

A SOFTWARE TOOL FOR EXPERIMENTAL STUDY LEAP MOTION

Georgi Krastev¹ and Magdalena Andreeva²

¹Department of Computer Systems and Technologies, Ruse University, Ruse, Bulgaria

²Department of Computer Science, Ruse University, Ruse, Bulgaria

ABSTRACT

The paper aims to present computer application that illustrates Leap Motion controller's abilities. It is a peripheral and software for PC, which enables control by natural user interface based on gestures. The publication also describes how the controller works and its main advantages/disadvantages. Some apps using leap motion controller are discussed.

KEYWORDS

Human computer interaction, Leap motion, Software tool, Virtual working space, gesture control.

1. INTRODUCTION

The computer devices are coming in strong. The increase of sale and usage of mobile, wireless and compact devices proves the entering upon a new age. The improvement in the display technologies, the graphical chipset, even in battery life causes the constantly increase of the information flow from the machines to the human. Although slowly there is a progress in the opposite communication – from the human to the computer. The human computer interaction is improved as a result of the microelectronics, sensors and computing power progress.

The human computer interface (HCI) defines the way people and machines interacts each other [1, 2]. The variety of the human behaviour is taken into account during the HCI design. Thus it is achieved easy and intuitional work with a computer device, as well as natural communication with it. Because of this the HCI is a multidisciplinary problem, which combines the practices of the computer science and the principles of the psychology, sociology and anthropology, together with approaches form the industrial design of interactive products.

Leap Motion is a peripheral and software for PC, which enables control by natural user interface based on gestures.

This paper focuses on developing software to study PC control by natural user interface based on gestures using Leap Motion.

2. APPS FOR LEAP MOTION

The leap motion controller is not very popular yet, but there are apps for it. Some of the best are discussed in [7]. Most of these apps are in the game category and education.

2.1. Molecules

“Molecules” [8] is a Mac-only app that lets you view and manipulate 3D rendering of molecules (fig. 1).



Figure 1. The usage of “Molecule” app

2.2. Cyber Science – Motion

Cyber Science [9] gives anatomy knowledge in 3D with Motion 3D, which allows you to explore, dissect and reassemble a human skull and learning the names and the location all 28 bones (fig.2).



Figure 2. The usage of “Cyber Science” app

2.3. Exoplanet

Mac-only Exoplanet [10] puts the Milky Way in your hands, letting you explore within our own solar system or explore the extensive catalogue of all known planets, a growing list that is updated as new discoveries are confirmed (fig.3).



Figure 3. The usage of “Exoplanet” app

2.4. Boom Ball

Remix the classic game Brick Breaker with racquetball, add explosions, and mix thoroughly, and you'll get Boom Ball [11] (fig. 4).



Figure 4. The usage of “Frog Dissection” app

2.5. Corel Painter Freestyle

Corel Painter Freestyle [12] combines the Leap Motion Controller with a full-featured paint program, letting you create unique artworks by painting in mid-air using your finger painting or detailed creation with pen or pencil (fig.5).

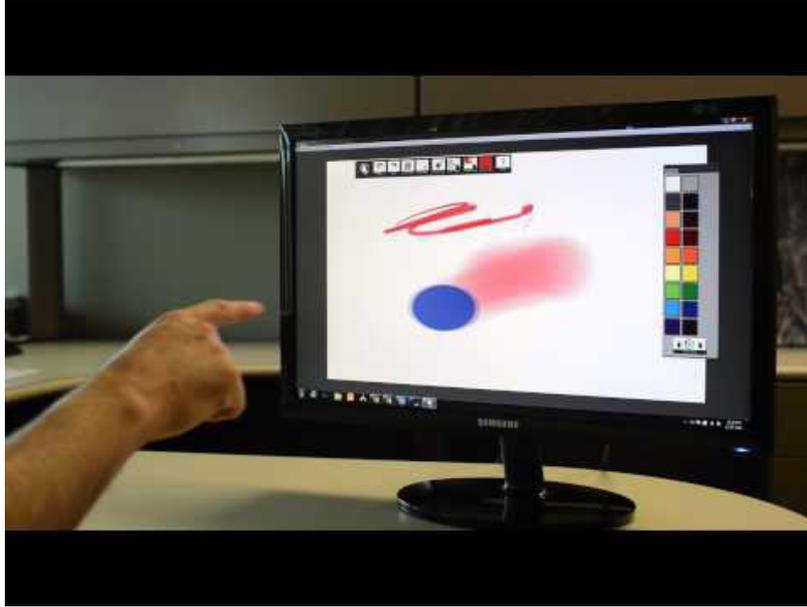


Figure 5. The usage of “Corel Painter Freestyle” app

2.6. NYTimes for Leap Motion

The New York Times app [13] lets you wave your way through the top stories of the day, navigating from one story to the next and scrolling within each article with the ease of gestures (fig. 6).

