Deep learning-based motion assistance system

directory

Summary: A healthy life is inseparable from daily exercise. Nowadays, people are also paying more and more attention to the effectiveness of sports and the health protection of sports, hoping that they can do standard and effective, and also hope that they can reduce the injuries caused by sports. Therefore, in view of these two popular directions, a sports assistance system is designed to identify the movement state of the human body through artificial intelligence technology, extract the bone state, and provide real-time feedback and prompts to the movements of athletes. AT THE BEGINNING OF THE RESEARCH, THE SYSTEM USED BLAZEPOSE AS THE CORE, BUT ENCOUNTERED AN ERROR IN THE PROCESS OF RESEARCH, SO IT FINALLY CHANGED TO OPENPOSE AS THE CORE OF THE SYSTEM CONTENT, AND FURTHER OPTIMIZED THE SYSTEM BASED ON THE BODY_25 OF OPENPOSE AND COCO18. At present, the system can realize the functions of prompting the action error and judging the standard degree of posture according to the feedback of the exercise state. The system can assist people to exercise, improve the efficiency and safety of sports, and explore the wisdom and standardization of smart fitness, as well as the intelligence and standardization of body testing.

Keywords: artificial intelligence; Exercise aids; computer vision; Posture recognition; Bone binding; Smart Fitness.

1 Introduction

1.1 Background

In today's environment of pursuing a healthy and quality life, the topic of exercise and health is widely discussed. For the general public, the primary purpose of sports is to keep fit. In 2014, the State Council's "Several Opinions on Accelerating the Development of the Sports Industry and Promoting Sports Consumption" clearly stated that national fitness has become a national strategy. With the rise in the number of fitness people, doubts arise spontaneously due to fitness.

First of all, for those who work out at home, without the company of a fitness coach, following the pictures or videos on the Internet to exercise, it is more likely to cause injuries due to wrong movements. How to effectively reduce accidental injuries and long-term damage to a certain part of the body is a problem encountered by most fitness people. As a result, I have concerns about exercising at home.

Secondly, many people who work out at home by themselves find it difficult to stick to it. After a period of exercise, I often don't see the results I want, either to lose weight or to gain muscle. After a short period of exercise, there is no effect, probably because the persistence time is not long enough to reach a certain amount of accumulation; However, for some people who have not achieved results for a long time in fitness, it is often because of the non-standard movements that lead to "ineffective fitness". "Ineffective fitness" is equivalent to doing superficial work, and the results will deviate greatly from your own presets. Non-standard posture can even lead to deviation from the goal, such as girls whose goal is to lose fat, and there will be anxiety about "muscle legs" and "muscle arms" during exercise.

The above situation is more common in Cloud Fitness. In response to this problem, if there is an auxiliary exercise equipment that can visualize one's own movements during exercise, correct and prompt wrong movements, and judge whether the movements are standard, it will help fitness people achieve their goals and eliminate some of their concerns.

The use of artificial intelligence has already covered many fields. The application of artificial intelligence to fitness sports should be the development trend of the fitness sports industry, and the fitness sports will be intelligent and standardized. A system that can correct and prompt wrong movements, and judge whether the movements are standard, will also be loved by the sports and fitness industry. At the same time, when applied to judging actions, the complementarity of artificial intelligence and humans will make the evaluation more scientific.

1.2 Research status

By reviewing a large number of literatures, it is found that most of the applications of artificial intelligence in sports and fitness at home and abroad are for the analysis and training of professional athlete data and the analysis of body base data. Since the epidemic, with the rise of demand for home exercise, the number of studies in related fields has also increased in the past two years, but the overall number is still small.

Jiang Mo, Liu Shanqiu, and Wang Yanyun analyzed the pros and cons of smart fitness in "Thinking about the "Cold" Behind the "Hot" of Smart Fitness" [1]. It is pointed out that its "heat" lies in the upgrading of fitness consumer demand, the breakthrough and innovation of artificial intelligence technology, and the support and guidance of the government. Its shortcomings lie in the emphasis on marketing over content, the immaturity of technology research and development, and the obvious phenomenon of investment surge.

Wen Xu et al. preliminarily established a non-contact measurement method of aerobics energy consumption based on OpenPose in "Non-contact Measurement Method of Aerobics Energy Consumption Based on OpenPos e Computer Vision Algorithm" [2].

Ding Zehao pointed out in "Investigation and Analysis of the Use of New Smart Devices in Domestic Leisure and Fitness Activities: A Case Study of Intelligent Personal Trainer System Based on Motion Capture Equipment" [3] that smart fitness equipment can make up for the limitations of gym teaching in the traditional sense, that is, the personal ability and quality of coaches largely determine the difficulty of injury for students. New smart devices can reduce this annoyance by providing participants with standardized movements and data-visualized fitness guidance.

Due to the relatively small and relatively recent research on related aspects, I have learned the theoretical feasibility of the application of artificial intelligence in sports assistance through the study of relevant materials. Through the study of some achievements in other fields, it is believed that the application of computer vision here will reap a relatively successful result.

1.3 Research Objectives

The research goal of this project is to develop an AI-based motion assistance system that can feedback the athlete's movement state through the capture of bone key points, prompt the movement errors, and judge the standard degree of posture, which is discussed as follows:

(1) Posture recognition function

The posture recognition function includes two parts: the error correction prompt function and the function of judging the standard degree of posture.

Error correction prompt function: the athlete points the camera at his whole body, and the system takes uninterrupted automatic pictures to extract the key points of the bones, and the system judges whether the human posture is unreasonable according to the key points in the photo. If there is an unreasonable posture, feedback will be given in time, and the athlete will be reminded to make adjustments and show the correct movement points.

Function of judging the standard degree of posture: Compared with the previous error correction function, this function is mainly aimed at those irregular postures that will not cause harm to the human body, but also cannot serve the purpose of this fitness action. The athlete points the camera at his whole body, and the system takes uninterrupted automatic photos to extract the key points of the bones, and the system judges whether the human body posture has achieved the desired exercise and fitness effect according to the key points in the photo. This function can also be used for evaluation, which is more intelligent and standardized as a human assistance system.

(2) Bone rigging function

To achieve this, you first need to rig the bones. Bone rigging is the most practical and straightforward visualization application. After the photo is entered, the key points of the human skeleton can be automatically marked and coordinated, so that the next operation can be easily carried out. Arguably, rigging is at the heart of this system.

2 Research Methodology

This project mainly adopts two methods: documentary method and experimental method, as follows:

2.1 Documentary Method

After formulating the general idea of the project, by entering keywords such as "artificial intelligence sports" and "intelligent fitness" on CNKI, by reading articles to understand the research status of students' homework and learn from the technical route of the article, determine the main functions and specific implementation methods of the system, and then search for literature through keywords such as "BLAZEPOSE", "OPENPOSE" and "computer vision", learn the technical principles, and evaluate the feasibility.

