# Universal Device Access with FreeMote

Markus HIPP, Thorsten MAHLER, Christian SPIKA, Michael WEBER

Institute of Media Informatics, University of Ulm, Ulm, Germany

**Abstract.** With the FreeMote system, we present a solution for easy and intuitive universal device control. Based on the Wii Remote Controller real world device functions can be accessed and controlled via gestures. Gestures include predefined sets apt for standard functions present in many devices as well as user definable gestures. Offering both possibilities, the users are free to choose and adapt the gestures to their needs while reducing the mental load. Along with the gesture interface, FreeMote offers a simple way of enabling everyday devices for FreeMote controlling. FreeMote makes use of the spatial nature of real world environments in device selection and in function controlling via gestures. Thus, FreeMote system supports the user in finding enabled devices, makes selecting devices easy and provides an intuitive way of controlling devices via gestures.

**Keywords.** Wii Remote Controller. Gesture Control. Universal Remote. Tangible User Interfaces. Spatial Interaction.

## Introduction

Mark Weiser predicted a new computer era based on miniaturization and the rise of implicit computer interaction in his visionary article "The Computer for the 21st Century" [16]. He had an environment in mind where computers react to human actions in the real world. Making the computer disappear, he envisioned a world where computers are supportive and ever present with all information in every place. Clearly, we do not live in such a world yet, but the miniaturization and computerization of everyday objects goes on and can be witnessed in our daily surrounding.

The ever increasing number of devices that are computerized and can be controlled electronically also dramatically increase the interaction possibilities. A once simple device nowadays provides an increasing number of very special functionality, sometimes even clouding the basic devices purpose [12]. The vast number of functions is by no means the only problem: the diversity of the devices and the very different interaction realizations make the interfaces ever more confusing; it complicates the handling and makes accessing functions difficult. The sheer number of different interfaces means high mental load and asks for highly experienced users or induces restraints to few basic functions.

# 1. Universal Remote Challenges

Controlling a multitude of different devices, possibly every single device equipped with its unique remote control, is getting more and more complex. A solution offered is the use of a single standard universal remote control to handle every remotely operated device.

However, this solution often results in even more chaos for the user. Crammed with buttons and different interaction opportunities, the use of a standard universal remote control is rather complicating than relaxing the interaction.

To achieve a practical and comfortable solution that will master this situation, a system has to solve the following problems:

- **Universal Remote:** Because of the high amount of devices controllable remotely, the offered system must be powerful enough to handle every available device.
- **High Expressivity:** In order to handle a high number of device specific functions the system has to offer many degrees of freedom to support the identification and execution of every function needed.
- **Mental Load:** The device control must not overstrain the user but reduce the cognitive load.
- **Simplicity of Use:** As proposed by Chang et al. in [2], the addition of buttons to a remote control does not help usability, as it does not reduce mental load for the user. Instead, an interface must provide an interaction realization that is simple in design and pleases the user [9].
- **Non Invasive:** To increase the simplicity of use, the system needs to be usable fast and without preparation. A system that is non invasive but easy graspable and known increases the comfort of the user.
- **Device Selection:** The system has to provide a way to transparently inform the user of which devices can be controlled in an environment, how they can be selected and which device is activated at a time.
- **Personalization:** The users should be able to define their personal, easy to remember interaction sequences. These personal ways of interaction should be easily transferable to other devices or even environments.

## 2. The FreeMote Solution

To tackle the challenges mentioned above we devised an interface which easily integrates into everyday life, the FreeMote. It provides the user with a new intuitive but at the same time facile option to interact with everyday devices. The FreeMote consists of two pieces of hardware: Firstly, the FreeMote controller used as a remote control replacement as well as for transmitting the identification data and secondly the FreeMote sensorbar equipped with three LED lights, working as the device identifier.

