国际通用工程学院

THE SCHOOL OF GENERAL ENGINEERING

A Walking Assistance System For USN

IGE B5

June 06th 2020

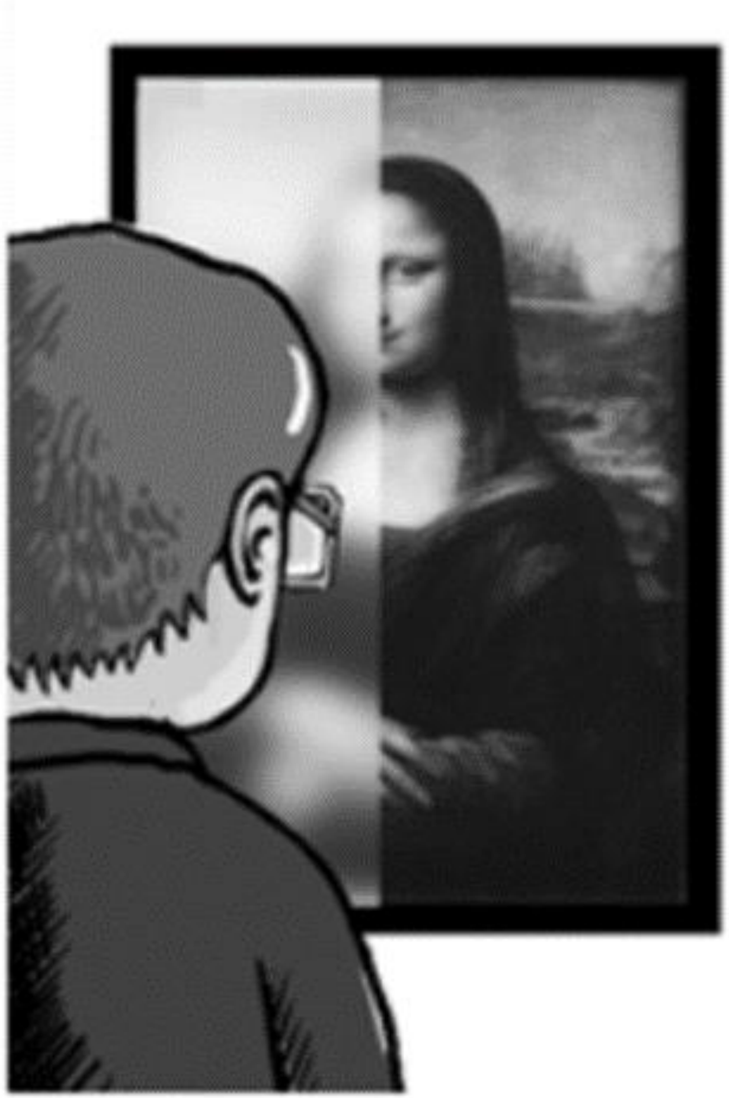
General design report

Client: Kamran Behdinan

Faculty advisor: Peng Lei

1· What is USN?

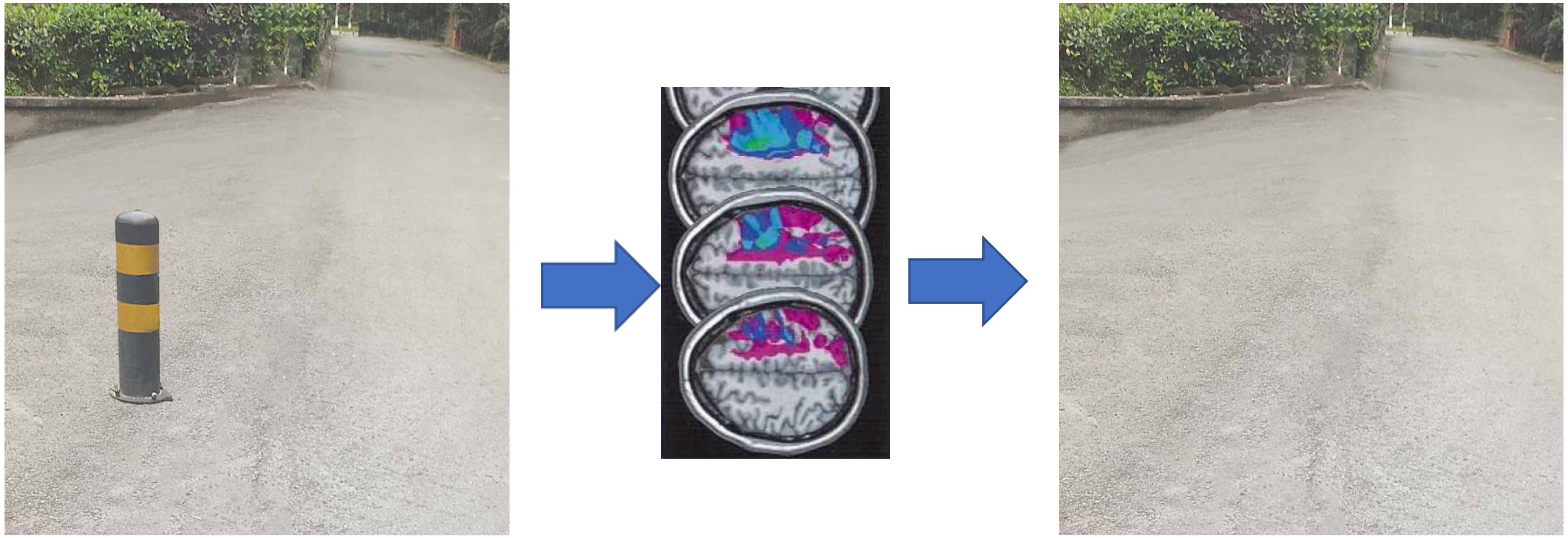
1 What is USN?



What is USN?

USN is a neuropsychological condition in which, after recovering from stroke, a deficit in attention to and awareness of one side of the field of vision is observed.

1 What is USN?

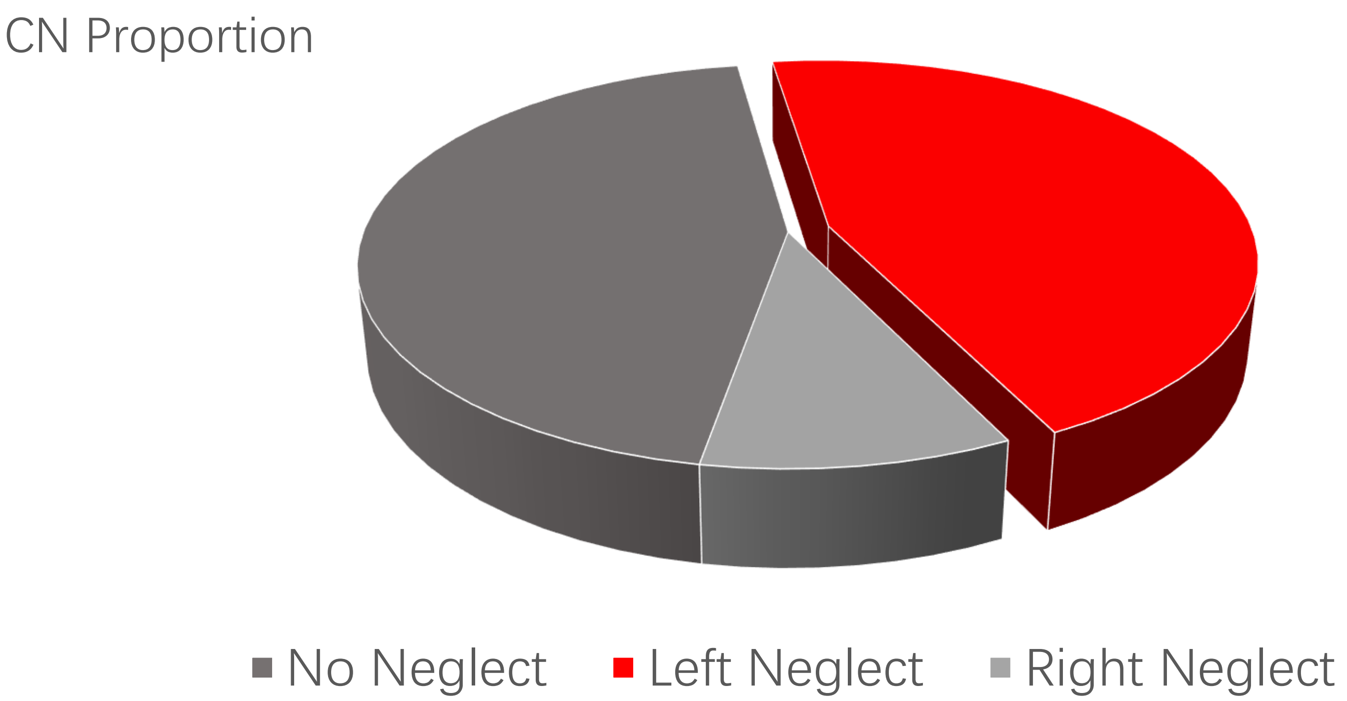


Injuries in the colored part may very well cause neglect. It interrupt Integration & transformation of optic Information.

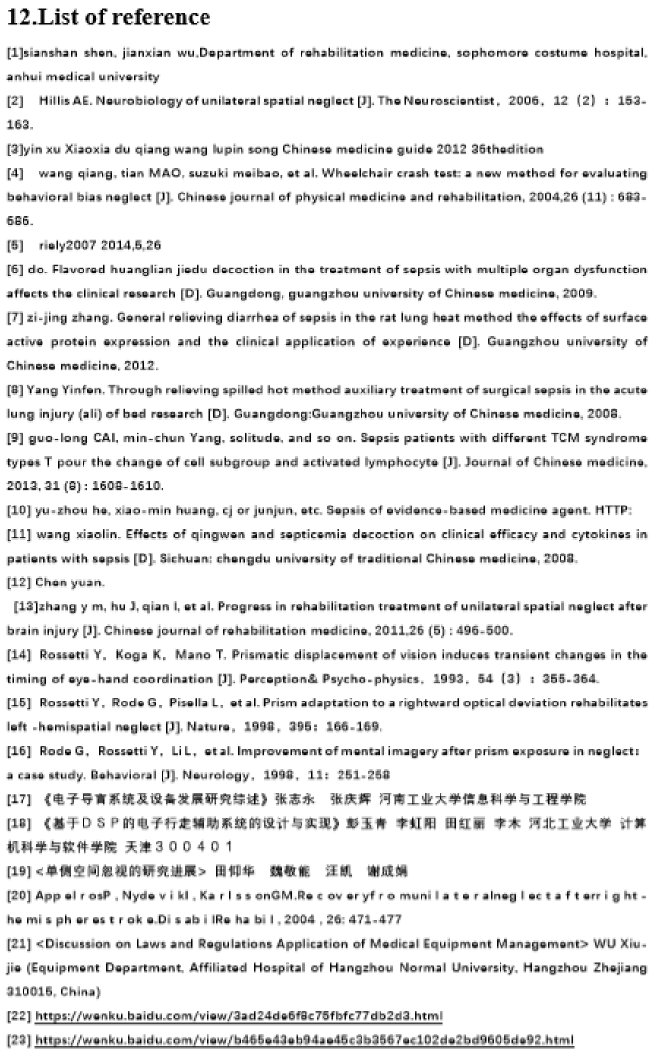
So patients have safety issues and suffer from self-abasement.

1 What is USN?

Unfortunately, more than **2.5 million** new patients suffer from head injuries in china every year!!!
And only 45% of them dose not suffer from neglect.



1 What is USN?



12.List of reference

[1]sianshan shen, jianxian wu,Department of rehabilitation medicine, sophomore costume hospital, anhui medical university

[2] Hillis AE. Neurobiology of unilateral spatial neglect [J]. The Neuroscientist，2006，12（2）：153-163.

[3]yin xu Xiaoxia du qiang wang lupin song Chinese medicine guide 2012 35thedition

[4] wang qiang, tian MAO, suzuki meibao, et al. Wheelchair crash test: a new method for evaluating behavioral bias neglect [J]. Chinese journal of physical medicine and rehabilitation, 2004,26 (11) : 683-686.

[5] riely2007 2014,5,26

[6] do. Flavored huanglian jiedu decoction in the treatment of sepsis with multiple organ dysfunction affects the clinical research [D]. Guangdong, guangzhou university of Chinese medicine, 2009.

[7] zi-jing zhang. General relieving diarrhea of sepsis in the rat lung heat method the effects of surface active protein expression and the clinical application of experience [D]. Guangzhou university of Chinese medicine, 2012.

[8] Yang Yinfen. Through relieving spilled hot method auxiliary treatment of surgical sepsis in the acute lung injury (ali) of bed research [D]. Guangdong:Guangzhou university of Chinese medicine, 2008.

[9] guo-long CAI, min-chun Yang, solitude, and so on. Sepsis patients with different TCM syndrome types T pour the change of cell subgroup and activated lymphocyte [J]. Journal of Chinese medicine, 2013, 31 (8) : 1608-1610.

[10] yu-zhou he, xiao-min huang, cj or junjun, etc. Sepsis of evidence-based medicine agent. HTTP:

[11] wang xiaolin. Effects of qingwen and septicemia decoction on clinical efficacy and cytokines in patients with sepsis [D]. Sichuan: chengdu university of traditional Chinese medicine, 2008.

[12] Chen yuan.

[13]zhang y m, hu J, qian l, et al. Progress in rehabilitation treatment of unilateral spatial neglect after brain injury [J]. Chinese journal of rehabilitation medicine, 2011,26 (5) : 496-500.

[14] Rossetti Y，Koga K，Mano T. Prismatic displacement of vision induces transient changes in the timing of eye-hand coordination [J]. Perception& Psycho-physics，1993，54（3）：355-364.

[15] Rossetti Y，Rode G，Pisella L，et al. Prism adaptation to a rightward optical deviation rehabilitates left -hemispatial neglect [J]. Nature，1998，395：166-169.

[16] Rode G，Rossetti Y，Li L，et al. Improvement of mental imagery after prism exposure in neglect: a case study. Behavioral [J]. Neurology，1998，11：251-258

[17] 《电子导盲系统及设备发展研究综述》张志永　张庆辉 河南工业大学信息科学与工程学院

[18] 《基于ＤＳＰ的电子行走辅助系统的设计与实现》彭玉青 李虹阳 田红丽 李木 河北工业大学 计算机科学与软件学院 天津３００４０１

[19] <单侧空间忽视的研究进展> 田仰华　魏敬能　汪凯　谢成娟

[20] App el r osP , Nyde v i kI , Ka r l s s onGM.Re c ov er yf r o muni l a t e r alneg l ec t a f t err i g ht -he mi s ph er es t r ok e.Di s ab i lRe ha bi l , 2004 , 26: 471-477

[21] <Discussion on Laws and Regulations Application of Medical Equipment Management> WU Xiu-jie (Equipment Department, Affiliated Hospital of Hangzhou Normal University, Hangzhou Zhejiang 310015, China)

