

参赛队员姓名：麦祺悦

中学：上海市复兴高级中学

省份：上海市

国家/地区：中国

指导教师姓名：包鸣

指导教师单位：上海市复兴高级中学

论文题目：《基于 MediaPipe 的花样

滑冰用刃检测系统》

2022 S.-T. Yau High School Science Awards
仅用于2022年成桐中学科学奖公示

基于 MediaPipe 的花样滑冰用刃检测系统

【摘要】随着国家对体育的重视，2022 年北京冬奥会的到来，越来越多人开始认识、学习花样滑冰（下简称花滑）这项运动。花滑中许多动作如步法、跳跃、旋转等都有规定的肢体动作和用刃要求。然而，在教学时，由于滑行速度快或场地起雾等原因，教练无法清晰地判断学生的用刃是否正确，从而导致教学出错、学习效率低等问题。由于该运动受众仍较小，目前市面上还没有出现有效的解决方案。因此本课题希望能将花滑时的用刃情况直观清晰地展现。本课题设计在冰鞋鞋底，冰刀两端安装红外测距传感器，采集数据进入 Arduino 主控中，通过计算分析显示用刃情况。并利用 MediaPipe 关键点识别技术判断花滑动作，与动作所要求的用刃进行比对，达到辅助花滑教学的效果。经过功能性、精准度（量程：1-15cm 精度：0.01cm）及实际使用实验后，本课题可以成功达到花滑用刃检测系统的需求。

【关键词】花样滑冰；辅助训练；检测系统；MediaPipe；Arduino；红外测距；人体姿态识别

仅用于2022丘成桐中学科学奖
2022 S.-T. Yau High School Science Awards

目录

1. 绪论引言	1
1.1. 问题的由来	1
1.2. 解决方案	2
1.3. 课题意义与创新点	2
1.4. 课题规划	3
2. 背景与需求调研	4
2.1. 现有课题或产品研究	4
2.2. 调查访谈	4
2.2.1. 线下访谈	4
2.3. 需求调研总结	6
3. 原型设计	7
3.1. 概念图	7
3.1.1. 灵感来源	7
3.1.2. 概念结构图	7
3.2. 流程图	9
3.2.1. 逻辑流程图与解释	9
3.3. 硬件框图	10
3.3.1. 硬件框图与解释	10
4. 原型设计	12
4.1. 设计过程	12
4.1.1. 开发环境	12
4.2. 项目原型	12
4.2.1. 开发过程	12
4.2.2. 项目原型	20
5. 实验设计与测试分析	21
5.1. 实验设计	21
5.2. 功能性测试	21
5.2.1. 实验一	21
5.2.2. 实验二	22
5.2.3. 实验三	23
5.3. 精准度测试	24
5.3.1. 实验四	24
5.3.2. 实验五	25
5.4. 实际使用测试	26
5.4.1. 实验六	26

5.4.2. 使用访谈.....	26
5.5. 结论概述.....	27
6. 总结.....	28
6.1. 课题成果.....	28
6.2. 未来展望.....	28
7. 收获与成长.....	30
8. 致谢.....	31
9. 参考文献.....	32
10. 附录.....	33
10.1. 相关代码.....	33
10.2. 花滑动作标准.....	1

仅用于2022丘成桐中学科学奖公示
2022 S.-T. Yau High School Science Awards

1. 绪论引言

1.1. 问题的由来

记得在 2018 年平昌冬奥会期间，我每天都会守在电视机前欣赏运动健儿们拼搏的身姿，而我被其中一个独特又美丽的项目所吸引——花样滑冰。选手们穿着华美的服装在冰面上跳跃、旋转、起舞，宛如精灵一般。至此，我就爱上了花滑这个运动。

我观看了许多选手的比赛视频、花滑的动作介绍、花滑爱好者们的评论，发现对花滑用刃的关注度是非常高的。花滑中的刃指：在一条曲线上的一只脚上可见的溜冰痕迹。花滑冰刀刀刃分为内刃和外刃，不同的动作都有不同的用刃标准。ISU（国际滑联）也对花滑中的用刃做了严格的规定与评分标准（在 ISU 的规则中“edge”出现 79 次，“blade”出现 10 次）。可见花滑中用刃正确的重要性。

然而，在日常教学或比赛场上错刃的问题依旧常见，主要原因是由于滑行、起跳速度快，或冰面起雾，有时无法分辨用刃是否正确，未进行及时的纠正，导致错误动作难以改正。目前只有录像与慢动作等方法，虽然成本较低，却没有时效性，教学效果也大打折扣。甚至，由于无法准确判断，有些花滑选手对错误的用刃视而不见，破坏了比赛的公平性。

因此，为了解决问题，我开始了这个课题的研究。

Death Spirals

f) Only the prescribed death spiral is permitted.

Death Spiral backward outside:

Both partners are skating on a **backward outside edge**. The Man performs a pivot and holds the hand of the Lady with the same arm as his skating foot fully extended. The Lady is leaning backwards to the ice and her arm is fully extended as she circles around the Man in this position. Any variation of the Man's position, skating direction or **edge** is allowed as long as he keeps the pivot position as described and the Lady circles around him on an **outside edge**.

116

Death Spiral forward inside:

The Man is skating on a **backward outside edge**, and the Lady is skating on a **forward inside edge**. The Man performs a pivot and holds the hand of the Lady with the same arm fully extended as his skating foot. The Lady is leaning sideways to the ice and her arm is also fully extended as she circles around the Man in this position. Any variation of the Man's position, skating direction or **edge** is allowed as long as he keeps the pivot position as described and the Lady circles around him on an **inside edge**.

Death Spiral backward inside:

The Lady is leaning towards the ice and her arm is fully extended and she circles around the Man on a **firm backward inside edge**.

Death Spiral forward outside:

The same as for the death spiral forward inside, except that the Lady circles around the Man on a **firm forward outside edge**.

图 1-3 ISU 规则中对用刃的提及

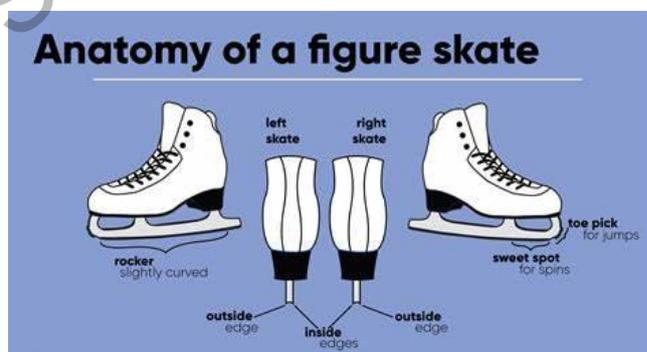


图 1-2 花滑用刃

Figure skate edges

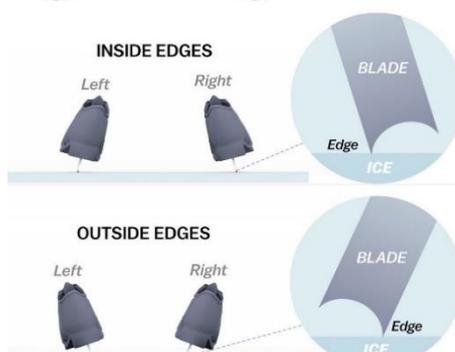


图 1-1 用刃解释

1.2. 解决方案

为解决问题，我将本课题主要分为两部分：用刃检测部分和动作检测部分。且该装置需满足能实时判定花滑用刃，能实时判定花滑动作，数据能无线传输，不受场地束缚的条件。

对于用刃检测部分，由于外部的解决方案如录像回放和慢动作等无法获得及时、准确的回应，所以我将目光放在了花滑冰鞋上。

花滑冰鞋冰刃部分太窄，无法进行设计

方案一：

花滑中，不同的用刃脚底的压力分布也会不同。在搜索过程中，我发现了一款常用于诊断足底畸形、脊椎疾病等问题的医疗器械——足底压力测试系统。受其启发，我设计了花样滑冰用刃检测系统方案一。

- ①在花滑鞋垫上安装薄膜压力传感器
- ②以 Arduino 作为主控采集压力数据
- ③软件系统无线处理数据，判定花滑用刃
- ④利用关键点识别技术判断花滑动作，与动作所要求的用刃进行比对