## 2.1. Hardware for Gesture Recognition

In 2006 Nintendo released his latest game console, the Nintendo Wii. The most innovative feature is its new game pad, the Wii Remote Control (see figure 1). With its acceleration sensor it is able to detect movements in all three dimensions. Also, it is equipped with an IR camera, which is able to recognize up to four light sources. Due to this fact we are able to use different FreeMote sensorbars for different devices. For the wireless connection the Wii Remote Controller uses bluetooth, which makes it easily accessible by every bluetooth enabled PC [10].

Another important reason is the growing profile of the Nintendo Wii and its remote controller: With its few buttons and its clean design it is a non-invasive and very easy



Figure 1. The Wii Remote Controller is used as uni-Figure 2. Early Prototype of the identification sysversal gesture input device. LEDs are enabled for frequency coding.

to handle remote control understandable for everyone. In addition, the Wii Remote Controller is highly available compared to other, often prototypical comparative products. From a application development point of view, the Wii Remote Controller is a good choice because it can easily be integrated into a software system with the help of many already existing frameworks in different computer languages. Considering these facts, we decided against assembling another accelerator enabled nicely designed artefact but to use the standard Wii Remote Controller to be our universal remote hand set.

## 2.2. Hardware for Device Identification

In order to identify the Wii Remote Controller's position, Nintendo uses a sensorbar usually located on top of the display to be controlled. The sensorbar consists of two infrared LED light sources placed at the right and left end of the roughly ten inch wide bar. Combined with the IR camera of the Wii Remote Controller it is possible to compute its position, usually used to control the mouse pointer on the screen.

FreeMote utilizes the capability of the Wii Remote Controller to even recognize up to four light sources. Thus it is possible to expand the sensorbar with one more LED light source: Basically, the FreeMote sensorbar is composed of three equispaced light sources lined up in a row. While two LEDs are needed for defining the gesture's shape recognized by the FreeMote system, the third light is blinking in a defined frequency. Having a FreeMote sensorbar on every controllable device it is possible to identify the devices due to the individual blinking frequency of their sensor bar. Figure 2 shows an early prototype of this identification system.

## 3. Implementation

The FreeMote system front end consists of the FreeMote Controller and the three LED FreeMote sensorbar for creating input data. The software application on the back end handling the input data is implemented in JAVA. It extends the WiiGee library [15] and runs on a standard PC platform. The only further requirement is a bluetooth stack (we tested the system with the WIDCOMM and Avetana stack) to connect to the FreeMote Controller.

## 3.1. Gesture Recognition

Due to the unhandiness of most common universal remote controls we decided to create a new innovative interface to handle every device. Based on the Nintendo Wii Remote Controller our system uses only three buttons in combination with gesture recognition for controlling devices. The FreeMote controller has a built-in accelerometer with the ability to sense acceleration along three axes. Connected over bluetooth, the accelerometer provides the input system with data consisting of vectors representing the acceleration of the controller in all three dimensions. We decided to base the input system on the WiiGee library [15]. This implementation is freely available and allows a stable gesture recognition which could be proven in user studies where Schlömer et al. reached an average of 90% correctly recognized gestures. The Freemote system offers to interact with a device by gesture control. To reduce the mental effort but at the same time preserve the freedom of use for the user, we divide gesture control into two different types of gestures:

- **Standard Gestures:** In order to reduce the cognitive load for the user, we equipped FreeMote with a variety of standard gestures used for the most common options of nowadays devices, e.g. on and off, louder and softer, darker and lighter. By using a set of standard gestures for every device the user is only forced to learn these once and thereby the mental load of common universal remotes (e.g. searching the right button for a specific device) is prevented. For instance a fast left to right movement is used to switch to the next channel for a TV set or the next song for the hifi set.
- **User Defined Gestures:** Besides the advantages of standard gestures we still give the user the freedom of choice to create and use his own set of gestures for one or even all devices and functions. The user is free to create any gesture that is possible in 3D space and bind it to a function of a specific device. This enables him to form a personal controlling of his devices and as a result, confers FreeMote individuality. For instance, in preliminary tests one user defined a cross movement to stop the hift playback.

