Project Proposal: Extending the WiiMote into 3D

João Lourenço

February 26, 2010

1 Principal Invesigator

João LOURENÇO. Currently completing a Bachelor of Science (Honours) in Computer Science at Rhodes University. Gavin Relly Post Graduate Village, P. Bag 1028, Grahamstown, 6139, South Africa. g07L2273@campus.ru.ac.za

2 Title of Project

Extending the WiiMote into 3D.

3 Supervisor

Dr. Hannah Thinyane, Computer Science Department, Rhodes University. H.Thinyane@ru.ac.za. Collaboration with Prof. Richard Foss (R.Foss.ru.ac.za) and Prof. Shaun Bangay (S.Bangay@ru.ac.za), both from the Computer Science Department, Rhodes University.

4 System Requirements

The following hardware is necessary for this project:

- Two Nintendo® Wii Remote Controllers - also known as "WiiMotes"
- Bluetooth-enabled Windows Computer (for connectivity with WiiMote)
- Infra-red LED array (used for IR projection)
- Infra-red reflectors
- Other systems for comparison (such as iPhone/iPad for TouchScreen, Standard Microsoft® Windows Computer for Mouse and Keyboard, Microsoft®'s Project Natal and Xbox 360 for 3D Interface comparison, Air-Mouse for further 3D Interface comparison)

The following software is necessary for this project:

- Microsoft® Visual Studio with C# .Net 3.5
- WiiMoteLib library (C# operability with the WiiMote)
- Gesture recognition library
- Testing framework (3D interactive application) to use library (possibility of using the Microsoft® XNA Framework or UCMAN with Google® SketchUp)

5 Objective of Research

Due to the migration from 2-dimensional output devices (such as standard monitors) towards 3-dimensional output devices (such as 3D televisions and Virtual Reality), there is a possibility to migrate from standard input using the mouse and keyboard to other 3-dimensional devices [6] (such as Microsoft®'s Project Natal, and, hopefully, the WiiMote 3D). To interact with 3-dimensional software using 2-dimnesional input devices (such as computer games and 3D modeling programs), software designers are currently using complicated keyboard sequences and mouse combinations to map the input into the third dimension [1, 16]. As such, there is an even greater need to address the issue of the third dimension.

The main aim of this project is to develop an intuitive 3D interaction framework that will allow a user to interact within 3-dimensions. A study will be undertaken to compare the intuitiveness of the proposed interaction technique with existing techniques and devices.

6 Background and History

6.1 Theory

6.1.1 Triangulation

The linear triangulation of a point in $\mathbb{R}^3$from two projections, whose views are known, is simple in a non-noisy environment using geometry [5]. For triangulation in 2 dimensions, let:

A Observer 1

B Observer 2

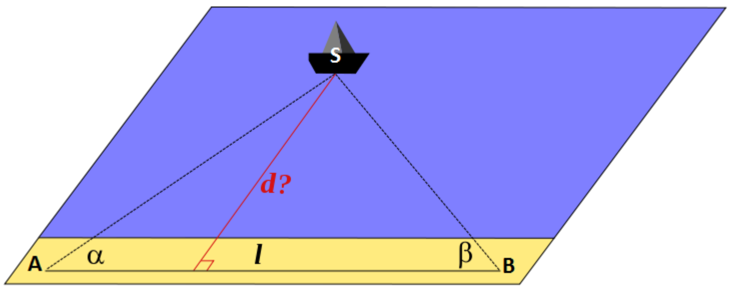
S The observed point

l The line between A and B

d The perpendicular distance from the line between A and B and the S

α The angle between the l and AS

β The angle between l and BS



Using Pythagoras' theorem, d can be defined as:

$$d = \frac{l}{\left(\frac{1}{\tan\alpha} + \frac{1}{\tan\beta}\right)} = \frac{l \cdot \sin\alpha \cdot \sin\beta}{\sin(\alpha+\beta)}$$

This can be extended into the third dimension using both horizontal triangulation and vertical triangulation to calculate an estimate of an object's position in $\mathbb{R}^3$.

6.1.2 Paths and Gesture Recognition

The paths need to be continuously recorded as these will be used for gesture processing, and will be needed in case of backtracking or an undo/redo framework.

A 3D gesture may be recognised and processed using a hidden Markhov model (HMM) [6], which is a simple dynamic Bayesian network. This spaciotemporal model, which is defined below, reduces the 3 dimensional complexity of the hand gesture into a 2 dimensional problem, and analyses and categorizes these gestures using a state machine. This model has been employed, with success, in speech [12] and handwriting recognition [4].

