Hacking the Nintendo Wii Remote

The global hacking community has collectively reverse-engineered a significant portion of one of the world's most sophisticated and common input devices. And they're putting it to uses its designers never intended.

Johnny Chung Lee
Carnegie Mellon University

In November 2006, Nintendo released its fifth home videogame console, the Nintendo Wii. The company's previous game console, the Gamecube, hadn't fared well in terms of market share against the much higher-powered alternatives released by its competitors, Microsoft and Sony. At first the Wii also seemed significantly underpowered relative to its competitors. However, one year later it became the market leader of its console generation, selling over 20 million units worldwide.[1] This success is largely attributable to the innovative interactive technology and game-play capabilities introduced by the console's game controller, the Wii remote, shown in Figure 1.

The Nintendo Wii remote, or Wiimote, is a handheld device resembling a television remote, but in addition to buttons, it contains a 3-axis accelerometer, a high-resolution high-speed IR camera, a speaker, a vibration motor, and wireless Bluetooth connectivity. This technology makes the Wii remote one of the most sophisticated PC-compatible input devices available today; together with the game console's market success, it's also one of most common. At a suggested retail price of US$40, the Wii remote is an impressively cost-effective and capable platform for exploring interaction research. Software applications developed for it have the additional advantage of being readily usable by millions of individuals around the world who already own the hardware.

I've recently begun using Internet video tutorials to demonstrate interaction techniques supported or enabled by the Wii remote. In just a few weeks, these tutorials have received over six million unique views and generated over 700,000 software downloads. In this article, I will talk about the Wii remote's technology, cover what's involved in developing custom applications, describe intended and unintended interaction techniques, and outline additional uses of the device.

Inside the Wii remote

Although the Wii remote's official specifications are unpublished, the global hacking community has collectively reverse-engineered a significant portion of the technical information regarding its internal workings. Much of this work has been collected in online wikis at http://wiili.org and http://wiibrew.org. The body of knowledge at these sites represents contributions from numerous individuals and constitutes the source for most of the information presented in this section.

Because many low-level details are available online and, furthermore, are likely to be refined and updated as more information is uncovered, the following descriptions of each major Wii remote component represent only higher-level details relevant to building custom applications.

Infrared camera tracker

In the tip of each Wii remote is an IR camera sensor manufactured by PixArt Imaging, shown in Figure 2. The camera chip features an integrated multiobject tracking (MOT) engine, which

Published by the IEEE CS ■ 1536-1268/08/$25.00 © 2008 IEEE

Figure 1. The Nintendo Wii remote game controller. (Copyright for all photos, Figures 1–10, Johnny Chung Lee.)

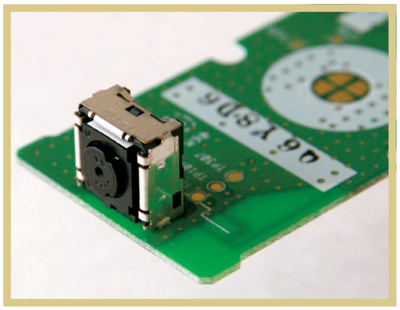


Figure 2. The PixArt IR camera chip. Integrated multiobject tracking minimizes wireless data transmission.

provides high-resolution, high-speed tracking of up to four simultaneous IR light sources. The camera sensor's exact specifications are unpublished, but it appears to provide location data with a resolution of 1,024 × 768 pixels, more than 4 bits of dot size or light intensity, a 100 Hz refresh rate, and a 45 degree horizontal field of view. The integrated hardware object tracking minimizes the data transmitted over the wireless connection and greatly simplifies the implementation of camera-based tracking applications.

These specifications outperform comparably priced webcams, which typically provide 640 × 480 tracking at 30 Hz. Webcams also require significant CPU power to perform real-time computer-vision tracking. Specialized IR camera trackers, such as those from Natural Point Systems (www.naturalpoint.com), can provide 710 × 288 tracking at 120 Hz, but at a significantly higher cost of $180.

