Practice Equipment for Attaching a Pouch to a Stoma in a Metaverse Environment

Mizoguchi Michiru[1], Masatoshi Kayaki[2], Takahiro Kunii[3], Yumeno Ryotaro[4], and Noborio Hiroshi[4,*]

[1] Faculty of Health Sciences, Nara Gakuen University, Nara, Japan
[2] Kayaki Koubou Co., Nara, Japan
[3] Kashina Co., Shiga, Japan
[4] Osaka Electro-Communication University, Shijo-Nawate, Japan
Email: michiru-m@nara-su.ac.jp (M.M.); mt14a002@oecu.jp (M.K.); kunii@tetera.jp (T.K.); nobori@osakac.ac.jp (N.H.)
*Corresponding author

***Abstract*—In this study, research was conducted on the development of software that enables stoma pouch application practice using Virtual Reality (VR). If this software can be used to practice wearing the pouch on VR at home, it is expected that people undergoing stoma surgery and their carers can learn how to wear the pouch before and after the surgery and adapt smoothly to the surgery through the realistic tactile and immersive visual effects of VR. In this study, the integration of 3D visual gesture feedback with VR and tactile feedback with the Phantom was specifically investigated: there are no previous studies of smart stomas integrating the visual effects of VR with Meta Quest2 and the tactile effects of the Phantom device, and the visual only or tactile. The experiential effect is much more realistic than either visual or tactile only. This allows patients to touch the stoma pouch rather than just see it, and to learn stoma pouch application skills without getting bored.**

***Keywords*—stoma, pouch, Virtual Reality (VR), tactile, head-mounted display, stoma, tele-training, metaverse**

I. Introduction

Since 1981, cancer has been the leading cause of death in Japan [1] and the incidence of colorectal cancer is expected to increase. Surgical treatment often involves the creation of a colostomy (also known as a stoma) in the abdomen to change the route of faucal evacuation depending on the site of injury. Stool is collected in a device called a pouch, which is used routinely. Providing education on stoma handling during the post-operative hospitalization period following stoma creation may reduce the survival rate of stoma patients [2]. However, under the current Japanese insurance system, hospital stays are short and many stoma patients are discharged without being informed about basic stoma care. In addition, many hospitals, except for large hospitals, do not have nurses skilled in stoma care. Furthermore, patients who are not familiar with the use of stomas may not be able to properly manage additional stomas [3].

Manuscript received October 15, 2023; revised November 25, 2023; accepted March 12, 2024; published October 25, 2024.

Against this social background, the following inexpensive systems and software were developed for use at home after discharge from hospital to enable stoma holders to lead a comfortable daily life.

(1) A self-stoma shape measurement system using a commercially available depth camera [4].

(2) A machine learning system to determine when to change the pouch from the pouch wearing image [5].

(3) A system that allows users to experience a metaverse 3D environment with an HMD and practice wearing it at home by manipulating the environment with gestures [6].

We hypothesized that 3D audiovisual and tactile devices could be used to practice attaching and detaching the pouch to and from the stoma. However, there is surprisingly little research on such smart stomas, both nationally and internationally [7–10], and we are independently conducting research and development on the above three topics.

The aim of this study was to develop a metaverse environment for patients with a colostomy due to gastrointestinal disease, where a pouch can be attached to the colostomy at home.

The study was conducted in the following order. First, two Phantom OMNI units were connected to provide a six-degree-of-freedom tactile experience with three degrees of freedom of force and three degrees of freedom of torque to a three-dimensional model of the human body and stoma on Unity. For example, when the tip of the virtual Phantom stylus on screen touches a 3D object of skin, the user can feel the texture of the skin.

However, attempts to improve the visual feedback by replacing the on-screen virtual Phantom stylus with a hand object did not produce the expected results. Therefore, visual feedback was improved by importing 3D scanned 3D data of the pouch and creating a 3D representation of the pouch on the human body in Unity. The pouch was implemented by editing the noise in the model and creating shaders. Hand movements were tracked using the hand tracking functionality of Meta Quest2 and displayed on the Phantom. Hand movements were also tracked using the hand tracking functionality of

doi: 10.18178/ijpmbs.13.4.104-111

Meta Quest2 to determine the shape of the pouch to push before tactile feedback was obtained from the Phantom. The hand tracking part was implemented to acquire hand geometry, but the part to acquire and record hand geometry was not yet implemented. This would allow the typical hand shape used when placing the pouch on the stoma to be stored.