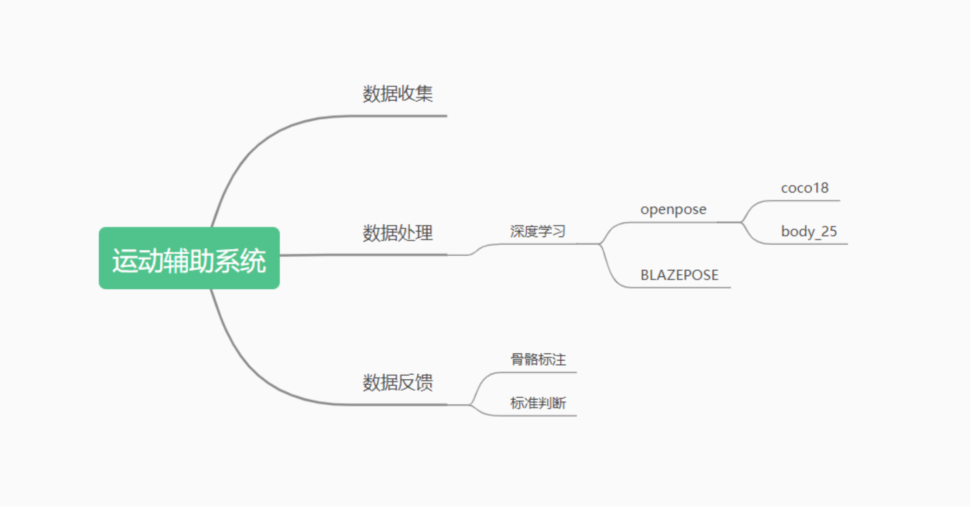
2.2 Experimental method

This project mainly uses deep learning under the Python framework to implement functions. Use the feature map generated by VGGG16 to import the neural network. After going through the stages of PAF, LTP, STp, etc., the confidence m ap is finally obtained. When the results are fed back, the human skeleton frame is drawn using a bipartite graph and a Hungarian matching algorithm. In the process of experimentation, the models and algorithms used were constantly adjusted, and after many local runs and online deployments, repeated tests, and error corrections. In the end, relatively good results were obtained.

3 Project Process

3.1 Design scheme

If the motion assistance system wants to realize the function of bone binding and posture judgment, it mainly includes three aspects



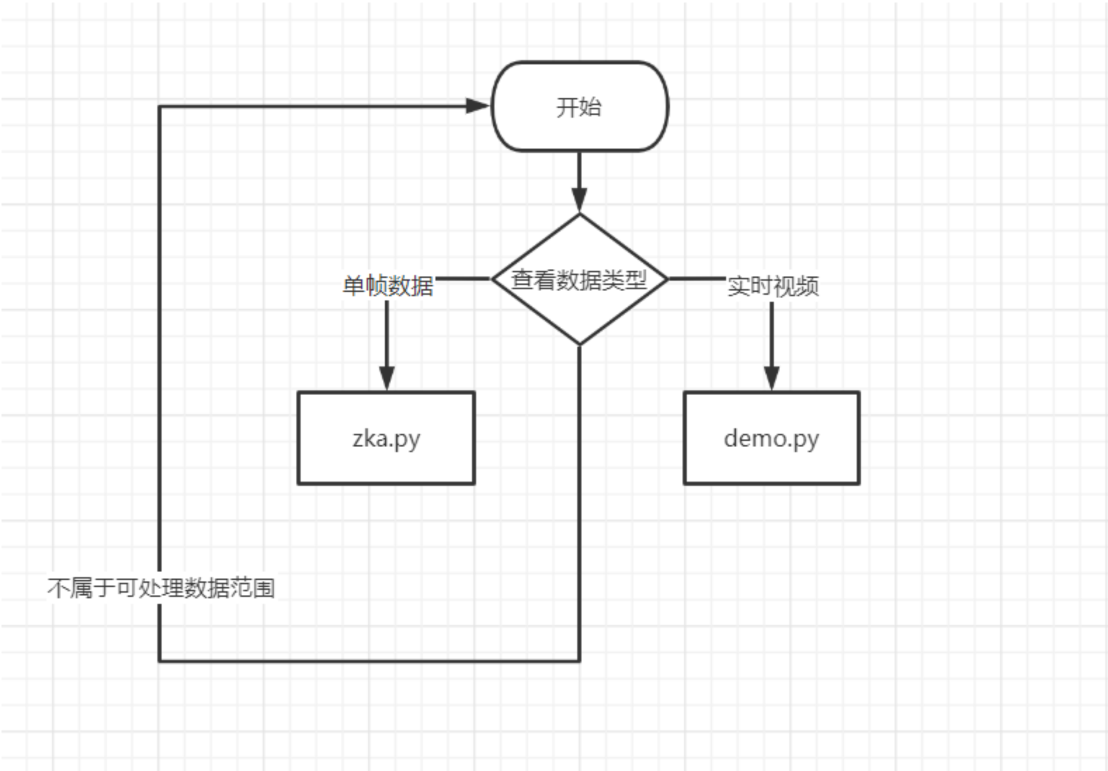
that need to be considered: data collection, data processing, and data feedback.

Figure 2 Functional diagram of the motion assistance system

3.2 Function Introduction

3.2.1 Data Collection

There are two types of data that can be received and processed by the system: JPG and MP4. The program flow diagram of the data acquisition part is shown in Figure 3, and the system can process single frame data as well as real-time video data. Single-frame photo



import analysis, evoking ZKA module; The data processing of real-time video needs to be further processed by calling the demo module. Other types of data will not be processed by the system.

Figure 3 Data collection flowchart

After getting the above information, the system will call the corresponding program in the background to enter the next link.

3.2.2 Data Processing

Data processing is mainly used to predict and judge bones (joints) through models obtained by deep learning.

Taking OPENPOSE's coco model as an example, a total of 18 feature points of the human body are prepared, and after the data is imported, the data will be initialized and loaded first, and then captured according to the existing model, and the effective point pairs will be judged, and the pair of points will be connected and finally the drawing of the human skeleton will be completed. Of course, it is not only necessary to focus on the point itself, but also to expand its scope to avoid interference from the surrounding environment. In this case, the accuracy of the bone drawing is inextricably linked to the chosen model and algorithm.

The schematic diagram is shown in Figure 4.

Fig. 4 Schematic diagram of COCO18 prediction

Considering that the processing of real-time video (especially in the CPU environment) requires a lot of computing power, there is often a shortage of computing power. In this regard, most models take the approach [4] [5] by first obtaining the key points of the human skeleton, and then matching the two bones by Euclidean distance to obtain the corresponding results.

However, not every test can detect a "healthy" bone, and it is often because the limbs are too twisted or part of the torso is obscured, and the key points are missing. In response to the phenomenon of missing key points in this detection, if it is a dynamic video, the bone information of the previous frame (if any) can be used to fill in and complete the entire frame, which is also the embodiment of motion prediction. This adds integrity to the bones and prevents them from flickering during video playback.

3.2.3 Data feedback

In addition to the schematic diagram and running time of the skeleton model shown in Figure 4, the pose recognition calculates the coordinates of the points corresponding to each joint point (not directly written, as covered in the code, see 4.3 for details). This feature enables the judgment of the position between the key points to be quickly realized, in other words, the difference between the posture of the person at a fixed point and the standard attitude can be judged by the coordinates between the two points.

3.3 Technical Methods

The key to the motion assistance system in this project is that the data processing has high requirements for algorithms and models. The purpose is to accurately obtain the position of the key points of the bone, and to complete the prediction and subsequent judgment and analysis of the human bone as quickly as possible, that is, to pursue accuracy and timeliness. IN RESPONSE TO THIS REQUIREMENT, WE CONSULTED A LARGE NUMBER OF MATERIALS UNDER THE "COMPUTER VISION" MODULE, AND AFTER COMPARISON AND STUDY, WE FINALLY LEARNED AND UNDERSTOOD THE TWO ALGORITHMS OF BLAZEPOSE (PUBLISHED BY GOOGLE RESEARCHERS AT CVPR 2020) AND OPENPOSE (AN OPEN SOURCE PROJECT DEVELOPED BY THE CARNEGIE MELLON UNIVERSITY TEAM). For these two fresh algorithms, the autonomous model training was attempted, and the mature model was used for reference.