[22] https://wenku.baidu.com/view/3ad24de6f8c75fbfc77db2d3.html

[23] https://wenku.baidu.com/view/b465e43eb94ae45c3b3567ec102de2bd9605de92.html

404
not found
采访稿

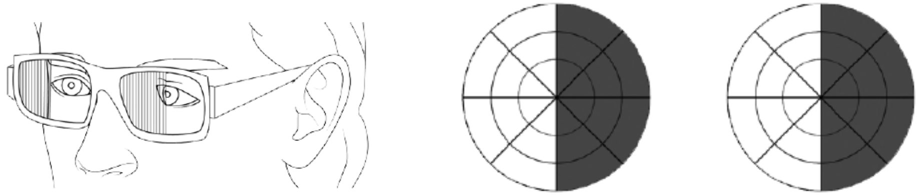
Huge Amount Of Document

Visit Hospital

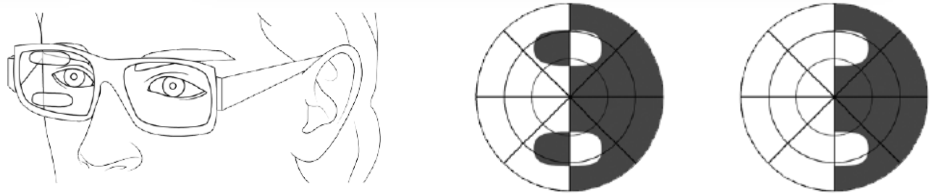
2· Market Survey

2 Exist Solutions

The Rehabilitation Train

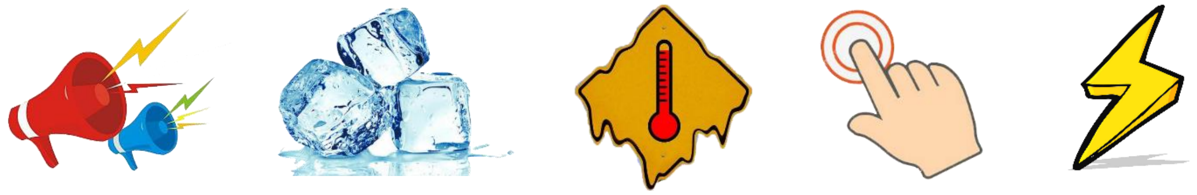


1 Prism Adaptation



2 Strong Stimulations

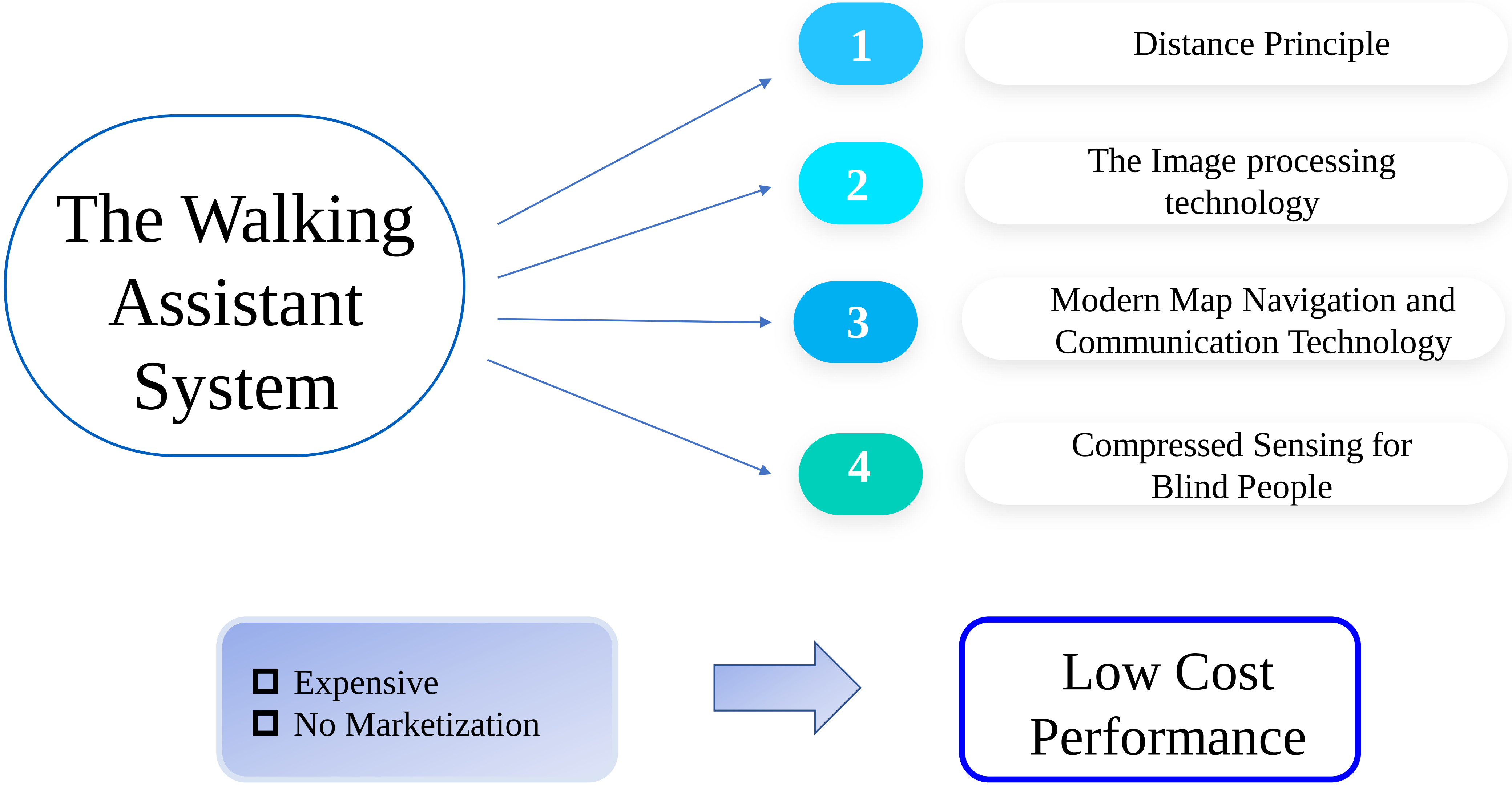
MOVE



□ Expensive
□ Need Long Time

Low Cost Performance

2 Exist Solutions



The Walking Assistant System

1 Distance Principle

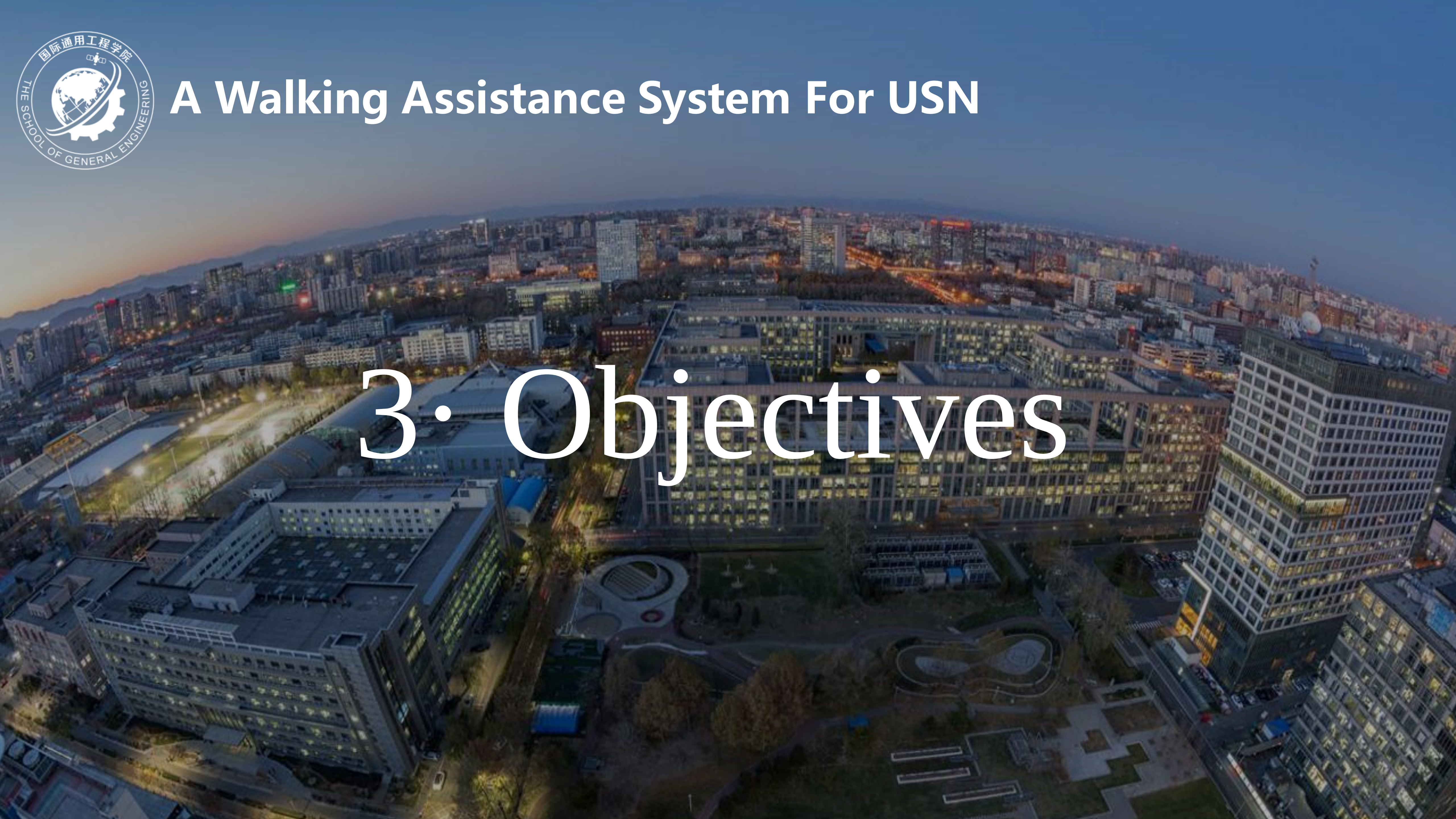
2 The Image processing technology

3 Modern Map Navigation and Communication Technology

4 Compressed Sensing for Blind People

- □ Expensive
- □ No Marketization

Low Cost Performance



3· Objectives

2 **Objectives**

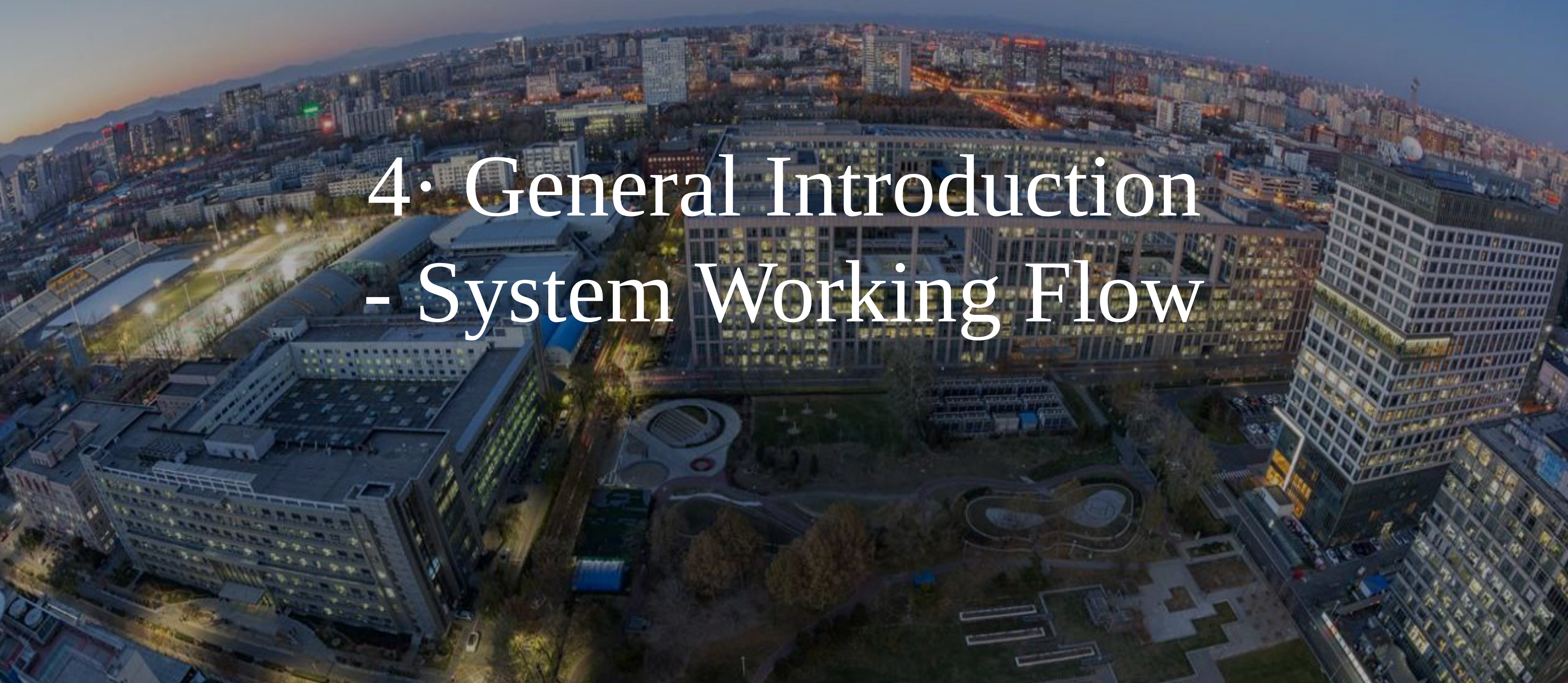
1. Obstacle Avoiding:

✓ identify and remind

✓ Range: 2m   success rate:95%

2. Dealing With Emergency:

✓ minimize danger



4· General Introduction
- System Working Flow

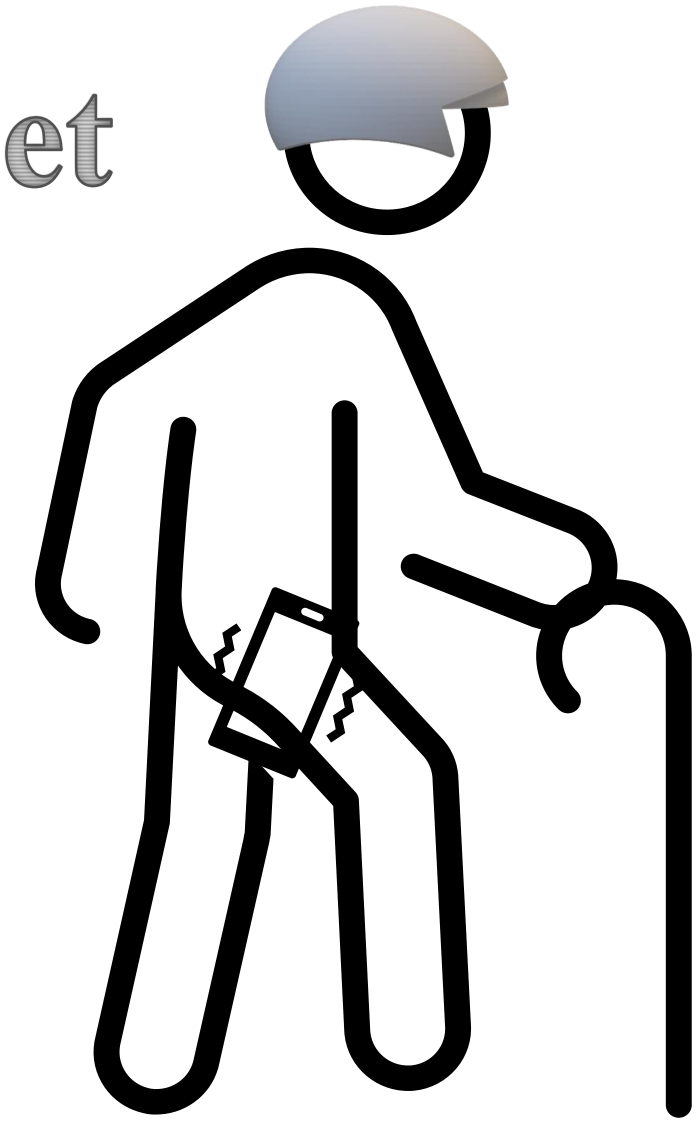
- Introduction of parts
- Obstacle detecting
- Fall detecting

To make the illustration better, let’s welcome our model.

The two part of our system {

helmet

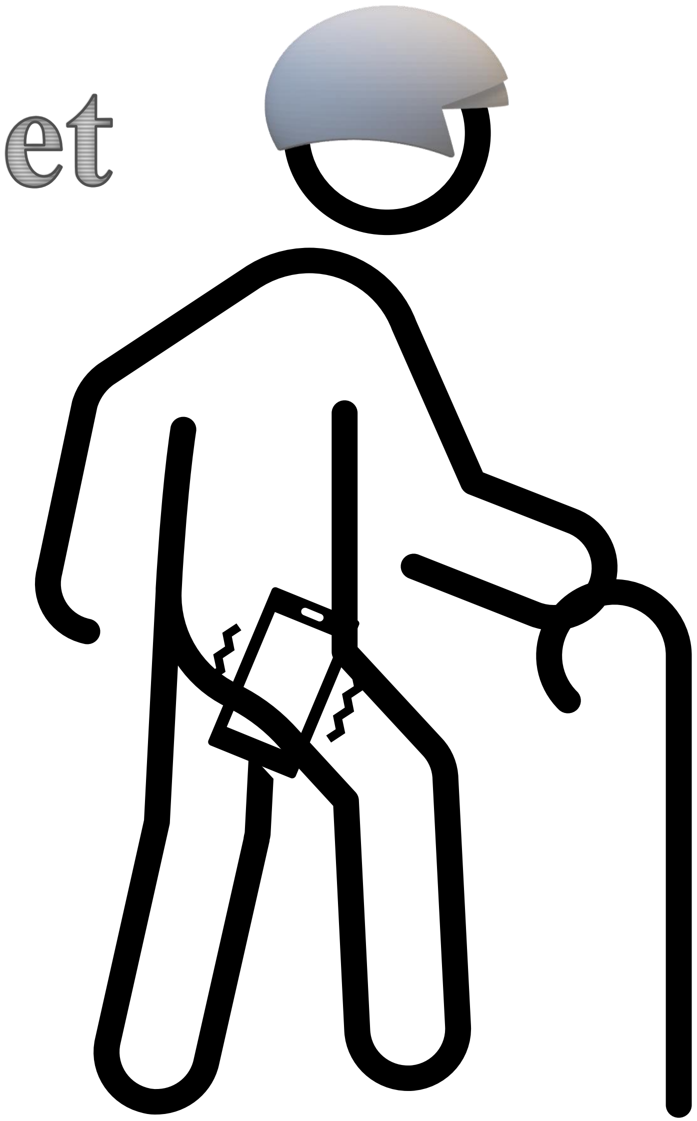
APP

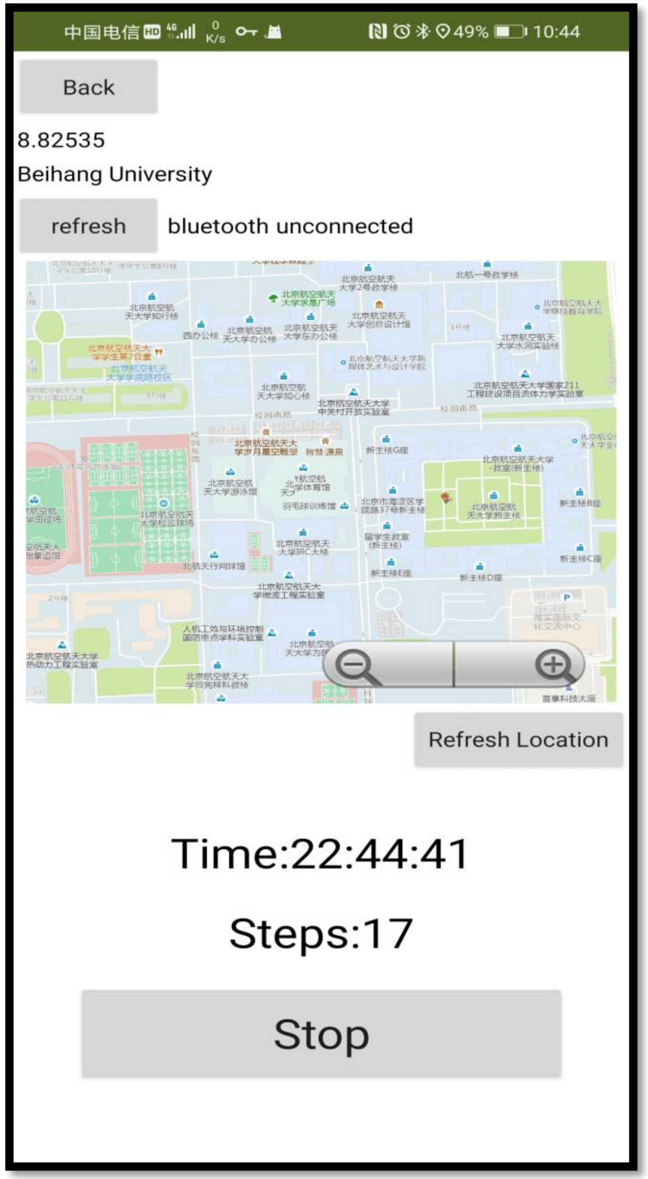


The two part of our system

helmet

APP





中国电信 49% 10:44

Back

8.82535

Beihang University

refresh bluetooth unconnected

Refresh Location

Time:22:44:41

Steps:17

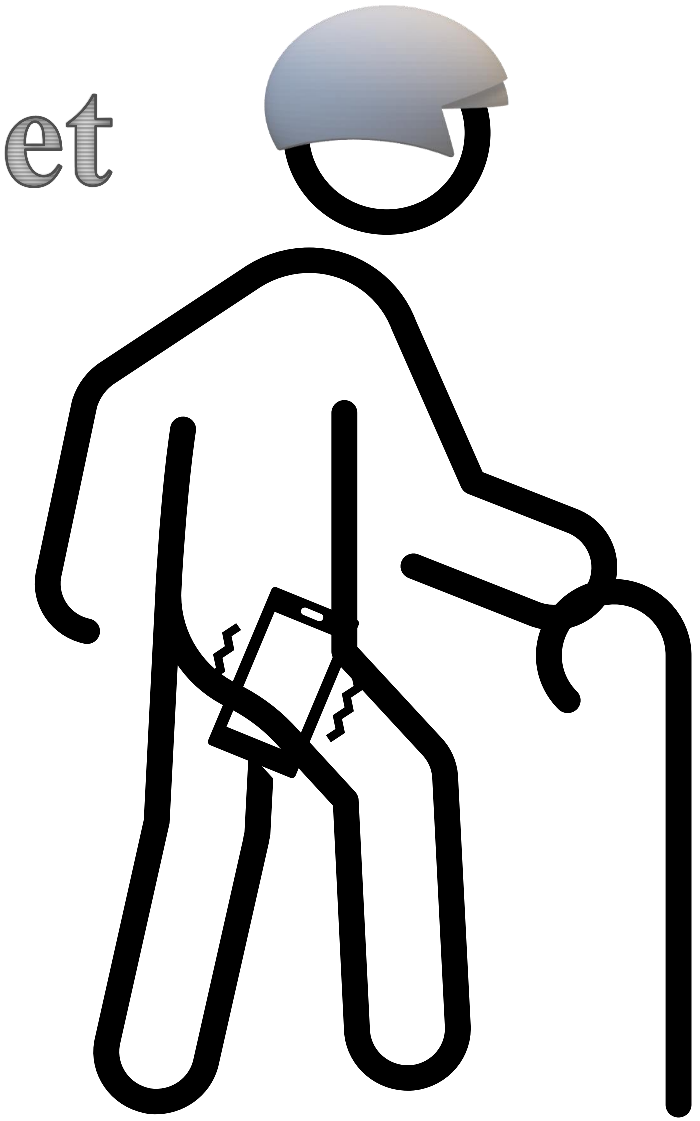
Stop

APP

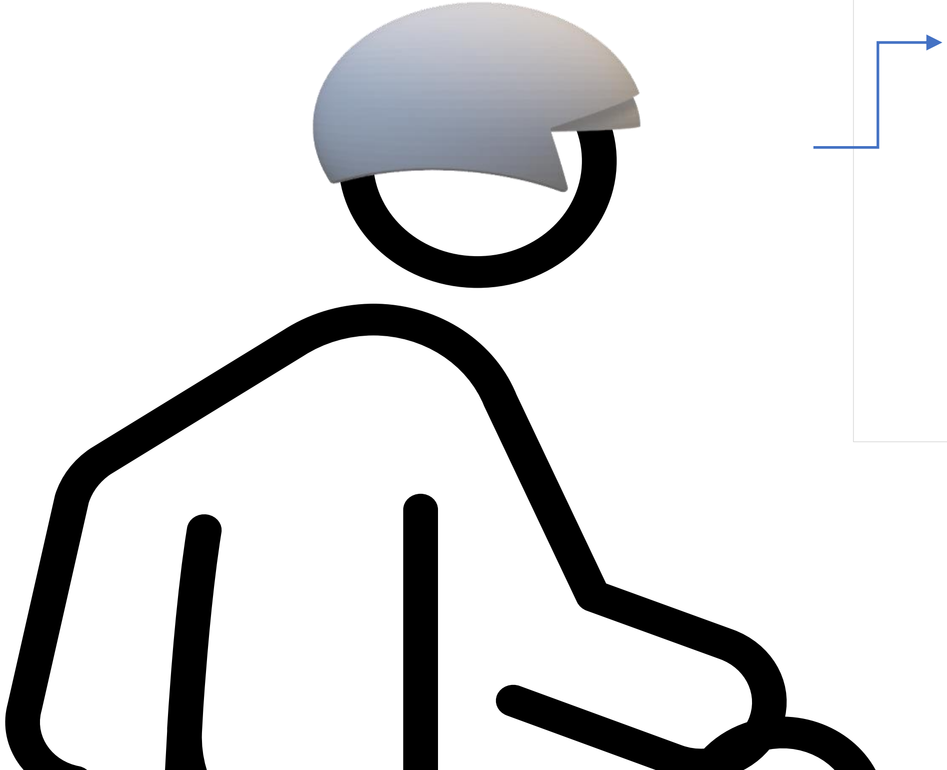
The two part of our project

helmet

APP



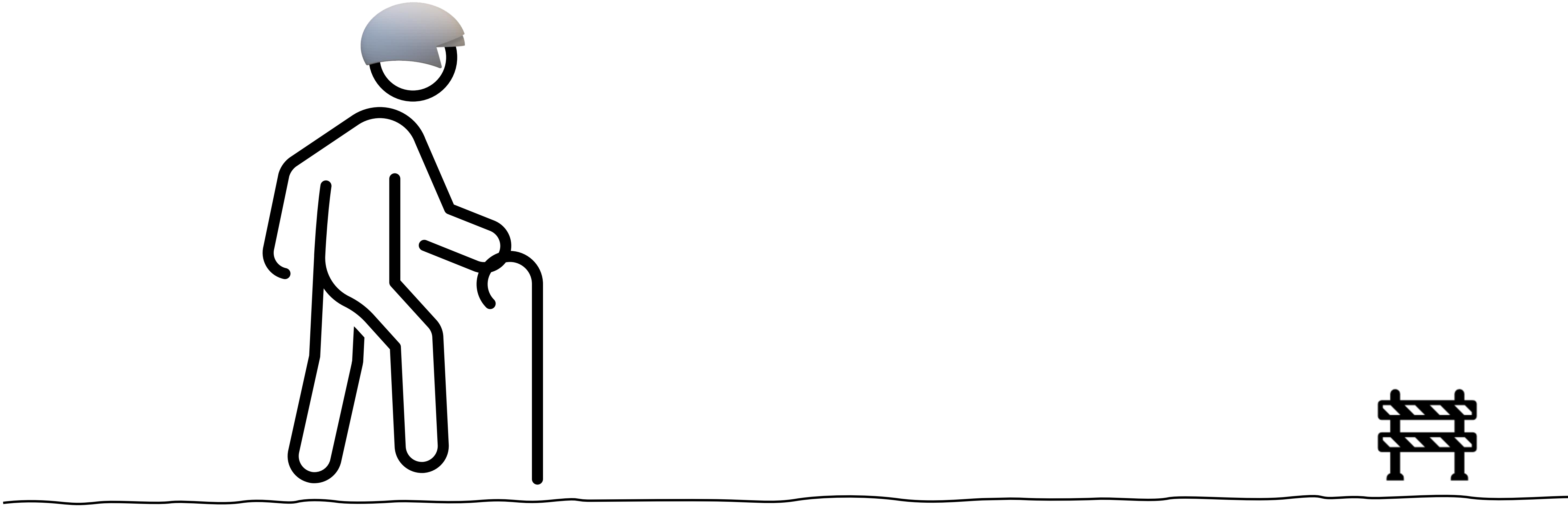
helmet



The structure of the helmet

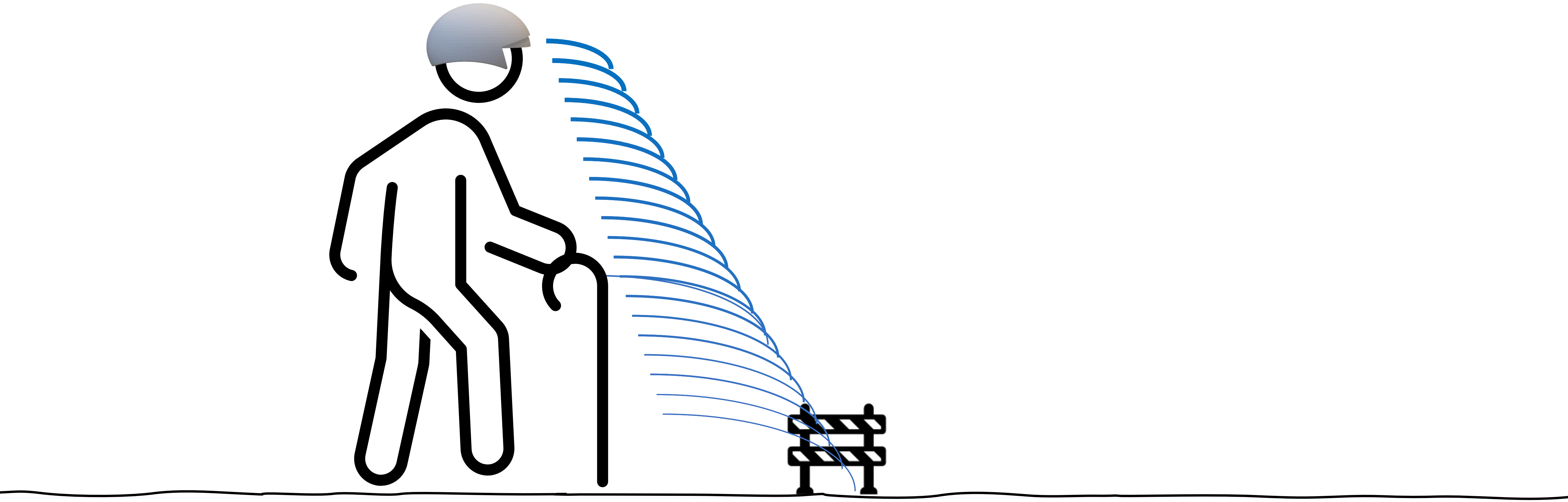
The structure of the helmet

Our project can help user avoid obstacle!