#### 3.2. Use of Gestures

In order to start the gesture recognition, a button on the FreeMote controller is need to be held down. This marks the beginning of the now following motion sequence that describes a gesture. It is now possible to execute one of the provided standard gestures or one of gestures that were created by the user. To end the gesture recognition the button is released. At that point, the FreeMote system automatically analyzes the performed gesture, compares it with the stored gestures and, if a match has been detected, executes the associated function of the device.

To activate the training mode for creating new gestures the user needs to press and hold down a different button on the controller and is now able to create a motion sequence which will be saved as a new gesture. By releasing the key followed by a quick press of a button that marks the end of the whole gesture training, the gesture will be stored together with the related function of this device. In order to improve recognition rate and smoothness of a gesture it is advised to iterate the training sequence three four times.

As a result of only one device being activated at a time, FreeMote does not need miscellaneous gestures for several devices. Instead, every device can be controlled by the same set of gestures. This again reduces the cognitive load for the user and simplifies the use and control of every device. See figure 3 for a typical gesture. For storing and

retrieving of gestures we use an XML structure, since it is easy to integrate and nowadays used as the standard for modelling hierarchically structured data. It also allows the device associated gestures to be portable on every FreeMote interface available.

## 3.3. Device Identification

Virtually, making the PC disappear, the interaction takes place between the user and the specific device or artefact. The user has the feeling of interacting with a controllable device instead of a PC, while technically a PC in the background translates the users actions into device functions.

The process of detecting controllable devices is divided in two steps: The first step is to detect the shape of a FreeMote sensorbar. Therefore we implemented a detection algorithm, which analyzes the light sources seen by the IR camera in order to detect the shape of the FreeMote sensorbar on the one hand and to filter lightsources caused by reflection or insolation on the other hand. On the basis of a probability function the software decides if a FreeMote sensorbar is detected or not.

If a FreeMote sensorbar is detected, the software proceeds with the analysis of the frequency of the third light. Computer vision algorithms are used to steady the detected image. This image is analyzed to identify the frequency, which can be determined even if the user is moving the FreeMote controller. With this simple approach the whole process of device detection and identification takes about one second and is confirmed to the user by vibration in case of success.

We also take advantage of the detection algorithm in another context. Without knowing which devices in a room are actually controllable by FreeMote the user can move the Freemote controller around in the room resulting in a short vibration whenever the system is detecting a FreeMote sensorbar. This very helpful feature of FreeMote makes it easy for the user to find out which devices are controllable in general.

To add a new device to the system, the user starts the hardware wizard. The FreeMote core thereupon checks the XML device description for available free frequencies and prompts the user to manually adjust the FreeMote sensorbar frequency. Depending on the chosen hardware template, the new device is added to the XML device file and therefor available, providing a standard set of gestures.

## 4. System Walk Through

In a typical scenario a user wants to activate one of many devices present in his living room, for instance. First of all, the corresponding device needs to be focused with the FreeMote controller in order to identify and thereby activate the device. Hereby, the device can be a complex device like a TV set or a hi-fi system as well as such devices as speakers or lights. The only necessary requirement is its tagging by a FreeMote sensorbar (see figure 4). After identification the user has full control over the functions offered by the device. In our particular scenario the set of controllable devices consisted of a computer controlled TV, a computer controlled hi-fi system and an ethernet accessible lamp with dimming function. We used the same gestures for similar functions on different devices, for example 'next channel' on the tv and 'next track' on the hi-fi system. Nevertheless, the user is free to define gestures to control the functions himself. We are also able to switch between devices by identifying them with aid of the combination of



Figure 3. Executing a user defined gesture control-Figure 4. One of the speakers connected to the hiling a device (stopping music playback in this case). fi-system which has been enabled for identification via the portable Freemote sensorbar.

the IR camera of the Wii Remote Controller and the FreeMote sensorbars due to the different frequencies of the pulsing LED.

In summary, we created a powerful software tool to support the user in his familiar device-handling. Above all, the Wii Remote Controller with its intuitive handling and its simplicity turned out to be a very good alternative to all existing universal remote controls.

