



XIAOCAI TONGXUE

A SMART MUSIC ASSISTANT

– HCI FINAL PROJECT OF TONGJI UNIV

1753837 陈柄畅

1753771 蔡奕阳

1754266 贺鹏宇

INSTRUCTOR: 沈莹

ACADEMIC YEAR 2018-2019

---

# INDICE

|          |                                  |           |
|----------|----------------------------------|-----------|
| <b>1</b> | <b>Introduction</b>              | <b>3</b>  |
| 1.1      | Description of Program . . . . . | 3         |
| 1.2      | Modules . . . . .                | 4         |
| 1.2.1    | Voice Control . . . . .          | 5         |
| 1.2.2    | Play Piano . . . . .             | 6         |
| 1.2.3    | Music Game . . . . .             | 6         |
| 1.2.4    | AI Compose . . . . .             | 7         |
| 1.2.5    | Play Previous Music . . . . .    | 7         |
| <b>2</b> | <b>Requirements</b>              | <b>7</b>  |
| 2.1      | Hardware Requirements . . . . .  | 7         |
| 2.2      | Software Requirements . . . . .  | 7         |
| <b>3</b> | <b>Advantages</b>                | <b>8</b>  |
| <b>4</b> | <b>Disadvantages</b>             | <b>9</b>  |
| <b>5</b> | <b>How to improve</b>            | <b>9</b>  |
| <b>6</b> | <b>Workload</b>                  | <b>10</b> |

---

# INTRODUCTION

## 1.1 DESCRIPTION OF PROGRAM

Xiaocaitongxue is a voice interactive music assistant. It can open the corresponding function through voice interaction. And after waking up the device, you can choose four modes to play:

1. Play piano
2. Music game
3. AI compose
4. Play previous music

And you can control the state of the equipment by gesture. We use lasers to make light projection on the fingers so that the camera can easily detect where is the fingers.

At first, we run the function `wake_up()`. In this function, we set a detector about XiaoCaiTongXue and then open it. After the detector is opened, it will detect the hotword "xiaocaitongxue"

If the hotword is detected, the program will callback the function `callback_xctx()`. The function will terminate the detector about XiaoCaiTongXue. And then open the detector about the function.

If the hotword about different function is detected, the program will terminate the detector about the function. And then execute the specific function, `callback_game()`, `callback_music()`, `callback_AI()` and `callback_piano()`.

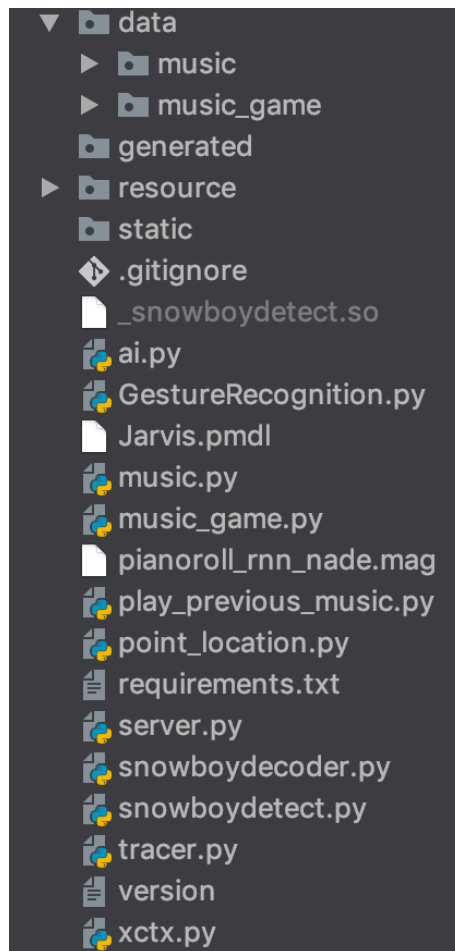
If "alexa" is detected, it will run the function `callback_piano()`. In this function, it will run the function `playMusic()` to play the piano.

If "Hey Friday" is detected, it will run the function `callback_game()`. In this function, it will run the function `music_game.run()` to play music game.

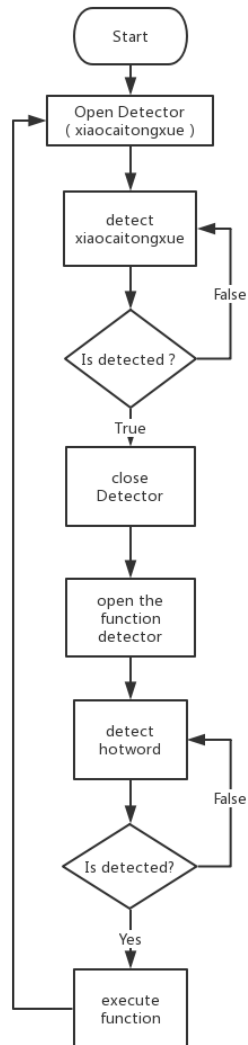
If "Jarvis" is detected, it will run the function `callback_music()`. In this function, it will run the function `play_previous_music()` to play the previous music.

