参赛队员姓名(Name of contestant): Mai Qiyue

中学(High School): Shanghai Fuxing Senior High School

省份(Province): Shanghai

国家/地区(Country/Region): China

指导教师姓名(Name of tutor): Bao Ming

指导教师单位(Tutor unit): Shanghai Fuxing Senior High School

论文题目(Paper Title): A Figure Skating Edge Detection System Based on MediaPipe

A Figure Skating Edges Detection System Based on MediaPipe

Mai Qiyue

Shanghai Fuxing Senior High School

Abstract

With the country's emphasis on sports and the arrival of the 2022 Beijing Winter Olympics, more and more people are getting to know and learn figure skating. Many movements in figure skating such as step sequences, jumps, and spins have prescribed body movements and requirements for using edges. However, when teaching, due to the fast skating speed or foggy field, coaches are unable to clearly determine whether students are using their edge correctly, which leads to problems such as teaching errors and low learning efficiency. Since the audience of the sport is still small, few effective solutions have appeared on the market. Therefore, this project hopes to show the use of edge during skating visually and clearly. This project is designed to install infrared distance sensors on the bottom of skates on both sides of the blade, collect data into the Arduino, and display the use of edge through calculation and analysis. And use MediaPipe human posture recognition technology to judge the skating action, and the action required to use the edge to compare, to achieve the effect of assisting skating teaching. After functional, accurate, and practical use experiments, this project can successfully meet the needs of the skating edge detection system.

Keywords: Figure Skating; MediaPipe; Arduino; Human Posture Recognition

Content

1 Introduction

1.1 Origin of the Problem

I remember during the 2018 PyeongChang Winter Olympics, I spent every day in front of the TV enjoying the athletes' performances, and I was attracted by one of the unique and beautiful events - figure skating. The skaters jumped, spun, and danced on the ice in their gorgeous costumes, just like elves. Since then, I have fallen in love with the sport of figure skating.

I have watched many videos of competitors' competitions, introductions to skating moves, and comments from skating enthusiasts and found that there is a great deal of interest in the use of the edge in skating.

A figure skate blade has two edges- inside and outside. The blade on the inside (between) your legs is called the inside edge, while the blade on the outside is called the outer edge. These two sharp precise edges are created by the hollow groove on the bottom of the blade [1]. The ISU also has strict rules and scoring criteria for the use of edge in skating, with 79 occurrences of "edge" and 10 occurrences of "blade" in the ISU rules [2]. This shows the importance of using the correct edge in skating.

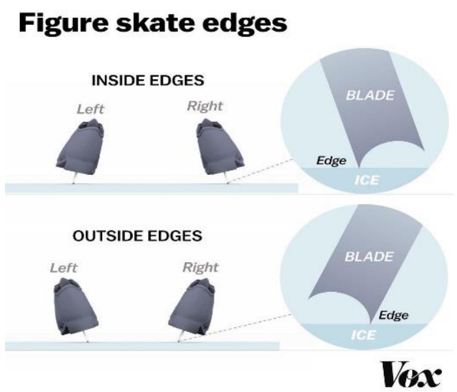


Fig. 1 Explanation of figure skate edges [3]

However, the problem of the wrong edge in daily teaching or competition field is still common, mainly due to the fast skating, jumping speed, or fogging ice surface, sometimes it is impossible to distinguish whether the correct edge is used, and timely correction is not made, which makes it difficult to correct the wrong action. At present, there are only methods such as video and slow motion, which are less costly but not time-sensitive, and the teaching effect is greatly reduced. Even, 'Since there is a physical limitation, judges can only see from one direction at a time, depending on the choreographers and skating instructors, some of whom program the jumps to be in the judges' blind spots. The take-off edge of the jumps cannot be distinguished in this way, which destroys the fairness of the game [4]', written by Yuzuru Hanyu in his paper 'A Feasibility Study on Utilization in Figure Skating by A Wireless Inertia Sensor Motion Capture System'.

Therefore, in order to solve the problem, I wanted to design a device that could detect whether the edge was used correctly in a particular action.

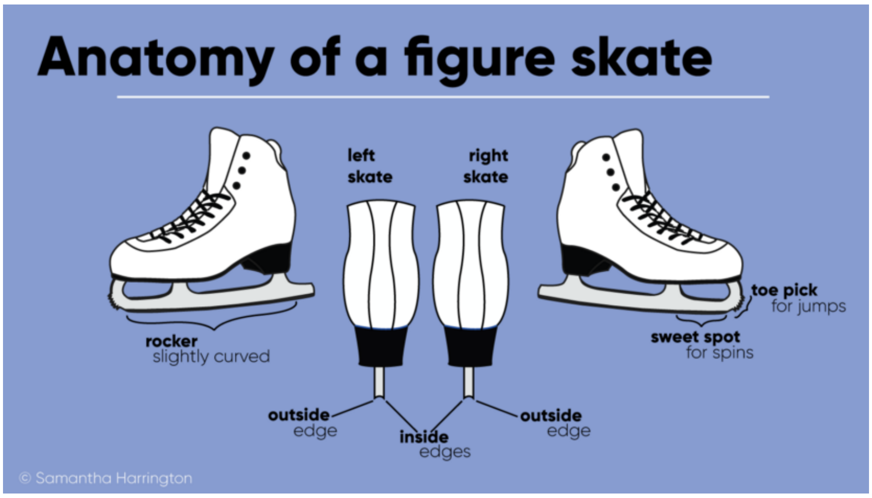


Fig. 2 Anatomy of a figure skate [5]

1.2 Solutions

To solve the problem, I divided this project into two main parts: the edge detection part and the motion detection part. The device needs to meet the conditions of being able to determine the skating edge in real-time, being able to determine the skating action in real-time, being able to transmit data wirelessly, and not being bound by the venue.

For the edge detection part, I have proposed two main solutions - one using a thin-film pressure sensor and the other using an infrared distance measurement sensor.

Solution 1.

In skating, the pressure distribution on the bottom of the foot varies with the different use of the blade. During my search, I found a medical device commonly used to diagnose plantar deformities, spinal disorders, and other problems - the plantar pressure testing system. Inspired by it, I designed the figure skating with edge testing system solution one:

First, install membrane pressure sensors on skate insoles. Second, acquire pressure data with Arduino. Third, transmit the data to the computer wirelessly with the Bluetooth module. Fourth, use the program to analyze the data, build a plantar distribution model and judge the use of the blade.

Solution 2.

By observing the shape of the ice skate blade when using different blades, it can be found that the distance between the front and back ends of the skate sole and the left and right sides from the ground is different when using different edges. However, in order to make the model more compact and lightweight, the design action is judged by the jumping motion as a reference, so the project only judges the inner and outer edges, not the front and back edges. Based on this, I designed the figure skating edge detection system plan 2:

First, install infrared distance sensors on both sides of the ice blade on the sole of the skate. Second, collect distance data with Arduino. Third, transmit the data to the computer wirelessly with the Bluetooth module. Fourth, use the program to analyze the data, compare the two distances and judge the use of the blade.

For the motion detection part, I designed to use Human posture recognition technology to help me complete the skating motion recognition. Currently, there are three main types of human pose recognition technologies. The first one is based on geometry, and through the recognition of the human body's key points, estimates the human skeleton to determine the action, such as MediaPipe, Alpha Pose, Open Pose, and Apple Vision. the second one is based on image features such as RGB, etc. through a large amount of data learning to determine the human pose, however, this method requires a lot of training, complex algorithm, and time consumption. The third one is to use a depth camera to build a 3D model to detect human pose, however, the depth camera used in this method is costly and not easy to install. Of course, the geometry-based approach also has its limitations such as the need for a better-differentiated background [6]. However, since the background is mainly white when skating, it is better to distinguish and does not affect the detection too much. Therefore, by comparison, I chose the first recognition method based on geometry, which is the easiest to implement, for human pose recognition.