Accelerometer

Analog Devices manufactures the ADXL330, a 3-axis linear accelerometer that provides the Wii remote's motion-sensing capability. It has a +/–3 g sensitivity range, 8 bits per axis, and a 100 Hz update rate.

Buttons

The Wii remote has 12 buttons. Four are arranged in a standard directional-pad layout. One button is on the bottom providing a trigger-like affordance for the index finger. The remaining seven buttons are intended to be used by the thumb. The remote design is symmetric, allowing use in either the left or right hand.

Vibration motor (tactile feedback)

A small vibration motor provides tactile feedback. The motor is similar to those used in cell phones. The motor state has only binary control (on and off), but you can vary the feedback intensity by pulsing the motor activation—that is, by rapidly turning the motor on and off at different duty cycles.

Light-emitting diodes (visual feedback)

Four blue LEDs at the bottom of the remote are typically used to indicate player IDs (1 to 4). Each LED's state is individually addressable. Similarly to the vibration motor, pulsing the state creates varying levels of brightness.

Speaker (auditory feedback)

A small speaker in the remote's center supports in-game sound effects and user feedback. The audio data streams directly from the host with 4-bit, 4 KHz sound similar in quality to a telephone.

Bluetooth connectivity

Communication runs over a wireless Bluetooth connection. The connection uses a Broadcom 2042 chip, which Broadcom designed for devices that conform to the Bluetooth Human Interface Device standard, such as keyboards and mice.

The remote isn't 100 percent compliant with the HID standard, but it can connect to many Bluetooth-capable computers.

Internal flash memory

The onboard memory is approximately 5.5 Kbytes. It's used for adjusting the device settings, maintaining output state, and storing data. Nintendo designed it to let users transport and store a personal profile, called a *Mii*. This memory allows data and identity to be physically associated to a given remote.

Expansion port

At the base of the remote is a proprietary six-pin connector used to communicate with and power extension controllers such as the Nintendo Nunchuk, Classic Controller, or a guitar controller. These extensions provide alternative form factors and additional input capabilities.

The port provides 3.3 V of power and 400 KHz of I2C serial communication, to which a microcontroller can easily interface and effectively provide a Bluetooth-to-I2C bridge.

Batteries

The Wii remote uses two AA batteries and has an operating time between 20 and 40 hours, depending on the number of active components. Approximately 8 bits of battery-level resolution are available.

Developing custom applications

Although Nintendo offers a relatively inexpensive development kit for the Wii console, its legal agreement severely limits the types of applications you're permitted to develop using its tools. Alternatively, you can quite easily connect the Wii remote to a personal computer via Bluetooth and immediately begin developing custom applications. The remote's compatibility with the Bluetooth HID specification manifests on the host computer as a joystick.

Software libraries for connecting to a Wii remote, parsing the input report data, and configuring the controller are available for nearly every major development platform on Windows, Mac OS, and Linux. The open development community has created these libraries, and you can download them for free. Because these software APIs are in active development and might change rapidly, I won't discuss them in detail. Visit http://wiili.org and http://wiibrew.org for more information.

Accessing the data is usually as simple as reading values from an array or an appropriately named member variable of a Wii remote class object, such as `accelX = remote.accelerometer.x`. The computer receives input reports 100 times per second, providing low-latency data. As the software libraries evolve, they might support event queues, derivative values, and utilities that compute useful transformation matrices or recognize gestures, thereby simplifying application development.

In many cases, the most difficult part of this process is getting the Bluetooth pairing to occur successfully. Because the Wii remote isn't 100 percent HID compliant, it might work only with certain Bluetooth chipsets and driver software. However, once a pairing is successful, the configuration is typically quite reliable. After you've connected the Wii remote and installed the software library, developing custom applications is straightforward.