# 5. Related Work

Considerable research has been done regarding spatial interaction and remote control, ranging from gesture recognition to systems for free device controlling. Freeman et al. for instance set up a system to control a TV set via free hand interaction. A camera tracks the hand position which is used as pointing into a graphical interface display on the TV. This way, they are able to attain a large number of functions while keeping the interaction simple and comprehensive [4]. While this approach clearly solves the controlling of home devices in a non invasive way, it heavily depends on a well equipped environment especially when it comes to controlling multiple devices.

While still using hand recognition, a different approach is taken by Kölsch et al. They use a more sophisticated way to track hands in order to recognize even hand poses which increases the gesture interaction possibilities dramatically [6]. Still using a camera to track the hand movement, they offer new possibilities by implementing the camera together with the tracking into a Head Mounted Unit (HMU). This way they are not bound to a specially equipped room, but are free to move [7]. Kölsch et al. also propose to use a graphical interface superimposed on a camera image via a HMU. Nevertheless, the system itself is very invasive contradicting one of the basic ideas of ubiquitous computing.

A system that is not dependent on cameras but uses a special glove to track hand movements is the Glove-TalkII system by [3] et al.. They implemented a gesture driven interface to translate hand movements into speech. By using the highly developed glove they can rely on high expressivity but they also stated that the interface itself needs training and experienced users to be used. In our opinion [3] et al. presented an approach that would be feasible for the gesture part of our system but itself is highly invasive.

A system that tackles the simplicity of use and focuses on the joy of use is the WorldBeat system presented by Borchers et al. [1]. Here, batons are used as devices for music creation. The batons were simply equipped with infrared lights enabling them to act as pointing devices. This simple but highly appreciated interface enabled the user for 3D interaction. While this exhibit is clearly a no gesture device, it impresses by it's simple usage and non invasiveness.

Ouchi et al. translated this approach to home appliances. In their MagicWand system, they already combine simple gestures with baton interaction [13]. However, this system lacks a proper way to identify devices. They use only a very simple and fixed gesture set, neither allowing for a high number of functions nor for user defined gestures.

The importance of both standard gesture sets and user defined interaction gestures are verified by Kela et al.. In [5] they present an evaluation and comparative study of accelerometer-based gesture controls in mobile interaction and different application scenarios. Users responded very well to the naturalness of such interaction.

Other studies indicate the feasibility of gesture recognition for application input. For a stereo camera environment Lichtenauer et al. validated a 3D hand-recognition system in a user study with 70 persons using a set of 120 3D-signs and could show very promising results for gesture recognition [8]. Similar systems are proposed based on a 3D-accelerometer and on the same algorithms by Pylvänäinen in [14].

In recent research papers, these findings are implemented in libraries for instance the freely available WiiGee library [15]. Schlömer et al. present a accelerator based gesture library that is capable to recognize trained gestures. It works with the widely available Wii Remote Controller, the device we also favor for our system. With its initial user evaluation and its promising results it builds a fast and solid foundation for our FreeMote gesture subsytem.

# 6. Conclusion and Future Work

With FreeMote we presented a universal remote control system. Based on the Wii Remote Controller we offer an interface that is both easy to use and well accepted by the users. Using the capabilities of the Wii Remote Controller integrated device we are able to recognize real world devices and to control them via 3D-gestures. While the identification of different devices can focus the number of possible interaction down on the device, the degrees of freedom of the integrated accelerometers make way for controlling a huge number of different functions.

By pointing the FreeMote on an enabled device like a lamp, a TV set, a hifi-system or even only a speaker, the respective device can be controlled. Thus the interaction is virtually reduced to an interaction between the actual device and the user. Therefore the set of functions can be further reduced making the interface easier to be handled. Furthermore, the interaction is dissolved from one single appliance like, for instance, the light switch for all lamps in the room. Instead the spatial awareness enables the targeting of exactly one lamp. It makes way for higher granularity while concurrently easing the identification and reducing function clutter. The use of one device makes it easy to use the same interaction pattern for similar functions on different devices, thus reducing mental load rendering the multitude of different remotes with different interfaces obsolete.