By further programming, I can achieve the requirement of recognizing skating movements.

Finally, Excel was used to integrate the two parts of the data and compare them with the standard dynamic edge to make a judgment

Project Planning

July 15, 2021~August 15, 2021

Literature review, offline interviews, research route determination, preparation of experimental equipment and processing tools.

August 15, 2021~August 31, 2021

Learning Arduino, and experimenting with thin-film pressure sensors to obtain edge data.

September 1, 2021~October 15, 2021

Experiments on the acquisition of edge data by infrared range sensors and ultrasonic range sensors.

October 15, 2021~November 1, 2021

Learning, programming, and experimenting with MediaPipe key point recognition of skating moves.

November 15, 2021~December 15, 2021

Programming and experimentation of the system for determining the blade for skating.

Dec. 15, 2021 ~ Dec. 30, 2021

Commissioning and effect testing of MediaPipe-based figure skating edge detection system, finishing research log and report writing.

1.3 Innovation Points

The main innovation points of this project are that I cleverly combine Computer Science, Engineering, and figure skating. The MediaPipe I used as the technical framework for human pose recognition can run smoothly on smaller mobile devices such as laptops and cell phones. What’s more, except for the study of Yuzuru Hanyu, there are few relevant studies, so this project has a large market prospect.

2 Background and Requirements Research

2.1 Existing Topics or Product Research

After searching, I found that the market (Google, Taobao, Baidu, etc.) has no related or close products.

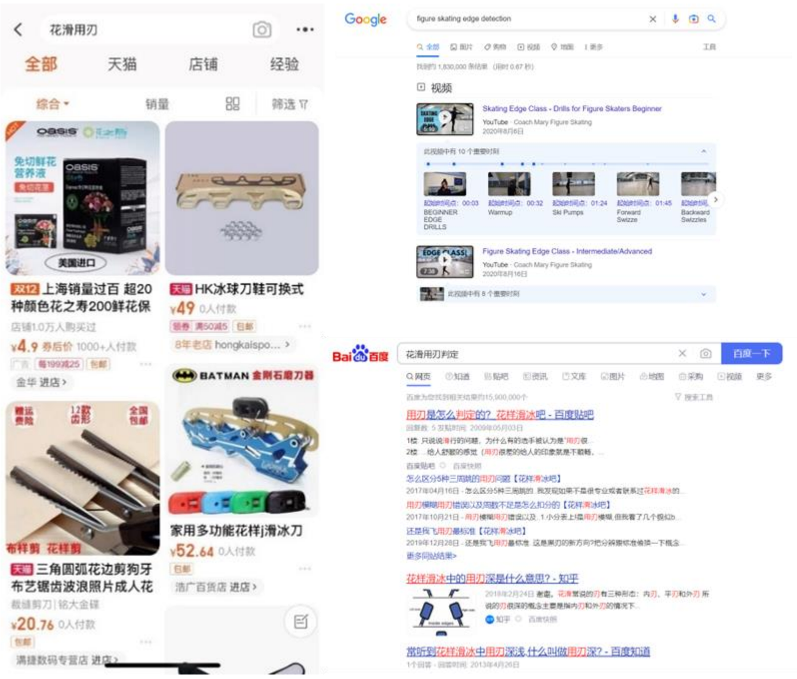


Fig. 3 Product research

The only related research I found was the paper ‘無線・慣性センサー式モーションキャプチャシステムのフィギュアスケートでの利活用に関するフィージビリティスタディ Feasibility study on the application of wireless inertial sensing type motion capture system to figure skating’ written by Yuzuru Hanyu.

Focusing on figure skating competitions, Yuzuru Hanyu questioned the fairness of judges' evaluation and scoring, pointed out the flaws in the existing skating scoring rules, and proposed the feasibility of introducing a wireless inertial sensor-based motion capture system to help judges. He wore a motion capture costume consisting of 31 sensors including angular velocity sensors and acceleration sensors. Data acquisition was mainly performed for three distinct kinds of jumps-the Loop jump, the Flip jump, and the Axel jump [4].

Hanyu's research has far-reaching implications for the fairness of figure skating competitions. However, professional motion capture systems are not suitable for use in teaching figure skating because of too many sensors, inconvenient assembly, and high cost.

Therefore, the research on this topic is innovative for the teaching of figure skating.

2.2 Survey Interviews

I conducted an offline interview to determine the use value of this design.

The following is a transcript of my interview.

Interview: Offline interview with MediaPipe-based figure skating with the blade detection system

Time: July 11, 2021

Location: Wujiaochang Bailian shopping mall Champion Ice Rink

Interviewee: Skating coach

Q: Hello, coach. What do you think is important to pay attention to in learning skating movements?

A: Skating is a bit like dancing, it requires many aspects of coordination. So, the upper body, lower body movements, use of edge, and so on are all more important.

Q: My topic focuses on the problem of incorrect edges. Do you see a high frequency of incorrect edges in your daily teaching?

A: There are definitely problems with wrong edges. In the footwork, there are mainly mistakes in the inner and outer edges, front and back. In jumping, the change of center of gravity can also lead to a wrong edge.

Q: Yes, thank you. Although I am not very good at skating, I usually find that the lutz and flip jumps are more prone to miss-edging when I watch a game.

A: The take-off of the flip jump begins from the back inside edge and one of the lutz jumpjumpsins from the back outside edge. The reason for the error is first of all the training time, followed by the training method, jumping method, while the game mentality is also quite influential.

Q: The jumping time is very short, does this hinder your judgment of the edge when teaching?

A: Yes, sometimes it's hard to judge the edge of the blade when the student is changing movements or dropping the ice very quickly.

Q: And how would you generally address this issue?

A: Generally, I will record a video to watch the replay

Q: Do you think it would be helpful to teach if there was a system that could judge students' use of the edge and compare it to the standard use of the edge?

A: If it doesn't affect the movement, or the skating, then it definitely helps.

Q: Yes, thank you very much for taking the time to answer.

Fig. 4 The offline interview

2.3 Summary

To summarize the above, this figure skating blade detection system needs to meet the following conditions. First, the use of edge and the skating action can both be determined in real-time. Second, data can be transmitted wirelessly. Third, the device should be easy to carry.

3 Prototyping

3.1 The Edge Detection Part

3.1.1 Design Process

I have designed two solutions for the edge detection part, one utilizing a thin-film pressure sensor and the other using an infrared distance measuring sensor.

Solution 1

The design of Solution 1 was inspired by the medical use of plantar pressure detection, which uses pressure sensors to detect the distribution of force on the bottom of the foot to analyze gait and body posture. I found that the way to shift between inside and outside edges is to slightly lean or change the center of gravity so that we can tell the use of edge by analyzing the plantar pressure distribution. Thus, I designed to install membrane pressure sensors on skate insoles, collect the pressure data, transmit the data to the computer wirelessly, and evaluate the use of edge.