The hand shape was finally determined and the action of pressing the pouch onto the human body was reproduced. Both Phantom hand tracking for visual and Meta Quest2 hand tracking for tactile can be used in this scene. Hand tracking grabs the pouch and brings it closer to the stoma. By using interference checking algorithms, the hand tracking part of lifting the object could be implemented. The part that pushes the pouch against the stoma is implemented in the object grabbing part of the Phantom. If this part could be combined with the grabbing part of the Meta Quest2, the pushing action would be completed.

As described above, in this study, visual feedback and hand tracking with VR, haptic feedback with Phantom, object movement processing, and interference checking processing were implemented. The pouch and stoma scanned with a 3D scanner were used in this study. The study provided an immersive experience of wearing the pouch in a room.

II. About the Software-Hardware Used

The hardware and software used in this study are described. A desktop computer was used to run the experimental software. Unity Editor 2020.3.26f1 was used to develop the experimental software. Meta Quest 2 was used for visual feedback and Phantom for haptic feedback.

A. *Computer Used*

The computer used in the experiment was a desktop type, with Windows OS, Intel Core i7-4770 CPU, 8 GB memory, and NVIDIA GeForce GTX 1660 Super graphics board. The Phantom used in the experiment only supports IEEE 1394 port connections, so a 1394 port was added with an expansion card.

B. *Meta Quest2*

Meta Quest2 is Meta's VR goggles and uses Android 10 as its operating system. It allows flexible application development and supports the hand tracking functionality described below.

C. *Phantom*

For holding and moving the stylus part, a phantom device developed by 3D Systems is used: one phantom device can freely manipulate positions and postures in six degrees of freedom in three-dimensional space and can experience forces in three degrees of freedom generated by collisions in that space. Therefore, if two units are connected, the position and posture of the six degrees of freedom in 3D space can be freely manipulated and the forces and torques of the six degrees of freedom generated by collisions can be experienced.

The 3D system phantom device is mainly used as a 3D modelling aid, i.e., a manipulation device. A motor is built into the joint between the base, body, and stylus to reproduce the resistance caused by the forces and torques. It is also equipped with sensors to recognize rotation and is designed to recognize a variety of movements. It was therefore deemed suitable for the present study, which required detailed motion and tactile feedback. The computer is connected to the device using an IEEE 1394 connector and cable.

D. *Development with Unity*

Phantom OMNI has 6 inputs (3-DOF position, 3-DOF attitude) and 3 outputs (3-DOF force), so connecting two units results in a metaverse 3D space with 6 inputs (3-DOF position, 3-DOF attitude) and 6 outputs (3-DOF force). As a result, a metaverse 3D space with 6 inputs (3-DOF force, 3-DOF torque) and 6 outputs (3-DOF force, 3-DOF torque) is realized (Fig. 1).

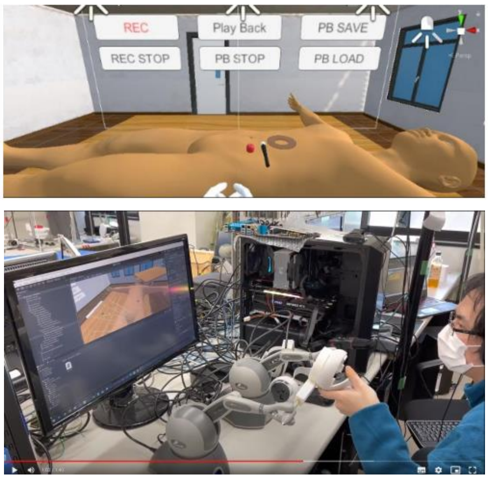


Fig. 1. Two tactile feedback devices with six inputs (3-DOF position, 3-DOF posture) and six outputs (3-DOF force, 3-DOF torque) are connected to touch the stoma.

