

Санкт-Петербургский государственный университет

Математико-механический факультет

Кафедра системного программирования

Стабилизация показаний G-сенсора при помощи камеры

Курсовая работа студента 361 группы

Говейновича Сергея Геннадьевича

Научный руководитель

...../Валентин Оносовский/

/подпись /

Санкт-Петербург

2012

Содержание

1. Введение.....	3
2. Постановка задачи.....	4
3. Описание работы.....	5
3.1 Класс-контроллер на основе G-сенсора.....	5
3.2 Класс-контроллер на основе камеры.....	6
3.3 Различные подходы использования контроллеров.....	6
4. Заключение.....	8

1. Введение

G-sensor или Акселерóметр (от лат. *accelero* ускоряю и греч. *μετρέω* измеряю), или прибор, измеряющий проекцию кажущегося ускорения. Кажущееся ускорение есть ускорение, вызванное равнодействующей сил негравитационной природы, действующая на массу и равное этой силе отнесённой к величине этой массы. Современные акселерометры позволяют измерять ускорение сразу в трех плоскостях.

Электронные акселерометры часто встраиваются в мобильные устройства (в частности, в телефоны) и применяются в качестве шагомеров, датчиков для определения положения в пространстве, автоматического поворота дисплея и других целей.

В своей работе я рассмотрю использование G-сенсора мобильного телефона для определения положения телефона в пространстве. Основной проблемой при таком применении является высокая погрешность датчика и очень резкие, скачкообразные значения выдаваемых ими углов поворота.

Если создать мобильное приложение, в котором положение 3D объекта будет напрямую зависеть от показаний сенсоров, эти недостатки станут очевидны. Пользователь будет наблюдать дрожание объекта, в то время как телефон находится в статичном положении. При движении телефона объект будет перемещаться не плавно, как этого хочется, а резкими рывками. Для решения этих проблем применяются математические алгоритмы сглаживания показаний датчиков. Для некоторых целей таких сглаживаний будет достаточно, но для некоторых нужно нечто большее.

Для решения этих проблем была выдвинута идея фильтровать показания датчиков при помощи камеры мобильного телефона. Исследованием возможности такого подхода я и занимался.

2. Постановка задачи

Задача заключалась в создании технологии управления 3D сценой путем изменения положения телефона в пространстве.

требования:

- Плавное движение сцены (без резких скачков)
- Отсутствие запаздывания
- Стабилизация сцены при отсутствии движения
- Высокая точность определения угла поворота
- Отсутствие накопления погрешностей

3. Описание работы

Для тестирования технологии и визуальной оценки полученных результатов было создано приложение для Android устройств, в котором отображалась круговая 3D сцена с виртуальной камерой в центре. Сцена реализована с использованием библиотеки OpenGL. Виртуальная камера была реализована следующим образом.

```
public void setCam(GL10 gl) {  
    gl.glRotatef((float)Math.toDegrees(zy_rotation), 1.f, 0.f, 0.f);  
    gl.glRotatef((float)Math.toDegrees(xy_rotation), 0.f, 1.f, 0.f);  
}
```

Этой виртуальной камерой планировалось управлять путем поворота телефона.

Сама технология управления представляет собой 2 класса-контроллера.

1. Класс-контроллер на основе G-сенсора
2. Класс-контроллер на основе камеры телефона

3.1 Класс-контроллер на основе G-сенсора

С первым контроллером все более-менее понятно. Методом `loadNewSensorData` получаем показания датчика 30 раз в секунду, сглаживаем показания и отображаем изменения поворотом сцены.

```
private void loadNewSensorData(SensorEvent event) {  
    final int type = event.sensor.getType(); //Определяем тип датчика  
    if (type == Sensor.TYPE_ACCELEROMETER) { //Если акселерометр  
        setFilteredaccelData(event.values.clone());  
    }  
    if (type == Sensor.TYPE_MAGNETIC_FIELD) { //Если геомагнитный датчик  
        setFilteredMagnetData(event.values.clone());  
    }  
}
```

Использовал в этом классе 2 датчика: акселерометр и магнитометр (геомагнитный датчик). Методы фильтрации `setFilteredaccelData` и `setFilteredMagnetData` реализованы следующим образом. Приведу для примера один из них. Отличаются друг от друга только выбором коэффициента «r».

```
private void setFilteredMagnetData(float[] newMagnetData) {  
    float r = 8000.0f/10000;  
    this.magnetData[0] = this.magnetData[0] * r + (1 - r) * newMagnetData[0];  
    this.magnetData[1] = this.magnetData[1] * r + (1 - r) * newMagnetData[1];  
    this.magnetData[2] = this.magnetData[2] * r + (1 - r) * newMagnetData[2];  
}
```

Суть метода – «взвешенные значения». Т.е. вычисляем текущее значение по формуле:

*Значение = текущее значение * R + новое значение * R; где $0 \leq R \leq 1$*

Стоит заметить, что такой подход дает хорошее сглаживание показаний акселерометра с коэффициентом 0.7

Для сглаживания показаний магнитометра такого метода недостаточно. При слишком малом R остаются рывки в показаниях, а при слишком большом появляется задержка при движении.

На этом этапе стало понятно, что устранять погрешности при помощи камеры придется в основном для магнитометра. У магнитометра очень большие погрешности (лежа на столе значения могут колебаться до +-15 градусов). Еще магнитометр чувствителен к переменному магнитному полю, создаваемому бытовыми приборами, электропроводкой и т.д.

