Karolina Nowak, Adrian Smagur
Lodz University of Technology, Faculty of Technical Physics, Information Technology and Applied Mathematics
Wólczańska 215, 90-924 Łódź, karolina.nowak@dokt.p.lodz.pl, adrian.smagur@gmail.com

HAND GESTURES PROVIDED BY LEAP MOTION FOR USER INTERFACE OF INTERACTIVE GALLERY

Abstract
In the digital age, more and more artists are starting to present their works in an interactive way. Due to the rapid progression of the virtual environment and its attractiveness, it appears that new explorations of intuitive and user-friendly galleries are required. We created a draft of a computer application interface and then prepared a special gallery driven by different hand motions performed in mid-air. We conducted research of our proposed solution to compare these gestures and to find out how it is being perceived by people from outside the IT industry.

Key words
interactive gallery, user interface, hand gestures, Leap Motion, Processing

Introduction

Presenting artistic works as an interactive gallery is getting more and more popular since it is more easily accessible to the masses. People can examine all creations anywhere using only their notebooks or smartphones. Many experiments with design and running this type of application were carried out. Normally, all digital images are displayed in a flat 2D image space. The 3D technique allows users to interact and walk through a virtual environment [1]. This idea can be as innovative as the proposition of changing the application controller. Eye, hand and body movement and ideas to measure gestures became popular decades before, but only the last ten years brought major breakthroughs that have since become widely known [2]. However, for the present, there are only a few galleries driven this way. We decided to prepare a special application controlled by hand gestures to increase its accessibility. We wanted to verify the presupposition a computer can sense finger movements, including the computer image zoom, rotate, move, instruction, precise control, and other empty writing, to one hundredth of a millimeter [3].

Related work

Kauri et al. [4] studied various techniques of hand gesture recognition. Firstly, they classified static and dynamic gestures specified as hand postures and gestures. They are similar as they can be recognized by shape, orientation and finger flex angle. The most significant difference is ability to identify motion with its speed and direction. As we planned to design the interactive gallery navigated by hand, we needed to use dynamic gestures.

Authors of that article [4] proposed a classification of hand gesture recognition techniques. There are three methods based on sensors, vision and depth. The first one draws on data from sensors attached to the hand. It can be put on the skin directly or mounted on gloves. The second method uses cameras which capture a series of images. The user does not wear anything as all processes and analyses are conducted on photographs. The last one also embraces cameras, but the main idea was to capture a sense of depth. However, the Kinect device launched by Microsoft uses RGB images where the depth of objects can be found. We decided to work with a vision based technique as it is very natural and intuitive for users. They need to move their hand without any additional sensors and this motion is well-recognized. We chose Leap Motion for our application because it is a very popular and easy to use input device these days.

Pambudi et al. [5] created a desktop based game application with Leap Motion technology. They proved that using Leap Motion as a software developer is not complicated and problematic. They defined the grab gesture by determining the Euclidean distance formula between the node of the index finger and the node of the thumb. If this value is smaller than the one specified, it means that user makes a grab gesture. Put gesture recognition is based on the same formula, but value must be bigger than the referential one. Authors also studied the influence of light intensity on working of a Leap Motion device. They found out that light with more than 45lx intensity can produce such noises that a calibration process is not able to redub them.

Huang et al. [6] used a Leap Motion device in a music creation application. They proposed a simple method for determining object selection from a hand gesture. Each active element is considered as a single, circle framed object with attributes such as coordinate position and object radius. The pointer (denotive hand position) has its coordinates and radius too. The application checks if these coordinates are within the overlapping range. To this end, it computes the distance between the object and hand marker. If the value is smaller than the sum of the element and the pointer's radiuses, they influence each other. The result is saved as a special variable accepting only Boolean values (true and false).

Sanders et al. [7] prepared a special interface to study the potential of virtual engineering laboratories and tested it with Oculus Rift and a Leap Motion device stuck on goggles. They proposed three sizes of buttons that are supposed to be selected. The side of each square equals 5 cm, 2.5 cm and 1.5 cm sequentially. The biggest one is an equivalent of size of 2-3 fingers held together while the smaller one is the size of a standard keyboard key. Authors also tested two ways to compose these buttons: with and without space left between objects. Research showed that users were the most accurate during selecting the biggest button. Denser configuration of elements also impaired the results.

Materials and methods

A. Project plan

At the beginning, we planned our application interface as it is the most major part of the gallery. If users do not know how to use it, they will not become acquainted with artistic works. The application has a few page content categories: title, some information about the author, a list of creations and the presentation of one of the artist's works. We suspected that navigation by hand may be quite problematic for beginners, so we made bigger buttons. The design is simple and all active elements are well-known to computer users (arrows for the next/previous page and X as exit). Fig. 1 presents sketches with the composition of interface elements, including computed dimensions.