E. *Handling Meta Quest2 on Unity*

This setup has the advantage of not requiring a PC, such as Oculus Link, which can be executed on Unity while viewing the Meta Quest2 screen, and Build Execution, which allows the Meta Quest2 to execute program on its own. However, an IEEE1394 port is required to connect a haptic device, which is described below, and Meta Quest2 only supports Type-C connection, so it cannot be converted to IEEE1394 due to USB specifications. Even if it could be converted, building the device is impractical as there is no Phantom driver for the Meta Quest2 running on the Android OS.

Oculus Link requires a PC, but the screen display is transferred using Meta Quest2 resources, so no drivers or connectors are needed. In addition, the AirLink functionality allows the device to work both wired and wirelessly. Therefore, with a high-speed network, the system can be operated in much the same way as the system we have built. For these reasons, we decided to develop rather than build a system that works with Oculus Link.

F. Working with Phantom on Unity

The Phantom haptic device is connected to the PC using a 6-pin IEEE connector Once connected to the PC, download the drivers for the Phantom from the 3D Systems website and follow the instructions to install. The connected Phantom is automatically calibrated during installation, so the installer must be run with the stylus attached to the device. Once the driver installation is complete, you must define the device name for the Phantom in the driver software. If this is not done, the Phantom will not be recognized by Unity. Double-click the "Phantom Configuration" icon created on the desktop and click "Add" to set the appropriate device name. Click "Apply" to close the window and apply this name to the entire system.

Once the setup is complete, you must ensure that the sensor and drive unit are in good working order before using the Phantom device. Failure to do so may cause the Phantom to behave unexpectedly, damaging the unit or causing the Phantom to move in an unintended direction, damaging or injuring surrounding objects. This test is performed by double-clicking the "Phantom Test" icon created on the desktop. The gyro sensor test reminds the user to keep the stylus close at hand as the drive unit moves during the test. When the bottom part of the screen turns green and 'Test Passed' is displayed, click 'Next' to proceed to the next test. Repeat this process until the last test, and if 'Test Passed' is displayed for all tests, you know that the Phantom you are using is working correctly.

The following experiments assume that the driver installation described in this chapter has been completed and that 'Test Passed' is displayed in all Phantom test programs.

III. Implementing Haptic Feedback with Phantom

To use the Phantom haptic feedback functionality on Unity, the "Open Haptics Unity Plugin" from the 3D System must be imported into the target project on Unity. You also need to define a device name for the Phantom.

A. Setting up Tactile Feedback with Phantom

The Haptic Material script that must be used to obtain haptic feedback in Phantom is provided with the asset. This script is attached to the object for which haptic feedback is required. After binding, the feedback can be modified by changing the Basic Properties property. Skin feedback is shown in Fig. 2, whereas elastic feedback is added by changing the value of Stiffness. In this experiment, the values were tweaked to obtain the closest skin feedback.

B. Virtual Stylus Implementation Principles

The currently implemented virtual stylus uses part of the Phantom Omni. The currently implemented virtual stylus uses part of the Phantom Omni model embedded in the Phantom asset – the 3D model of the Phantom originally existed on the left side, as shown in Fig. 3, and was synchronized with the Phantom's movements. This made it convenient for debugging the Phantom's behavior.

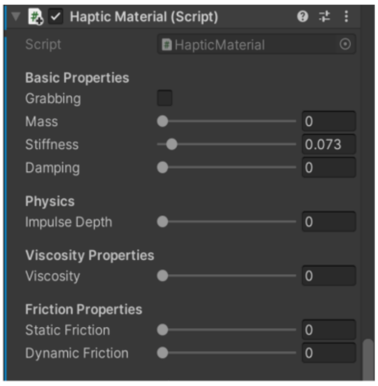


Fig. 2. Setting tactile properties.

Fig. 3. Principle of virtual stylus implementation.

However, as the purpose of this study was to fit a stoma pouch and the presence of the Phantom model within the virtual environment would have compromised the reality, the Phantom model was not validated using Unity functionality. As shown in Fig. 4, the Phantom model part was disabled using Unity functionality and only the virtual stylus was displayed instead of the hand.