3.3.1 BLAZEPOSE ALGORITHM

Blaze is a series of algorithms that pursue "extreme speed".

BlazePose, is a lightweight convolutional neural network architecture dedicated to human pose estimation.

BLAZEPOSE IS A POWERFUL MODEL WITH 33 KEY POINTS, RUNNING AT MORE THAN 30 FRAMES PER SECOND, AND THE RESULTING POSE IS RELATIVELY COMPLETE. His real-time and accuracy are expected to be used for

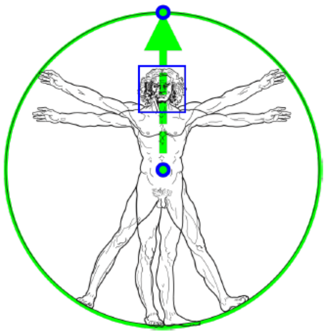


Figure 2. Vitruvian man aligned via our detector vs. face detection bounding box. See text for details.

fitness tracking and real-time use cases.

FIGURE 5 INSPIRATION FOR THE BLAZEPOSE ALGORITHM

THERE ARE CURRENTLY FEW STUDIES ON BLAZEPOSE IN CHINA. Typing "BLAZEPOSE" as a keyword on CNKI, only one related article was searched [6]. Therefore, we chose to learn more fully about its content on the CSDN forum, and found that it is very suitable for use in the sports assistance system of this topic.

3.3.2 OpenPose algorithm

OPENPOSE IS CURRENTLY THE MOST POPULAR ATTITUDE NETWORK ESTIMATION SYSTEMIt is a bottom-up detection algorithm that detects key points first and then people. Bottom-up algorithms, which perform well in a small number of people, especially for individual people.

COMPARED TO BLAZEPOSE'S RAPIDITY,OPENPOSE HAS BETTER COMPATIBILITY AND FUNCTIONALITY, IN ADDITION TO LIMB BONES, THERE ARE ALSO MODELS FOR HAND MOVEMENTS AND FACIAL FEATURES, AND THE APPLICATION FUNCTIONS

ARE QUITE EXTENSIVE.

AFTER READING THE REVIEW LITERATURE ON THE OPENPOSE ALGORITHM[7] IT IS FOUND THAT THE ORIGINAL VERSION OF OPENPOSE HAS THE GREATEST

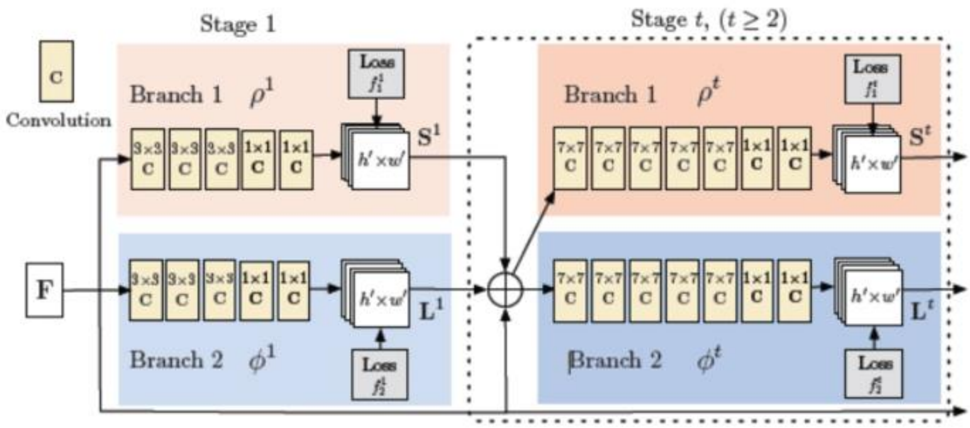


Figure 3. Architecture of the two-branch multi-stage CNN. Each stage in the first branch predicts confidence maps $\mathbf{S}^t$, and each stage in the second branch predicts PAFs $\mathbf{L}^t$. After each stage, the predictions from the two branches, along with the image features, are concatenated for next stage.

APPLICABILITY, AND TAKES INTO ACCOUNT THE SPEED AND ACCURACY, WHICH IS MORE BALANCED, AND IS MORE SUITABLE FOR THE RESEARCH OF THIS TOPIC.

Fig.6 Neural network structure diagram of OpenPose

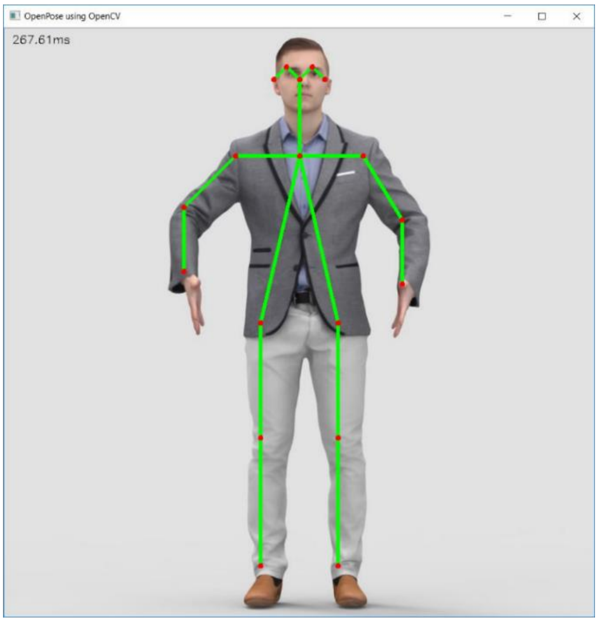


Figure 7 OpenPose (CoCo18)

3.4 Technical implementation

For the whole motion assistance system, after careful study and understanding of the deep learning of posture recognition, I tried the practice and innovation of this technology.

3.4.1 PROCESS FOR TRAINING A NEW MODEL (BLAZEPOSE).

First, try to train a new model. BLAZEPOSE is extremely cost-effective due to its accuracy of 33-point skeletal features and its speed. Since the laptop used did not have a GPU to use, it was impossible to complete the local training of the model, so the cloud platform was borrowed for online deployment. In the study, *two platforms, Kaggle* and HUAWEI CLOUD, were used.

Do the following (first experiment on the Kaggle platform):

Step 1: IMPORT BLAZEPOSE FILE AND LSP DATA.

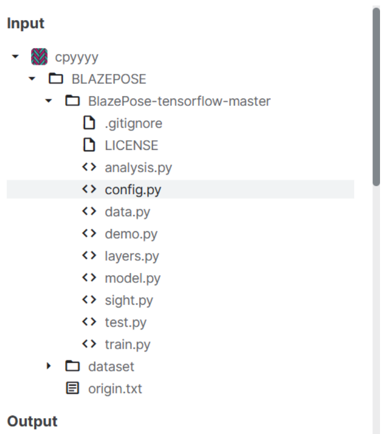


Figure 15: Document display

(Note: The project comes with the download of the LSP dataset, but the official website of the LSP dataset cannot be directly accessed in the mainland, you can find the dataset on the Internet and put it in the project*py*.)

Step 2: Heatmap

Adjust the config.py train *mode=0 withcontinue_train=0*

```
train_mode = 0
continue_train = 0  # 0 for random initialize, >0 for num epoch
```

Figure 16: File parameters

Go to the folder where the file is located (the default location

```
[3]: cd /kaggle/input/cpyyyy
/kaggle/input/cpyyyy
```

of Kaggle is the WorK folder).

