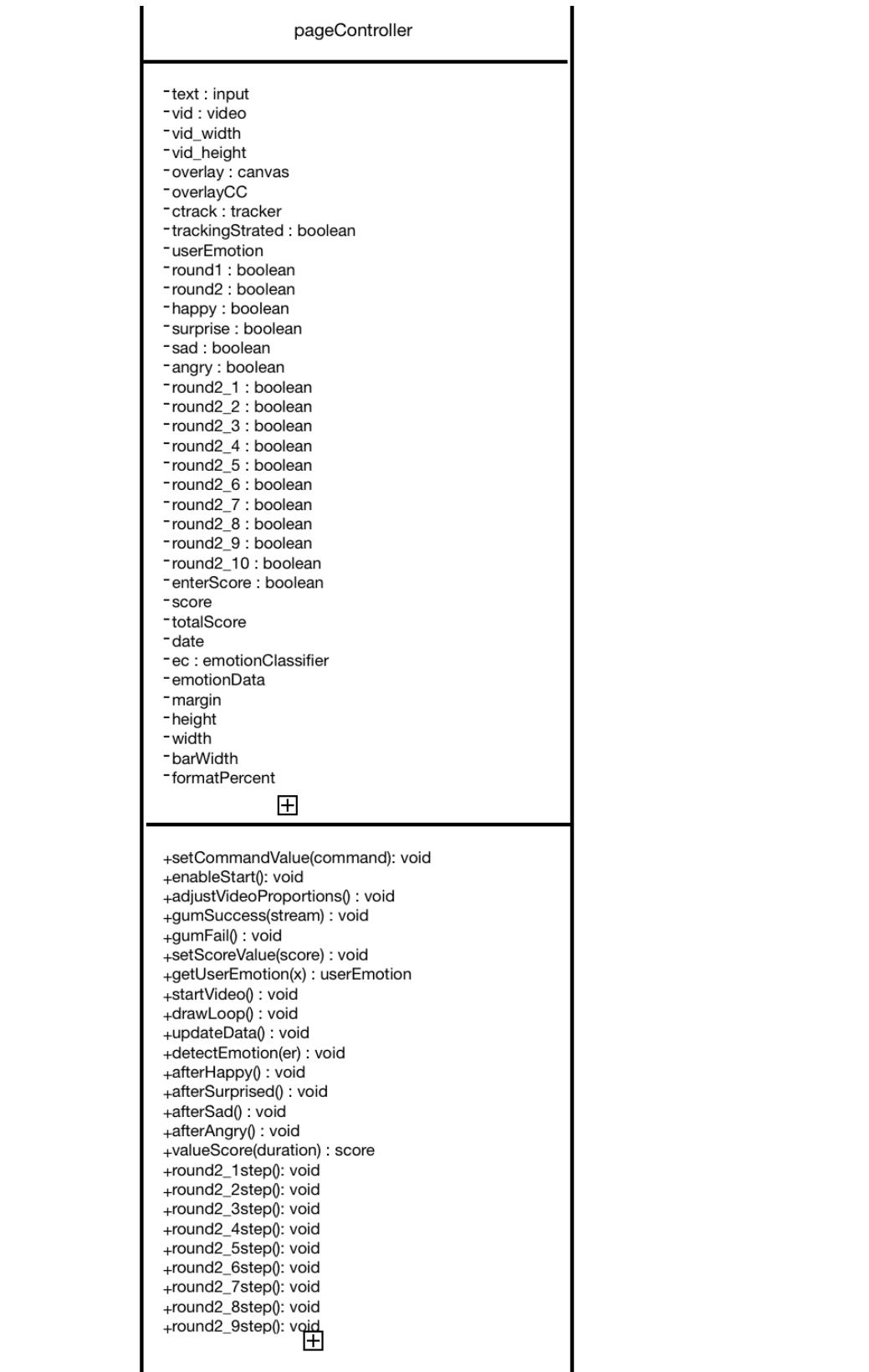
In this game, the View layer includes all HTML elements of the mobile game such as button, images, video stream and text area. The Controller layer is the pageController.js which controls the interaction between user and View layer and update the data of Model layer. The Model layer is the clmtrackr.js which includes the all information of face model.



(Figure 3: Viewer)



(Figure 4: Model)



**(Figure 5: Controller)**

## **4.2 Advantage of MVC model**

High reusability. The MVC model can access the data via different View Layers by only changing the data format in the Controller and without changing the data in the Model Layer.

Maintainability. The business logical layer, View layer and the Data Layer have been separated, so if there are some bugs which you don't need to go through all the classes in the project. Only need to change some specific parts. Thus, it also increases the logicality and the readability.

Transportability. Since the View and the Controller layer are separated, it can be transplant to other operation system

Efficiency. Because of the separation of the layer, different developer can deal with different parts, which increase the working efficiency.

# 5 Implementation

## 5.1 GUI Design

The game is designed to analyze the face recognition capabilities. As a mobile game, the UI design should be significantly considered. The usability heuristics is a standard of assessing the user interface design of a software[29]. The listed usability heuristics are considered in the implementation of the game.

**User freedom and control**[30]. In this game, users are free to take the whole control of the software. Users can shake their heads, show any emotions and so on. The software can automatically recognize and track the users' face and detect their emotion.

**Match between application and real world**[30]. The game is an entertainment mode for users. The orders shown in the game are based on the real world. In addition, the game has different response according to users' different emotions.