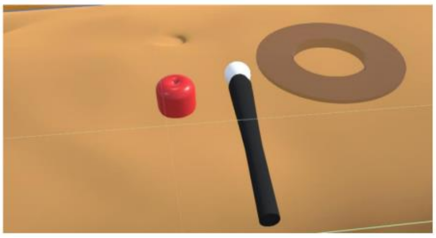


Fig. 4. Virtual stylus.

C. Handset Implementation

The script has a 'Joint' function that can be specified on any object. In this case, we modified this script and applied it to a purchased asset, the 'VR male hand' object in Fig. 5. The 'Joint' function is treated as a joint part of the device in the Phantom asset, so the same virtual stylus movements as when moving the Phantom, we thought it would be possible to move the hand object in the same way as the movement of the virtual stylus when moving the Phantom. Virtual stylus when Phantom is moved. We wondered if it would be possible to move the hand object in the same way as the movement of the virtual stylus when Phantom is moved. We wrote the wrist part of the hand asset in a Joint function and executed it. However, when I rotated the stylus, the hand object rotated but did not move as freely as the virtual stylus.

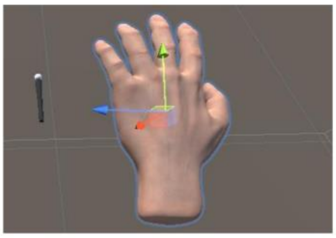
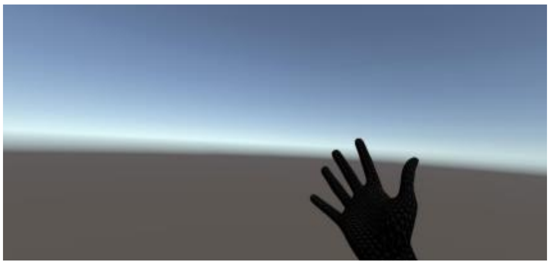


Fig. 5. Implementation of purchased hand assets.

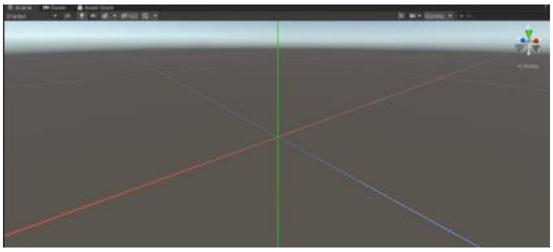
By setting up the hand as described above, the technique of attaching the nurse pouch to the stoma in Fig. 6 can be realized and the trajectory of each finger joint during the procedure can be recorded (Fig. 7). Therefore, we want to quantitatively distinguish the differences between the skilled and novice hand techniques and record the problems of the novice by machine learning the differences.

(a)
(b)
(c)
(d)

Fig. 6. Stroboscopic images (a)–(d) with a pouch attached to the stoma. A standard hand shape is created in each scene.



(a)



(b)

Thumb Fingertip x	Index Fingertip x	Middle Fingertip x	Ring Fingertip x	Pinky Fingertip x
0.1367421	0.1448041	0.1776824	0.1902054	0.1968176
0.1291159	0.1262937	0.1534823	0.1630802	0.1660477
0.1349047	0.1313454	0.1446195	0.1576756	0.1677165
0.1435203	0.1393247	0.1505377	0.1623389	0.1737107
0.1474914	0.1423903	0.1532259	0.1647424	0.1760357
0.1474932	0.1423932	0.1532564	0.1647587	0.1760343
0.1487353	0.1437702	0.1578859	0.1680166	0.1778766
0.1474983	0.143617	0.1579954	0.1675998	0.1782301
0.1471829	0.1435698	0.1577543	0.1670946	0.1779962
0.1473242	0.1436397	0.1576229	0.1668257	0.1777699
0.1456769	0.1434846	0.1594086	0.1685385	0.1792921
0.1450215	0.1435987	0.1601771	0.1692637	0.1797855
0.1406362	0.1440869	0.1583639	0.1680901	0.1823346
0.1380856	0.1445637	0.1578012	0.1676855	0.1834893
0.1364518	0.1447924	0.1575956	0.1674995	0.1838293
0.137826	0.143743	0.1552395	0.1659657	0.1815448
0.1367421	0.1448041	0.1776824	0.1902054	0.1968176
0.1291159	0.1262937	0.1534823	0.1630802	0.1660477
0.1349047	0.1313454	0.1446195	0.1576756	0.1677165
0.1435203	0.1393247	0.1505377	0.1623389	0.1737107
0.1474914	0.1423903	0.1532259	0.1647424	0.1760357
0.1474932	0.1423932	0.1532564	0.1647587	0.1760343
0.1487353	0.1437702	0.1578859	0.1680166	0.1778766
0.1474983	0.143617	0.1579954	0.1675998	0.1782301
0.1471829	0.1435698	0.1577543	0.1670946	0.1779962
0.1473242	0.1436397	0.1576229	0.1668257	0.1777699

