Software Engineering Group Project
COMP2043. GRP
Final Group Report

Vision - based Analysis System for Football Match

GROUP.14

Supervisor: Heng YU

Chen LIAO / scycl2

Bolin QU / scybq1

Minghao ZHOU / scymz1

Zihui LUO / scyzl3

Xinyi OUYANG / scyyo1

Shihong HUANG / zy18708

Date: 2020/4/22

Abstract

At present, there are mature products developed for analyzing sports match based on videos, such as Viz Libero. With the rapid development of mobile electronic devices, however, there is still a lack in the field of analyzing football videos on the smart phone. In this way, after a certain thinking and improvement, an applet based on WeChat application and can be used on the smartphones to analyze football videos is required to be developed. Under the guidance of Behavior-Driven Development method, some black-box technologies are reconstructed and combined. Full-Convolutional Siamese Networks is selected to trace each character and Siammask function of computer vision is used to extract information from a scene. As for training of image classifier and counting for statistics, classic LeNet-5 is referred to. The applet combines advantages of methods, quickly and accurately analyzes football anytime and anywhere, posts the obvious and readable results on the screen, maximizes the efficiency to save time cost of user and improve user experience.

Content

1. Introduction & Problem Description

Smart phones, as one of the representatives of mobile devices, are developing toward convenience and intelligence. With the improvement of performance, they gradually enable more resource consuming and environment demanding tasks to be reimplemented on their own platforms theoretically, including ones with machine learning and computer vision. However, relating projects is scarce after a literature review, especially for domestic platforms such as WeChat. Therefore, to explore this potential, a football analysis applet is considered to be transplanted on such kind of platforms. The project, named 'vision-based analysis system for football match', mainly focuses on tracking a certain player in a given video and count for statistical data. When a user provides a piece of video and the chosen player, the program should generate a visualized feedback including charts and graphs of various data for this player regarding to the match. This actually arouses various problems for this software: how to interact with users for object tracking but based on a mobile context, how to actually implement tracking and merge it with recognition part and how to enable communications between two environments that are not fully compatible? These are the crucial issues for implementation and would be elaborated in this report.

2. Background Information & Research

2.1 Information About Framework

To standardize the development of WeChat Mini Program, Compony Tencent provides developers a platform named WeChat-developer tool, where programmers can do the design of their projects. Consequently, there are some certain differences in the framework between WeChat Mini Program development and general web development. To benefit programmers, the architecture design in WeChat Mini Program development absorbs some advantages of web development and at the same time discards the disadvantages of it. Several special examples of the framework design of Mini Program are as follows.

1. Different from web development, the design of Mini Program is multithreading, which means view rendering and business logic are

running in different threads. The view layer is rendering in Webview which corresponds to one page and the business logic layer runs in a same JSCore thread.
2. There are three main ways of page rendering: web rendering, native rendering and Hybrid rendering, the combination of first two. The UI is mainly rendered by mature Web technology, supplemented by a large number of APIs to provide abundant native client capabilities.
3. In regard to web rendering, Mini Program cannot be rendered by html labels but use Exparser frame, which is built into the basic library of Mini Program to provide basic support for various components of the applet and interestingly, the naming convention of the components all starts with *wx-*.
4. For control and security, Mini Program provides a JavaScript sandbox environment to run JavaScript codes, and the codes cannot access any browser-related interfaces which means JavaScript cannot operate DOM (Document Object Model) and BOM (Browser Object Model), otherwise an error will be reported.

When building the framework, as one of the programing languages applied around the world, network architect around the world have the requirement to build web framework with Python. With the purpose of developing the web framework without creating the components at the bottom level, Django was created to fulfill the requirement. By providing an integrated web architecture, Django not only sets out infrastructure which helps reduce workload of programmers to developing clean, maintainable code (Holovaty, Kaplan-Moss, & Legassick, 2009), but also utilizes MVC pattern making clear classifications of work between UI and backend. Besides, the security in Django is another reason to select the web framework. The Django template provides cross site scripting protection, cross site request forgery protection and SQL injection. Such protection could prevent data leakage and server crash caused by malicious attack on the server at a certain level.

HTTP (Hyper Text Transfer Protocol), as the protocol with highest amount of implementation, is the transfer protocol for World Wide Web to communicate by hypertext with local server (Tanenbaum, Andrew S., and D. Wetherall), which connects the user end and the server end. The HTTP request and HTTP response are applied as the methods for the whole project to communicate. The front end (user end) sends HTTP request to the back end (server end), with the back end completes the process, returns HTTP response to the front end.

BDD (behavior driven development) is selected as the development approach of the project, testing is considered as a part should be given a certain amount of attention. Apart from the tests for bottom level components, the communication between the front end and back end is considered (Sommerville). Last but not least, test for the whole system based on pre-designed scenario or not are both performed.

2.2 Information About Algorithms

It could be divided into two aspects when actually considering functionalities of the software: strategies and implementations. With the fine line of abstract and concrete, strategies defines the mechanism to solve a problem but performs by specific implementations. Since problems to be solved by this software are object tracking and image classification, various computer vision algorithms (***Figure 1***) and image nets are scrutinized on the strategy level.

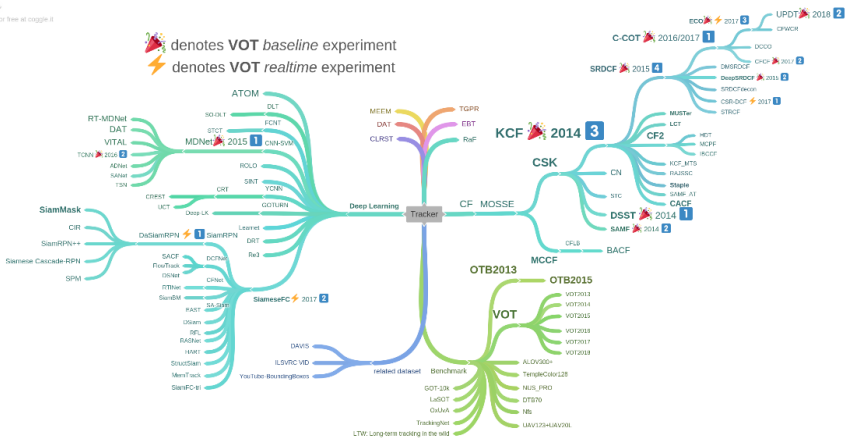


Figure 1 achievement distribution of visual tracking (CVPR, 2019)

During the thorough literature review, various tracking algorithms are scrutinized, including filter-based ones, such as MOSSE (Bolme et al, 2010) and KCF (Henriques et al, 2014), but mostly deep-learning based, such as SiameseRPN (Li et al, 2018). Finally, Siammask is selected among them (Wang et al, 2018). Though it has its own complexity comparing to other rudimentary ones, the algorithm delivers the best efficiency and performance under the same environment. This is because different from other algorithms, Siammask provides a binary judgement of whether this pixel of the image belongs to or does not belong to the tracking object that effectively precludes the influences form size changing, object proximity and environment variation (***Figure 2***). These issues are considered crucial

when analyzing a football video. LeNet-5 (Lecun et al, 1998) is a simply structured network for recognizing gray-scale hand-written digits since it only owns five layers, including two conv-layers, two max-pooling layers and a fully connected layer in the end (***Figure 3***). Because of its simplicity, the network is appropriate to be transplanted for other classifications with similar purposes.

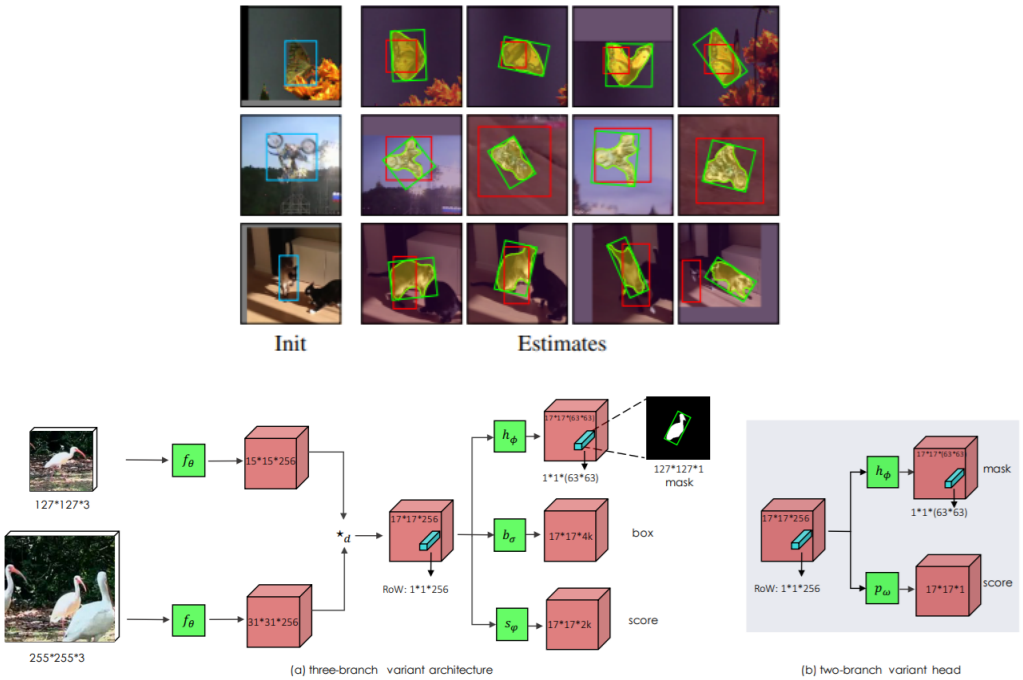


Figure 2 Siammask illustration and its structure (Wang et al, 2018)

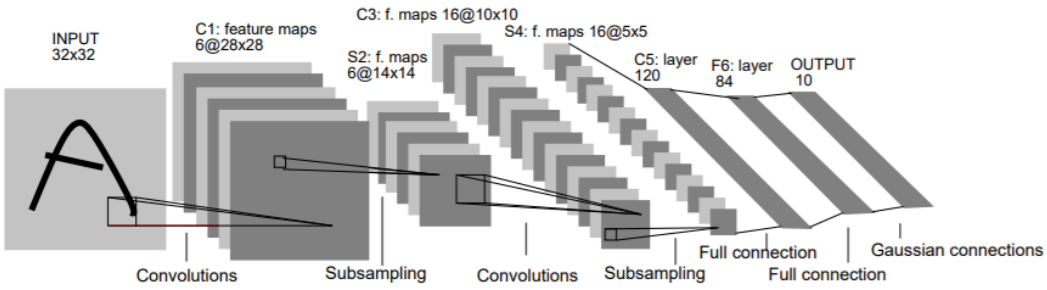


Figure 3 structure of LeNet-5 (Lecun et al, 1998)

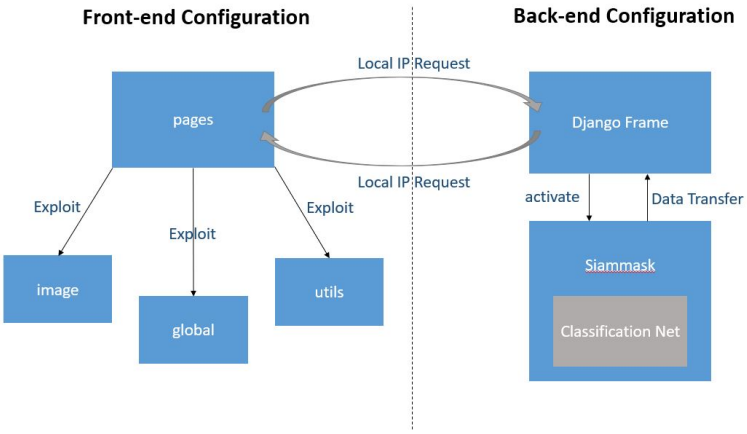
For implementations, both the two algorithms apply Pytorch (PyTorch 2020) and OpenCV (OpenCV 2020) as concrete tools. Pytorch is an open source python machine learning library based on Torch. It supplies substantiate support for machine learning with predefined building blocks and data structure, such as torch.functional for various loss functions and propagation strategies as well as torch.ToTensor for wrapping data to tensor format. These would be helpful when actually building and training the

networks. OpenCV is a multi-platform computer vision library with interfaces to various languages, including python. This library provides supports diverse operations of images, such as visualization, normalization and boundary detection which could further simplify the codes and could provide intuitional presentations of video manipulations.

3. Updated Designs

3.1 Big Picture

As mentioned, the software could be divided into three major parts: front-end interface, Django frame and algorithms. UI allows intuitive user interactions with the system and also convey requests of them to the back-end by internet protocol requests (IP Resquest). Data feedbacks are also visualized in an intuitive manner for users through the UI. As the back-end receptor, Django frame handles the received request and manipulate other back-end components respectively. It should be regareded as a layer between the UI and the real algorithm part which enables communications. Finally, for algorithms, the classification net is contained within the Siammask as a involved model which is pretrained and validated elsewhere. They are the components that actually provide problem solving capacities in this software. The results from algorithms will be transferred back to the frame and the frame post feedbacks to the front end which visualizes the results, a full working cycle is thus completed (***Figure 4***).