The HMM can be defined as such[12]:

- states: $S = \{s_1, s_2, \ldots, s_N\}$ where the state at time t is q_t and N is the number of states in the model
- symbols: $V = \{v_1, v_2, \ldots, v_M\}$ where M is the number of distinct observation symbols per state
- probablity vector (state transition probability distribution): $A = \{a_{ij}\}$ where $a_{ij} = P(q_{t+1} = s_j \mid q_t = s_i)$ $1 \le i, j \le N$
- observation state probability distribution in state j: $b_j(k) = P(v_k att | q_t = s_j)$ $1 \le j \le N$ $1 \le k \le M$
- initial state distribution: $\pi = \{\pi_i\}$ where $\pi_i = P(q_1 = s_i)$

Due to the requirement of large training sets and the inflexibility of its classification model, another Bayesian approach might be preferable, as a study in head gesture recognition showed[15]. The sparse classification model used in that study could suit the needs of this project better, but is more complicated to implement.

Under the assumption that hand gestures with four points that need to be followed do not vary from person to person as much as speech or handwriting does (there are much fewer degrees of freedom), the HMM will be investigated first. Furthermore, the number of states the model will be limited, which removes the requirement for excessively large training sets.

6.2 Hardware

6.2.1 Nintendo® WiiMote

The WiiMote, whose development started in 2001 and completed in 2005[13], is a Bluetooth-enabled device that "offers the user an intuitive, natural way to play games" [10]. The device works by detecting infra-red light sources and sending the position data to a master device that subsequently processes the data. The WiiMote can detect up to four light sources at a time.

6.2.2 Microsoft® Project Natal

Project Natal, set for release in 2010, is Microsoft®'s attempt at controller-free interaction. The device uses an infra-red projector and a monochrome CMOS sensor to detect a user's position and posture under any light conditions. The device and bundled software will be capable of full body motion capture of multiple bodies in the same room, employing facial and gesture recognition [7].

6.3 Software

6.3.1 Microsoft® Visual Studio with C# .Net 3.5

This integrated development environment will be used to use the WiiMoteLib in a Visual C# .Net project.

6.3.2 WiiMoteLib

This library, developed by Brian Peek, exposes the WiiMote's functionality, enabling the acquisition of infra-red positioning of up to four sources [11].

6.3.3 Microsoft® XNA Framework

The Microsoft® XNA Game Studio framework was created by Microsoft to enable users to create games for deployment on the Windows and Xbox platforms [8]. This application can be used to create an immersive 3D environment for testing purposes.

6.3.4 Google® SketchUp

Google® SketchUp is a software package used primarily for 3D modeling in the conceptual stages of design [2], and provides an API that is programmable in Ruby [3].

6.3.5 UCMAN

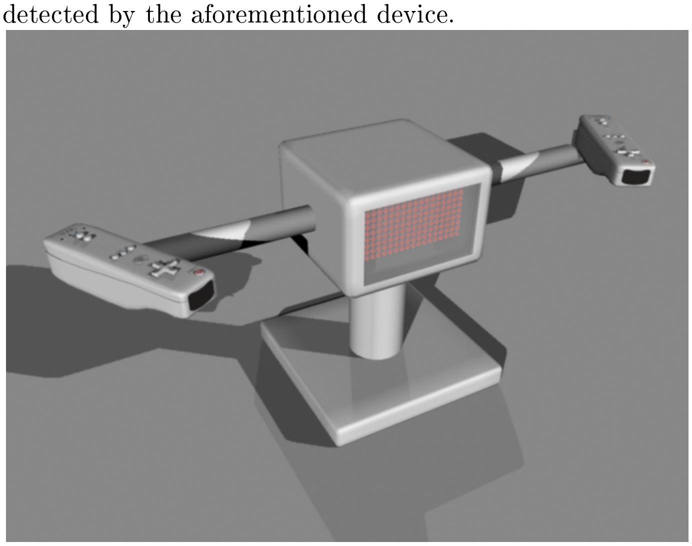
UCMAN is a live audio network configuration tool used for the control of large networks from a single computer console, a system currently in development for UMANnet and similar to the current product, UNOS [14]. Plans to integrate Google SketchUp into this system are currently underway, and, as such, a new interaction interface could be used in this software.

7 Approach

The approach will be divided into several sections, namely WiiMote 3D Hardware, WiiMote 3D Framework Development, Testing Framework Development and, finally, Comparison of Interfaces.

7.1 WiiMote 3D Hardware

A harness, holding the two WiiMotes in fixed positions and rotations in space (relative to each other), will be constructed. An array of infra-red LED's will be fitted to the harness such that they are not in view of the WiiMote's IR cameras. The user will place up to four IR reflectors on his hand(s), which will be detected by the aforementioned device.