Figure 17: Notebook commands

Start training

```
run /kaggle/input/cpyyyy/BLAZEPOSE/BlazePose-tensorflow-master/train.py

Found lsp dataset.
Reading dataset...
2023-02-01 16:04:16.750918: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:937] success
ful NUMA node read from SysFS had negative value (-1), but there must be at least one NUMA nod
e, so returning NUMA node zero
2023-02-01 16:04:16.848310: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:937] success
ful NUMA node read from SysFS had negative value (-1), but there must be at least one NUMA nod
e, so returning NUMA node zero
2023-02-01 16:04:16.849260: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:937] success
ful NUMA node read from SysFS had negative value (-1), but there must be at least one NUMA nod
e, so returning NUMA node zero
2023-02-01 16:04:16.851711: I tensorflow/core/platform/cpu_feature_guard.cc:142] This TensorFlo
w binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU
instructions in performance-critical operations:  AVX2 AVX512F FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2023-02-01 16:04:16.852088: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:937] success
ful NUMA node read from SysFS had negative value (-1), but there must be at least one NUMA nod
e, so returning NUMA node zero
```

Figure XVIII: Training commands

(Note: If training in Terminal, change run to python)

An error occurred in Kaggle showing that the memory was insufficient. The official solution given by Kaggle is to use the Google cloud disk service, but it is not available in mainland China.

```
2023-02-01 16:04:37.463058: W tensorflow/core/framework/cpu_allocator_impl.cc:80] Allocation of
1572864000 exceeds 10% of free system memory.
2023-02-01 16:04:40.166828: W tensorflow/core/framework/cpu_allocator_impl.cc:80] Allocation of
1572864000 exceeds 10% of free system memory.
2023-02-01 16:04:42.315392: W tensorflow/core/framework/cpu_allocator_impl.cc:80] Allocation of
1571291136 exceeds 10% of free system memory.
2023-02-01 16:04:45.019126: W tensorflow/core/framework/cpu_allocator_impl.cc:80] Allocation of
1571291136 exceeds 10% of free system memory.
2023-02-01 16:04:47.014613: W tensorflow/core/framework/cpu_allocator_impl.cc:80] Allocation of
1572864000 exceeds 10% of free system memory.
```

Figure 19: Screenshot of the error

Later, I tried to modify the parameters and changed the batch _size from 256 to 8, but it still failed, so I had to give up Kaggle and continue the project on the HUAWEI CLOUD platform.

HUAWEI CLOUD

The GPU provided by Kaggle is a P100 for training. On HUAWEI CLOUD, the specifications shown in the following figure are used.

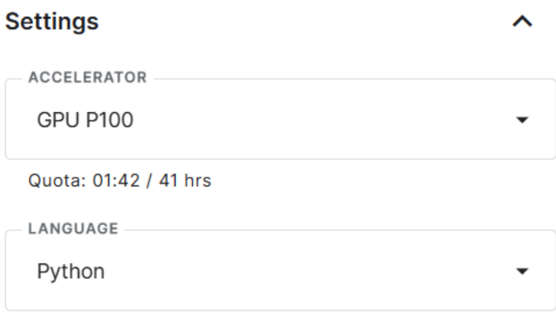
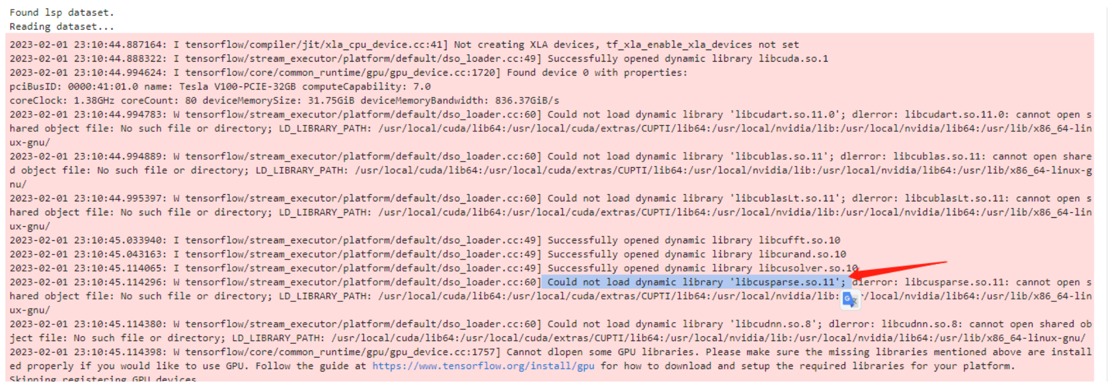
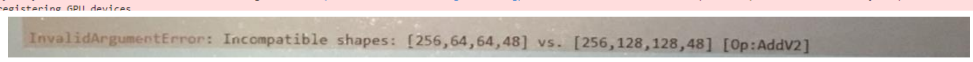


Figure 19, 20: GPU specifications

Complete the preceding steps in Kaggle again in HUAWEI CLOUD and





heat map them. A new error has appeared.

Figure 20, 21: Error

Eventually, it was found that the error appeared in the shape calculation. After trying to change the shape value, only the initial acc appeared, and then the weight error appeared. However, there is too little information to solve this problem in a short period of time.

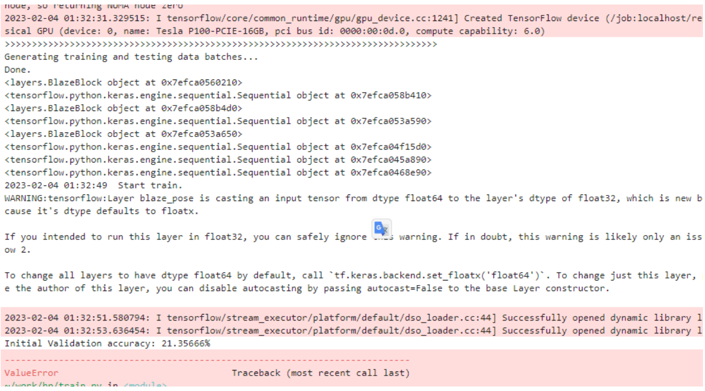


Figure XXII: Failed model

In the end, only a mutilated model was obtained. CONSIDERING THE TIME AND EFFECT, TRY TO DEPLOY A MATURE MODEL OPENPOSE INSTEAD.

3.4.2. Use of mature models (on-premise).

OPENPOSE HAS TWO MATURE MODELS, WHICH ARE DESCRIBED SEPARATELY HERE AND COMPARED IN CHAPTER 3.5.

3.4.2.1 openpose(coco18)

Local environment: Python 3.8.2

Library requirements: cv2, numpy, argparse

Configuration file: openpose.py(Start the program)

graph_opt.pb(The trained model is only about 7MB in size)

image(Test pictures)

(Note: When downloading *OpenCV after* attempting, you need to download the lower version, otherwise an error will be reported).

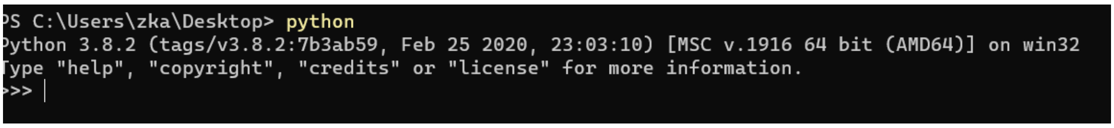
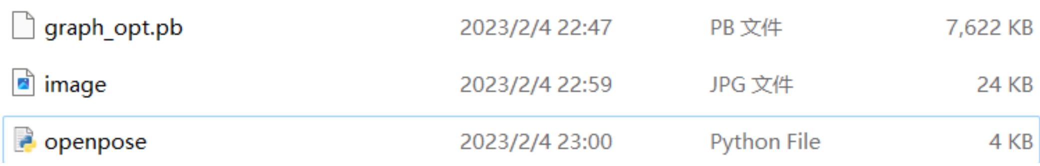


Figure 8: OpenPose (CoCo18) environment is displayed

Figure 9: Local installation of OpenCV

Figure 10: The configuration file is displayed



How to use it: Enter python *openpose - input image in the console*

(Note: If you want to change to batch reading, you only need to modify the read statement in the openpose .py file to change it to a loop.))