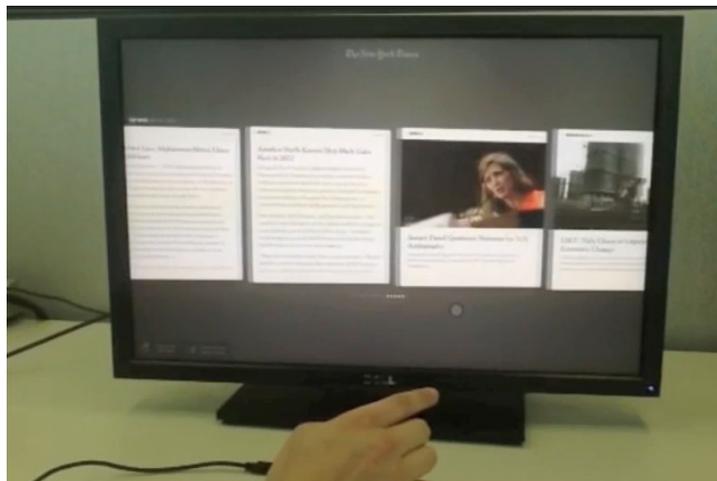


Figure 6. The usage of “NY Times” app

2.7. Google Earth

One of the famous geographic information systems Google Earth has the option of using the Leap Motion Controller for gesture-based navigation [14] (fig. 7).



Figure 7. The usage of “NY Times” app

3. THE WORKING PRINCIPLES OF LEAP MOTION

The software of this device analyses the objects, which are in its virtual space. It recognizes hand, fingers and tools and stores the information about the position, gesture and its motions. The virtual working space is in the form of inverted pyramid, situated in the center of the device. The operation range is approximately from 25 to 600 mm above device.

The device works with Cartesian coordinate system (right oriented). The coordinates are in mm and the beginning of the system is the middle of the device (fig. 8) [3]. These coordinates must be converted to screen coordinates depending on the screen size or window size of the application using the formulas (1):

$$\begin{aligned} \text{screenX} &= ((\text{leapX} * \text{WindowWidth} * 2) / \text{WindowWidth}) + (\text{WindowWidth} / 2); \\ \text{screenY} &= \text{WindowHeight} - \text{leapY}; \end{aligned} \quad (1)$$

where *WindowWidth* and *WindowHeight* are the width and height of the application window; *leapX* and *leapY* – the position of the element (for example the finger) on the X and Y axes in the virtual working space; *screenX* and *screenY* – the screen coordinates.

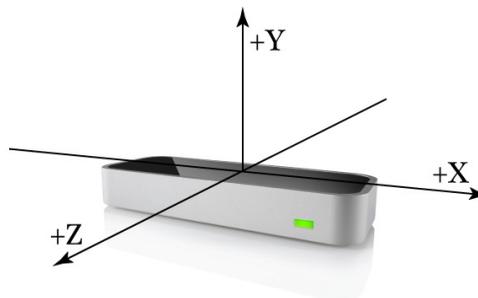


Figure 8. Frame of reference

The gestures covered by controller's API functions are:

- Circle Gesture, fig 9;

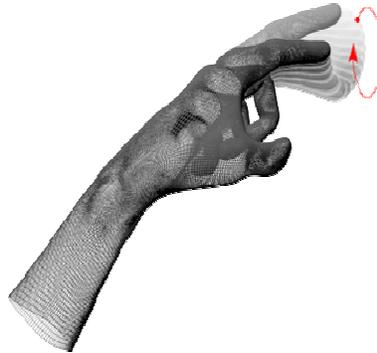


Figure 9. Circle Gesture with Index Finger

- Swipe Gesture, fig. 10 – linearly hand motion;



Figure 10. Swipe Gesture with Index Finger

- Key Tap Gesture, fig. 11 – imitating the movement key press;



Figure 11. Key Tap Gesture with Index Finger

- Screen Tap Gesture, fig. 12 – imitating the movement screen touch (vertical screen).

