Walking Simulation for VR Game Character using Remote Sensoring Device based on AHRS- Motion Recognition

J.Y. Kim1, Member, IEEE, W.H. Lee 2, Member, IEEE
1Graduate School of Game, Gachon University, 1342 Seongnam Daero, Sujeong-Gu, Seongnam-Si, Gyeonggi-Do, Republic of Korea.
2 GSAIM, Chung-Ang University, 84 Heuksuk-ro, Dongjak - Gu, Seoul 06974, Korea

ABSTRACT In this paper, the remote sensoring device for controlling the avatar in the VR(virtual reality) game which is being worn by the HMD(head-mounted display) was investigated. This remote sensoring device, which people use while sitting on a swivel chair, minimizes sensor attachment and removes parameters for pelvic rotation angle in standing position. The rotating chair, which also functions as the center-weighted stationary belt of the conventional treadmill-type input device, has the advantage that the head trekking function mounted on the HMD can be utilized freely and the balance feeling of the user can be stabilized in a sitting position on the chair. In this posture, the sensoring input device was developed using Bluetooth wireless communication to freely turn and walk. Using the data from the inertial measurement device, which was a combination of the speedometer and the accelerometer, a sensoring input device for recognizing the user's leg motion was created. The running speed of the recognized leg was designed to be close to the average walking speed and the running speed of the person, so that the running and walking in the game can be separated and implemented. The motion velocity of the leg would be calculated through each speedometer measurement, the value would be integrated, and the error would be corrected by the Kalman filter to obtain the user's knee angle. In other words, applying a Kalman filter to data obtained from sensors such as accelerometers, angular speedometers, and geomagnetic sensor can determine the direction of motion of the legs. This integrated system is called the AHRS(attitude and heading references system).The knee joint angle and the value of the pressure sensor were used as threshold values to determine the state of the leg when walking, and the angular velocity derived from the movement of the leg was used to determine the moving speed of the avatar in the VR game. The recognition rate was calculated by comparing the number of times the user attempts to move the avatar by moving the avatar and the number of times the avatar of the game was moved, and the distance of the avatar was measured to evaluate the performance of the proposed system.

INDEX TERMS Remote sensing, Input device, Wearable computer, VR environment, AHRS

I. INTRODUCTION

After Ouclus Lift DK1 was unveiled to ‘Kickstarter’ in the US cloud funding site in 2012, the size of the virtual reality content market has exploded [1]. Virtual reality (VR) is a virtual environment similar to a real one using a computer, but after the Oculus Lift DK1 HMD was released, a head-mounted display (HMD) is now being worn. It has become a word that means “technology that makes you feel three-dimensional” [2].

As the virtual reality begins to attract attention to the public, virtual reality contents are being produced in various fields such as travels and movies, and various types of hardware for expressing them are rapidly being developed. In order to experience the virtual reality contents, the user needs to wear the HMD, which results in restriction of the user's behavior. First, there are input restrictions. It is difficult to use keyboard and mouse, which are input devices used in computer, for users whose vision is blocked by wearing an HMD. If you keep your head spinning with HMD you can miss the exact keyboard and mouse position[3]. Second, there are restrictions on movement. For the reasons stated above, since
the HMD is not suitable for use with a keyboard and a mouse, another input method for moving the views must be provided.

In order to solve the above two problems, this paper developed an AHRS (attitude and heading reference system)-based motion recognition input device that can control the user's avatar according to the user's step, and apply it to the VR game using it.

The input device was designed to compensate the disadvantages of the commonly used remote control system, laser system, and treadmill system, and to stabilize the balance sense, considering that the HMD is being worn to block the user's view.

Generally, a wearable device through motion recognition determines avatar status based on body data. However, contents that can use an avatar to receive such telegraph data are complicated in the development stage, and such input devices require many sensors and are expensive. In addition, they are difficult to apply to contents such as games developed in the past.

The input device developed in this study minimizes the amount of devices attached to the body and was developed so that it can be easily applied to existing games through key input hooking. The user's recognition rate of the developed input device will be verified.

II. RELATED RESEARCHES

A. Movement in VR environment game

As shown in the Fig. 1. <HTC Vive> is device for virtual reality that implements Room Scale through a laser sensor called a base station, which senses the movement of the user in the indoor space and moves the avatar along with the movement of the actual user [6]. HTC is a smartphone vendor in Taiwan. This is a method of detecting the position of the HMD and the position of the hand controller, which is advantageous in that the rotation delay time is short in the real-time sensing process. However, in the disadvantage is that a space for installing the laser sensor is required, and the sense of balance becomes dull and the risk of collision with obstacles is present when the HMD is blocked by the field of view. In addition, since it is necessary to make a game in consideration of the moving radius in the game development stage, it cannot be used in a genre game in which an avatar must move freely.