The FreeMote turns an equipped room itself together with the devices in it to an interface making use of its spatial nature. The interaction virtually takes place between the user and the devices themselves being controlled directly.

The gestures used to control functions of devices can be freely defined by the user. Nevertheless, we found it useful to provide a set of standard gestures that represent often needed functionality. These standard gestures simplify interaction by reducing the number of overall gestures with the goal to lower the mental load.

In the future we plan to work on making the standard gesture set even more comprehensible for the users. This includes to implement the system in an intelligent living or meeting room where the system can be tested and evaluated over a long time under real life conditions. Furthermore, in the current form the identification of the devices is limited to a relatively small number of devices. To be used in large rooms or buildings their number can be quite high however. Here, coding information is a suitable way to reach this goal.

All in all, we could present a universal interface suitable as alternative for the everyday collection of remote controls that is easy to use and comprehensible.

## Acknowledgements

The authors are thankful to the students taking part in programming and completing the FreeMote system, Frank Wahnschaffe and Kevin O'Brien.

### References

- [1] Borchers, Jan O., WorldBeat: designing a baton-based interface for an interactive music exhibit, *Proceedings of the SIGCHI conference on Human factors in computing systems*, (1997), 131–138.
- [2] Chang, Angela and Gouldstone, James and Zigelbaum, Jamie and Ishii, Hiroshi, Simplicity in interaction design, TEI '07: Proceedings of the 1st international conference on Tangible and embedded interaction, (2007),135–138.
- [3] Fels, Sidney and Hinton, Geoffrey, Glove-TalkII: an adaptive gesture-to-formant interface, Proceedings of the SIGCHI conference on Human factors in computing systems, (1995), 456–463.
- [4] Freeman, W. T. and C. Weissman, Television control by hand gestures, *Intl. Workshop on automatic face- and gesture-recognition*, (1995), 179–183.
- [5] Kela, J., Korpipää, P., Mäntyjärvi, J., Kallio, S., Savino, G., Jozzo, L., and Marca, D., Accelerometerbased gesture control for a design environment, *Personal Ubiquitous Computing* 10, 5, (2006), 285–299.
- [6] Kölsch, M. and Turk, M. and Hollerer, T., Fast 2D Hand Tracking with Flocks of Features and Multi-Cue Integration, *Conference on Computer Vision and Pattern Recognition Workshop*, (2004), 158–158.
- [7] Kölsch, M. and Turk, M. and Hollerer, T., Vision-based interfaces for mobility, *International Conference on Mobile and Ubiquitous Systems: Networking and Services*, (2004), 86–94.
- [8] Lichtenauer, J.F. and ten Holt, G.A. and Hendriks, E.A. and Reinders, M.J.T., Sign language detection using 3D visual cues, *IEEE Conference on Advanced Video and Signal Based Surveillance*, (2007), 435–440.
- [9] Maeda, John, *The Laws of Simplicity (Simplicity: Design, Technology, Business, Life)*, The MIT Press, Cambridge, MA, 2006.
- [10] Wii at Nintendo, Nintendo, http://www.nintendo.com/wii, March, 2009.
- [11] Norman, Donald A., The Psychology of Everyday Things, Basic Books, New York, NY, 1988.
- [12] Norman, Donald A., Simplicity is highly overrated, interactions, 14, 2, (2007), 40-41.
- [13] Ouchi, K. and Esaka, N. and Tamura, Y. and Hirahara, M. and Doi, M., Magic Wand: an intuitive gesture remote control for home appliances, *Proceedings of the 2005 International Conference on Active Media Technology*, (2005), 274.
- [14] Timo Pylvänäinen, Accelerometer Based Gesture Recognition Using Continuous HMMs, Pattern Recognition and Image Analysis, (2005), 639–646.
- [15] Schlömer, Thomas and Poppinga, Benjamin and Henze, Niels and Boll, Susanne, Gesture recognition with a Wii controller, *TEI '08: Proceedings of the 2nd international conference on Tangible and embedded interaction*, (2008), 11–14.
- [16] Mark Weiser, The Computer for the 21st Century, Scientific American, 265, (1991).