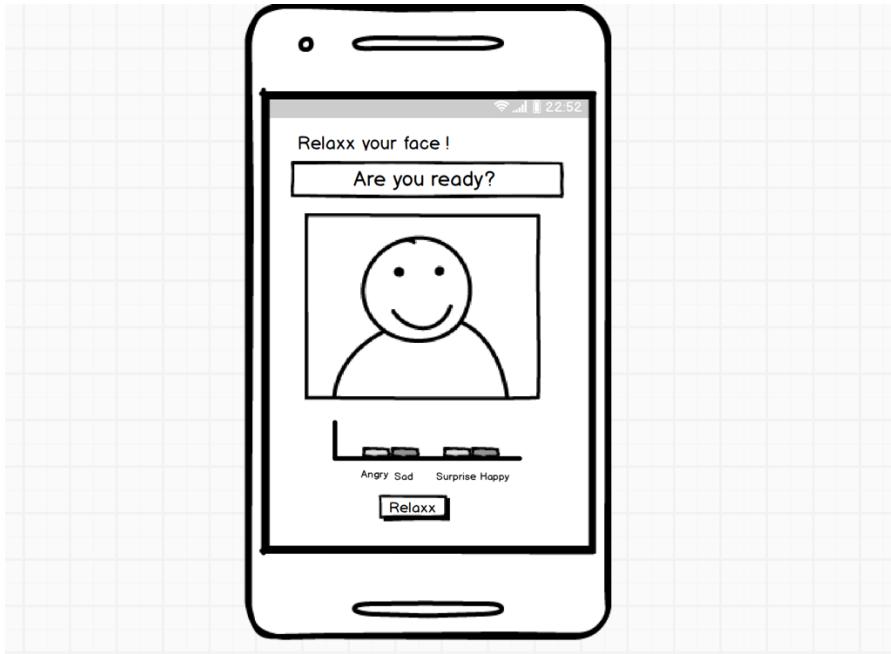
**Consistency and standards**[30]. This mobile game is designed in a unified standard. All functions and orders are designed in a unique and recognizable name. User can easily work with this system.

**Error prevention**[30]. After each order is displayed, the game will not switch to another appropriate order until the user correctly follow the order and show the right emotion.

**Flexibility and efficiency of use**[30]. This game is based on mobile web. That means users can play it by mobile phones or computers as long as connected to internet. Users do not need to install any software and they just open web browser and input MrDodow.github.io.

**Aesthetic and minimalist design**[30]. The game page only has necessary information for users and is without irrelevant information.

**Help users recognize from errors**[30]. After each order is displayed, the order will not change until the user correctly follow the order and show the right emotion.



**(Figure 6: Mock-up of mobile version)**

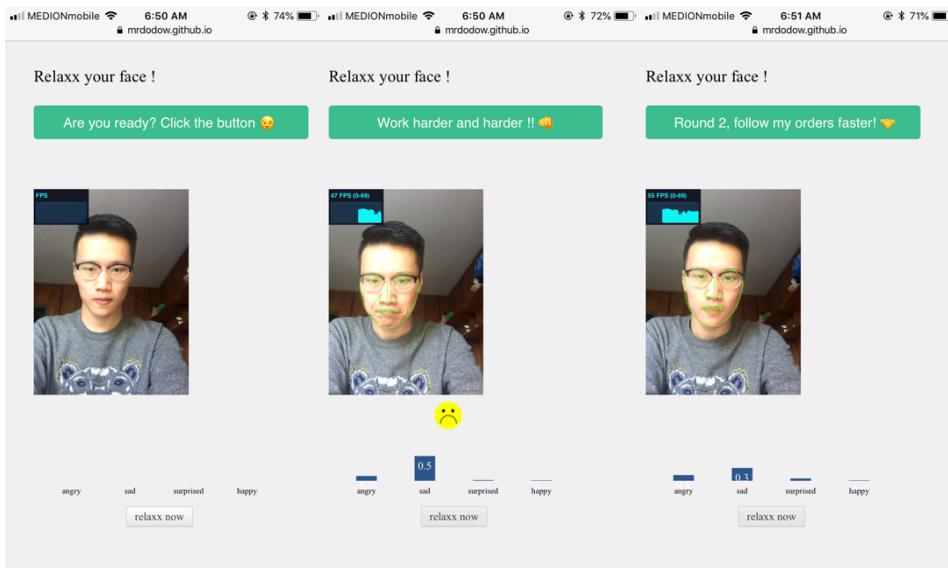
## 5.2 Game process

Firstly, user inputs the website <https://mrdodow.github.io> in his mobile web browser or laptop web browser. Then he should allow the use of webcam. Otherwise there is a pop-up box which shows “Fail to fetch video from camera, please try again.”. After that, the user enters the main page of this game. And this main page is divided into three parts. The upper part is the name of this game “Relaxx your face !”. And a text box which shows “Are you ready? Click the button”. The middle part is the main part of this page, the video stream of user himself from the webcam. And the lower part is the emotion chart and a start button.

If user clicks the “relax now” button, the software runs and the green text area will show 3 commands orderly and they are “System is loading...”, “Round 1, ready?” and “Now, follow my orders”. After that, the software begins to recognize, track user’s face and recognize the user’s emotion. The user’s face is outlined by dynamic green line. At the same time, the green text area shows “Show your happy to endless work” and software waits for user’s happy emotion. The command doesn’t change to “Good, what a positive attitude” until the user shows a happy emotion. Several seconds after that, the command shows “Show your surprise for overtime today”. And the command doesn’t change to “Relax, it is normal” until the user shows a surprised emotion. Then the command will show “See your boss new Ferrari, show your sad”. If the user follows the command and shows a sad emotion, the command changes to “Work harder and harder !!” and “Then your boss can buy another new one!” after some seconds. After that, the command changes to “Show your angry for coming deadline”. The command will not show “Just do it after game” until the user shows a sad emotion.

