International Journal of Advanced Science and Technology
Vol.118 (2018), pp.57-66
http://dx.doi.org/10.14257/ijast.2018.118.06

A Study on Software Framework Design for Spatial Drawing Virtual Reality Application

Ginam Ko[1] and SangHun Nam[2*]

[1]*placeB Inc., Guro-gu, Seoul, Korea*
[2]*Department of Newmedia, Seoul Media Institute of Technology, Seoul, Korea*
[1]*gnko@placeb.com*, [2]*shnam@smit.ac.kr*

Abstract

Drawing has been a familiar way of expression for considerable time to mankind, and virtual reality technology has allow using space as canvas. A system for spatial drawing consists of display device which projects three dimensional image for user to see, controller that replaces a brush and software which creates curved surface and visualizes virtual space using user's input. In this paper, a software framework is designed for developing spatial drawing applications within virtual environment. The framework consists of following modules: head tracking, hand tracking, trajectory analysis, Base curve generator, mesh generator, surface generator, particle generator, 3d visualization and palette, and functions of each module is explained.

Keywords: *Spatial Drawing, Virtual Reality, Framework Design*

1. Introduction

Mankind has been consistently sharing information with others, using drawings to deliver messages and has felt joy of creating drawing as a unique way of expression. Primitive cave painting, Egyptian hieroglyphic and Renaissance fresco were intended to tell a story through images, and as pencil and paper are used for sketching, and computer painting are used for coloring in modern time, drawing has been adapted by humans as a way of expressing visual and sensual experience for long time [1]. With advancing technology, keyboards and mouse has become the primary interface, and since the developing technology of pen-type interface and touch interface, there are ascending cases of drawing directly on the device with hand. Also, drawing are being applied not only to the manufacturing industry but also to the arts due to the rapid development of virtual reality technology [2, 3]. Virtual reality technology has expanded range of expression from plane to space by visualizing three dimensional image real-time for users, using immersive display devices such as CAVE (Cave Automatic Virtual Environment) and HMD (Head Mount Display). Cave Painting has presented spatial drawing using a device like a real brush within CAVE system [4]. Tilt Brush and Quill are the most well-known virtual reality painting applications, and users wear HMD to freely and creatively paint in 3D space [5, 6]. Drawing in a space has progressed into a performance art at exhibitions which draw much attention from viewers, and spatial drawing curriculum are being established.

In the early stage of 3D drawing study, as sensors which acquire spatial position and orientation in space and immersive displays were expensive, most studies revolved around creating 3D models with computer and monitor. 3D modeling researches were conducted to come up with algorithm that transforms 2D drawing data into 3D model automatically, while designing pen-shaped user interface for designers to use easily [7, 8].

Received (January 5, 2018), Review Result (March 10, 2018), Accepted (March 19, 2018)
*Corresponding Author

ISSN: 2005-4238 IJAST

Following the commercialization of 3DTV and rise in sale, visualization of three dimensional image, using spatial sketch with 3D input device and 3DTV has become a popular research topic [9]. Since a system that uses one screen has provided limited visual range, it is difficult to freely draw, thus a method where a wall of monitors built with multiple 3DTV expands FOV has been used. Recently, due to fast progression of virtual reality technology, high-end immersive displays such as Oculus Rift, HTC Vive and HoloLens, which have been utilized mostly for research purposes are being sold for consumer version, and this suggests potential as a new medium not only spatial drawing but also various industries. However, because drawing in three dimensional space entails struggle of drawing on non-resistant space, there are studies devising spatial drawing algorithm with hardware like haptic device [10]. In this paper, in response to increased need for developing spatial drawing application that will enable users to use space freely based on virtual reality technology, related studies on spatial drawing have been analyzed to design spatial drawing software framework, considering required modules and functions found via the analysis.