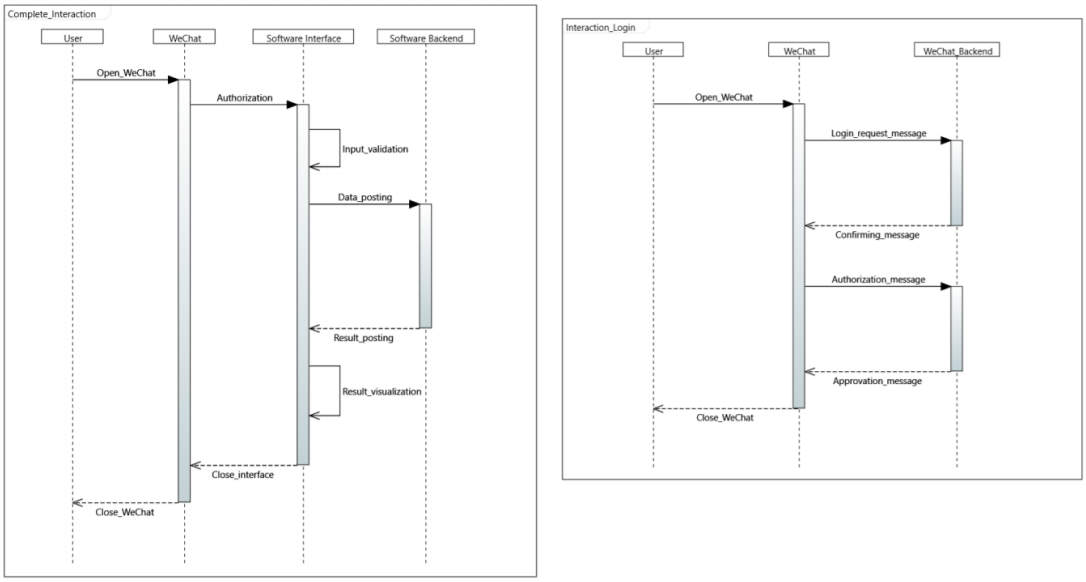


Figure 4 System configuration and sequence diagrams

From the perspective of various users, functionality details could be concealed on different levels. A general user simply uses the exposed functions to analyze videos whereas an administrator could manipulate backend directly including data management and performance update (detail functions in ***Figure 5***).

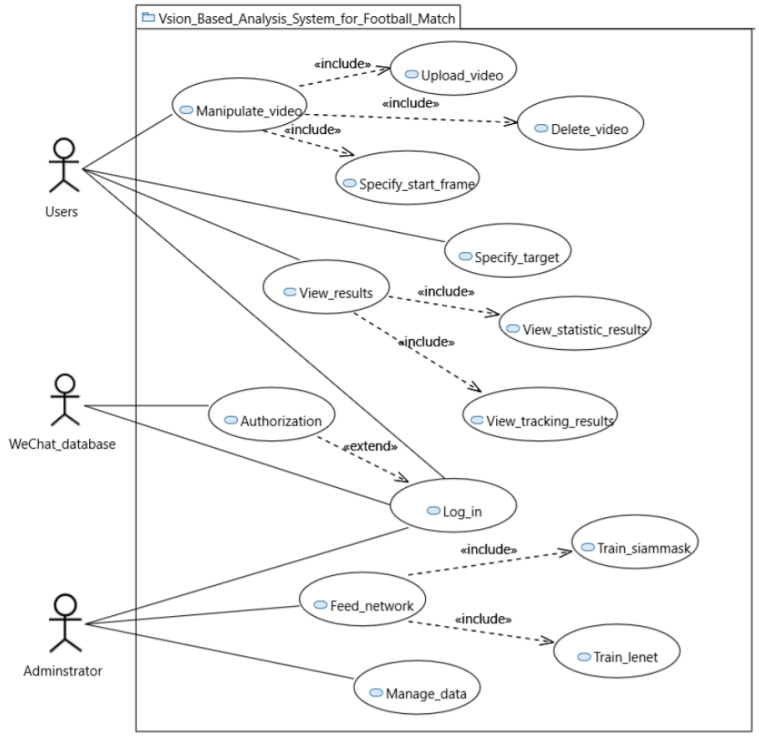


Figure 5 use case diagram

3.2 UI Design

After comprehensive consideration of user requirements and actual situation, the updates of UI are showed as follows. The main functions are divided into three parts: uploading new videos, getting the analyzing result and check the personal information. Therefore, in each page of this WeChat Mini Program, as which has been designed before, there are three buttons at bottom that allow users to use. The blendent changes to light blue and white instead of orange and black to comfort users and achieve a better user experience. The screen shots below are all provided by a real user.

On the "Upload" page, before uploading the video, users can click the "HELP" button to get the correct operators to upload the video and enclose the tracking target *(**Figure 6**)*. The "Upload" button allows users to choose a football match video from his video album *(**Figure 7**)*, before which, the program will ask for the authorization to open the user's photo album. After choosing the video, the transparent rectangle on the top left corner of the video will allow users to adjust its size and move it to enclose the tracking target *(*the first picture on ***Figure 7**)*. The algorithms will start when users click the "Start Tracking" button, and there will be a loading symbol to reminder users to wait for a little time. After finishing the algorithms, the text on button will change to "Succeed" and the original video will be replaced by a new video in which the tracking target are enclosed in real time (***Figure8***).

On the "Result" page, users can click the "obtain predicted result" button to receive the analyzing result from back-end algorithm and when the button change to green and show "receive successfully", user can click the "show the result" button and the result will be shown as four visualization charts: bar chart, line chart, pie chart and ring chart (***Figure 9***).

The "Me" page is shown as ***Figure 10**,* that allow users to check their own information. At the same page, the information of the development team and the version control of the applet will also be shown.

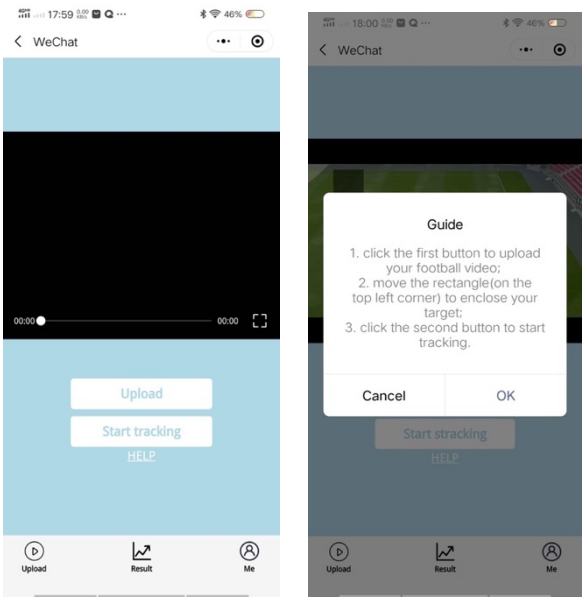


Figure 6 Upload – button:Help

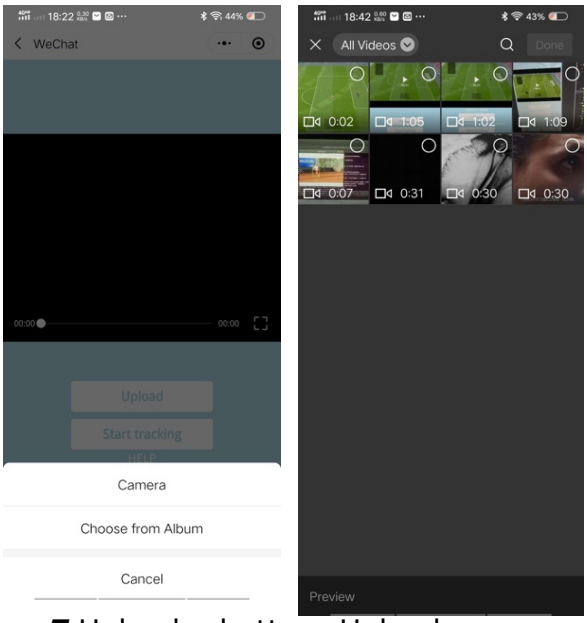


Figure 7 Upload – button: Upload

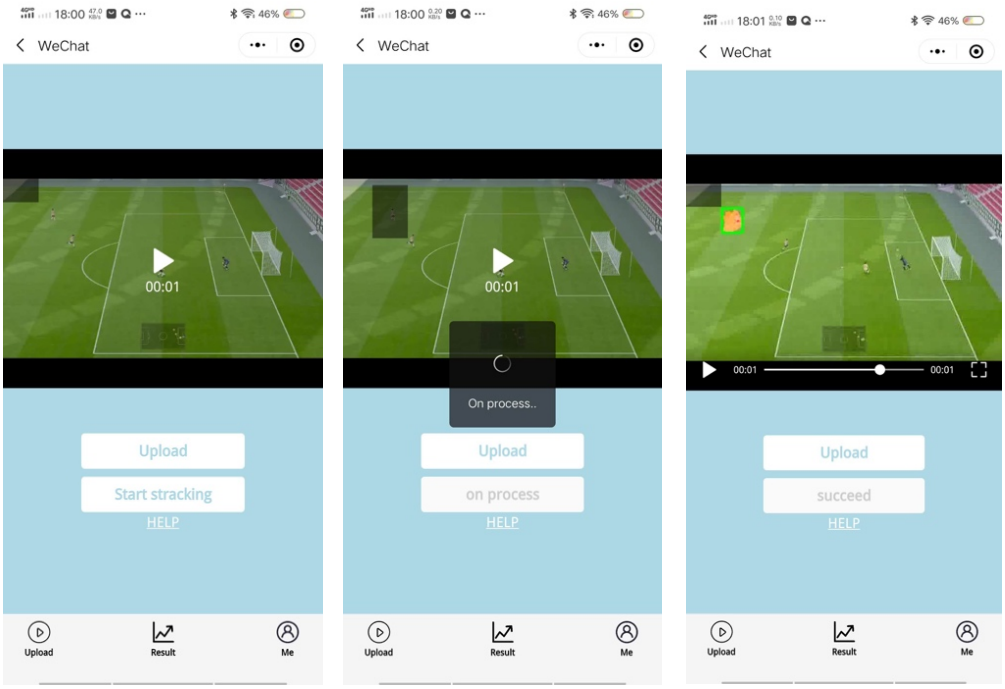


Figure 8 Upload – button: Start tracking

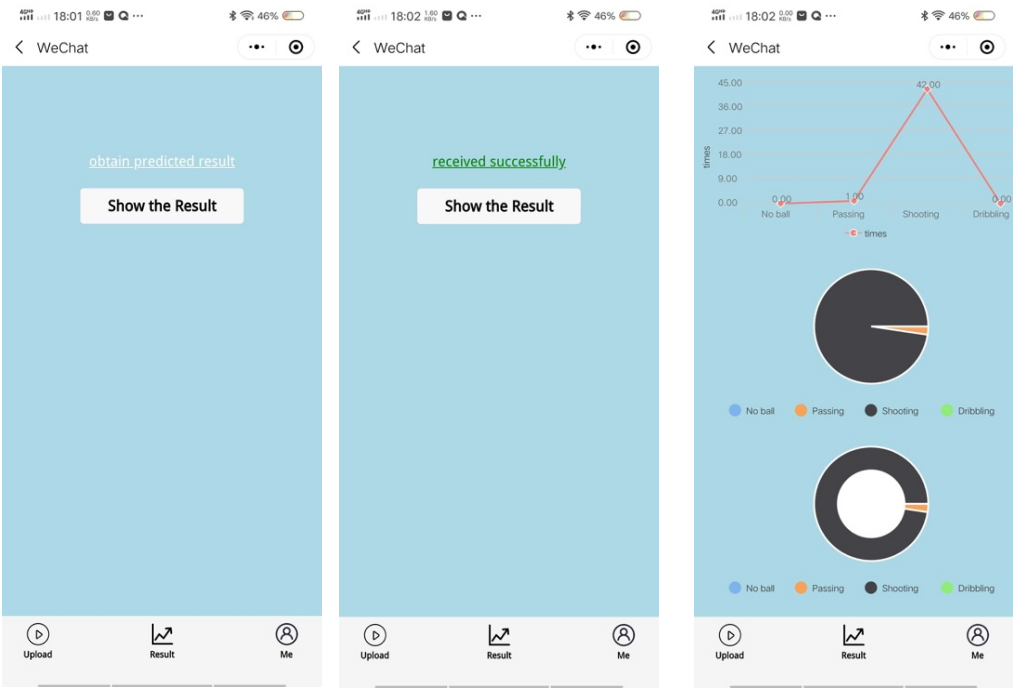


Figure 9 Result – Show the result

Figure 10 Me – show the information of user and developers

3.3 Frame

The basic frame design of the project has not been completed until February. The frame designed before was using JavaScript to send the file

to the core algorithm created by Python, when the core algorithm completes the calculation, the result will be sent to the front end and exhibited. The incomplete design has a defect, which is not taking the different working mechanisms of JavaScript and Python into consideration. JavaScript sends data by HTTP (Hyper Text Transfer Protocol) requests and receives data by HTTP responses. However, the core algorithm is a pure Python project, which is unable to process HTTP request or send HTTP response without processing. To solve the defect, the back end was changed from a pure Python project to a Django project. The reasons to select the Django are based on two aspects. First, Django provides function associated with content administration, which assists the project by processing different types of HTTP requests and sending HTTP responses. In addition, the calculating functionality required by Django is at a low level. For the core algorithm requires a certain level of the processing ability, less resource occupied by the frame could causes positive effect to the project. The updated design of the frame is concluded as Follow. Using Django project to receive HTTP request sent by JavaScript and modify the request to be readable data for the core algorithm. Then send the data to the core algorithm and start processing the data. When the core algorithm returns a result, turn the result into a HTTP response and send the result to the front end.

3.4 Algorithm

Two major parts could be divided when implementing core functions of this software. One is the object-tracking, the other is the image classification. Actually, during a whole functional cycle, the tracking algorithm keeps tracing positions of the target object by bounding boxes and saving contents in the boxes. When all the contents are saved with a fixed size and a fixed format, they will be classified by a convolutional neuron network into four clusters. Finally, statistics are derived by calculation to provide the data visualization its magnitude support.

When considering specific tracking algorithm, Siammask is referred to. This is an open-source object tracking algorithm based on deep learning (convolutional neuron network) which is published by Qiang Wang on 2019. For classification algorithm, the classic LeNet-5 network is reimplemented and optimized by the group. This algorithm contains, including its supplementary codes, four segments: training dataset, network architecture, training and validation as well as the model itself. Categorizing by utilization, only the well-trained model will be contained in the back end as a functional part to provide classification predictions, the other parts are auxiliary for supplying its capacity of classifying.