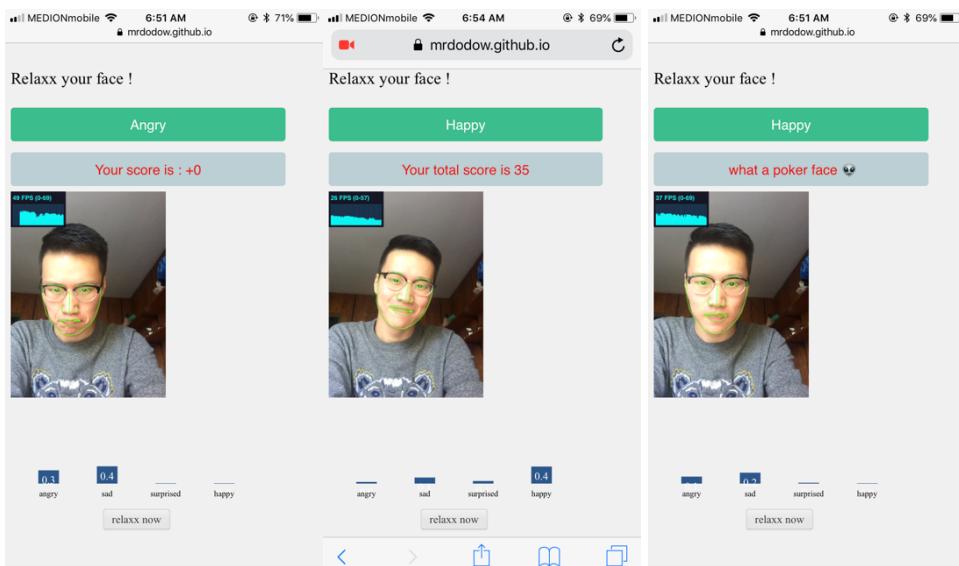
Then, the command shows “Good, you have passed round 1” and “Round 2, follow my orders faster! ” several seconds later. That means the user has passed the round 1 and begins the round 2.

Round 2 is a scoring mode. The command orderly shows “happy”, “surprised”, “sad”, “angry”, “happy”, “sad”, “happy”, “sad”, “surprised”, and “happy”. There are 10 commands and every time user shows the right emotion, there will be another text area which shows “Your score is : + score” and it is below the command area. The score depends on the time user consumes every time. If the reaction time is less than 1 seconds, the score is 10 and if it is less than 2 second, the score is 5. Otherwise, the score is 0. After finished the whole 10 commands, the user will get his total score and unique conclusion which is below the command area. If the total score is less than 20, the conclusion is “what a stiff lip” and when it is less than 50, the conclusion will be “what a poker face”. When the total score is less than 80, the conclusion is “what a flexible face” and if it is larger than 80, the conclusion changes to “what an emotion madman”.



(Figure 7: Main page) (Figure 8: Round 1)

(Figure 9: Round 2)



(Figure 10: Score)

(Figure 11: Total score)(Figure 12: Conclusion)

# 6 Evaluation and verification

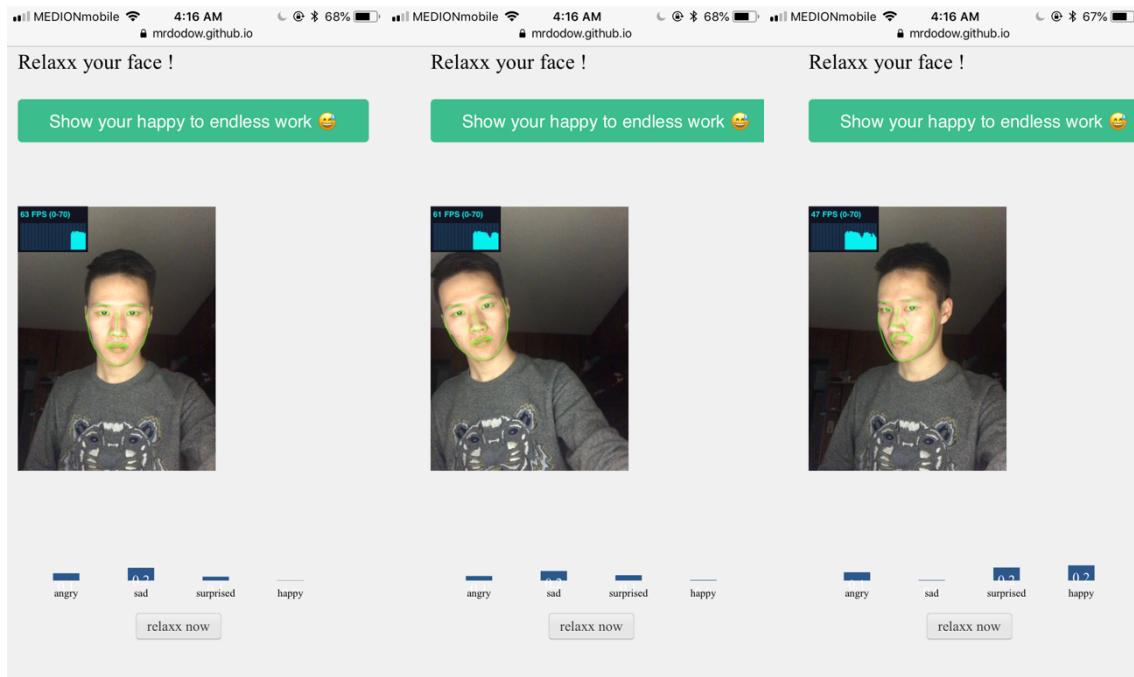
## 6.1 Verification by boundary test

There is no input area from users except the video stream from webcam. So the functions of the software is tested in different environments. In addition, the face recognition accuracy is affected more by illumination[5], face gesture[7] and facial decoration such as glasses[7]. So the boundary test is divided into four cases and they are normal illumination without glasses, normal illumination with glasses, weak illumination without glasses and weak illumination with glasses.