(c)

Fig. 7. Recording the position of hand joints in 3D space. (a) Fingers and joints. (b) X, Y, and Z coordinates of the finger joints (in part) from left to right, with the red axis as X, the green axis as Y, and the blue axis as Z. (c) This can be plotted in 3D space to visualize the movement of the joints during the process of placing the pouch on the stoma. Differences between skilled and novice users can be quantitatively evaluated. In the future, it can be linked to machine learning.

D. Grabbing an Object with Phantom

Phantom's Material Script functionality includes the ability to simulate grasping. By implementing this function, it is possible to reproduce the action of lifting an object while pressing the two buttons on the stylus part of Phantom. By specifying the concept of weight, an object in the virtual space can be grasped by pressing the buttons while applying a load to the hand holding the stylus. This load is intended to simulate holding a weighted object, and the Phantom set includes several sample scenes, including one in which an object is grasped. The script used in this sample is shown below.

In this case, the part of the script used in this sample that handles object grasping was used. The object-specific parts were modified. Specifically, the first step is to specify which button is assigned to the object to perform the grasping operation. It was possible to assign only one button. However, because of actual trials with several users, it was not possible to assign a button other than the button assigned when the user performed the grasping operation. However, when users performed the grasping operation, they operated as if they were grasping the stylus. This result suggests that the user is wearing VR goggles and may need to use one of the buttons. In this case, it is considered better to use one of the buttons. Therefore, two buttons were enabled.

In the "Device ID" field of the material script, add the device ID that is displayed when the PHANToM_Configuration software is started. Then, the value immediately below (Grabble) has an initial value of 0, so change this to 1 to enable grab operation. Since the grab operation is not available at this point, attach a Rigid body to the grab object using Unity's Functional Component. Enable the 'Use Gravity' property of the attached Rigid body, press the 'Run' button in Unity, move the virtual stylus to the object you want to grab using the Phantom stylus, and hold down either or both buttons to move it upwards. The lifting motion is reproduced on the Unity screen, as shown in Fig. 8. It is also possible to simulate the concept of weight during the lifting motion by changing the value of the Mass function in the Phantom's material script, using the motor built into the stylus joint.

Fig. 8. Gripping a faceplate with a virtual stylus.

IV. Implementation of VR Visual Space by Meta Quest2

Phantom alone can only provide tactile feedback. In addition, the practice of using a stylus to apply a pouch on the Unity screen lacks realism. Therefore, we thought that by using Meta Quest2 virtual reality goggles from Meta, we could practice without losing the sense of realism. We also aimed to incorporate the shape of one's own hands into the pouch pasting practice by implementing hand tracking using a high-precision tracking camera (Fig. 9).

Fig. 9. Hand tracking.

