ISSN 2250-3153

Framework for Accelerometer Based Gesture Recognition and Seamless Integration with Desktop Applications

Vijay D Rajanna

National Instruments, R&D Division Bangalore, M.S. Software Systems. BITS Pilani, India

Abstract- Accelerometer is one of the prominent sensors which are commonly embedded in new age handheld devices. Accelerometer measures acceleration forces in three orthogonal axes X, Y, Z. The raw acceleration values obtained due to the movement of device in 3D space which is hosting accelerometer can be used to interact and control wide range of applications running on the device and can also be integrated with desktop applications to enable intuitive ways of interaction.

The goal of the project is to build a generic and economic, gesture recognition framework based on accelerometer sensor and enable seamless integration with desktop applications by providing natural ways of interaction with desktop applications based on the gesture information obtained from accelerometer sensor embedded in Smartphone device held in user's hand.

This framework provides an alternative to the conventional interface devices like mouse, keyboard and joystick. With the integration of gesture recognition framework with desktop applications user can remotely play games, create drawings, control key and mouse event based applications. And since this is a generic framework, it can be integrated with any of the existing desktop applications irrespective of whether the application exposes APIs or not, or whether it is a legacy or a newly programmed application.

A communication protocol is required to transfer Accelerometer data from handheld device to desktop computer, and this can be achieved either through Wi-Fi or Bluetooth communication protocol. The project achieves data transmission between handheld device and desktop computer through "Bluetooth" protocol.

Once the Accelerometer data is received at desktop computer, the raw data is initially filtered and processed into appropriate gesture information after many computations through multiple algorithms. The key event publisher will take the processed gestures as input and converts them into appropriate events and publishes them on to the target applications to be controlled.

This framework makes interaction with desktop applications very natural and intuitive. And it also enables game and application developers to build creative games and applications which are highly engaging.

Index Terms- Gesture, Smart phone, Bluetooth, Sensors, Accelerometer, Pervasive Computing, Human Computer Interaction, Machine learning

I. INTRODUCTION

With the unparalleled advances in chipsets, sensors, computing technologies, we can change the way we interact virtually with the physical world around us. Each minute information associated with our day to day life is now modeled as living information and integrated into our professional and personal lives. Many of the novel ideas which were once though impossible to productize are now realized into tangible solutions by embedding sensors in hand held devices that can detect many number of events like location shift, minor displacement, barometric pressure, weather changes , light intensity, magnetic field proximity of objects etc. [6]

Cell phones can also impact the world around us in ways we cannot see, at least physically. The use of mobile devices in augmented reality has marked an era of innovative mobile solutions for the modern world. The augmented reality helps people to leave behind virtual artifacts like text, photos, video, avatars, and game clues for people to discover with their phones.

As an example, you can enter a building, view the lobby through your cell phone, and see messages and art pieces left behind by others for you to see and enjoy. Or, if you're at a conference in some place, you can view a nearby restaurant or bar through a mobile device and see comments made by other diners on food, service, atmosphere, or anything that they want to leave behind. [6]

These days sensors have become integral part of mobile devices, people use mobile devices beyond just for calling and messaging services. All the augmented reality solutions wouldn't have been possible if mobile devices were just a piece of hardware without sensors. These sensors are basically Micro-Electro-Mechanical Systems, or MEMS, which is a technology that in its most general form can be defined as miniaturized mechanical and electro-mechanical elements.

Gesture recognition framework implemented in this project, functions based on continuous stream of accelerometer values due to static and dynamic acceleration of the device.

II. OUTLINE

The project aims at building a comprehensive generic framework which enables a naive user to control any desktop computer application using simple and intuitive gestures processed based on accelerometer data which is embedded in handheld device.

This is achieved by combining and processing data from three major systems. Firstly the data from accelerometer is obtained, which is then transmitted from Smartphone to desktop computer through Bluetooth communication protocol and finally this raw data is converted into useful gesture information which controls various applications.

III. ACCELEROMETER

Accelerometer is an electromechanical device that will measure acceleration forces. These forces may be static, like the constant force of gravity pulling at your feet, or they could be dynamic forces caused by moving or vibrating the accelerometer.

Accelerometer sensor measures the linear acceleration in X, Y and Z directions based on the movement of hand held device. It simply measures acceleration due to motion, relative to gravity. These can be used to measure acceleration in one, two, or three orthogonal axes. They are typically used in one of three modes. [5]

- As an inertial measurement of velocity and position
- As a sensor of inclination, tilt, or orientation in 2 or 3 dimensions, as referenced from the acceleration of gravity (1 g = 9.8m/s^2);
- As a vibration or impact (shock) sensor.