The projects I describe in this article are C# Windows software applications using Brian Peek's Managed Library for the Wiimote.[2]

Wii console interaction techniques

Wii users hold the remote controller in one hand and point it at a television that has a Wii sensor bar either above or below the screen. The term "sensor bar" is a misnomer because the device doesn't contain sensors; rather, it contains two groups of infrared LEDs. The Wii remote's IR camera sees the two groups and provides a method of laser-pointer-style input. The software can transform the x, y coordinate dot pairs to provide an averaged x, y coordinate pair, a rotation, and a distance. The x, y, and rotation values correspond to the controller's yaw, pitch, and roll, respectively, and the distance is estimated using the known physical separation of the two IR groups and the camera's fixed field of view.

Common Wii game interactions using the controller as a pointer include selection, navigation, aiming a weapon or tool, drawing, rotating objects, and push-pull interactions. Although the remote is frequently used as a pointer, no game currently makes a significant attempt to ensure that the cursor's on-screen position accurately matches the screen plane's intersection with the ray defined by the axis of the Wii remote. Assumptions are made regarding the screen's visual angle and the scale of movement. However, this doesn't appear to have a significant impact on users' pointing ability in most contexts. This might be because the game provides constant visual feedback of the cursor position, which lets users rely on relative movements rather than absolute aiming.

Use of the accelerometer data within Wii games varies from basic shake triggering, to tilt-and-balance control, to simple gesture recognition. WiiSports, the mini-game that comes with the Wii console, might currently involve the most intricate use of the accelerometer data. WiiSports encourages players to swing the remote in the imaginary context of bowling, boxing, or playing tennis, baseball, or golf. The game appears to register subtle variations in swing dynamics and thus affect the simulation. Though the motion recognition might not necessarily be accurate, the experience is quite compelling. However, the majority of existing games simply employ shake recognition to trigger an event similar to a button press.

As in other game consoles, the buttons of the Wii remote are heavily employed for triggering input events. Frequently, games use the Nunchuk attachment, which is designed to be held in the nondominant hand and adds more buttons, an analog joystick, and another accelerometer for independent motion sensing in each hand. In total, the Wii remote with Nunchuck attachment provides 13 digital inputs, 12 analog controls, and auditory, visual, and tactile feedback.

Remote interaction techniques without the Wii console

The Wii remote's rich level of input and output combined with the ease of PC connectivity have made it a popular platform for exploring alternative control schemes for existing applications. Many initial projects in the developer community involved using the motion- and tilt-sensing capabilities for robotic control and synthesized musical performance. For example, see the Wii remote's use in making the iSobot perform combat motions (www.robodance.com/Nintendo-wii-i-sobot.php) and in composing music in the Kyma X development environment (www.youtube.com/watch?v=ESDzYYl0__s).

Software libraries to replicate the remote's cursor-pointing capabilities

> The open development community has created software libraries for connecting to a Wii remote for nearly every major development platform.

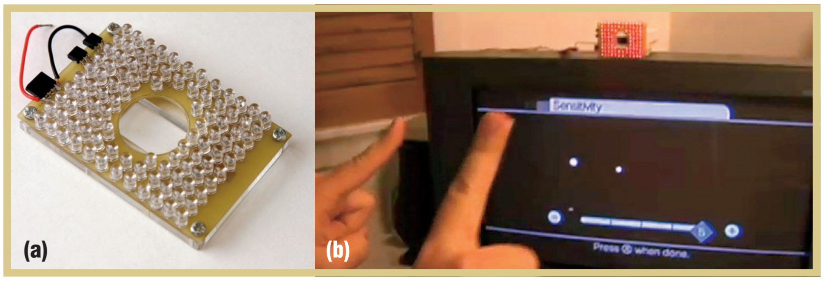


Figure 3. Finger tracking. (a) The Wii remote's IR LED array illuminator. (b) A reflector tag increases visibility for tracking.

for controlling mouse input were also among the early projects. As a result, some people began using the device for media navigation or to play mouse-based PC games. However, these uses have been somewhat limited because the Wii remote requires an IR sensor bar to enable pointer tracking.

My work has so far focused on how custom IR emitters can extend the usefulness of the controller's IR camera beyond merely distant pointer tracking. When you hold the Wii remote in your hand, the camera sees the IR dot movements primarily in correspondence to the controller's yaw, pitch, and roll. The tracking data is relatively insensitive to translational movement.