A. Asset Implementation and Setup OpenXR Backen

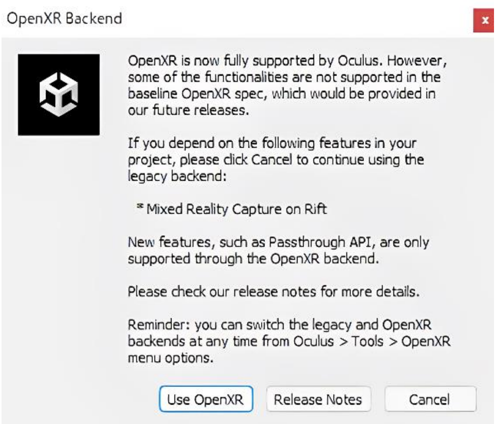


Fig. 10. Confirmation dialog for OpenXR use.

First, import Meta Quest2 assets into your Unity project. To import an asset, search for it in Asset Management as you would any other asset and click 'Import'. However, you will be asked if you want to use OpenXR, as shown in Fig. 10. In this study, OpenXR did not work correctly with some of the haptic simulations in Phantom, so we decided not to use OpenXR by clicking 'Cancel'.

B. Realization of Viewpoint Shift by VR

To reproduce the pouch-wearing environment in a VR environment, the player's head must be moved in any direction to shift the viewpoint, as in commercially available VR games. However, with this method, the viewpoint is fixed and the entire screen follows the player's perspective, so the player can only look in one direction. As a result, the player could not get a close-up view of the mount and had to change cameras. Therefore, the default main camera was removed: once the Oculus Integration was successfully installed, a folder called Oculus was created in the project, and the OVRCameraRig was placed in the folder where the main camera was originally placed (Fig. 11). Now, when you move your head, your line of sight will move.

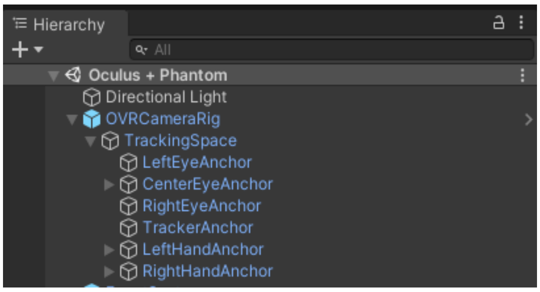


Fig. 11. Camera settings.

C. Allowing the Hand to Be Placed on the Virtual Space

For the hand shaping scene, the hand shapes were obtained using the hand tracking functionality of Meta Quest2. First, open the Oculus Project Config in the Project Oculus folder. On the right side of the project, under "Hand tracking Support", select "Hands Only" to enable hand tracking. Next, prepare the hand objects that will appear in virtual space when the hand is placed in front of the Meta Quest2: associate the OVRHandPrefab in the Prefab folder with the OVRCameraRig's Left and Right respectively. At this time, the Left property of the OVRHand is set to Right by default. Therefore, rewriting the Right part to the Left will work correctly.

Fig. 12. Implementation Structure of grasp judgment.

Next, enable hand tracking in Meta Quest2: open the Meta Quest2 Settings app and select the System item. To make the camera recognize the hand, turn on the Hand Tracking switch. By applying these implementations to your program, you can implement hand tracking as shown in Fig. 12.

D. Implementing Hand Collision Detection and Grabbing Process

Phantom can be held, but the stylus, which is suitable for tactile feedback, limits the range of movement and it was difficult to visually recognize how far Phantom could be moved while wearing the Meta Quest2. We solved this problem by using hand tracking as an aid to object grasping. First, the hand is determined in a hand geometry scene. Next, hand tracking is used to align the 3D models of the pouch and colostomy in the correct orientation. Once the models are aligned, haptic feedback using the Phantom reproduces the pushing action.

In this case, a script for the grabbing action was created and another scene was created in Unity to check the action. There, the player can grab a red ball placed on the ground, as shown in Fig. 13. The OVR hand, which is displayed when recognized by hand tracking, has a sphere at the tip of the index finger for judgment. The sphere is visible in Fig. 12 but is set to be transparent when in use. When the thumb detects the sphere, the script is set so that the object floats up. As a result, the object appears to float. At this point, the fingertips of the hand are slightly stuck in the object and this needs to be corrected. Collision detection occurs when the OVR hand collides with an object. This collision detection is when the virtual hand collides with an object when the virtual hand is moved by hand tracking. For example, if the virtual hand collides with a faceplate or pouch, the hand will move in response to the collision without slipping through. A way to implement this is to use the Physics Capsules function. This function makes it easy to implement collision detection. If the collision detection is implemented correctly using this function, the collision detection by the virtual hand will succeed.