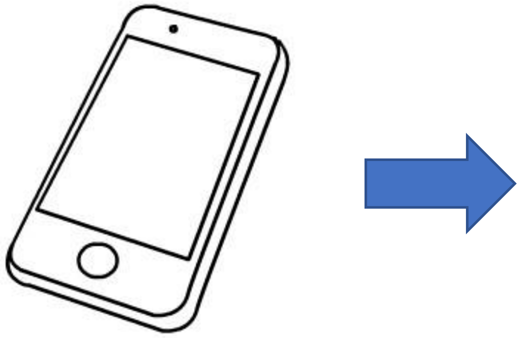


THE DISTANCE & DIRECTION

---ultrasonic detector



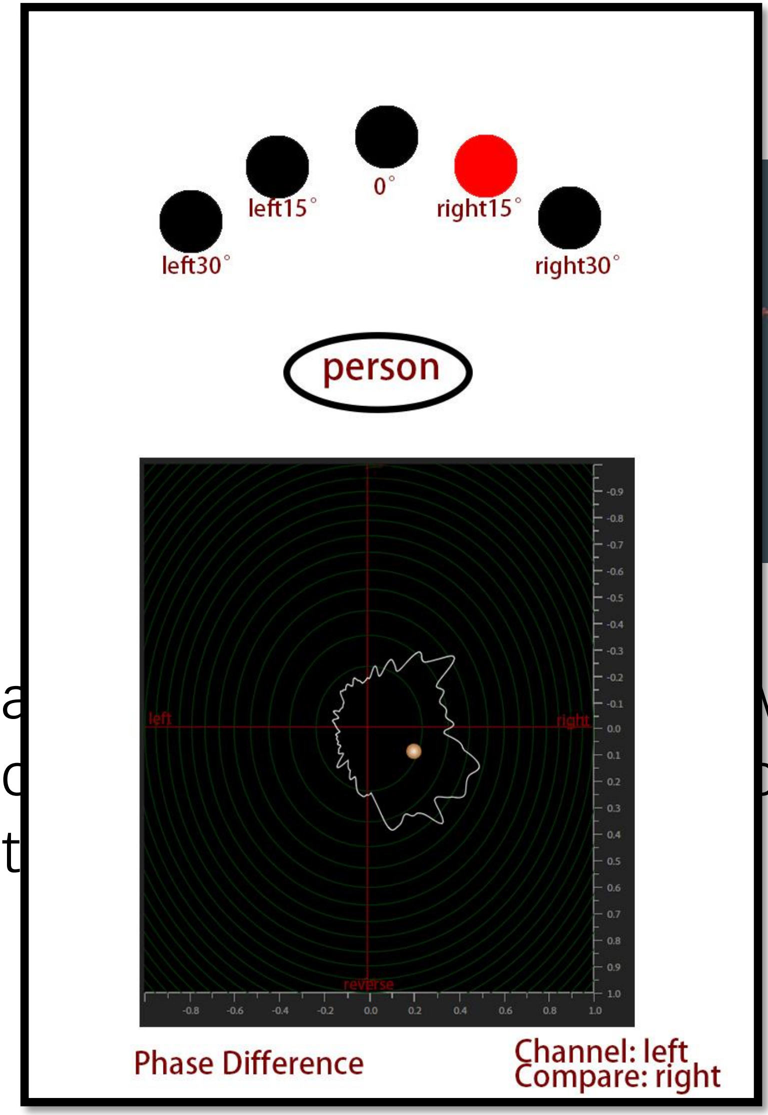
Earphone （Dynamic Stereo Sound）



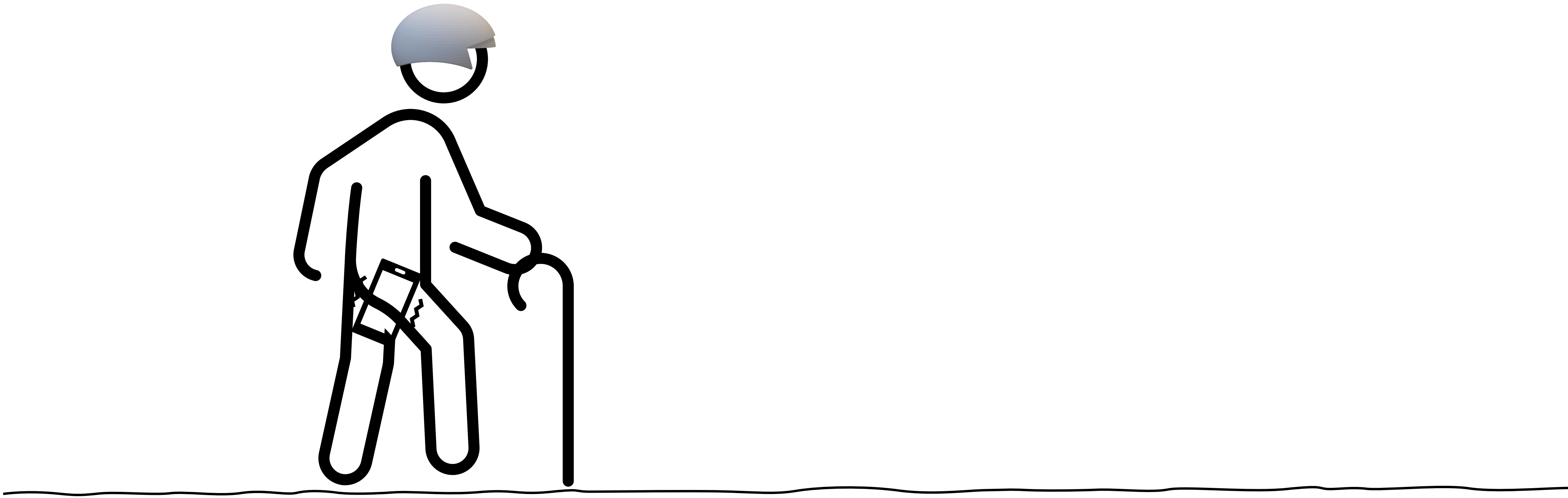
THE DISTANCE & DIRECTION

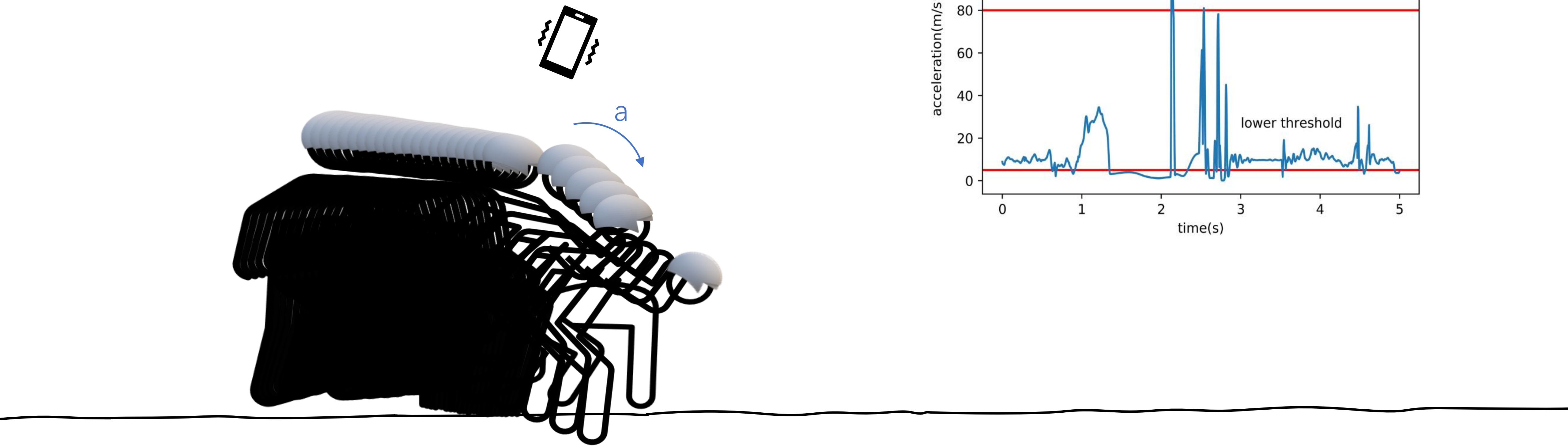
---ultrasonic detector

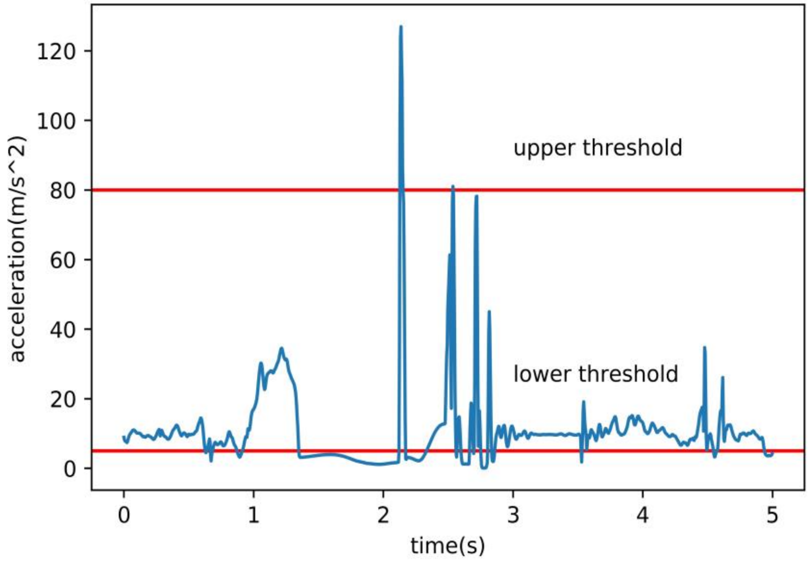
Huma wn by the
sound ponding
direct

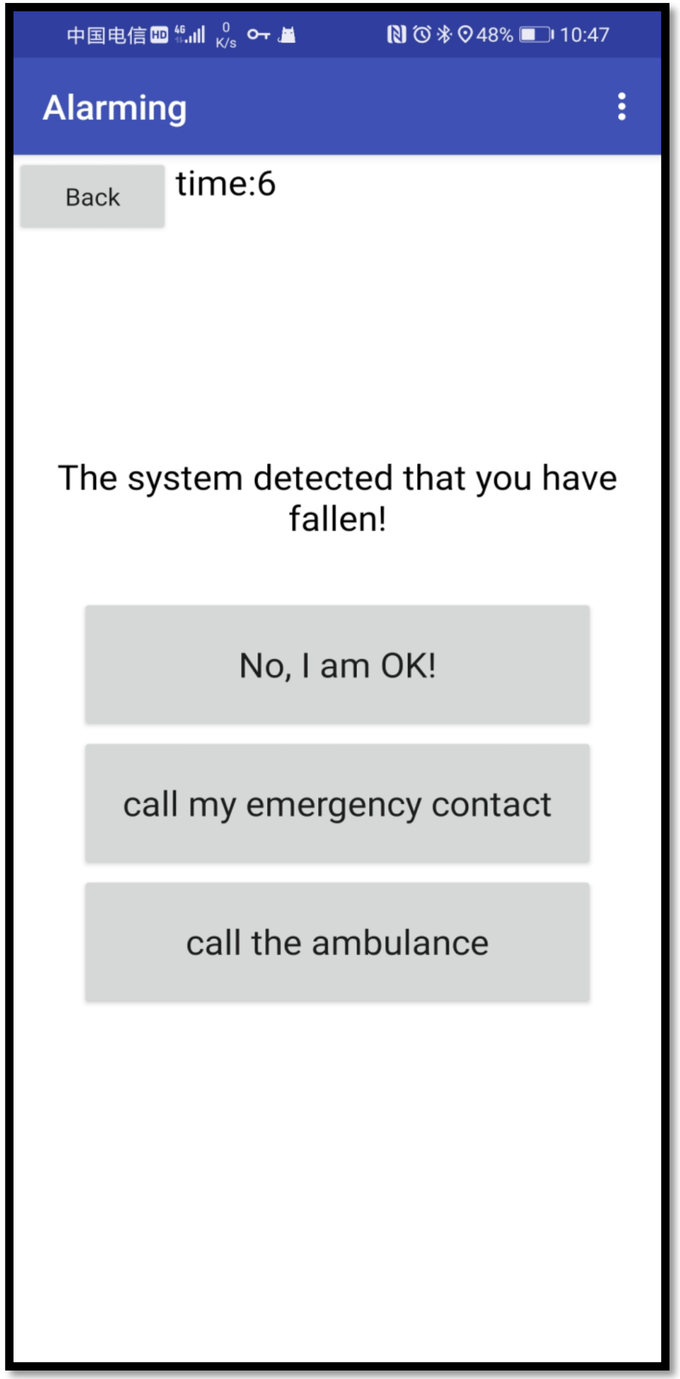


Our project can also deal with emergency !




a


acceleration(m/s^2)
0
20
40
60
80
100
120
time(s)
0
1
2
3
4
5
upper threshold
lower threshold



Alarming

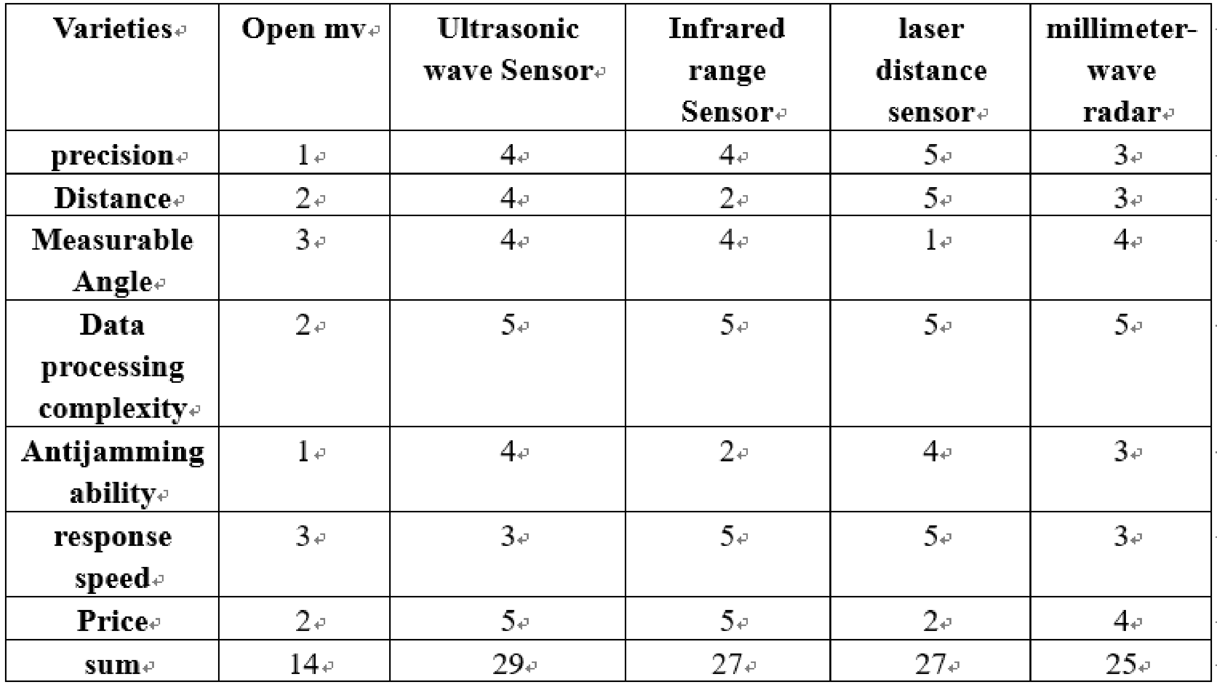
time:6

The system detected that you have fallen!

5· Subsystem Introduction - Obstacle Detecting

5 Intro - Obstacle Detecting

Ultrasonic sensors



Varieties	Open mv	Ultrasonic wave Sensor	Infrared range Sensor	laser distance sensor	millimeter-wave radar
precision	1	4	4	5	3
Distance	2	4	2	5	3
Measurable Angle	3	4	4	1	4
Data processing complexity	2	5	5	5	5
Antijamming ability	1	4	2	4	3
response speed	3	3	5	5	3
Price	2	5	5	2	4
sum	14	29	27	27	25

1. Enough performance for our project

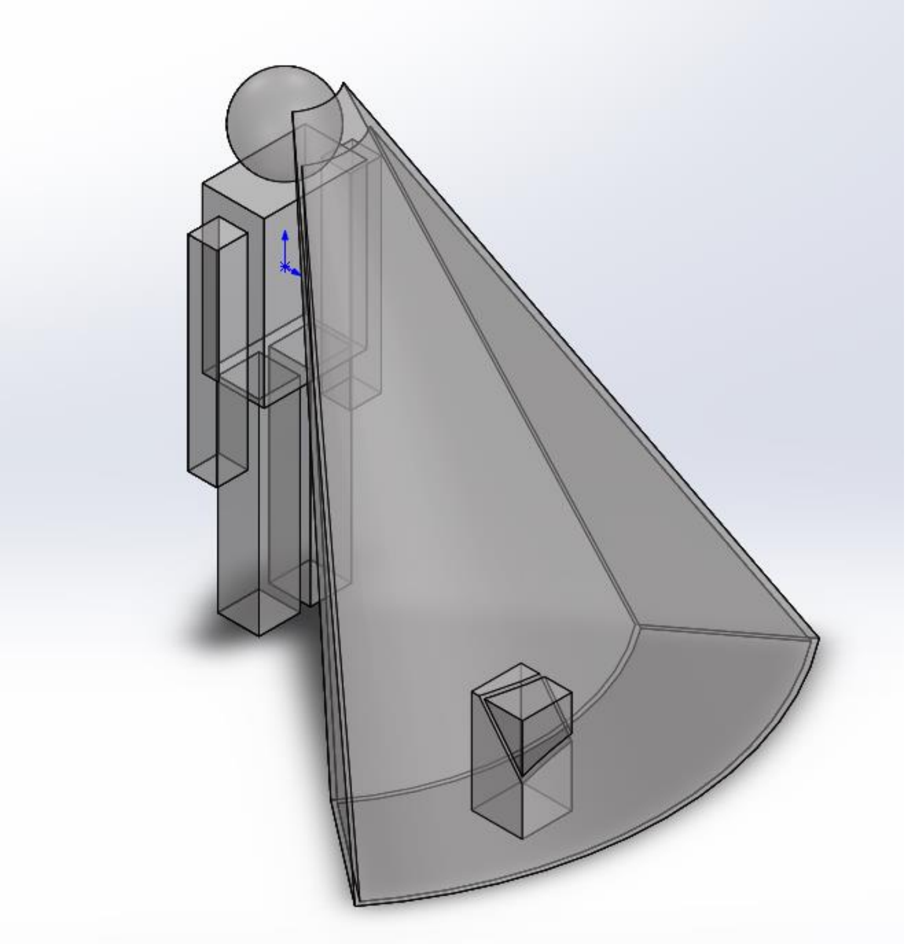
2. cheap

3. easy

5

Intro - Obstacle Detecting

Ultrasonic sensors



Detecting Range Model

Attributes:

height:175cm

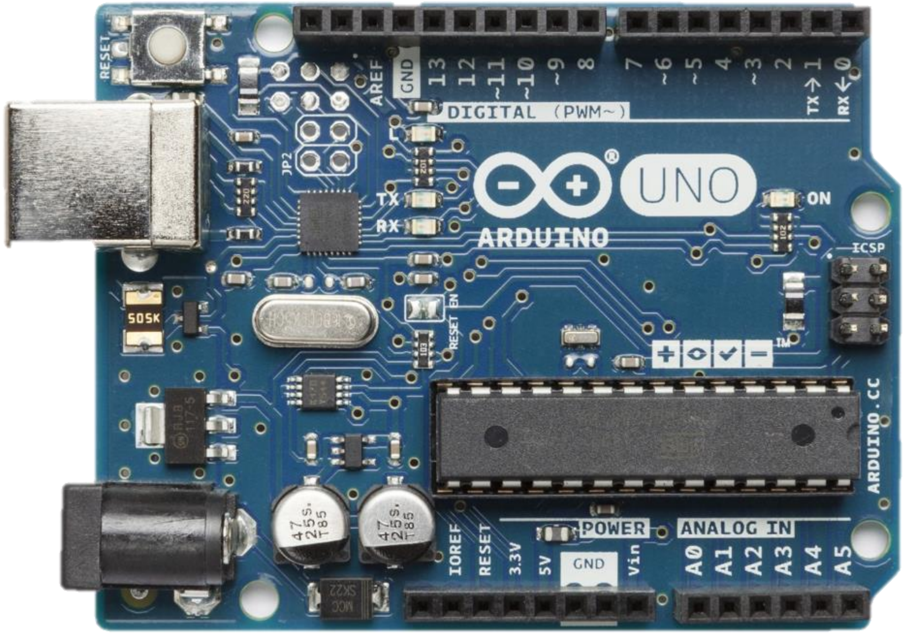
angle with respect to horizontal:30°

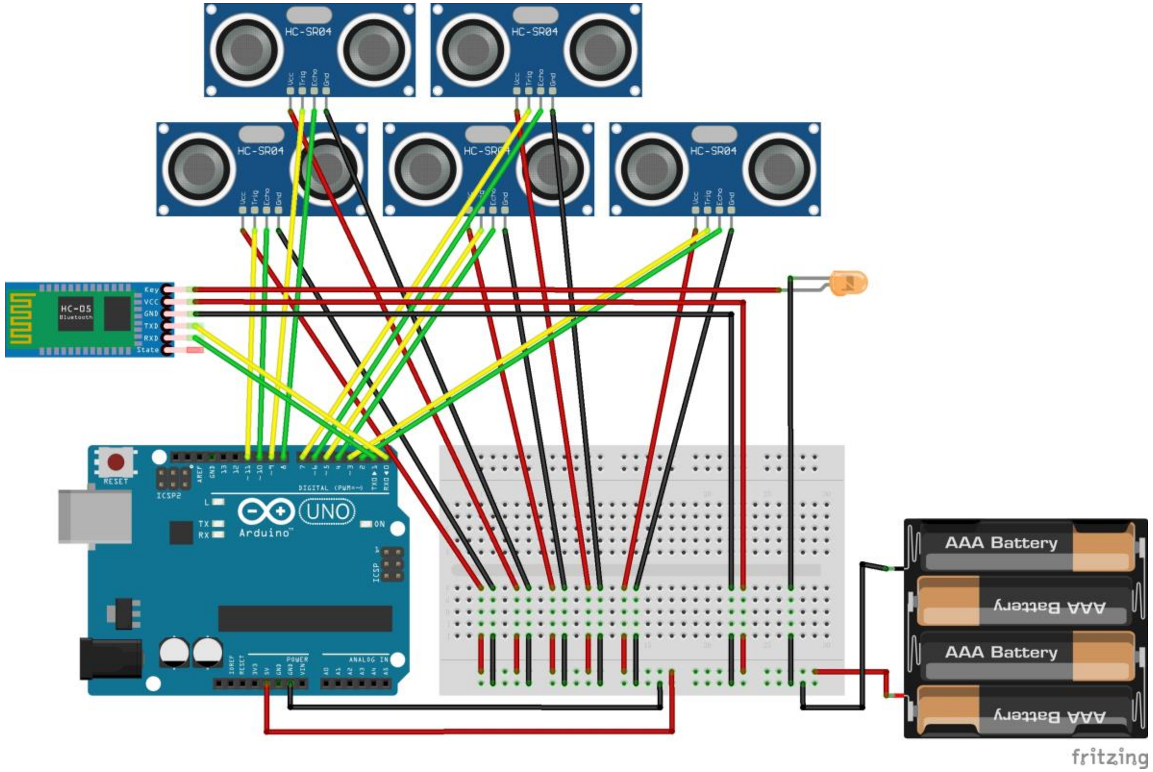
Argument:15°

Range:100cm

5 Intro - Obstacle Detecting

Controller: Arduino





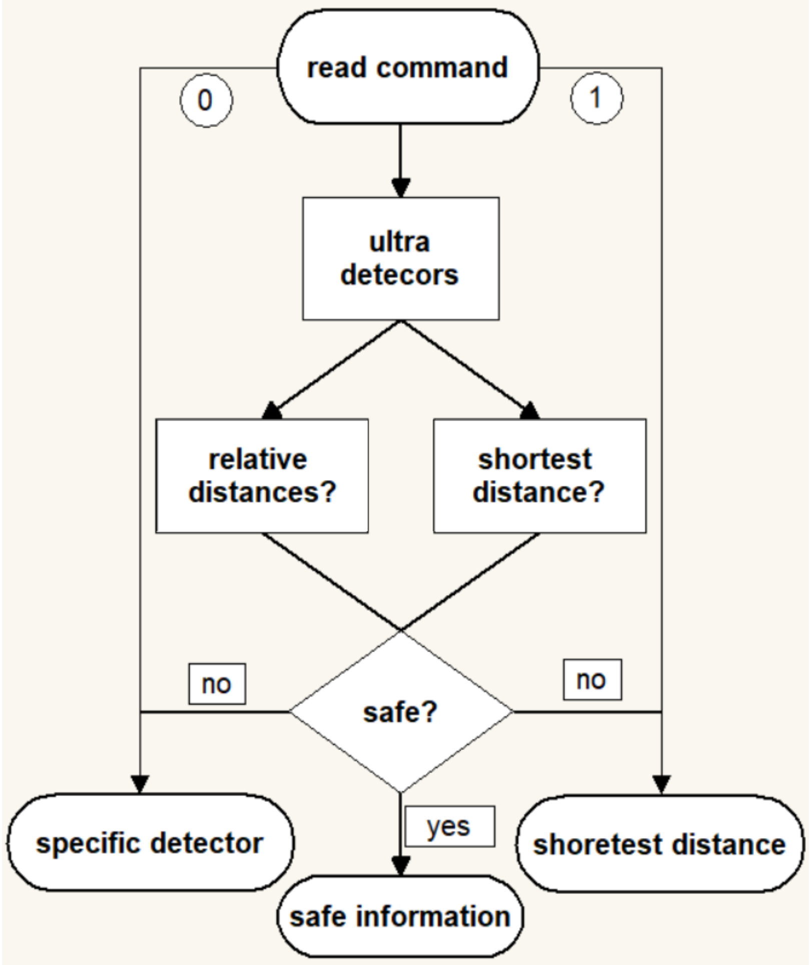
1.Cheap 2.Easy 3.Experienced

Circuit Diagram

5

Intro - Obstacle Detecting

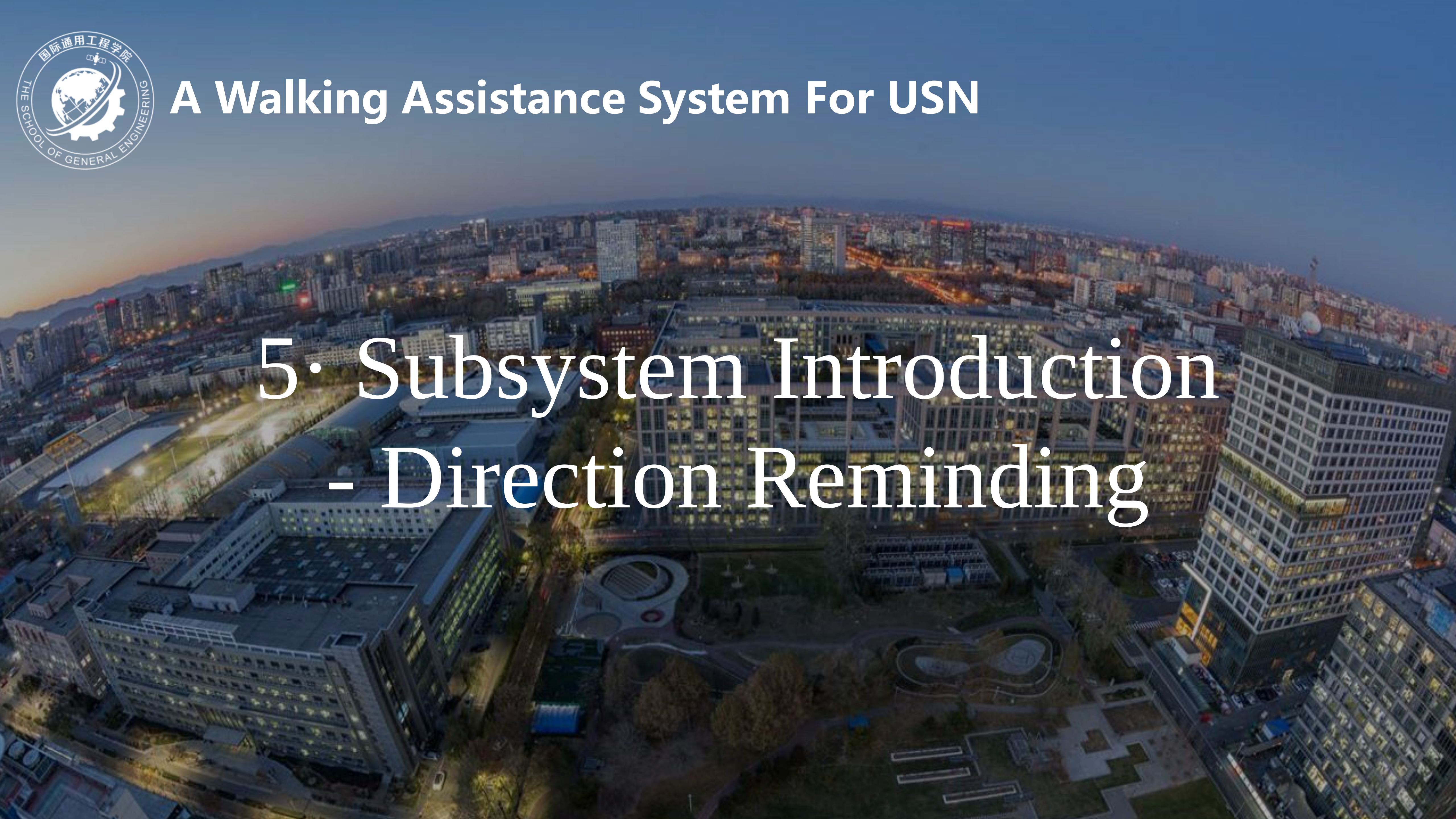
Algorithm



Two parameters

slope

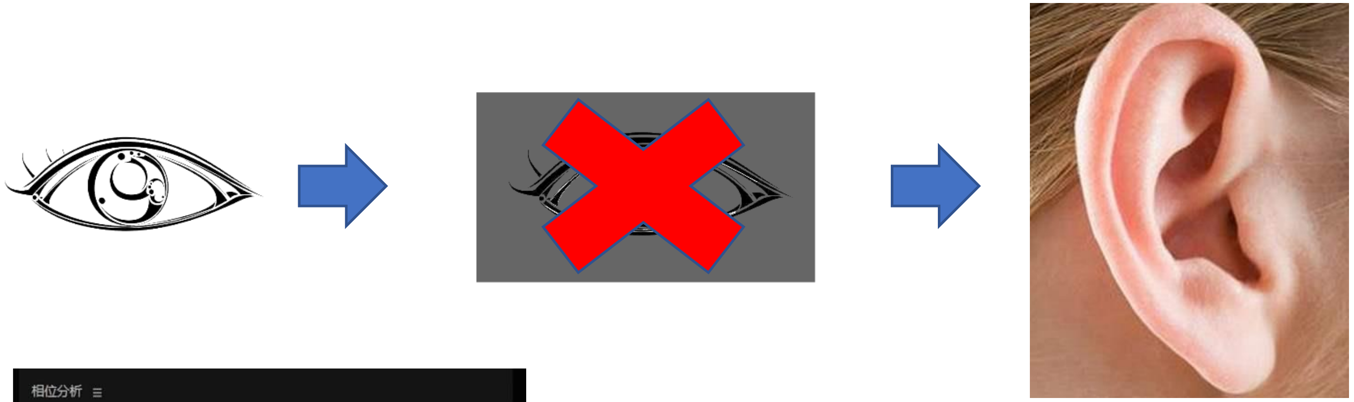
table



5· Subsystem Introduction - Direction Reminding

5 Direction Reminding - Reason

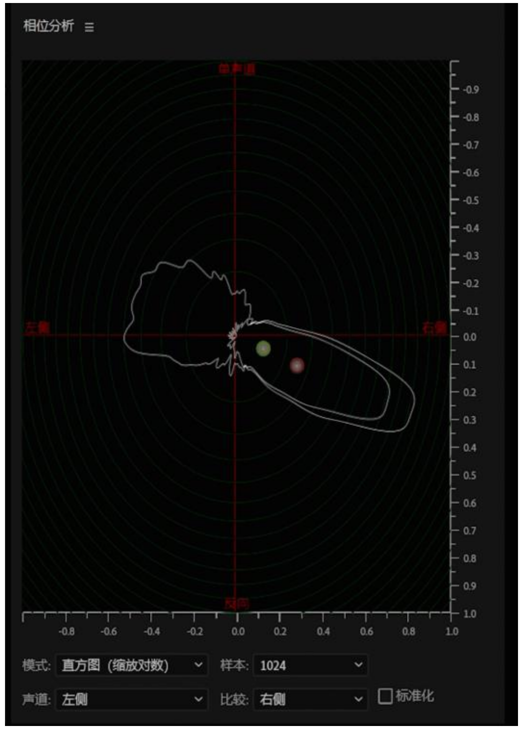
Perceptual Compensation



+

MOVE

=



相位分析 ≡

模式: 直方图（缩放对数） 样本: 1024

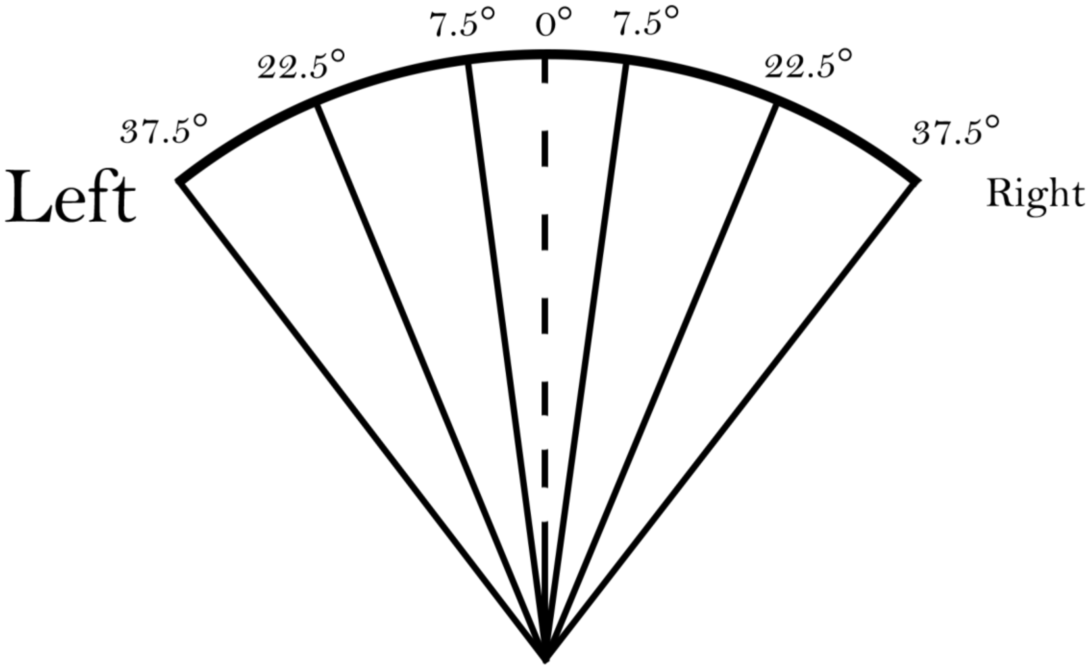
声道: 左侧 比较: 右侧 □标准化

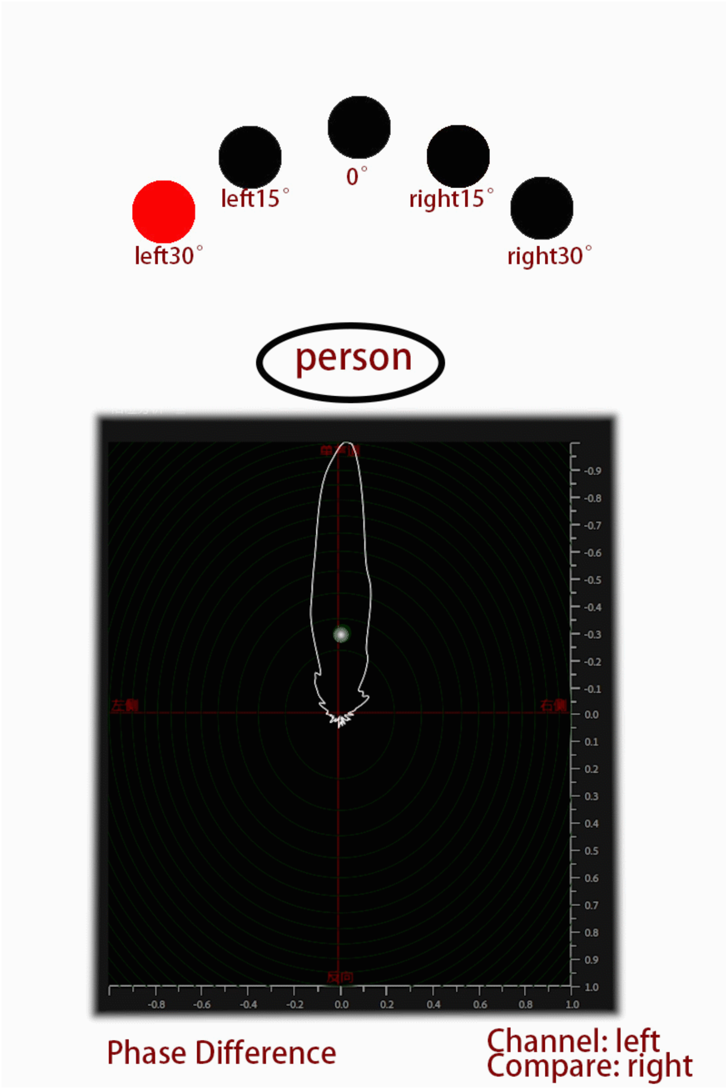
Phase Difference

5 Direction Reminding – Editing and working

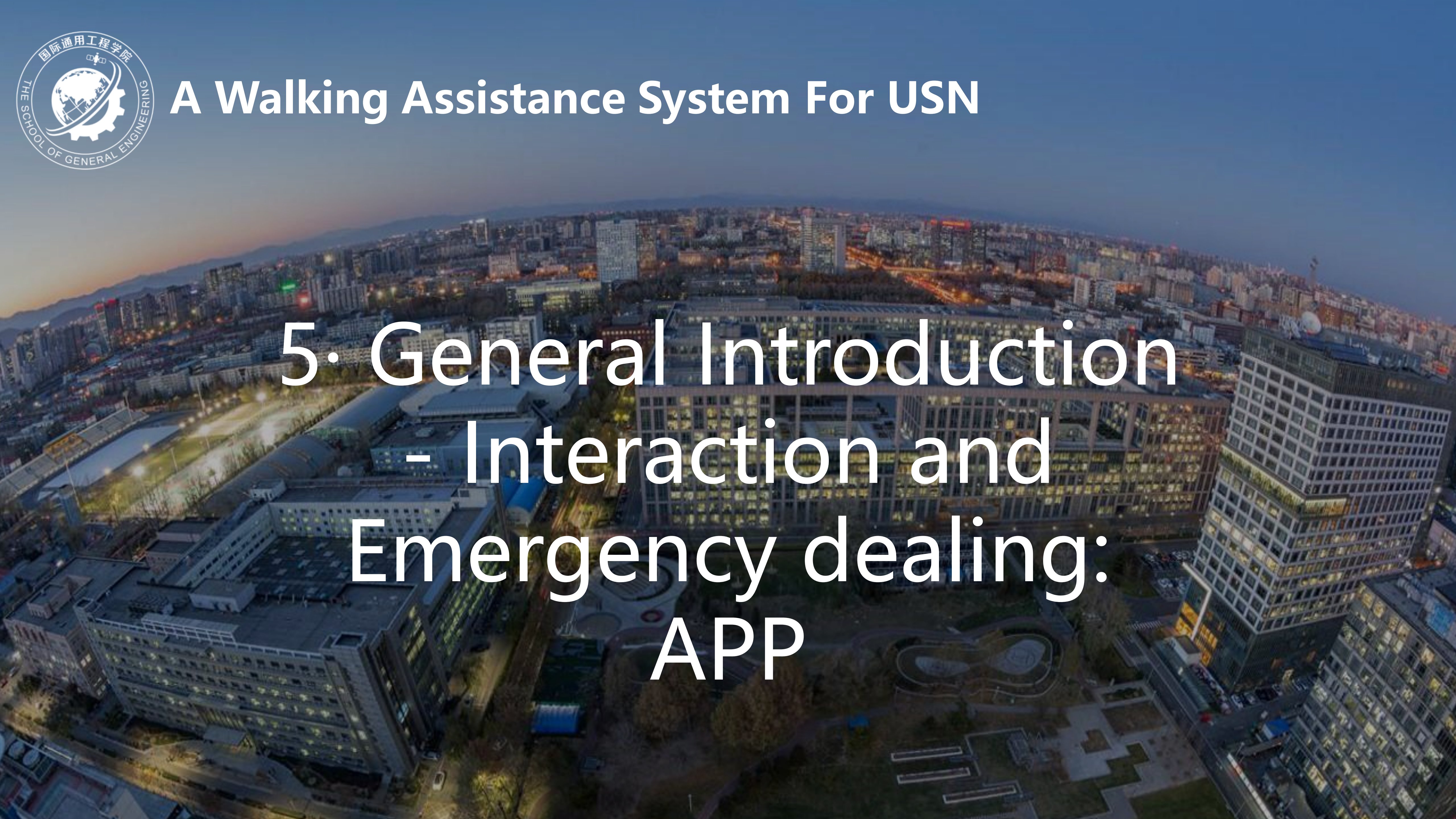
The resolution of human ears is

10°~15°





Work Process

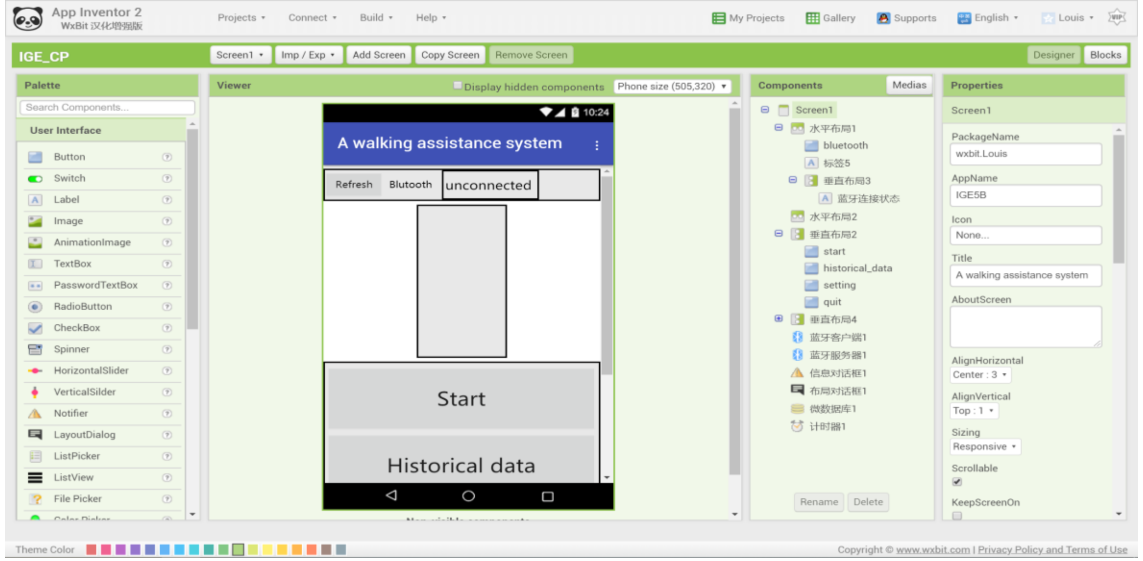


5· General Introduction - Interaction and Emergency dealing: APP

5 Interaction and Emergency Dealing - APP

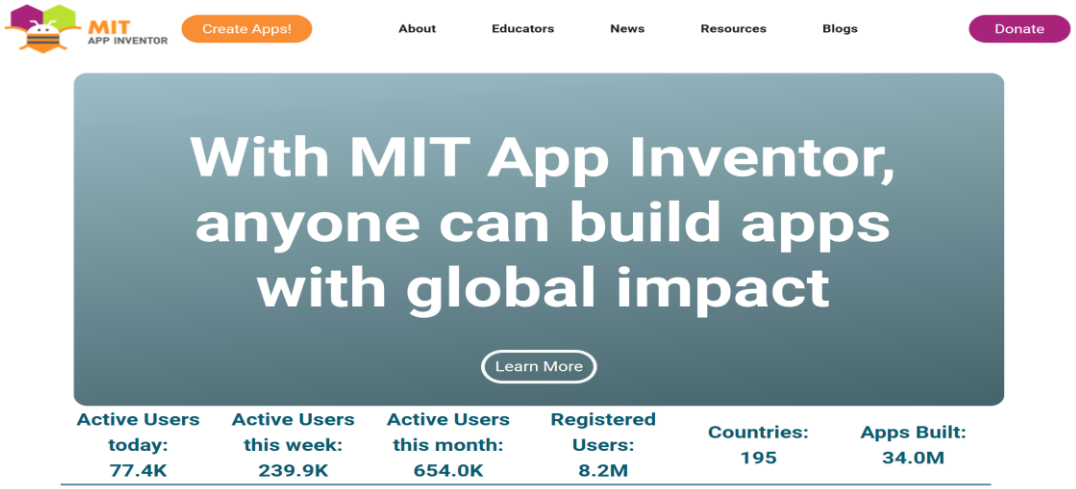
Why App?