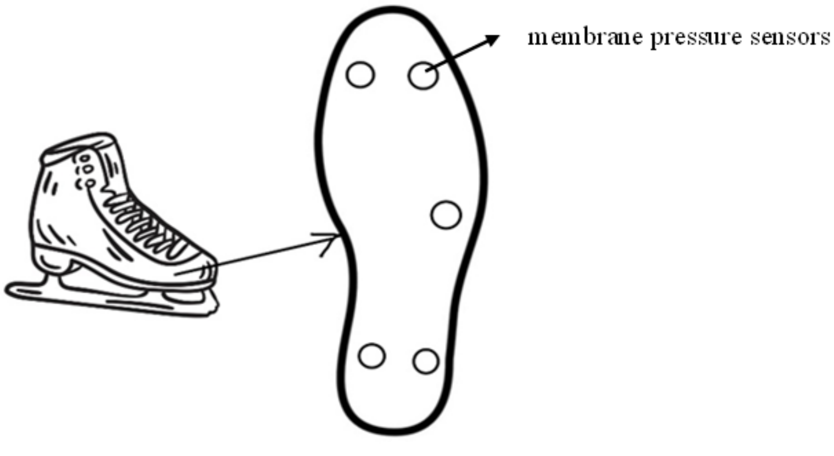


Fig. 5 Structure diagram of Solution 1

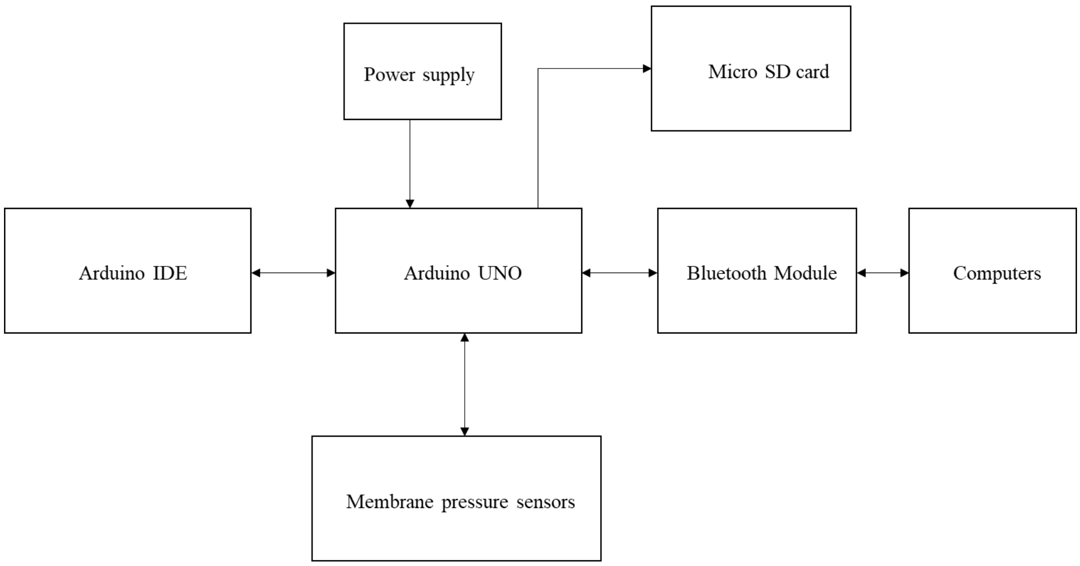


Fig. 6 Hardware block diagram of Solution 2

The hardware block diagram for this solution is shown above, the components are as follows:

Sensor: Membrane pressure sensors

Controller: Arduino UNO

Communication system: Bluetooth module

Power system: 5V lithium battery

Data storage device: MicroSD card, microSD card module

Solution 2

After further reflection, I found a simpler way to detect the use of edge. I noticed that when using different edges, the distance from the ice surface to both sides of the ice blade is different. Therefore, I designed to place infrared distance sensors on both sides of the ice blade on the skate's sole, collect the distance data, transmit the data to the computer wirelessly, and determine the use of edge by comparing the distances.

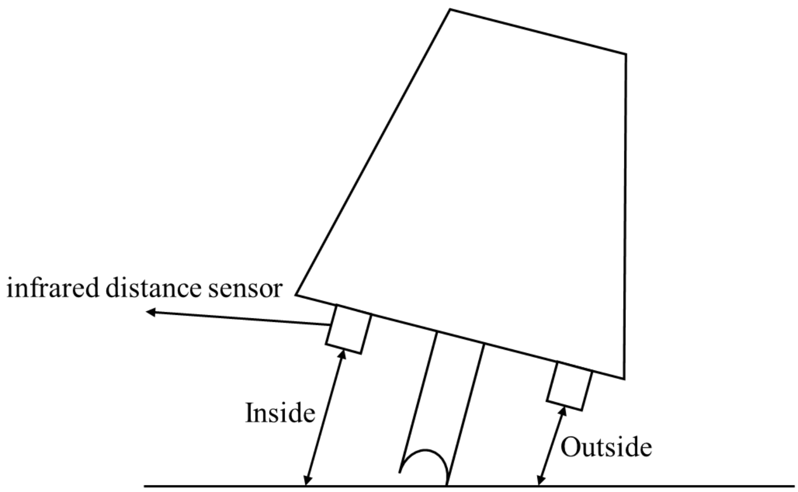


Fig. 7 Structure diagram of Solution 2 (Right foot)