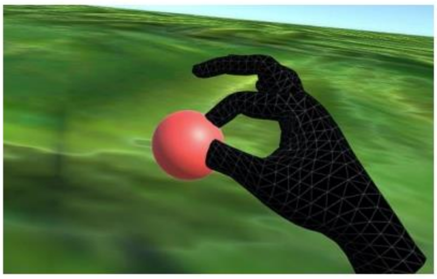


Fig. 13. Grabbing with hand tracking.

V. Pouch Implementation

As mentioned above, the human body and room models were already completed (Fig. 8). Therefore, the unimplemented parts of the simulation were to enable feedback from the Phantom and to improve the immersive experience of VR with Meta Quest. The pouch worn by the patient in the simulation was also implemented.

A. Noise Correction of the 3D Scanned Pouch

To improve the realism of the pouch used in the virtual space, an actual stoma pouch was used in the pouch application training. However, this model was contaminated with noise, so the model was edited as shown in Fig. 14 using MeshMixer software that can edit 3D objects. The left side is before the modification and the right side is after.

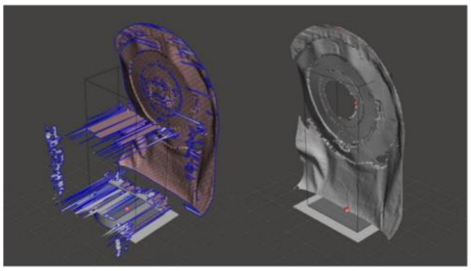


Fig. 14. 3D model of the pouch before and after noise removal.

The jagged noise generated when the back was scanned was removed and the uneven shape was reflected, but the data as an object was missing. In addition, there was a hole in the center of the scan data. As the scan data was flat and had no thickness, a stacking process was used to add thickness like the real object. This data was used to fit a stoma pouch in a virtual environment.

B. Fitting the 3D Scanned Pouch

The 3D data of the pouch edited earlier was imported into Unity and the object was placed. However, as shown in Fig. 15, only the back side of the porch was displayed, and the front side was not displayed correctly. The material color was not set, so the appropriate color was set, but to no effect.

Fig. 15. Pouch is not drawn correctly.

Referring to the official Unity documentation, it was found that some 3D objects can be imported using the ‘Cull’ function. This is a feature that omits the depiction of invisible parts and reduces the weight of run-time behaviors. In this case, the user grabs the pouch in virtual space and puts it on. If the surface of the pouch is not drawn, the surface is not visible when the user grabs the pouch, which makes it less realistic. We also found that once this feature is enabled, it cannot be disabled from the GUI, i.e., the Unity Hierarchy.

Therefore, we decided to create shaders that circumvent the depiction limitations caused by this feature. Shaders are originally used to change an object's shadow to something arbitrary when depicting shadows in Unity. However, we found that using these shaders would prevent Cull from omitting the shadows. The shaders we created are based on the shader files used by default in Unity as described in the Unity documentation, with a description added to disable the Cull functionality. These shaders must be associated with Material, a Unity feature. As a result, the image now displays correctly, as shown in Fig. 16.

Fig. 16. Pouch is displayed correctly.

C. *Specifying Pouch Color and Creating Shader Settings*

The 3D object of the imported stoma pouch has no color specified, so it appears white when moved as it is. As the actual stoma pouch is close to the color of human skin, appropriate RGB values were set to increase reality. Therefore, a Unity material was created to make the RGB values closer to the skin color. In this case, R: 245, G: 222, and B: 179 were set. The material also can set and reflect arbitrary shaders. As a result, colors were added and surfaces were rendered correctly, as shown in Fig. 17.

Fig. 17. Pouch with material set.