4. Implementation Details

4.1 Key Implementation Decisions

The key decisions are generally about operating system, running environment, programming language, framework, algorithms and hardware. Based on the limitation of WeChat, the WeChat-developer tool is selected to implement UI of this project, and in order to achieve visual tracking and to apply AI into project, python language should be a great choice. Linux based operating systems are considered because of the python language. Due to the need on the support of OpenCV and TensorFlow, Ubuntu 18.04 is chosen as the basic platform and a Conda environment was chosen as the environment because Conda could provide the virtual integrated simulator the prototype requires. For the tracking algorithm, Siammask is a wonderful choice because of its higher efficiency and user friendly. Although object tracking algorithm performs better when device chosen as sever are equipped with higher configuration, we still use ordinary computer (GPU: Intel Coporation HD Graphics 620, cpu: i7-7500U) as development environment due to difficulties on installing Ubuntu in PC with NVIDIA cpu.

4.2 UI

The UI part is totally written by our group and implement on WeChat-developer tool, a platform to develop a WeChat Mini Program. It is kind like general web development, since programmers use WXML to complete the page design, use WXSS to change the style and use JavaScript to implement the inside logic. And the encoding rules of WXML and WXSS resemble regular web development. But different from general web developing, there are some restrictions when developing a program on this platform. For instance, the file size of the pages that reaches the user device should under 5MB. Because of those restrictions, the implementation of UI of this applet is based on some principles, and those design standardization can achieve better human-computer interaction which can increase users' satisfaction with the product(Fu, 2010) ：

1. The function of the program is succinct and clear;
2. Make the operation of user distinct and make the interaction within the system simple and convenient;
3. Use simple page logic as far as possible to implement the basic function.
4. Suitable for the platform.

According to user requirements, the program should allow users uploading their own football videos and setting the tracking target before doing the algorithms, and after the algorithm, the visualization of the analyze result should be appeared to users. To meet those functions and based on those principles, the UI design can be implemented by some useful APIs provided by WeChat platform, which makes developing more convenient. For example, when intending to upload the video to the server, the wx.uploadFile interface can be used to allow developer upload the video by url. But it gets a little harder when implementing the visualization of the result because the original WeChat-developer tool does not include some suitable APIs to generate various kinds of charts. Getting through some research, a plug-in named *wxcharts.js* has been found to download and import into the project. By using this plug-in, programmer can use *wxCharts* function to draw the charts on the canvas such as line chart, pie chart, ring chart and so on.

Since the convenience provided by the platform, the implementation of UI seems straightforward and uncomplicated. But in practice, there still come some obstacles when considering the corresponding of the built-in APIs and the back-end algorithms, and considering of the conformity is a quite necessary part in the UI design of a software engineering project. For example, in our project, the back-end algorithm called SiamMask can analyze the exact target in a rectangle during a video, which means that if we want to implement the back-end algorithm, the UI should provide a rectangle to allow users to enclose their target and upload the target to the back-end. It seems a huge challenge to send the image of the target to the back-end algorithms because there is no correlative interface can be used. After deeply thinking and discussing, the task of processing the image is handed over to back-end algorithm, and the front-end is responsible for transmitting the coordinates and the size of the target instead of transmitting the whole image. Following the algorithm determined, the relevant built-in interfaces, such as *wx. createSelectorQuery*, can be used to implement the algorithm. And eventually, the built-in API can correspond to the back-end algorithms.

4.3 Frame

Based on the fact WeChat and Django have the specific IDE (Image Develop Environment) of each, the development of the web frame applied two software, which are WeChat-Developer tool and PyCharm. The WeChat Developer tool enable developer to design user interface and send http

request, while the PyCharm provides the environment for the Django project to run.

As the Django frame is designed for web development, the main purpose of the implementation is to process HTTP requests, invoke the core algorithm and return HTTP response when calculation results are demanded by the front pages.

Two parts are modified as the first step to implement the framework. Several paths are added to the framework to invoke the corresponding method. The path to send request and the path of receive response of the front end are modified to be the same as the paths set in the framework. The request type of upload the video or the coronates of the target player is “post”, while the request type of present the results implements “get”. The HTTP requests to upload video and coordinates is contained in JavaScript interfaces provided by WeChat-Developer tool.

As the WeChat applet send the video files by the “post” type of HTTP request, a function in the framework is created to receive this kind of HTTP request and save the attached file. When an HTTP request invokes the function, the function will check the type of the HTTP request is “post” or not. As the type of the uploaded file is limited to video, the framework will save the file and transform the video into the form which is acceptable by the core algorithm. For the coordinates of the target player is sent by HTTP request, another function associated with a specific path is created. If there is the “post” request to upload the coordinates of the target player, such data will be sent to the core algorithm to start tracking multiple types of actions performed by the player. If the tracking is completed, user could browse the results by column and line chart or by pie chart. The charts will be exhibited to user if a “get” request is sent to the back end then the results will be post to the front end by HTTP response as a series of numbers. The numbers are gathered for data visualization.

A method named “index” (***Figure 11***) was designed to receive the “post” request to save the video and send the coordinates of the tracking area to the script to start the calculation. To transfer the result to the front end, a method named “result” (***Figure 12***) was created to read results from a file, then post the results to the front end by HTTP response. Each method has a specific path for the front end to invoke, in addition, the front end is designed to invoke to corresponding method by the pre-set path.

```
@csrf_exempt
def index(request):
    #this is the function called when a video is received from the front end
    if request.method == 'POST':
        delete_path = "../../SiamMask/Video/"
        del_list = os.listdir(delete_path)
        for f in del_list:
            file_path = os.path.join(delete_path, f)
            if os.path.isfile(file_path) :
                os.remove(file_path)
            elif os.path.isdir(file_path) :
                shutil.rmtree(file_path)
        forms = UploadFileForm(request.POST, request.FILES)
        print('received!')
        if forms.is_valid():
            print("a " + request.method + " request occurs!")
            handle_uploaded_file(request.FILES['file'], filename=request.FILES.get('file'))
            return HttpResponse('file for upload ok')

    #this is the function called when the front end requires data to draw the graphs
    if request.method == 'GET':
        #call the start function here, set res equals to the return result
        top = request.GET.get("top", default='0')
        left = request.GET.get("left", default='0')
        height = request.GET.get("height", default='0')
        width = request.GET.get("width", default='0')
        print(left + " " + top + " " + width + " " + height + "aaaaaaaaaaaaaaaaaaaaaa")
        os.system("bash ../../execute.sh " + left + " " + top + " " + width + " " + height)
        BASE_DIR = os.path.dirname(os.path.abspath(__file__))
        result_file = "../../SiamMask/Video/processed/processed.mp4"
        if not  os.path.exists(result_file):
            #return HttpResponse(content="file not exist!", status=200)
            return HttpResponse(content=result_file, status=200)
        else:
            return HttpResponse("new video generated")
    else:
        print("a " + request.method + " request occurs!")
        return HttpResponse('Wrong Method!')
    return HttpResponse('file for upload ok')
```

Figure 11 Function index

```
def result(request):
    a, b, c, d = 0, 0, 0, 0
    file=open("../../SiamMask/Video/processed/result.txt", "r")
    content=file.read()
    for i in content.split(" "):
        if i == '0':
            a=a+1
        elif i == '1':
            b=b+1
        elif i == '2':
            c=c+1
        elif i == '3':
            d=d+1
    file.close
    return HttpResponse(str(a) + "," + str(b) + "," + str(c) + "," + str(d))
```

Figure 12 Function result

4.4 Algorithm

Siammask

As mentioned in previous sections, the source code is partially referred. However, code analysis and refactoring are conducted to make the frame and the algorithm compatible in order to transplant its functions to this software. The main modifications exist in the file demo.py in the source code. Since the purpose of adopting this algorithm is to gather target information (images specifically to the target) as the data for further classification, the contents in the bounding box has to be captured and connected to the classifier. Therefore, we managed to save the target zone

within the tracking result of every frame (***Figure 13***). The video is represented with a list of matrices in data structure, target matrices are cut from original matrices with coordinate calculation. After regulated to a uniform size of 50*50, sub-matrices representing target areas are thus acquired and saved under predefined relative path with the format of .jpg. Subsequently, corresponding target images are also represented as list of matrices and get prepared for classification. The classifier (or model), which will be elucidated in the later sections, are pretrained and validated. The model itself and the generated parameters are saved in net.kpl as a model file which is able to be included in demo.py. Therefore, to get prediction results, every matrix is designed to be fed into the model and the outcome is collected. However, the format of elements is Numpy Array in the list but only format of Tensor is accepted by the model. Thus a transformation is exerted through adding one dimension to the original element with torch.unsqueeze() (***Figure 14***). Then the reversed procedure is applied for transforming the predicting results back to the Numpy element. Finally, by coalescing results, the prediction array with integers ranging from 0 to 3 is generated.

```
# define target saving path
save_path = '../resultSet/'+str(f)+'.jpg'
# openCV.getTarget(im, target, save_path)
x1, y1, w1, h1 = location
points = [[x1, y1], [(x1+w1), y1], [(x1+w1), (y1+h1)], [x1, (y1+h1)]] #target position
target = im[y1:(y1+h1), x1:(x1+w1), :] #derive target image from the frame
resized = cv2.resize(target, (50, 50)) #data preprocessing for a uniform size
cv2.imwrite(save_path, resized) #save data
```

Figure 13

```
# prediction
result = []
transform = torchvision.transforms.ToTensor()

for counter, frame in enumerate(data):
    tensor = torch.unsqueeze(transform(frame), dim = 1)
    predict = lenet(ts) # predict for every frame of the video
    prediction_y = torch.max(predict, 1)[1].data.squeeze()
    result.append(prediction_y) # add result to the result set
```

Figure 14

4.5 Model Training

Classification net

For the artificial classification network, the classic LeNet-5 is referred to,

because this network shares a similar orientation with the network that needed. This means that they own the same classification purpose, result pattern and domain of use. For example, the result of LeNet-5 should be an integer lies in the range from 0 to 9, as an indicator of ten hand-write digits. Similarly, the network in this software should output an integer in range 0 to 3 for four kinds of movement. However, there are still differences between these networks. Most obviously, when identifying a hand-write digit, due to the simplicity of the images and apparent features, it is acceptable of training data with a relatively low resolution. However, in this project, the effective features are harder to be identified and the environment is more complex, thus some adjustments are necessary.

Network structure

The network structure in this software is similar to the LeNet-5, both with five visible layers hierarchically including two convolutional layers, two max pooling layers and a fully connected layer (***Figure 15***). Layers are organized alternately as one pooling layer follows one conv layer and a fully connected layer as the output layer at the end of the network. For the size of input in each layer, as mentioned before, it is more difficult to extract features and identify nuances from the input images, the size of input images is adjusted to 50*50 comparing to 32*32 of the original LeNet-5 in order to keep more details (Sinha et al, 2017). Correspondingly, all relating scales are modified to optimize the performance of this network.

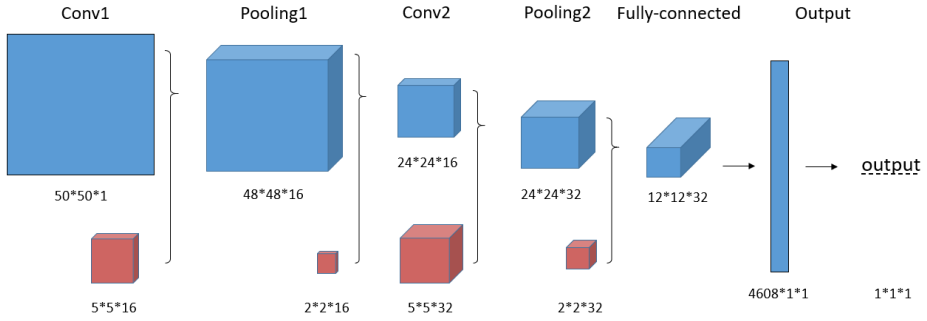


Figure 15: network structure

When defining parameters within each specific convolutional calculation, four of them are scrutinized, including channels, kernels, strides and paddings. Generally, the number of input and output channels refer to the number of different feature categories to be extracted which is determined by the number of kernels within a single convolutional operation. Kernels here are various minuscule matrices with a predefined size performing as extractors orienting diverse features. Images convolute with kernels and generate new matrices which underlines different properties (Sinha et al,

2017). Strides and paddings control the size of output matrices. A stride determines the operation density within each convolution whereas paddings supplement an extra frame surrounding the output matrix. The effect of strides and paddings to the results could be measured quantitively as follows (as ***Eq 1*** where W_{new} denotes ouput size, W denotes input size, F for kernel size, P for padding and S for stride). Considering the concrete problem to be solved, parameters are set into suitable values and finally determined the specific structure of the network in this project.

$$W_{new} = \frac{W - F + 2 * P}{S} + 1$$

Eq 1

Dataset organization

Abundant researches are done attempting to find appropriate open-source football image dataset regarding to movement classification. However, the related resource is limited. Therefore, a dataset is built by the group for training the network by imitating the digit recognition dataset (Chowdhury et al, 2016). For a basic dataset both training data and validation data are required, thus two separate folders are created when designing the dataset folder. Additionally, as mentioned above, there are mainly four gestures designed to be recognized: no ball, passing, shooting and dribbling, labeled as 0, 1, 2, 3 correspondingly. According to these categories, four correctly labeled folders are created for both training folder and validation folder.

For the content in every folder, as initially designed, totally 4000 images, 1000 of each category, with a size of 50*50 would be sufficient. In these pictures, validation data is separated from training data randomly following a ratio of 4 to 1. Finally, the overall frame of the dataset forms as 800 images for each category in training data, 200 images each for validation. To derive images, football video games and some functions of Siammask are exploited. As mentioned in the former sections, Siammask could track an object in a piece of video and is modified in this software to save the image contents of the tracking object, we thus run the algorithm on several pieces of football videos generated by video games and collect the generated images. Subsequently, by selecting impactful images and transformations on images such as rotating, squeezing and mirroring to improve robustness, the content of the dataset is produced (***Figure 16***).