3.4.2.2 openpose(body_25)

Local environment: Python 3.8.2

Library requirements: cv2, time, math, numpy, config

Profile:

models

pose

body_25

pose_deploy.prototxt (prototxtdata)

pose_iter_584000.caffemodel(model)

Images

pose.jpg(Test image)

config.py (parameter).

predict.py (Core Forecast File).

zka.py (Startup File).

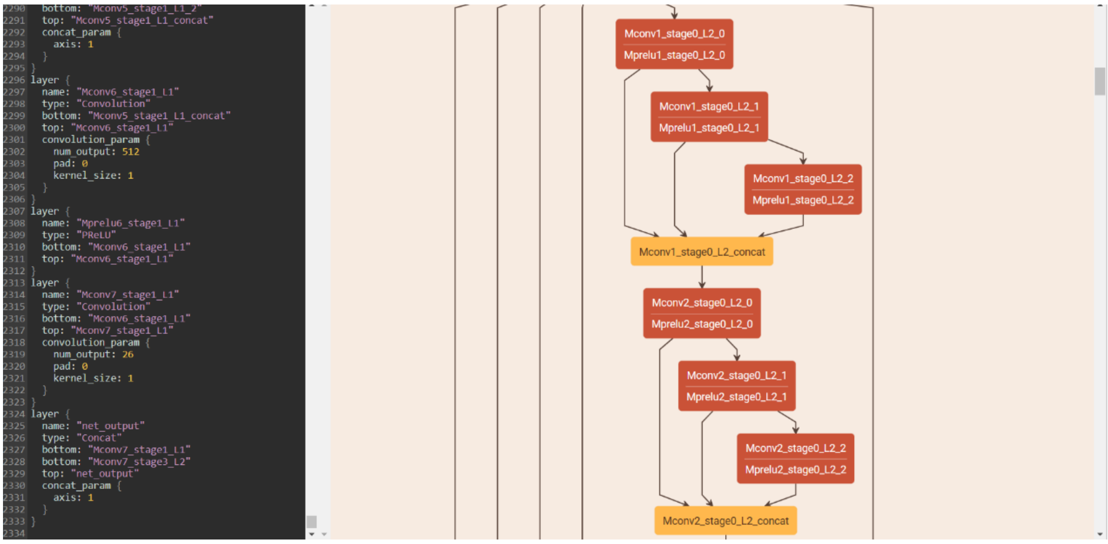
How to use: Double-click the zka *.py*

```python
import cv2
import time
import numpy as np
import matplotlib.pyplot as plt
from config import *

class general_mulitpose_model(object):

    # 初始化 Pose keypoint_num: 25 or 18
    def __init__(self, keypoint_num):

    # 加载openpose模型
    def get_model(self):

    # 获取关键点
    def getKeypoints(self, probMap, threshold=0.1):

    # 获取有效点对
    def getValidPairs(self, output, detected_keypoints, width, height):

    # 连接有效点对，获取完整的人体骨骼图
    def getPersonwiseKeypoints(self, valid_pairs, invalid_pairs, keypoints_list):

    # 关键点连接后的可视化
    def vis_pose(self, img_file, personwiseKeypoints, keypoints_list):

    # 预测（推理）关键点
    def predict(self, imgfile):
```

Figure 11: The *core code of the predict .py is displayed*

```python
prototxt_25 = "models/pose/body_25/pose_deploy.prototxt"
caffemodel_25 = "models/pose/body_25/pose_iter_584000.caffemodel"

point_name_25 = ['None', 'Neck', 'RShoulder',
                 'RElbow', 'RWrist', 'LShoulder',
                 'LElbow', 'LWrist', 'MidHip',
                 'RHip', 'RKnee', 'RAnkle',
                 'LHip', 'LKnee', 'LAnkle',
                 'REye', 'LEye', 'REar',
                 'LEar', 'LBigToe', 'LSmallToe',
                 'LHeel', 'RBigToe', 'RSmallToe',
                 'RHeel']
point_pairs_25 = [[1, 8], [1, 2], [1, 5], [2, 3], [3, 4], [5, 6],
                  [6, 7], [8, 9], [9, 10], [10, 11], [8, 12], [12, 13],
                  [13, 14], [1, 0], [0, 15], [15, 17], [0, 16], [16, 18],
                  [2, 17], [5, 18], [14, 19], [19, 20], [14, 21], [11, 22],
                  [22, 23], [11, 24]]

map_idx_25 = [[26, 27], [40, 41], [48, 49], [42, 43], [44, 45], [50, 51],
              [52, 53], [32, 33], [28, 29], [30, 31], [34, 35], [36, 37],
              [38, 39], [56, 57], [58, 59], [62, 63], [60, 61], [64, 65],
              [46, 47], [54, 55], [66, 67], [68, 69], [70, 71], [72, 73],
              [74, 75], [76, 77]]

colors_25 = [[255, 0, 0], [255, 85, 0], [255, 170, 0],
             [255, 255, 0], [170, 255, 0], [85, 255, 0],
             [0, 255, 0], [0, 255, 85], [0, 255, 170],
             [0, 255, 255], [0, 170, 255], [0, 85, 255],
             [0, 0, 255], [85, 0, 255], [170, 0, 255],
             [255, 0, 255], [255, 0, 170], [255, 0, 85],
             [255, 170, 85], [255, 170, 170], [255, 170, 255],
             [255, 85, 85], [255, 85, 170], [255, 85, 255],
             [170, 170, 170]]

prototxt_18 = "./models/coco/pose_deploy_linevec.prototxt"
caffemodel_18 = "./models/coco/pose_iter_440000.caffemodel"
```

Figure 12: Part of the *code in the config file (including the joints*



that are aligned by the points).

Figure XIII: *Prototxt* file visualization (using the netscope platform).

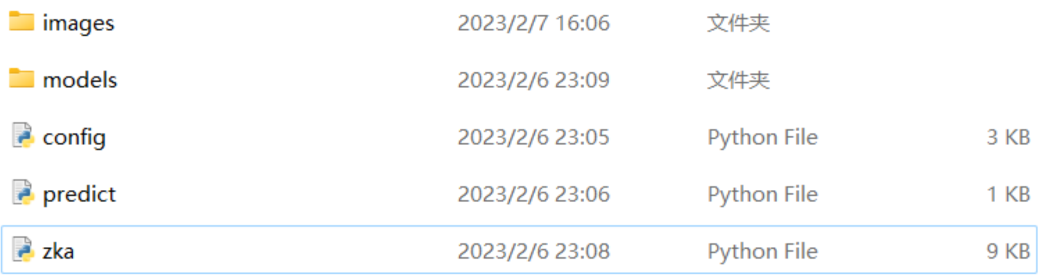


Figure 14: The outermost path is shown

3.5 Testing and Optimization

BASED ON OPENPOSE, TWO PRIMARY VERSIONS OF THE ALGORITHM WERE TRIED, NAMELY OPENPOSE-COCO18 AND OPENPOSE-BODY_25. OPENPOSE-COCO18 GRABS 18 BONE KEYS, WHILE OPENPOSE-BODY_25 SETS 25 KEYS. After deploying separately and forecasting, the following issues were found.

