

平成25年度 情報教育コース 卒業研究 最終審査

タブレットを使った センサー教材の開発

所属: 山下研究室

指導教員: 山下 茂教員

発表者: 津田一成

目次

- ▶ 研究動機
- ▶ 研究概要
- ▶ 最終審査までの取り組み
- ▶ センサーを使う手順
- ▶ センサーを使ったコンパス
- ▶ センサーの値の視覚化
- ▶ 描画サンプル
- ▶ 課題
- ▶ センサーの値を描画に反映
- ▶ 開発環境
- ▶ 参考文献・サイト

研究概要

タブレットを使用して
理科の授業の中で使えそうな教材や道具の考案



タブレットの中のセンサー機能に注目
(加速度センサー、照度センサー、方位センサーなど)

タブレットで測定機などの理科実験の「道具」を開発

最終審査までの取り組み

<7・8月>

- ・センサーについての学習
- ・加速度センサー・方位センサー・磁気センサーの取得を学習


<9・10月>

- ・センサーの値の取り出し・処理を学習
- ・グラフィック描画の学習

<11月・12月>

- ・センサーの値を描画・画像へ反映

センサーを使ったコンパス

 compass01

コンパスの赤い針が北を
指すように動く



実行画面コード

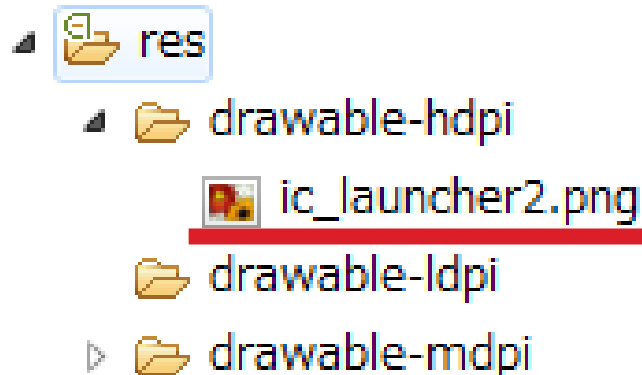
加速度センサーと
磁気センサーで方
位角を取得

```
sensorManager=(SensorManager) getSystemService(  
    Context.SENSOR_SERVICE);  
//センサー取得◎  
//加速度センサー  
sensor1=sensorManager.getDefaultSensor(  
    Sensor.TYPE_ACCELEROMETER);  
//磁気センサー  
sensor2=sensorManager.getDefaultSensor(  
    Sensor.TYPE_MAGNETIC_FIELD);  
  
//イベントリスナーの登録◎  
sensorManager.registerListener(sensorlistener, sensor1, SensorManager.SENSOR_DELAY_NORMAL);  
sensorManager.registerListener(sensorlistener, sensor2, SensorManager.SENSOR_DELAY_NORMAL);  
}  
protected void onPause()  
{  
    super.onPause();  
}
```

→中間審査時は方位センサーを使っていたが、加速度センサーと磁気センサーを使ったものに書き換えた。

実行画面コード

```
//ビットマップデータ作成  
bmp=BitmapFactory.decodeResource(  
    getResources(),R.drawable.ic_launcher2);  
varImageView=new ImageView(this);  
varImageView.setImageBitmap(bmp);  
varLayout.addView(varImageView);  
sensorlistener=new SensorListener();
```



```
res  
├── drawable-hdpi  
│   └── ic_launcher2.png  
├── drawable-ldpi  
└── drawable-mdpi
```



実行画面コード

```
//端末の向きを取得Ⓞ  
SensorManager.getOrientation(R2,v);  
//取得した方位角を度単位に変換Ⓞ  
float degrees=(float)Math.toDegrees(v[0]);  
//Matrixクラスのインスタンスを生成Ⓞ  
Matrix m=new Matrix();  
//北向きに回転Ⓞ  
m.postRotate(-degrees);  
//頂点が北に向いたイメージを作成Ⓞ  
Bitmap nbmp=Bitmap.createBitmap bmp,0,0,bmp.getWidth(),bmp.getHeight(),m,true);  
//イメージをImageViewにセットⓄ  
varImageView.setImageBitmap(nbmp);
```