2. Spatial Drawing Framework Design

A system for spatial drawing consists of display device that presents three dimensional image to users, controllers acting as brushes and software that creates a curved surface and visualizes virtual environments. For visualization, VR HMDs such as Samsung Gear VR, HTC Vive, Oculus Rift, Google Cardboard, Daydream VR, PlayStation VR, OSVR and AR HMD such as Microsoft HoloLens are being used. In order to support HMD devices and input devices, spatial drawing framework should include VR specific functions like user interface system and world system for virtual environment. Each hardware can be inter-operated by using SDK and API supplied by manufacturer. Still, to support different HMDs and input devices being released timely, it is essential to construct and develop a framework that has abstract interface through using middle ware such as OpenVR. OpenVR SDK is created by Valve for supporting SteamVR(HTC Vive) and provides interface for other types of HMD. OSVR(Open Source Virtual Reality) is an open-source software project which has been made in conjunction with partnered experts from several fields, and it provides various HMD and interfaces [11].

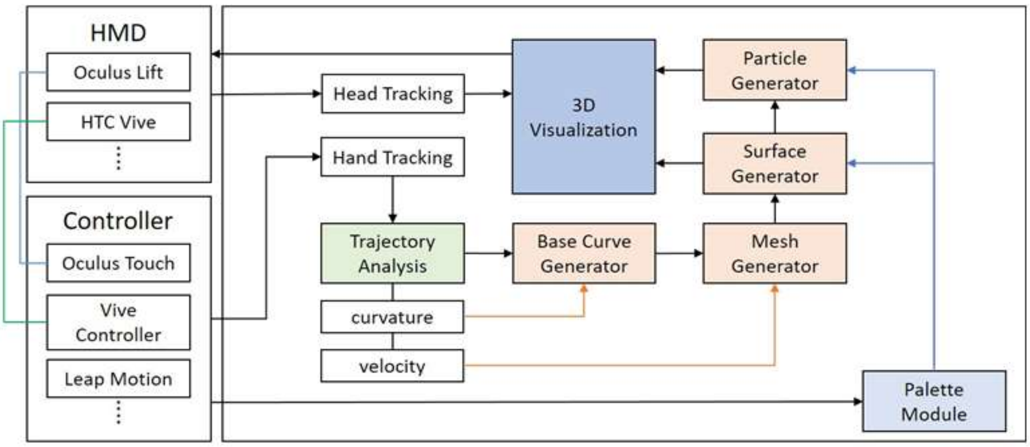


Figure 1. Spatial Drawing Framework Structure

To draw in VR environment, an input device is needed, and it is convenient to use controllers supplied by the HMD companies. The sensors used to track the position and orientation of the HMD can be used together to accurately track the controller's movement. Recently, studies for researching interface which bare hands are used intuitively are conducted, and LEAP Motion devices or hand glove is selected as well. The use of the hand as an input interface require additional technical research to solve the inconvenience of adding an additional sensor to the hand and the overlapping the hand.

Application for spatial drawing has a framework structure similar to Figure 1. During the spatial drawing process, if a user draws by moving controller or bare hands, hand tracking module obtains position and orientation data of input device and saves them. The module can store movement of controller, but it can also do the same with data from placement of finger joints or the finger tips depending on types of hardware. In this module, hardware-dependent input data with different forms is converted to the same output interface (position and orientation) so that hardware dependency is eliminated. The output of hand tracking module are transmitted to trajectory analysis module to calculate geometric attributes such as velocity and curvature with continuous data of the movement. Base curve generator creates base curve which represents center of curves on surface by using position data from trajectory analysis module. Since creating a base curve using all positions from controllers creates inefficient base curves, it is important to create the base curve based on the speed and curvature calculated by trajectory analysis module. Mesh generator creates the mesh of curved surface by using orientation data of controllers and width of the surface based on base curve. The velocity of controller calculated by the trajectory analysis module is used to create a surface that can express the velocity of the brush. A user can use palette module to set values of curved surface drawn for spatial drawing, such as color, texture and particle. Surface generator and particle generator produce diverse shapes of curved surface by applying values of color, texture and etc. from palette module to mesh created by mesh generator. A user-created work is created as stereoscopic images that is visualized in the HMD through 3D visualization module, such as rendering engine or game engine. The 3D visualization module updates stereoscopic images in real-time depending on user's movement and perspective by acquiring position and orientation of HMD.

