

Multimodal Human Robot Interaction

Mark Swaine

Project Goals

- To develop different methods of interaction between humans and robots
- To develop a system that manages inputs from different interfaces
- To ascertain which interface is best suited to effective human-robot interaction

System Architecture



The Server

- Manages the data being sent from the different modalities
- Handles the communication between the Lego NXT Robot and the computer
- Communicating with the Lego NXT robot is done using a channel created between the computer and the robot by means of a Bluetooth COM port.

The Server

- Developed in C# using Microsoft Visual Studio 2008 and the .NET Framework 3.5
- Use of NKH.Mindsqualls library – API to communicate with the Lego NXT robot

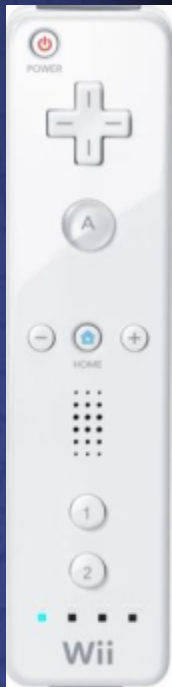
The Server

- Listens for incoming connections on port 3000
- Multithreaded

The Server

- Server protocol:
 - `<modality>;<power>;<turnRatio>`
 - Eg: xbox360;60;7

Nintendo Wiimote



- Connects to computer over Bluetooth
- Picks up changes in motion, thereby possibly being an effective gestural interface
- Tilting motions used to control speed and direction

Nintendo Wiimote

- Developed in C# using Microsoft Visual Studio 2008 and the .NET Framework 3.5
- WiimoteLib library as an API to communicate with the Wiimote

Nintendo Wiimote

- The accelerometer is extremely sensitive, so adjustments had to be made to desensitize the returned values

Microsoft Xbox 360 Controller

- is a suitable tangible user interface for controlling robot movement
- connects to a computer using a Microsoft wireless receiver

Microsoft Xbox 360 Controller



Microsoft Xbox 360 Controller

- Developed in C# using Microsoft Visual Studio 2008, the .NET Framework 3.5 and the Microsoft XNA Framework
- Microsoft XNA Framework is aimed at game development for the Windows and Xbox 360 platforms – simplifies integration of game controllers

Web

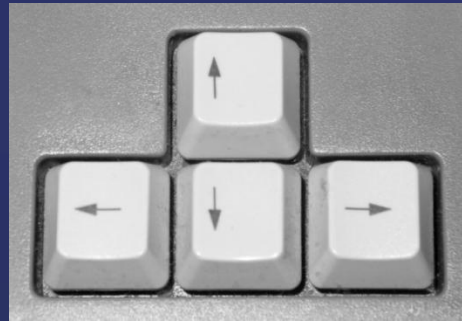
- Web 2.0 revolutionises methods of interaction, styles of development and content
- The idea of Web 2.0 is to let users run applications entirely through a browser without any local machine installation
- Achieved with AJAX (Asynchronous JavaScript And XML)

Web

- Developed in Java using the Eclipse IDE and the Google Web Toolkit (GWT)
- GWT enables rapid development of AJAX applications through asynchronous remote procedure calls
- GWT cross-compiles from Java to JavaScript

Web

- Use arrow keys to navigate the Robot



- Java applet displays webcam

Mobile

- Allows for remote mobile robot interaction
- Makes use of a Mobile Information Device Profile (MIDP)-enabled mobile phone

Mobile

- Developed in Java using the Eclipse IDE and the Java 2 Micro Edition framework
- Use arrow keys or numeric keypad to navigate the robot

Results

- Currently in the process of obtaining results.
- Test involves navigating the robot around obstacles using each of the modalities and determining which method of interaction is the most effective

Demo

Questions...

?