The moving operation method also varies depending on the genre of the game. As a typical example, you do not need to walk an avatar in a racing game. In the VR game, which is the most engaging issue, there are many cases where the manipulation method is similar to the gesture of the situation in the game. Therefore, in the racing game in the VR environment, the gesture of turning the steering wheel and stepping on the accelerator is being used as the manipulation method [7] [8].

Many companies that produce virtual reality contents use a method to synchronize a user's actions and an avatar's actions in order to increase immersion in VR environment [9]. For the same reason, studies are being actively carried out to synchronize the movement of the user and the movement of the avatar. The results of these studies are the light house method and the treadmill method.

The only device that has been commercialized as a light house is the HTC Vive base station introduced in Section 2.1 above. The base station can only use 5m of space. It is not easy to use 5m of indoor space as a field for VR games only. Even if you use it, if you move your views 5m away with your eyes closed, you will lose your sense of position, and it is likely to cause an accident with an obstacle. Thus, content that uses the base station to move an avatar can be set to a narrow range of travel or use other available input devices so that the user does not have to walk away.

The treadmill method basically consists of a footrest and a waist belt. The waist fastening belt serves to prevent the body from escaping from the treadmill which must be moved into place, and the footrest recognizes that the user's feet are moving. The recognition method differs depending on the product. For example, Virtuix's <Omni> consists of well-slippery shoes and a pair of pads attached to the shoe so that the tracking pad recognizes the user's movements and enters data into the computer [10]. On the other hand, Cyberith's <Virtualizer> is a flat pad that can be used while wearing ordinary socks. A slippery socks slide the foot on the footrest, and the friction sensor on the footrest inputs the data to the computer [11].

The treadmill method implements the input method closest to the user's walking gesture, but the problem is the price. In the Fig. 2. (a) <Omni> has a bundle price of $1,296 and [12], and <Virtualizer> in (b) is priced at $1049, which is quite expensive [13].

![FIGURE 1. HTC Vive Base Station](image)

![FIGURE 2. A demonstration scene of <Omni> (a) and Operating principle of <Virtualizer> (b)](image)
B. Avatar moves in VR environment game

In order to construct the motion recognition device studied in this paper, the speed and angle when the leg moves should be expressed as numerical values by data. This section provides information on the sensors required in the study.

a. MEMS IMU

The Inertial Measurement Unit (IMU) is an accelerometer, gyroscope, and sensor that is coupled to a geomachine and altimeter, depending on performance. It is used to measure acceleration, angular velocity, and gravity. In addition, the position and direction of the sensor can be estimated by processing the data of the sensor. It has been used mainly in aviation and drone field, and recently, researches are being actively used for motion recognition that tracks human motion.

MEMS is an acronym for Micro-Electro-Mechanical Systems, which means a technology for fabricating microelectromechanical elements with dimensions of 0.1mm to 0.0001mm. Micromachines created with this technique have sub-micrometer accuracy.

The device manufactured in this study uses the IMU sensor made by MEMS technology to minimize the weight without disturbing the step because the foot angle should be recognized by measuring the angle of the leg at the top of the ankle.

MEMS is an acronym for Micro-Electro-Mechanical Systems, which means a technology for fabricating microelectromechanical elements with dimensions of 0.1mm to 0.0001mm. Micromachines created with this technique have sub-micrometer accuracy.

The device manufactured in this study uses the IMU sensor made by MEMS technology to minimize the weight without disturbing the step because the foot angle should be recognized by measuring the angle of the leg at the top of the ankle.

Combining the accelerometer and gyro spokes can represent as much as 6DoF. If the magnetic sensor is additionally connected, it can measure up to 9DoF, i.e. 10DoF at atmospheric pressure. In this study, 6DoF was used because it deals only with acceleration and angular velocity.

b. Kalman filter

In MEMS modules, accelerometers are sensitive to shocks, and gyroscopes are sensitive to temperature, resulting in varying noise. Accumulated errors arising from noise generated by the IMU module during the processing of accelerometer and gyroscope data do not result in the desired results. Therefore, in order to obtain the desired result, the data of various sensors must be integrated to overcome the disadvantages of each sensor and to infer the present state by compensating the error generated from noise [14].

The Kalman filter is an algorithm that recursively processes the noise of input data over time in a linear system and minimizes errors to predict the optimal state. This algorithm can be applied to the case where the measurement value contains an error and the state at the previous time point has a linear relationship with the state at the current time point. Typical examples are analog signals such as object position, velocity, and sound.