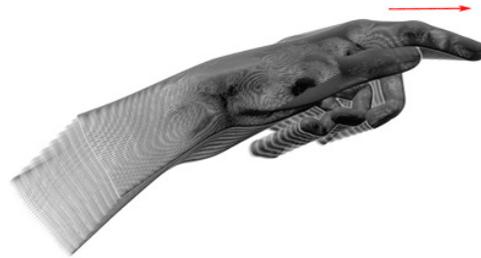


Figure 12. Screen Tap Gesture with Index Finger

Before use the programmer must activate the supported by the controller gestures.

4. DEVELOPING THE SOFTWARE TOOL

Leap motion supports the following programming languages: C++, Objective-C, C#, Java, Python, JavaScript. Other languages can be used by data binding. The supported operating systems are: OS X 10.6+ и Windows 7+. An experimental version for Linux exists, but only for developers.

The proposed software for Leap Motion is developed in Microsoft – XNA 4.0 using C# language. In addition to track hands, fingers and tools in its virtual space Leap Motion checks for new data and updates them in the frame structure. Each frame contains list for hands, fingers and tools data, as well as data for the recognized gestures and attributes describing the motion as a whole. The received and processed by the controller information is presented on fig. 13:

- Frame id;
- Hands – the number of the recognized hands;
- Fingers – the number of the recognized fingers (the white squares);
- Tools – the number of the recognized tools;
- Gestures – the number of the recognized gestures;
- Hand has X fingers – the number of the recognized fingers;
- Average finger tip position;
- Hand sphere radius – we assume that the hand is holding virtual sphere and its radius is calculated;
- Palm position – visualized as red square;
- Hand pitch – the degrees which the hand is inclined;
- Roll the degrees which the hand is rolled;
- Yaw – deviation.

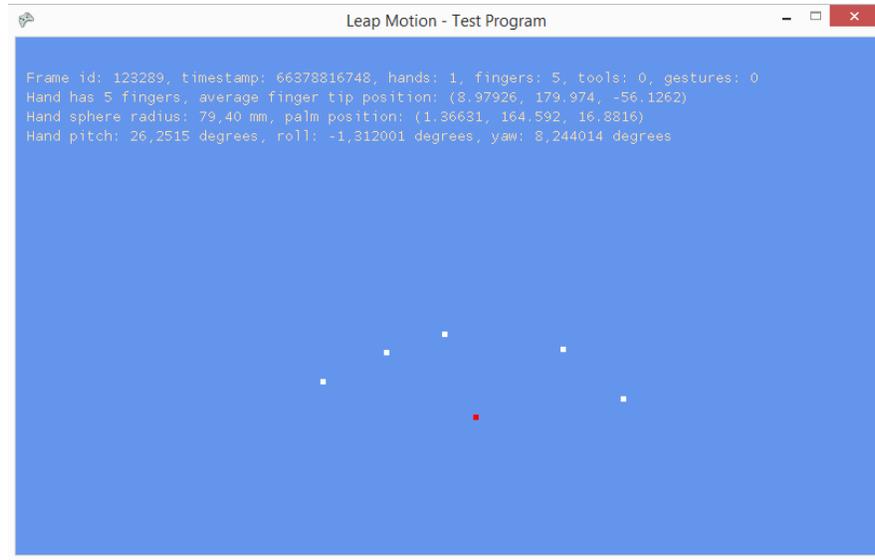


Figure 13. The frame status – one hand with five fingers

When one of the gestures is recognized the background changes to dark blue (fig.14) and the following information about the gesture is displayed:

- Gesture ID – the gesture Swipe;
- Gesture state – STATESTART, STATESTOP, STATEUPDATE;
- Position – current position;
- Direction – motion direction;
- Speed – motion speed.