方案二：

通过观察不同用刃时冰刀刀刃的形态可发现：不同用刃时，冰鞋鞋底前后两端与左右两侧距地面的距离是不同的。但为了使模型更加精简轻便且设计动作判断是以起跳动作为参照的，所以本课题只判断内外用刃，不判断前后用刃。基于此，我设计了花样滑冰用刃检测系统方案二。

- ①在花滑冰鞋鞋底冰刀两侧分别安装红外测距传感器
- ②以 Arduino 作为主控采集距离数据
- ③软件系统无线处理数据，判定花滑用刃
- ④利用关键点识别技术判断花滑动作，与动作所要求的用刃进行比对

对于动作检测部分，我设计利用网上已有的开源人体关键点识别项目来帮助我完成花滑动作的识别。目前网络上能找到的有百度人体关键点识别、Openpose 和 MediaPipe 关键点识别。通过进一步编程达到识别花滑动作的需求。

最后用 Excel 将两部分数据整合与标准动作用刃比对，做出判断

1.3. 课题意义与创新点

- 1.将工程与花滑巧妙地融合
2. MediaPipe 在笔记本的 cpu 配置中就能顺畅运行，且支持手机等更小的移

动设备

3.有较大市场前景

1.4. 课题规划

①2021年7月15日~2021年8月15日

文献查阅、线下访谈、研究路线确定、实验装备和加工工具准备

②2021年8月15日~2021年8月31日

学习 Arduino, 对利用薄膜压力传感器获取用刃数据的实验

③2021年9月1日~2021年10月15日

对红外测距传感器和超声测距传感器获取用刃数据的实验

④2021年10月15日~2021年11月1日

对 MediaPipe 关键点识别花滑动作的学习、编程和实验

⑤2021年11月15日~2021年12月15日

对判定花滑用刃的系统的编程及实验

⑥2021年12月15日~2021年12月30日

基于 MediaPipe 的花样滑冰用刃检测系统的调试和效果检测、研究日志、报告撰写

2. 背景与需求调研

2.1. 现有课题或产品研究

经过搜索，市面上（淘宝、百度）没有相关或接近产品



图 2-2 淘宝搜索



图 2-1 网络搜索

相关研究只能搜索到相近的关于跳跃落冰所受冲击力的影响

2.2. 调查访谈

2.2.1. 线下访谈

为了确定本设计的使用价值和设计方向，我进行了一项线下访谈。

访谈：基于 MediaPipe 的花样滑冰用刃检测系统的线下访谈

时间：2021 年 7 月 11 日

地点：五角场百联又一城冠军溜冰场

采访对象：花滑教练

采访内容：

Q: 教练您好。请问您认为花滑动作学习中要注意哪些方面?

A: 花滑有点像跳舞,是需要多方面的配合的。所以,上肢、下肢动作、用刃等等都是比较重要的。

Q: 我的课题主要关注的时错刃方面的问题,您在日常教学中,学生发生错刃问题的频率高吗?

A: 错刃问题肯定是有的。在步法中的话主要有内外刃、前外和后外等出错。跳跃中因为重心变化也会导致错刃。

Q: 好的,谢谢。虽然我不太会滑冰,但是平时看比赛时会发现后外点冰跳(luzt跳)和后内点冰跳(flip跳)是比较容易发生错刃的,能不能请您以专业的视角解释一下原因?

A: 后外点冰跳起跳是右后外刃,后内点冰起跳是左前内起跳,一般后内点冰错刃会少一点,后外点冰跳错刃多。出错原因首先是训练时间,其次是训练方式、起跳方法,同时比赛时的心态也是挺有影响的。

Q: 跳跃起跳时间很短,这会不会对您教学时对用刃的判断造成妨碍?

A: 也会啊,有的时候学生在动作转换或者落冰这种很快的动作时是很难判断用刃的。

Q: 那一般会如何解决这个问题呢?

A: 一般就会录视频看回放

Q: 如果有一个系统可以对学生用刃进行判断,并和标准用刃做对比,你认为这对教学会有帮助吗?

A: 如果不会对动作、滑行造成影响,那肯定是有帮助的。

Q: 好的,非常感谢您能抽出时间回答。



图 2-3 访谈记录

2.3. 需求调研总结

总结以上，该花样滑冰用刃检测系统需满足以下条件：

- 1.能实时判定花滑用刃
- 2.能实时判定花滑动作
- 3.数据能无线传输，不受场地束缚
- 4.装置简单易携带

仅用于2022丘成桐中学科学奖公示
2022 S.-T. Yau High School Science Awards

3. 原型设计

3.1. 概念图

3.1.1. 灵感来源

方案一的设计灵感来源于医学上使用的足底压力检测,通过压力传感器检测足底受力分布从而分析出步态及体态。



图 3-1 足底压力检测仪

3.1.2. 概念结构图

方案一:

- ①在花滑鞋垫上安装薄膜压力传感器
- ②以 Arduino 作为主控采集压力数据
- ③软件系统无线处理数据, 判定花滑用刃

④利用关键点识别技术判断花滑动作，与动作所要求的用刃进行比对

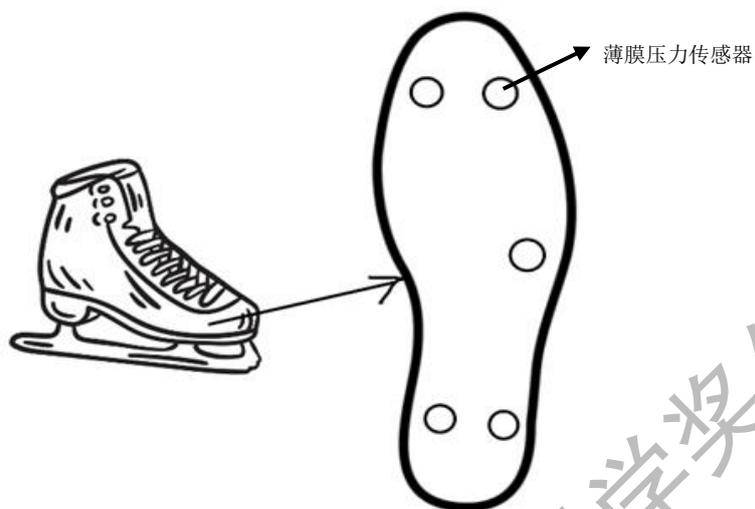


图 3-2 方案一概念结构图

方案二:

①在花滑冰鞋鞋底冰刀两侧分别安装红外测距传感器

②以 Arduino 作为主控采集距离数据

③软件系统无线处理数据，判定花滑用刃

④利用关键点识别技术判断花滑动作，与动作所要求的用刃进行比对

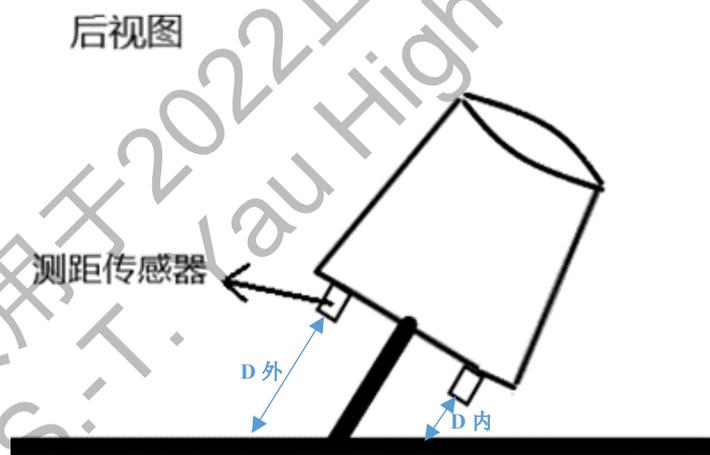


图 3-3 方案二概念结构图

3.2. 流程图

3.2.1. 逻辑流程图与解释

用刀检测部分

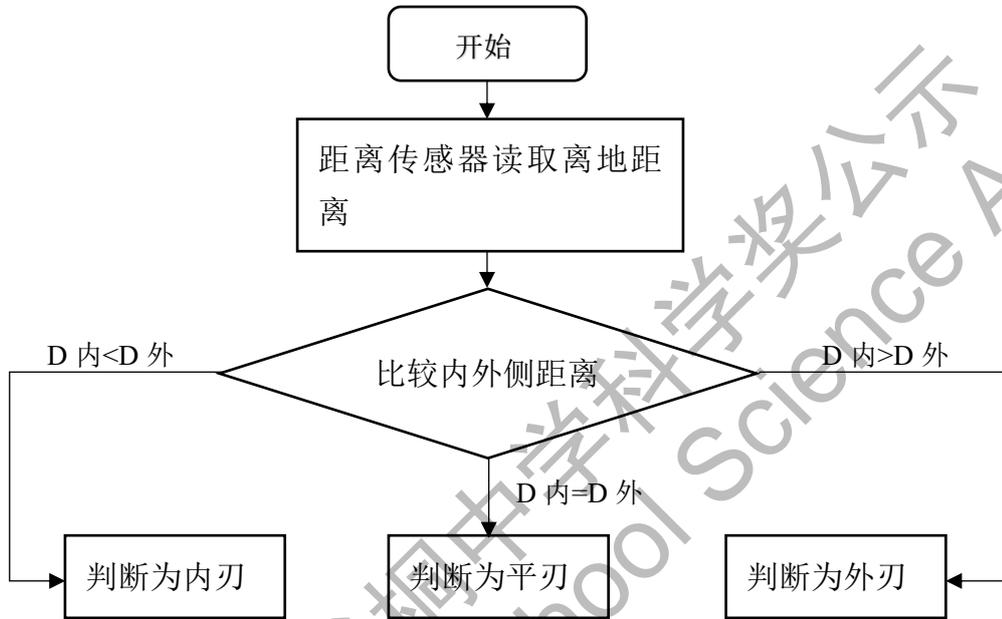


图 3-4

动作检测部分

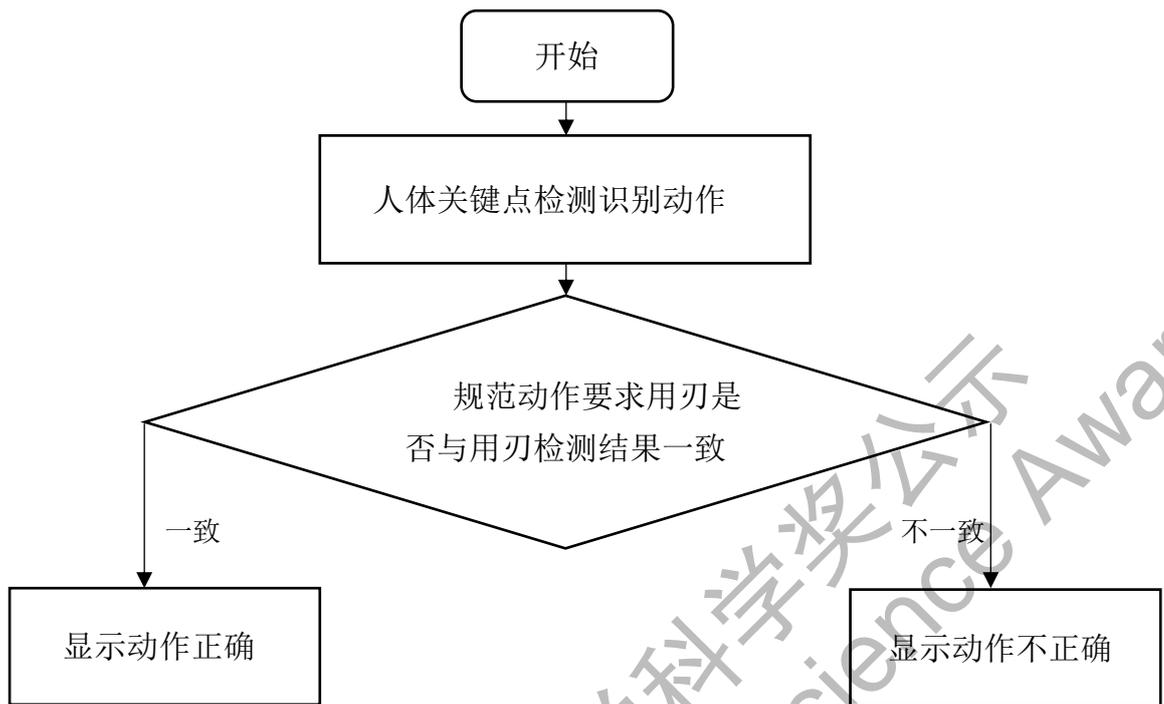


图 3-5

3.3. 硬件框图

3.3.1. 硬件框图与解释

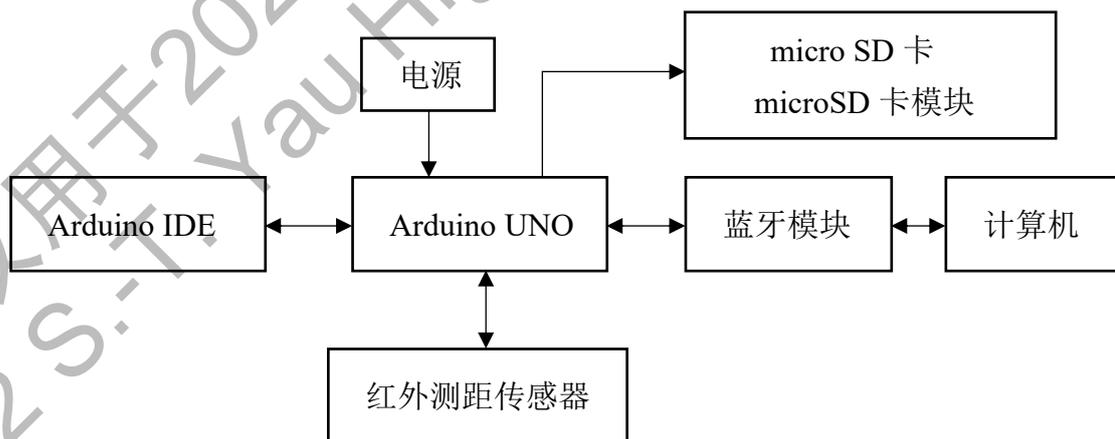


图 3-6

此方案的硬件框图如上，其中的自动化元素有：

传感器：红外测距传感器

控制器：Arduino UNO

通讯系统：蓝牙模块

电源系统：5V 锂电池

数据存储设备：micro SD 卡，microSD 卡模块

仅用于2022丘成桐中学科学奖公示
2022 S.-T. Yau High School Science Awards

4. 原型设计

4.1. 设计过程

4.1.1. 开发环境

本课题用刃检测部分采用 Arduino IDE 进行程序编译。Arduino IDE 是针对 Arduino 的开发工具，主要用于编写和开发 Arduino 程序。只需在 IDE 中编写程序，将程序上传到 Arduino 板，Arduino 就会运行相应程序。

本课题动作识别部分采用 PyCharm 进行程序编译，借助谷歌人体姿态识别技术检测人体关键点。

4.2. 项目原型

4.2.1. 开发过程

本课题用到的材料有：

- 1.Arduino UNO，作为硬件主板
- 2.跳线/杜邦线/面包板，用于连接电子元器件
- 3.数据传输线，用于向 Arduino 主板传输数据
- 4.HC-06 蓝牙模块，用于实现数据无线传输
- 5.microSD 卡，用于数据存储
- 6.microSD 卡模块，用于读写数据
- 7.5V 锂电池，用于供电