As shown in the Fig. 3, the Kalman filter estimates the current value based on the estimated value at the immediately preceding time. It cycles through these two steps: a prediction step that calculates the expected state when the input is applied, and a correction step that estimates the current state inductively using the error between the measured state and the state predicted in the previous step [15].

\[
\begin{align*}
\hat{x}_{k-1} &= A\hat{x}_{k-1} + Bu_k \quad (2) \\
\hat{P}_{k-1} &= AP_{k-1}A^T + Q \quad (3)
\end{align*}
\]

(2) is a prediction for the estimated value \( \hat{x}_{k-1} \) and \( \hat{P}_{k-1} \) at the entry point. Select the initial value before entering the recursive algorithm. \( k \) is the index over time, \( \hat{x}_{k-1} \) is the estimated estimate, and \( \hat{x}_k \) is the initial value. Similarly, \( P_{k-1} \) is the error covariance and is denoted by \( P_0 \) at the entry point.

\[
\begin{align*}
\hat{x}_k &= A\hat{x}_{k-1} + Bu_k \\
\hat{P}_k &= AP_{k-1}A^T + Q \\
K_k &= P_k^eH^T(HP_k^eH^T + R)^{-1} \\
\hat{x}_k &= \hat{x}_k^e + K_k(z_k - H\hat{x}_k^e) \\
P_k &= (I - K_kH)P_k^e
\end{align*}
\]

The description of the above algorithm is as follows. The initial estimates for \( \hat{x}_{(k-1)} \) and \( P_{(k-1)} \) are:

\[
\begin{align*}
\hat{x}_k &= A\hat{x}_{k-1} + Bu_k \\
\hat{P}_k &= AP_{k-1}A^T + Q \\
K_k &= P_k^eH^T(HP_k^eH^T + R)^{-1} \\
\hat{x}_k &= \hat{x}_k^e + K_k(z_k - H\hat{x}_k^e) \\
P_k &= (I - K_kH)P_k^e
\end{align*}
\]

As shown in the Fig. 3, the Kalman filter estimates the current value based on the estimated value at the immediately preceding time. It cycles through these two steps: a prediction step that calculates the expected state when the input is applied, and a correction step that estimates the current state inductively using the error between the measured state and the state predicted in the previous step [15].
In the correction step, calculation was performed to correct the difference between the predicted state and the corrected state. H is a transform coefficient such as A, and R is a dispersion variable such as Q. Using these values, the Kalman gain $K$ in (4) was calculated. The Kalman gain will be used to calculate the estimated value $x_k$ in (5) and the error covariance $P_k$ in (6).

Proceeding to step (6) sequentially will repeat the recursion to equation (2) and will repeat the calculation.

c. **AHRS algorithm**

The Attitude and Heading Reference System (AHRS) algorithm is a state estimation algorithm based on Kalman filter in the Fig. 4. It is the principle of implementation of the AHRS algorithm that the direction of the sensor can be estimated by correcting the angular velocity measured by the MEMS IMU sensor with acceleration, eliminating noise such as gravity and temperature, and integrating and accumulating the angular velocity.

![The basic form of the Kalman filter algorithm](image)

**FIGURE 4. The basic form of the Kalman filter algorithm**

C. **Avatar moves in VR environment game**

The legs of a person are largely classified into thighs and thighs at the top, shins and calves at the bottom, connected by the knees and the backs. The upper leg is moved by the pelvic joint, and the lower leg is moved by the knee joint.

Although there are differences according to the flexibility of each person, the pelvic joints are usually able to rotate 180° left and right by 45° or more, and the knee joints can rotate only to the rear, and the rotation angle is more than 100°. Therefore, when recognizing the angle of the leg of a standing person, the angle of the pelvic joint and the knee joint should be sensed, and the number of cases for the state of the entire leg is at least 810,000 (180 x 45 x 100).

Walking is a complex process in which the human nerves and skeletal muscles are used as a collective process, and it is a repetitive movement in which both legs are alternately used for stance and advancement [16].

The foot is a support that allows the other foot to move freely, and the forward foot moves in the direction of movement and becomes a stepping foot while stepping on the ground. At this time, the center of gravity moves between the supporting foot and the advancing forward foot, as in Fig. 5. [17].

![The stride of a pedestrian](image)

**FIGURE 5. The stride of a pedestrian**

The equation for stride is as follows.

$$D = L \sqrt{2 \times [1 - \cos(\alpha + \beta)]}$$  \hspace{1cm} (7)

Equation 7 can be constructed by the law of cosines.

The walking distance can be obtained through the pelvic rotational angle $\alpha$ moved forward and the pelvic rotational angle $\beta$ moved forward while walking. Conversely, the pelvic rotation angle can be obtained using the stride distance.

$$\cos(\alpha + \beta) = \frac{2L^2 - D^2}{2L^2}$$  \hspace{1cm} (8)

When walking on a regular basis, the rotation angle of the knee is meaningless.