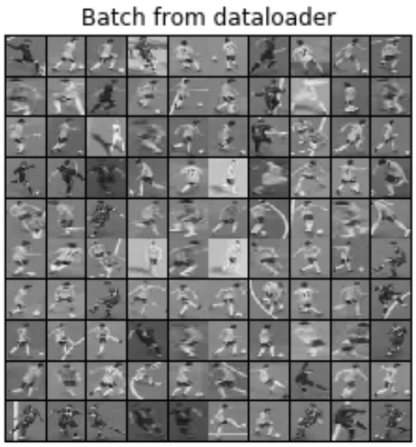


Figure 16

Parameters

There are various hyper parameters to be specified during the actual training process. Considering the scale of the training dataset, it is necessary to learn for several times to guarantee the learning outcome, which means to train the network with current data iteratively. The parameter epoch defines the number of iterations within the training process. By comparing the network performance under different epoch values (***Figure 17***), it is apparent that accuracy increases with the number of iterations. However, when scrutinizing the network performance, overfitting occurs when the training continues for about 11 to 13 iterations. Therefore, a mid-value 10 is defined as the training epoch. Additionally, to compromise to hard-ware capabilities, batched training is utilized and various batch sizes are evaluated (***Figure 18***). After comparison, obvious fluctuations of the loss value occur that it stagnates rather smoothly decreases when the batch size is small (around 20). This is because the differences of accuracy on different categories are significant, a small batch size would cause the model to converge rapidly, resulting in an unbalanced model that could have different loss value when testing on different categories. Therefore, a relatively large batch size of 100 is chosen without severely sacrificing training efficiency. Learning rate, as a coefficient in back propagation, controls the learning capacity of the model, in another world, controls how fast does the model learn within every training step. Initially, exponential descending learning rate which decreases exponentially as the training goes on is the first priority. However, when actually experimenting, though the accuracy can increase statically, there is a severe fluctuation on the loss value and the final asymptotic accuracy is not in satisfactory comparing to normal constant learning rate (***Figure 19***). Therefore, a constant learning rate of 0.001 is set for model training.

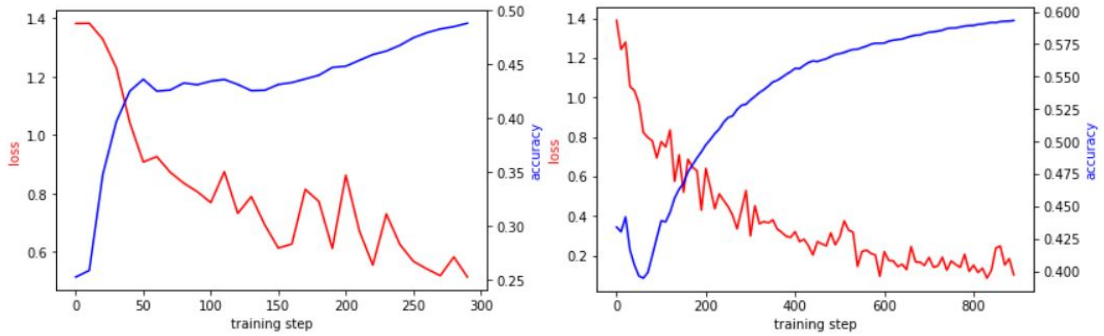


Figure 17 loss and accuracy under different epoch values

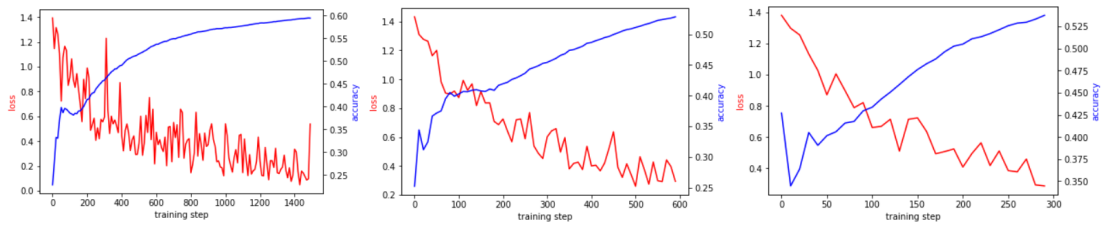


Figure 18 loss and accuracy under different batch sizes

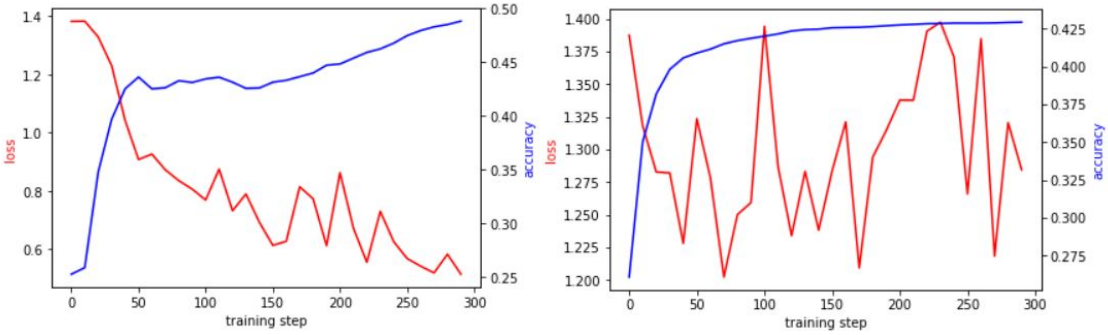


Figure 19 comparison between constant LR and exponential descent LR

4.6 System Integration

Since backend of the system constructs based on Linux environment, integration is used by shell script. For instance, when calling for function of tracking by SiamMask, particular route should be incorporated into base path and the corresponding conda environment needs to be activated. These requirements are usually achieved by scripts.

Moreover, adjustments of data are necessary. Usually the format of data is inconsistent between front and end. For example, the whole video should be separated into a list of pictures of each frame which is the format required by the neural network. When the tracking process is done, the list of pictures needs to be integrated into a new video. The whole process is achieved by the tool named ffmpeg, which is also handled by shell scripts.

Moreover, unprocessed parameters received from frontend are also futile. At first, the parameters representing position and size of selection box user has drawn is only relative to the video window inside interface. In order to convert parameters into data representing position with respect to the videos, we use scripts to scale original data into appropriate one.

Shell script:

Shell script is used for combining components backend and process into suitable format. When Django receive request, it will execute corresponding shell script. Then, the scripts execute program and adjust result returned from program into appropriate format before sending back to sever. When receiving videos, script decomposes videos using FFMPEG. When receiving tracking request, scripts activate SiamMask and Lenet respectively, then use FFMPEG to convert result of images to processed videos, calculating frequencies of each motions predicted by Lenet and finally collect and gather all result into one folder.

FFMPEG:

FFMPEG is one tools used for converting one video into list of images representing each frame or accomplish the reverse things. When receiving the videos from interface, FFMPEG decompose videos to format readable by neural network. When the tracking process has accomplished, FFMPEG would receive list of images processed by SiamMask and combine the whole set into a new video.

4.7 Version Control

In order to ensure that everyone in the group can clearly understand the progress of the project, view the front-end and back-end codes online anytime and can conduct structural research and modification on the basis of it, so that the cooperation between the team members is more convenient, based on git the version control method is used.

A new project is created. Team members are invited. And their identities are set as maintainers. Use “git clone” instruction followed by http URL to establish a connection between local files and remote websites. Every time members update the code version, use “git add” instruction to add new and changed Files, meanwhile, the specific branch of the added file can be

selected: master branch, front-end branch and back-end branch. Use "git commit" instruction to add specific version information comments. Finally, use "git push" instruction to directly push the updated file library to the code repository. After several uploads, the code warehouse is as follows (***Figure 20***):

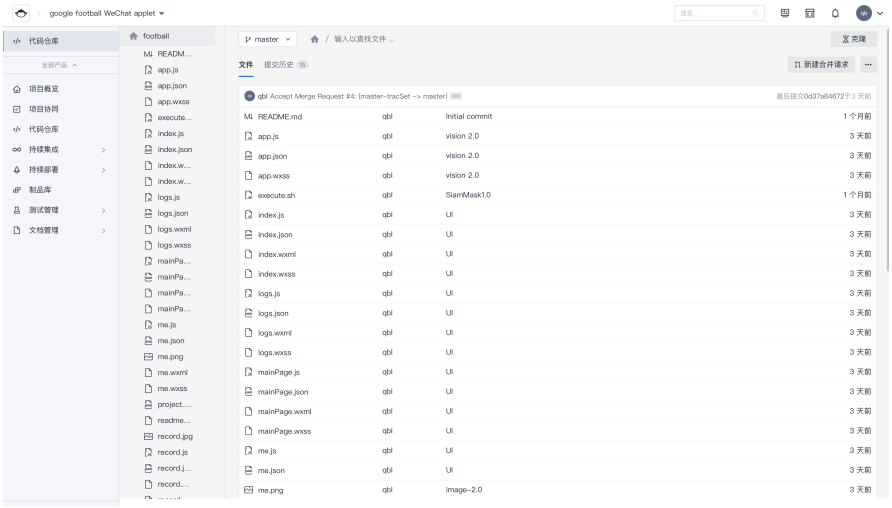


Figure 20

5. Testing

5.1 Overview

Before the development, the BDD (Behavior-Driven Development) method is confirmed. Compared with the other ways of testing, such as ATDD (Acceptance Test Driven Development) and TDD (Test-Driven Development), BDD method is more flexible and it can lead to higher test group participation. The members of test group are considered as users in the early stages of development and some expected features are provided. These features are constructed gradually as expected. During the process of development, the test group test the whole system to ensure completeness of the mini-program after every reconstruction. Therefore, BDD can be regarded as the combination of ATDD and TDD. It requires the test group play users and testers two different identities to take part in the whole developing process. In this way, BDD is the most suitable test method for this project. The whole process can be full of rhythm and always efficient by using this test method.

5.2 Attention in Stages

In the early stage of development, the BDD method require user to attend to provide specific suggestions and because the lack of user, test group play user roles to propose general requirements. Several features are putted forward by users (test group). As for data processing and algorithms: types of data displayed, the accuracy of all kinds of result data and total calculation time are required clearly. For the front-end interface and interaction design: the method of adjusting the frame by fingers and the specific interface are confirmed.

During the develop process, the BDD method requires test group to get formally involved in the test phase as testers. Just like standard in TDD method, after each reconstruction, test group is required to test the whole applet and record the result of each test. According to the record, more specific recommendations for specific aspects are proposed.

After each version of the mini program is launched, the front-end, interface and the back-end three aspects must be tested respectively. Due to the fact that integration and server domain name filling still exist problem, the front-end and back-end should be tested in different operating environments and integration testing cannot be accomplished in the middle of the project. In this way, the test work can be more complicated on different machines.

5.3 Test Plan

The system has three basic components: front-end page, back-end algorithm and the web frame. Each of the component has a specific testing method. In addition to the basic components, tests for the whole system were designed to evaluate the quality of the software. In addition to the test cases performed by developer, scenario test was designed by concerning the practicability of the software.

Test name	Input	Expected output	Actual output	Test result

Figure 21 Test case figure

Front end

As two parts are contained in the front end: WXML (WeChat Text Mark-up Language) documents and JS (JavaScript) code, the tests of the frond-end can be separated into two parts. First is checking if all the required components are exhibited as expected, second is interacting with the components to exam whether the pre-design events will be triggered.

1	Access the index page to check if all pre-set components are presented		All pre-set components are presented	Pass

Figure 22 An example of tests for display of components

2	Click the 'enter' button to jump to the main page		Jump to the main page	Pass

Figure 23 An example of tests for pre-set events

Back end

As the part perform calculation according to the core algorithm, tests of this part focus on if each unit of the core algorithm could provide the results as anticipated. To retrieve the result of the units, the code fragment is operated by function.

Data format test	A batch of processed training data	The shape of this batch in **tensor format** with the dimension of **epoch*1*50*50**	`torch.Size([100, 1, 50, 50])`	Pass

Figure 24 An example of the tests of a component of the core algorithm

Web Frame

For the web frame connect the front end to the back end, the tests is performed by click a button of the front page then exam the console of

the web frame for preset data or view the front page to check returned HTTP responses.

Send back analyzing result back to frontend	http request: "http://192.168.0.108:8000/ result"	http response containing array	[0 28 12 0]	pass

Figure 25 An example of tests for HTTP response

System test

The system test includes the whole software, with three basic steps contained. All the units associated with each step are tested.

1. login the applet though WeChat.
2. upload video in the local video album to the applet.
3. view the data retrieved from the video.

Scenario test

For practicability of the software, a scenario based on the requirements of the reality is designed. Both the functionality and user experience of the software are evaluated.

Eric is the coach of UNNC fooball team. He wants to analysis the football game videos and identify all area ability of his team member. First, he opens the WeChat app and logins football movement recognition and analysis applet with WeChat account. He clicks the "choose" button to choose one video and clicks the "upload" button to upload it from the system album to the applet. Then, he uses the frame to select a player in the video to track and clicks the "start tracking" button to start tracking. When the tracking accomplished, the final data will show on the "record" page and he clicks the "record" button to view the results. Finally, he clicks "upload" button to reupload another video and analysis again.

Figure 26 Introduction of the scenario

5.4 User Feedback

According to the research ethics checklist, the development of the project allows group members and supervisor only. As a result, the feedback from the supervisor is accepted as from user and stakeholder. However, based on the situation of COVID-19 and the low efficiency of setting environment

for the software to run, a remote demonstration is provided to the supervisor instead of operating the software.
With the steps shown below presented to the supervisor, the group receive positive feedback, together with requirement for improvement in user experience. The positive feedback is given based on each of the steps was succeed and the calculation was correct with acceptable efficiency. Supervisor required more system status to be exhibited. One of the most demand part is to post a message when video uploading is completed. In addition, the user experience could be refined at a certain level, such upgrade might be achieved by exhibiting more system status at each running stage.