B. Implementation

Our interactive gallery is written in Processing language as it is an open source tool for coding within the context of the visual arts. The interface sketches were our reference. We created a very similar solution (Fig. 1), but we decided to hide the left arrow on the first page until the user sees the entire gallery.

We also added a special hover effects for all active buttons. The next/previous buttons change their color to the opposite one – white arrow becomes black while black background alters into white. The exit button in presentation of one artistic work had to get different effect as changing colors made it to be too similar to the disabled parts in backdrop. Now it is getting darker slowly when user points it. The position of his/her forefinger in application is marked as small red circle.

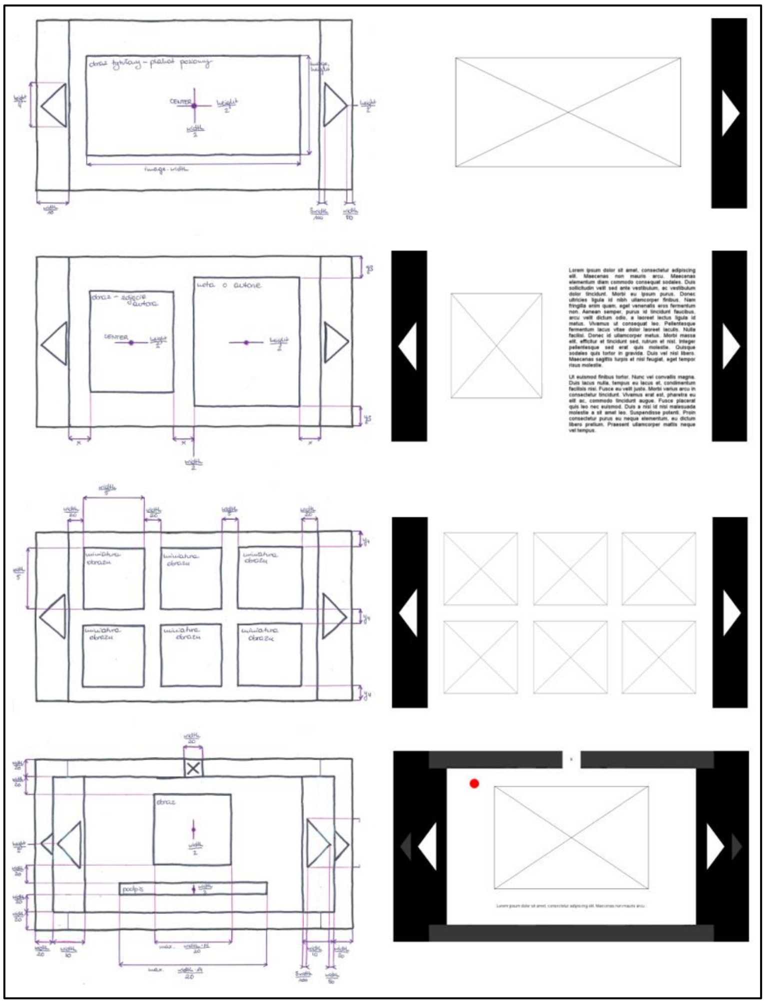

Fig 1. Sketches of gallery interface with dimensions and point (left) and application written in Processing language (right). Sequentially: application page with title, artist's information, miniatures of works and presentation of creation.
Source: Author's.

There are several functions that had to be implemented:

- Setup function
  It defines the window size and background color of the application. It loads and scales the content of the gallery using additional functions. All images are given by their file paths and are stored in tables with PImage type. The setup function also implements Leap Motion as an individual variable having access to all gesture recognition.

- Draw function
  This function executes in a loop and therefore the view of the application is refreshing all the time. There can be defined gallery pages that are switching with some events' occurring. It has also definitions of all used colors in the application so we can easily test many features.

- Hover function
  It is responsible for checking the position of the pointer and changing the colors of buttons when the forefinger is over them. Since the gallery interface design was planned first, the position of all active elements is known. This function checks which pixels the pointer is on and then decides about buttons' color.

- Click function
  This function executes when the user clicks or makes a circle gesture above an active element. Depending on what kind of item is chosen, the proper action is called.

- Swipe function

  This function detects the appearance of a user's swipe gesture and recognizes its direction. It changes the pages of the application with respect to active gallery view. Depending on the direction of the swipe gesture, the appearing view is chosen (right – previous page or artistic work, left – next one).