When walking on a normal occasion, a person arbitrarily moves the center of gravity by grabbing the force, whereas when walking on a treadmill, the user will be forced out of the treadmill if force will be applied to the same muscles. Therefore, in the treadmill type moving input device introduced in Section 2.1.1, the waist fixing belt is used to prevent the body from coming out.

III. **AHRS algorithm implementation and input device development**

A. **Application of AHRS algorithm**

There are various filter algorithms to implement the AHRS algorithm. In this study, the AHRS algorithm based on the Kalman filter algorithm was applied to the device fabrication. The Kalman filter algorithm for the IMU sensor was used as follows.

$$Q_{\text{angle}} = 0.001, Q_{\text{gyro}} = 0.003, R = 0.03, a_{n0} = 0, bias_{a0} = 0, dt = 1FPS$$  \hspace{1cm} (9)

$Q$ and $R$ are system variables and can be set arbitrarily. $a_{n0}$ is the user input value, $a_{n0}$ is the initial value of the estimation value. Then the acceleration measurement value acc and the angular velocity measurement value gyro are inputted and the angle is outputted. $dt$ is the time difference between the current
frame and the previous frame in the running program. The syntax to obtain 1 FPS in C# programming is as follows.

\[
dt = T_{\text{current}} - T_{\text{previous}}; \\
T_{\text{previous}} = T_{\text{current}};
\]

If the above command will be executed in the Kalman filter algorithm which continuously repeats the prediction and the correction, the \( T_{\text{previous}} \) will be stored in \( dt \) after returning to the beginning of the algorithm and reaching the prevtime again.

After the initial value is set, gyro and acc are entered as the measured values at the prediction stage where the Kalman filter starts. In the program, \( k \) means frame.

\[
an_k = an_{k-1} + dt \times (\text{gyro} - \text{bias}_{k-1})
\]

The error covariance \( P \) is expressed in the form of the following matrix.

\[
P_k = \begin{bmatrix}
\alpha_k & \beta_k \\
\gamma_k & \delta_k
\end{bmatrix}
\]

\[
\begin{align*}
\alpha_k &= \alpha_{k-1} + \delta_{k-1} dt \\
\beta_k &= \beta_{k-1} - \gamma_{k-1} dt \\
\gamma_k &= \gamma_{k-1} - \delta_{k-1} dt \\
\delta_k &= \delta_{k-1} + Q_{\text{gyro}} dt
\end{align*}
\]

\( dt \) is the time during which the previous frame to the current frame. This has values that differs in every frame. However, \( dt_k \) and \( dt_{k-1} \) are not expressed differently over time because they do not use the \( dt \) value of the previous frame. When the equation \( \gamma_k = \gamma_{k-1} - \delta_{k-1} dt \) is expanded, the equation \( \gamma_k = \gamma_{k-1} - Q_{\text{gyro}} dt^2 \) will not be applied anymore. The \( dt \) of the previous frame and the current frame are used as different values.

Having gone through two stages of prediction, the next step is to calculate the formula for the correction step. Next, the Kalman gain in the Kalman gain calculation step represents the ratio of the measured value to the estimated value. The minimum value of the Kalman gain is 0, the maximum value is 1, and the lower the value of the Kalman gain \( K \), the lower the ratio of the measured value and the higher the ratio of the measured value. That is, the closer the value of \( K \) is to 1, the closer to the actual object state the corrected value, and the closer to 0 the value of \( K \), the lower the reliability of the corrected value.

The Kalman gain is in the form of a matrix, and the program computes the Kalman gain of the value to be output and the value to be corrected in the next step.

\[
K[0] = \frac{\alpha_k}{\alpha_k + R}, \quad K[1] = \frac{\gamma_k}{\alpha_k + R}
\]

\( R \) is a covariance to the measurement noise, which is initialized when the system is started. The larger the value of \( R \) is set, the greater the proportion of the estimation.

The calculated Kalman gain is used in the next step.

\[
an_k = an_{k-1} + k[0](\text{acc} - an_{k-1}) \\
bias_k = bias_{k-1} + k[1](\text{acc} - an_{k-1})
\]

The value output through the Kalman filter is \( an_k \), and the \( bias_k \) is used to calculate the predicted value in the Kalman filter of the next frame. In addition, the error covariance \( P \) is subtracted from the formula for calculating the Kalman gain in the prediction equation calculated above, and more accurate prediction is performed in the next frame.

\[
P_k = \begin{bmatrix}
\alpha_k & \beta_k \\
\gamma_k & \delta_k
\end{bmatrix}
\]

\[
\begin{align*}
\alpha_k &= \alpha_{k-1} - K[0]\alpha_k \\
\beta_k &= \beta_{k-1} - K[0]\beta_k \\
\gamma_k &= \gamma_{k-1} - K[1]\alpha_k \\
\delta_k &= \delta_{k-1} - K[1]\beta_k
\end{align*}
\]

If the AHRS algorithm by applying the above Kalman filter algorithm will be implemented, the following flow chart(Fig. 6.) will be shown.