1. → login·the·applet·though·WeChat.↵
2. → upload·video·in·the·local·video·album·to·the·applet.↵
3. → Select·a·player·in·the·video·to·retrieve·data↵
4. → view·the·data·retrieved·from·the·video.↵

Figure 27 Basic Steps of the video demonstration

6. Achievements

The application designed has achieved most of initially proposed system requirements (details for requirements are in appendixes).

First of all, the program satisfies initial requirement to be executed on WeChat platform. When opening the index page, the program recommends user to make authorization: the button “Access permission of basic info” ask user for permission of accessing to personal information and user account (whereas request for current location in requirement is eliminated because program does not need such information). This function achieves requirements on linkage between mini program and account. Due to the fact that the program is able to directly view video album as long as user has given WeChat permissions accessing to document of phone, requirement of requesting authorization for album is achieved automatically.

Secondly, after entering mini program, uploading football videos is achieved by clicking button “Upload”. Whereas the requirement of checking

whether video size suit for program is achieved partially due to the fact that setting limit (set parameter maxDuration = 60) on uploading files only works for iOS and fails to take effect on Android device due to unknown reason.

After user has chosen the video and pass the checking limits, system pass the test that video would be compressed by front end and uploaded into sever. Meanwhile, by relocation of bounding box inside video area and adjusting the size, system achieves requirement of selecting tracking target in the first frame of video.

Afterwards, the backend satisfies requirements of data preprocessing (mainly includes adjusting parameters of bounding-box to fit in size of videos and convert video into set of images). Then, the backend pass tests on tracking visual objects and analyzing motions of target at each frame, collecting statistics of predictions.

In the next step, the backend successfully sends collected results to frontend and interface would receive data soon after. Requirement of visualizing data is accomplished by presenting statistics with three different formats: histogram, line chat and pie chart. Meanwhile users are able to select type of visualization format to be presented.

Overall, the major functionalities of applications (including interaction between UI and backend, implementation of core algorithms) are realized. On the other hand, some requirements of constructing database for storing information of user and history files, demands for downloading predicting results and checking if contents of uploaded video relates to football remains to be incomplete.

7. Problems Encountered

7.1 Problems on Technical

When implementing designs on the project, there are plenty of unexpected problem having been encountered, especially on the framework.

As illustrated before, python language was chosen to implement the project. When constructing the framework, we found that though python is one of the most popular programming languages, the number of resources of framework oriented to WeChat applet is very limited. Therefore, due to the unfamiliarity with python language and the framework, large amount of efforts is spent upon learning the new language and trying web framework. Finally, after a long period of exercise, the Django framework was selected as the framework for the whole project. And some obstacles such as difficulties of streaming videos to frontend through Django have been overcame and some particular functionalities have been achieved.

Another problem on technical is integrating the project because of the different system between implementation UI and back-end algorithms. The platform, WeChat-developer tool, which implements the developing of UI, is generally based on Windows or MacOS system but the whole back-end code should be implemented on Linux since some algorithms like SiamMask can only be executed on Linux. the solution is that the front-end and back-end operate on separate computers but run on the same LAN. But, since the IP address of the computer in same LAN will constantly change, the operator should manually check the IP address and make adjustment every once in a while, which still remains to be a hard-coding problem.

There also come great difficulties when data processing in back-end algorithms. In most circumstances, football players are relatively small compared with the whole image, thus neural network might has great difficulties tracing target if video are not clear enough. This problem can be exaggerated when analyzing motions by Lenet. Since Lenet only process image with same size and image needs gray-scale transformation, this processed image has become quite vague and hard to be identified as input for Lenet. Thus, the consequence is that the high accuracy cannot be preserved in analyzing movement of players. This is the fly in the ointment of the implementation of this project.

7.2 Problems on Project Management

Project management is a series of management actions that are carried out with the progress of the project in order to ensure that the project can achieve the desired results, which is a significant tool in present business environment (E, 2011). The division of labor is an important part of project management. In our team, teammates are divided into three groups to take charge of different parts of this project, UI, algorithms and test. Each group

focused on what they were responsible for, and informal meetings were held every week to report the progress in each group and get the information exchange.

But there came some problems on account of the separation of the project. During the development, each group did well in their own parts, but the integration of all parts was not as simple as imagined and we miscalculated the time of its implementation. As the problem discussed before, though both UI and algorithms were completed, because of the different system, we tried several ways but the integration still came different problems and the project failed. The lack of a completed design caused negative emotion effect to the group members by anxiety. Such situation could be a result of excessive follow to the BDD. Most of test cases provided by the group members focusing on the components of the project, while a limited amount of attention was paid to the framework. Making the decision on the framework earlier could prevent the situation. Fortunately, we had reserved a lot of flexible time and after some researching, a solution was found to implement integrating the project. This episode allowed us to reflect deeply and understand the significance of integrality in a project.

7.3 Problems on Teamwork

The teamwork problem came out at the beginning of the project, because the teammates did not know each other very well in the early days of team formation. But the leader of our team was keenly aware of this problem and took corresponding measures in time. To make the atmosphere in our team more harmonious, the leader brought everyone together to discuss and work with the project frequently. At the very start of the project, we stayed together to understand the user requirement and discuss how can we implement the project. And after with the clear understand of what to do, as discussed before, our team was divided into three groups to take charge in different parts of the project. Leader set staged targets every week and held meetings to check the progress of the project every week. When some technical problem came out, we worked out together and the teamwork help us overcoming the limitations of single thinking.

Unpredictably, after the winter holidays, the new semester did not come as scheduled and we were asked to study at home. Online communication became the only method to communicate with each group member which did cause some problems and negative effect to the project. Without in-person communication, misunderstanding to parts of the code may not be

able to solve at an instance. Being remote to each other leads to a higher requirement to fit in the schedule of each other. Discussion could be performed during some pass-over situations in campus, however, remote working limits the possibility for instant solving to problems. But this was a problem that everyone faces, so we adjusted mentality and kept in closer touch with each other to know the progress of project clearly. Each time we updated the progress of the project, assigned new tasks, and kept the project moving steadily, and online meetings became more efficient.

8. Summary and Reflection

At the beginning of the project, as shown in ***Figure 28***, an original time plan was made as a Gantt Chart to have an overall control of the progress. As planned before, the three parts of the project, UI, algorithms and test, should start at the same time and the project should be finished within 23 weeks.

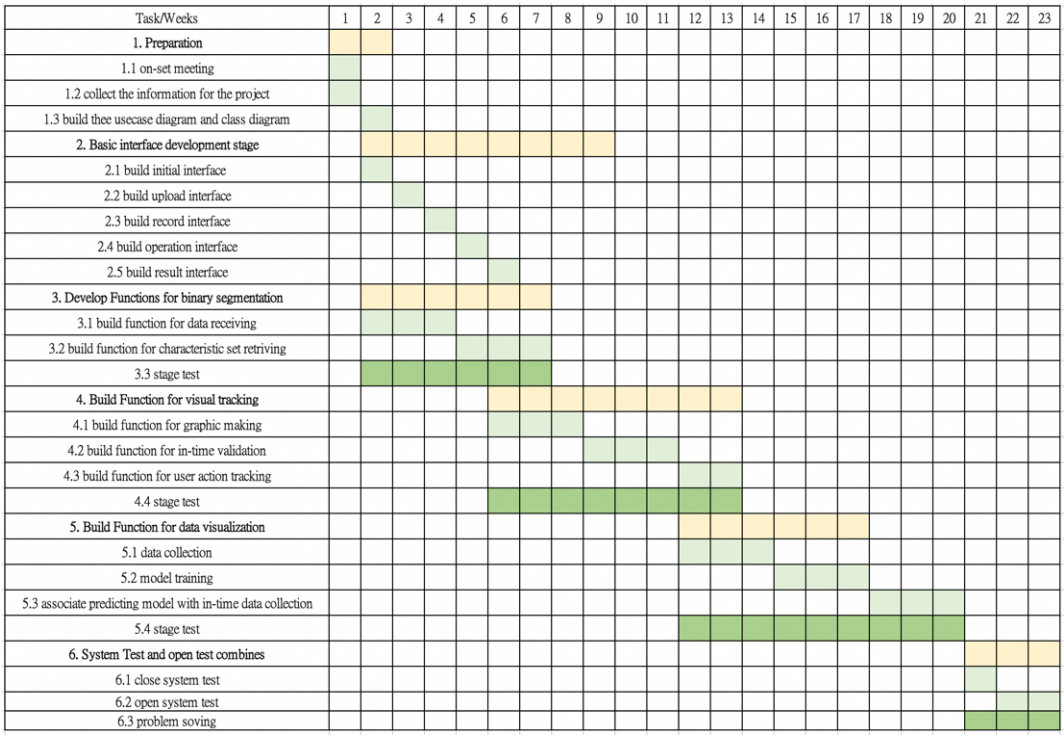


Figure 28 original time plan

Because of the selecting of BDD (Behavior-Driven Development) method, the actual project activities and schedule are generally similar to the original plan. After discussing on the requirements and researching the background

information, we had our initial design with some prototypes. When coding part starts, as the original plan, coding on UI and back-end algorithms started simultaneously. When we were not sure about how to do next, the plan helped us reflect and assign new tasks.

Though some similarity between original plan and actual schedule, when we actually implement the project, some unreasonable arrangements in the original plan still came out. The start of coding was too early and the period of it seemed quite long. When we intended to begin programming, it was found difficult because of the lack knowledge of the whole project. therefore, we adjusted the plan and decided to do more research on programming language and algorithms of both UI and back-end algorithms before actual coding. And after several weeks of knowledge reserves, the implementation of the program became clear and did not spend a long time as the original plan. Another unreasonable part on the original plan was that we did not leave time for project integration. And as illustrated before, when integrating the project, we met a number of difficulties and spent much time on it. Fortunately, due to the fast implementation on coding part, we had a flexible time to deal with the integration problems and make up for the shortcomings of the original plan. And because of those problem came out, we realized that the time plan of a project is not just about the coding and testing part, the preparing time and integration also need to be considered, which will be beneficial on the future software engineering project.

To sum up, though the actual schedule does not fully comply with the original plan, the vision-based analysis system for football match have been implemented. Users upload videos through front-end page interaction and adjust the frame to select target football player. The position information of target will be transmitted to backstage in coordinates and the algorithm observes it, calculates data and returns results to front-end rapidly. The main components of the system are UI, frame, algorithm and test. In the develop process, strategies and implementations are two functionalities of the software to be concerned. Under the guidance of the BDD test development method, UI and algorithm are completed according to the preset functions. Front-end and back-end interaction framework is modified and redesigned. At the same time, object-tracking and the image classification are considered in the core functions, the traditional technical get upgraded in this system. As for updated design, some specific requirements also be achieved at the end of project to optimize the user experience and improve software efficiency. Some time management methods are used, such as Gantt Chart and regular online meetings. Three

teams of Group 14 work hard on their job and overcome the difficulties together. In the end, the vision-based analysis applet on the WeChat application fills the market gap, makes the applet system more complete and has certain application prospects.

WORD COUNT: 8117

9. Reference List

A. M. S. Chowdhury and M. S. Rahman, 2016, "Towards optimal convolutional neural network parameters for bengali handwritten numerals recognition," *2016 19th International Conference on Computer and Information Technology (ICCIT)*, Dhaka, 2016, pp. 431-436.

Bolme, D. et al., 2010. Visual object tracking using adaptive correlation filters. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.2544–2550.

CVPR 2019, *Visual Tracking Paper List*, viewed 8 December 2019, <https://github.com/foolwood/benchmark_results>

Henriques, J.F. et al., 2015. High-Speed Tracking with Kernelized Correlation Filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3), pp.583–596.

Holovaty, A., & Kaplan-Moss, J. (2009). *The definitive guide to Django*: Web development done right. Apress.

Li, B et al., 2018, High Performance Visual Tracking with Siamese Region Proposal Network. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, 2018, pp. 8971-8980.

OpenCV 2020, *documentations*, viewed 9 February 2020, < https://docs.opencv.org>

PyTorch 2020, *documentations*, viewed 15 March 2020, < https://pytorch.org/docs/stable/index.html>

Kotogianni, E. *Project Management Maturity: A Case Study of Payne in Nottingham*(2011). Web.

Sommerville, Ian. Software Engineering. Tenth Ed, Global ed. Harlow, Essex, England: Pearson, 2016. Print.

Tanenbaum, Andrew S., and D. Wetherall. Computer Networks. 5th Ed, Pearson New International ed. Harlow: Pearson, 2013. Print. Pearson Custom Library.

T. Sinha, B. Verma and A. Haidar, "Optimization of convolutional neural network parameters for image classification," *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, Honolulu, HI, 2017, pp. 1-7.

Wang, Q. et al., 2018. Fast Online Object Tracking and Segmentation: A Unifying Approach.

Xiangqian Fu. "Mobile Phone UI Design Principles in the Design of Human-machine Interaction Design." *2010 IEEE 11th International Conference on Computer-Aided Industrial Design & Conceptual Design 1* 1 (2010): 697-701. Web.

Y. Lecun et al, "Gradient-based learning applied to document recognition," *in Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.

10. Appendices

10.1 Formal Meeting Minutes

GRP_TEAM14_1st Formal MEETING MINUTES			
Date & Time	2019.10.21 18:05–19:00	Position	PMB426
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **Briefing of the project: The project is related to AI and Neural Networks. There are seven videos of small-scaled restaurants from several different cameras.**
- **Detailed requirements of project: content identification, data collection, question solution.**

Possible Problems:

- **The face is occluded by plants or decorations.**
- **The screenshot pixels are blurred.**
- **The training set can be not meticulous enough.**
- **Is it necessary to recognize the people's identify based on their face?**
- **Can we recognize dishes by color?**
- **Recognize the number of customers, substances and timestamp.**
- **Can we train the machine on some specific training sets firstly, then put it on the videos from school?**

Additional Possible Functions:

- **Analysis the impact of climate on the number of purchases.**
- **Predict the probability of frequent guest having meals in restaurant in the second day.**

Note: the prediction is selectable, the primary task is recognizing videos accurately, based on accurate identification and then make predictions.