2.1. Head Tracking

Head tracking is a technology that tracks movement of human head. By tracking the movement of actual head or the movement of HMD device worn by a person, it calculates the person's eye position. With the location of both eyes, VR application can produce realistic, three dimensional image, and it is an important part in providing immersive experience to users by allowing them to interact with virtual world while walking around. Traditionally, the user’s movement had been tracked using magnetic and sound technology, but these days, optical tracking using IR diode is more widely used in VR technology. For tracking an object's movement, there are two methods: one is outside-in method where a camera is installed outside and the moving object works as a marker, and another is inside-out method where a module is installed outside as a marker when an optical device attached to moving object tracks its own movement [12].

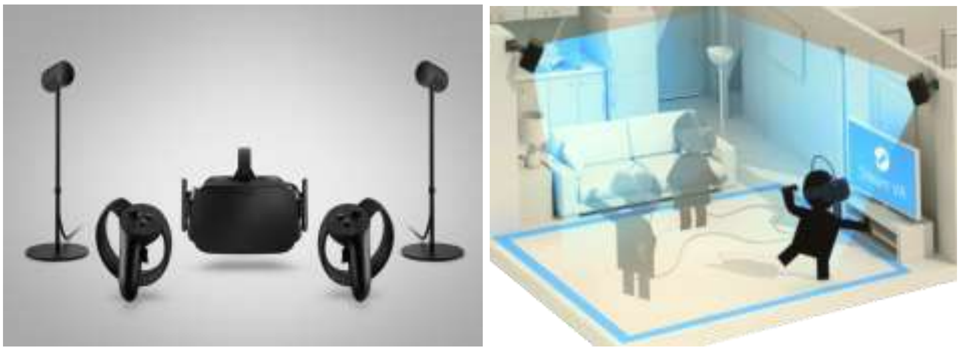


Figure 2. Vision-based HMD Tracking

As seen on Figure 2(left), Oculus Rift crescent bay headset uses outside-in method where position and rotation is calculated by 2 IR cameras tracking IR LEDs of HMD controller. Contrarily, like Figure 2(right), HTC Vive uses inside-out method where a lighthouse installed outside sends out sync blink as well as both horizontal and vertical laser sweep signals. First signal synchronizes all sensors by being transmitted to affect the 3D space simultaneously, and second one is an infrared light laser beam that scans the 3D space. On the surface of Vive headset and controller, there are multiple photo diode installed. A photo diode can tract its position by using time discrepancy in laser hitting into the diodes, and it can also calculate headset's position and rotation with sensors' geometric information. Moreover, it can utilize dead reckoning (path integration) with IMU (Inertial Measurement Units) within headset to update position and orientation fast [13]. Lighthouse is used to eliminate and correct accumulations of errors within the dead reckoning based on inertial measurement [14].

2.2. Hand Tracking

In order to draw in space, controllers supplied by HMD companies can be used, and a particular one may be necessary. Additionally, methods where a human hand is directly used for spatial drawing are being researched. The most common way for tracking user's hand movement is one using a controller. Even if users choose an identical application program, usability is different depending on interface design. As shown in Figure 3, Oculus Rift touch controller is designed that wraps around a hand, and it has been designed to be effective for grabbing or putting an object with hand, while being visualized as a hand in the virtual reality application. For 'Quill' application, spatial drawing is possible by wearing HMD and using Oculus touch controller [6]

Figure 3. Oculus Touch Interface Design

Figure 4. HTC Vive Interface Design

As Shown in Figure 4, HTC Vive is designed to mimic the grabbing of a stick. In the middle, there is a touch sensor that is pertinent for expressing menu such as palette. This type of design is appropriate for coordination of grabbing an object to move it or utilizing weapons such as guns or swords. Within 'Tilt Brush' application, it is possible to spatially draw by wearing HMD and using Vive controller [5].