However, when the remote is stationary and the IR emitters move, this property is reversed. Dot movement corresponds primarily to translation, and the tracking data is relatively insensitive to orientation. This is the arrangement that motion-capture systems typically use. Thus, using the remote in this manner transforms it into a relatively high-performing, commodity motion-tracking system.

The rest of this section explores project applications of this configuration. Video demonstrations of all these applications are available on my projects Web site at http://johnnylee.net. (For a useful overview of general tracking technologies, techniques, and issues, see B. Danette Allen and her colleagues' Siggraph course notes.)[3]

Finger and object tracking

Because the Wii remote camera is sensitive only to bright sources of IR light, tracked objects must emit a significant amount of near-IR light to be detected—for example, an IR LED. However, instrumenting surfaces or objects with active LED emitters can be mechanically prohibitive or undesirable due to battery weight and size constraints.

Hands and fingers are good examples of surfaces that benefit from minimizing tracking instrumentation. Camera-based motion-capture systems often employ a technique that uses specialized markers to increase the visibility of tracked points. The systems can further increase visibility by using retroreflective tags and colocating specialized light sources with the tracking camera rather than the tracked point. Vicon motion-capture systems use this approach (www.vicon.com).

Figure 3 shows the Wii remote LED array and the use of reflective tags to track fingers. This approach provides simple, reliable tracking of multiple objects. It could work without the reflective tags, but the tracking data can be noisy and the working volume is small and adjacent to the front of the Wii remote.

You can trigger events by curling and extending the finger to make points appear and disappear. The difficulty of hitting screen targets without a persistent on-screen cursor poses a usability issue. One approach to resolving this is to have the software respond to a point's disappearance rather than its appearance. This would be similar to making graphical buttons respond when you release the mouse button events instead of when you press the mouse button. Alternatively, you could track thumb and forefinger pairs to provide an on-screen cursor and then trigger events by pinching them together.

Attaching the retroreflective markers to gloves or other wearable accessories can help create removable, highly reusable markers. The onset of fatigue is very rapid in mid-air hand manipulation, so this approach might be practical for only some application types or better used on more horizontal surfaces for productivity applications.

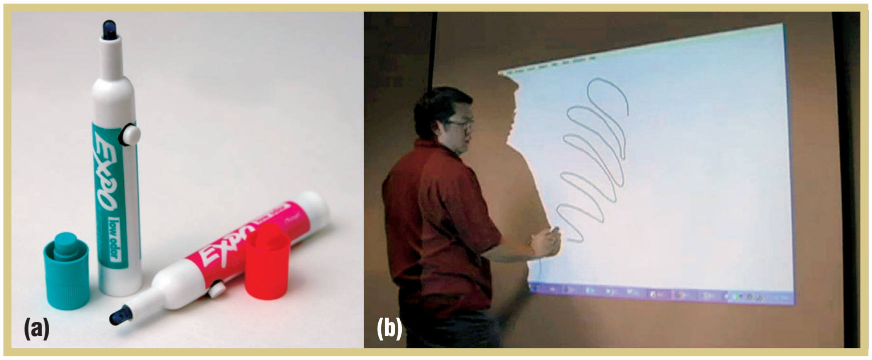
The technique can also be used to track arbitrary objects such as sporting equipment, physical input devices, or even animals. However, unintentional IR illuminator reflections can generate spurious tracking data, complicating retroreflecting marker tracking. Thus, if instrumentation of the object or surface is acceptable, then active LED markers will provide less tracking interference and tracking at longer distances.

Interactive whiteboards and tablet displays

By constraining the movement of IR emitters to a planar display surface, you can map the Wii remote camera's coordinate system to the display's coordinates. For example, if you point the camera at a projected image on a wall and then place an IR-emitting light pen on the surface, you can use the IR camera data to compute which display pixels correspond to the pen's location. This lets you interact with the projected image as if it were an interactive whiteboard system, as shown in Figure 4.