### 6.1.1 Normal illumination without glasses

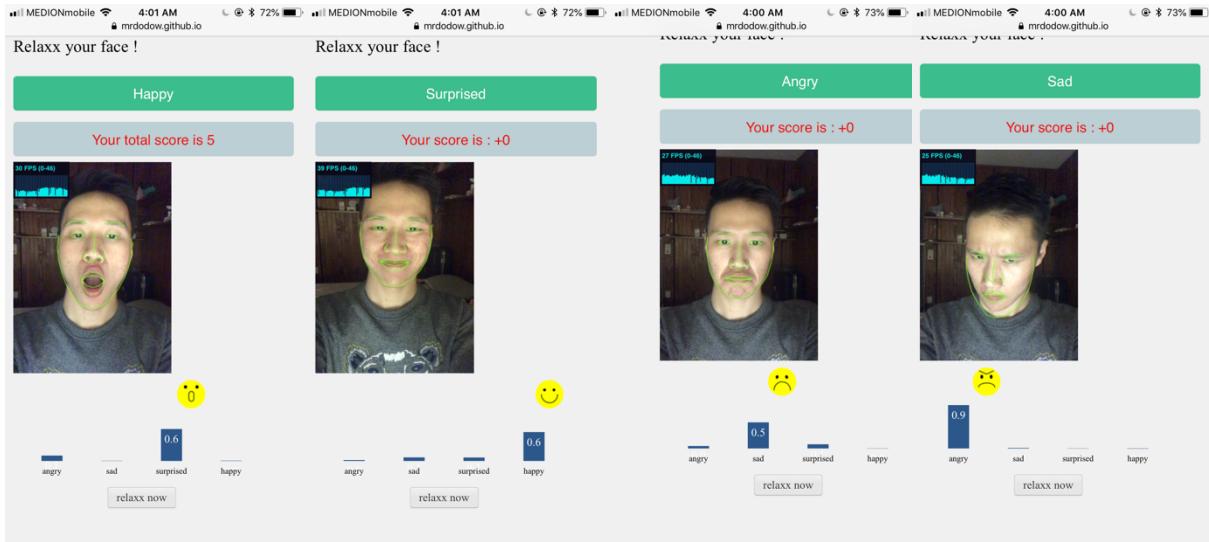
The primary functions of this game are face recognition, face tracking and face gesture recognition.

So firstly, the software is tested whether user's face can be detected and tracking. The software is tested when user's face is in the center, edge of the screen or with skewness. The software successfully detects and tracks the face.



(Figure 13: Face in the center)(Figure 14: Face on the edge)(Figure 15: Face with skewness)

Then the software is tested whether the face emotion can be detected.



(Figure 16: Surprised in case 1) (Figure 17: Happy in case 1)(Figure 18: Sad in case 1)(Figure 19: Angry in case 1)

Under the case of normal illumination without glasses, the software can detect the four kinds of emotions successfully.

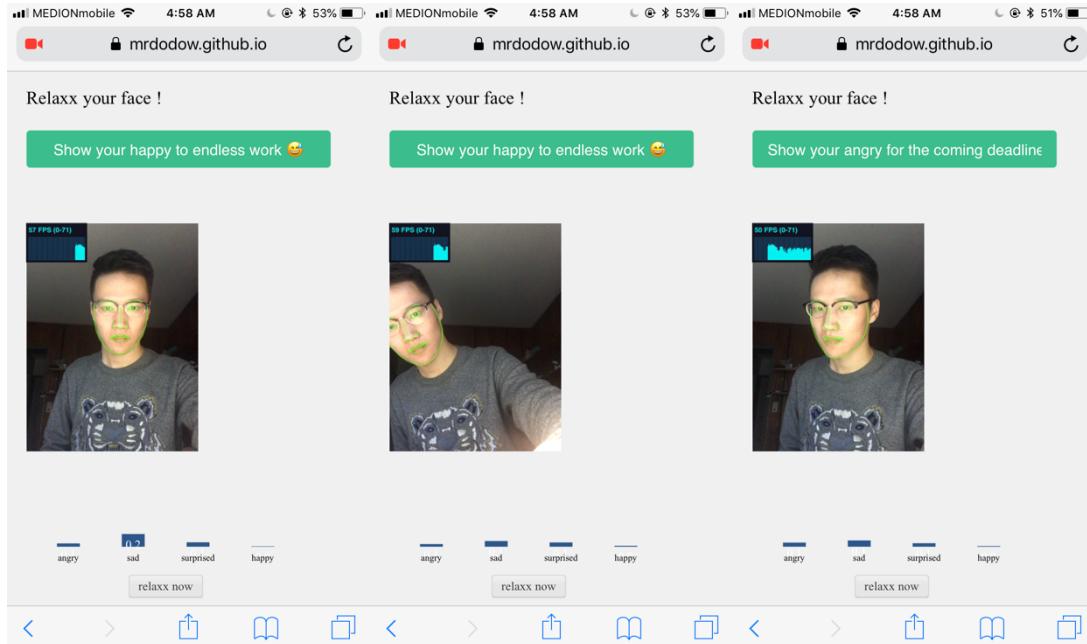
Secondly, the other non-primary functions are tested.

Tested function	User's input	Test result	Meet the requirement
Access to camera	Cancel	A popup box	Y
	Allow	Enter the game	Y
Page element display	No	Normal display	Y
	Any Click	Normal display	Y
Button triggered	Click	Command changes	Y
	No	No changes	Y
After Happy	Happy emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
After Surprised	Surprised emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
After Sad	Sad emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
After Angry	Angry emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
Round 2	Emotions from command	Command changes	Y
	Any other emotions	Command doesn't change	Y
Show total score	Any click	Total score shows	Y
	No	Total score shows	Y

(Table 7: Non-primary functions test result under case 1)