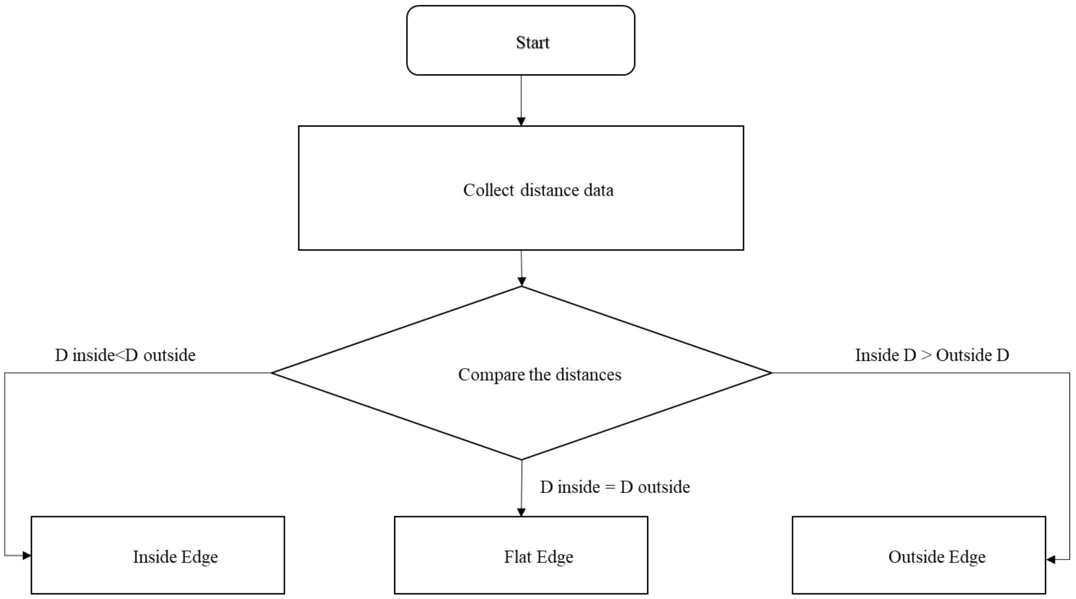


Fig. 8 Flow chart of Solution 2

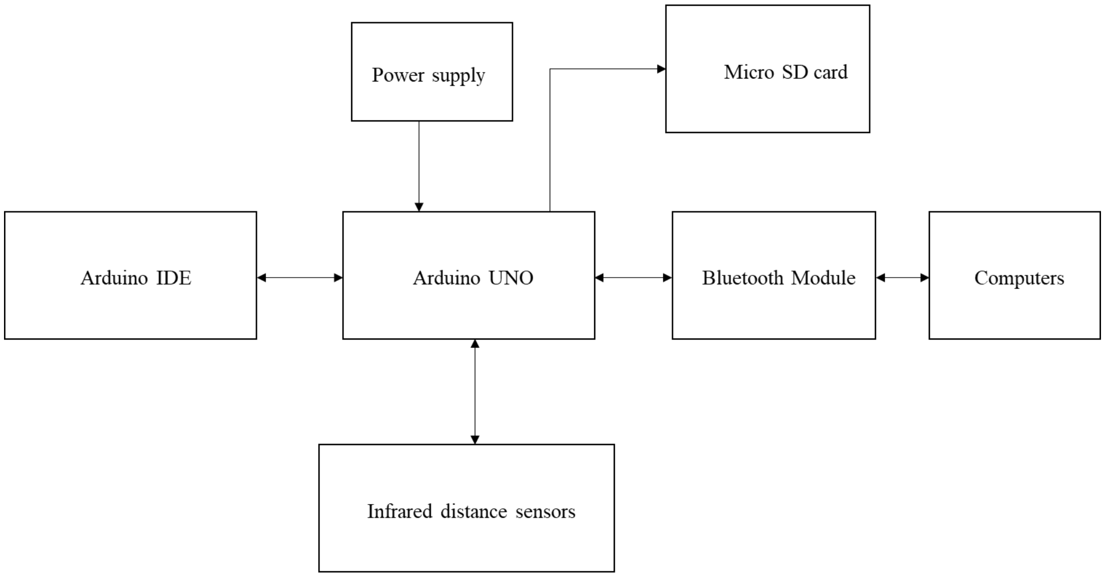


Fig. 9 Hardware block diagram of Solution 2

The hardware block diagram for this solution is shown above, the components are as follows:

Sensor: Infrared distance sensors

Controller: Arduino UNO

Communication system: Bluetooth module

Power system: 5V lithium battery

Data storage device: MicroSD card, microSD card module

3.1.2 Production Process

The components used in the production process are as follows:

Arduino UNO development board (64mm×54mm)

DuPont wires for connecting electronic components

Breadboard (85mm×55mm with 400 holes) for connecting circuits

Data transfer cable for transferring data between Arduino UNO and computer

MicroSD card and microSD card module for data storage

Bluetooth mode was utilized for measurements in order to connect with the PC wirelessly. The Bluetooth module is HC-06 (17mm×39mm). When using Bluetooth mode, a mobile battery is required, and the one used in this case was a 643960 2-port lithium battery with a capacity of 4000mAh. The size of it was approximately 64×45×17mm, the weight was approximately 80g, and the output was 2.75±0.3V/1A.

The development environment for the edge detection part is Arduino IDE, which is a development tool for Arduino, where developers can write and develop Arduino programs.

Solution 1

The membrane pressure sensor used in Solution 1 is IMS-C15A with 15mm diameter, and the measurement range is 500g-20kg.

During the production process, I first designed and connected the circuit for one single membrane pressure sensor and wrote a program to test whether the membrane pressure sensor works properly. Then I stuck five membrane pressure sensors on a shoe insole model separately at the top right, top left, bottom left, bottom right, and middle which the pressure can clearly show the use of edge. By comparing the pressure data the program can determine the use of edge.

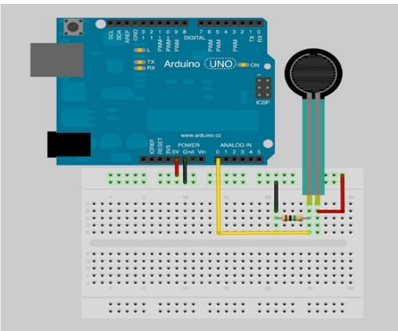


Fig. 10 Circuit connection diagram

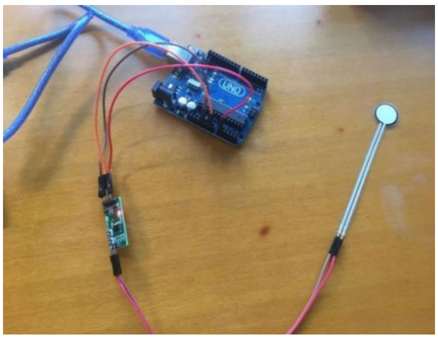


Fig. 11 Physical image

Fig. 12 Program for Solution 1

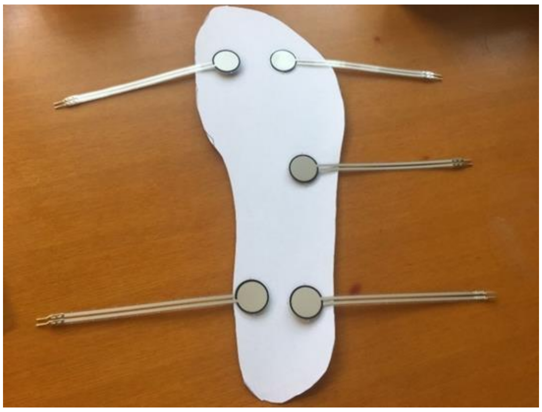


Fig. 13 Physical image

Solution 2

The infrared distance sensor used in Solution 2 is Sharp's GP2Y0A51SK0F. The measuring distance is 2.00-15.00cm, the size is 27.0×10.8×12.0mm, and the weight is approximately 2.7g, working at 5V. This infrared distance sensor can also measure the distance to moving objects and thus can be used to identify the edge while skating.

During the production process, I first designed and connected the circuit for one single infrared distance sensor and wrote a program to obtain its measurement data. Then I added the second sensor. The program can read and compare two distance data and determine the use of edge.

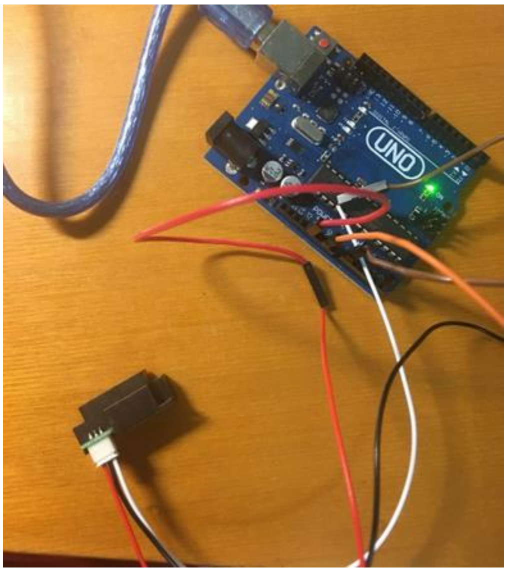


Fig. 14 Physical image

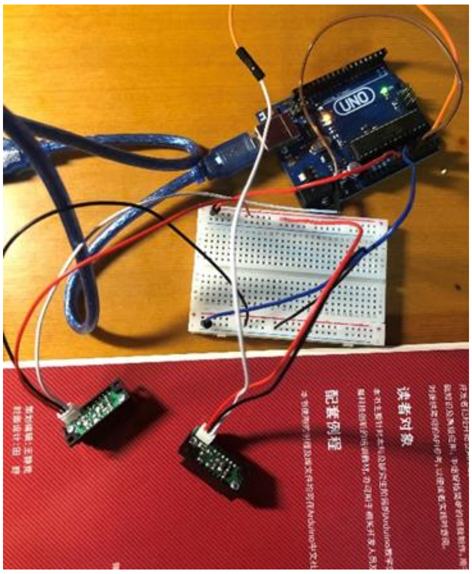


Fig. 15 Physical image

distancecode | Arduino 1.8.15

文件 编辑 项目 工具 帮助

distancecode

```
int sharpleft = A0;//把左红外测距传感器连接在模拟量端口0
int sharpright = A1;//把右红外测距传感器连接在模拟端口1
int valleft;
int valright;
int distanceleft;
int distanceright;
void setup(){
  Serial.begin(9600);
}
 void loop() {
valleft= analogRead(sharpleft);//读取红外测距传感器left模拟量数据
float valleft = analogRead(sharpleft)*0.0048828125;
float distanceleft = 65*pow(valleft, -1.10);    //将模拟值换算成距离
valright= analogRead(sharpright);//读取红外测距传感器right模拟量数据
float valright = analogRead(sharpright)*0.0048828125;
float distanceright = 65*pow(valright, -1.10);
if (distanceleft>distanceright)
{
  Serial.println("outside edge");
}
else if (distanceleft==distanceright)
{
  Serial.println("flat edge");
}
else if (distanceleft<distanceright)
{
  Serial.println("inside edge");
}
delay(500);//延时500ms
}
//本程序针对左脚
```