C. Controller

A Leap Motion device was chosen to control the application. It recognizes the position and orientation of hands and fingers as it translates them to the bone system. The controller uses optical sensors and infrared light with a 150 degree field of view. There are two IR cameras generating 2D frames of reflected data with speeds up to 200 frames per second. The most effective range of Leap Motion is up to 60 cm above the device. All data analyses are made on the host computer, which gets information through connection with Leap Motion by USB cable. The 3D position of the hand/fingers is calculated on the grounds of 2D frames.

A special library called Leap Motion for Processing is available. It provides hands and fingers recognition with a list of simple gestures. The drawing functions are also included. They can draw not only hands and fingers, but also a whole arm as they use bones and joints. We decided to examine all motions provided by this library: Circle, Screen Tap and Key Tap gestures for the forefinger as clicking on a button and the Swipe gesture for the hand as switching pages of a gallery.

Chan et al. [9] studied one-hand gestures and they found out that tap motion is the most common one. It is identified with selection function as it resembles interaction with remote controllers like mice, touchpad or gaming device. Swipe gesture was second in terms of popularity as respondents used it for directional tasks like moving something or scrolling in any direction. These gestures were frequently used because they are simply and easy to reproduce by everyone, not only people inside the IT industry.

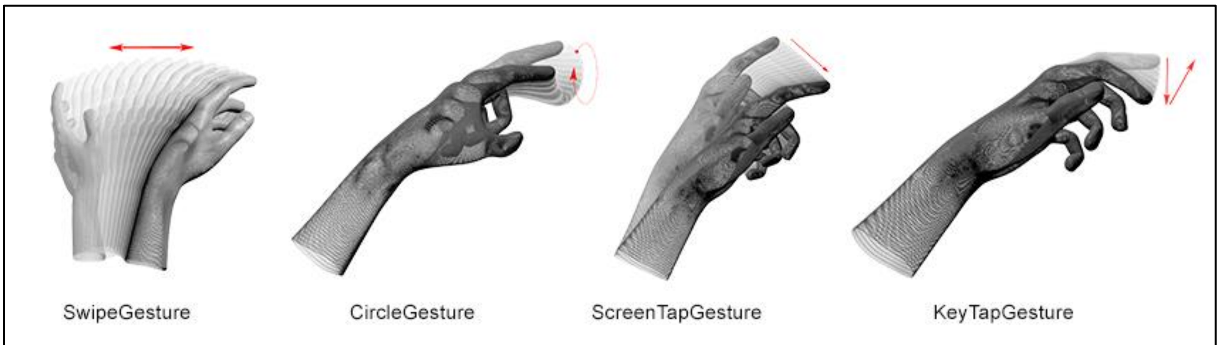

Fig 2. Examined one-hand gestures available in library Leap Motion for Processing.
Source: [10].

Experiment design

We carried out a study of two groups of 10 students (20-22 years old) from Lodz University of Technology from different courses. There were 5 men and 5 women in each group. None of them had used Leap Motion before, but they had a notion of an interactive gallery driven by hand gestures. After the first group test, there were 7 opinions that the gallery sometimes acted peculiarly. Users did not recognize the problem, but we supposed that the application might catch Swipe gesture even if the user did not aim to do it. The second group worked with almost the same version of the gallery, the only difference being that switching pages by hand gesture was turned off.

Respondents were sitting in front of the monitor displaying interactive gallery. They had Leap Motion on table ahead and they could move their hand in mid-air above this device freely. The only constraint was height of the interactive space as it equaled maximum 60 cm.

At the beginning of the study, the whole application was demonstrated. We used a mouse to change pages and select one artistic work from miniatures to zoom in. Then, we presented the instruction of the task: "Please, repeat this navigation, from the first page to the presentation of one work, using only hand gestures". We switched the mouse for the Leap Motion and started timing. The responders had to go over this task three times since the clicking gesture was changed every time (Circle, Screen Tap and Key Tap).

We made some hypothesis about this research:
- Respondents will have similar time results for Screen Tap and Key Tap because the gestures are conterminous.
- The best time will occur for Circle gesture as it is too specific motion to make it unconsciously.
- Changing pages of interactive gallery by Swipe gesture will be unaffected and convenient for respondents.

Results
In Fig. 3 we present results of our research. There is an average time (with marked measurement error with 95% confidence intervals) that responders needed to complete the task. The red bars concern the first group of students testing the application with Swipe gesture on. The blue ones in the second group working on the gallery with the Swipe gesture are excluded.