To discover the correspondence between the camera and projector coordinates, you use a four-point calibration process typical for any touch-screen system. First, you display four crosshairs at known locations in each corner of the projected display, then you activate the pen at each of these crosshair locations

Figure 4. Interactive display. (a) Infrared LED pens used as a stylus for (b) an interactive whiteboard.



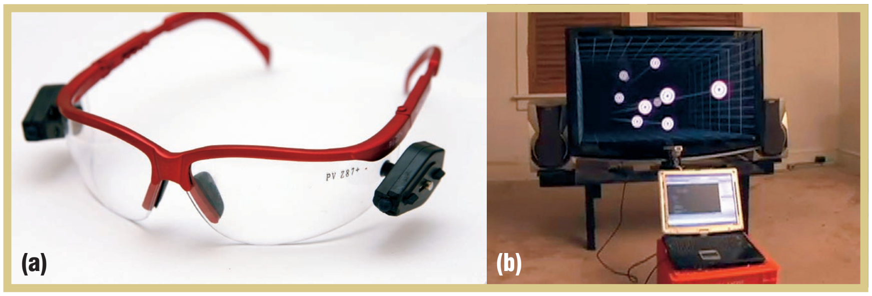


Figure 5. Desktop VR. (a) Rigid IR emitters on glasses together with (b) the Wii remote can render view-angle-dependent displays that simulate motion parallax and a changing field of view.

to register the corresponding camera coordinates. From these four registered points, you can compute a *homography*, a warping matrix for mapping any new point visible to the camera to the correct pixel location in the projected image.[4] This approach also works with any flat display surface, such as an LCD or plasma television. However, displays that have a thick glass surface can cause unwanted reflections that result in erratic tracking behavior.

The homography calculation is robust against display orientation and mirroring, so it supports a variety of camera-projector geometric relationships. Additionally, because the Wii remote can track up to four points, you can track multiple pens simultaneously, creating multitouch interactive surfaces.

The software that performs the four-point touch calibration and mouse emulation is available at my projects site, along with the video demonstration of this work. The software has been downloaded more than 500,000 times as of 1 March 2008. Several educators are already using it in their classrooms as a low-cost interactive whiteboard alternative for certain applications.

The approach's primary limitations are a maximum tracking resolution of 1,024 × 768 and the high sensitivity of tracking quality to camera position and occlusions. Thus, the Wii remote's placement is key to obtaining good performance. Overhead or off-to-the-side placement will reduce the likelihood of obstructions but also reduce tracking uniformity. If a rear-projected arrangement is possible, it provides ideal tracking performance. Multiple Wii remotes could also increase performance.

Head tracking for desktop VR displays

Two rigidly connected IR points provide the same tracking capabilities as the sensor bar: *x*, *y* coordinates, rotation, and estimated distance. If you place the Wii remote adjacent to the display in a known location and a set of wearable IR emitters on a user's head, you can track the head's location relative to the display and render view-angle-dependent views of a virtual environment. Figure 5 shows a system implementation that uses glasses with IR emitters. By responding to head movement, the display can simulate the behavior of a window providing motion parallax and a changing field of view, thus increasing the illusion of depth and realism.

Using the known physical separation of the IR emitters, you can estimate the head's distance from the screen. Similarly, using the display's known physical dimensions, you can calculate the remaining values of vertical and horizontal head displacement at the appropriate scale. Several game and data-visualization companies are already exploring the use of this technique in future products.

Because the software renders a custom viewpoint for the person wearing the IR glasses, the perspective will be incorrect for other observers. Some method of using a split screen or shutter-glass technology could support multiple users simultaneously, but implementing such an approach would depend on the display technology. The Wii remote's horizontal field of view might limit the range of movement to a smaller usable volume than desired for certain applications. However, multiple remotes could increase the field of view. Additionally, conflicting stereo depth cues from each eye can weaken the illusion. Combining head-tracking with polarized or shutter stereo-vision goggles could enhance the 3D experience. However, implementing stereo-vision techniques can be difficult, depending on the display technology.

Spatial augmented reality

You can augment the appearance of physical objects by using projected light to present colocated information on

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.