Fig. 16 Program for Solution

Wireless connection

To achieve wireless data transmission, I first used an SD card and an SD card module to store data. I also tried to complete the wireless connection between the computer and the physical object through Arduino's Bluetooth module library.

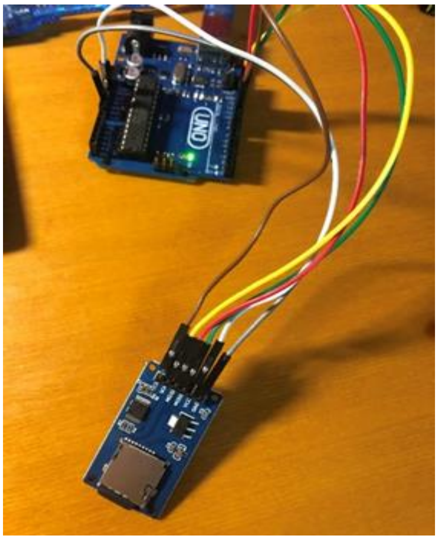


Fig. 17 Physical image of SD card

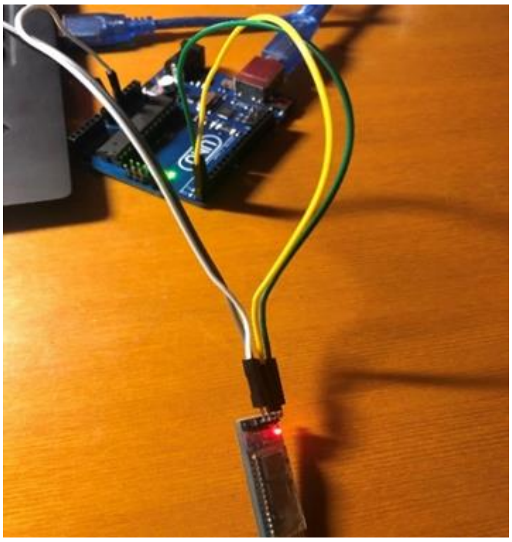


Fig. 18 Physical image of Bluetooth module

However, after experiments, I found that the Bluetooth connection is not stable enough, and the Bluetooth connection has a distance limitation, which is not suitable for larger skating venues. Therefore, I came up with the idea of using a Wi-Fi module to achieve data transmission.

I applied TCP communication in the Internet of Things (IoT) technology and chose the Bemfa Cloud platform. Bemfa Cloud IoT platform adopts the design idea of front and back-end separation and is committed to attacking high-performance asynchronous and concurrent IoT end servers to ensure the integrity of data under low power consumption [9].

Fig. 19 Bemfa Cloud platform

Connect to Wi-Fi via Arduino and transmit the data to the IoT platform. This way, real-time data can be obtained from any location with internet access. The computer subscribes to the IoT platform via python, and then the data of inside and outside edges can be obtained in real time.

```python

import paho.mqtt.client as mqtt

HOST = "bemfa.com"
PORT = 9501
client_id = "4d9ec352e0376f2110a0c601a2857225"

def on_connect(client, userdata, flags, rc):
    print("Connected with result code "+str(rc))
    client.subscribe("led00202")


def on_message(client, userdata, msg):
    print("主题:"+msg.topic+" 消息:"+str(msg.payload.decode('utf-8')))


def on_subscribe(client, userdata, mid, granted_qos):
    print("On Subscribed: qos = %d" % granted_qos)

def on_disconnect(client, userdata, rc):
    if rc != 0:
        print("Unexpected disconnection %s" % rc)


client = mqtt.Client(client_id)
client.username_pw_set("userName", "passwd")
client.on_connect = on_connect
client.on_message = on_message
client.on_subscribe = on_subscribe
client.on_disconnect = on_disconnect
client.connect(HOST, PORT, 60)
client.loop_forever()

```

Fig. 20 Program

The program will record the time when the skater made a wrong edge and the picture at that time so that it can be reviewed fairly later.

```python
if arduino_Force != pc_Force:
    font = cv2.FONT_HERSHEY_SIMPLEX
    cv2.putText(img, 'Force error', (10, 100), font, 0.5, (255, 255, 0), 2)
    now_time = datetime.datetime.now()
    cv2.imwrite('Force_error%s.jpg' %now_time ,img,[int(cv2.IMWRITE_JPEG_QUALITY),70])
```

Fig. 21 Program

3.1.3 Summary

By comparing the two solutions, I found that although Solution 1 can visualize the plantar pressure distribution, the program and wiring are more complicated and less accurate. At the same time, due to the strong pressure at the moment of landing, the pressure of the skater may exceed the range of the membrane pressure sensor, and the pressure cannot be measured accurately. Solution 2 is based on simple theory, simple program, and simple wiring. Therefore, I finally chose Solution 2 to determine the edge by infrared distance measurement sensor.

Finally, by assembling the individual materials on the skates, the prototype of the edge detection part is completed.

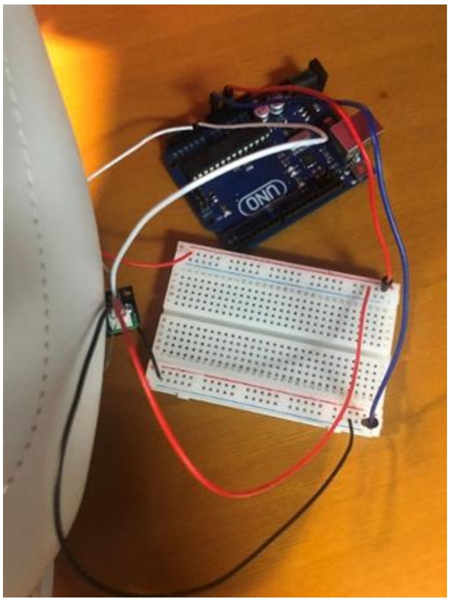


Fig. 22 Physical image

Fig. 23 Physical image of edge detection part

3.2 The Motion Detection Part

3.2.1 Design Process

I intended to apply Human posture recognition technology to assist in the motion detection part. Because of the complicated and time-consuming procedure of the big data method based on image features, and the 3D depth camera is costly and inconvenient to install. By comparison, I chose the simplest method of detecting movements by identifying key points of the human body, for example, MediaPipe, Open Pose, Apple Vision, and Alpha Pose.