Recently, there are numerous studies conducted for creating intuitive interface with use of hands. A method that uses gloves magnifies engagement by taking advantage of physical limitation fueled by haptic feedback provided to users and experienced during interaction with objects inside virtual reality. To apply the physical limitation, it has been required to wear a bulky device on hand, thus there are other studies proceeded to devise a way to give sense of reality with vibration from light gloves [15]. As shown in Figure 5, a vision-based LEAP motion controller is attached to HMD without mounting the device in hand. As LEAP motion controllers can track users' hands and fingers within three dimension space, many applications uses a LEAP motion controller for complicated interactions and simulation within virtual reality and augmented reality.

Figure 5. Leap Motion with HMD for Hand Tracking

Various data input from various hardware are converted into position and orientation data in hand tracking module. When the controller is used as an input device, the position and orientation data are calculated relative to the end of the brush that visually represented in virtual reality application. When a user's hand is used as input interface, finger tip positon of glove or the position data of 22 joints mapped to the LEAP motion controller are used for calculation. Mostly, index finger is consider to figure out the position of brush tip, and its orientation is calculated with surrounding joint data values.

2.3. Trajectory Analysis

The hand tracking module computes the position and orientation data of the controllers or the hands operated by the user in the 3D space and outputs it in the form of a set C including the position and orientation of the end of brush such as Equation (1). C can calculate v(velocity), ω(angular velocity), a(acceleration), α(angular acceleration), κ(curvature) and τ(torsion) in each position by differential calculation.

$$\begin{aligned} C &= \{P(position), O(orientation)\} \\ P &= \{(x_1, y_1, z_1), (x_2, y_2, z_2), \cdots (x_n, y_n, z_n)\} \\ O &= \{(\theta_{x_1}, \theta_{y_1}, \theta_{z1}), (\theta_{x2}, \theta_{y2}, \theta_{z2}), \cdots (\theta_{xn}, \theta_{yn}, \theta_{zn})\} \end{aligned} \quad (1)$$

2.4. Base Curve Generator

Frequency of data C input from trajectory analysis module is between 30~60Hz depending on the controller used in hand tracking. In case of creating basic curves using all input points, it is hard to guarantee the real time due to increased amount of calculation, and is inefficient as well because curve quality is not proportionate to the number of points considered. According to the frequency of input device of the device, the N_{skip} value is set as shown in Equation (2), and the basic curve is generated using C' obtained by collecting the data at regular intervals without using the set C of all points.

$$C' = \{P_i, O_i\}, (i' = i\%N_{skip}) \tag{2}$$

In the case of using the same Ns_{kip} value, the quality of created curves was appropriate on a smooth curved line, but when curves were large bending or direction of drawing changed, there is a problem of deteriorated quality of curves. It is possible to figure out the factors showing the degree of curves with curvature calculation, base curve generation module uses an algorithm that variably applies value of N_{skip} according to curvature of input points as shown in Equation (3).

$$C' = \{P_i, O_i\}, (i' = i\%N_{skip}), N_{skip} = A \times Curvature \tag{3}$$

When creating curves with input points, curves that show unstable curvature variation due to user input accuracy lowering and sensor noise from controller in open-air. For making smooth curves, it is needed to use curvature algorithm with C' data. Equation (4) represents Hermite Curve which can create curves with both starting point and direction vector and ending location of the point and direction vector.

$$\begin{gathered} P(t) = H_1(t)P_0 + H_2(t)V_0 + H_3(t)P_1 + H_4(t)V_1 \\ H_1 = 2t^3 - 3t^2 + 1, H_2 = -2t^3 + 3t^2, H_3 = t^3 - 2t^2 + t, H_4 = t^3 - t^2 \end{gathered} \tag{4}$$

2.5. Mesh Generator

Once the basic curve is created, mesh generator generates the position data using the equation of the base curve. Since the orientation data is not generated by the base curve equation, interpolation is used between orientation data between two points. The user can set the width W of the curved surface using palette menu in the application, the position of the vertices are calculated using width W data. The vertices V_1, V_2, V_3 and V_4 are generated using the position of vertex on the base curve and interpolated orientation data of vertex as shown in Equation (5). Four meshed can be generated using four vertices (V_1, V_2, V_3, V_4) as shown in Equation (6).