- Cellphone has acceleration sensors(used for fall detecting)
- Everybody has cellphone
- Always carrying around a mobile phone
- Easy to use



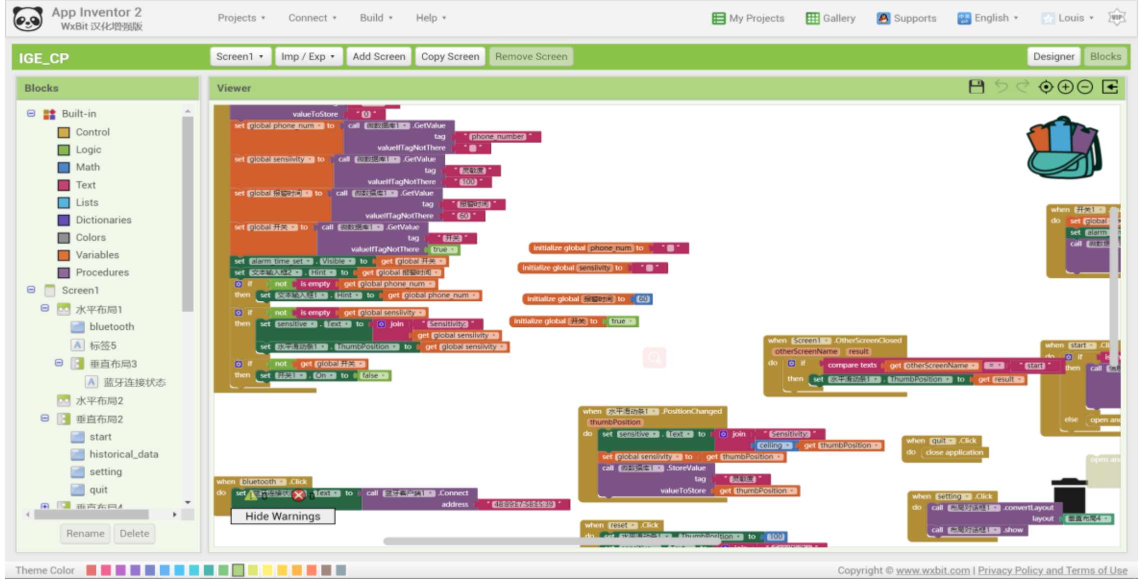
PackageName: wxbit.Louis
AppName: IGE5B
Icon: None...
Title: A walking assistance system
AboutScreen:
AlignHorizontal: Center : 3
AlignVertical: Top : 1
Sizing: Responsive
Scrollable: ☑
KeepScreenOn: □

Programming Platform： App Inventor

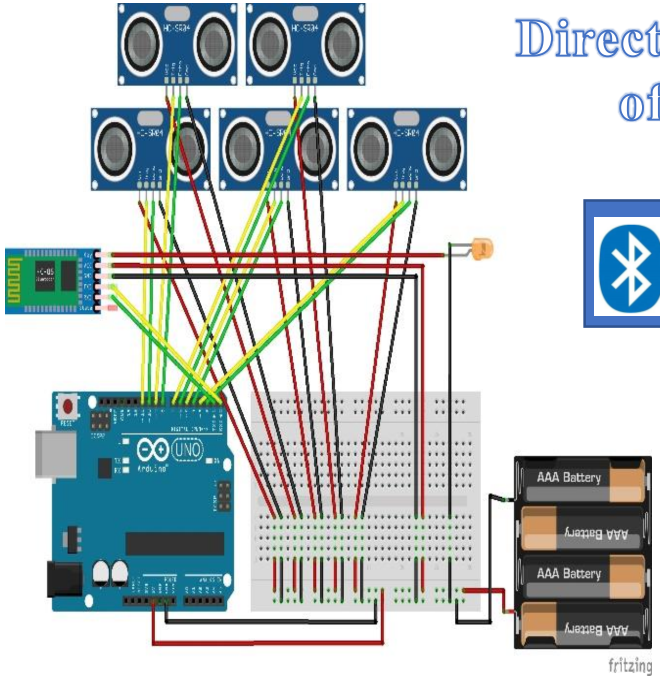


With MIT App Inventor, anyone can build apps with global impact

Active Users today:	Active Users this week:	Active Users this month:	Registered Users:	Countries:	Apps Built:
77.4K	239.9K	654.0K	8.2M	195	34.0M

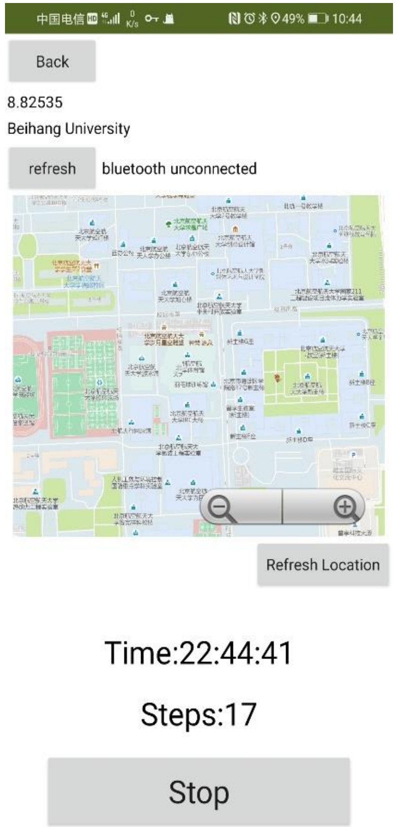


5 APP—Information Transition



Direction and Distance of the Obstacle

Bluetooth™



Back

8.82535

Beihang University

refresh bluetooth unconnected

Refresh Location

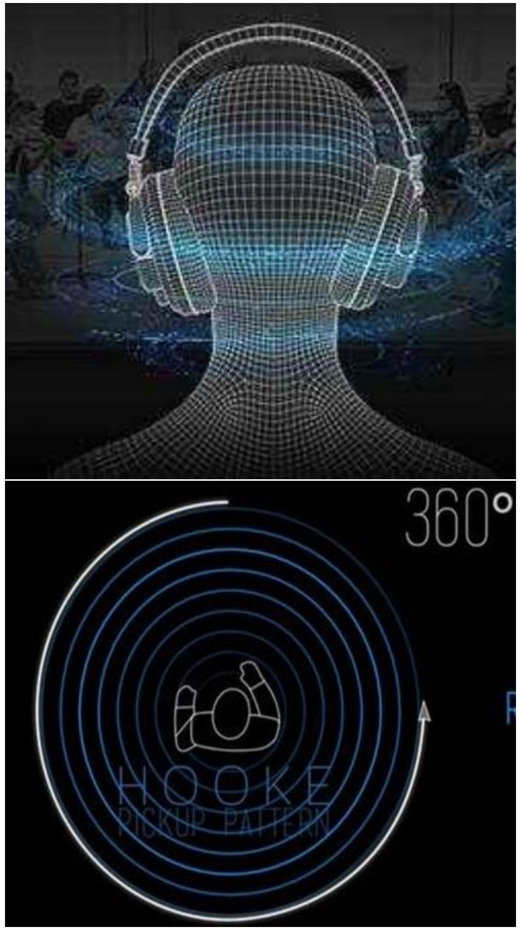
Time:22:44:41

Steps:17

Stop

play

Earphone



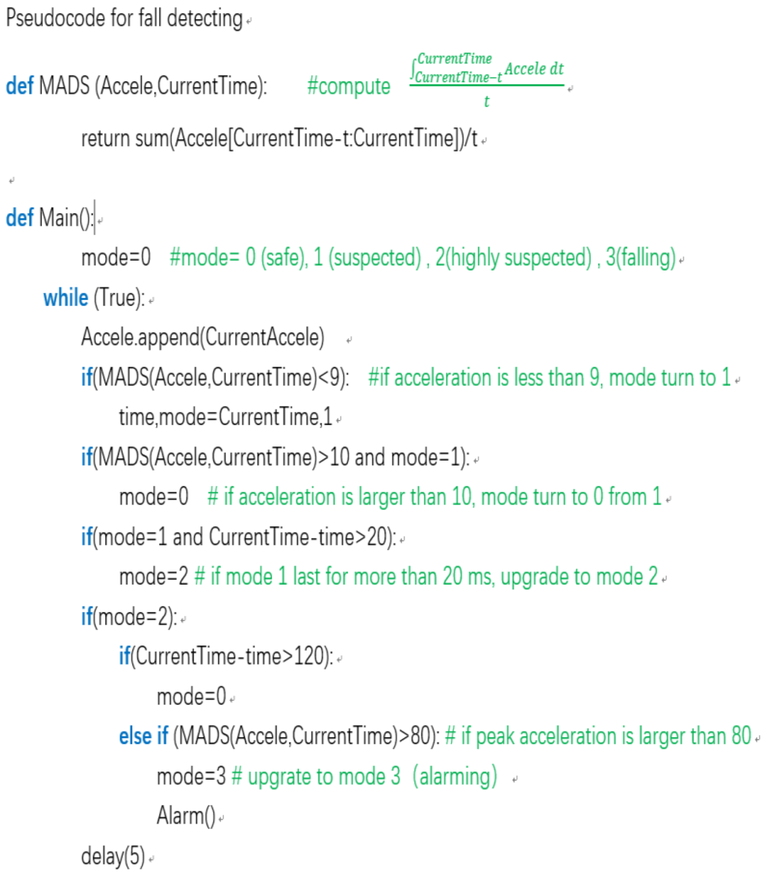
Detector

App

Stereo Sound

5 APP—Fall Detecting

Fall detecting algorithm

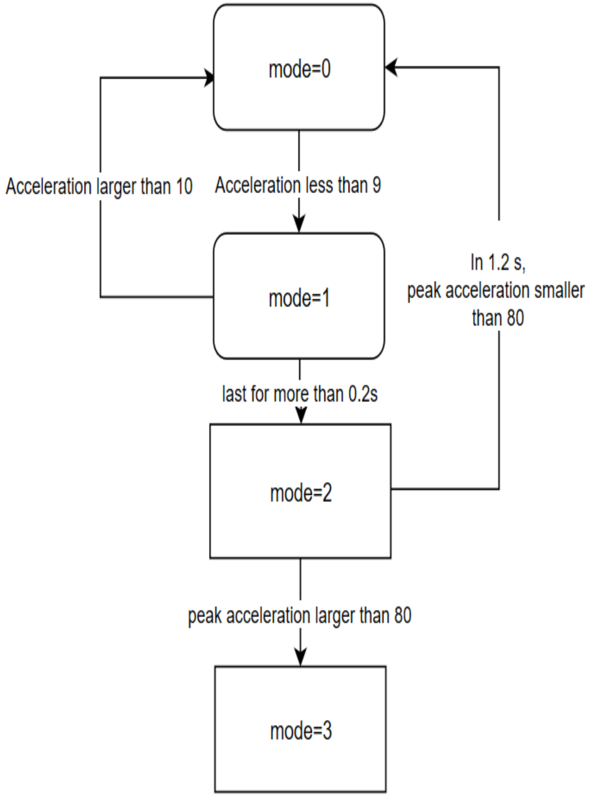


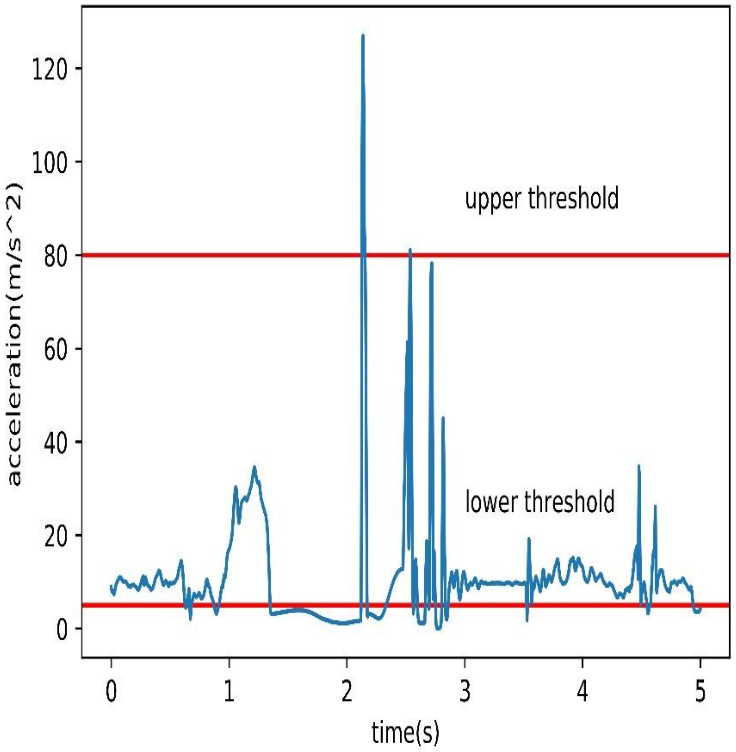
Pseudocode for fall detecting

```
def MADS (Accele,CurrentTime):      #compute $\frac{\int_{CurrentTime-t}^{CurrentTime} Accele\,dt}{t}$
    return sum(Accele[CurrentTime-t:CurrentTime])/t

def Main():
    mode=0   #mode= 0 (safe), 1 (suspected) , 2(highly suspected) , 3(falling)
  while (True):
    Accele.append(CurrentAccele)
    if(MADS(Accele,CurrentTime)<9):   #if acceleration is less than 9, mode turn to 1
      time,mode=CurrentTime,1
    if(MADS(Accele,CurrentTime)>10 and mode=1):
      mode=0   # if acceleration is larger than 10, mode turn to 0 from 1
    if(mode=1 and CurrentTime-time>20):
      mode=2 # if mode 1 last for more than 20 ms, upgrade to mode 2
    if(mode=2):
      if(CurrentTime-time>120):
        mode=0
      else if (MADS(Accele,CurrentTime)>80): # if peak acceleration is larger than 80
        mode=3 # upgrate to mode 3 (alarming)
        Alarm()
    delay(5)
```

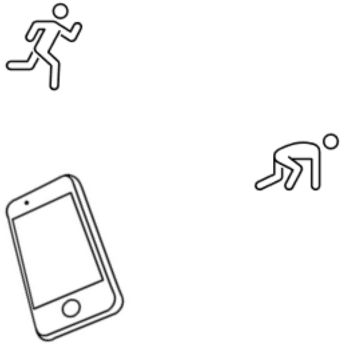
Lower Threshold: Weightlessness Before Fall Down

Upper Threshold: Peak Acceleration When Hit The Ground

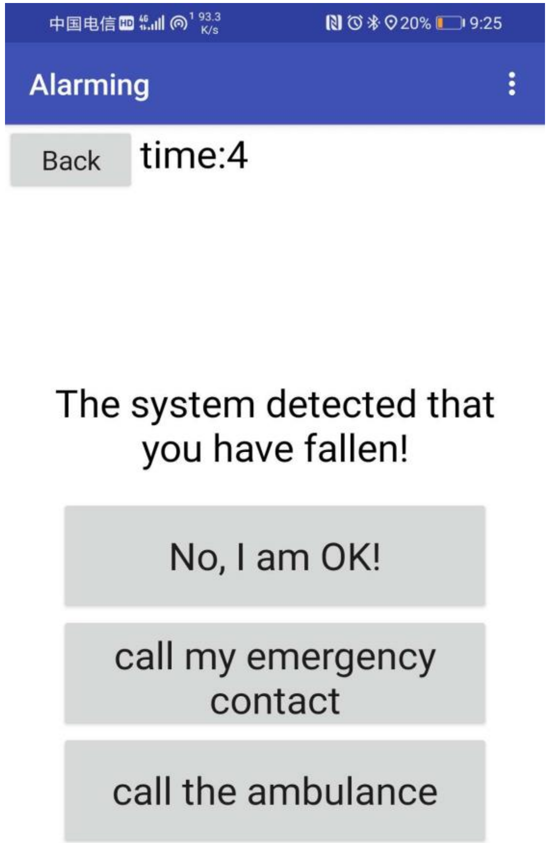

mode=0
Acceleration larger than 10
Acceleration less than 9
mode=1
In 1.2 s,
peak acceleration smaller than 80
last for more than 0.2s
mode=2
peak acceleration larger than 80
mode=3



5 APP—Alarming System



Falling



time:4

The system detected that you have fallen!

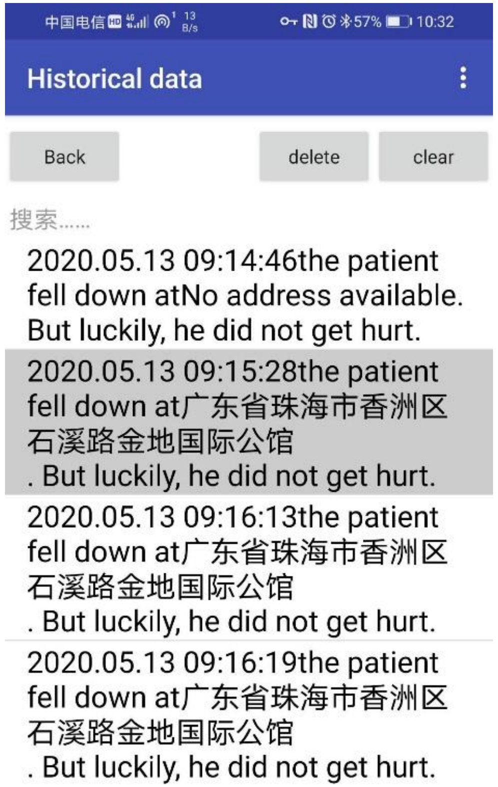
Did not get hurt

☎ Ambulance or families

💬

If no response after 60s

5 APP—Historical Data



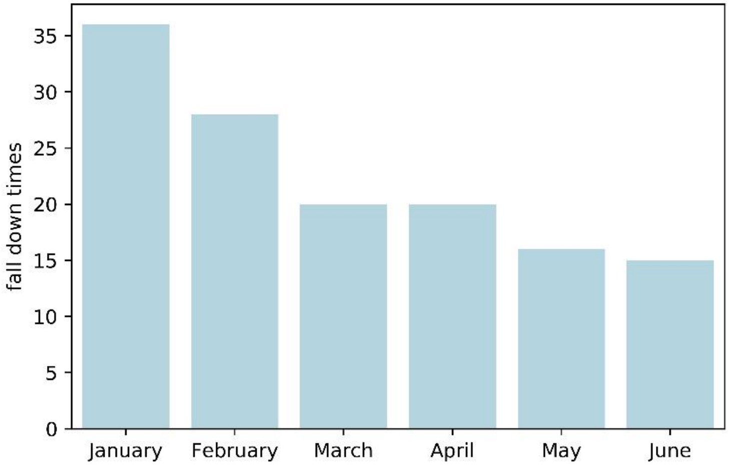
2020.05.13 09:14:46the patient fell down atNo address available. But luckily, he did not get hurt.

2020.05.13 09:15:28the patient fell down at广东省珠海市香洲区石溪路金地国际公馆
. But luckily, he did not get hurt.

2020.05.13 09:16:13the patient fell down at广东省珠海市香洲区石溪路金地国际公馆
. But luckily, he did not get hurt.

2020.05.13 09:16:19the patient fell down at广东省珠海市香洲区石溪路金地国际公馆
. But luckily, he did not get hurt.

Where?
When?
Did he get hurt?