3.2 Класс-контроллер на основе камеры телефона

Этот класс был реализован при помощи библиотеки OpenCV. Вычисляя оптический поток определяем смещение центра картинки полученной с камеры и таким образом определяем на сколько сильно было изменено положение телефона и в какую сторону его повернули.

3.3 Различные подходы использования контроллеров

В своей работе я рассмотрел возможность использования этих контроллеров, как по отдельности, так и совместно. Далее я опишу рассмотренные способы.

1) Отдельное использование контроллера на основе g-сенсоре

Как было сказано выше не дает хороших результатов из-за больших скачков показаний магнитометра.

2) Отдельное использование контроллера на основе камеры телефона

Позволяет получить довольно плавное движение, но накапливает погрешность. Т.е. если повернуть телефон например на 90 градусов, а затем вернуть в исходное положение, то сцена окажется повернута относительно

своего начального положения. И чем больше таких поворотов будет совершено – тем больше будет это отклонение.

3) Взвешенная сумма контроллеров

Не дала положительного результата. На малых движениях все равно заметны сильные рывки из-за больших отклонений магнитометра. «Взвешенность» лишь уменьшает их примерно в 2 раза.

4) Спуск при движении камеры

Дает неплохой результат. Погрешность не накапливается т.к. используются показания датчиков, а камера служит лишь спусковым крючком. Хорошая стабилизация сцены при отсутствии движения. Из недостатков можно выделить то, что движение не плавное. Возможно этого удалось бы избежать на устройстве с более мощным процессором. Данная работа тестировалась на телефоне с тактовой частотой процессора 1000 мГц.

5) Камера при сильных движениях, сенсоры при слабых

Как и ожидалось такой подход дает самый плохой результат. Говорить тут не о чем.

6) Камера при слабых движениях сенсоры при сильных

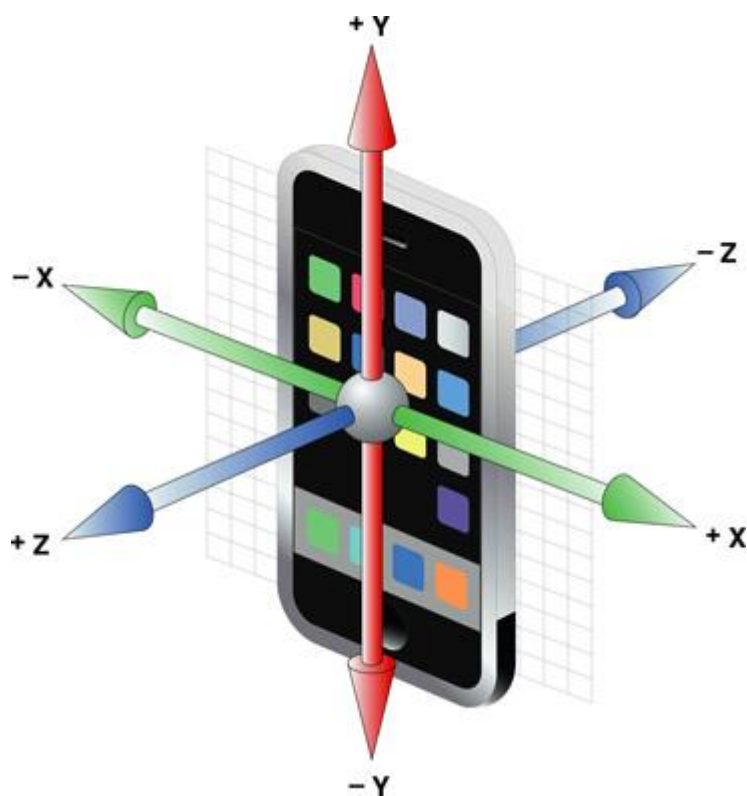
Дает хороший результат. Принцип действия заключается в следующем. При движении телефона на основе камеры определяем насколько сильное было смещение. Если смещение слабое – то используем показания контроллера основанного на камере. Если смещение сильное – используем контроллер на сенсорах. Метод не универсален, но для конкретной задачи можно подобрать коэффициенты которые дадут нужный результат. Основная проблема в накоплении погрешности при слабых движениях и ее резкой компенсации сенсорами при первом сильном движении. Такая компенсация довольно заметна. Но тут опять же можно пробовать компенсировать расхождения в показаниях контроллеров разными способами с разной частотой.

Этот метод показал себя наиболее перспективным и требует более глубокого изучения.

4. Заключение

Данная работа показала возможность компенсации погрешности датчиков и оптимизации контроллеров под конкретные задачи. Все требования поставленной задачи одновременно выполнить не удалось, но повысить одну характеристику в ущерб другой возможно. Был выявлен наиболее перспективный метод, который при более детальном изучении, возможно, даст универсальную технологию.

Так же в процессе разработки приложения была выявлена неприятная особенность магнитометра, о существовании которой я ранее не подозревал.



Показания магнитометра (проще говоря компаса), которые определяются поворотом оси Y относительно сторон света, каким то образом зависят еще и от поворота телефона вокруг оси Y. Т.е. при вращении телефона вокруг оси Y показания магнитометра заметно меняются (+-20 градусов) Такое поведение объяснить не удалось. Возможно проблема в сборке телефона на котором тестировалась программа. Проблема требует исследования.