作成したビットマップを
ここにセット

加速度センサーの値を数値化



数値を整数化

実行画面コード

センサーを取得する
プログラム

```
//センサーマネージャーのインスタンスの取得
mSensorManager=(SensorManager) getSystemService(Context.SENSOR_SERVICE);
//センサーの取得
List<Sensor> list=mSensorManager.getSensorList(Sensor.TYPE_ACCELEROMETER);
if(list.size()>0){
    mAccelerometer=list.get(0);
}
```

加速度センサー
を指定

実行画面コード

センサーの値を
数値で表示

値を四捨五入

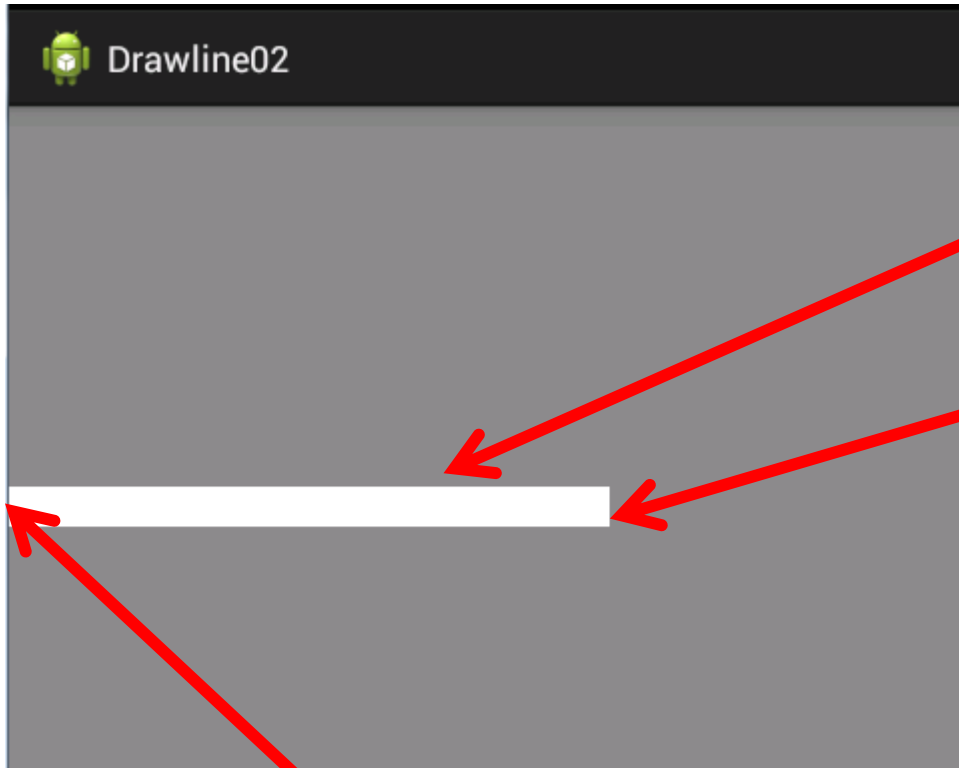
```
class TickHandler extends Handler {
    ...
    public void handleMessage(Message msg) {
        StringBuffer bufStr=new StringBuffer();

        bufStr.append("変化後のx軸の値" +Math.round(nValues[0])+"\n");
        bufStr.append("変化後のy軸の値" +Math.round(nValues[1])+"\n");
        bufStr.append("変化後のz軸の値" +Math.round(nValues[2])+"\n");
        bufStr.append("=====\n");
        bufStr.append("変化前のx軸の値" +Math.round(oValues[0])+"\n");
        bufStr.append("変化前のy軸の値" +Math.round(oValues[1])+"\n");
        bufStr.append("変化前のz軸の値" +Math.round(oValues[2])+"\n");

        txtView.setText(bufStr.toString());
        if(tickHandler !=null)tickHandler.sleep(1000);
    }
}
```

→ 値を整数化することで描画しやすく、対象である小学生にも分かりやすく表記した。

描画サンプル



線の描画

終点の座標(300,200)

始点の座標(0,200)