$$\begin{gathered} V_{1n} = P(t_n) + O_n \times W_n, \quad V_{2n} = P(t_n) - O_n \times W_n \\ V_{3n} = P(t_{n+1}) + O_{n+1} \times W_{n+1}, \quad V_{4n} = P(t_{n+1}) - O_{n+1} \times W_{n+1} \end{gathered} \tag{5}$$

$$\begin{gathered} M_{1n} = mesh(V_{1n}, P_{1n}, V_{3n}), M_{2n} = mesh(V_{2n}, P_{2n}, P_{1n}) \\ M_{3n} = mesh(V_{3n}, P_{1n}, P_{2n}), M_{4n} = mesh(V_{4n}, P_{2n}, V_{1n}) \end{gathered} \tag{6}$$

In the drawing process, applying width of line variably give similar effect as actually drawn with a brush. When drawing in space, using controller's velocity which has been calculated in trajectory analysis module, it is possible to apply the effect of variably the width of the surface. If the value of W is applied as Equation (7), mesh data having a width that is inversely proportionate to drawing speed, when drawing speed is high then stroke would get narrower, and when the speed is low, the stroke would become wider.

$$W_n = B \div Velocity_n \tag{7}$$

2.6. Surface Generator & Particle Generator

As shown in Figure 6, the mesh created in the mesh generator can be used to visualize a variety of surfaces through applying various color and texture. By selecting textures with alpha value from the menu, it is possible to create curved surface with diverse effects. Because the curved mesh generated from mesh generator is drawn with user's location, it is optimized to visualize according to user's input location. In a virtual environment, when users are free to move and observe the spatial drawing, the surface may not be visualized normally. In particular, a multi-view surface design that can respond to user's real-time viewpoint changed is required, and a space drawing algorithm that can visualize multi-view change in real time curved surface is under study. As for network based collaboration and design process, regular curved surface is considered more effective, and for art and entertainment field, multi-view ones is considered more suitable.

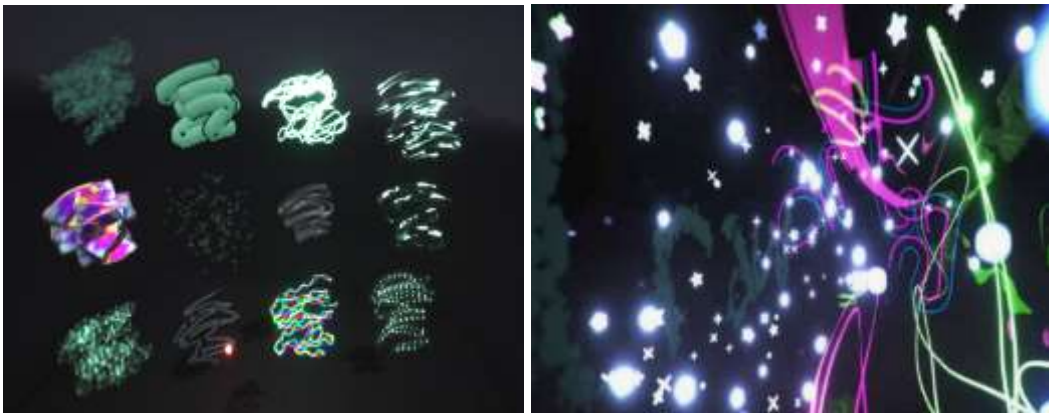
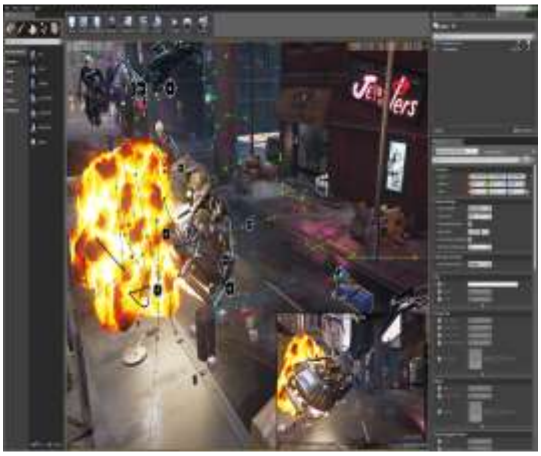