If "snowboy" is detected, it will run the function `callback_ai()`. In this function, it will run the function `playMusicWithAI()` to play the piano with ai.

## 1.2 MODULES



### 1.2.1 VOICE CONTROL



1. We set a detector about XiaoCaiTongXue and then open it.
2. After the detector is opened, it will detect the hotword "xiaocaitongxue"
3. If the hotword is detected, the program will terminate the detector about XiaoCaiTongXue. And then open the detector about the function.
4. If the hotword about different function is detected, the program will terminate the detector about the function. And then execute the specific function.

### 1.2.2 PLAY PIANO

1. Set 4 variables `last_status_x`, `last_status_y`, `now_status_x`, `now_status_y`. `last_status_x` and `last_status_y` record the area where your finger last clicked. `now_status_x` and `now_status_y` record the area where your finger clicked this time.
2. Run `getPointLocation` method to get a list of the positions of laser points reflected by your fingers which called `mc`.
3. If the size of `mc` is 0, set `last_status_x=0` and `last_status_y=0`. This step This step allows you to make a sound even if you click on the same area twice.
4. If the size of `mc` is larger than 0, update `now_status_x` and `now_status_y` to record the area you clicked this time.
5. If the area you clicked this time is different from the area you clicked last time, then play the sound of this area.
6. Update `now_status_x` and `now_status_y` to record the area you clicked this time.
7. Recognize your gesture. If you want to end, the program will be terminated or it will perform step 2 again.

#### Get Position of the point

1. Reset the size of the image to 120\*90 to reduce the amount of calculation.
2. Convert the image into a binary image.
3. Use `morphologyEx` method in `openCV` to perform a close operation on the image. This step will merge those close patterns
4. Use `findContours` in `OpenCV` to find the profile of the patterns on the image.
5. Calculate the centroid of each pattern.

### 1.2.3 MUSIC GAME

The program randomly generates 12 blocks and their length and move speed. At the same time, the program will detect the distance between user's finger and the blocks from both 'x' and 'y' axes. Different distances will display different notice sound and the nearer distance is, the rapid the sound is displayed. If user's finger crashes a block, the game will get over.

#### 1.2.4 AI COMPOSE

The program is based on Magenta (Google). Because of the different of CPU structure, this framework cannot work on Raspberry Pi. So we build a backend on Aliyun to solve this problem. We use a public Deep Learning model from Github. This model applies language modeling to polyphonic music generation using an LSTM combined with a NADE, an architecture called an RNN-NADE. When user play some sound, the program will send a list of number to the server. After the program running on the server inputs the sequence into the model and gets the result. It will send it back. Raspberry Pi will play the music AI writes based on user's input.

#### 1.2.5 PLAY PREVIOUS MUSIC

When user play piano, the program will automatically save the melody into a 'txt' file. When user wants to play previous music, the program will read the file and replay the music.

## 2

---

## REQUIREMENTS

### 2.1 HARDWARE REQUIREMENTS

1. Raspberrypi 3b \*2
2. Laser \*4
3. 3D-Print Box \*1
4. Speaker \*2
5. Camera \*1
6. Microphone \*1

### 2.2 SOFTWARE REQUIREMENTS

1. Python3

2. Opencv3
3. Snowboy
4. Request
5. Flask
6. Magenta

# 3

---

## ADVANTAGES

1. A new way of interaction No one before has used lasers combined with camera to give users a totally new kind of interaction. User can play piano just like they are drawing by their fingers. This kind of interaction will help users get rid of the influence of vision and enjoy the sound.
2. Voice Wake-up: We use the snowboy, an highly customizable hotword detection engine that is embedded real-time and is always listening (even when off-line) compatible with Raspberry Pi, (Ubuntu) Linux, and Mac OS X.
3. Gesture Interaction: Through gesture recognition, we can achieve different operations and interactions.
4. The usage of laser: This is a novel way of interacting. Through the camera, the device can recognize the position of the finger, which is reflected by the laser. Because we use laser to get the position of fingers, theoretically it can recognize more accurately than totally by computer vision.
5. Multiple modes: We have four modes: 1. Play piano. 2. Music game 3. AI compose 4. Play previous music.
6. Different play situation: Because of our usage of laser, user can play piano on different material, such as hard desks or soft sofas. If user would like to, he or she can play while hanging their hands in the air.



# 4

---

## DISADVANTAGES

1. Due to time and labor cost constraints, our sound model training is not perfect. Sometimes the recognition rate is low.
2. Due to the limited ability of the Raspberry Pi to process data, gesture recognition has a very little delay.

# 5

---

## HOW TO IMPROVE

1. Improve the sound model by collecting sound models online and improve accuracy.
2. Reduce the weight of the device by updating the hardware
3. Add new mode to enhance fun. Such as adding voice recognition, which can make our music assistant more smart and easier to use.

# 6

---

## WORKLOAD

We built this product from vary basic layer. So our workload are huge, not only including coding, but also including the construction of the whole product. We welded the laser with wires. We designed and built the box through 3-D print. We adjusted the equipment to fit Raspberry Pi and camera. So our hardware workload is as heavy as our coding workload.