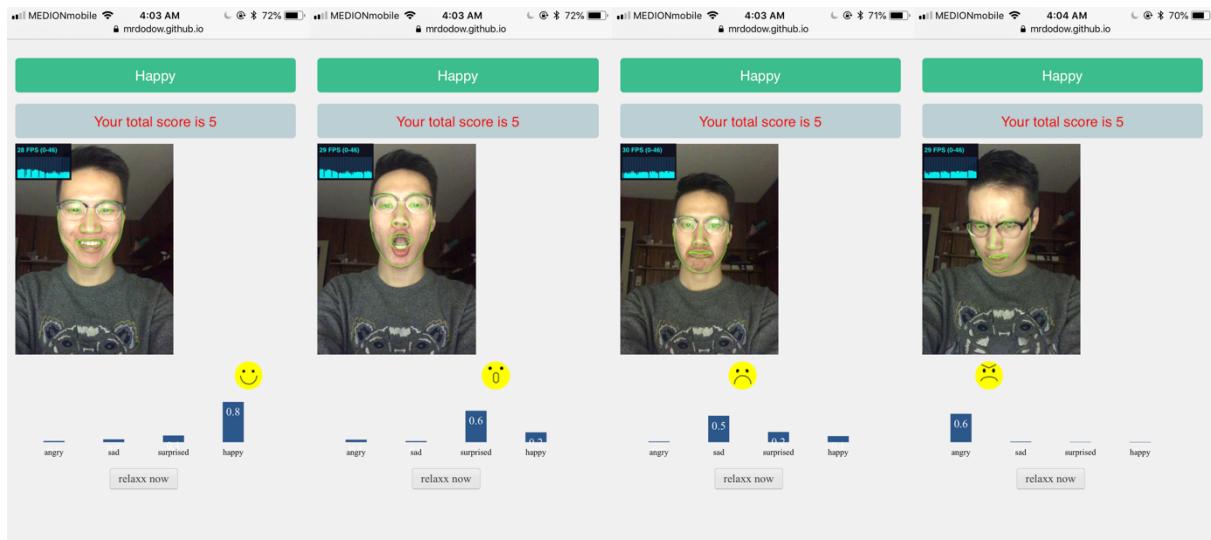
### 6.1.2 Normal illumination with glasses

Firstly, the software is tested whether user's face can be detected and tracking. The software is tested when user's face is in the center, edge of the screen or with skewness. The software successfully detects and tracks the face.



(Figure 20: Face in the center)(Figure 21: Face on the edge)(Figure 22: Face with skewness)

Then the software is tested whether the face emotion can be detected.



(Figure 23: Happy in case 2) (Figure 24: Surprised in case 2)(Figure 25: Sad in case 2)(Figure 26: Angry in case 2)

Under the case of normal illumination with glasses, the software can detect the four kinds of emotions successfully.

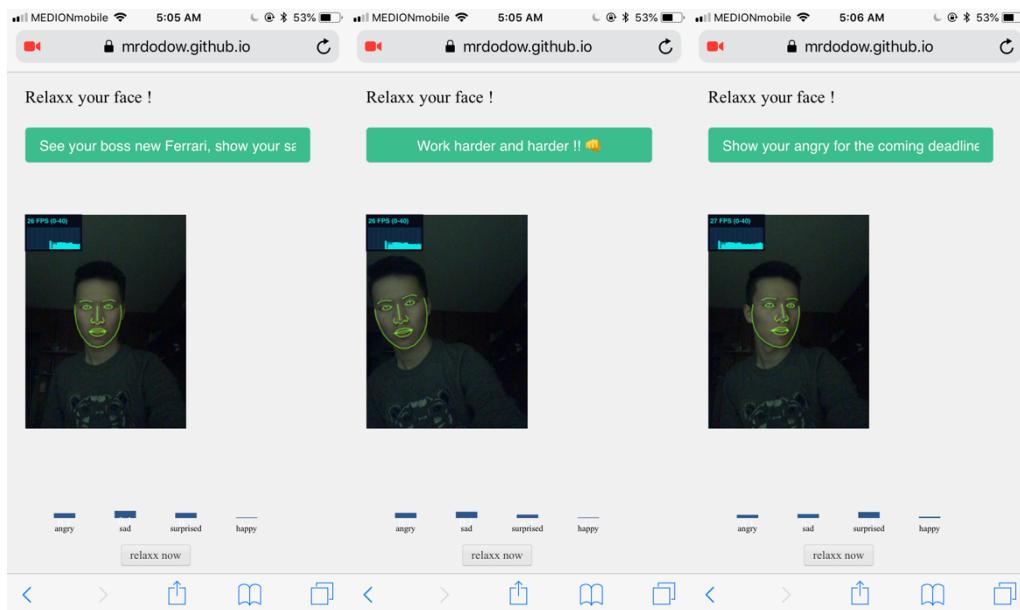
Secondly, the other non-primary functions are tested.

Tested function	User's input	Test result	Meet the requirement
Access to camera	Cancel	A popup box	Y
	Allow	Enter the game	Y
Page element display	No	Normal display	Y
	Any Click	Normal display	Y
Button triggered	Click	Command changes	Y
	No	No changes	Y
After Happy	Happy emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
After Surprised	Surprised emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
After Sad	Sad emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
After Angry	Angry emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
Round 2	Emotions from command	Command changes	Y
	Any other emotions	Command doesn't change	Y
Show total score	Any click	Total score shows	Y
	No	Total score shows	Y

(Table 8: Non-primary functions test result under case 2)