Most accelerometers are Micro-Electro-Mechanical Sensors (MEMS). The basic principle of operation behind the MEMS accelerometer is the displacement of a small proof mass etched into the silicon surface of the integrated circuit and suspended by small beams. Consistent with Newton's second law of motion (F =ma), as an acceleration is applied to the device, a force develops which displaces the mass. The support beams act as a spring, and the fluid (usually air) trapped inside the IC acts as a damper, resulting in a second order lumped physical system. This is the source of the limited operational bandwidth and non-uniform frequency response of accelerometers. [1],[2],[5]

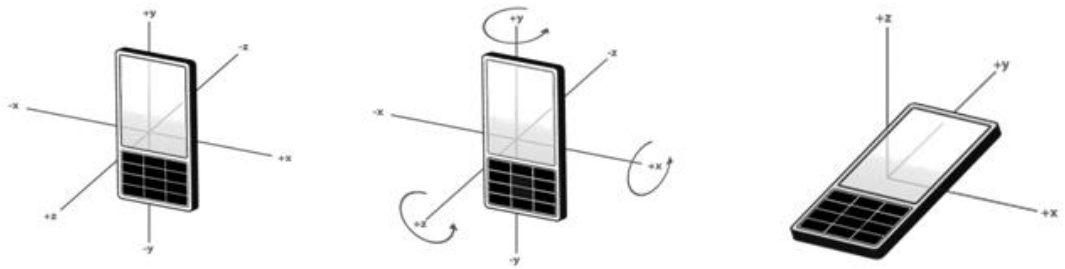

Figure 1: Accelerometer Axes - Courtesy - qt.nokia.com

IV. BLUETOOTH

Bluetooth is a short-range communications technology that is simple, secure, and everywhere. You can find it in billions of devices ranging from mobile phones, computers to medical devices and in home entertainment products. It is intended to replace the cables, which connects devices to enable data transfer, while maintaining high levels of security. In the current project Bluetooth is used to transmit accelerometer data from Smartphone device to the desktop computer at a constant rate, making it a completely wireless system.

The key features of Bluetooth technology are robustness, low power and low cost. The Bluetooth specification defines a uniform structure for a wide range of devices to connect and communicate with each other. [3],[4]

SETTINGUP A BLUETOOTH SERVER

- Selecting a destination device to be connected to, which sometimes involves device discovery?
- Agreeing on a protocol to be communicated.
- Making an outgoing connection or accepting an incoming connection.
- Establishing connection for a published service on a port specified.

SETTING UP BLUETOOTH CLIENT

- Search for published services with specific UUID
- Make an outgoing connection with the peer Bluetooth device with the same UUID as client.

- Start exchanging data through client server sockets.

V. DESIGN OF GESTURE RECOGNITION FRAMEWORK

The high level architecture of gesture recognition framework is modularized into two major components.