用刃检测部分

方案一：

- 1.薄膜压力传感器，用于测量足底压力分布
- 2.电压转换模块，用于将电阻信号转换为电压信号

制作过程如下：

- 1.设计并连接电路

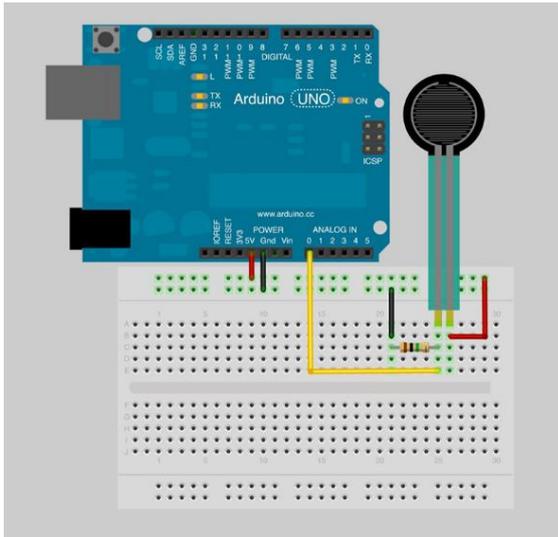


图 4-2 电路连接图

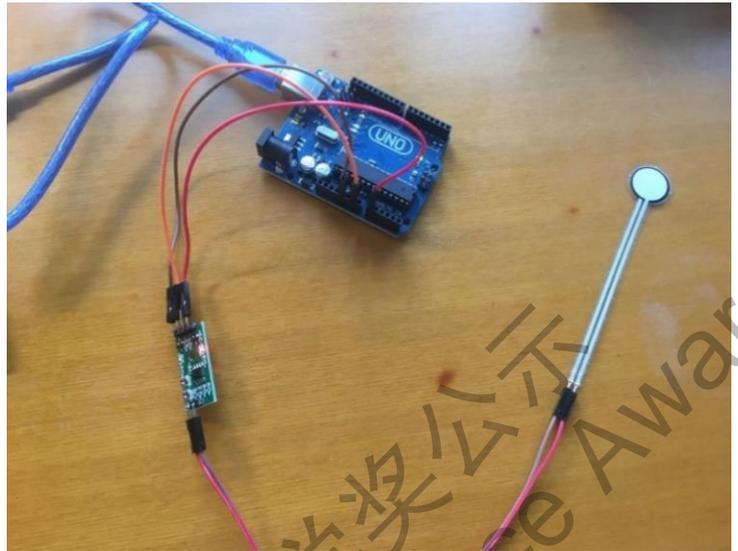


图 4-1 实物图

2.编写程序

```

presscode | Arduino 1.8.15
文件 编辑 项目 工具 帮助

presscode

void loop()
{
  long Fdata = getPressValue(sensorPin);
  DEBUGSerial.print("F = ");
  DEBUGSerial.print(Fdata);
  DEBUGSerial.println(" g,");
  delay(300);
}

long getPressValue(int pin)
{
  long PRESS_AO = 0;
  int VOLTAGE_AO = 0;
  int value = analogRead(pin);

  DEBUGSerial.print("AD = ");
  DEBUGSerial.print(value);
  DEBUGSerial.print(",");

  VOLTAGE_AO = map(value, 0, 1023, 0, 5000);

  DEBUGSerial.print("V = ");
  DEBUGSerial.print(VOLTAGE_AO);
  DEBUGSerial.print(" mv,");

  if (VOLTAGE_AO < VOLTAGE_MIN)
  {
    PRESS_AO = 0;
  }
  else if (VOLTAGE_AO > VOLTAGE_MAX)
  {
    PRESS_AO = PRESS_MAX;
  }
  else
  {
    PRESS_AO = map(VOLTAGE_AO, VOLTAGE_MIN, VOLTAGE_MAX, PRESS_MIN, PRESS_MAX);
  }

  return PRESS_AO;
}

```

图 4-3 程序

由于方案一接口不够、程序复杂、精度不够，因此选择方案二

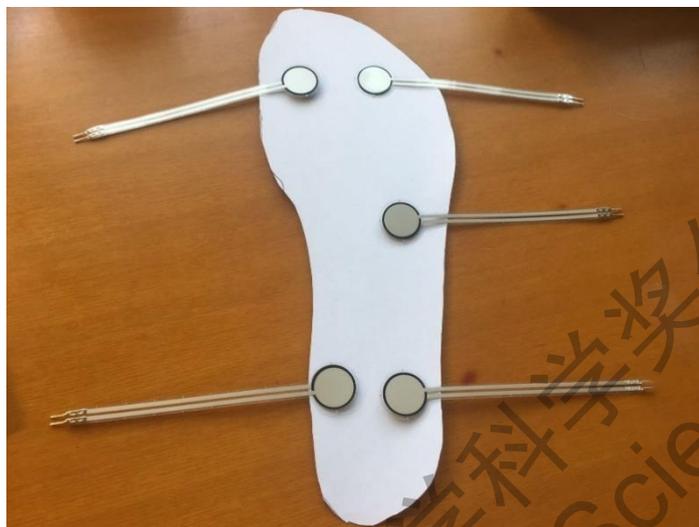


图 4-4

方案二：

红外测距传感器，用于测量鞋底到冰面距离

精度：0.01cm

制作过程如下：

1、实现单个红外测距传感器距离测量

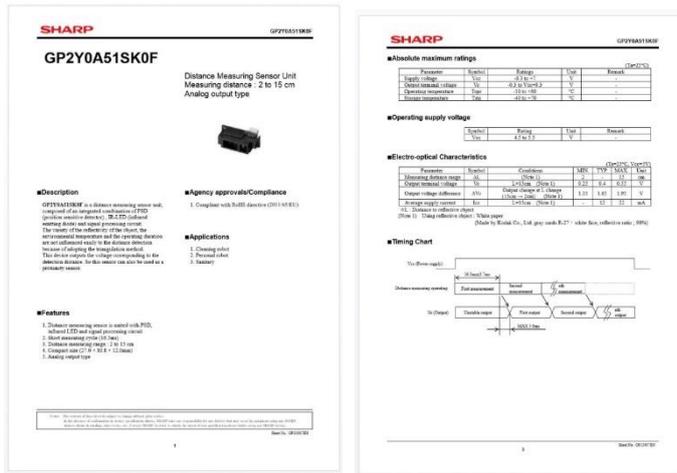


图 4-5 产品信息

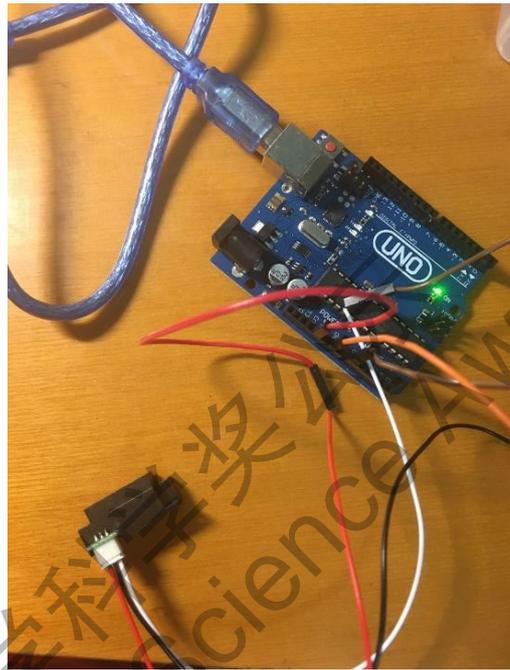


图 4-6 实物图

2、连接两个红外测距传感器

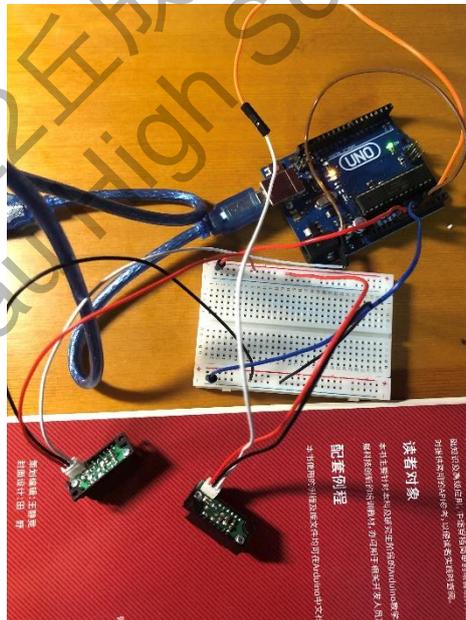


图 4-7

3、编写程序实现两个红外测距传感器距离测量及对比

```
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
```

