

Using Sensors in Android

Location, State and Environment Sensors

A.R. Kazemi

Android Sensor Categories

- Motion Sensors
- Position Sensors
- Environment Sensors

Supported Sensor Types

Sensor	Type	Description	Common Uses
<code>TYPE_ACCELEROMETER</code>	Hardware	Measures the acceleration force in m/s^2 that is applied to a device on all three physical axes (x, y, and z), including the force of gravity.	Motion detection (shake, tilt, etc.).
<code>TYPE_AMBIENT_TEMPERATURE</code>	Hardware	Measures the ambient room temperature in degrees Celsius ($^{\circ}\text{C}$). See note below.	Monitoring air temperatures.
<code>TYPE_GRAVITY</code>	Software or Hardware	Measures the force of gravity in m/s^2 that is applied to a device on all three physical axes (x, y, z).	Motion detection (shake, tilt, etc.).
<code>TYPE_GYROSCOPE</code>	Hardware	Measures a device's rate of rotation in rad/s around each of the three physical axes (x, y, and z).	Rotation detection (spin, turn, etc.).
<code>TYPE_LIGHT</code>	Hardware	Measures the ambient light level (illumination) in lx.	Controlling screen brightness.
<code>TYPE_LINEAR_ACCELERATION</code>	Software or Hardware	Measures the acceleration force in m/s^2 that is applied to a device on all three physical axes (x, y, and z), excluding the force of gravity.	Monitoring acceleration along a single axis.
<code>TYPE_MAGNETIC_FIELD</code>	Hardware	Measures the ambient geomagnetic field for all three physical axes (x, y, z) in μT .	Creating a compass.

Supported Sensor Types

<code>TYPE_ORIENTATION</code>	Software	Measures degrees of rotation that a device makes around all three physical axes (x, y, z). As of API level 3 you can obtain the inclination matrix and rotation matrix for a device by using the gravity sensor and the geomagnetic field sensor in conjunction with the <code>getRotationMatrix()</code> method.	Determining device position.
<code>TYPE_PRESSURE</code>	Hardware	Measures the ambient air pressure in hPa or mbar.	Monitoring air pressure changes.
<code>TYPE_PROXIMITY</code>	Hardware	Measures the proximity of an object in cm relative to the view screen of a device. This sensor is typically used to determine whether a handset is being held up to a person's ear.	Phone position during a call.
<code>TYPE_RELATIVE_HUMIDITY</code>	Hardware	Measures the relative ambient humidity in percent (%).	Monitoring dewpoint, absolute, and relative humidity.
<code>TYPE_ROTATION_VECTOR</code>	Software or Hardware	Measures the orientation of a device by providing the three elements of the device's rotation vector.	Motion detection and rotation detection.
<code>TYPE_TEMPERATURE</code>	Hardware	Measures the temperature of the device in degrees Celsius (°C). This sensor implementation varies across devices and this sensor was replaced with the <code>TYPE_AMBIENT_TEMPERATURE</code> sensor in API Level 14	Monitoring temperatures.

Android Sensor API (Classes)

- In the `android.hardware.*` there are 4 classes (interfaces) which provide API for working with sensors:
 - `SensorManager`
 - `Sensor`
 - `SensorEvent`
 - `SensorEventListener`

Sensors Available in Different Versions

Sensor	Android 4.0 (API Level 14)	Android 2.3 (API Level 9)	Android 2.2 (API Level 8)	Android 1.5 (API Level 3)
TYPE_ACCELEROMETER	Yes	Yes	Yes	Yes
TYPE_AMBIENT_TEMPERATURE	Yes	n/a	n/a	n/a
TYPE_GRAVITY	Yes	Yes	n/a	n/a
TYPE_GYROSCOPE	Yes	Yes	n/a ¹	n/a ¹
TYPE_LIGHT	Yes	Yes	Yes	Yes
TYPE_LINEAR_ACCELERATION	Yes	Yes	n/a	n/a
TYPE_MAGNETIC_FIELD	Yes	Yes	Yes	Yes
TYPE_ORIENTATION	Yes ²	Yes ²	Yes ²	Yes
TYPE_PRESSURE	Yes	Yes	n/a ¹	n/a ¹
TYPE_PROXIMITY	Yes	Yes	Yes	Yes
TYPE_RELATIVE_HUMIDITY	Yes	n/a	n/a	n/a
TYPE_ROTATION_VECTOR	Yes	Yes	n/a	n/a
TYPE_TEMPERATURE	Yes ²	Yes	Yes	Yes

Identifying and Accessing to Sensors

```
private SensorManager mSensorManager;  
...  
mSensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
```

```
List<Sensor> deviceSensors = mSensorManager.getSensorList(Sensor.TYPE_ALL);
```

```
private SensorManager mSensorManager;  
...  
mSensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);  
if (mSensorManager.getDefaultSensor(Sensor.TYPE_MAGNETIC_FIELD) != null) {  
    // Success! There's a magnetometer.  
}  
else {  
    // Failure! No magnetometer.  
}
```

Sensors Events

- Sensor Accuracy Changes
- Sensor Reports an new measured value

Monitoring Sensors Events

```
public class SensorActivity extends Activity implements SensorEventListener {
    private SensorManager mSensorManager;
    private Sensor mLight;
    public final void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        mSensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
        mLight = mSensorManager.getDefaultSensor(Sensor.TYPE_LIGHT);
    }
    public final void onAccuracyChanged(Sensor sensor, int accuracy) {
        // Do something here if sensor accuracy changes.
    }
    public final void onSensorChanged(SensorEvent event) {
        // The light sensor returns a single value.
        // Many sensors return 3 values, one for each axis.
        float lux = event.values[0];
        // Do something with this sensor value.
    }
    protected void onResume() {
        super.onResume();
        mSensorManager.registerListener(this, mLight, SensorManager.SENSOR_DELAY_NORMAL);
    }
    protected void onPause() {
        super.onPause();
        mSensorManager.unregisterListener(this);
    }
}
```

Sensors Coordinate System

- Following Sensors use this reference coordinate
 - Acceleration Sensor
 - Gravity Sensor
 - Gyroscope
 - Linear Acceleration Sensor
 - Geomagnetic Field Sensor

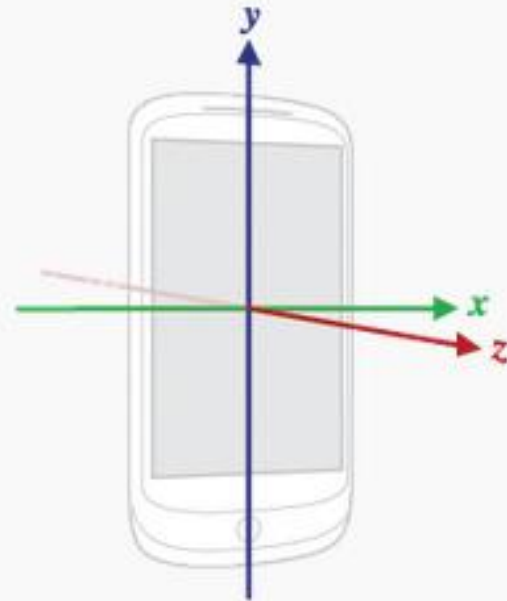


Figure 1. Coordinate system (relative to a device) that's used by the Sensor API.