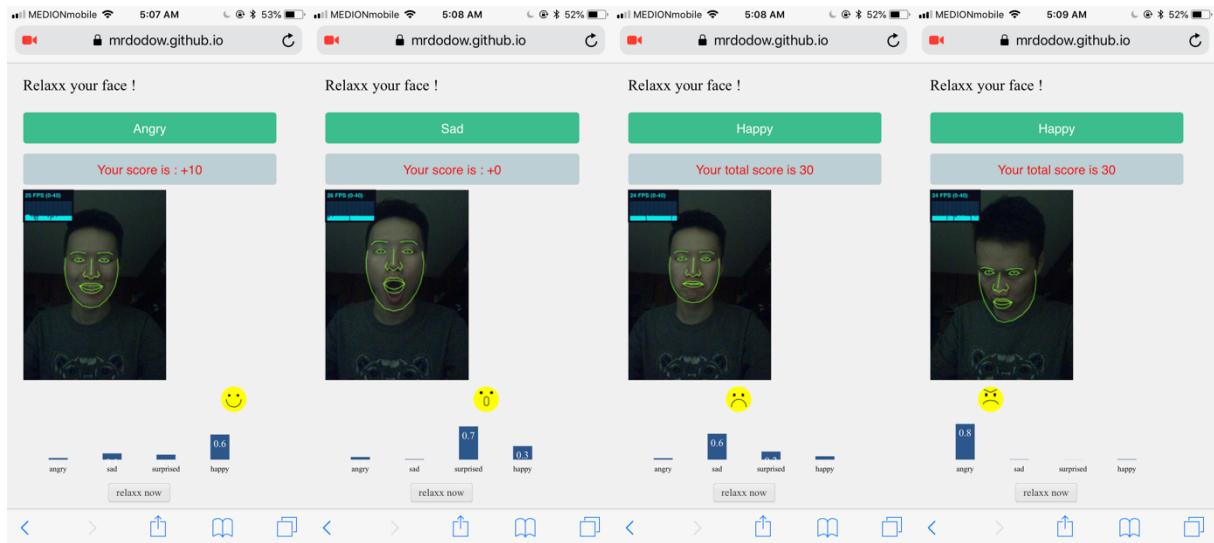
### 6.1.3 Weak illumination without glasses

Firstly, the software is tested whether user's face can be detected and tracking. The software is tested when user's face is in the center, edge of the screen or with skewness. The software successfully detects and tracks the face.



(Figure 27: Face in the center)(Figure 28: Face on the edge)(Figure 29: Face with skewness)

Then the software is tested whether the face emotion can be detected.



(Figure 30: Happy in case 3) (Figure 31: Surprised in case 3)(Figure 32: Sad in case 3)(Figure 33: Angry in case 3)

Under the case of weak illumination without glasses, the software can detect the four kinds of emotions successfully.

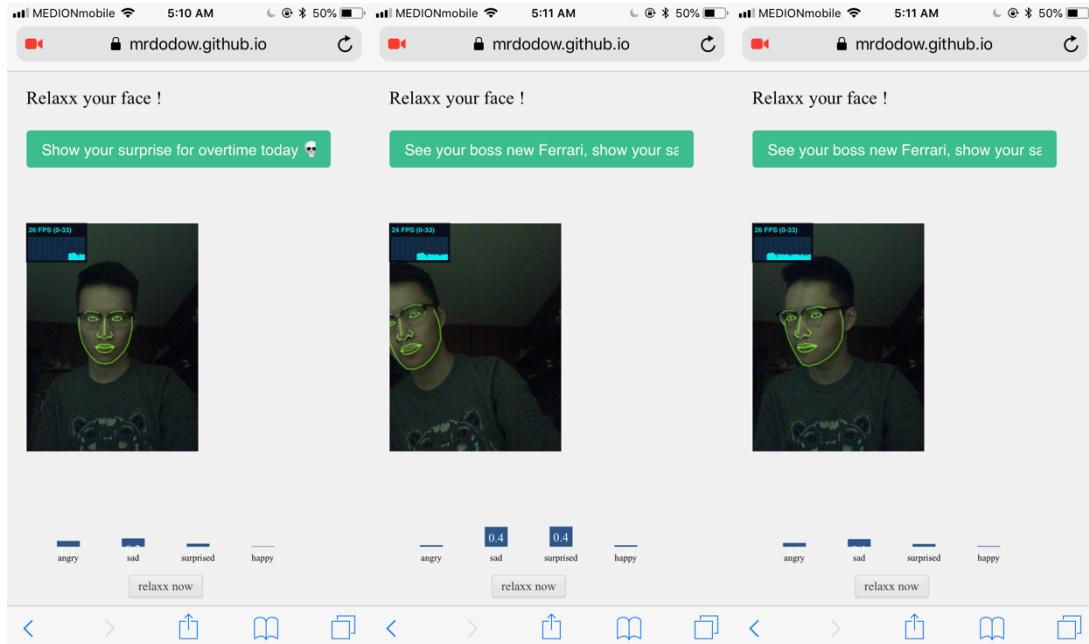
Secondly, the other non-primary functions are tested.

Tested function	User's input	Test result	Meet the requirement
Access to camera	Cancel	A popup box	Y
	Allow	Enter the game	Y
Page element display	No	Normal display	Y
	Any Click	Normal display	Y
Button triggered	Click	Command changes	Y
	No	No changes	Y
After Happy	Happy emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
After Surprised	Surprised emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
After Sad	Sad emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
After Angry	Angry emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
Round 2	Emotions from command	Command changes	Y
	Any other emotions	Command doesn't change	Y
Show total score	Any click	Total score shows	Y
	No	Total score shows	Y

(Table 9: Non-primary functions test result under case 3)