(1) The practical effect of OPENPOSE-COCO18 is mediocre. After testing, OPENPOSE-COCO18 can only fully stretch the limbs, face the camera, and make predictions for unobstructed single photos, which has relatively high limitations. In addition to the demonstration diagram provided by Figure 7 and its official explanation, many of the rest of

the situation have serious problems, the effect is not ideal, the recognition is low, for the motion assistance system, such a need to achieve partial occlusion, distortion of the situation, is unqualified.

Figure 23 OPENPOSE-COCO18 EFFECT (HAS BEEN RUN BUT NOT MARKED)

(2) OPENPOSE-COCO18 DOES NOT MODIFY THE SIZE OF THE PHOTO. It has been tested that when using the model, the output image is the same size as the input image, which results in the output image may be crowded together, and the result cannot be recognized, and it will not play a supporting role for fitness people.

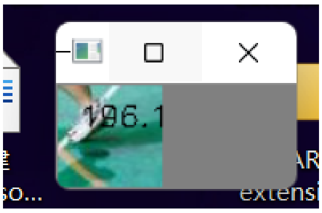


Figure 24 Model effect

The accuracy of the COCO18 test set is about 39.8%.

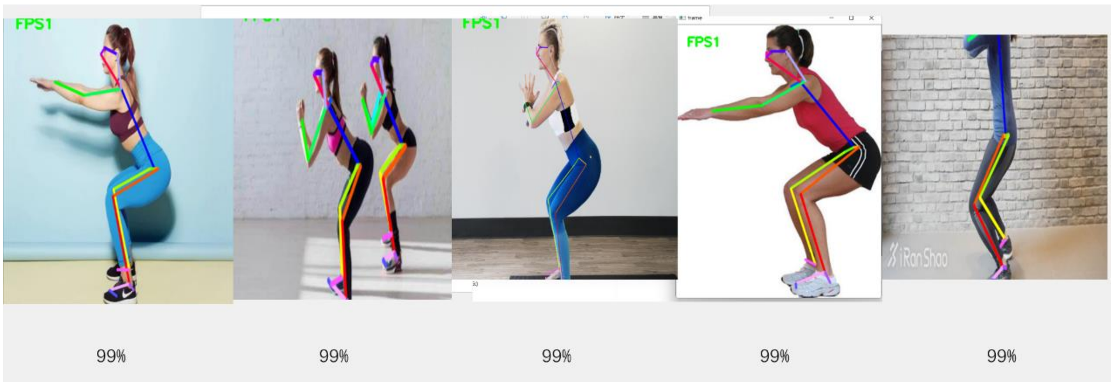


Figure BODY_25 test set has an accuracy rate of about 99%.

As mentioned above, the actual effect of OPENPOSE-COCO18 is not ideal, and it is speculated that it may be related to its too small model. At the same time, the size of the image should be exported without modification, resulting in unclear observation. IN THIS REGARD, THE PROPOSED SOLUTION IS TO ABANDON THE SIMPLER OPENPOSE-COCO18 AND SELECT THE RELATIVELY MORE ACCURATE OPENPOSE-BODY_25 AS THE CORE ARCHITECTURE OF THE MOTION ASSISTANCE SYSTEM, AND EXPAND IT ON THE BASIS OF OPENPOSE-BODY_25 TO GIVE IT THE ABILITY TO CORRECT ERRORS AND JUDGE THE STANDARD DEGREE OF POSTURE. PENPOSE-BODY_25 has been able to mark every key point that is needed.

4 Achievement display

4.1 Data entry function

Place the dataset used for testing in the images folder

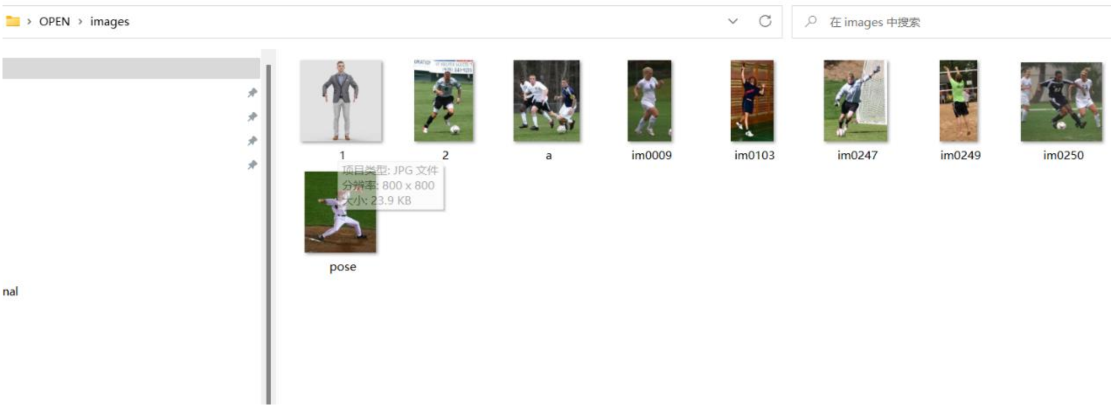
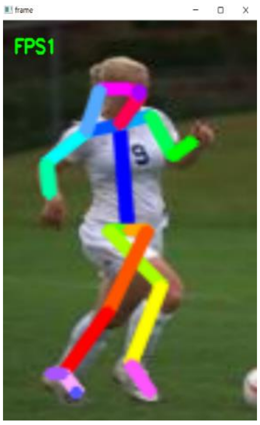


Figure 25: Dataset import

4.2 Bone rigging function

Double-click the zka .py

Figure 26: Photograph of the group members' riggings



Figs. 27 and 28: Rigging effects for single-player and multiplayer sports photos

4.3 Posture recognition function

C:\Windows\py.exe

```
[INFO]Time Taken in Forward pass: 2.5467476844787598
Keypoints - None : [(49, 31, 0.7863382)]
Keypoints - Neck : [(42, 43, 0.82401514)]
Keypoints - RShoulder : [(30, 46, 0.6958159)]
Keypoints - RElbow : [(16, 59, 0.77184814)]
Keypoints - RWrist : [(17, 72, 0.7270052)]
Keypoints - LShoulder : [(53, 39, 0.7927536)]
Keypoints - LElbow : [(61, 56, 0.6969919)]
Keypoints - LWrist : [(68, 50, 0.8239833)]
Keypoints - MidHip : [(45, 87, 0.63739425)]
Keypoints - RHip : [(38, 87, 0.63138723)]
Keypoints - RKnee : [(57, 110, 0.70225596)]
Keypoints - RAnkle : [(46, 143, 0.60189754)]
Keypoints - LHip : [(52, 87, 0.59757096)]
Keypoints - LKnee : [(38, 121, 0.7801058)]
Keypoints - LAnkle : [(22, 147, 0.6387138)]
Keypoints - REye : [(45, 28, 0.8212958)]
Keypoints - LEye : [(49, 28, 0.79237956)]
Keypoints - REar : [(35, 28, 0.7466217)]
Keypoints - LEar : []
Keypoints - LBigToe : [(27, 155, 0.45248455)]
Keypoints - LSmallToe : [(26, 155, 0.41696894)]
Keypoints - LHeel : [(17, 147, 0.5959119)]
Keypoints - RBigToe : [(53, 155, 0.59437376)]
Keypoints - RSmallToe : [(50, 154, 0.5096509)]
Keypoints - RHeel : [(42, 145, 0.49583942)]
No Connection : k = 17
No Connection : k = 19
```

Figure XXVII: Keypoints

The coordinates of the keys are represented here, and the code can be modified accordingly. (The code for coco18 is shown here with body_25 for comparison).