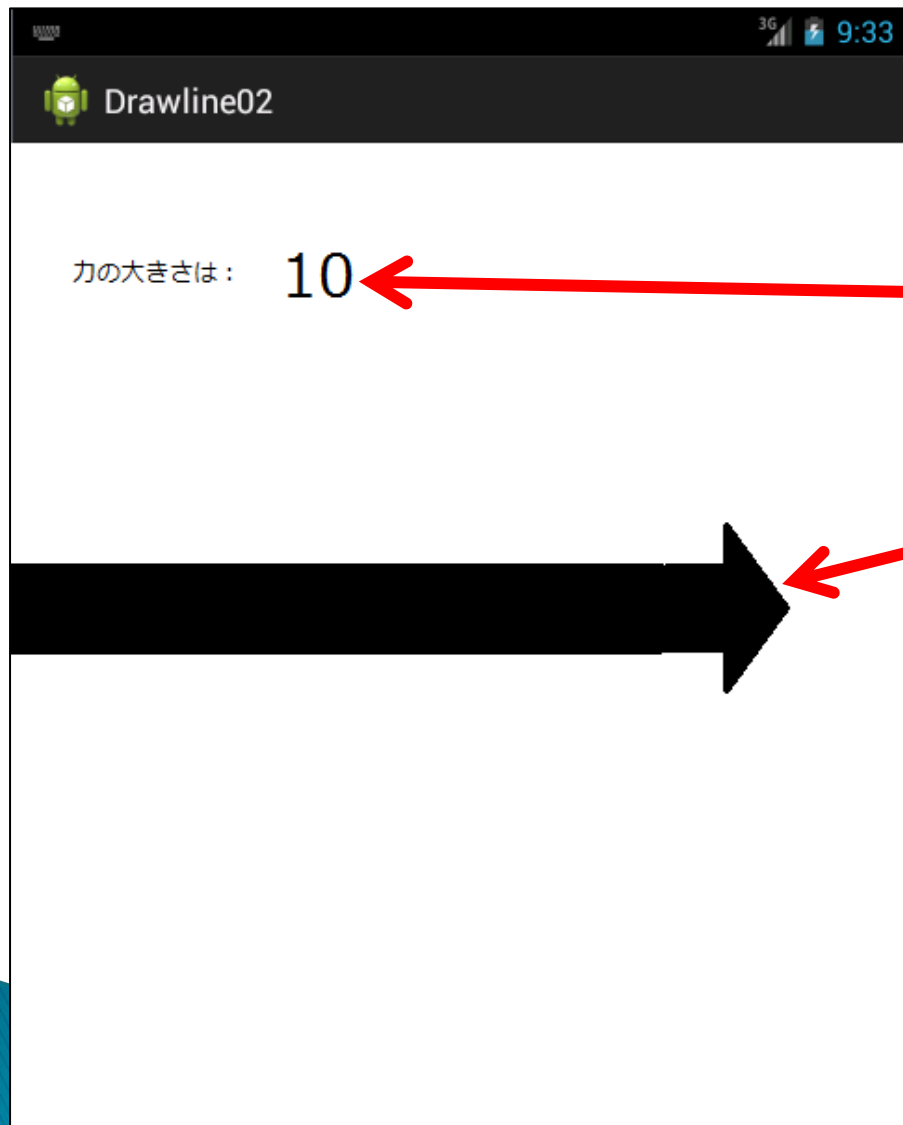
実行画面コード

直線の描画を実際
に行う

直線の太さ

```
protected void onDraw(Canvas canvas) {  
    Paint paint = new Paint();  
    paint.setColor(Color.WHITE);  
    paint.setStrokeWidth(20);  
  
    canvas.drawLine(0, 200, 300, 200, paint);  
}
```

センサーの値を描画に反映



測定した値を数値で表示

矢印で力の大きさを表示

苦勞している点・今後の課題

§ センサーで値を取得できるもののそれをうまく描画に反映できない。

原因: 値の引継がうまくできていない。

§ 加速度センサーの値(x軸、y軸、z軸)全ての値を使わず、軸1つ1つの値を使うようにプログラムを組む。

開発環境

OS	Windows7
開発言語	Java
開発ツール	<ul style="list-style-type: none">・Java Development Kit(JDK)・Android Software Developer's Kit(SDK)・Eclipse

参考文献・サイト

<文献>

- ▶ 作ればわかるわかる！Androidプログラミング
- ▶ Androidプログラミング～パーフェクトマスター～

<サイト>

- ▶ <http://techbooster.org/android/device/12524/> (TechBooster)
- ▶ <http://dixq.net/Android/> (Androidの館)
- ▶ <http://www.adakoda.com/android/>
(逆引きAndroid入門)