图 4-8 程序

4、连接蓝牙模块

5、编写程序实现数据无线传输

6、连接 SD 卡及 SD 卡模块

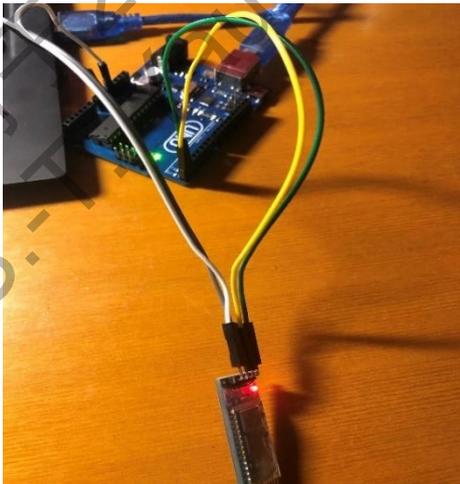


图 4-10

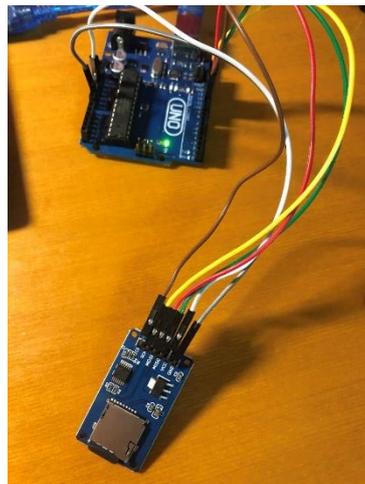


图 4-9

7、编写程序实现数据存储

8、整合安装

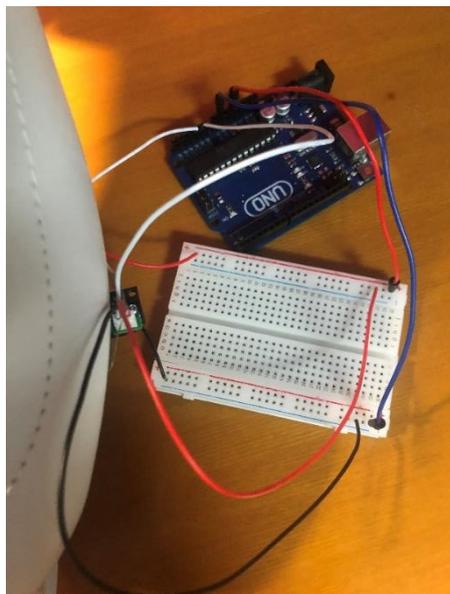


图 4-11



图 4-12

动作检测部分

由于百度人体关键点检测技术和 Openpose 运行容易卡顿，仅靠电脑 cpu 无法顺利运行，因此选用谷歌人体姿态识别技术检测关键点 (pycharm+opencv+pipemedia)

Mediapipe: MediaPipe Pose 是一个 ML 解决方案，用于高保真身体姿势跟踪，利用我们的 BlazePose 研究，从 RGB 视频帧推断出全身 33 个 3D 标志和背景分割掩码，该研究也支持 ML Kit 姿势检测 API。目前最先进的方法主要依赖于强大的桌面环境进行推理，而我们的方法在大多数现代手机、台式机/笔记本电脑、python 甚至 web 上实现了实时性能。

1、编写程序

```
main.py x
1 import cv2
2 import mediapipe as mp
3 mp_drawing = mp.solutions.drawing_utils
4 mp_drawing_styles = mp.solutions.drawing_styles
5 mp_pose = mp.solutions.pose
6
7 # For static images:
8 IMAGE_FILES = []
9 BG_COLOR = (192, 192, 192) # gray
10 with mp_pose.Pose(
11     static_image_mode=True,
12     model_complexity=2,
13     enable_segmentation=True,
14     min_detection_confidence=0.5) as pose:
15     for idx, file in enumerate(IMAGE_FILES):
16         image = cv2.imread(file)
17         image_height, image_width, _ = image.shape
18         # Convert the BGR image to RGB before processing.
19         results = pose.process(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
20
21         if not results.pose_landmarks:
22             continue
23         print(
24             f'Nose coordinates: ('
25             f'{results.pose_landmarks.landmark[mp_pose.PoseLandmark.NOSE].x * image_width}, '
26             f'{results.pose_landmarks.landmark[mp_pose.PoseLandmark.NOSE].y * image_height})'
27         )
28
29         annotated_image = image.copy()
30         # Draw segmentation on the image.
```

```
59
60 # To improve performance, optionally mark the image as not writeable to
61 # pass by reference.
62 image.flags.writeable = False
63 image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
64 results = pose.process(image)
65
66 # Draw the pose annotation on the image.
67 image.flags.writeable = True
68 image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)
69 mp_drawing.draw_landmarks(
70     image,
71     results.pose_landmarks,
72     mp_pose.POSE_CONNECTIONS,
73     landmark_drawing_spec=mp_drawing_styles.get_default_pose_landmarks_style())
74 # Flip the image horizontally for a selfie-view display.
75 cv2.imshow('MediaPipe Pose', cv2.flip(image, 1))
76 if cv2.waitKey(5) & 0xFF == 27:
77     break
78 cap.release()
```