Here, a brief demonstration of the functions is presented.

(1) First, demonstrate its ability to correct errors.

Take the squat as an example to demonstrate its error-correcting ability. The key to the squat pose is that when viewed from the side, the knees should not be positioned beyond the position of the toes. Otherwise, after long-term exercise, the knee joint will be damaged

```
#print("Keypoints - {} : {}".format(self.point_names[part], keypoints))
if part==10:
    print("右膝盖：")
    print(keypoints[0][0])
    xa=int(keypoints[0][0])
```

```
if part==22:
    print("右脚大拇指:")
    print(keypoints[0][0])
    xb=int(keypoints[0][0])
```

```
if xa>=xb:
    print("正确")
else:
    print("脚尖超过膝盖，请收回脚尖")
```

and the habit of slumping will be formed. Therefore, simple compilation with python is required. Under the condition that the position relationship between the knee and the foot has been extracted, the effect can be achieved by modifying the following figures 28 and 29.

Figures 28 and 29: Modified code (partial) (body_25).

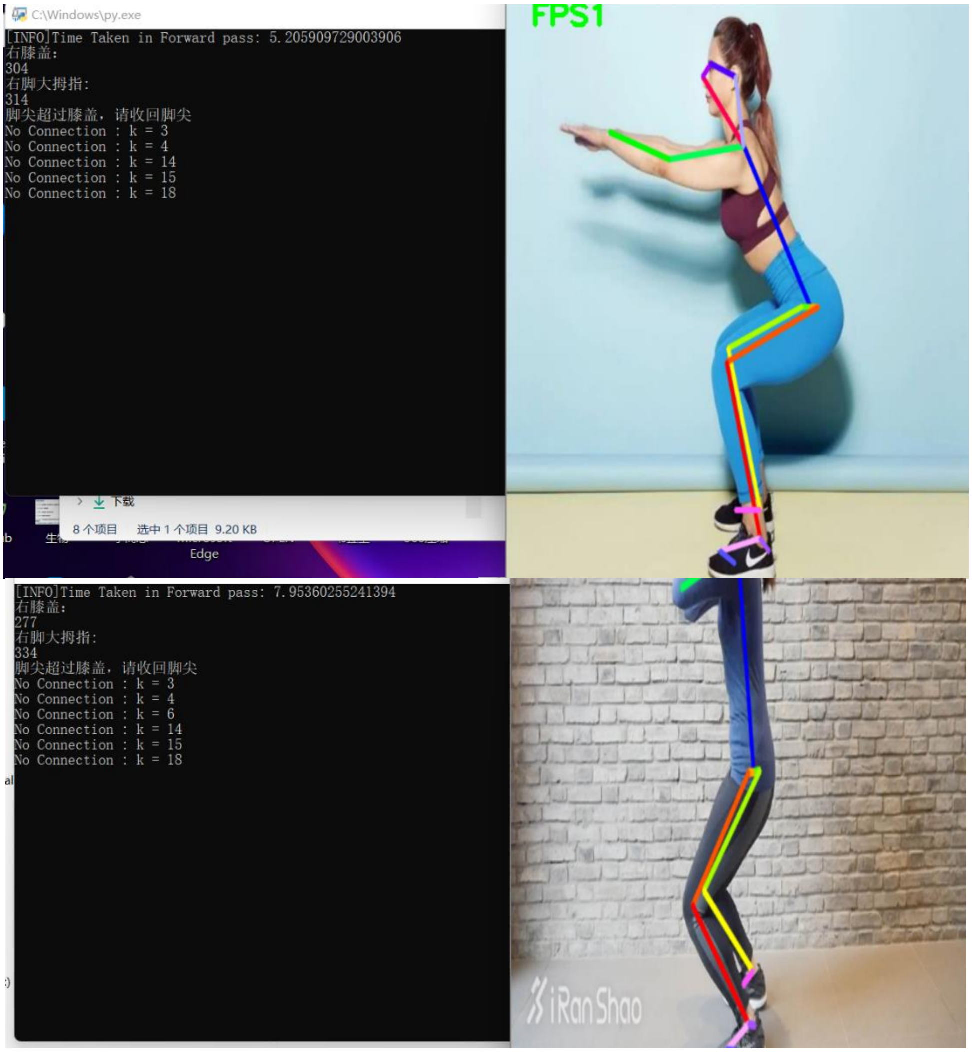
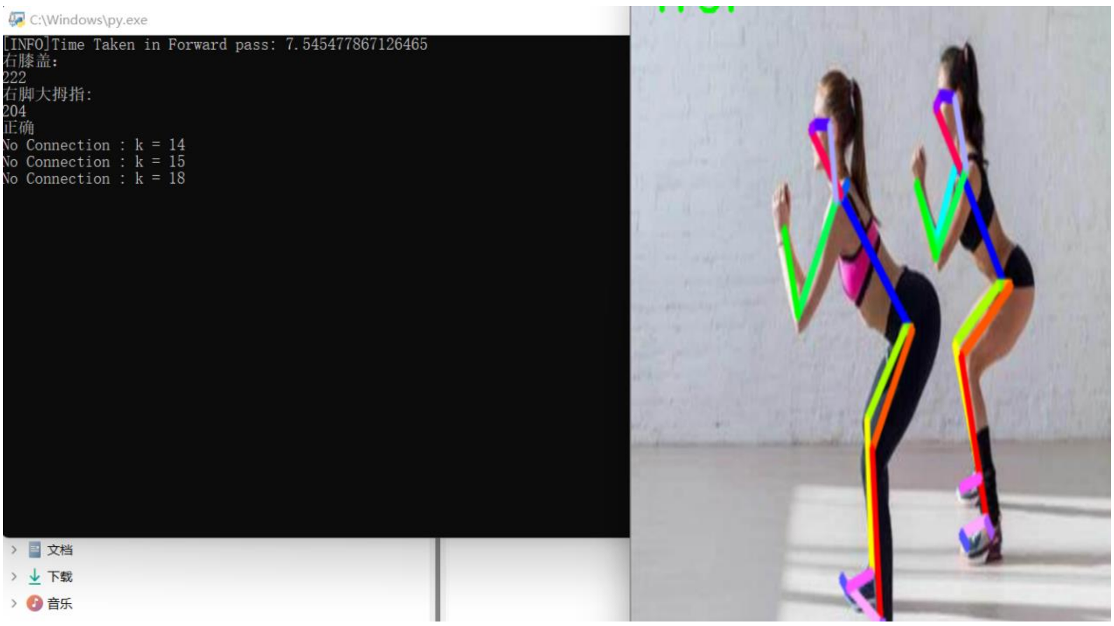