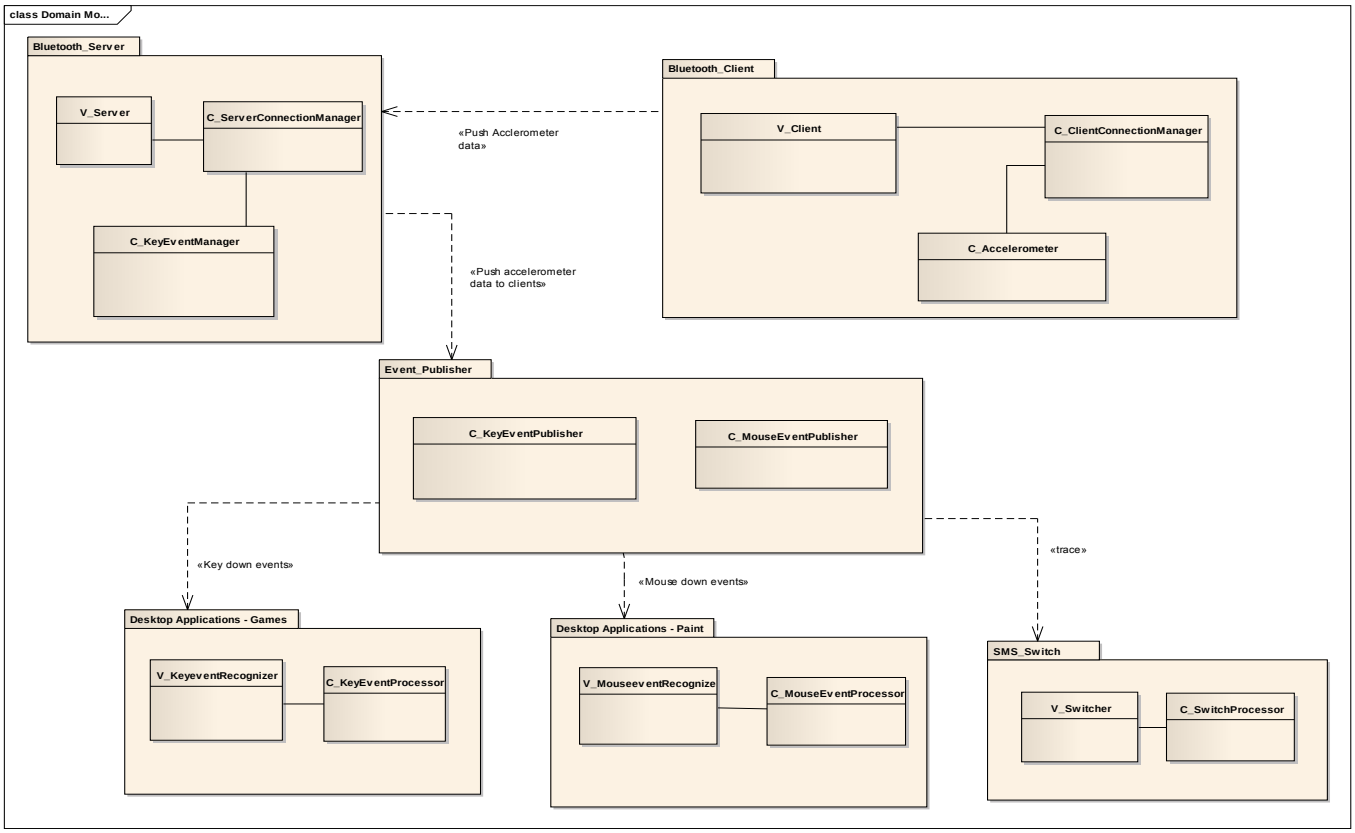

Figure 2: Domain Model

1. GESTURE GENERATOR

Gesture generator module is responsible for transferring the gesture information in the form of raw accelerometer values from user's handheld device to desktop computer, which is running the target application to be controlled. Gesture generator further comprises of two independent modules.

1.1. ACCELEROMETER INITIATER

The accelerometer initiator module running on user's hand held device will initiate accelerometer hardware by setting up initial values, time interval for each sample and starts listening to accelerometer output data which returns the acceleration due to gravity of the device in three dimensions. This is helpful in understanding which way the user is moving or tilting the phone, and based on this information it is programmatically possible to control applications and games running on user's PC

1.2. BLUETOOTH CLIENT

Bluetooth client running on hand held device will search for peer Bluetooth servers publishing various services, In this case the client will search for file or text transfer services. After discovering the server which is publishing file/text services client makes an outgoing connection with Bluetooth server and starts sending accelerometer data to server using series of "send" function calls.

2. GESTURE RECOGNIZER AND APPLICATON CONTROLLER

Gesture recognizer is responsible for receiving raw accelerometer data from Bluetooth client, filtering out redundant information, and running an intelligent algorithm to convert raw accelerometer data into useful gesture information. Once raw accelerometer data is processed into gesture information, this is fed to Application controller module which directly controls target applications through emulating key and mouse events. Gesture recognizer comprises of three independent modules.

2.1. BLUETOOTH SERVER

Bluetooth server running on user's PC will publish text/file transfer services, and waits for an incoming connection from the client. The published services are discovered by Bluetooth client application running on user's hand held device. Once the service is discovered the client will establish a connection with the server and starts pushing accelerometer data into server socket. Server will continue to receive Bluetooth client data through a series of "Receive" function calls.

2.2. GESTURE PROCESSOR

Gesture processor receives raw accelerometer data from Bluetooth server module. Accelerometer generates continuous stream of data, but not all of the information generated is essential for the applications. Hence in the first stage of processing all the redundant accelerometer data is filtered out and only those values which indicate significant change in the position of the device is retained. And in the second stage filtered data is fed into an intelligent algorithm to convert raw accelerometer data into useful gesture information.

2.3. APPLICATION CONTROLLER

Application controller module receives processed gesture information from gesture processor module and converts them into appropriate events. The intelligence is built into the application controller so that it takes control of the target application's key and mouse event handlers and starts feeding the application with emulated keyboard and mouse events. This makes it possible to achieve expected functionality from the target application using gesture information.

I. IMPLEMENTATION OF GESTURE RECOGNITION FRAMEWORK

As mentioned in the earlier part, the objective of the project is to build generic and economic gesture recognition framework to bridge the gap and enable seamless interaction between desktop applications and sensors embedded in hand held devices.

Accelerometer being the sensor of interest, the project uses accelerometer sensor available on Nokia mobile phones. Nokia phones run on Symbian Operating system, and Symbian Operating system provides access to all the embedded sensors in two ways.