Then I read through the papers and by comparing the data related to different models, I finally chose BlazePose used by MediaPipe as the technical framework for human pose recognition. Because MediaPipe has high recognition accuracy, recognition speed can reach 45 frames per second, can run smoothly on lightweight devices such as cell phones or laptops, and can accurately estimate when body parts are obscured.

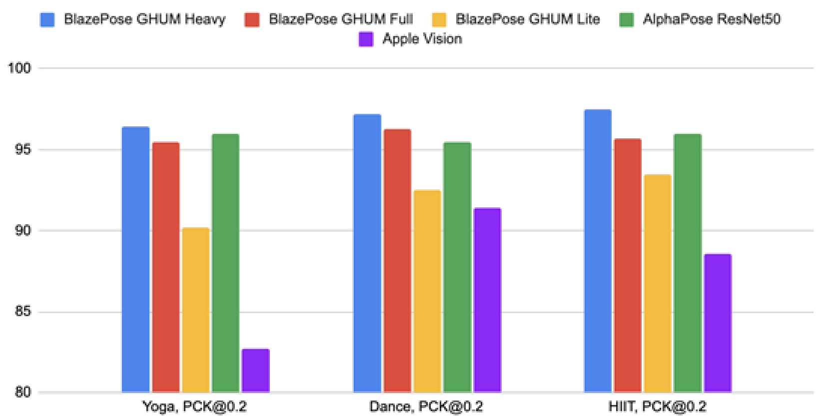


Fig. 24 Quality evaluation in PCK@0.2 [7]

3.2.2 Production Process

BlazePose used in MediaPipe is a lightweight convolutional neural network architecture designed for real-time human posture estimation on mobile devices. The network generates 33 body key points for a single individual during inference and runs at over 30 frames per second on a Pixel 2 phone. This makes it ideal for real-time applications such as fitness tracking and sign language recognition. [8]

PyCharm is used as the development environment to compile the program and detect the key points of the human body with the help of MediaPipe.

The specs of the PC used are as follows:

ThinkPad (LAPTOP-CUA4MQ7S)

Processor: Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz 1.80 GHz

Memory: 16 GB

I first tested the MediaPipe open-source program for key point recognition of static images and live dynamic people on camera respectively. It turned out that the program could identify the key point successfully.

Fig. 25 Test on static image

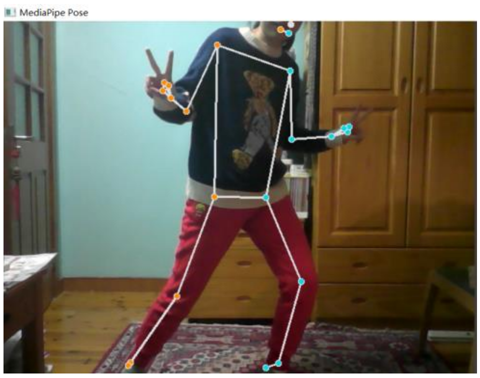
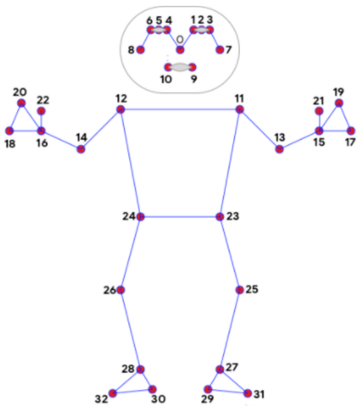


Fig. 26 Test on camera capture

```python
import cv2
import mediapipe as mp
mp_drawing = mp.solutions.drawing_utils
mp_drawing_styles = mp.solutions.drawing_styles
mp_pose = mp.solutions.pose

# For static images:
IMAGE_FILES = []
BG_COLOR = (192, 192, 192)_# gray
with mp_pose.Pose(
    static_image_mode=True,
    model_complexity=2,
    enable_segmentation=True,
    min_detection_confidence=0.5) as pose:
  for idx, file in enumerate(IMAGE_FILES):
    image = cv2.imread(file)
    image_height, image_width, _ = image.shape
    # Convert the BGR image to RGB before processing.
    results = pose.process(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))

    if not results.pose_landmarks:
      continue
    print(
        f'Nose coordinates: ('
        f'{results.pose_landmarks.landmark[mp_pose.PoseLandmark.NOSE].x * image_width}, '
        f'{results.pose_landmarks.landmark[mp_pose.PoseLandmark.NOSE].y * image_height})'
    )

    annotated_image = image.copy()
    # Draw segmentation on the image.
```

Fig. 27 Program

MediaPipe employs a two-step detector-tracker machine learning process. The pipeline first uses a detector to find the person/pose region-of-interest (ROI) inside the frame. Using the ROI-cropped frame as input, the tracker then predicts the posture landmarks and segmentation mask inside the ROI. Then, a MediaPipe graph will be presented by using a pose landmark subgraph from the pose landmark module and renders using a dedicated pose renderer subgraph. The landmark model in MediaPipe Pose predicts the location of 33 pose landmarks. [7]



0. nose
1. left_eye_inner
2. left_eye
3. left_eye_outer
4. right_eye_inner
5. right_eye
6. right_eye_outer
7. left_ear
8. right_ear
9. mouth_left
10. mouth_right
11. left_shoulder
12. right_shoulder
13. left_elbow
14. right_elbow
15. left_wrist
16. right_wrist
17. left_pinky
18. right_pinky
19. left_index
20. right_index
21. left_thumb
22. right_thumb
23. left_hip
24. right_hip
25. left_knee
26. right_knee
27. left_ankle
28. right_ankle
29. left_heel
30. right_heel
31. left_foot_index
32. right_foot_index

Fig. 28 33 pose landmarks [7]

In order to identify a particular movement, first, by observation, I found the physical features of different movements, for example, Axel jump is the only jump that jumps forward and starts with an angle between the legs of less than 90 degrees. Then I found the key points needed to represent the angle from 33 key points and extracted the key point coordinates. Finally, the angle size is obtained by vector operation.

Suppose that the three key points I choose are P_i, P_j, P_k (i, j, and k are representing the pose landmarks from 0 to 32).

$$P_i(x_i, y_i),\ P_j(x_j, y_j), P_k(x_k, y_k)$$

Suppose that θ is the angle formed by $\overrightarrow{P_iP_j}$ and $\overrightarrow{P_iP_k}$

$$\theta = \arccos\left(\frac{\overrightarrow{P_iP_j} \cdot \overrightarrow{P_iP_k}}{\left|\overrightarrow{P_iP_j}\right| \cdot \left|\overrightarrow{P_iP_k}\right|}\right)$$

$$\theta = \arccos\left(\frac{x_i^2 + y_i^2 - x_ix_j - y_iy_j - x_ix_k - y_iy_k + x_jx_k + y_jy_k}{\sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \cdot \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}}\right)$$

The program can calculate the angle of the key points to determine the motion.

3.3 Integration

After finishing the edge detection part and the motion detection part, I integrated two parts. As this figure skating edge detection system consists of two parts, to ensure that each movement can accurately match the use of edge I set a timeline. When the figure skater starts skating, both parts will start recording at the same time, and the data will be written in an Excel sheet. Each motion, such as jump, corresponds to a standard edge. Therefore, by comparing the actual use of the edge with the standard use of the edge, it is possible to determine whether the edge is used correctly for this motion.