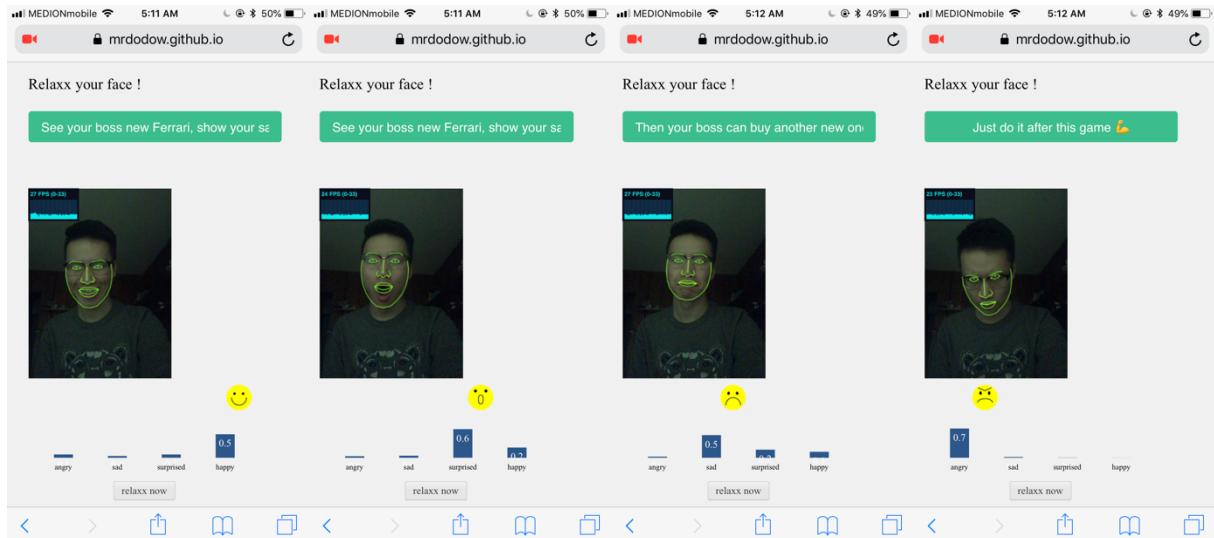
### 6.1.3 Weak illumination with glasses

Firstly, the software is tested whether user's face can be detected and tracking. The software is tested when user's face is in the center, edge of the screen or with skewness. The software successfully detects and tracks the face.



(Figure 34: Face in the center)(Figure 35: Face on the edge)(Figure 36: Face with skewness)

Then the software is tested whether the face emotion can be detected.



(Figure 37: Happy in case 4) (Figure 38: Surprised in case 4)(Figure 39: Sad in case 4)(Figure 40: Angry in case 4)

Under the case of weak illumination with glasses, the software can detect the four kinds of emotions successfully.

Secondly, the other non-primary functions are tested.

<b>Tested function</b>	<b>User's input</b>	<b>Test result</b>	<b>Meet the requirement</b>
Access to camera	Cancel	A popup box	Y
	Allow	Enter the game	Y
Page element display	No	Normal display	Y
	Any Click	Normal display	Y
Button triggered	Click	Command changes	Y
	No	No changes	Y
After Happy	Happy emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
After Surprised	Surprised emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
After Sad	Sad emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
After Angry	Angry emotion	Command changes	Y
	Any other emotions	Command doesn't change	Y
Round 2	Emotions from command	Command changes	Y
	Any other emotions	Command doesn't change	Y
Show total score	Any click	Total score shows	Y
	No	Total score shows	Y

(Table 10: Non-primary functions test result under case 4)

## 6.2 Evaluation by UI test

In order to test the UI, ten people who have few knowledge about computer science were asked to play this game. Test cases were thought out before asking the users. And the test users do not get help during the usage and the time for each test case was noted in seconds and the total score is recorded.

- Test case 1: Pass the round 1
- Test case 2: Pass the round 2 and record the total score.

Here is the table for the time each user consumed for each test case and the final total score.

Tester number	Test case 1	Test case 2	Total score
No.1	63	36	30
No.2	90	50	10
No.3	76	52	15
No.4	61	47	0
No.5	110	60	10
No.6	70	51	20
No.7	55	34	35
No.8	82	49	15
No.9	94	69	10
No.10	79	54	20

(Table 11: UI Test result)

During the UI test, following problems are noticed:

- 1) Original page is too small, its size must be adjusted before playing.
- 2) The “relax now” button is not remarkable.
- 3) The elements of page is not so neat.
- 4) The commands changes too fast to read it.
- 5) The text of command are is a little small.
- 6) The software needs a short time adjust before accurate detection.
- 7) Sometimes user’s emotion is detected as two different categories at the same time
- 8) The phone becomes very hot during playing the game.

## 7 Lesson learned

In the process of retrieving literature about face recognition technology and its history. Face recognition has more advantages in the public area. Although many new face recognition approaches have high accuracy in the lab, there are many variations such illumination and facial emotions. The recognition accuracy decreases significantly in practical use. The face recognition technology has been combined with neural network, which will increase the recognition efficiency and accuracy.

In the process of retrieving literature about face recognition libraries, there are many libraries which based on C, C++, Python or Java. In order to collect the face recognition libraries based on JavaScript, GitHub is used and 34 projects are analyzed to figure out the used libraries. Every different library is suitable for different conditions. After getting many results, a table is a better choice. Every result is listed with its description and whether appropriate for the application. It is more straightforward to collect the suitable libraries.

In the process of assessing and comparing different libraries, the documentation of each library is essential for evaluation. Through the used algorithms and development history, the assessment can be more precise. For example, clmtrackr.js uses convolution via FFT to significantly decrease the operation quantity and uses GPU by WebGL to increase the efficiency.

In the phase of game concept, the entertainment should be emphasized. Because only users feel interested in this game, the game makes sense. Originally, there is only one round in this game. After considering the entertainment, round 2 is added to this game. In addition, more interesting human-computer interaction are integrated into this game.

At the state of implementation, the clmtrackr.js is used as the face recognition library. The source code is carefully read due to many functional requirements. There are thousands of lines of codes and hard to understand. The MVC architecture is used to divide the library files into the controller and model. Furthermore, the MVC Model is also used in game implementation. Because the processors of computer and mobile phone run at different speeds. The game runs at a less accuracy and efficiency at mobile phones. In order to solve this problem, more concise code and functions are used to increase the efficiency.