Count the Fall Down Times

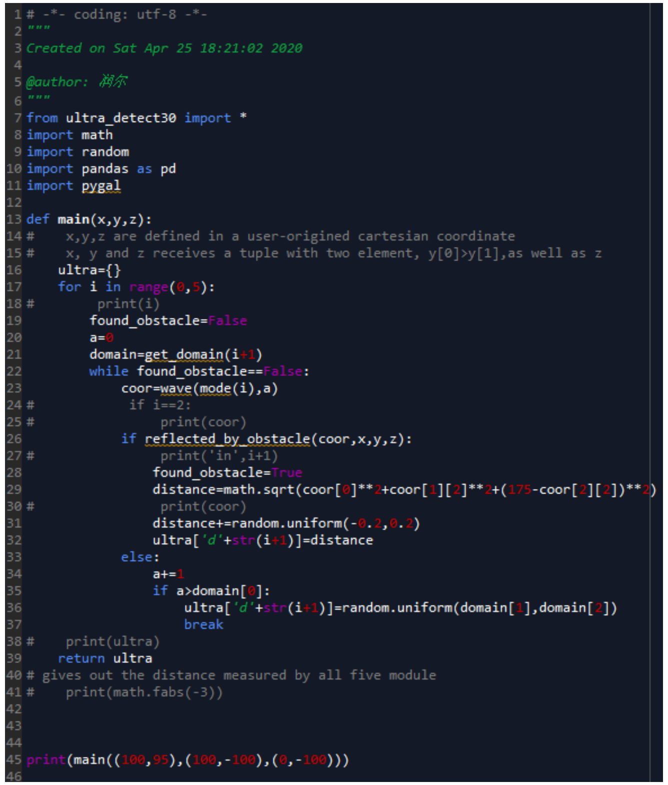
Help to Analyze

State of illness

6· Proof of Concept

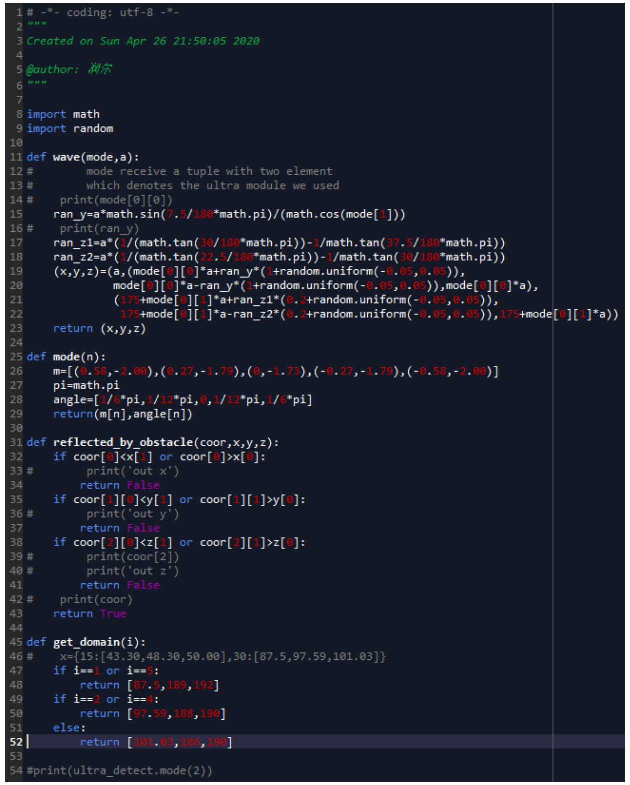
6 Simulations – Obstacle Detecting

□ **Platform:**	Spyder
□ **Language:**	Python
□ **Components:**	5 ultra-detectors & a virtual space
□ **Parameters:**	air density & wind speed



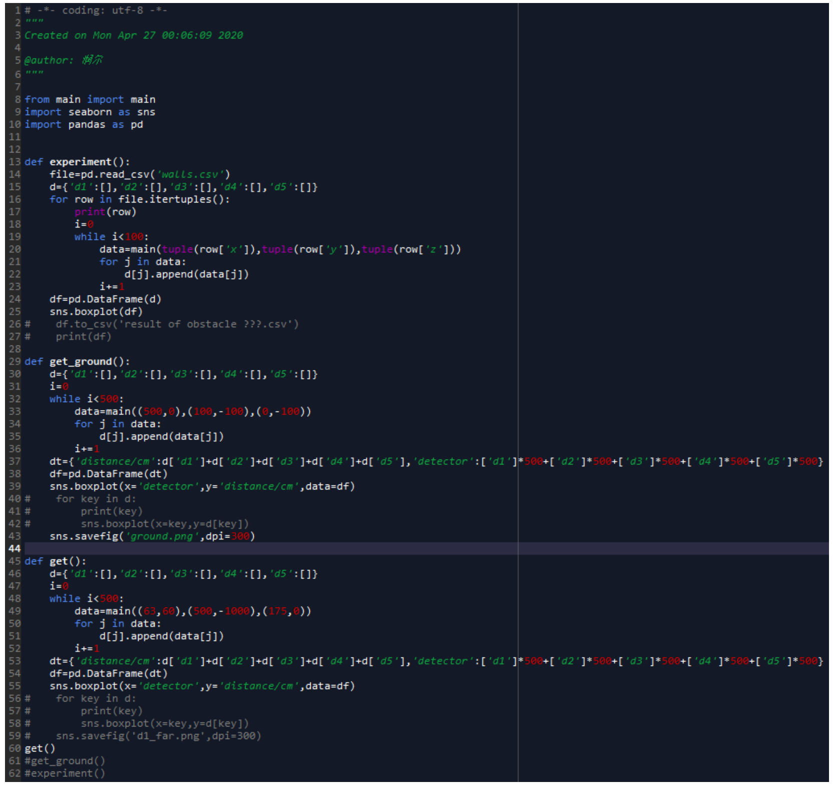
```python
# -*- coding: utf-8 -*-
"""
Created on Sat Apr 25 18:21:02 2020

@author: 润尔
"""
from ultra_detect30 import *
import math
import random
import pandas as pd
import pygal

def main(x,y,z):
#    x,y,z are defined in a user-origined cartesian coordinate
#    x, y and z receives a tuple with two element, y[0]>y[1],as well as z
    ultra={}
    for i in range(0,5):
#        print(i)
        found_obstacle=False
        a=0
        domain=get_domain(i+1)
        while found_obstacle==False:
            coor=wave(mode(i),a)
#            if i==2:
#                print(coor)
            if reflected_by_obstacle(coor,x,y,z):
#                print('in',i+1)
                found_obstacle=True
                distance=math.sqrt(coor[0]**2+coor[1][2]**2+(175-coor[2][2])**2)
#                print(coor)
                distance+=random.uniform(-0.2,0.2)
                ultra['d'+str(i+1)]=distance
            else:
                a+=1
                if a>domain[0]:
                    ultra['d'+str(i+1)]=random.uniform(domain[1],domain[2])
                    break
#    print(ultra)
    return ultra
# gives out the distance measured by all five module
#    print(math.fabs(-3))



print(main((100,95),(100,-100),(0,-100)))

```

Main



```python
# -*- coding: utf-8 -*-
"""
Created on Sun Apr 26 21:50:05 2020

@author: 润尔
"""

import math
import random

def wave(mode,a):
#        mode receive a tuple with two element
#        which denotes the ultra module we used
#    print(mode[0][0])
    ran_y=a*math.sin(7.5/180*math.pi)/(math.cos(mode[1]))
#    print(ran_y)
    ran_z1=a*(1/(math.tan(30/180*math.pi))-1/math.tan(37.5/180*math.pi))
    ran_z2=a*(1/(math.tan(22.5/180*math.pi))-1/math.tan(30/180*math.pi))
    (x,y,z)=(a,(mode[0][0]*a+ran_y*(1+random.uniform(-0.05,0.05)),
             mode[0][0]*a-ran_y*(1+random.uniform(-0.05,0.05)),mode[0][0]*a),
             (175+mode[0][1]*a+ran_z1*(0.2+random.uniform(-0.05,0.05)),
              175+mode[0][1]*a-ran_z2*(0.2+random.uniform(-0.05,0.05)),175+mode[0][1]*a))
    return (x,y,z)

def mode(n):
    m=[(0.58,-2.00),(0.27,-1.79),(0,-1.73),(-0.27,-1.79),(-0.58,-2.00)]
    pi=math.pi
    angle=[1/6*pi,1/12*pi,0,1/12*pi,1/6*pi]
    return(m[n],angle[n])

def reflected_by_obstacle(coor,x,y,z):
    if coor[0]<x[1] or coor[0]>x[0]:
#        print('out x')
        return False
    if coor[1][0]<y[1] or coor[1][1]>y[0]:
#        print('out y')
        return False
    if coor[2][0]<z[1] or coor[2][1]>z[0]:
#        print(coor[2])
#        print('out z')
        return False
#    print(coor)
    return True

def get_domain(i):
#    x={15:[43.30,48.30,50.00],30:[87.5,97.59,101.03]}
    if i==1 or i==5:
        return [87.5,189,192]
    if i==2 or i==4:
        return [97.59,188,190]
    else:
        return [101.03,186,190]

#print(ultra_detect.mode(2))
```

Functions

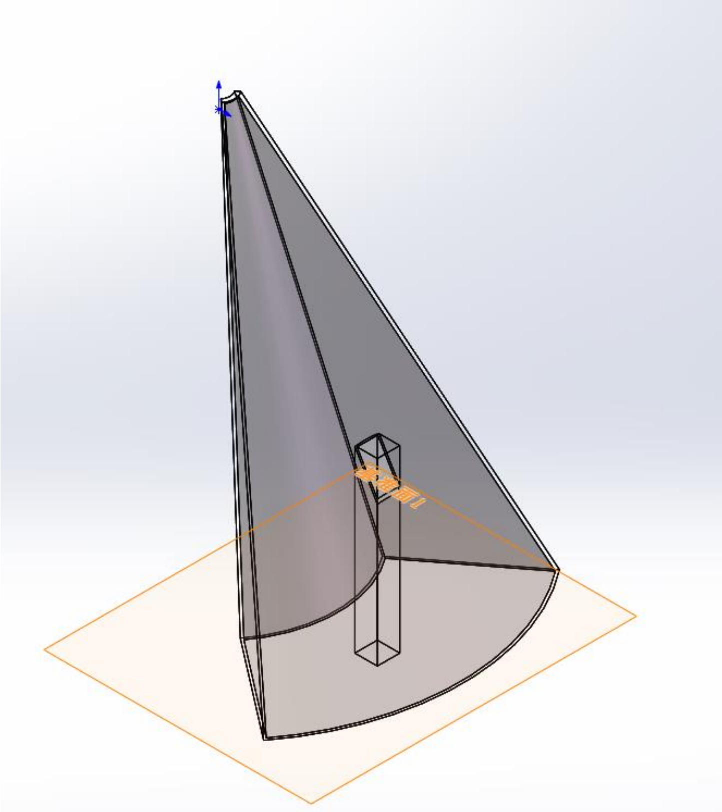


```python
# -*- coding: utf-8 -*-
"""
Created on Mon Apr 27 00:06:09 2020

@author: 润尔
"""

from main import main
import seaborn as sns
import pandas as pd


def experiment():
    file=pd.read_csv('walls.csv')
    d={'d1':[],'d2':[],'d3':[],'d4':[],'d5':[]}
    for row in file.itertuples():
        print(row)
        i=0
        while i<100:
            data=main(tuple(row['x']),tuple(row['y']),tuple(row['z']))
            for j in data:
                d[j].append(data[j])
            i+=1
    df=pd.DataFrame(d)
    sns.boxplot(df)
#    df.to_csv('result of obstacle ???.csv')
#    print(df)

def get_ground():
    d={'d1':[],'d2':[],'d3':[],'d4':[],'d5':[]}
    i=0
    while i<500:
        data=main((500,0),(100,-100),(0,-100))
        for j in data:
            d[j].append(data[j])
        i+=1
    dt={'distance/cm':d['d1']+d['d2']+d['d3']+d['d4']+d['d5'],'detector':['d1']*500+['d2']*500+['d3']*500+['d4']*500+['d5']*500}
    df=pd.DataFrame(dt)
    sns.boxplot(x='detector',y='distance/cm',data=df)
#    for key in d:
#        print(key)
#        sns.boxplot(x=key,y=d[key])
    sns.savefig('ground.png',dpi=300)

def get():
    d={'d1':[],'d2':[],'d3':[],'d4':[],'d5':[]}
    i=0
    while i<500:
        data=main((63,60),(500,-1000),(175,0))
        for j in data:
            d[j].append(data[j])
        i+=1
    dt={'distance/cm':d['d1']+d['d2']+d['d3']+d['d4']+d['d5'],'detector':['d1']*500+['d2']*500+['d3']*500+['d4']*500+['d5']*500}
    df=pd.DataFrame(dt)
    sns.boxplot(x='detector',y='distance/cm',data=df)
#    for key in d:
#        print(key)
#        sns.boxplot(x=key,y=d[key])
#    sns.savefig('d1_far.png',dpi=300)
get()
#get_ground()
#experiment()
```

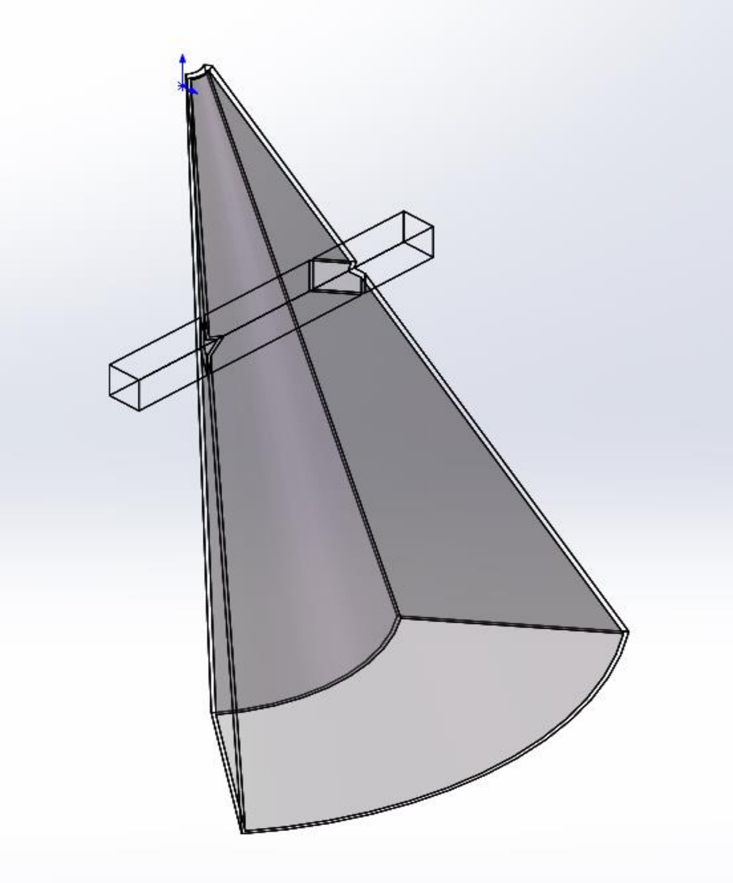
Experiments

6 Simulations – Obstacle Detecting

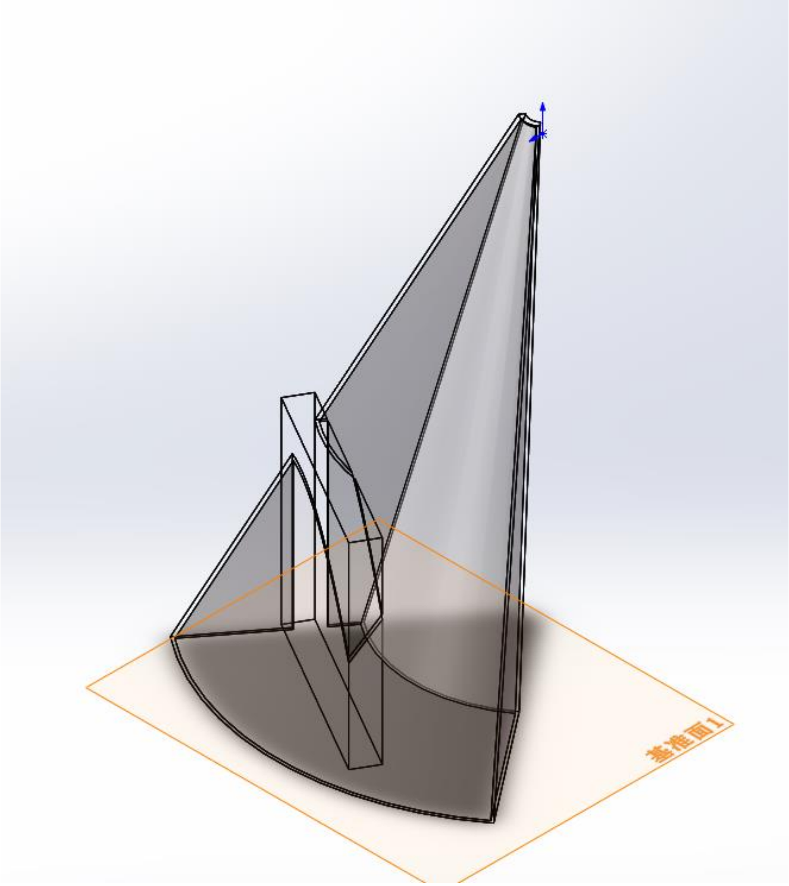
Vertical bar



Horizontal bar

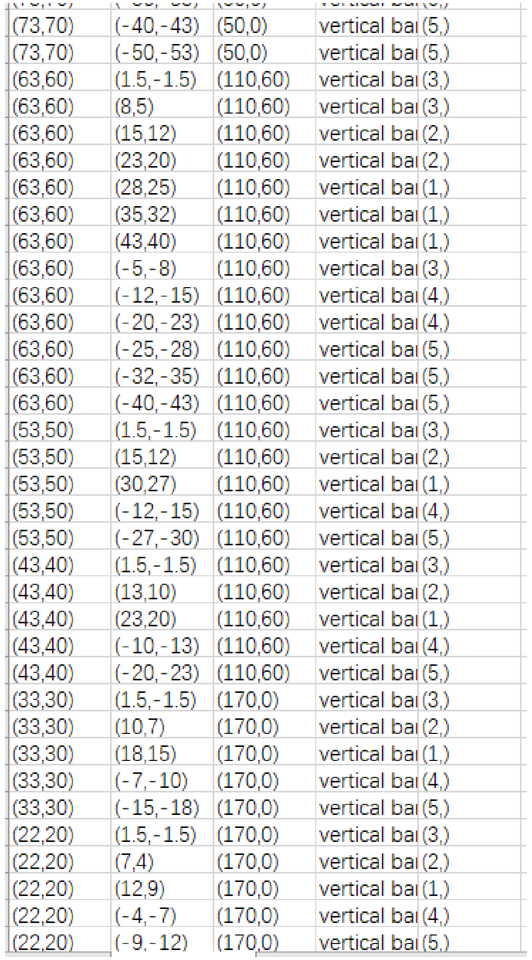


wall

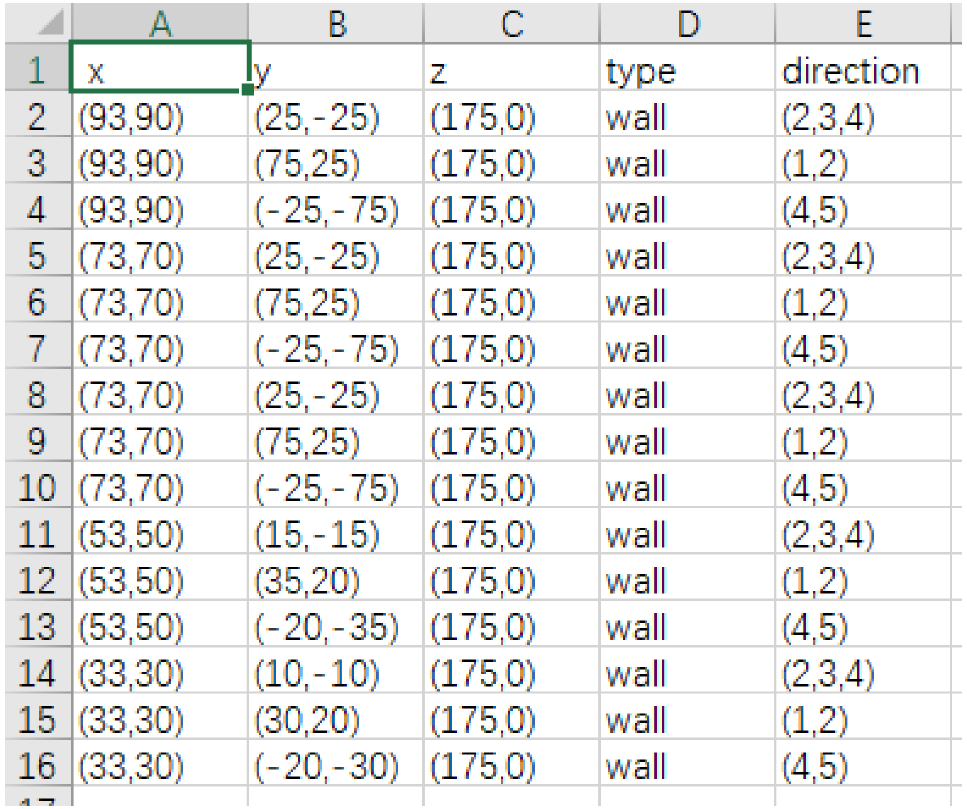


6 Simulations – Obstacle Detecting

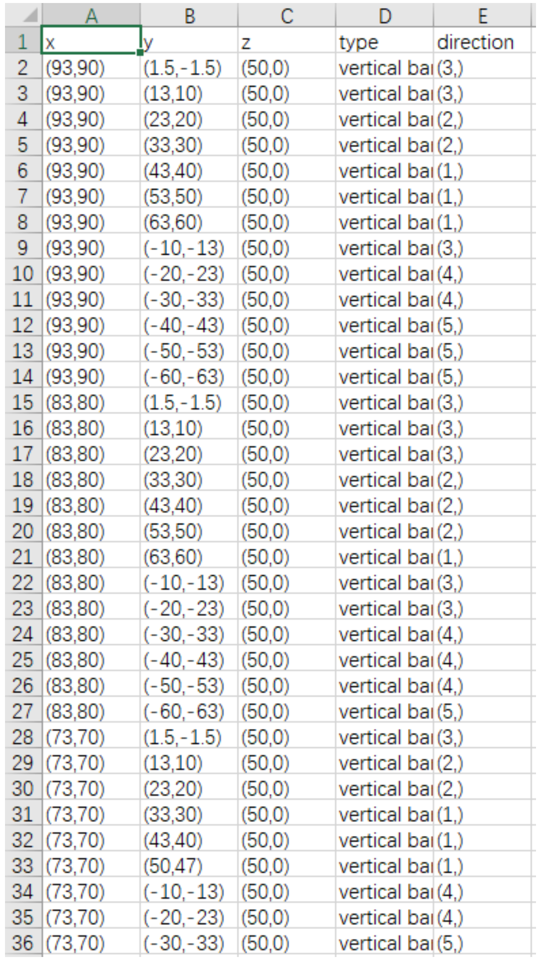
Tested data



(73,70)	(-40,-43)	(50,0)	vertical ba	(5,)
(73,70)	(-50,-53)	(50,0)	vertical ba	(5,)
(63,60)	(1.5,-1.5)	(110,60)	vertical ba	(3,)
(63,60)	(8,5)	(110,60)	vertical ba	(3,)
(63,60)	(15,12)	(110,60)	vertical ba	(2,)
(63,60)	(23,20)	(110,60)	vertical ba	(2,)
(63,60)	(28,25)	(110,60)	vertical ba	(1,)
(63,60)	(35,32)	(110,60)	vertical ba	(1,)
(63,60)	(43,40)	(110,60)	vertical ba	(1,)
(63,60)	(-5,-8)	(110,60)	vertical ba	(3,)
(63,60)	(-12,-15)	(110,60)	vertical ba	(4,)
(63,60)	(-20,-23)	(110,60)	vertical ba	(4,)
(63,60)	(-25,-28)	(110,60)	vertical ba	(5,)
(63,60)	(-32,-35)	(110,60)	vertical ba	(5,)
(63,60)	(-40,-43)	(110,60)	vertical ba	(5,)
(53,50)	(1.5,-1.5)	(110,60)	vertical ba	(3,)
(53,50)	(15,12)	(110,60)	vertical ba	(2,)
(53,50)	(30,27)	(110,60)	vertical ba	(1,)
(53,50)	(-12,-15)	(110,60)	vertical ba	(4,)
(53,50)	(-27,-30)	(110,60)	vertical ba	(5,)
(43,40)	(1.5,-1.5)	(110,60)	vertical ba	(3,)
(43,40)	(13,10)	(110,60)	vertical ba	(2,)
(43,40)	(23,20)	(110,60)	vertical ba	(1,)
(43,40)	(-10,-13)	(110,60)	vertical ba	(4,)
(43,40)	(-20,-23)	(110,60)	vertical ba	(5,)
(33,30)	(1.5,-1.5)	(170,0)	vertical ba	(3,)
(33,30)	(10,7)	(170,0)	vertical ba	(2,)
(33,30)	(18,15)	(170,0)	vertical ba	(1,)
(33,30)	(-7,-10)	(170,0)	vertical ba	(4,)
(33,30)	(-15,-18)	(170,0)	vertical ba	(5,)
(22,20)	(1.5,-1.5)	(170,0)	vertical ba	(3,)
(22,20)	(7,4)	(170,0)	vertical ba	(2,)
(22,20)	(12,9)	(170,0)	vertical ba	(1,)
(22,20)	(-4,-7)	(170,0)	vertical ba	(4,)
(22,20)	(-9,-12)	(170,0)	vertical ba	(5,)