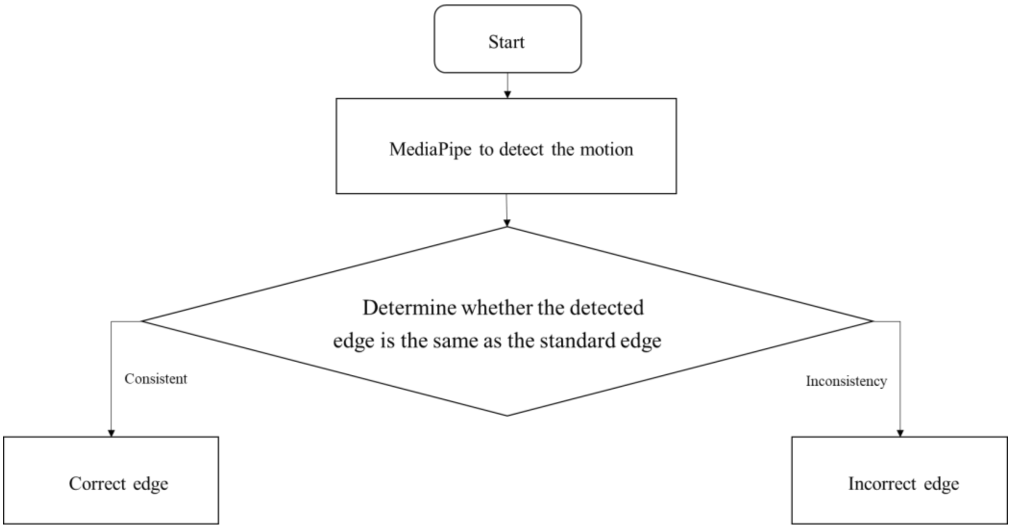


Fig. 29 Flow chart

4 Experiment

4.1 Experimental Design

To verify the feasibility of the project, I tested the project in three aspects: functionality, accuracy, and practical use. These include: verifying whether the skating edge can be accurately judged, verifying whether the skating action can be determined, and verifying whether the edge is used correctly under a given action, etc.

4.2 Functional Test

4.2.1 Experiment 1

Experiment purpose: To test whether the edge can be identified

Experimental materials: Ice skates, the edge detection part

Experimental object: The edge detection part

Experimental method: Change the edge by swinging the skates with hands, observe the edge judgment on the window monitor, and record the data to compare with the real edge.

Table 1 Experiment 1

Index	Real edge	Detected edge	Is it accurate (+ for yes, - for no)
1	Inside Edge	Inside Edge	+
2	Inside Edge	Inside Edge	+
3	Inside Edge	Inside Edge	+
4	Outside Edge	Outside Edge	+
5	Outside Edge	Outside Edge	+
6	Outside Edge	Outside Edge	+
Accuracy	100%		

Experiment summary: The device can accurately detect the use of edge.

4.2.2 Experiment 2

Experimental purpose: To test whether the skating motion can be identified

Experimental materials: Computer, skating images (for easy judgment, all use jump pictures), the motion detection part

Experimental object: The motion detection part

Experimental method: Use the program in the motion detection part to perform human posture recognition on pictures of known skating actions, determine the actions, and record the data.

Table 2 Experiment 2

Index	Real motion	Detected motion	Is it accurate (+ for yes, - for no)
1	T	T	+
2	T	F	-
3	S	S	+
4	Lo	Lo	+
5	F	F	+
6	F	F	+
7	Lz	Lz	+
8	A	A	+
Accuracy		87.5%	

Experimental summary: The device can accurately detect figure skating motion.

4.2.3 Experiment 3

Experimental purpose: Verify whether the edge is used correctly under a given action

Experimental materials: Computer, skating images (for easy judgment, all use jump pictures), the prototype

Experimental object: Figure skating edge detection system

Experimental method: Change the edge by swinging the skates with hands, observe the edge judgment on the window monitor, and record the data in Excel. Use the program in the motion detection part to perform human posture recognition on pictures of known skating actions, determine the actions, and record the data in Excel. Combine the two data, compare it with the correct blade under the corresponding skating action, and verify whether the use of edge is correct. (The full names and abbreviations of the jumps are shown in the appendix)

Table 3 Experiment 3

Index	Real motion	Real edge	Detected motion	Detected edge	Correct combination	Judgment results(+ for True, - for False)	Is it accurate (+ for yes, - for no)
1	A	Left Outside edge	A	Left Outside edge	A+Left Outside edge	+	+
2	F	Left Inside edge	F	Left Inside edge	F+Left Inside edge	+	+
3	L	Right Outside edge	L	Right Outside edge	L+Right Outside edge	+	+
4	Lz	Left Outside edge	Lz	Left Outside edge	Lz+Left Outside edge	+	+
5	F	Left Outside edge	F	Left Outside edge	F+Left Inside edge	-	+
6	Lz	Left Inside edge	Lz	Left Inside edge	Lz+Left Outside edge	-	+
Accuracy				100%			

Experiment summary: The device can accurately determine whether the edge is used correctly under a given action

Conclusion: After the experiment, each function can be realized as expected, which proves the feasibility of the device.

4.3 Accuracy Test

4.3.1 Experiment 4

Experimental purpose: To verify whether the infrared distance measurement sensor can successfully measure the distance to the ice surface in the skating rink environment

Experimental method: Place the infrared distance sensor above the ice surface, observe the distance data on the serial monitor, and record

Expected results: Infrared distance measuring device can accurately measure the distance from the ice surface

Experimental equipment: Infrared distance sensors, ice

Variable: Distance from the ice surface

Experimental steps:

1. Hold the Infrared distance sensor above the ice surface and change the distance
2. Observe serial monitor data and record

Table 4 Experiment 4

Index	1	2	3	4	5	6	Success rate	Precision
Measurement data (unit: cm)	1.45	2.39	4.50	5.96	6.49	3.40	100%	0.01cm
Is it accurate (+ for yes, - for no)	+	+	+	+	+	+		

Experimental results: The infrared distance sensor can accurately measure the distance from the ice surface at the low temperature and humidity in the ice rink.

4.3.2 Experiment 5

Experimental purpose: To test whether the range and precision of the infrared distance sensor meet the needs of detection

Experimental method: Calculate the required range and precision with the edge detection device, compare product information and experimental data

Experimental equipment: Infrared distance sensors, ice skates

Experimental steps:

1. Use mathematical modeling to find the theoretical range

2. Collect infrared range sensor data and combine it with product information to determine

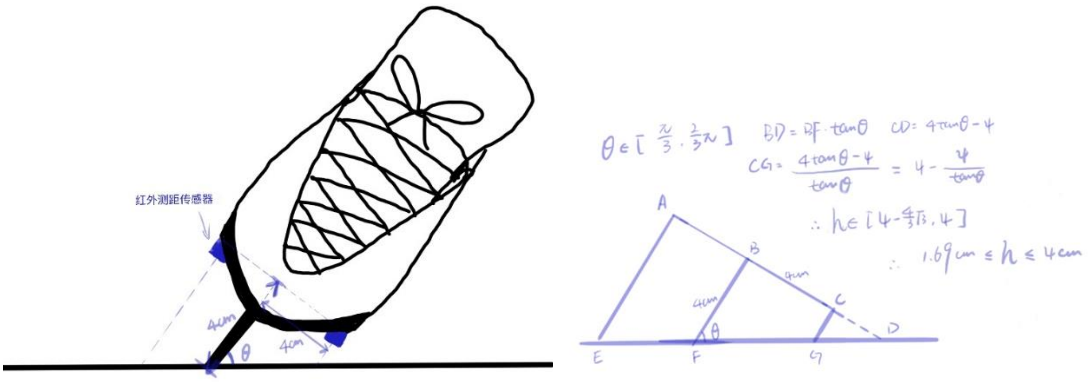


Fig. 30 Mathematical modeling

Data analysis:

The theoretical range of distance is 1.69cm-4.00cm, and the accuracy should be 0.01cm. According to the product information, the range of the infrared distance sensor is 1-15cm, the accuracy is 0.01cm.