图 4-13 程序

2、实现镜头前人体姿态识别

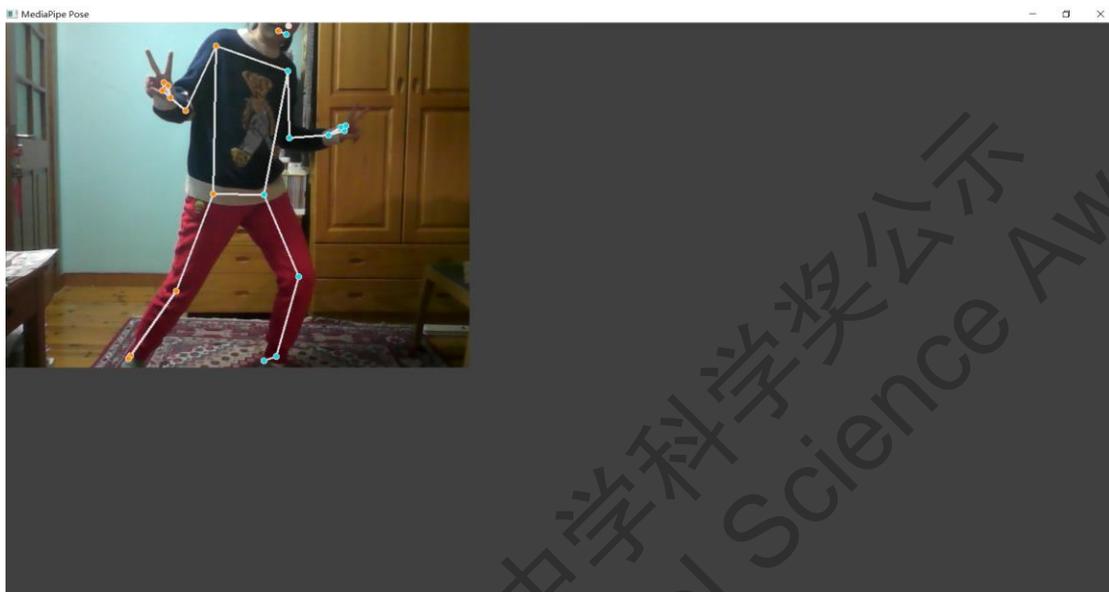


图 4-14

3、实现屏幕中的人体姿态识别



图 4-15

4.2.2. 项目原型

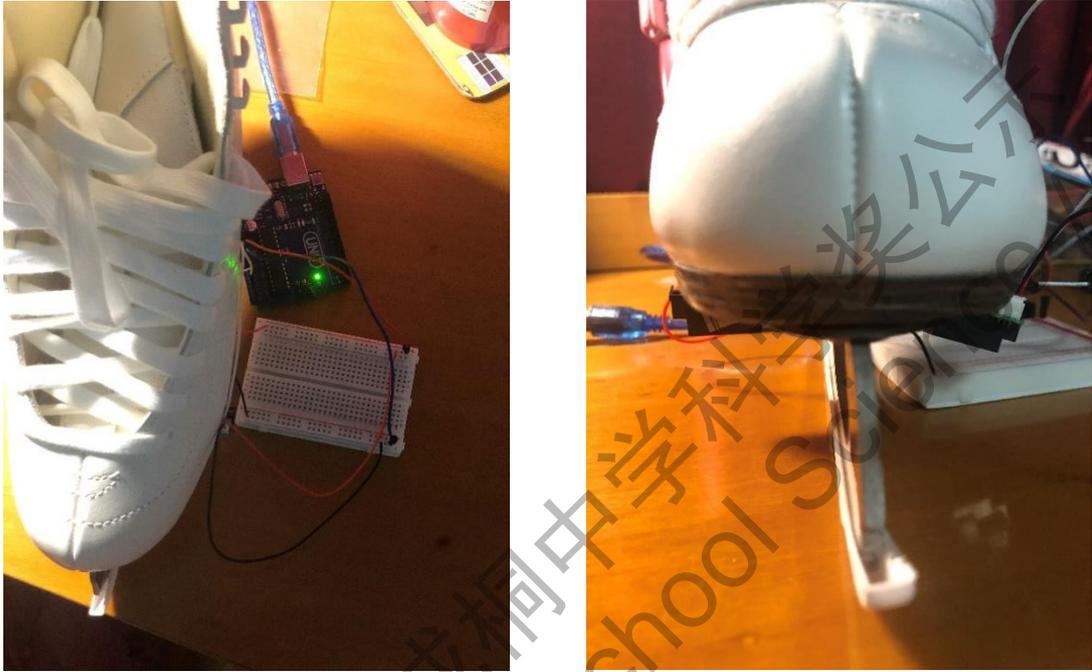


图 4-16 项目原型

这是最后的项目原型。

5. 实验设计与测试分析

5.1. 实验设计

为验证本课题的可行性，及验证是否能准确的无线检测花滑用刃、判定花花动作、进行比对，我设计了多个实验。其中包括：验证是否能准确判断花滑用刃、验证是否能判定花滑动作、验证是否能将花滑用刃与动作进行比对，以对功能性进行测试；验证红外测距传感器是否能测量距离冰面距离，以对科学性进行测试；实际使用测试以对实用性进行测试。

5.2. 功能性测试

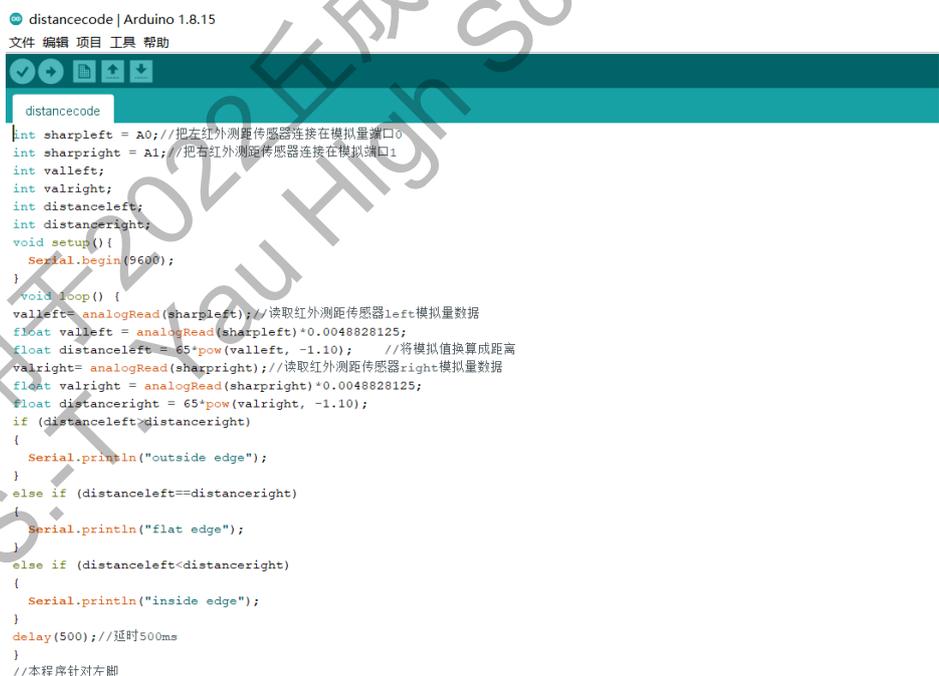
5.2.1. 实验一

实验目的：检测花滑用刃是否判定准确

实验材料：冰鞋、红外测距传感器装置

实验对象：花滑用刃判定

实验方法：在红外测距传感器装置安装在冰鞋上时，用手改变冰鞋的用刃状态，观察窗口监视器上的用刃判断，并记录数据



```
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量端口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
```

图 5-1

上图为花滑用刃检测部分代码，具体代码可以在附录中查看。

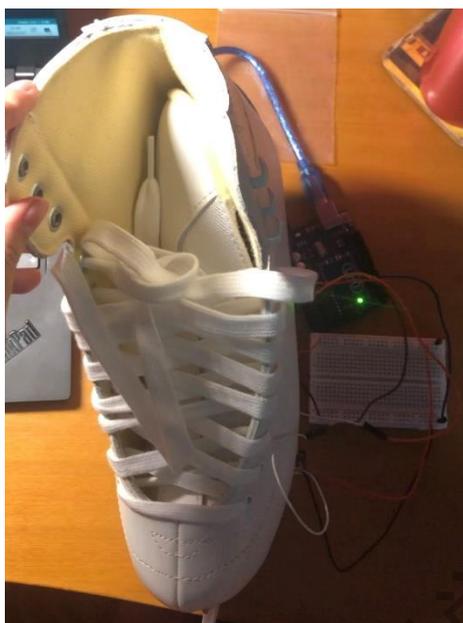


图 5-2

实验序号	用刃状态	用刃检测	是否准确(是为+, 否为-)
1	内刃	内刃	+
2	内刃	内刃	+
3	内刃	内刃	+
4	外刃	外刃	+
5	外刃	外刃	+
6	外刃	外刃	+
实验成功率	100%		

表 5-1

实验总结: 该装置可以准确检测花滑用刃。

5.2.2. 实验二

实验目的: 检验花滑动作是否能与用刃比对

实验材料: 计算机、花滑动作图片(为便于判断,均采用起跳图片)、冰鞋、红外测距传感器装置

实验对象: 花滑动作与用刃比对

实验方法: 对已知花滑动作的图片进行人体关键点检测,判断动作,并记录数据

实验序号	图片花滑动作	检测动作	判断是否准确(是为+, 否为-)
1	T	T	+
2	T	F	-
3	S	S	+
4	Lo	Lo	+
5	F	F	+
6	F	F	+
7	Lz	Lz	+
8	A	A	+
实验成功率	87.5%		

表 5-2

实验总结: 该装置可以较准确地判断花滑动作

5.2.3. 实验三

实验目的: 检验花滑动作是否判定准确

实验材料: 计算机、花滑动作图片(为便于判断,均采用起跳图片)

实验对象: 花滑动作判定

实验方法: 对已知花滑动作的图片进行人体关键点检测,判断动作,记录进 Excel 中;并用手改变冰鞋的用刃状态(不同动作对左右脚用刃要求不同),判断用刃,记录进 Excel 中;将两数据结合,与相应花滑动作下的正确用刃进行比对,判断花滑动作的用刃是否正确(正确为+, 错误为-; 是为+, 否为-)

实验序号	图片动作	用刃状态	动作检测	用刃检测	正确组合	判定结果	是否准确
1	A	左外	A	左外	A+左外	+	+
2	F	左内	F	左内	F+左内	+	+
3	L	右外	L	右外	L+右外	+	+
4	Lz	左外	Lz	左外	Lz+左外	+	+
5	F	左外	F	左外	F+左内	-	+
6	Lz	左内	Lz	左内	Lz+左外	-	+
成功率	100%						