![Figure 6. AHRS algorithm](image)

The gyro sensor has a measurable 5V voltage reference, and the rotational angular velocity per second is from -250 to 250. The sensitivity is 131 (LSB / ° / sec). The data measured by the sensor is expressed as ± 16,375 (LSB / g) according to the following equation.

\[
\text{Raw data}(\text{LSB/g}) = G_M(°/\text{sec}) \times G_s(\text{LSB}/°/\text{sec}) \div A_M(g)
\]

In equation (16), \( G_M \) is the maximum rotational speed per second of the gyro sensor that can be measured, \( G_s \) is the gyro sensor sensitivity, and \( A_M \) is the maximum gravitational acceleration of the measurable acceleration sensor. Therefore, by dividing the raw data measured by the gyro sensor by 131, which is the sensitivity of the gyro sensor, the rotational speed of the gyroscope can be calculated.
φ and θ output from Fig. 6. is used as Pitch and Roll in input devices.

B. Application of AHRS algorithm

A chair with wheels on its legs, which can rotate around the yaw axis, is very common. These are called wheelchairs, swivel chairs, office clerks and many other names. The input device manufactured in this study is operated by walking on this rotating chair.

'Sitting and walking' refers to the action of pushing forward with a force on the leg while sitting on a wheeled chair. When using the input device, the chair moves with the appropriate force to prevent the wheel from rolling forward.

In terms of the angle of the shin, the angle when the two feet are standing straight on the ground and the angle when the leg is raised and the knee is folded at right angles like Fig. 7 are the same. That is, when the pelvic angle and the knee angle with respect to the standing posture are defined as 0 °, the angle of the shin is the same as the difference between the pelvic angle and the knee angle.

These variables can be ignored when sitting on a chair.

When sitting on a chair like Fig. 8., the pelvic angle becomes meaningless on the gait. In this condition, the sensors need to be worn only under the knee to distinguish the walking condition. The rotating chair also serves as a waist fixing bend of the treadmill input device. In 3D games as well as in virtual reality games, the rotation of views and the rotation of avatars are handled individually. The rotation of the line of sight is determined by the direction of the camera projected on the screen. In the virtual reality game, the rotation of views is processed by the head trekking technique using the gyro sensor built in the HMD. In the treadmill input device, the direction of movement is determined by the center of gravity tilted forward when walking on the front. In this paper, the center of gravity of the input device is fixed to the hip of the wearer, which recognizes the user’s rotating the chair and changes the direction of movement.

C. Configuration of input device

a. Operation settings

As described above, the angle can be ignored when walking in a sitting position on a chair. Precisely, the pelvic angle of the pitch axis can be ignored. In the chair, the pelvis is already rotated 90 ° forward, and the rotation of the pitch axis in this state is used when the sole is heard from the ground. Therefore, this angle is meaningless.

The knee joint can be rotated only to the rear of the pitch axis. As in Section 3.3, the knee angle in a straight standing position is 0 ° relative to the pitch axis. The knee angle is 90 ° when sitting on a chair. Depending on the size of the chair and the length of the leg, the knee angle may rotate by 90 ° or more, but this is not a variable. In calculating the leg angle for the input device, the state in which the knee angle is bent 90 ° as in the sitting position in Fig. 8. is defined as the minimum value of the leg angle 0 °, and the maximum value of the leg angle is defined as 90 °.

In the standing posture, the roll axis of the pelvis moves the position of the knee, but when the chair is in the sitting position, the knee is rotated in the roll axis. Actually, the knee joint does not rotate in the roll axis, but the position of the knee is fixed and the position of the foot is moved.

The input device created in this study moves the avatar by recognizing the movement of the user who is sitting on the chair and. The movement of the foot forward and inward is recognized by the IMU sensor.

Moving directions are front-back and left-right. The user advances his / her foot forward, pulls the ground down, pulls inward, and the avatar advances. On the other hand, if the user steps the foot first and then pushes forward, the avatar moves backward.

Fig. 9. is a graph of repetitive movements that the foot is put on the ground in front of the body, and then pulled toward the
body, again and again. In the leg-pushing operation (A), the speed graph has a positive value, while in the pulling operation (B), it is negative. If you move an avatar in the game based on this graph, it will alternate between forward and backward. The moving of the foot to move the avatar can be divided into four types: 1) pulling while the foot is on the ground, 2) pushing while the foot is on the ground, 3) pulling while the foot is in the air, and 4) pushing while the foot is in the air. For distinction, the FSR-402 pressure sensor was used to set the movement to operate only when the foot touches the ground.