Primary Mission:

- **Clear the project and imagine the risk points inside.**
- **New ideas in the AI recognition process.**
- **Clear the work of the periphery.**

GRP_TEAM14_2nd Formal MEETING MINUTES			
Date & Time	2019.10.29 11:00–11:30	Position	PMB426
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **Reporting team division.**
  Front-end: Xinyi OUYANG
  Core Algorithm: Chen LIAO, Minghao ZHOU
  Development Test: Zihui LUO, Bolin QU
  Project Releasing test: Shihong HUANG
- **Make the decision (use the Google football video to replace the video of restaurant) together and submit the applications.**

Advice from supervisor:

- **In order to avoid the ethical issue (guests and waiters), members have thought many different methods, the supervisor purposed that project can change the videos, use virtual videos instead of real videos, Google football can provide some virtual players and standardized behavioral law, which is helpful for the accuracy of analysis.**

Note: It still need to submit applications.

Primary Mission:

- **Start to try to run the Google football on all kinds of operating systems.**
- **Do research according to the team division.**
- **Clear the work of the periphery.**

GRP_TEAM14_3rd Formal MEETING MINUTES			
Date & Time	2019.11.7 15:00–15:50	Position	PMB426
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **Brief reporting the research have been done: software using of the front-end, the process of core algorithm, expand with Unit test and test step by step for different classes.**
- **The access of AI lab room for each team member has been allowed.**

Possible Problems:

- **The function of wechat applet can be incomplete.**
- **In the process of development, the server might be needed.**

Primary Mission of Next Stage:

- **Read the papers from supervisor.**
- **Achieve the technical reserve in advance.**
- **Think the user requirements and the software requirements.**
- **Load the virtual machine and try to run Google football.**
- **Image the test cases of the project.**

Notes: The personal division of labor should be showed in the report. The overall plan of framework can be show in the report.

GRP_TEAM14_4th Formal MEETING MINUTES			
Date & Time	2019.11.15 11:00–11:50	Position	PMB426
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **Report the process of this stage: the proto graph (Specific interaction interface) has almost finished.**
- **Running the whole project on the Linux system will be perfect.**
- **Use TDD (Test-Driven Development) method to develop the whole applet.**
- **The content of paper provided by supervisor is a technique, which is aimed to track the specific object in a video.**
- **Brief report the finished requirements to supervisor.**
- **Working on the whole test cases, which includes front-end test cases, core algorithm test cases and release test cases.**

Possible Problems:

- **The core algorithm team mainly try Google Football on Linux system. But it still not works, the other operating systems, like MacOS or Windows, also not the right one, the mistake may be related to the VGA Driver.**
- **The requirements still need to be discussed with the test group to refine the details.**
  Notes: the process should be pushed.

Primary Mission:

- **Draw the UML diagram.**
- **Finish the user requirements and the system requirements.**

GRP_TEAM14_5th Formal MEETING MINUTES			
Date & Time	2019.12.2 11:00–11:50	Position	PMB426
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **Report the process of stage: the Google football can run on a specific computer now, if there are some running problems, core algorithm group can invite members of test group and front-end group to figure out problems together or e-mail to Google football official mailbox.**
- **Brief the specific division of labor of midterm report.**
- **Elaborate on what the report contains.**
- **The proto graph can be improved by using the software called Modao, after improving, people can directly implement interface interaction on the graph.**
- **The Google football can run on a laptop, the training sets can be collected by now.**
- **The algorithm can generate many kinds of videos, which include all information of whole game, like position of each player and the timestamps.**
- **Introduce the possible functions of the applet, according to the algorithm, the requirements should be changed.**
- **Midterm report mainly contains UML diagrams, software requirements, the final result of core algorithm, test cases and introduction of the tools.**
- **Supervisor is satisfied with progress of project.**

Primary Mission:

- **Finish the midterm report.**
- **Try to improve the report.**
- **carry out both technical ones and teamwork ones.**
- **choose critical ones to expand.**

Division of labor according to the detailed requirements:

Each team:

• Update and extend the description of the problem to be resolved.
• Background information and research, such as
- Investigate any existing systems that handle similar issues
- the results of any market research
- Technical research results on platforms, tools, technologies, algorithms, data structures, etc.
• Discuss any issues encountered so far, including technical and management issues, such as group work.

Xinyi OUYANG and Bolin QU
• The requirements specification for the system to be built.

The front-end team (Xinyi OUYANG):
• Preliminary design of the proposed system and its user interface.

Technical team (Chen LIAO and Minghao ZHOU):
• Records of key implementation decisions, such as programming languages, operating systems, computers, and any other software and hardware to use, and the reasons for these decisions.

Test team (Zihui LUO Shihong HUANG and Bolin QU):
• The result of any initial implementation steps/prototypes.

The gantt chart from bids:
• The time plan for the project.

Bolin QU:
• Meeting minutes, including as an appendix.

Chen LIAO:
• Combine each part.

Notes: The deadline of each part is December 2nd.
The final deadline of the whole report is December 4th.

GRP_TEAM14_6th Formal MEETING MINUTES			
Date & Time	2019.12.11 18:00–19:00	Position	PMB426
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **Brief report the process of Interim Report and get some advice from supervisor.**
- **Summary the whole work of this semester.**

Revise the Interim Report from several aspects:

- Description of Problem to be solved: Using the literature to reflect gaps in theory and practice and try to explain how to fill gap.
- Background Information & Research: Synthesize two subtitles: existing moodles existing system, detailed research implementation method or research on thesis (language and algorithms and platforms and tools).
- Requirements Specification (agreed between Team and Supervisor): Wait for advice and feedback of Supervisor.
- Combine (Initial) Design and (Initial) User Interface (UI): Requires explanation, may need to be put on the process. Add the previous general test cases (cases after negotiating changes with the front-end group).
- Record/Discussion of Key Implementation Decisions (OS, Programming Language, Hardware, Software): Each group writes the problems encountered.
- Results of (Initial) Implementation steps / prototypes: Add a test-specific test chart to the prototype.
- Discussion of problems encountered (technical, personal, management …): Add problems about management.
- Time Plan for the project: On the bid website, compare the current progress with the planned progress and make reflections.
- Quality and usage of sources (in-text citations) to support content/development of argument: Add a little literature, the technical group has three literatures, the test group finds three more literatures, and the front end finds another 1-2.
- References/Bibliography list: Pay attention to writing format.

Note: the DDL of version2 is 18:00pm on December 10th.

Primary Mission:

- **According to the advice of supervisor, revise the Interim Report, adjust word order, change writing format, add the background information and add the reference list.**
- **Improve the appendix.**
- **DDL: 22:00 on December 11th.**

GRP_TEAM14_7th Formal MEETING MINUTES			
Date & Time	2019.1.7 14:00–15:30	Position	PMB
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **Summarize the main points of the last study**
- **Initial connection to the server. The main research is that the direction of java**
- **Based on the python language.**
- **The front-end interface is basically completed, and the rest is filled up.**
- **The main goal of the core algorithm is network connection.**
- **Ask supervisor some suggestion about the core algorithm**

Primary Mission:

- **The rest of task of front-end team is filling up basic information.**
- **Get through the process and pass the video to the algorithm**
- **Research whether script files can be used**

GRP_TEAM14_8th Formal MEETING MINUTES			
Date & Time	2019.1.18 14:00–16:00	Position	Online
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **Report on project progress.**
- **The front-end interactive box has been built and is studying computer visualization.**
- **The background test group builds the jungle framework and links with the front end.**
- **Algorithm transplantation and classification have been solved, and the data set is being perfected.**

Primary Mission:

- **Time of the week: The main task is integration and testing.**
- **Domain name registration issues can't be solved until the DDL.**
- **At least run the program on this machine, due to the influence of the domain name, the WeChat applet can be stranded.**
- **Leave at least one week to write Final report**
- **Test case functional test is sufficient: efficiency time accuracy**

Questions and Answers:

- **When submitting the final report, is it necessary to submit the supporting documents together?**
- **Better have.**
- **In what way is the test data result reflected?**
- **Reserve a chapter in the report, screenshot input and output, reflected in the report**

Content of Documentation:

- **Menu: Software manual: use process, precautions for use, features of the software, functions that can be realized, divided into several sections, each**

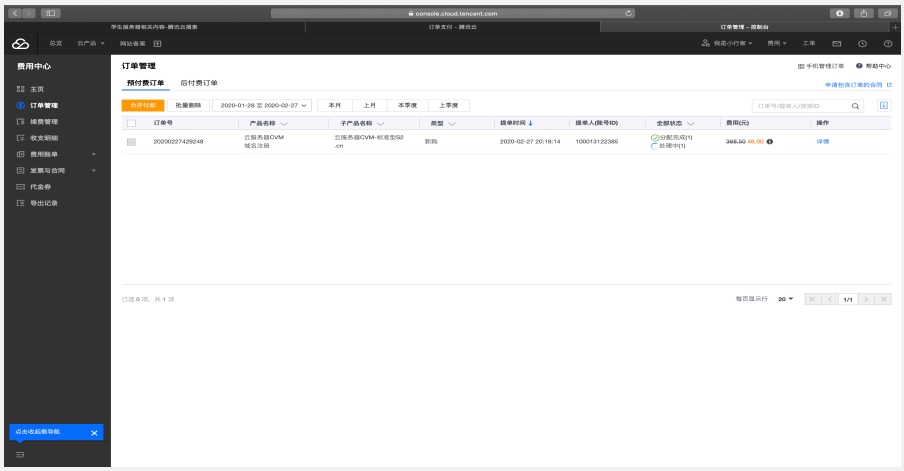
section has a function, record how the customer can use, attach pictures.

- **Test design and test program.**

GRP_TEAM14_9th Formal MEETING MINUTES			
Date & Time	2019.2.3 19:00–20:30	Position	Online
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **Brief the current progress to supervisor in the form of report.**
- **Front end: Test to draw the video capture rectangle.**
    - **The cloud server confirms: Purchase the right to use the student version cloud server and domain name for three months from the Tencent Cloud website. As bellow:**



Primary Mission:

- **Research data transmission and video transmission to the server.**

GRP_TEAM14_10th Formal MEETING MINUTES			
Date & Time	2019.2.17 20:00–21:30	Position	Online
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **Requirements according to the actual situation: within one minute of uploading the video.**
- **The test group has a development program that calls a method on the back end through the front end. The function can download and save from WeChat front end to a place on WeChat back end.**
- **Front-end creates a rectangle to cover objects in the video.**
- **The core algorithm group has made significant progress and can save the video.**

Primary Mission:

- **Add a function to the upload button and connect the front end to the test group.**
- **The core algorithm data set should start playing, training neural network.**
- **SSL certificate also needs one to ensure the domain name can work normally.**
- **Write some documents and papers to prepare for final report.**
- **Read the requirements of the final report in detail, the document is very important.**
- **Write some test report which is about the test results.**
- **The content above can refer to intern report.**
- **Prepare for DDL not to extend: Finish as soon as possible**

GRP_TEAM14_11th Formal MEETING MINUTES			
Date & Time	2019.2.27 20:00–21:30	Position	Online
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **The core algorithm group made progress.**
- **The backend is basically open, only the neural network is trained, and the model file is replaced after training.**
- **The script file runsuccessfully.**
- **Change the file structure of the data in the data set and change the file naming to avoid duplication**
    - **Insufficient data set for passing and shooting**
- **Front end: The rectangle is ready, you can use your fingers to control and zoom in and out.**
- **Information can be returned to the console**

New question:

- **Many people use the applet at the same time, the location may be changed**
- **The language used in the back-end and front-end is different, and the type of data transmitted may be different**

Primary Mission:

- **Computer visualization is still to be studied: return the data of each frame in the later section**
- **Test: Avoid testing third-party code. Add own code, there are not many added functions. Test functional aspects first, then consider unit tests.**
- **The code must not only have comments, but also have auxiliary documents like java documents, readme**
- **Consider using git with version control. Bolin QU is responsible for adding the readme file, uploading the first version on March 16**
- **Start thinking about test documents and test sample.**

Implementation details:

- **Back-end neural network: The main purpose is to put some pictures of the**

test results and talk about the results of the test

- **Test front end: test interface conversion and function buttons, as well as coordinates and length and width, screenshots, put in files**

Deadline:

- **Before 25th of this month, complete the work and begin writing the final report and individual report.**

GRP_TEAM14_12th Formal MEETING MINUTES			
Date & Time	2019.3.15 20:00–21:30	Position	Online
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **The module of Training network is put on the picture.**
- **Report progress of each team.**
- **The software is basically completed.**

Primary Mission:

Final report: Write 1200-1500 words per person, study the writing of final report: Structure of report is confirmed:

- **Menu**
- **Abstract**
- **Introduction**
- **Background & Research (800)**

- **Updated Design (1000)**
  *** UI**
  *** frame**
  *** algorithm**

- **Implementation details (2000)**
  ***UI**
  ***frame**
  ***algorithm**
  *** model training**
  *** system integration**
- **Testing (700)**

- **Achievements (800)**

- **Reflection (2000)**
  *** technical issue**

*** project management**
*** teamwork**

- **Summary (400)**

All group members write report according to the structure and each person is responsible for a part.