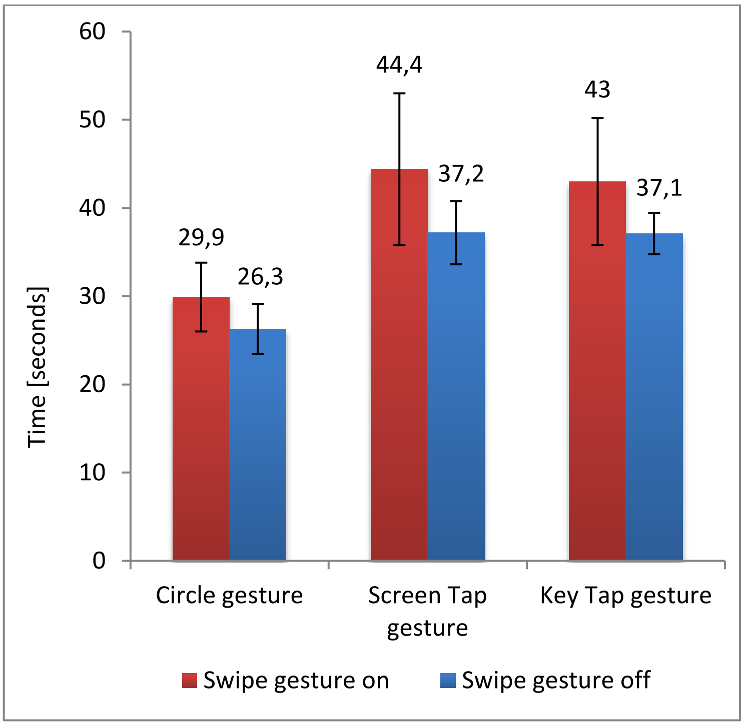

Fig. 3. Chart presenting average time that responders needed to complete the research task.
Source: Author's.

To know more about the results of the research, we calculated some values (Tab. 1). Having variance and variation, we can evaluate how mixed the distribution is. The 95% confidence interval was also calculated and is indicated on the chart (Fig. 3).

We also tested if our results differ significantly from each other (Tab. 2). It is not intuitive as the smaller value is computed, the more meaningful difference is. We calculated Student's T-test which extracts Student's T-distribution widely used in statistics and probability. There are 3 types of this test and we used the one for groups independent on each other as we had two different crops of respondents.

Table 1. Calculated time values for research results. Red values (first for each gesture) attained for application with active Swipe gesture, blue ones (second ones) with Swipe gesture off.

	Circle gesture		Screen Tap gesture		Key Tap gesture	
Average	29.9	26.3	44.4	37.2	43	37.1
Median	30	27	43	36.5	43.5	36
Maximum	40	35	80	51	67	44
Minimum	22	20	31	30	28	32
Variance	43.4	23.1	212.3	36.8	148.4	15.7
Standard Deviation	6.6	4.8	14.6	6.1	12.2	4
Confidence Interval	29.9 ± 3.9	26.3 ± 2.8	44.4 ± 8.6	37.2 ± 3.6	43 ± 7.2	37.1 ± 2.3

Source: Author's.

Table 2. Significant difference between used types of hand gestures. Red values (first for each gesture) attained for application with active Swipe gesture, blue ones (second ones) with Swipe gesture off.

	Circle gesture		Screen Tap gesture		Key Tap gesture	
Circle gesture			0	0	0	0
Screen Tap gesture	0	0			1	0.8
Key Tap gesture	0	0	1	0.8		

Source: Author's.

Discussion

After going over our results, it is obvious that the Circle gesture is working out the best that confirmed our hypothesis. Comparing all averages, minimum and maximum times this motion needs the least to be done successfully. Users learnt this gesture quickly and did it with high precision. There was no problem for them with clicking on one selected element whereas using the Screen Tap or Key Tap gesture was sometimes mistaken by the application. Users moved their hand to get to the designated part of the screen while Leap Motion recognized some gestures. Circle is such a specific motion that the output device did not over interpret it. However, it is so basic that everyone can do it easily.

The interactive gallery with active Swipe gesture was less intuitive for users than the version without this motion. It is evidenced by the times that responders needed to complete the task, as they are significantly different. The minimum time is comparable in both cases, but the maximum time for the gallery with the Swipe gesture is almost the double of the one without this motion. The Leap Motion device recognized this gesture even as respondents did not do it knowingly. This is the reason why our hypothesis was wrong and Swipe gesture did not work well.

Tests with Screen Tap gesture times are the most diverse. Some people completed the task in thirty second but for others it took more than 60 seconds. Because of that range, the confidence interval is the largest for this motion. However, the average time for the Screen Tap and Key Tap gesture does not differ greatly. We can presume that these two motions are very similar for users. Our assumption is evident as the difference between the Screen Type gesture and Key Tap gesture is insignificant (Table 2).