## **8 Conclusion**

In this thesis, the face recognition technology and its history are summarized. Then 34 projects are collected and analyzed to filter its library. And according to the functional requirements of game, 3 qualified libraries are selected. The performance and efficiency of each library is compared and analyzed. In addition, the features, opportunities and limitations of each library is assessed. The clmtrackr.js is filtered as the library which is used in game implementation. An interesting mobile game concept is developed and realized using many technologies, such as HTML, CSS, JavaScript and so on. In addition, the MVC architecture is used in the implementation. Finally, the game is executable in a modern mobile HTML5 web browser and tested by boundary test and UI test.

There are still some areas should be improved. Firstly, the MVC architecture should be more concise and the index.html is too redundancy. Secondly, the CSS style should be improved for aesthetic interface. Finally, the recognition accuracy and the game running speed should be increased.

## Reference:

- [1] ICAO Doc 9003(2015). Machine Readable Travel Documents. [online]. Last accessed 30 March 2018 at: [https://www.icao.int/publications/Documents/9303\\_p9\\_cons\\_en.pdf](https://www.icao.int/publications/Documents/9303_p9_cons_en.pdf)
- [2] Rabia Jafri and Hamid R. Arabnia (June 2009). A Survey of Face Recognition Techniques
- [3] J. N. K. Liu, M. Wang, and B. Feng (2005) "iBotGuard: an Internet-based intelligent robot security system using invariant face recognition against intruder"
- [4] K. Kim, "Intelligent Immigration Control System by Using Passport Recognition and Face Verification," in International Symposium on Neural Networks. Chongqing, China, 2005, pp.147-156.
- [5] S. Gong, S. J. McKenna, and A. Psarrou., Dynamic Vision: From Images to Face Recognition: Imperial College Press (World Scientific Publishing Company), 2000.
- [6] J. Yang, X. Chen, and W. Kunz, "A PDA-based face recognition system," in Proceedings of sixth IEEE Workshop on Applications of Computer Vision. Orlando, Florida, 2002, pp.19-23.
- [7] X. Q. Ding and C. Fang, "Discussions on some problems in face recognition," in Advances In Biometric Person Authentication, Proceedings, Vol. 3338, Lecture Notes In Computer Science: Springer Berlin / Heidelberg, 2004, pp.47-56.
- [8] A. F. Abate, M. Nappi, D. Riccio, and G. Sabatino, "2D and 3D face recognition: A survey," Pattern Recognition Letters, Vol.28, pp.1885-1906, 2007.
- [9] T. Kanade, "Picture Processing System by Computer Complex and Recognition of Human Faces," Kyoto University, Japan, PhD. Thesis 1973.
- [10] R. J. Baron, "Mechanisms of Human Facial Recognition," International Journal of Man -Machine Studies, Vol.15, pp.137-178, 1981.
- [11] D. DeMers and G. W. Cottrell, "Non-linear dimensionality reduction," Advances in Neural Information Processing Systems, Vol.5, pp.580-587, 1993.
- [12] T. Jebara, "3D Pose Estimation and Normalization for Face Recognition," Center for Intelligent Machines, McGill University, Undergraduate Thesis May, 1996.
- [13] R. Cendrillon and B. C. Lowell, "Real-Time Face Recognition using Eigenfaces," in Proceedings of the SPIE International Conference on Visual Communications and Image Processing, Vol.4067, 2000, pp.269-276.
- [14] C. Beumier and M. Achteroy, "Automatic Face Recognition," in Proceedings symposium IMAGING. Eindhoven, The Netherlands, 2000, pp.77-89.
- [15] L. Torres, L. Lorente, and J. Vilà, "Face recognition using self-eigenfaces," in International Symposium on Image/Video Communications Over Fixed and Mobile Networks. Rabat, Morocco, 2000, pp.44-47.
- [16] R. Chellappa, C. L. Wilson, and S. Sirohey, "Human and machine recognition of faces: A survey," Proceedings of the IEEE, Vol.83, pp.705-740, 1995.
- [17] M. C. Chiang and T. E. Boult, "Local Blur Estimation and Super-resolution," in Proceedings, IEEE Conference on Computer Vision and Pattern Recognition, 1997, pp.821-826.
- [18] M. H. Yang, D. Kriegman, and N. Ahuja, "Detecting faces in images: a survey," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.24, pp.34-58, 2002.
- [19] Z. Liposcak and S. Loncaric, "A scale-space approach to face recognition from profiles," in Proceedings of the 8th International Conference on Computer Analysis of Images and Patterns, Vol. 1689, Lecture Notes In Computer Science. London, UK: SpringerVerlag, 1999, pp.243-250.
- [20] A. Howell and H. Buxton, "Towards unconstrained face recognition from image sequences," in Proceedings of the Second IEEE International Conference on Automatic Face and Gesture Recognition, 1996, pp.224-229.
- [21] Xiaoguang Yan(March 24, 2011) Constrained Local Model for Face Alignment, a Tutorial
- [22] J. Saragih, S. Lucey, and J. Cohn. Face alignment through subspace constrained mean-shifts. In ICCV, 2009.
- [23] Retrieved from <https://trackingjs.com/>
- [24] Retrieved from <https://github.com/liuliu/ccv>

- [25] Luo GuangChun, WangYanhua, Lu Xianliang and Hanhong (March,2003). A Novel Web Application Frame Developed by MVC.
- [26] Kevin McArthur, Pro PHP: Patterns, Frameworks, Testing and More, Apress, 2008.
- [27] A Freeman, S Sanderson, Pro ASP.NET MVC 3 Framework, Apress, 2011.
- [28] W. J. Gilmore, Easy PHP Websites, Columbus, Ohio: W.J. Gilmore, LLC, 2009.
- [29] Nielsen, J., and Molich, R. (1990). Heuristic evaluation of user interfaces, Proc. ACM CHI'90 Conf. (Seattle, WA, 1–5 April), 249–256
- [30] Jakob Nielsen (January 1, 1995). 10 Usability Heuristics for User Interface Design.