GRP_TEAM14_13th Formal MEETING MINUTES			
Date & Time	2019.3.26 20:00–21:30	Position	Online
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **Overview of progress:**
  - **Currently, the front-end UI has been written more than half, and the upload of video is currently being tested. The server is tentatively designated as Tencent Cloud Server, and the system is tentatively designated as Ubuntu 18. For the core algorithm, the LeNet source code has been written and the Pytorch toolkit is used. The Siammask algorithm will be invoked on the server using script files.**

To-do list:

1. Front end:

- **API server communication**
- **Rectangular marquee**
- **data visualization**

2. Server:

- **API front-end communication**
- **Server deployment**

3. Core algorithm:

- **Football dataset compilation**
- **Network training and optimization**

4. Test:

- **Anomaly test**
- **Threshold test**
- **Efficiency test**
- **Accuracy test**

Primary Mission:

- **Realize data interaction between server and WeChat front end**
  - **Complete the data set**

Current problem:

- The .cn domain name applied for the server needs to be filed, and the filing time is too long
- There is no suitable data set for LeNet to do classification training

GRP_TEAM14_14th Formal MEETING MINUTES			
Date & Time	2019.4.17 20:00–21:30	Position	Online
Attendees	Attendees included Heng Yu, Chen Liao, Zihui luo, Minghao Zhou, Xinyi Ouyang, Shihong Huang, Bolin Qu.		
Absents	None		
Conference content			

Topic of Discussion:

- **Show the supervisor a complete small demo to make comments.**
  - **A prompt box pops up for the user and prompts that the upload was successful.**
  - **Confirm whether the ddl problem is enough time and submit according to the original ddl.**
  -

Primary Mission:

- **Refine user interaction and take a moment to improve user experience.**
- **user manual confirmation: According to different users and show certain technical details to users.**
- **test part: test interface consideration process test result test method test plan plus.**

Some advice for modifying report:

- **Put a screenshot in the report.**
- **Plus typical usage scenarios.**
- **The writing template follows the mid-term report.**
- **add ten more references.**
- **Plus more intuitive content, demonstrative content.**
- **Plus SiamMask presentation pictures.**
- **use case flow chart plus system user how to deploy the system.**
- **Refer to the block diagram in the dbi and sm classes. You can borrow the interim report.**
- **Explain that this version is a beta applet. Note that it can only be used by yourself.**
- **Write readme to explain the role of each document.**
- **The appendix can be put together with the final report or listed separately.**

10.2 Requirements

System requirements

1.1 The system can communicate, access and modify data with a separated cloud database that store the information of users.

2.1 The system can be used based on WeChat.
2.2 The system will ask user to log in by WeChat ID.

3.1 The system will let user to upload a football game video.
3.2 The system will check if the video is satisfactory.
3.3 The system will allow user to choose which behavior/members he wants to analyze.
3.4 The system will link to the back end and send the video to it then begin to analyze.

4.1 The system will receive the result from back end after analyzing.
4.2 The system will allow user to choose which way (table / bar chart/..) to view the visualization result.
4.3 The system will allow user to download the analyzing result.

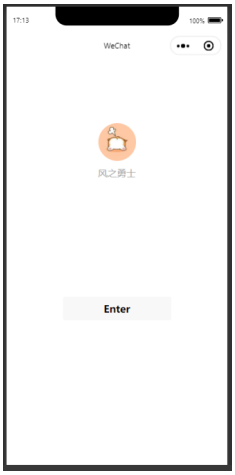
5.1 The system will allow user to share this WeChat applet to others.

User requirements

1. The system shall be used on phone by WeChat applet.
2. The system shall allow users to log in with their unique ID and password.
3. The system shall allow users to upload a video.
4. The system shall keep tracking and analyzing the specified target provided by the users.
5. The system shall show the analysis result on the screen, which includes pass times, number of shots and number of fouls after the video finish.
6. The system shall allow users change the target football player and do the analysis again after the video finish.

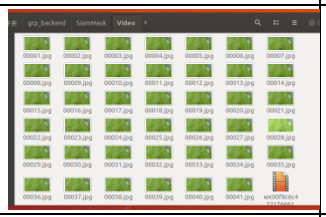
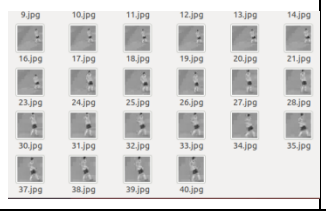
10.3 Test Case

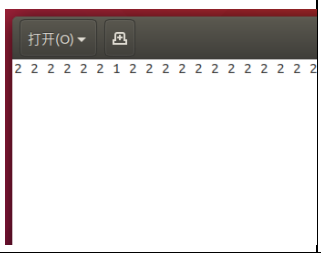
1. UI

Test name	Input	Expected output	Actual output	Test result
Authorization for user account	Pressing button "accessing permission of basic info" in index page	Presenting username and head portrait		Pass
Video selection	Path of the video selected from album in smartphone	Overview of video in user interface		Pass
Send tracking command	Press the button start tracking	Button change from "start tracking" to "on process"		pass

Receive tracking result	Tracking result sent from backend	1.Button changed from "on process" to "succeed" 2.result video presented on front end	1. 2. 	pass
Tracking failed	Receive	Button changed from "on process" to "tracking failed"		pass

2. Backend

Test name	Input	Expected output	Actual output	Test result
data preprocessing	video	List of images of each frame		pass
SiamMask tracking object	1.list of frames 2.parameters of bounding box selecting object in first frame	1.result video 2.list of images that object is filtered out	A frame of the list:  2. 	pass

Object movement analyzing	List of pictures (Snapshot cut by the bounding box in SiamMask)	File containing array showing result predicted for each frame		pass
Data preparation test	Relative path of the image repository	Total number of images (for both train data and test data) and total number of batches (for both train data and test data) in console	Number of training images: 3000 Number of testing images: 30 Number of training batches: 379 Number of testing batches: 8	Pass
Data format test	A batch of processed training data	The shape of this batch **in tensor format** with the dimension of **epoch*1*50*50**	torch.Size([100, 1, 50, 50])	Pass
Network structure validation test	No actual input, the source code of network structure	Printed network architecture in Pytorch supported format	Figure (1)	Pass
Training process supervision test	No actual input, the starting command of the training process	Dynamic updated network training status in console	Figure (2)	Pass
Model saving test	Saving path of the network model	Updated model in correct location with correct name and '.pkl' format	net.pkl	Pass

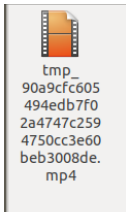
```
CNN(
  (conv1): Sequential(
    (0): Conv2d(1, 16, kernel_size=(5, 5), stride=(1, 1), padding=(1, 1))
    (1): ReLU()
    (2): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  )
  (conv2): Sequential(
    (0): Conv2d(16, 32, kernel_size=(5, 5), stride=(1, 1), padding=(2, 2))
    (1): ReLU()
    (2): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  )
  (out): Linear(in_features=4608, out_features=4, bias=True)
)
```

Figure (1)

```
Epoch: 0 | train loss: 1.3901 | test accuracy: 0.460000
Epoch: 0 | train loss: 1.3901 | test accuracy: 0.460000
Epoch: 0 | train loss: 1.3901 | test accuracy: 0.480000
Epoch: 0 | train loss: 1.3901 | test accuracy: 0.360000
Epoch: 0 | train loss: 1.3901 | test accuracy: 0.340000
Epoch: 0 | train loss: 1.3901 | test accuracy: 0.460000
Epoch: 0 | train loss: 1.3901 | test accuracy: 0.500000
Epoch: 0 | train loss: 1.3901 | test accuracy: 0.413793
Epoch: 0 | train loss: 1.2436 | test accuracy: 0.560000
Epoch: 0 | train loss: 1.2436 | test accuracy: 0.460000
Epoch: 0 | train loss: 1.2436 | test accuracy: 0.440000
Epoch: 0 | train loss: 1.2436 | test accuracy: 0.400000
Epoch: 0 | train loss: 1.2436 | test accuracy: 0.500000
Epoch: 0 | train loss: 1.2436 | test accuracy: 0.480000
Epoch: 0 | train loss: 1.2436 | test accuracy: 0.300000
Epoch: 0 | train loss: 1.2436 | test accuracy: 0.275862
```

Figure (2)

3. Web frame

Test name	Input	Expected output	Actual output	Test result
Send a video to the webframe	Video sent from frontend using http request: e.g. "http://192.168.0.108:8000/"	Presenting username and head portrait		pass
Return the uploaded video to frontend	http request to the path: "http://192.168.0.108:8000/download"	Http response containing result video	Figure (3)	pass
Return analyzing results to frontend	http request: "http://192.168.0.108:8000/result"	http response containing an array	[0 28 12 0]	pass

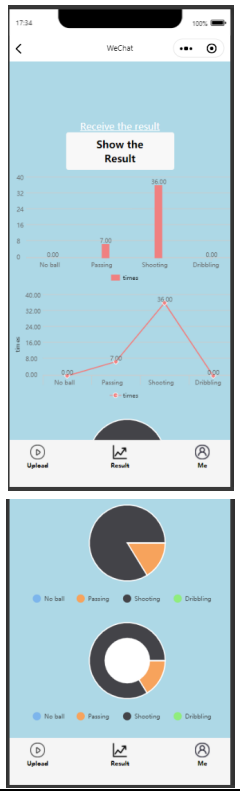
Get result from analyzing data	http request to get analyzing result	Visualization result		pass

Figure (3)

4. System test

In order to test the functionality of the whole system, procedure shown below is the operation to perform the test.

1. login the applet though Wechat.
2. upload video in the local video album to the applet.
3. use the frame to select a player in the video to track.
4. view the data retrieved from the video.

5. Scenario test

Eric is a coach of a fooball team. He wants to analysis the football game videos and identify all area ability of his team member. First, he opens the

WeChat app and logins football movement recognition and analysis applet with WeChat account. He clicks the “choose” button to choose one video and clicks the “upload” button to upload it from the system album to the applet. Then, he uses the frame to select a player in the video to track and clicks the “start tracking” button to start tracking. When the tracking accomplished, the final data will show on the “record” page and he clicks the “record” button to view the results. Finally, he clicks “upload” button to reupload another video and analysis again.

The scenario test is performed by non-developer of the project try to use the software for a pre-designed purpose. The test evaluates whether the interface is user-friendly, efficiency and correctness of the calculation, and the time required for user to achieve the purpose. Each of the criteria has been fulfilled except the interfaces might have the potential to be improved when supervisor performs the part of “Eric” introduced before.

10.4 User-Manual

1. Writing purpose:

Inform users about the functions and precautions provided by the vision-based football match analysis system.

Explain the meaning of some result parameters in the record interface of the applet.

Help users to figure out the whole process of using the applet to analysis football video

2. How to Use the Application

Preparation:

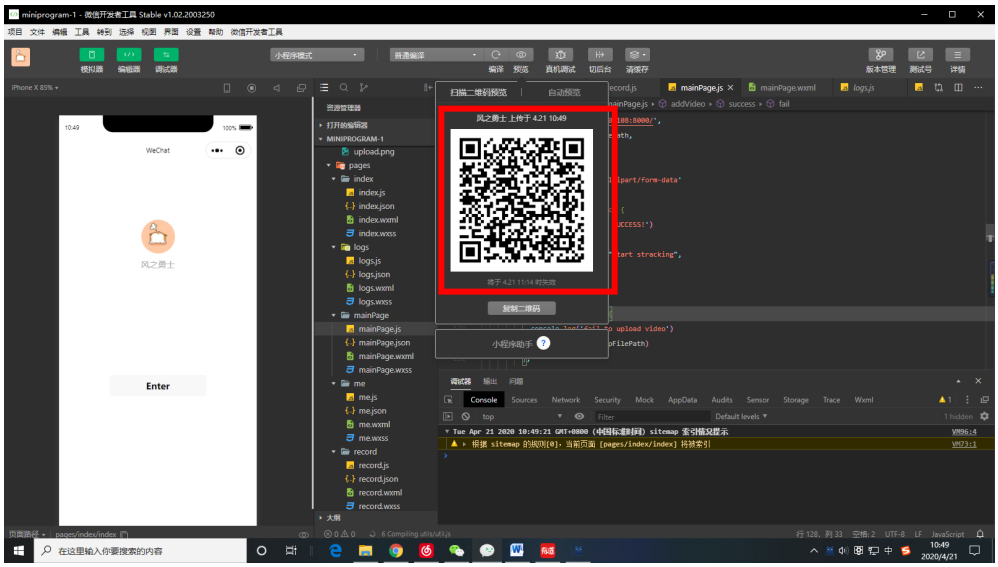
The WeChat application, which is a product of Tencent. Vision: 7.0.12

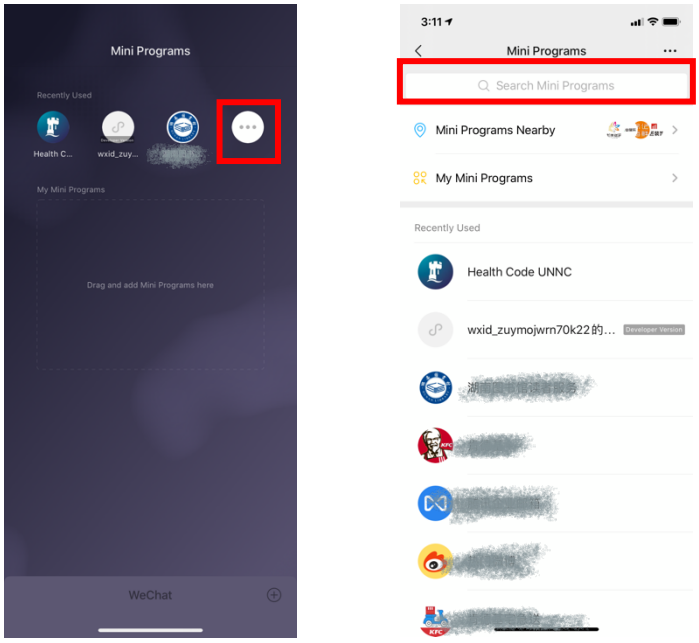
A logical and available WeChat account.