7.2 WiiMote 3D Framework Development

The WiiMote 3D framework will be separated into several layers, upon which an interactive 3D application will be built.

3D Interactive Application
GestureLib
WiiMote3DLib
WiiMoteLib
Operating System and Drivers

7.2.1 WiiMoteLib

The current WiiMoteLib library will reside at the bottom of this stack, interacting with the operating system. This library provides access to up to four Bluetooth-paired WiiMotes, and the various data sent by the devices.

7.2.2 WiiMote3DLib

This library will be developed, and will provide real-time access to the four IR sources that will be detected by the WiiMotes. The library will include motion tracking capabilities, exposing the paths of each of the sources, and providing intelligent tracking of each of the sources.

7.2.3 GestureLib

This library will developed unless a suitable library is found. This should provide the ability to detect functional and descriptive gestures. Functional labels "describe what the gesture does in an interface" while descriptive labels "describe how the gesture is performed, such as its movement" [9].

7.3 3D Interactive Application - Testing Framework

There are several possibilities for the application (such as a modeling program or game using the Microsoft® XNA Framework, live audio configuration tool using UCMAN and Google® SketchUp). These will both provide suitable comparisons to other applications.

7.4 Comparison of Interfaces

A comparison between the usability and intuitiveness of the chosen standard application(s) and the one(s) developed will be carried out.

8 Project Timeline

Deadline	Activity
22 February 2010	Present proposal to staff
26 February 2010	Project proposal completed in LaTeX
22 March 2010	Gather relevant papers and do readings
29 March 2010	WiiMote3DLib completed
3 May 2010	GestureLib completed
21 May 2010	Literature review
5 July 2010	3D Interactive application complete
19 July 2010	Design tests for statistical analysis
20 July 2010	Oral presentation
17 August 2010	Poster presentation
27 August 2010	Statistical tests completed
13 September 2010	Short paper submitted
25 October 2010	Final oral presentation complete
1 November 2010	Project complete and submitted
8 November 2010	Website complete

References

[1] Chen, M., Mountford, S. J., and Sellen, A. A study in interactive 3-d rotation using 2-d control devices. In *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1988), vol. 22, ACM, pp. 121–129.

[2] Google. Google sketchup. Internet - http://sketchup.google.com/, February 2010.

[3] Google. Google sketchup ruby api. Internet - http://code.google.com/apis/sketchup/, February 2010.

[4] Ha, J. Y., Oh, S. C., Kim, J. H., and Kwon, Y. B. Unconstrained handwritten word recognition with interconnected hidden markov models. *IWFHR 3* (May 1993), 455–460.

[5] Hartley, R. I., and Sturm, P. Triangulation. *Computer Vision and Image Understanding 68*, 2 (November 1997), 146–157.

[6] Kim, I.-C., and Chien, S.-I. Analysis of 3d hand trajectory gestures using stroke-based composite hidden markov models. *Applied Intelligence 15*, 2 (2001), 131–143.

[7] Microsoft. Project natal. Internet - http://www.xbox.com/en-US/live/projectnatal/default.htm, February 2010.

[8] Microsoft. Xna creator's club. Internet - http://creators.xna.com/en-US/, February 2010.

[9] Nielson, M., Moeslund, T. B., Storring, M., and Granum, E. *HCI Beyond the GUI: Design for Haptic, Speech, Olfactory and Other Non-traditional Interfaces.* Morgan Kaufman, 2008, ch. 3 - Gesture Interfaces, pp. 75–106.

[10] Nintendo. About wii. Internet - http://www.nintendo.co.za/, February 2010.

[11] Peek, B. Wiimotelib - .net managed library for the nintendo wii remote. Internet - http://www.brianpeek.com/blog/pages/wiimotelib.aspx, February 2010.

[12] Rabiner, L. R. *Readings in Speech Recognition.* Morgan Kaufmann Publishers Inc., 1990, ch. 18, pp. 267–296.

[13] Rothman, W. Unearthed: Nintendo's pre-wiimote prototype. Internet - http://gizmodo.com/gadgets/exclusive/unearthed-nintendo-2001-prototype-motion+sensing-one+handed-controller-by-gyration-294642.php, August 2007.

[14] UMANnet. Uman. Internet - http://umannet.com/index.php/Home.html, February 2010.

[15] Wong, S.-F., and Cipolla, R. Continuous gesture recognition using a sparse bayesian classifier. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition* (Washington, DC, USA, 2006), IEEE Computer Society, pp. 1084–1087.

[16] Zeleznik, R. C., Forsberg, A. S., and Strauss, P. S. Two pointer input for 3d interaction. In *Proceedings of the 1997 Symposium on Interactive 3D Graphics* (1997), pp. 115–120.