The Screen Tap gesture has the most variable time results. It means that some users did this motion easily while others had troubles with completing the task. It can be caused by a combination of two factors. Firstly, Leap Motion sometimes did not recognize this motion. It is not as specific as the Circle gesture. Users had to learn it from scratch and testing during the experiment took time. Secondly, the Screen Tap is similar to the Key Tap gesture. It was easier for users to make the last gesture because it resembled other moves. If we had changed the order of the gestures, it could have turned out that the Key Tap would have had the most disparate times.

The Circle gesture had the smallest confidence interval because it had the least difference between the minimum and maximum results. This motion is the least differential in terms of time needed for application users to learn and use it successfully. The opposite situation occurs for the Screen Tap gesture.

Conclusions

Our proposal for an interactive gallery driven by hand gestures was tested in terms of four motions. We got the best results for the Circle gesture. This move is very specific so users were not able to do it unknowingly. A circle is not similar to other gestures facilitated by Leap Motion, and so it was recognized by the device effortlessly. What is more, this motion is easy to do so every user could navigate the application without making mistakes.

The Swipe gesture applied as the gallery page changer brought many problems during navigation. Users moved their open hand unconsciously and the Leap Motion device recognized this motion as a Swipe gesture. Pages turned and responders did not know why. This problem is difficult to figure out since users did not do it intentionally. Unfortunately, in our gallery this type of interface element is not a good solution.

Massive elements of interface were helpful in navigation tasks. It was much easier for responders to select one element when it was large. Our assumptions about the low accuracy of hand gesture navigation turned out to be true. Responders with a lack of Leap Motion experience can be the cause for that. We suspect that seasoned users can complete these tasks faster.

In our prospective work, we are planning to add an eye tracking method to our tests. We want to know what responders are looking at and where the marker of a forefinger is at the same time. The results can point to improvements for a more intuitive interactive gallery driven by hand gestures.

References

[1] C. Sinthanayothin, N. Wogwean, W. Bholsithi. Interactive Virtual 3D Gallery Using Motion Detection of Mobile Device. International Conference on Mobile IT Convergence. 2011.

[2] Gergely Sziládi, Tibor Ujbányi, József Katona. Cost-effective Hand Gesture Computer Control Interface. IEEE International Conference on Cognitive Infocommunications. 2016.

[3] He Ling, Li Rui. VR Glasses and Leap Motion Trends in Education. The 11th International Conference on Computer Science & Education. 2016.

[4] Harpreet Kauri, Jyoti Rani. A Review: Study of Various Techniques of Hand Gesture Recognition. IEEE International Conference on Power Electronics. Intelligent Control and Energy Systems. 2016.

[5] Rizal Asrul Pambudi, Nana Ramadijanti, Achmad Basuki. Psychomotor Game Learning Using Skeletal Tracking Method with Leap Motion Technology. International Electronics Symposium. 2016.

[6] Ting-Hsiang Huang, Zhen-Qi Zhuang, Chia-Yen Chen, Bao Rong Chang, Po-Chuan Lin, Yang-Yen Ou. An Interactive Musical Application with Hand Motion and Gesture Controlled Interface. International Conference on Orange Technologies. 2015.

[7] Brian Sanders, Dennis Vincenzi, Yuzhong Shen. Scale and Spatial Resolution Guidelines for the Design of Virtual Engineering Laboratories. W: Jussi Ilari Kantola, Tibor Barath, Salman Nazir, Terence Andre (Eds.). Advances in Human Factors, Business Management, Training and Education. Proceedings of the AHFE 2016 International Conference on Human Factors, Business Management and Society, July 27–31, 2016, Walt Disney World, Florida, USA. 2017. ss. 372-382.

[8] Alberto Scicali, Hans-Peter Bischof. Usability Study of Leap Motion Controller. Conference Proceedings, Conference on Modeling, Simulations and Visualization Methods. Volume 19. 2015. ss. 105-109.

[9] Edwin Chan, Teddy Seyed, Wolfgang Stuerzlinger, Xing-Dong Yang, Frank Maurer. User Elicitation on Single-hand Microgestures. 2016.

[10] Library Leap Motion for Processing. https://github.com/nok/leap-motion-processing/blob/master/reference/leap_gestures.jpg. online: July 2017.

[11] Ionut-Alexandru Zaiti, Stefan-Gheorghe Pentiuc, Radu-Daniel Vatavu. On free-hand TV control: experimental results on user-elicited gestures with Leap Motion. 2015.