



Enjoy Jogging

Pedometer based on Android Smart Phone



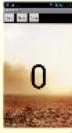
Thanks!
Any Questions?

Algorithm



Demonstration

- User Interface
- Start
 - Stop
 - Reset



Show Time!

Conclusion

Improvement Limitation

- Accuracy
- Function

Future Works

- Time threshold
- Dynamic threshold



Enjoy Jogging

Pedometer based on Android Smart Phone

Thanks!
Any Questions?



Smartphone Team#2

YAO	ZHANG	----Leader
CHUNGUANG	LI	----Tester
BITIAN	JIANG	----Searcher
TIANCHI	SHEN	----Designer



Introduction

What's pedometer?

- Counts each step a person takes by detecting the motion of the person's hands or legs.
- Motivation tool for people wanting to increase their physical activity.

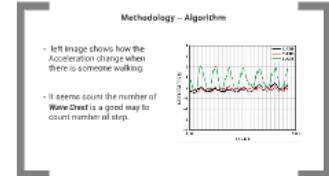
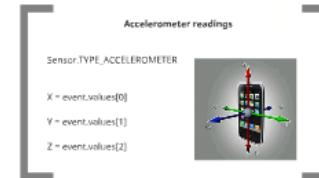


Goal:

- Use accelerometer to record the acceleration caused by walking.
- Analyze the accelerometer readings and count the number of steps (i.e., the accelerometer readings cross a threshold).
- Build a simple UI to show the count of steps.

Algorithm

Accelerometer readings



Methodology -- Algorithm

Code

Code

```
public void onSensorChanged(SensorEvent event) {  
    if (event.sensor.getType() == Sensor.TYPE_ACCELEROMETER) {  
        float x = event.values[0];  
        float y = event.values[1];  
        float z = event.values[2];  
        //...  
    }  
}
```

Annotations:

- Sensors available through Java API (Mac OS X, Linux, Windows)
- Global events
- Local events
- Sensors available documented
- Different sensors send different data
- Minimum resolution depends from hardware
- If no add environmental info

Team#2

---Leader
---Tester
---Searcher
---Designer

Accelerometer readings

Sensor.TYPE_ACCELEROMETER

X = event.values[0]

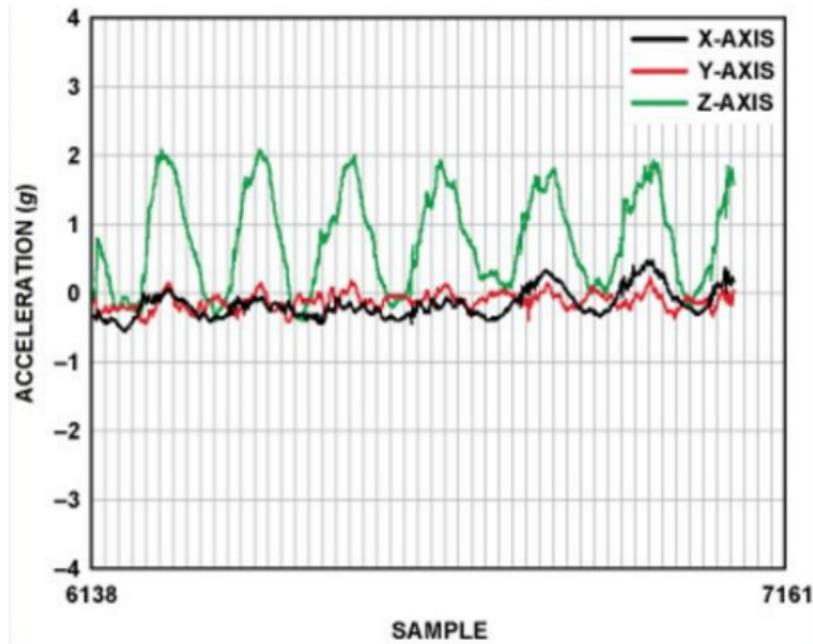
Y = event.values[1]

Z = event.values[2]



Methodology -- Algorithm

- left image shows how the Acceleration change when there is someone walking.
- It seems count the number of **Wave Crest** is a good way to count number of step.