1. Native Symbian APIs to access the sensors.
2. Through Qt framework APIs.

This current implementation uses Qt framework to access phone's accelerometer. Qt framework was chosen over native Symbian APIs since Qt framework provides a well-established framework to work with these sensors and also exposes significant amount of functionality of these sensors and finally, Qt framework is developer friendly.

To demonstrate the integration of gesture recognition framework with existing desktop applications, the project was successful in integrating this gesture recognition framework with, Need For Speed game and Windows Paint application on Windows Operating System.

1. RACING GAMES

Racing games are a category of games which require users to control and steer the virtual racing object through keyboard or joystick interface. The gaming applications should be able to interpret gestures and execute desired functionality, the state of the racing objects continues to be in its last well known state until a new gesture is received and interpreted. With the integration of gesture recognition framework, user will be able to control racing objects through gestures generated from handheld device. User will be able to steer the racing object to left, right, accelerate, decelerate and brake just by corresponding gestures from handheld device. The integration consists of four modules.

- Accelerometer module on mobile device
- Bluetooth server running on user's PC
- Bluetooth Client running on mobile device
- Gesture Recognition Framework for Gaming

1.1 DEMONSTRATION OF WORKING MODEL

The video of the demonstration of working model, which shows integration of gesture recognition framework with Need For Speed racing game, can be found here.
http://youtu.be/EYsi7V2p1m4

2. PAINT BURSH

Paint Brush is an accelerometer controlled painting application which is an integration of gesture recognition framework with windows paint application. Paint brush enables user to draw basic line diagrams in 2D space. Input for the application are the gestures generated from user's handheld device, as the user tilts or waves his handheld device in the air. The integration consists of four modules.

- Accelerometer module on mobile device
- Bluetooth server running on user's PC
- Bluetooth client running on mobile device
- Gesture Recognition Framework for Painting

2.1 DEMONSTRATION OF WORKING MODEL

The video of the demonstration of working model, which shows integration of gesture recognition framework with windows paint application can be found here.
http://youtu.be/sunhfFyPYJs

VI. FUTURE ENHANCEMENTS

The generic nature of framework enables its integration with various applications and games from multiple domains and across different platforms. Below are the few feasible future enhancements.

- Gesture to voice conversion
- Gesture based multiplayer games,
- Economic smart class solutions,.
- NFC integration,
- Photo viewer,
- Controlling power point presentation

ACKNOLEDGEMENT

I would like to thank, Birla Institute of technology and the faculty members, Prof. Neena Goveas and Prof. Mangesh Bedekar who guided me with valuable suggestions during the course of project.

I offer my sincere gratitude to my mentor Mr. Ashwin Rao, and colleagues Mr. Anand Reddy and Mr. Santhosh Rao at NOKIA who shared all their expertise and experience at each stage while I worked on this project.

I am indebted to all those forum members on MSDN, Qt group, Code guru and Code Project, who answered my every query and provided valuable guidance.

Finally I would like to thank my organization NOKIA who allowed me to carryout this project.

REFERENCES

[1] Uwe Hansmann, Lothar Merk, Martin Nicklous (2004) “Principles Of Mobile Computing”, Wiley India.

[2] Tan tran Duc(2009), “Modeling And Simulation Of The Capacitive Accelerometers”, Grin Verlag,

[3] Muller J N, (September 8, 2000) “Bluetooth Demystified”, McGraw-Hill Professional.

[4] Lawrence Harte (September, 2007), “Bluetooth Essentials for Programmers”, Cambridge University Press.

[5] http://sensorwiki.org

[6] http://www.last100.com

[7] Dee Bakker, Diane McMichael Gilster, Ron Gilster. Bluetooth End to End. Wiley. January , 2002.

[8] Ville Kaajakari. Practical MEMS: Design of microsystems, accelerometers, gyroscopes, RF MEMS, optical MEMS, and microfluidic systems. Small Gear Publishing. March, 2009.

[9] QT, Cross platform application and UI framework document.

[10] QT Mobility Project Reference Documentation.

AUTHORS

Vijay Dandur Rajanna, National Instruments, R&D Division Bangalore., M.S. Software Systems. BITS Pilani, India, E-Mail : mail2vijaydr@gmail.com

Vijay Dandur Rajanna received his Master of Science degree from Birla Institute of Technology and Science, Pilani in "Software Systems", and his Bachelor of Engineering degree in "Computer Science and Engineering" from Visvesvaraya Technological University. Before joining National Instruments, Vijay worked with Robert Bosch and Nokia Research and Development divisions, majorly working on research projects. His areas of research interests include "Artificial Intelligence, Human Computer Interaction and Pervasive Computing".