The yellow line indicates the input value of the pressure sensor. The default value of the force is zero, but when force is applied, the magnitude is expressed as an analog value. As shown in Fig. 9, when the blue line of the angular velocity y is negative (-), it is when the user pulls the foot. Therefore, the pattern shown in red box in Fig. 10 means when the foot touches the ground and pulls. This process distinguishes between forward and backward motion.

Similarly, the left and right movement can be distinguished from the value of the angular velocity x and the pressure. The value of angular velocity x has a positive value when the leg moves from the left to the right, and a negative value when it goes from the right to the left. Therefore, this value can be used to distinguish between left and right motion.

\[ \gamma = \frac{\text{acos}(c/l)}{\pi} \times 180 \]  \hspace{1cm} (17)

However, if the chair height C exceeds the calf length L, the pressure sensor cannot operate and the input device cannot be used. However, in the scope of this study, the height of the chair and the length of the calf were not automatically recognized and the default was defined as l=50, c=45. It was specified as a variable in the program so that this value can be changed especially for situations where the environment is different.

The values of \( \phi \) and \( \theta \) from the AHRS algorithm have the same as the angle of the leg. If the angle of the legs is more than a certain angle to the front or side, it is necessary to block the change of the angular velocity in the unintended direction.

At this time, when the length from the knee to the foot is l and the height of the chair is c, the maximum angle \( \gamma \) at which the feet touch the ground is obtained as follows.

\[ \gamma = \frac{\text{acos}(c/l)}{\pi} \times 180 \]  \hspace{1cm} (17)

However, if the chair height C exceeds the calf length L, the pressure sensor cannot operate and the input device cannot be used. However, in the scope of this study, the height of the chair and the length of the calf were not automatically recognized and the default was defined as l=50, c=45. It was specified as a variable in the program so that this value can be changed especially for situations where the environment is different.

Then, considering the error range of \( \gamma \) calculated in Eq. (17) and the easy-to-walk angle, 0.75 \( \gamma \) was set, which is 1/4 angle before reaching the maximum, as the critical angle to block the change of angular velocity in unintended direction. Once the critical angle is reached, the direction of movement is maintained until the feet are raised from the ground. Also, when the angle of the leg is the maximum value that touches the ground, the sole does not touch the ground like Fig.13.

Considering this case, from 0.75 \( \gamma \) to 1 \( \gamma \), the pressure sensor was set to be movable even if no force acts on the pressure sensor.
The overall operation setting is shown in Fig. 14. The black box can be operated when the pressure device is depressed, and the operation outside the box is possible regardless of the input device.

b. Setting travel distance

The input device manufactured in this study has shorter foot travel distance than the existing treadmill method. The treadmill input device has a longer travel distance of the foot and longer time to walk than moving while standing [20]. On the other hand, the input device of the present invention which is used by sitting sideways has a short moving distance of the foot used for the operation, and therefore, the operation time is short.

When creating a new game, the moving speed of the avatar may be set to match the input device. However, when the game is combined with an existing game, the moving speed is not changed, and the moving distance is determined according to the operation time. That is, the input device of the present invention has a problem that the moving distance is short because the walking operation time is short, and the moving distance as much as the treadmill input device can be moved when the operation is repeated frequently. To solve this problem, the moving speed of the input was modified so that the operation can be done for a long time.

The condition shown in Fig. 14 is the angular speed input to the movement operation, which is equal to the movement speed of the foot. The formula for the stride in the standing state is as described in Section 2.3. You can use the stride formula to calculate the stride in the sitting position. Based on Fig. 15, the length from foot to knee and length to pelvis were assumed to be 1:2.

\[ L \approx 2l \]  \hspace{2cm} (18)

\[ d = l\sqrt{2} \times [1 - \cos \gamma] \]  \hspace{2cm} (19)

\[ l \] is the length from the knee to the foot, and \[ \gamma \] is the angle of the leg in the sitting position. When the angle of \[ \gamma \] defined in Section 3.4.2 is substituted, the equation for the sitting posture \[ d \] is developed as follows.

\[ d = \frac{1}{2} \sqrt{2} \times [1 - \cos(\cos(c/l)) \times 180 \div \pi] \]  \hspace{2cm} (20)

In this paper, the length of the shin and the height of the chair were defined as 0.5m and 0.45m, respectively. When this value is substituted, the value of \[ d \] is about 43.58, so the stride in sitting position is 0.217m.