Figure6. Texture and Particle in Tilt Brush

2.7. 3D Visualization

For visualizing virtual environment, the 3D visualization module use a graphic library directly or a game engine that includes graphic engine. To use virtual reality device, it is most common to use a game engine to build a virtual reality applications such as Unity or Unreal Engine as shown in Figure 7. In addition, it is possible to draw in a virtual environment using research-oriented engines such as CRSF (Coexistent Reality Software Framework) [16].

(a) Unity Game Engine

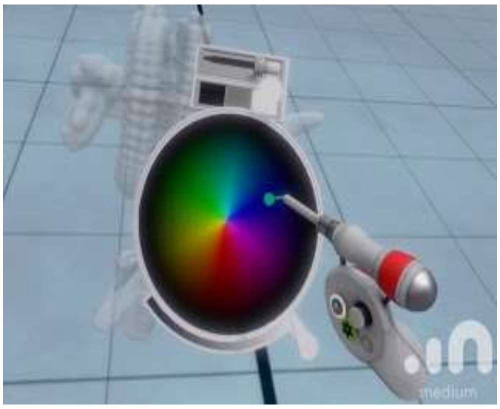


(b) Unreal Game Engine

Figure 7. Texture and Particle in Tilt Brush

2.8. Palette Module

Spatial drawing applications should allow users to create various curved surfaces. Users can use palette menu to set attributes of surface's color, texture and particle in virtual environment. Palette menu can be manipulated by user with controller, and it is saved in spatial drawing application to be used when curved surface and special effect needed to be generated from surface generator and particle generator. As shown in Figure 8, palette menu within virtual space can be designed diversely [17]. Since the menu of spatial drawing is placed in virtual environment different from the menu of application used in the general flat monitor, palette menus must be modeled in 3D. Depending on the number of HMD controllers or the number of hand, the spatial drawing application must design a palette menu design suitable for single-handed, two-handed interfaces. When using HMD controller, selecting menu is easy with multiple buttons on the controller, but if a hand is used directly on interface, there is a need for adding additional input interface that can trigger events and considering sensory range of hand tracking sensor.



(a) Medium by Oculus

(b) Quill by Story Studio

(c) Modbox by Alientrap

(d) Tile Brush by Google

Figure 8. Palette Design in Virtual Reality

3. Conclusion

This study was conducted to analyze studies related to spatial drawing in virtual reality, which is attracting large attention artistically and industrially and to design a software framework which can be used in creating an application for spatial drawing. The spatial drawing software framework can effectively create curved surface, using various input devices, and it has been devised to have users apply various effects on the surface with the palette menu. The software framework includes modules of head tracking, hand tracking, trajectory analysis, base curve generator, mesh generator, surface generator, particle generator, 3D visualization and palette, and function of each module has been explained. In future, it is expected that remote users connected with network will be able to draw simultaneously in virtual space across distance. The future research will be conducted to expand the multi-user software framework and the application that can be used by a large number of users for collaboration of drawing.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(No. NRF-2017R1D1A1B03034041).