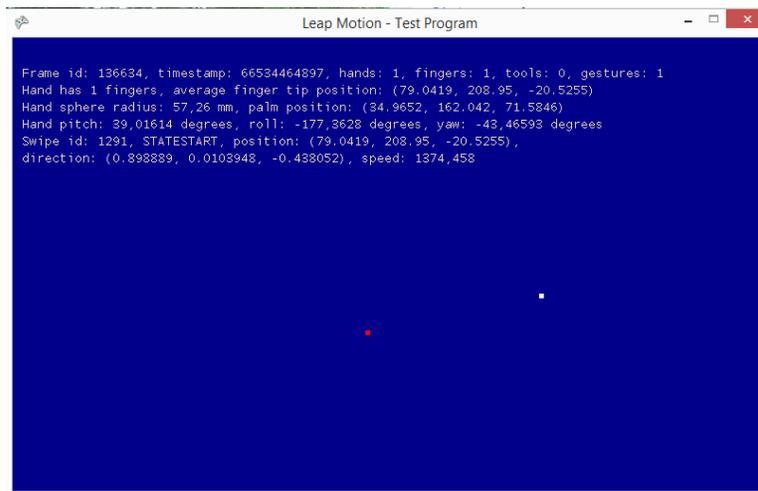


Figure 14. The frame status – one hand with five fingers and the gesture Swipe

The developed tool can be used for experimental study the functionality of leap motion.

5. DISADVANTAGES OF LEAP MOTION

API of the controller determines and saves as much as it can information. Nevertheless Leap motion may not define all the attributes in each frame and that causes the following disadvantages:

- When the hand is clenched the fingers are invisible for the Leap Motion controller;
- The controller can not distinguish between left and right hand but allows working with more than two hands and/or tools associated with hands. This leads to decline optimum quality of motion tracking.
- Does not work well in the presence of strong light (hardware problem).
- Does not recognize well reflected and transparent surfaces (hardware problem).
- The dimension of the working space limits the recognition of the distant finger motions (hardware problem).

Classification of the fingers and the tools is according their shape. The tools are longer, thinner and straighter than the fingers. Therefore when working with the device is not a problem to wear gloves, watch, rings etc., though sometimes this leads to deterioration of recognition (for example, if you wear thick gloves). On the other hand the device can easy be “lied”. Some times the device recognizes pen, pencil even chain as a finger. This happens because the API functions of the Leap motion analyze the length of the visible part of the object, the average width of the object, the direction of the pointing object, the position of the tip of the object, the speed of movement of the tip of the object and on this basis decides whether this object is a finger or tool.

6. CONCLUSIONS

The idea for the study of Leap motion sensor came because of the faster widening of its field of application. There are a lot of applications using this controller for many devices as robot-helicopter, applications for 3D modeling, education apps and PC games.

Despite the disadvantages of the device its sensors detects the hand movements and each finger with great precision real-time. The results demonstrate that the proposed ideology not only ensure the same level of interaction, both in handling keyboard and mouse, but that it can be adapted to work with different applications.

REFERENCES

- [1] Turk, M. RTV4HCI: A Historical Overview. In B. Kisanin, V. Pavlovic, and T. Huang (eds.), Real-Time Vision for Human-Computer Interaction, Springer, 2005. Springer, 2005, pp 3-13
- [2] <https://developer.leapmotion.com>
- [3] https://raw.githubusercontent.com/voidplus/leap-motionprocessing/master/reference/leap_gestures.jpg
- [4] <https://sites.google.com/site/chuongnguyen/blog/workingprincipleofleapmotiondevice/LeapPrinciple.png>
- [5] <http://redac.socialinput.fr/wp-content/uploads/2013/04/leap-motion.jpg>
- [6] https://developer.leapmotion.com/documentation/images/Leap_Axes.png
- [7] [http://www.pcmag.com/article2/0,2817,2422165,00.asp-10 Best Apps for the Leap Motion Controller, B. Westover, 2013](http://www.pcmag.com/article2/0,2817,2422165,00.asp-10%20Best%20Apps%20for%20the%20Leap%20Motion%20Controller,%20B.%20Westover,%202013)
- [8] <https://apps.leapmotion.com/apps/molecules/osx>
- [9] <https://apps.leapmotion.com/apps/cyber-science-motion/windows>
- [10] <https://apps.leapmotion.com/apps/exoplanet/osx>
- [11] <https://apps.leapmotion.com/apps/boom-ball/windows>
- [12] <http://www.digitalartsonline.co.uk/news/illustration/corel-painter-freestyle-offers-kinect-style-painting-by-making-gestures-in-air/>
- [13] <https://apps.leapmotion.com/apps/nytimes-for-leap-motion/windows>
- [14] <https://apps.leapmotion.com/apps/google-earth/weblink>