Experimental results: The range and precision of the infrared distance sensor meet the needs of edge detection

4.4 Practical Test

4.4.1 Experiment 6

Experimental purpose: To test whether the figure skating edge detection system can be put into practical use

Experimental materials: MediaPipe-based figure skating edge detection system

Experimental method: Wear this figure skating edge detection device and test its use at the ice rink. For the reason that most technical movements in figure skating can only be done by professional skaters, I did some simple steps such as Cross roll and Ina bauer as test motions

Experimental Interviews:

I conducted brief interviews with users and recorded feedback from different users, and the feedback is as follows.

1. User A: I feel that the installation of the device is not stable enough, and I am afraid that the device will fall down when skating. the Arduino UNO and Bluetooth module can only be fixed on the side of the skates, and the wiring is also hidden at the bottom of the skates, which does not affect skating but is still not very neat. Since the computer only has a front camera, it is very inconvenient to operate when tracking and shooting.
2. Figure skating coach: Although only judging the inside and outside edges can determine whether the motion is correct, many movements also require the front and back edges, thus the judgment of front and back edges can be added to the detection of the edge.

Experimental results: The subject can detect the edge and the motion in real time and wireless data transmission can be achieved properly. Combining two parts, the edge detection system can accurately judge whether the edge is used correctly under a specific motion. Of course, in the process, I also found some aspects that need to be improved. First, the installation of the device is not solid enough. This can be improved by fixing it with hot melt adhesive or waterproof silicone tape. Second, all devices including Arduino and Bluetooth modules are mounted on the side of the skate and the wiring is exposed and not neat enough. In the future, the device can be replaced with a smaller Arduino NANO or integrated circuit, or the device can be modified to be embedded. Third, since the computer only has a front-facing camera, it is not convenient to operate when tracking the skater. In the future, I can connect an external camera, or make the program a mobile app for easy use. Fourth, the edge detection part can be more complete, the membrane pressure sensor may be used to identify the front and back edges.

5 Summary

5.1 Conclusions

Firstly, the use of the edge is a very important element in figure skating, all correct jumps, spins, and step sequences require the correct edge. Secondly, the figure skating edge detection system based on MediaPipe combines figure skating, Computer science, and Engineering, which can accurately identify the skater's edge and motions with over 90% probability, wirelessly transmit the data through Wi-Fi, and determine whether the edge is correct in a specific motion, which can greatly improve the efficiency of figure skating teaching. Since the audience for figure skating is relatively small, there is almost no research related to edge detection for figure skating teaching, so this project is innovative. Third, I tested the system in three aspects: feasibility, accuracy, and practical use, and gained valuable feedback from the users to clarify the future improvement plan.

5.2 Future Outlook

According to the feedback from the experiment, there are three main aspects of the subject that need to be improved: reinforcing the installation, improving the way of shooting, and improving the judgment with the edge. Meanwhile, in the future, big data learning can be performed on the data of different motions to achieve a better recognition effect.

6 Acknowledgement

This project stems from my love of figure skating. It is also the rigor of this sport's pursuit of standard movements that motivates me to find problems in what I love and care about and try to solve them. During the whole practical process of the project, experienced many setbacks. The first time I contacted Arduino, I knew nothing about it, so I bought an introductory tutorial to teach myself. Step by step, I tried to figure out and try; I had never worked on human posture recognition before, so I used internet resources and got a preliminary understanding and application of the technology with the help of my tutor. Although there were many obstacles in the way of scientific research, I have grown and gained a lot from trial and error.

During the research process, the strict requirements for accuracy, and the multi-dimensional experimental design have made me appreciate the rigor of scientific research. The continuous exploration also allowed me to discover more shortcomings of the project, which await my further improvement and optimization.

The production of the project has greatly improved my scientific literacy, self-learning ability, and essay writing ability, and also allowed me to learn Arduino, Python programming, and Excel.

I would like to thank teacher Ming Bao (uncompensated) for offering valuable guidance and help throughout the development of my project. He helped me with the project's direction, guided me through my experiment, and advised me on how to improve this article.

Finally, I hope that figure skating will be brought to the attention of more people, and that more projects will emerge that combine sports with other fields such as Engineering and Computer science.

7 References

[1] Chang, Emily. “FIGURESKATINGETC: My Skating Blog.” FigureskatingETC, FigureskatingETC, 1 Nov. 2019, https://www.figureskatingetc.com/new-blog/2018/11/14/what-is-an-edge-and-why-is-it-important.

[2] Constitution & Regulations. Constitution & Regulations - International Skating Union. (n.d.). Retrieved November 12, 2022, from https://www.isu.org/inside-isu/rules-regulations/isu-statutes-constitution-regulations-technical.

[3] Zarracina, J. (2018). Explanation of figure skate edges. Winter Olympics 2018: figure skating jumps, explained. Retrieved from https://www.vox.com/culture/2018/2/8/16919618/winter-olympics-2018-skating-jumps.

[4] Hanyu, Y. (2021, March 18). 無線・慣性センサー式モーションキャプチャシステムのフィギュアスケートでの利活用に関するフィージビリティスタディ (thesis). 人間科学研究 = Waseda Journal of Human Sciences.

[5] Harrington, S. (2017). Anatomy of a figure skate. What is figure skating? Retrieved from https://medium.com/edge-crunch/what-is-figure-skating-ac0f456fd4a4.

[6] Hua, M., Gao, M., & Zhong, Z. (2021). Scn: dilated silhouette convolutional network for video action recognition. Computer aided geometric design, 85, 101965.

[7] Pose Estimation Quality. (2020). MediaPipe Pose. Retrieved from https://google.github.io/mediapipe/solutions/pose.

[8] Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang, F., & Grundmann, M. (2020). BlazePose: On-device Real-time Body Pose tracking (thesis). Google Research, California.

[9] 巴法科技&巴法云-巴法设备云-巴法物联网云平台. (n.d.). Retrieved November 25, 2022, from https://cloud.bemfa.com/

[10] Technical panel handbook - isu.org. (n.d.). Retrieved November 24, 2022, from https://www.isu.org/figure-skating/rules/sandp-handbooks-faq/24781-tphb-single-skating-2020-21-final/file

[11] Wikimedia Foundation. (2022, November 16). Figure skating jumps. Wikipedia. Retrieved November 25, 2022, from https://en.wikipedia.org/wiki/Figure_skating_jumps

8 Appendix

ISU abbreviations	
1Eu	Euler jump
T	Toe loop
F	Flip
Lz	Lutz
S	Salchow
Lo	Loop
A	Axel

Fig. 31 Figure skating jumps [10]

Abbr.	Jump	Toe assist	Change of foot	Change of edge	Change of curve	Change of direction	Takeoff edge	Landing edge
A	Axel	—	✓	—	—	✓	***Forward*** outside	Outside (opposite foot)
Lz	Lutz	✓	✓	—	✓	—	Backward outside	Outside (opposite foot)
F	Flip	✓	✓	✓	—	—	Backward ***inside***	Outside (opposite foot)
Lo	Loop	—	—	—	—	—	Backward outside	Outside (***same foot***)
S	Salchow	—	✓	✓	—	—	Backward ***inside***	Outside (opposite foot)
T	Toe loop	✓	—	—	—	—	Backward outside	Outside (***same foot***)
Eu	Euler (half-loop)	—	✓	✓	—	—	Backward outside	***Inside*** (opposite foot)

Fig. 32 Classification and distinction of basic figure skating jumps [11]