References

[1] C. Tricart, "Virtual Reality Filmmaking: Techniques & Best Practices for VR Filmmakers", Taylor & Francis, **(2017)**.
[2] L. P. Berg and J. M. Vance, "Industry use of virtual reality in product design and manufacturing: a survey", Virtual reality, vol. 21, no. 1, **(2017)**, pp. 1-17.
[3] C. Davies and J. Harrison, "Osmose: towards broadening the aesthetics of virtual reality", ACM SIGGRAPH Computer Graphics, vol. 30, no. 4, **(1996),** pp. 25-28.
[4] D. F. Keefe, D. A. Feliz, T. Moscovich, D. H. Laidlaw and J. J. LaViola Jr., "CavePainting: a fully immersive 3D artistic medium and interactive experience", In Proceedings of the 2001 symposium on Interactive 3D graphics, North Carolina, USA, **(2001)** March 19-21.
[5] "Tilt Brush", https://www.tiltbrush.com, **(2018)**, Accessed: 2018-03- 01.
[6] "Quill", https://www.oculus.com/experiences/rift/1118609381580656/, **(2018)**, Accessed: 2018-03- 01.
[7] T. Igarashi, S. Matsuoka and H. Tanaka, "Teddy: a sketching interface for 3D freeform design", In Proceedings of the 26th annual conference on Computer graphics and interactive techniques, Los Angeles, USA, **(1999)** August 8-13.
[8] A. Nealen, T. Igarashi, O. Sorkine and M. Alexa, "FiberMesh: designing freeform surfaces with 3D curves", ACM transactions on graphics (TOG), vol. 26, no. 3, **(2007)**, p. 41.
[9] S. Nam and Y. Chai, "SPACESKETCH: Shape modeling with 3D meshes and control curves in stereoscopic environments", Computers & Graphics, vol. 36, no. 5, **(2012)**, pp. 526-533.
[10] B. Ens, A. Byagowi, T. Han, J. D. Hincapié-Ramos and P. Irani, "Combining Ring Input with Hand Tracking for Precise, Natural Interaction with Spatial Analytic Interfaces", In Proceedings of the 2016 Symposium on Spatial User Interaction, Tokyo, Japan, **(2016)** October 15-16.
[11] M. McCaffrey, "Unreal Engine VR Cookbook: Developing Virtual Reality with UE4", Addison-Wesley Professional, **(2017)**.
[12] D. Bowman, E. Kruijff, J. J. LaViola Jr and I. P. Poupyrev, "3D User interfaces: theory and practice", Addison-Wesley, Boston, **(2004)**.
[13] D. C. Niehorster, L. Li and M. Lappe, "The accuracy and precision of position and orientation tracking in the HTC vive virtual reality system for scientific research", i-Perception, vol. 8, no. 3, **(2017)**, 2041669517708205.
[14] O. Kreylos, "Lighthouse tracking examined", http://doc-ok.org/?p=1478, **(2016)**, Accessed: 2018-03-01.
[15] C. M. Wu, C. W. Hsu, T. K. Lee and S. Smith, "A virtual reality keyboard with realistic haptic feedback in a fully immersive virtual environment", Virtual Reality, vol. 21, no. 1, **(2017)**, pp. 19-29.
[16] "Coexistent Reality Software Frameworks Documentation", http://www.crseed.org/developer /index.html, **(2018)**, Accessed: 2018-03- 04.
[17] "Tool Pallet design in virtual reality", https://www.etc.cmu.edu/projects/blueprint/index.php /2017/02/22/tool-pallet-design-in-virtual-reality/, **(2017)**, Accessed: 2018-03- 04.

Authors

Ginam Ko, he has worked as a CEO at placeB Inc. by researching user interface and experience of new media service, service design of augmented and virtual reality, 2D/3D content design technology, and image recognition of biometrics. He received the B.S degree in new media from Hoseo University, and he received M.S degree in Convergence Engineering from Hoseo Graduate School of Venture.

SangHun Nam, he has worked as an assistant professor at Seoul Media Institute of Technology by researching user interface and experience, game and biomedical technology in virtual and augmented reality in digital human interaction lab. He has worked as an interactive media art technician in collaboration with various artists. He received the B.S degree in mechanical design from Chung-Ang University. He received M.S and Ph. D degree in computer graphics and virtual environment from graduate school of advanced imaging science, multimedia & film in Chung-Ang University. He completed a post-doctoral course at virtual environment Lab in Chung-Ang University by researching spatial sketch and design. He researched coexistence reality, which is mixed reality that measure and express emotions with remote people, at the center of human-centered interaction for coexistence.