The pedestal diameter of a conventional treadmill input device shown in the Fig. 16. that is commercialized in the past is 55 inches, and the stride used for walking is about 42% and \( D \) is 0.59 m.
The average walking speed of a person is 5 km/h. When the user is walking on the treadmill input device, the stride length \( D \) becomes \( 1.38 \times s \) if the input is possible by the average walking speed. According to the equation, \( D = 0.59 \), so that the movement operation time \( S \) is 0.42 seconds. In order to operate 0.43 m in the sitting posture for 0.42 seconds, the foot should be moved at 1.02 m/sec. The value of the angular velocity \( \delta \) can be obtained by using this value.

\[
\cos \delta (°/sec) = \frac{2(50)^2 - 21.7^2}{2(50)^2} = 0.905
\]

\[
\therefore \delta = 25.07 \quad (21)
\]

Therefore, the angular velocity of the legs to be 5 km/s in the sitting position is about 25 (°/sec), and it is set to be recognized by the operation up to 21.6 (°/sec) which is 14% slower.

In addition, in 3D games released today, including virtual reality games, walking and running are separately implemented as shown in Fig. 17.

![FIGURE 17. Walking and Running in Avatar](image)

The angular velocity of the leg for running the avatar in the sitting position is calculated through the same process as above. The running speed was set at a minimum speed of 3.01 m/s, which is less than 35 years of age, based on the fitness test chart used in the year 2015. The operating speed at 3.01 m/s is 0.196 sec and the foot must move at 2.19 m/s. The angular velocity \( \delta \) calculated based on the above velocity is 36.23. If the angular velocity of the leg is 21.6 (°/sec) or more, the avatar will walk, and if the angular velocity is 36.23 (°/sec) or more, the avatar will run.

To measure the above data, the input device shown in the Fig. 18 is configured.

![FIGURE 18. Configuration of input device](image)

The experimental procedure is shown in Fig. 19. As shown in Fig. 19., prior to the experiment, a test game was created using the Unity3D physics engine to easily analyze the state of the avatar for the experiment. This game can move the avatar by typing the keyboard ‘w’, ‘s’, ‘a’, ‘d’. According to the angular speed of the leg, it moves by 1.0 (m/s) when the avatar is walking, and 3.0 (m/s) when the avatar is running. The number of movements of the avatar is counted each time it is manipulated. Also, when operating the input device manufactured in this study, the height of the user, the height of the chair, and the keys necessary for moving the avatar in the test game are set to be inputted in the software.

In the experiment, 'HTC Vive' was used as the HMD, and the input device was rented the day before the experiment so that the subjects could fully familiarize themselves with the input device and the HMD. The number of actual leg movements was compared to the number of times the avatar was operated when the subjects wore the input device and repeated the operation [21], and the distance traveled by the avatar was measured to analyze the decrease in travel speed with time. The number of times the subject actually moved the leg was recorded by the experimenter. The subject repeated the motion for 10 minutes.
each for forward, backward, left, and rightward movements. They took a 10-minute break after each action. Variables related to physical fitness were excluded from the study.

C. Recognition rate test result and analysis

The software and game interface used in the recognition rate test are shown in Fig. 20. And Fig. 21. The maximum leg angle and the angular velocity at walking / running were calculated for each subject according to the formula summarized in Chapter 3 by inputting the leg length and chair height. The test game shows the distance the avatar has moved, and the software is designed to stop 10 minutes after the start of the experiment.

When the subject repeated the operation of the input device for 10 minutes for the recognition rate test experiment, the following results (Fig. 22–23, Table 2–3) were obtained for each subject. In the graph, the left line shows the actual number of times the user moved the leg for the operation, and the right line shows the number of times the leg motion was recognized and operated. The recognition rate is expressed as a percentage by dividing the actual operation frequency and the recognition frequency.

<table>
<thead>
<tr>
<th>Subject</th>
<th>Forward</th>
<th>Backward</th>
<th>Left side</th>
<th>Right side</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>91.85</td>
<td>95.03</td>
<td>81.23</td>
<td>85.80</td>
</tr>
<tr>
<td>B</td>
<td>95.02</td>
<td>97.26</td>
<td>87.41</td>
<td>86.00</td>
</tr>
<tr>
<td>C</td>
<td>97.45</td>
<td>99.33</td>
<td>89.90</td>
<td>89.31</td>
</tr>
<tr>
<td>D</td>
<td>97.24</td>
<td>98.34</td>
<td>78.26</td>
<td>83.96</td>
</tr>
<tr>
<td>E</td>
<td>96.96</td>
<td>98.29</td>
<td>79.60</td>
<td>77.83</td>
</tr>
</tbody>
</table>

![FIGURE 20. Software interface](image)

![FIGURE 21. Game for test](image)

![FIGURE 22. Actual number of operations and Number of times recognized as input of subjects](image)
Consistent results shown in the graphs are that recognition rates for front and rear are 95.70% and 97.65%, respectively, and recognition rates for left and right are 83.28% and 84.58%, respectively.

When the avatar is walked by the manipulation of the subject, it moves by 1.0 meter per second in the moving direction and by 3.0 meter per second when the avatar is running. The distance traveled by each subject is as follows.