VI. CONCLUSION

In this paper, the authors created a tool to help stoma holders practice pouch application techniques remotely from home. The authors had already measured the shape of the stoma with a smartphone and built a model of it in the UNI metaverse space. They also measured various pouches with a 3D scanner and prepared them in the same metaverse space. In this study, basic research on the practice of remotely attaching a virtual pouch to a virtual stoma was conducted using the gesture function of OculusQuest2 and the haptic feedback function of the Phantom OMNI. The main future tasks include the construction of a viscoelastic material deformation reality specific to stomas and pouches, the estimation of human tactile perception functions through vision, and the realization of in-home trials through information networks.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Mizoguchi Michiru and Noborio Hiroshi contributed to the research concept and design; Mizoguchi Michiru and Noborio Hiroshi conceived the project and the study hypothesis; Noborio Hiroshi and Takahiro Kunii designed the algorithm with programming, and Masatoshi Kayaki, Takahiro Kunii, and Yumeno Ryotaro designed the software, respectively; Mizoguchi Michiru and Noborio Hiroshi analyzed the data executed from them; Mizoguchi Michiru, Takahiro Kunii, and Noborio Hiroshi checked the data; Mizoguchi Michiru and Noborio Hiroshi analyzed and evaluated the data; Hiroshi Noborio clearly wrote the paper by manuscript rewriting; all authors have approved the final version.

FUNDING

This study was partly supported by the 2021 Specific Joint Research, Fundamental Mechatronics Research Institute, Osaka Electro-Communication University. Also, this study was partly supported by the 2022 Specific Collaborative Research at Fundamental Mechatronics Research Institute, Osaka Electro-Communication University.

REFERENCES

[1] Ministry of Health, Labor and Welfare. Basic plan for cancer control. [Online]. Available: https://www.mhlw.go.jp/stf/seisakunitsuite/bunya/0000183313.html

[2] Foundation for Promotion of Cancer Research. Cancer statistics in Japan of 2022. [Online]. Available: https://ganjoho.jp/public/qa_links/report/statistics/pdf/cancer_statistics_2022.pdf

[3] E. Takamizawa, *et al.*, "A longitudinal study of changes in the subjective quality of life of patients after surgical construction of colostomy," *J. Jpn. Soc. Cancer Nurs.*, vol. 13, no. 1, pp. 35–42, 1999.

[4] M. Mizoguchi, M. Kayaki, T. Yoshikawa, *et al.*, "Selection and evaluation of color/depth camera for imaging surgical stoma," in *Human-Computer Interaction. Interaction Techniques and Novel Applications*, M. Kurosu, Ed., Springer International Publishing, 2021, vol. 12763, pp. 601–614.

[5] M. Mizoguchi, S. Watanabe, M. Nakahara, and H. Noborio, "Comparison of the accuracy of pouch replacement timing decisions using image generation artificial intelligence and machine learning," in *Proc. International Conference on Human-Computer Interaction*, 2023, pp. 107–120.

[6] T. Kunii, K. Watanabe, M. Mizoguchi, and H. Noborio, "System to check organs, malignant tumors, blood vessel groups, and scalpel paths in DICOM with a 3D stereo immersive sensory HMD," in *Proc. International Conference on Human-Computer Interaction*, 2023, pp. 431–449.

[7] T. Shigeno, *et al.*, "Relationship of stoma self-care status with anxiety and QOL in ostomates," *J. Jpn Soc. Stoma Continence Rehabil.*, vol. 33, no. 3, pp. 71–80, 2017.

[8] P. B. Jeppesen, M. Vestergaard, E. B. Boisen, and T. A. Ajslev, "Impact of stoma leakage in everyday life: Data from the ostomy life study 2019," *Br. J. Nurs.*, vol. 31, no. 6, pp. 48–58, 2022.

[9] J. O. Silva, *et al.*, "Original article Quality of Life (QoL) among ostomized patients – across-sectional study using stoma-care QoL questionnaire about the influence of some clinical and demographic data on patients' QoL," *Journal of Coloproctology*, vol. 39, no. 1, pp. 48–55, 2019.

[10] C. Liao and Y. Qin, "Factors associated with stoma quality of life among stoma patients," *International Journal of Nursing Sciences*, vol. 1, no. 2, pp. 196–201, June 2014.