Figure: Successfully judged the wrong action



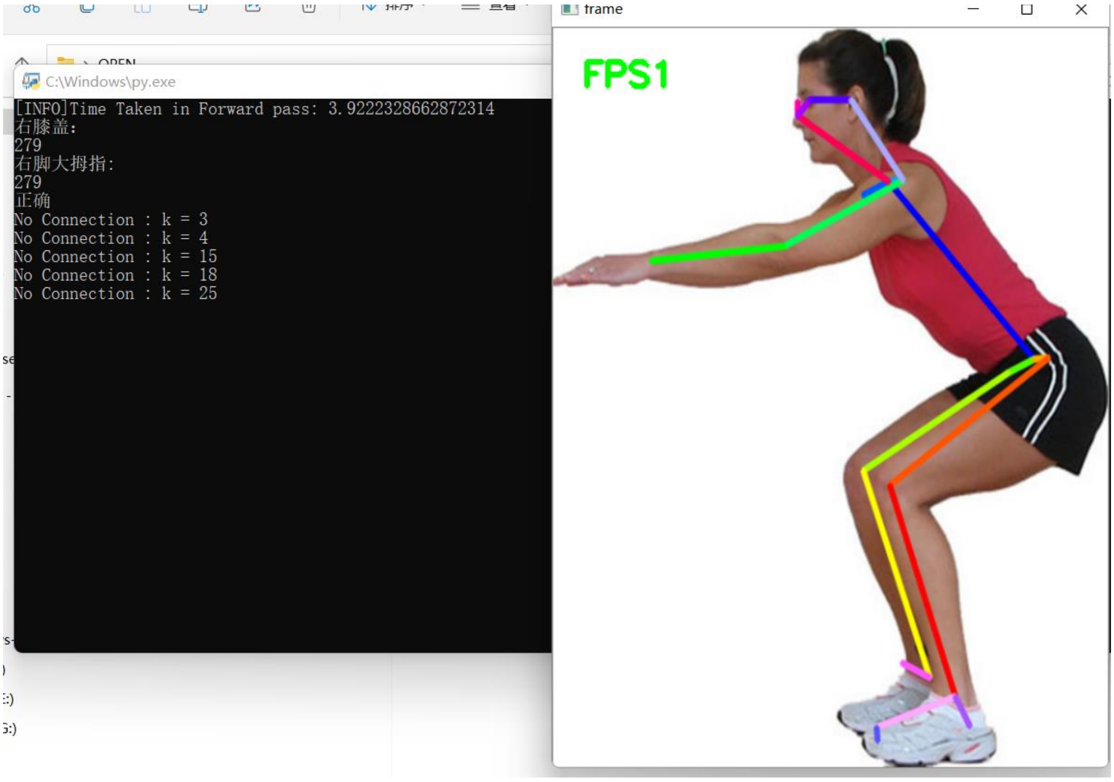


Figure Successfully judged the correct action

After judgment, 9 of the 10 photos of squats were correctly judged, and all 10 of them were correctly bound to the bones.

According to this idea, it can be realized to judge whether most sports movements are prone to injury and whether they will cause injury. You only need to adjust the parameters in the judgment statement and call the keypoints of different keypoints.

and (2) demonstrate their ability to judge the standard degree of posture.

Take pull-ups as an example to demonstrate their ability to judge the standard degree of posture.

```
#脖子

    heatMap = out[0, i, :, :]
    _, conf, _, point = cv.minMaxLoc(heatMap)
    x5 = (frameWidth * point[0]) / out.shape[3]
    y5 = (frameHeight * point[1]) / out.shape[2]


    i=1

#左肘

    heatMap = out[0, i, :, :]
    _, conf, _, point = cv.minMaxLoc(heatMap)
    x6 = (frameWidth * point[0]) / out.shape[3]
    y6 = (frameHeight * point[1]) / out.shape[2]


    i=3

if(y5>=y6):
    print("标准")
else:
    print("不标准")
```

Figs. 31 and 32 Modified code (partial) (coco18).

```
C:\Users\zka\Desktop>python openpose.py --input image.jpg
标准
```

Figure 33 Screenshot

The idea of implementing its functions is roughly the same as above. The reason for separating this function is that "non-standard" and "harmful" have different degrees and need to be differentiated and prompted. In addition, the system can also be used in sports standard testing equipment such as the sports high school entrance examination, assisting the human eye and infrared devices to better make real-time judgments.

5 Conclusions and prospects

5.1 Conclusion

After trial and optimization, the design of the motion assistance system was finally completed. Implemented bone rigging during motion, i.e., visualization of bones. At the same time, it can also make gesture recognition functions, including error correction and the ability to judge the standard degree of posture. THE SYSTEM USES OPENPOSE-BODY_25 AS THE CORE ALGORITHM (ACCURACY RATE IS CLOSE TO 99%), AND ITS

AVAILABILITY IS MUCH HIGHER THAN THAT OF OPENPOSE-COCO18 (ONLY 39.8% ACCURACY). In the era of national fitness, the research on sports assistance systems is bound to become a boom. The process of research is a process of efficient learning, in which trying to train immature models and making timely adjustments to deploy mature models is not only the re-cognition and learning of artificial intelligence, but also the exercise of thinking and judgment ability.

5.2 Outlook

In the end, due to technical and time reasons, there are still many parts of the system that need to be improved. The original idea was to be able to include as many movements as possible to complete the compilation of the entire assisted motion system. AT THE SAME TIME, THE SYSTEM IS EXPECTED TO BE IMPROVED IN THE FUTURE, AND THE BASIC ALGORITHM OF OPENPOSE-BODY_25 WILL BE ADJUSTED TO USE THE BLAZEPOSE MODEL WITH MORE NOVEL ACCURACY AND HIGHER SPEED. If there is a suitable opportunity, I also hope that this system can be used in "cloud fitness", or in sports standard testing, to play a role, reduce people's ideological burden on sports, and promote the further development of the era of national fitness.

5.3 Innovation

WHETHER IT IS OPENPOSE OR BLAZEPOSE, THE CURRENT ONLINE APPLICATION IS QUITE SIMPLE, MAINLY TO PROVIDE TECHNICAL SUPPORT FOR COMMERCIAL FITNESS SOFTWARE. However, due to the variety of shooting angles and the differences in human body structure, it has been difficult to complete the judgment of human movements and the training of motion models. In this optimization, in order to improve this problem, the method of comparing coordinates is adopted, and a fixed camera position is used to complete the judgment of the action standard. (e.g. judging whether the knee exceeds the toe during a squat). The biggest advantage of this method is that it is simple and fast, and the judgment process does not need to go through tedious training again, but uses the joint points that have been calculated in the bone drawing composition to complete the realization of the judgment function without affecting the running speed. The method can also have many extensions: such as adding more points to make the action standard judgment more perfect, becoming an independent action error correction system, or assisting the referee to complete the scoring ...... of the candidate's action in the sports testing projectAs long as the camera is fixed, there is room for it to play. This innovation opens up more possibilities for the application of the pose detection function.

References:

[1] Jiang M, Liu Shanqiu, Wang Yanyun. Hubei Sports Science and Technology,2022,41(03):217-219+257

[2] Wen Xu, Yang Yuxin, Gao Yuanlong, Li Kuan, Yang Jiandang, Yang Xiaoqian, Liu Yong. Non-contact measurement method of energy expenditure in aerobics based on OpenPose computer vision algorithm[J].Journal of Shanghai University of Sport,2022,46(08):23-32+55.)

[3] Tan Lixing, Lu Jiaqi, Zhang Xiaonan, Liu Yuhong, Zhang Rongfen. Gesture interaction system of phantom machine based on lightweight OpenPose[J].Computer Engineering and Applications,2021,57(16):159-166.)

[4] Yang Jun, Zhang Sujun, Zhang Chuanghao, Huang Jingjing. Sensors and Microsystems,2021,40(01):5-8.)

[5] Li Lin, Li Dalin, Lin Haohong . Research on Lion Dance Training Auxiliary System Based on OpenPose [J]. Journal of Television Technology,2022,46(9):37-41

[6] Lv Songjie, Qi Yuming, Deng Sanpeng, Xia Yuhong, Liu Hao. Research

on human action recognition algorithm based on 3D bone point data[J].Equipment Manufacturing Technology,2022,No.335(11):9-11+30.)
[7] Ma Ziyue, Peng Ruiyang, Sun Xiaohan, Wang Yuze, Li Xinyue, Kong Xiangyong. A review of human pose estimation technology based on OpenPose[J].Software Guide,2022,21(11):247-252.)