表 5-3

实验总结：该装置可以准确地将花滑动作与用刃比对，并判断动作是否正确
总结：经实验，各功能如预期一样可以实现，证明了该装置各功能的可行性。



图 5-3

5.3. 精准度测试

5.3.1. 实验四

实验目的：验证红外测距传感器是否能测量到冰面的距离

实验方法：将红外测距传感器至于冰面上方，观察串口监视器上的距离数据，进行记录

预期结果：红外测距装置能准确测量距冰面距离

实验设备：红外测距传感器装置、冰块

变量：离冰面距离

实验步骤：

1. 将红外测距传感器至于冰面上方，改变距离
2. 观察串口监视器数据并记录

数据记录表格：

实验序号	1	2	3	4	5	6	成功率	精度
测量数据 (单位： cm)	1.45	2.39	4.50	5.96	6.49	3.40	100%	0.01cm
是否准确 (是为+， 否为-)	+	+	+	+	+	+		

表 5-4

数据分析:

每次实验均可测得距冰面距离

实验结果: 红外测距传感器装置能够准确测量距冰面距离

5.3.2. 实验五

实验目的: 检验红外测距传感器的量程及精度是否满足用刃检测的需求

实验方法: 计算用刃检测装置所需量程及精度, 对比产品信息及实验数据进行判断

预期结果: 红外测距传感器的量程及精度满足用刃检测的需求

实验设备: 红外测距传感器装置、冰鞋

实验步骤:

1. 利用数学建模求出理论量程
2. 采集红外测距传感器数据并结合产品信息判断

实验记录:

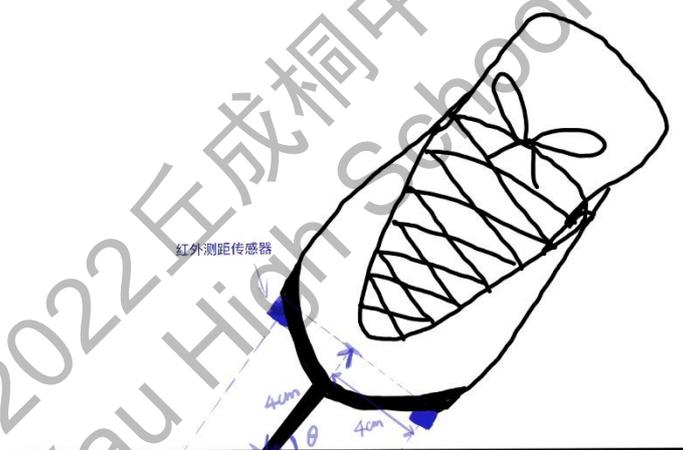


图 5-4 建模草图

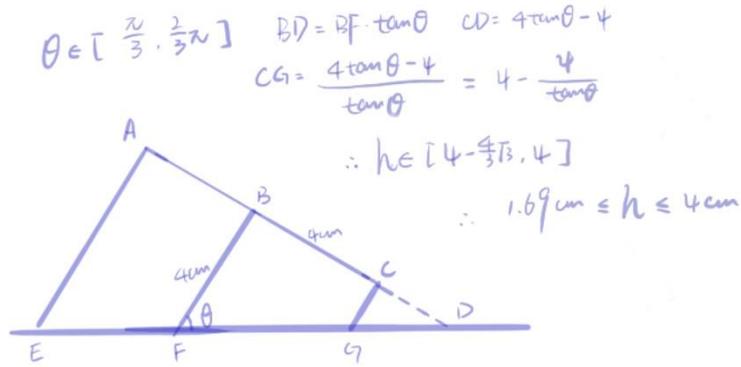


图 5-5 计算过程

数据分析:

1. 由图 5-5 可知理论量程为 1.69cm-4cm，精度应为 0.01cm
2. 由图 4-5 与表 5-4 可知，该红外测距传感器量程为 1-15cm，精度为 0.01cm

实验结果: 红外测距传感器的量程及精度满足用刃检测的需求

5.4. 实际使用测试

5.4.1. 实验六

实验目的: 测试该花样滑冰用刃检测系统是否能投入实际使用

实验材料: 基于 MediaPipe 的花样滑冰检测系统、冰鞋

实验步骤: 将该装置安装于冰鞋上，在冰场测试使用情况

实验结果: 基本的用刃检测和无线数据传输都可正常实现。然而由于我不会用于检测的几种跳跃动作，仅检测了能否通过摄像头识别人体关键点。为完善实验，我增加了对简单步法的检测，以葫芦步为例，经实验能够准确识别用刃、动作及二者的比对。除此之外，我也发现了一些待改进点。

5.4.2. 使用访谈

我们对使用者进行简单的访谈，并记录不同使用者的使用反馈，反馈如下:

1. 使用者 A (本人): 感觉装置安装还不够稳固，在滑时很害怕装置掉下来。Arduino UNO、蓝牙模块等只能固定在冰鞋侧面，接线也藏在冰鞋底部，虽不影响滑冰，仍不太美观。由于电脑只有前置摄像头，

在追踪拍摄时操作很不方便。

2. 冰场教练（并未使用，只从花滑专业角度评估）：虽然只判断内外用刃也能判断动作准确与否，但很多动作对前后用刃也有要求，可以做进一步的改进。

5.5. 结论概述

经实验，我从可行性、科学性、实用性的维度上分别进行了测试，从上述实验我们可以得知以下情况：

1. 本课题能够实现实时、准确地判定花滑用刃
 2. 本课题能利用人体关键点识别技术实时判定花滑动作
 3. 本课题能通过蓝牙实现数据的无线传输，不受场地束缚
 4. 红外测距传感器能测量离冰面距离
 5. 红外测距传感器精度为 0.01cm，量程为 1-15cm，满足判定花滑用刃的需求
 6. 装置安装还有待改进
 7. 装置可更加轻便、美观
 8. 实际应用中的跟拍动作识别还有待改进
 9. 用刃判断机制还有待完善
- 因此本课题基本达到预期，但是不同的实验也暴露出了许多不足之处，有待将来改进

6. 总结

6.1. 课题成果

基于 MediaPipe 的花样滑冰用刃检测系统基本达到预期。

本文设计的基于 MediaPipe 的花样滑冰用刃检测系统，通过 Arduino IDE 与 Pycharm 实现，具有以下突出特点：

- 1.能够实时、准确地在冰面上判定花滑用刃
- 2.能利用人体关键点识别技术实时判定花滑动作
- 3.能将动作与用刃整合与标准动作比对判断是否正确用刃
- 4.能通过蓝牙实现数据的无线传输，不受场地束缚
- 5.MediaPipe 在笔记本的 cpu 配置中就能顺畅运行，且支持手机等更小的移动设备

本论文完成的主要工作包含以下几个方面：

- 1.发现问题
- 2.背景需求调研
- 3.查阅相关文献
- 4.设计解决方案
- 5.进行初步开发
- 6.确定方案
- 7.开发用刃检测部分并实验
- 8.开发花滑动作检测部分并实验
- 9.将用刃检测部分与花滑动作检测部分整合并实验
- 10.整合全部投入实际使用并实验
- 11.根据实验发现问题并优化
- 12.撰写研究报告

6.2. 未来展望

1. 该课题仍有不足之处：装置安装不牢固，可在今后设计更能够贴合冰鞋的结构，并牢固安装；Arduino UNO 占用面积仍较大，装置不够轻便、美观，今后可进一步精简用 Arduino NANO 代替，并设计更方便贴合冰鞋使用的方案；用电脑前置摄像头跟拍不方便，可将该程序制作为手机 APP，用手机跟拍，使操作更便捷；判断用刃机制还可以更加完善，可以重新用被放弃的方案一，及用薄膜压力传感器检测来添加前后刃的检测。

2. 若该系统投入商业使用可帮助花滑教学进行更加顺利，甚至可投入比赛的用刃判断中。

仅用于2022丘成桐中学科学奖公示
2022 S.-T. Yau High School Science Awards

7. 收获与成长

对花样滑冰这项富有魅力的体育运动的爱是本课题诞生的源头。也正是这种运动追求标准动作的严谨性，促使我能从自己所热爱、所关注的事物中发现问题，并试着解决它。在整个课题的实践过程中，我经历了许多第一次，第一次进行市场调研，第一次设计机械结构，第一次自学一门新的编程语言……当然，我也经历了许多挫折，第一次接触 Arduino 的我对它一窍不通，于是就买了一本入门教程自学，一步步地摸索、尝试；在实物制作时要简化原本繁杂的电路，将所有元件都集中在冰鞋上，同时保证不影响滑行；我还必须对不同的方案做实验，做出取舍；从未接触人体姿态识别的我利用网络资源并在辅导老师的帮助下对该技术有了初步的认识和应用。虽然科学研究路上有重重阻碍，但一次次的试错也让我有了莫大的成长与收获。