Code

```
float[] gravity = new float[3];
float[] linear_acceleration = new float[3];

final float alpha = (float) 0.8;

gravity[0] = alpha * gravity[0] + (1 - alpha) * event.values[0];
gravity[1] = alpha * gravity[1] + (1 - alpha) * event.values[1];
gravity[2] = alpha * gravity[2] + (1 - alpha) * event.values[2];

linear_acceleration[0] = event.values[0] - gravity[0];
linear_acceleration[1] = event.values[1] - gravity[1];
linear_acceleration[2] = event.values[2] - gravity[2];

float vSum = (float) Math.sqrt((double) (linear_acceleration[0] * linear_acceleration[0] +
    linear_acceleration[1] * linear_acceleration[1] +
    linear_acceleration[2] * linear_acceleration[2])) / 3.0f;
float direction = (lastValue-vSum>0.5? 1:(lastValue-vSum<-0.5?-1:lastdirection));
if(direction == -1*lastdirection){
    int type = (direction>0?0:1);
    LastExtremes[type] = lastValue;
    float diff = LastExtremes[type] - LastExtremes[1-type];
    if(diff>Min_SENSITIVITY && diff<Max_SENSITIVITY) {
        System.out.println("The vSum is: "+vSum+"The difference is:" +diff );
        textView.setText(String.valueOf(++count));
    }
}
lastdirection = direction;
lastValue = vSum;
```

Eliminate contribution of gravity by high pass filter, get readings from x, y, z-axis.

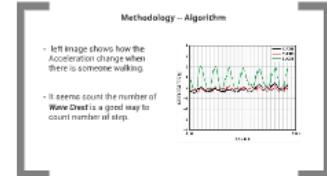
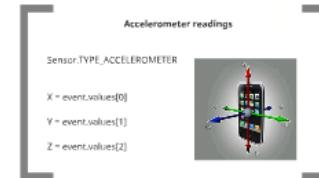
Standard metric = $\frac{\sqrt{x^2+y^2+z^2}}{3}$

Acceleration:
Keep increasing OR Keep decreasing?

Difference between wave crest and wave trough:
Greater than minimum threshold?
Smaller than maximum threshold?
If so, add one to number of step.

Algorithm

Accelerometer readings



Code

Code

```
if (event.sensor.getType() == Sensor.TYPE_ACCELEROMETER) {  
    float x = event.values[0];  
    float y = event.values[1];  
    float z = event.values[2];  
  
    if (Math.abs(x - lastX) >= 0.1 ||  
        Math.abs(y - lastY) >= 0.1 ||  
        Math.abs(z - lastZ) >= 0.1) {  
        lastX = x;  
        lastY = y;  
        lastZ = z;  
  
        // do something  
    }  
}
```

The diagram shows a snippet of Java code within a code editor interface. The code is a fragment of an event listener for the accelerometer sensor. It checks if the sensor type is an accelerometer and then compares the current values (x, y, z) with the previous values (lastX, lastY, lastZ). If any value has changed by at least 0.1, it updates the last values and performs some action (indicated by a placeholder "do something"). A text box to the right of the code provides context: "Electronics vibration signal through piezoelectric material (PZT)" and "Gyro sensor".

Team#2

---Leader
---Tester
---Searcher
---Designer

Demonstration

User Interface

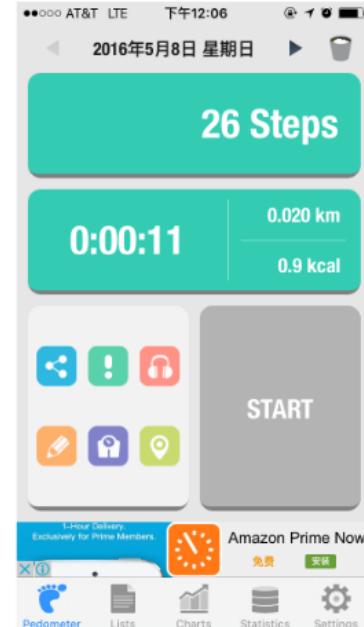
- Start
- Stop
- Reset



Show Time!

Compare with other pedometer

(Download from apple store)



Conclusion

Improvement Limitation

- Accuracy
- Function



Future Works

- Time threshold
- Dynamic threshold



Team Contr

Leader	Yao	Zhang	Hold to make the
Tester	Chungung	U	Test in
Designer	Tianchi	Shan	Coordinate
Searcher	Bilan	Jiang	Search related

Team Contribution

Leader	Yao	Zhang	Hold team meetings, make the Pedometer algorithm
Tester	Chunguang	Li	Test and debug the software program
Designer	Tianchi	Shen	Coordinate with leader, design the program
Searcher	Bitian	Jiang	Search and gather information related to the project, User Interface

Enjoy Jogging

Pedometer based on Android Smart Phone

Thanks!
Any Questions?