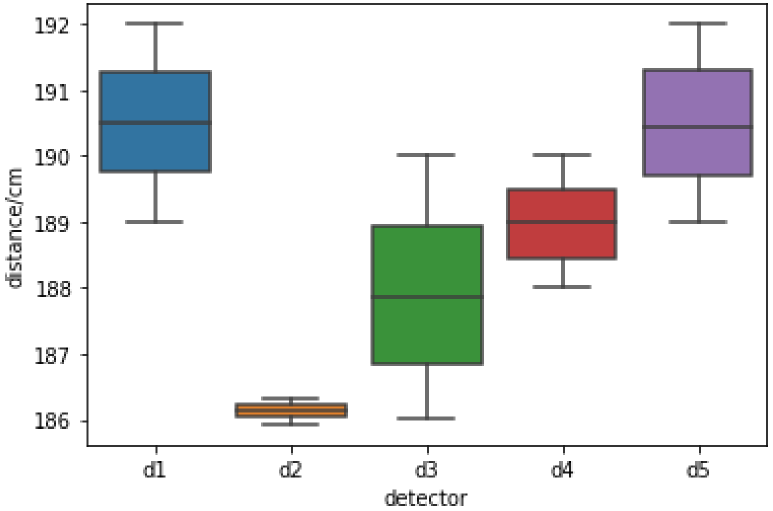
	A	B	C	D	E
1	x	y	z	type	direction
2	(93,90)	(25,-25)	(175,0)	wall	(2,3,4)
3	(93,90)	(75,25)	(175,0)	wall	(1,2)
4	(93,90)	(-25,-75)	(175,0)	wall	(4,5)
5	(73,70)	(25,-25)	(175,0)	wall	(2,3,4)
6	(73,70)	(75,25)	(175,0)	wall	(1,2)
7	(73,70)	(-25,-75)	(175,0)	wall	(4,5)
8	(73,70)	(25,-25)	(175,0)	wall	(2,3,4)
9	(73,70)	(75,25)	(175,0)	wall	(1,2)
10	(73,70)	(-25,-75)	(175,0)	wall	(4,5)
11	(53,50)	(15,-15)	(175,0)	wall	(2,3,4)
12	(53,50)	(35,20)	(175,0)	wall	(1,2)
13	(53,50)	(-20,-35)	(175,0)	wall	(4,5)
14	(33,30)	(10,-10)	(175,0)	wall	(2,3,4)
15	(33,30)	(30,20)	(175,0)	wall	(1,2)
16	(33,30)	(-20,-30)	(175,0)	wall	(4,5)



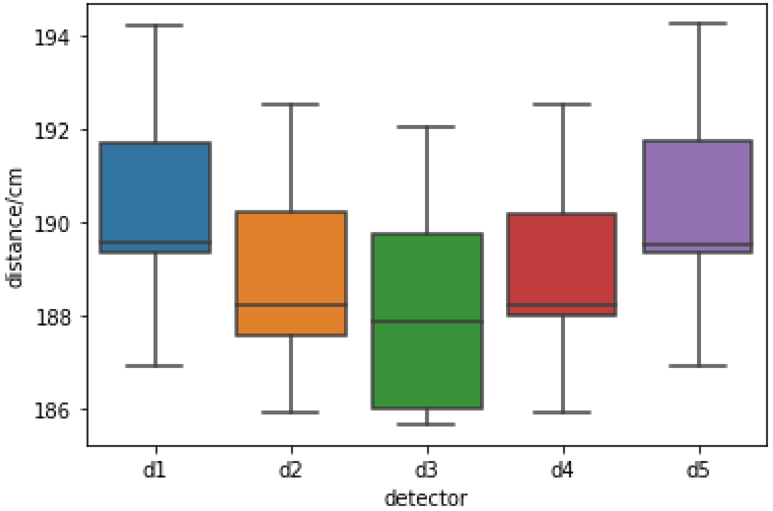
	A	B	C	D	E
1	x	y	z	type	direction
2	(93,90)	(1.5,-1.5)	(50,0)	vertical ba	(3,)
3	(93,90)	(13,10)	(50,0)	vertical ba	(3,)
4	(93,90)	(23,20)	(50,0)	vertical ba	(2,)
5	(93,90)	(33,30)	(50,0)	vertical ba	(2,)
6	(93,90)	(43,40)	(50,0)	vertical ba	(1,)
7	(93,90)	(53,50)	(50,0)	vertical ba	(1,)
8	(93,90)	(63,60)	(50,0)	vertical ba	(1,)
9	(93,90)	(-10,-13)	(50,0)	vertical ba	(3,)
10	(93,90)	(-20,-23)	(50,0)	vertical ba	(4,)
11	(93,90)	(-30,-33)	(50,0)	vertical ba	(4,)
12	(93,90)	(-40,-43)	(50,0)	vertical ba	(5,)
13	(93,90)	(-50,-53)	(50,0)	vertical ba	(5,)
14	(93,90)	(-60,-63)	(50,0)	vertical ba	(5,)
15	(83,80)	(1.5,-1.5)	(50,0)	vertical ba	(3,)
16	(83,80)	(13,10)	(50,0)	vertical ba	(3,)
17	(83,80)	(23,20)	(50,0)	vertical ba	(3,)
18	(83,80)	(33,30)	(50,0)	vertical ba	(2,)
19	(83,80)	(43,40)	(50,0)	vertical ba	(2,)
20	(83,80)	(53,50)	(50,0)	vertical ba	(2,)
21	(83,80)	(63,60)	(50,0)	vertical ba	(1,)
22	(83,80)	(-10,-13)	(50,0)	vertical ba	(3,)
23	(83,80)	(-20,-23)	(50,0)	vertical ba	(3,)
24	(83,80)	(-30,-33)	(50,0)	vertical ba	(4,)
25	(83,80)	(-40,-43)	(50,0)	vertical ba	(4,)
26	(83,80)	(-50,-53)	(50,0)	vertical ba	(4,)
27	(83,80)	(-60,-63)	(50,0)	vertical ba	(5,)
28	(73,70)	(1.5,-1.5)	(50,0)	vertical ba	(3,)
29	(73,70)	(13,10)	(50,0)	vertical ba	(2,)
30	(73,70)	(23,20)	(50,0)	vertical ba	(2,)
31	(73,70)	(33,30)	(50,0)	vertical ba	(1,)
32	(73,70)	(43,40)	(50,0)	vertical ba	(1,)
33	(73,70)	(50,47)	(50,0)	vertical ba	(1,)
34	(73,70)	(-10,-13)	(50,0)	vertical ba	(4,)
35	(73,70)	(-20,-23)	(50,0)	vertical ba	(4,)
36	(73,70)	(-30,-33)	(50,0)	vertical ba	(5,)

6 Simulations – Obstacle Detecting

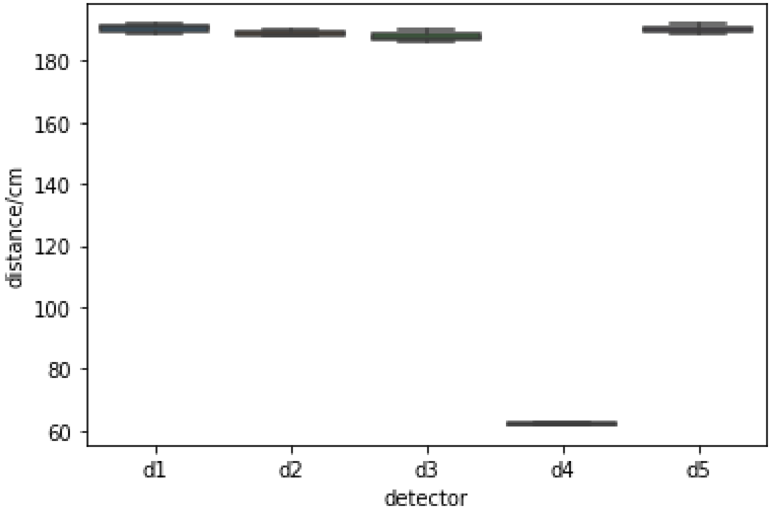
Far front left



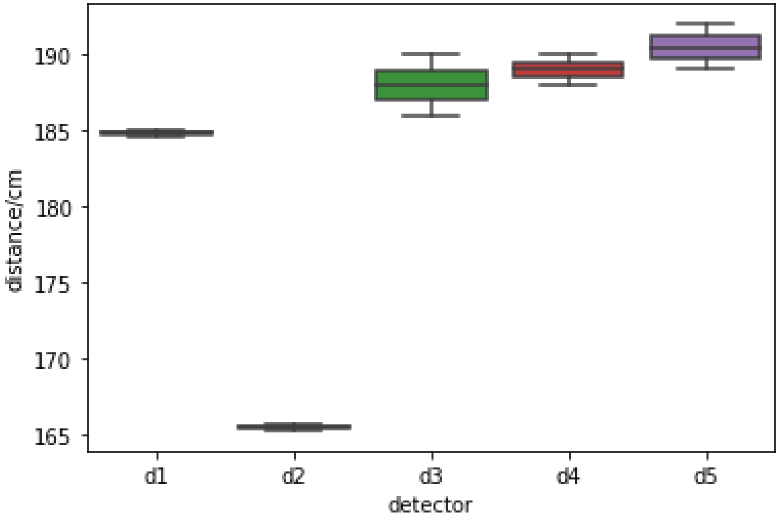
Ground



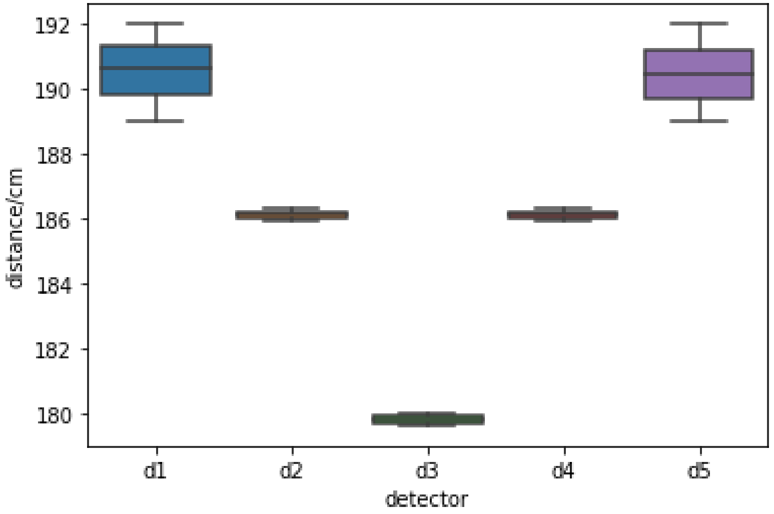
Close front right



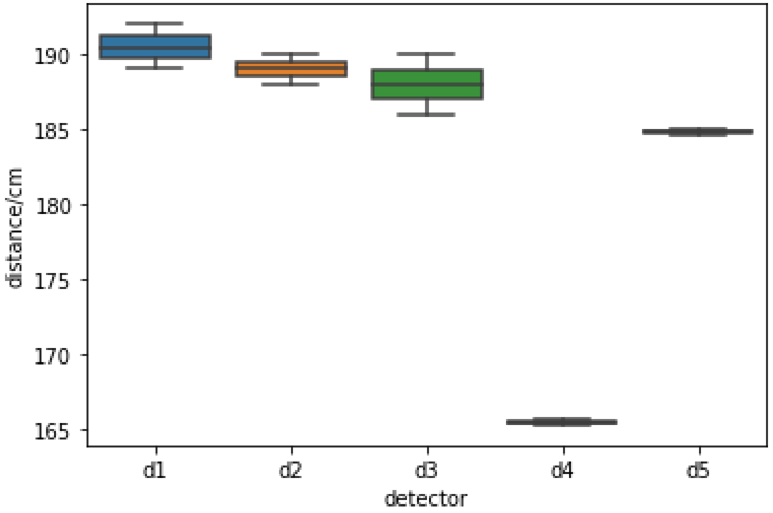
Complex left



Complex middle



Complex right



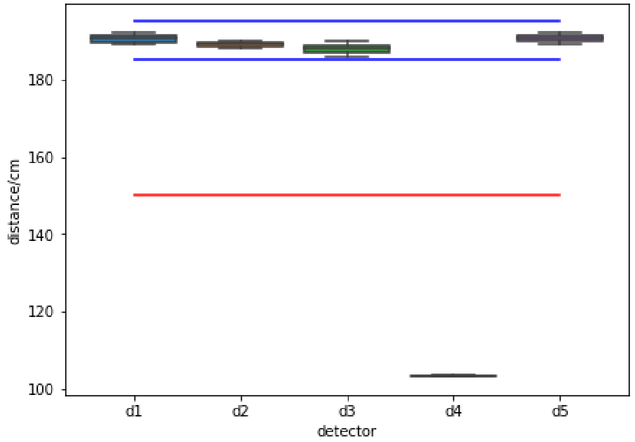
6 Simulations – Obstacle Detecting

Algorithm for identifying obstacle:

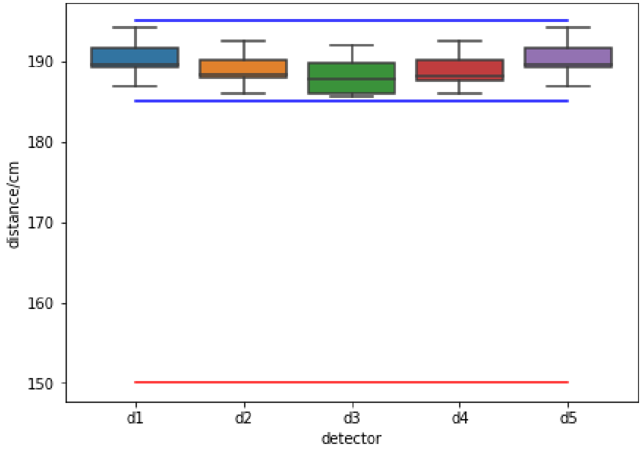
- □ Shortest distance
- □ Relation of distance in 5 directions

Ability:

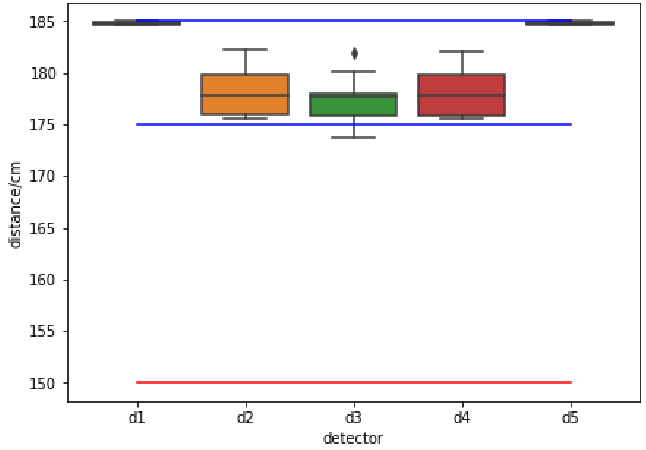
- □ **Identify different types** of obstacles that block in the way
- □ **Avoid being triggered by roads** that are slightly inclined or not so flat



obstacle

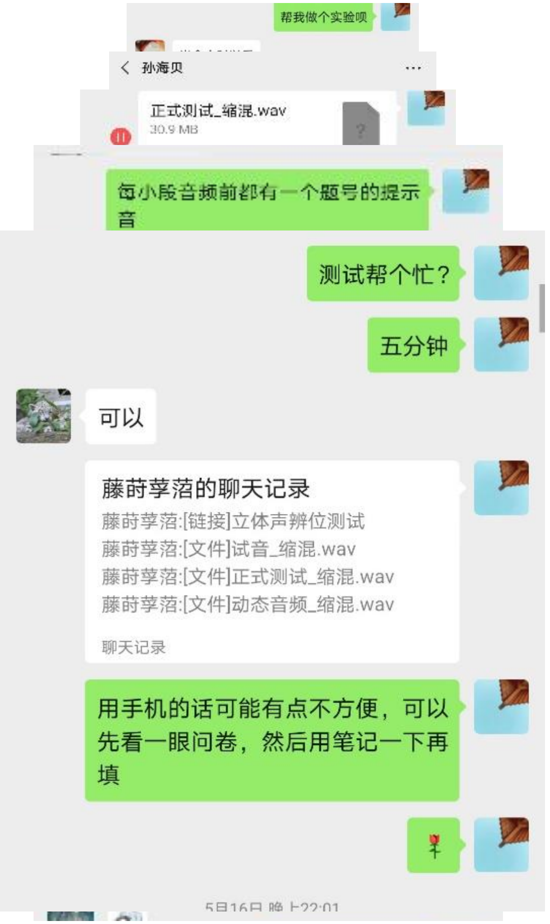


ground



Rough and inclined roads

6 Research – Stereo Sound



帮我做个实验呗

孙海贝

正式测试_缩混.wav
30.9 MB

每小段音频前都有一个题号的提示音

测试帮个忙？

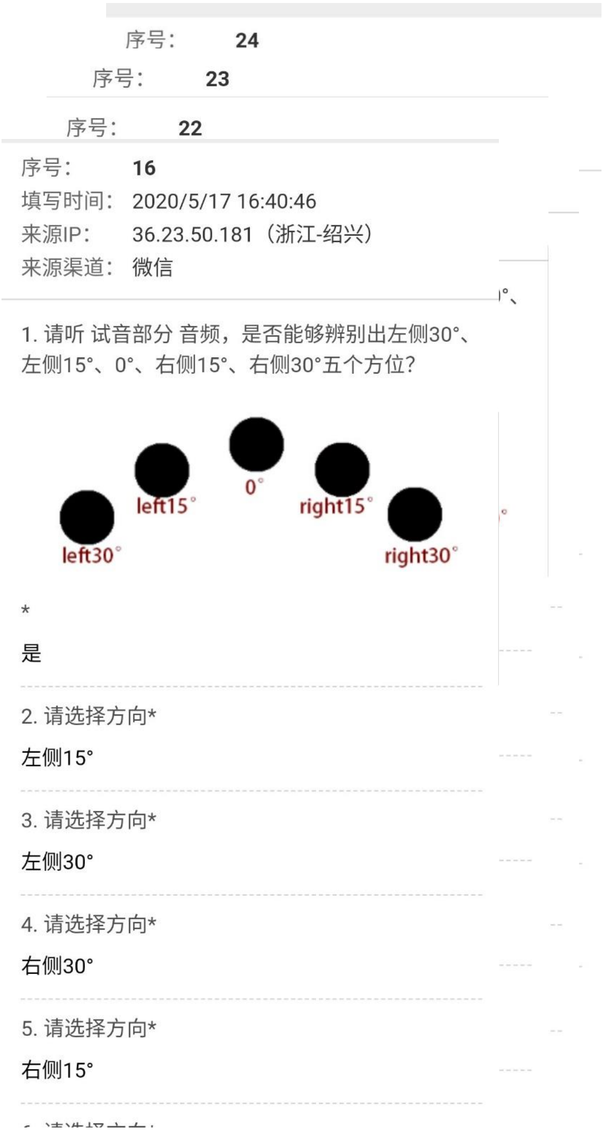
五分钟

可以

藤莳莩湉的聊天记录
藤莳莩湉:[链接]立体声辨位测试
藤莳莩湉:[文件]试音_缩混.wav
藤莳莩湉:[文件]正式测试_缩混.wav
藤莳莩湉:[文件]动态音频_缩混.wav
聊天记录