在课题研究过程中，对不同方案的取舍、对精度的严格要求、多方面多角度的实验设计都让我体会到了科学研究的严谨。不断的探索也让我发现了更多课题的不足之处，等待着我进一步的提升优化。

课题的制作极大地提升了我的科学素养、自学能力、论文撰写能力，也让我学习到了 Arduino、Pycharm 编程、Excel 中的基础技能。

最后，我希望花样滑冰这项仍较小众的体育运动被更多人所关注，也希望能出现更多将体育项目与工程、计算机技术等其他领域结合的课题。

8. 致谢

选题来源、研究背景：花滑中许多动作如步法、跳跃、旋转等都有规定的肢体动作和用刃要求。然而，在教学时，由于滑行速度快或场地起雾等原因，教练无法清晰地判断学生的用刃是否正确，从而导致教学出错、学习效率低等问题。本课题通过工程结构、人体姿态识别程序、物理建模等，加以大量实验以达到识别花滑用刃及花滑动作的目的，并判断花滑用刃是否符合规范以帮助花样滑冰教学并提高花样滑冰比赛的公平性。

队员在论文撰写中承担的工作以及贡献：队员：麦祺悦；承担工作：所有工作。

指导老师：包鸣，提供理论指导，实验建议及论文撰写指导；指导无偿。

感谢学校课题辅导老师对我的指导和帮助。

感谢一直支持我的课题研究的家人们。

最后，感谢我自己，感谢自己的执着与坚持。

9. 参考文献

- [1]King, D. L., Hawks, et al. Instrumented figure skating blade for measuring on-ice skating forces[J]. Measurement Science & Technology, 2014.
- [2]刘颖. 花样滑冰鲁茨跳常见技术问题诊断与辅助训练[J]. 冰雪运动, 2014(5):5.
- [3]王旋, 王树本. 花样滑冰跳跃动作的原理与标准[J]. 冰雪运动, 2002(2):2.
- [4]韦启航, 卢世壁. 人体步态分析系统:足底压力测量系统的研制[J]. 中国生物医学工程学报, 2000, 19(1):32-40.
- [5]王军, 徐新智. 足底压力图像分析系统及骨科应用[J]. 第四军医大学学报, 2001, 22(11):1.
- [6]殷海艳, 刘波. 基于部位检测的人体姿态识别[J]. 计算机工程与设计, 2013, 34(10):5.
- [7]姜桂萍, 纪仲秋, 王凯. 花样滑冰点冰跳的运动学研究[J]. 北京体育大学学报, 2006, 29(3):3.
- [8]许兆晓, 赵国娜. 国际滑联花样滑冰新旧裁判评分系统的比较[J]. 冰雪运动, 2007, 29(4):5.
- [9]杨振华. 红外测距传感器:, CN306420683S[P]. 2021.

10. 附录

10.1.相关代码

```
int sensorPin = A0;
#define PRESS_MIN    500
#define PRESS_MAX    20000
#define VOLTAGE_MIN 150
#define VOLTAGE_MAX 3300
void setup()
{
    DEBUGSerial.begin(9600); // setup serial
    Serial.println("setup end!");
}
void loop()
{
    long Fdata = getPressValue(sensorPin);
    DEBUGSerial.print("F = ");
    DEBUGSerial.print(Fdata);
    DEBUGSerial.println(" g.");
    delay(300);
}
long getPressValue(int pin)
{
    long PRESS_AO = 0;
    int VOLTAGE_AO = 0;
    int value = analogRead(pin);
    DEBUGSerial.print("AD = ");
    DEBUGSerial.print(value);
    DEBUGSerial.print(" ,");
    VOLTAGE_AO = map(value, 0, 1023, 0, 5000);
    DEBUGSerial.print("V = ");
    DEBUGSerial.print(VOLTAGE_AO);
```

```

DEBUGSerial.print(" mv,");
if(VOLTAGE_AO < VOLTAGE_MIN)
{
    PRESS_AO = 0;
}
else if(VOLTAGE_AO > VOLTAGE_MAX)
{
    PRESS_AO = PRESS_MAX;
}
else
{
    PRESS_AO = map(VOLTAGE_AO, VOLTAGE_MIN,
VOLTAGE_MAX, PRESS_MIN, PRESS_MAX);
}
return PRESS_AO;
}

```

```

int sharpleft = A0;
int sharpright = A1;
int valleft;
int valright;
int distanceleft;
int distanceright;
void setup(){
    Serial.begin(9600);
}
void loop() {
    valleft= analogRead(sharpleft);
    float valleft = analogRead(sharpleft)*0.0048828125;
    float distanceleft = 65*pow(valleft, -1.10);
    valright= analogRead(sharpright);

```

```
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
```

```
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.
    # To improve segmentation around boundaries, consider applying a joint
    # bilateral filter to "results.segmentation_mask" with "image".
    condition = np.stack((results.segmentation_mask,) * 3, axis=-1) > 0.1
    bg_image = np.zeros(image.shape, dtype=np.uint8)
    bg_image[:] = BG_COLOR
    annotated_image = np.where(condition, annotated_image, bg_image)
    # Draw pose landmarks on the image.
    mp_drawing.draw_landmarks(
        annotated_image,
        results.pose_landmarks,
        mp_pose.POSE_CONNECTIONS,

landmark_drawing_spec=mp_drawing_styles.get_default_pose_landmarks_style())
    cv2.imwrite('/tmp/annotated_image' + str(idx) + '.png', annotated_image)
    # Plot pose world landmarks.
    mp_drawing.plot_landmarks(

```

```

        results.pose_world_landmarks, mp_pose.POSE_CONNECTIONS)

# For webcam input:
cap = cv2.VideoCapture(0)
with mp_pose.Pose(
    min_detection_confidence=0.5,
    min_tracking_confidence=0.5) as pose:
while cap.isOpened():
    success, image = cap.read()
    if not success:
        print("Ignoring empty camera frame.")
        # If loading a video, use 'break' instead of 'continue'.
        continue

    # To improve performance, optionally mark the image as not writeable to
    # pass by reference.
    image.flags.writeable = False
    image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
    results = pose.process(image)

    # Draw the pose annotation on the image.
    image.flags.writeable = True
    image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)
    mp_drawing.draw_landmarks(
        image,
        results.pose_landmarks,
        mp_pose.POSE_CONNECTIONS,

        landmark_drawing_spec=mp_drawing_styles.get_default_pose_landmarks_style())
    # Flip the image horizontally for a selfie-view display.
    cv2.imshow('MediaPipe Pose', cv2.flip(image, 1))
    if cv2.waitKey(5) & 0xFF == 27:
        break
cap.release()

```

10.2.花滑动作标准

T:后外点冰跳 (Toe loop jumps) 以逆时针选手为例, 后外点冰跳是左脚刀齿点冰, 右脚后外刃起跳。

F:后内点冰跳 (Flip jumps) 以逆时针选手为例, 后内点冰跳, 是右脚刀齿点冰, 左脚后内刃起跳。

Lz:勾手跳 (Lutz jumps) 以逆时针选手为例, 勾手跳是右脚刀齿点冰, 左脚后外刃起跳。

S:后内结环跳 (Salchow jumps) 以逆时针选手为例, 后内结环跳, 是左脚后内刃起跳。

L:后外结环跳 (Loop jumps) 以逆时针选手为例, 后外结环跳是右脚后外刃起跳。

A:阿克塞尔跳, 又称前外跳 (Axel jumps) 以逆时针选手为例, 阿克塞尔跳是左脚前外刃起跳。它是唯一一个用前刃起跳的动作。

详见国际冰联官网:

<https://www.isu.org/figure-skating/rules/fsk-regulations-rules/file>