2.1 Account Operation:

Login

On the "chats" interface of WeChat, user can click the search bar to search the "football video analysis" applet, which is a kind of Mini-program. Click the top option. If it is the first time for the user to login the applet, the user is supposed to click the avatar and green "allow" button will appear on the screen, click it to allow the applet obtain user's name, profile photo, region and gender. This is aimed to allow users to upload local videos to the front-end of the applet. (this is how to use of normal WeChat mini program, since this app have not been uploaded and established to public as formal one, you only can execute on personal computer with WeChat Development tools.)

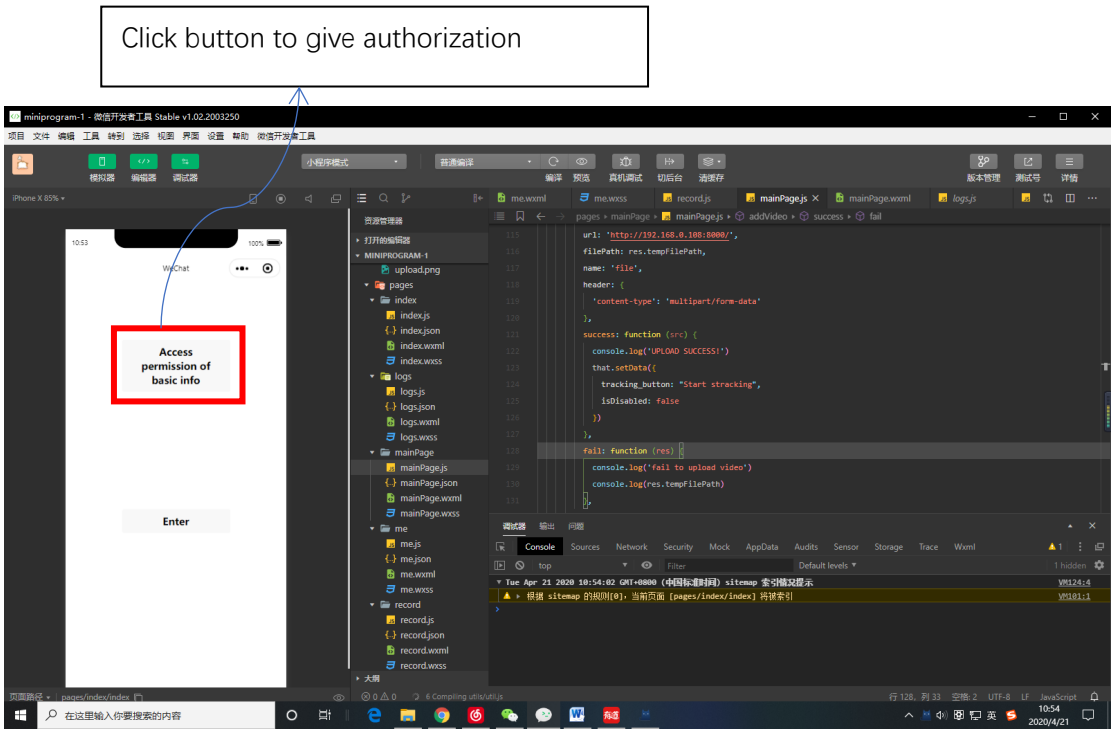




Search the Mini Program

2.2 Index page:

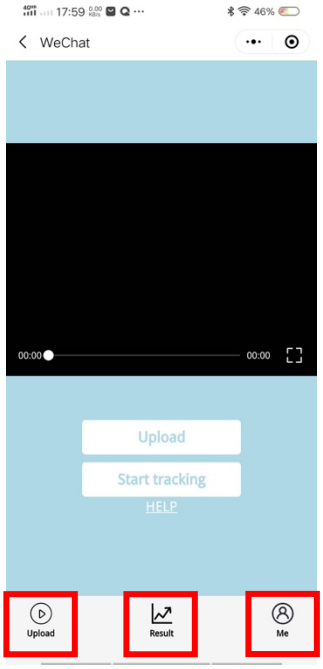
The user can click "Enter" button to enter the applet directly. In the first time, the user will be asked for accessing information of WeChat account. The mini program can always access account later on once being authorized.



Special illustration:

There is no need to concern register or change the password of account. The applet is related to the WeChat application. Therefore, the account is internally linked, only if an available WeChat account can be used in WeChat applet. Besides, the basic account operation can be achieved in the "Me" interface of WeChat.

2.3 User Interface:



Upload Page:

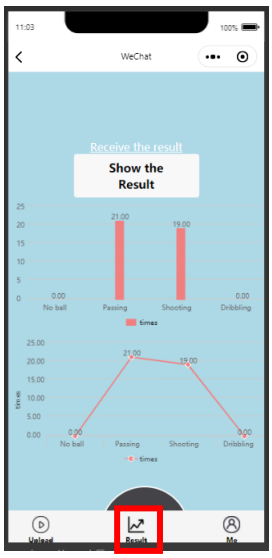
In the upload interface, there are three buttons for operation (show above). Firstly, user can click "Upload" button to choose a local video from local album or record a new video. After choosing, in the album page, user click the green "send" button to send the video to applet. Then, the video is shown on the screen and a rectangle can be dragged from the upper left corner of the video. Next, the user can click the "HELP" button to know about how to move the frame by one single finger and adjust size of the frame with two fingers to let the rectangle cover one specific football player. Finally, user click the "Start tracking" button to start the tracking process. If the user wants to track a new video, first click "Upload" to upload the new video, then click "start tracking" to re-analyze the new video (the rectangle will be in the place where the old video was set)

Note: As long as moving the rectangle to choose the target before starting tracking is feasible.

If the tracking is successful, there will be a new video to replace the old video (in the new video, people are tracked in real time by the rectangle)--this functionalities only works on when simulating on smartphone but not work on PC.

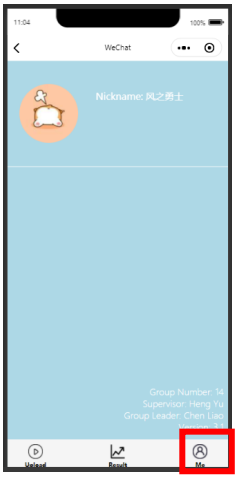
Result Page:

User click the "Result" button to jump to the result interface to check out the result of analysis. Then click the "show" button. There will be four different ways to show the data: histogram, line chart, pie chart, ring chart. And there are four actions (integer 0: noball 1: passing 2: shooting 3: dribbling) shown in different kinds of chart. User can see the result clearly and intuitionally.



Me Page:

In this interface, users can check the avatar and nickname of their own. The introductions of group 14 and this applet are also shown under the white line.



2.4 The whole process of analyzing a football video:

User opens WeChat application, find the mini-program and login it. Then, allow access to local albums and choose one video from it. Next, adjust position and size of the rectangle, let the rectangle cover target player. Click "Start tracking" button. The result of analysis will show on the "Result" interface shortly. If the user wants to reanalyze the old video or analyze a new video, user should reupload a video and repeat the above operations.

Note: Due to the system's own bug, applet ran on IOS system can limit the video time in 60s, however, in Android system, the time of video can't be limited.

3. How to Install the Application:

The entire system is divided into two parts, which is required to install on separated devices.

Preparations: One computer with Windows system (or Mac OS) running UI, one computer with Linux system as backend, one Local Area Network.

3.1 Install backend:

Firstly, your computer with Linux system should be equipped with conda

environment (if not, install conda first).

Then, activate conda environment (you may use command "conda activate")

Install package grp_backend.zip and decompress the package

- Setup environment for program SiamMask(you may refer to instructions from: https://github.com/foolwood/SiamMask)

```
cd grp_backend/SiamMask
export SiamMask=$PWD
```

setup python environment:

```
conda create -n siammask python=3.6
source activate siammask
pip install -r requirements.txt
bash make.sh
```

Add the project to your PYTHONPATH

```
export PYTHONPATH=$PWD:$PYTHONPATH
```

Download the SiamMask model

```
cd $SiamMask/experiments/siammask_sharp
wget http://www.robots.ox.ac.uk/~qwang/SiamMask_VOT.pth
wget http://www.robots.ox.ac.uk/~qwang/SiamMask_DAVIS.pth
```

Now you should be able to run sample(demo.py) of SiamMask (not the whole program)

```
cd $SiamMask/experiments/siammask_sharp
export PYTHONPATH=$PWD:$PYTHONPATH
python ../../tools/demo.py --resume SiamMask_DAVIS.pth --
config config_davis.json
```

- Running the sever

Go to file of sever

```
cd ../../../Django_sever
```

Execute sever:

```
Python3 manage.py runsever 0.0.0.0:8000
```

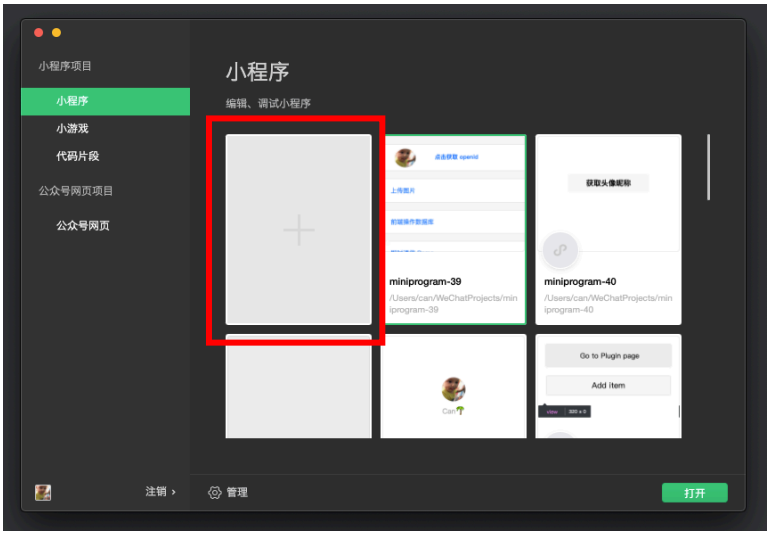
3.2 Setup frontend:

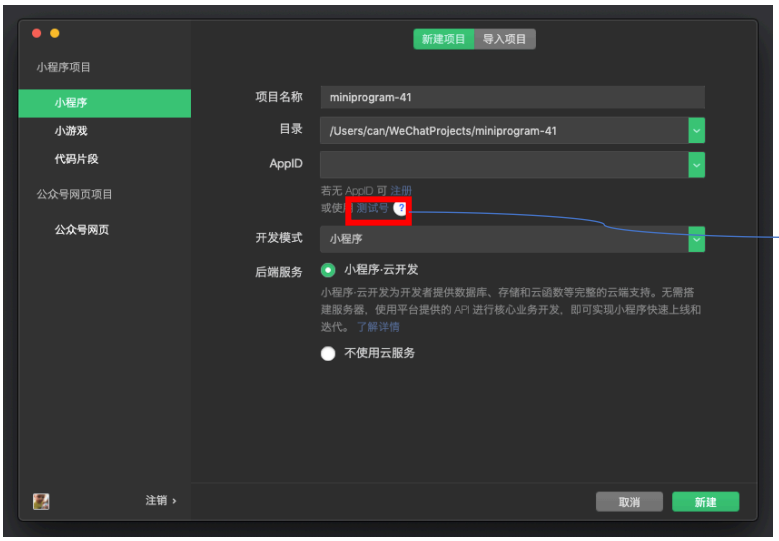
The whole frontend is developed on Windows/macOS (need another computer)

Firstly, install WeChat Development tool which is available at: https://developers.weixin.qq.com/miniprogram/en/dev/devtools/stable.html

Then download grp_frontend.zip and decompose the package

Open the WeChat Development tool and import project:





After entering the frontend code, please change the IP address inside code:

From minprogram/pages/MainPage/MainPage.js: line 115, line 185, line 199

From miniprogram/pages/record/record.js: line 105

You should replace the IP address with the IP address of your backend sever (you may use "ifconfig -a" to check IP in Linux sever.)

Be careful the two computers are placed within same LAN!

After these operations, you should successfully execute the program.

4. How to Use for Admin

4.1 How to update performance of classification net

Generally, to update the network, the administrator needs to modify the

training dataset, train the network and replace the model. The training data is under the folder '\dataset', the administrator could replace or add images into each of the categories to alter the dataset for a better training result. However, be aware that the folder structure and naming within the dataset should never be modified. Additionally, images should be uniform 50*50 gray-scale ones within .JPG format.

For retraining the network, run 'train.py' individually and there will be a model file named 'net.pkl' appearing in the same folder with the script. The administrator could monitor the training process with the updated information in console.

```
Epoch: 7 | train loss: 0.5125 | test accuracy: 0.700000
Epoch: 7 | train loss: 0.5125 | test accuracy: 0.540000
Epoch: 7 | train loss: 0.5125 | test accuracy: 0.580000
Epoch: 7 | train loss: 0.5125 | test accuracy: 0.620690
Epoch: 8 | train loss: 0.4792 | test accuracy: 0.700000
Epoch: 8 | train loss: 0.4792 | test accuracy: 0.540000
Epoch: 8 | train loss: 0.4792 | test accuracy: 0.680000
Epoch: 8 | train loss: 0.4792 | test accuracy: 0.660000
Epoch: 8 | train loss: 0.4792 | test accuracy: 0.560000
Epoch: 8 | train loss: 0.4792 | test accuracy: 0.540000
Epoch: 8 | train loss: 0.4792 | test accuracy: 0.720000
Epoch: 8 | train loss: 0.4792 | test accuracy: 0.724138
Epoch: 8 | train loss: 0.4475 | test accuracy: 0.500000
Epoch: 8 | train loss: 0.4475 | test accuracy: 0.600000
Epoch: 8 | train loss: 0.4475 | test accuracy: 0.600000
Epoch: 8 | train loss: 0.4475 | test accuracy: 0.600000
Epoch: 8 | train loss: 0.4475 | test accuracy: 0.720000
```

The administrator could copy or cut this newly trained model file and supersede the original model file in grp_backend/SiamMask/tools/net.kpl, thus the software will use the mew model for classification the next time it runs.