用手机的话可能有点不方便，可以先看一眼问卷，然后用笔记一下再填

5月16日 晚上22:01



序号： 24

序号： 23

序号： 22

序号： 16
填写时间： 2020/5/17 16:40:46
来源IP： 36.23.50.181（浙江-绍兴）
来源渠道： 微信

1. 请听 试音部分 音频，是否能够辨别出左侧30°、左侧15°、0°、右侧15°、右侧30°五个方位？

*
是

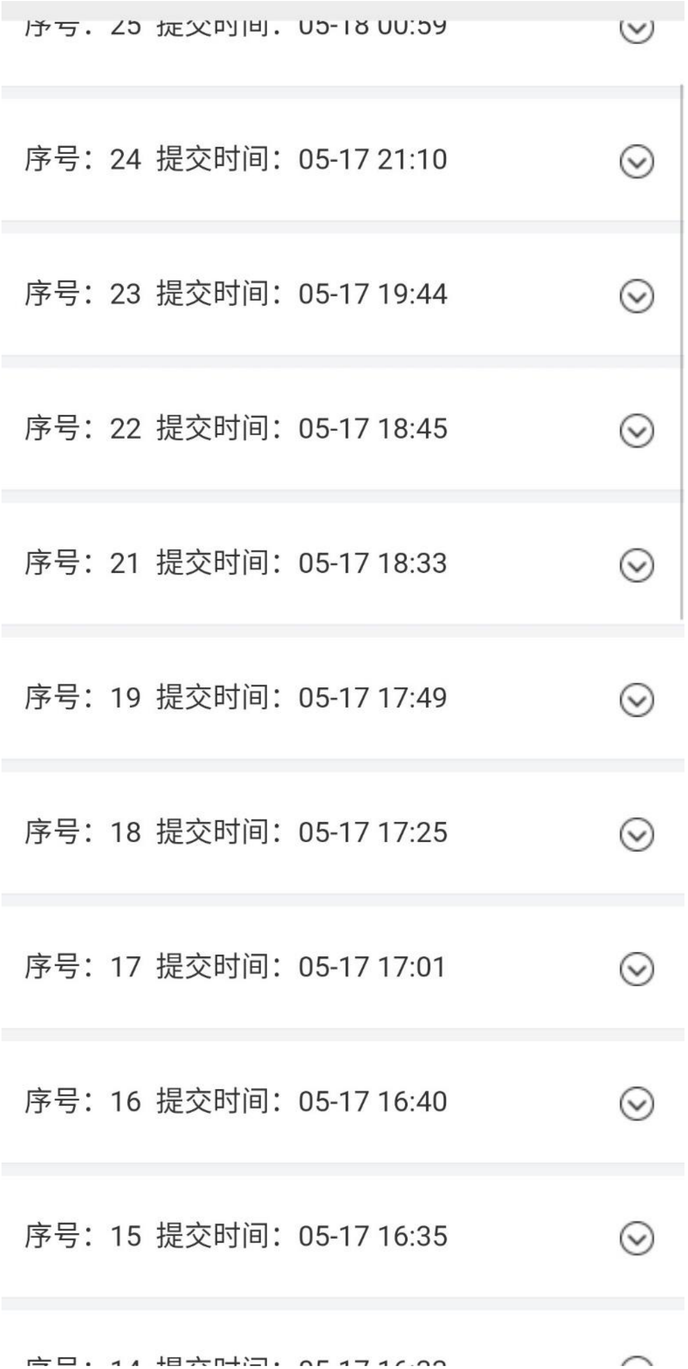
2. 请选择方向*
左侧15°

3. 请选择方向*
左侧30°

4. 请选择方向*
右侧30°

5. 请选择方向*
右侧15°

6. 请选择方向*



序号：25 提交时间：05-18 00:59

序号：24 提交时间：05-17 21:10

序号：23 提交时间：05-17 19:44

序号：22 提交时间：05-17 18:45

序号：21 提交时间：05-17 18:33

序号：19 提交时间：05-17 17:49

序号：18 提交时间：05-17 17:25

序号：17 提交时间：05-17 17:01

序号：16 提交时间：05-17 16:40

序号：15 提交时间：05-17 16:35

序号：14 提交时间：05-17 16:22

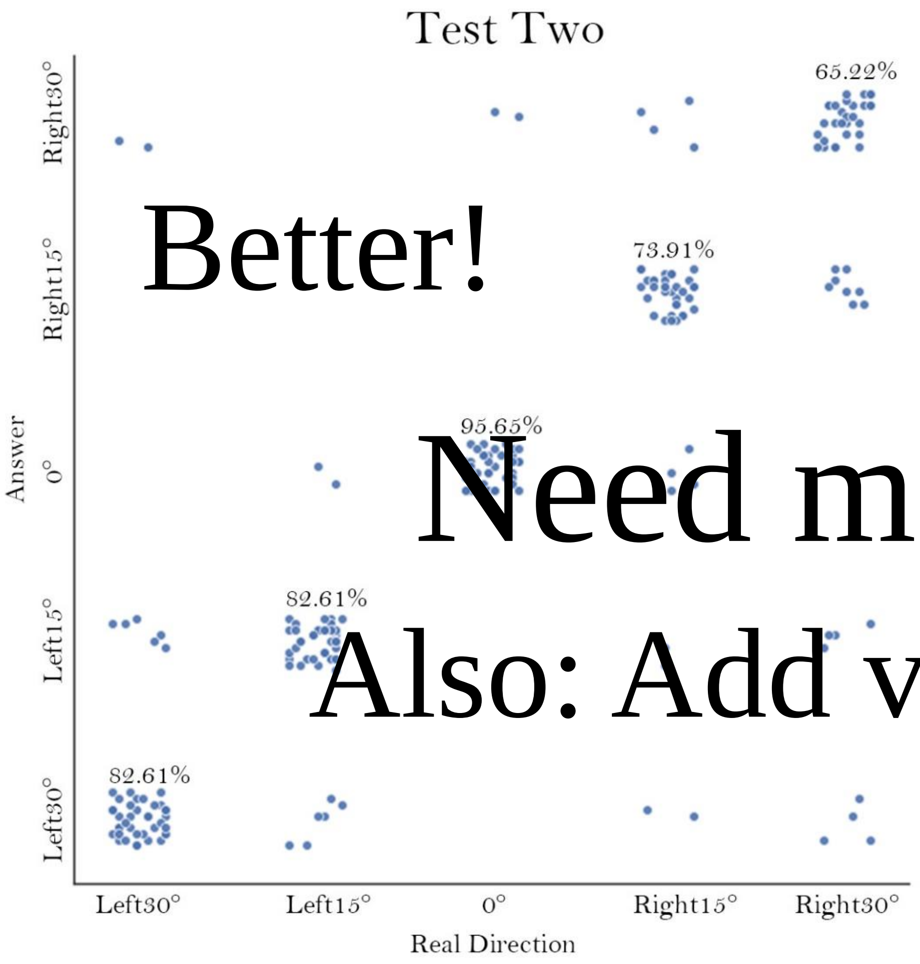
6 Research – Stereo Sound

Test One

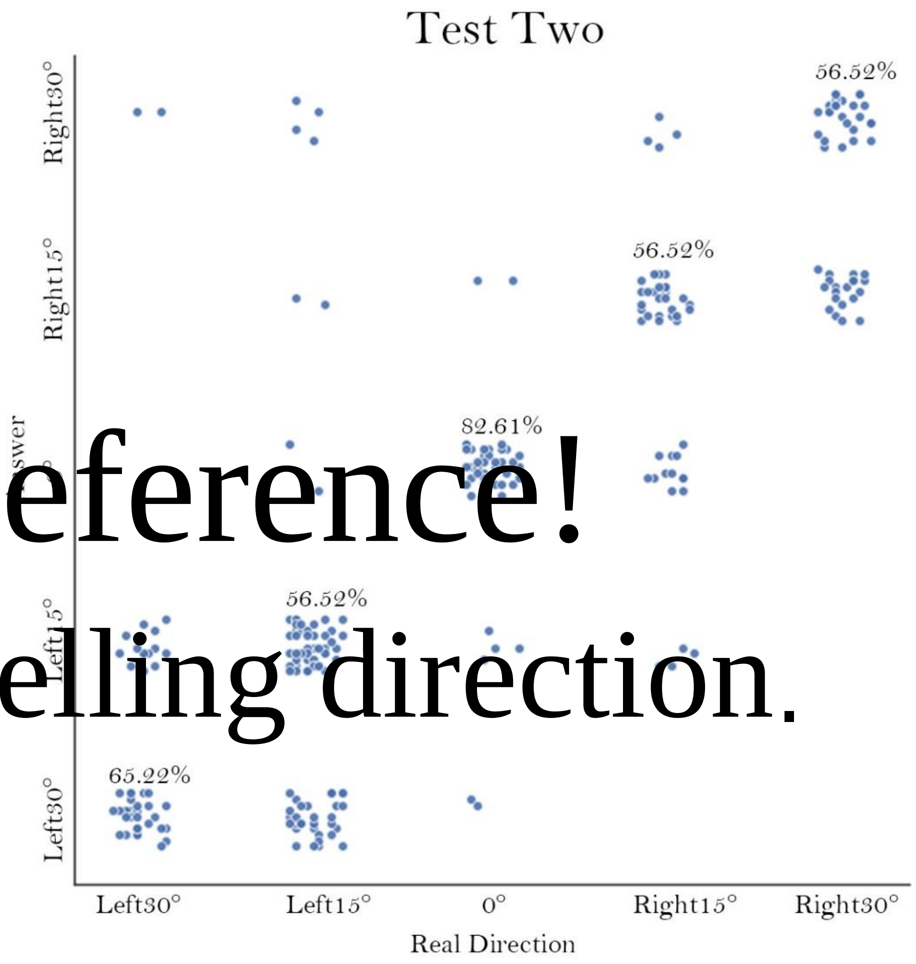
	Static Sound (5 directions)	Dynamic Sound (5 directions)
Correctness (%)	84	90

Can give intuitive sense of direction and catch attention.

6 Research – Stereo Sound



Better!



Need more reference!

Also: Add voice telling direction.

Order 1: Left15° ➡ Left30° ➡ Right15° ➡ Right30° ➡ 0°

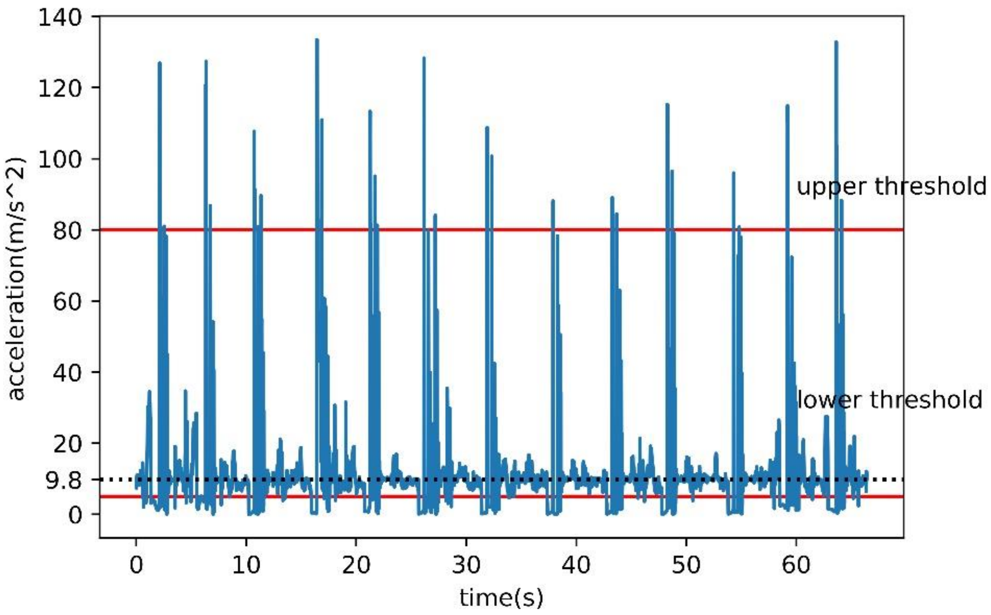
Order 2: Left30° ➡ Right30° ➡ Right15° ➡ 0° ➡ Left15°

6 Experiment – Fall Detecting

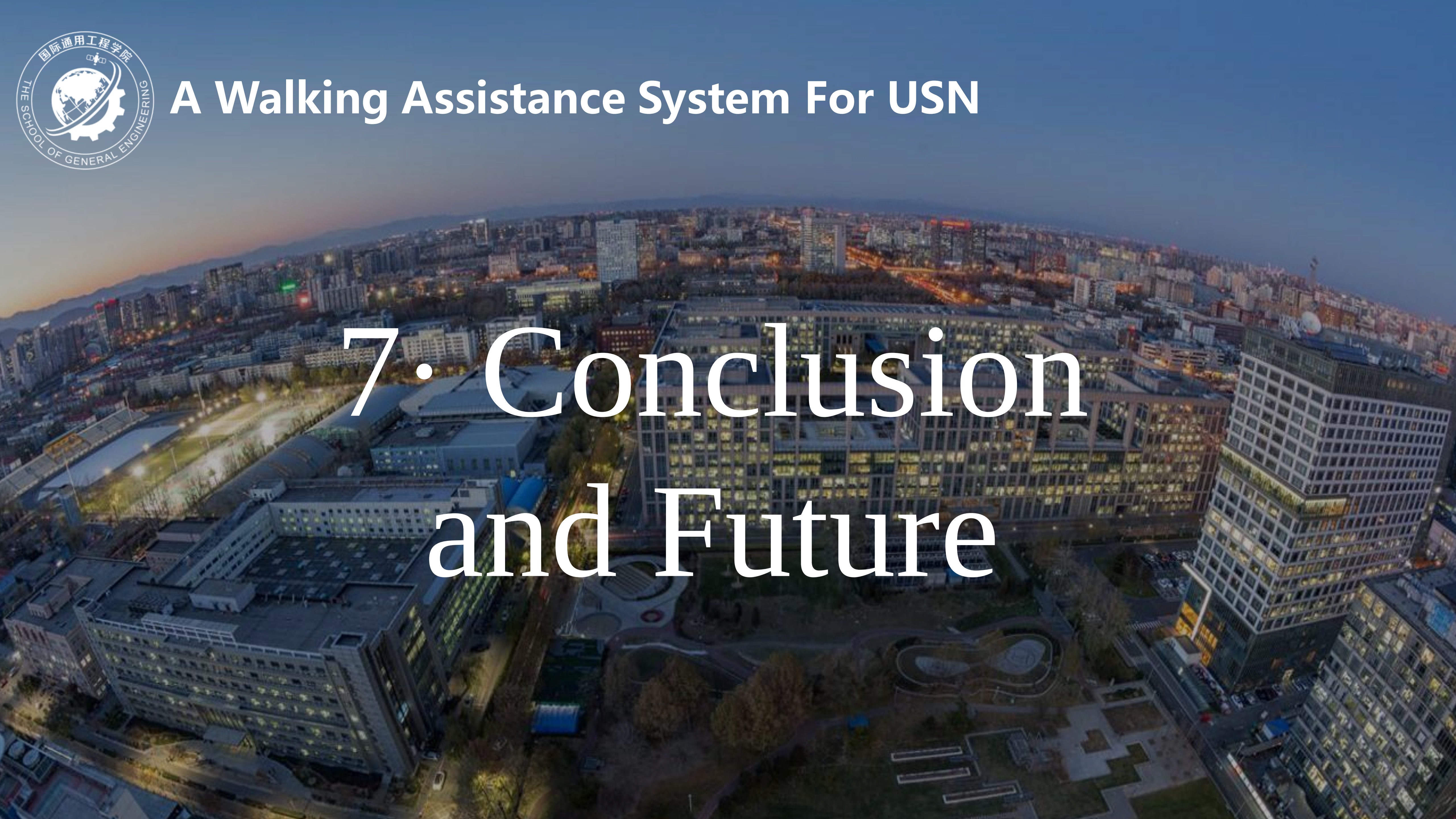
Test1: **Reality testing**. (Put cellphone inside experimenters’ the pocket, fall down on the cushion)

Test2: **Free fall test**. (the cellphone falls down from 100cm height)

Test3: **Simulated test**. (Tied to a stick and fall down on the cushion. Height: 94cm, Elasticity modulus:100kPa)



Test	Reality Test	Free Fall Test	Simulated Test
Successful/total	43/50	47/50	42/50
Success rate	86%	94%	84%

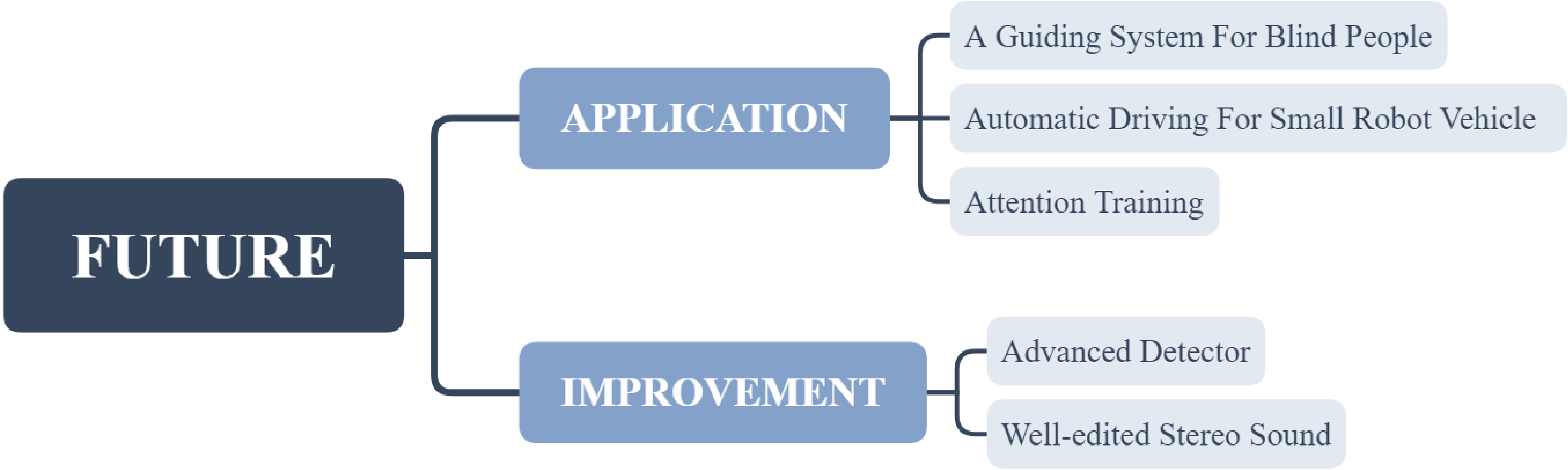


7· Conclusion and Future

7 Conclusion

Attribute	Parameters
Detection Range	100cm
Obstacle Avoidance Warning Time	35s
Success Rate of Fall Detection	>86%
Total Cost	¥327
Endurance	>30h
Weight	<400g
Volume	350x350x120(mm)

7 Future



FUTURE

APPLICATION

A Guiding System For Blind People

Automatic Driving For Small Robot Vehicle

Attention Training

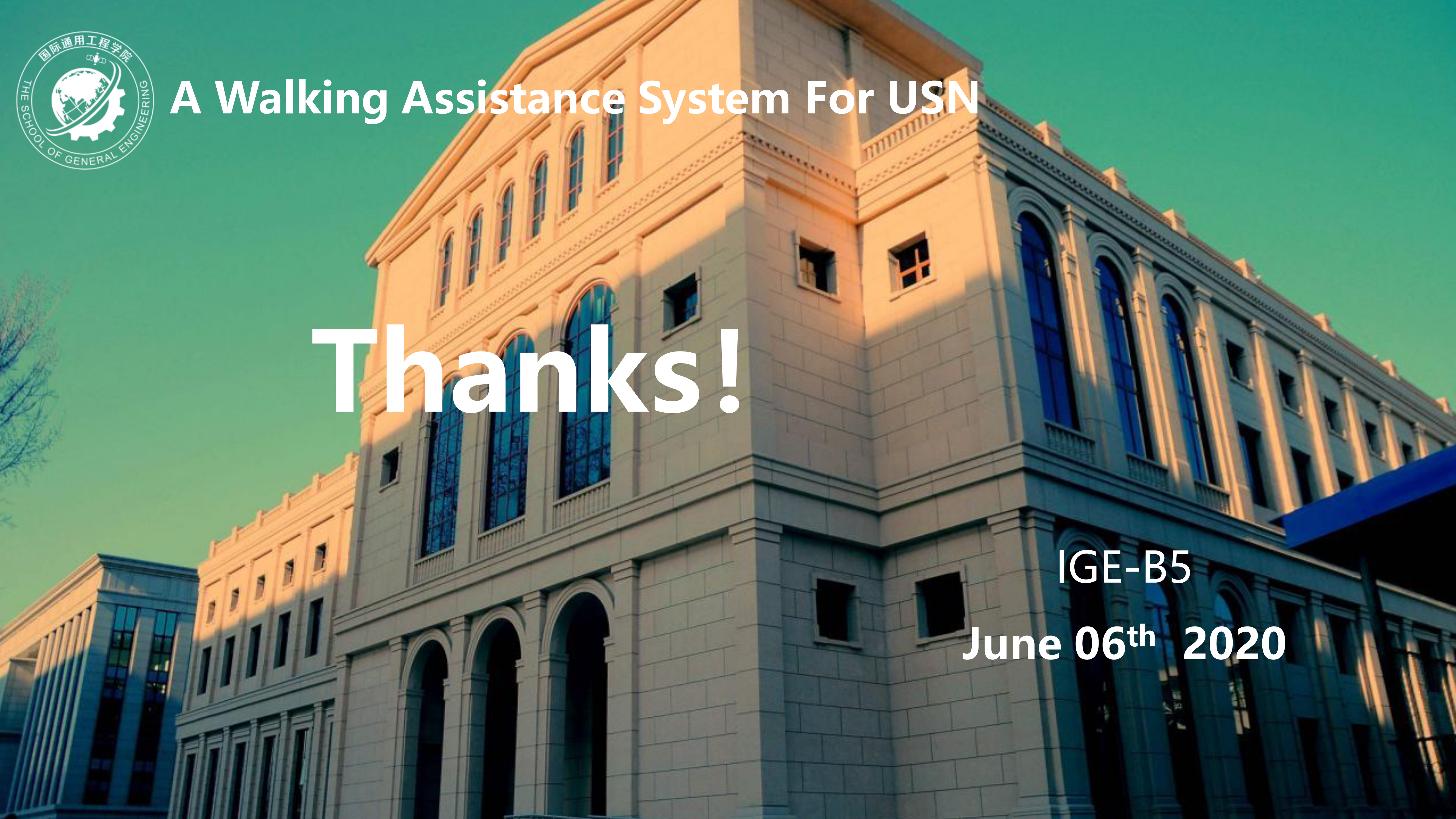
IMPROVEMENT

Advanced Detector

Well-edited Stereo Sound

国际通用工程学院 THE SCHOOL OF GENERAL ENGINEERING

A Walking Assistance System For USN



Thanks!

IGE-B5

June 06th 2020