The recognition rate of left and right motion is lower than the recognition rate of backward and forward motion because it is difficult to sit and move the pelvis. Table 3 also shows that for a 10-minute distance, the front-backward movement moved farther than the left-right movement. This is considered to be a limitation of low-cost sensors and low performance microprocessor.

### Table 3. 10 minutes moving distance by direction

<table>
<thead>
<tr>
<th>Subject</th>
<th>Forward</th>
<th>Backward</th>
<th>Left side</th>
<th>Right side</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subject A</td>
<td>146</td>
<td>130</td>
<td>87</td>
<td>92</td>
</tr>
<tr>
<td>Subject B</td>
<td>156</td>
<td>149</td>
<td>110</td>
<td>93</td>
</tr>
<tr>
<td>Subject C</td>
<td>149</td>
<td>137</td>
<td>87</td>
<td>86</td>
</tr>
</tbody>
</table>

### V. Conclusion

In this paper, a motion recognition remote sensing input device that can be used in a sitting position using a wheeled swivel chair was developed and evaluated.

As game developers traditionally develop their games, they have used input devices that fit their development platform. However, there is no standardized hardware or software in the field of virtual reality games. Therefore, when planning development for virtual reality game, it is necessary to select whether to move the avatar along with the game genre or how to move it.

There are a lot of hand controller devices as an input device for virtual reality environment. Virtual reality game developers use input methods to simulate the behavior of the game environment in real situations using various hand controller devices [22][23]. However, in moving the avatar, the device recognizing the motion is very large, and it is sold at a high price.
Therefore, there will be very little demand to supply games to only a handful of users who own such equipment.

This study suggests that game creators can plan more various virtual games by allowing users to move freely rather than moving within a limited range. Furthermore, it can also be used as a relatively inexpensive remote sensing device capable of measuring the joint range of a handicapped person or low-growth children.

Acquiring patient details by traditional methods has various constraints because of overhead and inconvenience to wear [24-25]. Because remote sensing is necessary to overcome this problem, the proposed method will contribute to many areas related to healthcare. In addition, optimization by mobile distance minimization algorithm [26] or pattern matching using artificial intelligence and genetic algorithms [27] will lead to better results.

In order to make the remote sensing device developed in this paper, it is not necessary to purchase expensive equipment, nor to install a huge device. Compared to existing treadmill input devices, this device still lacks recognition rate and ergonomic design research. The recognition rate of lateral movement is especially low, so it is necessary to study the angle of the pelvic joint and to practice the user. However, by attaching the sensoring input device to various parts of humans, there are advantages that the usable range and speed of large joints of the human upper and lower body can be measured.

In many existing virtual reality games, users move within a limited range as they are based on the functionality of the HMD with the largest share. If a variety of mobile input devices are developed in addition to the input device in this study, it is expected that the unlimited range of motion can be freely registered in the virtual reality game and it can be utilized as various healthcare solution by increasing recognition rate and miniaturization.

REFERENCES


J.Y. Kim was born in Seoul, South Korea in 1979. He received the B.S and M.S. degree in Game engineering from Hoseo University in 2002 and 2006. He received Ph.D. degree in Game engineering from graduate school of advanced imaging science, multimedia & film in Chung-Ang University in 2013. From 2004 to 2005, his game industry experience started as game designer for online casual game. From 2006 also he started to teach game design at Chung-Ang University game specialized school. From 2009 to 2014, he was CEO of Nextgames and also leading project as well. From 2009 to 2013, he was professor of Chungkang College of cultural industries. He is an assistant professor in Graduate School of Game Gachon University also he is currently professor of Gachon University and also center director of Start-up Education Center at Gachon University. Additionally he was former vice president of KGDA (Korea Game Developer Associations) from 2015 to Apr, 2018. He is currently serving Editor-in-chief for Korea Computer Game Association since 2016. Also he published many researching journal and books. His research area is about IT and published subject which are technique of Computer Game, AI, Virtual Reality Technology and Interactive Technology.

W. H. Lee was born in Pohang City, South Korea in 1957. He received the B.S. and M.S. degree in college of fishery from National Bu- Kyung University in 1981, and 1983. And also received M.S. and Ph.D. in college of law degree from Kook min University. In 1995 and 1998. He researched ultrasound physics at Tokyo Institute of Technology during 1983-1984 and also ultrasound medical device at University of New South Wales as Ph.D. course student and Australia National Ultrasonic Institute as part-time engineer at same time during 1984-1987. He was vice director of computer center and also supercomputer center at Chung-Ang University. And he is professor at Graduate School G.S.A.I.M. of Chung-Ang University and also president of Korean Society for Computer Game. His research interests are AI for computer game, game technology and information law.