A study of an implementation of the kinesthetic feedback on the game framework applying the haptic

Hye-Young Kim¹, Jongwan Kim²

¹ Department of Game Software, School of Games, Hongik University, Jochoiwon-eup, Chungcheongnam-do, 339-701, Korea
hykim@hongik.ac.kr

² Department of Computer Engineering, Sungkyul University, 53 Sungkyuldaehakro, Manan-gu, Anyang, 430-742, Korea
kimj@sungkyul.edu

Abstract. In this paper, we proposed a game framework that applied the haptic device to the game for efficient conveyance of the tactile sensation and it is based on the input and output in the 3D. Also, we propose the kinesthetic feedback on the game framework and implement a realistic game in serious game using this proposed framework. Then we verify the efficiency of ours framework by exam the satisfaction of the user in this game.

Keywords: haptic, interaction, feedback, game framework

1 Introduction

In PC games, user interfaces comprised of a mouse, a keyboard, and a joystick have the disadvantage that there is a lack of satisfaction from tactile experiences. On the other hand, user demands for a realistic experience based on the characteristics or the sensation of the material have increased rapidly. Therefore there is a need for a third device for a realistic simulation of the interactions between computers and the users.[1]

In this paper, we proposed a game framework that applied the haptic device to the game for efficient conveyance of the tactile sensation and it is based on the input and output in the 3D. Also, we propose the kinesthetic feedback on the game framework and implement a realistic game in serious game using this proposed framework. Then we verify the efficiency of ours framework by exam the satisfaction of the user in this game.

The paper is structured as the following: Section 2 reviews related works on the technical knowledge and theory about haptic control. The implemented scheme is described in Section 3. In Section 4, we test and analyze the performance of the implementation of the game engine that has been applied with the proposed scheme. Finally, we summarize the paper in Section 5.

2 Correspondent author: Jongwan Kim

ISSN: 2287-1233 ASTL
Copyright © 2015 SERSC
2 Related Work

A haptic device is convey information to the user using the double feedback force and the interaction between the user and the haptic device is a symmetric and bidirectional it. [2][3]

The Fig 1 represents the interaction between the haptic devices and users.

![Fig. 1. Interaction between user and haptic device](image)

We have used the Haptic SDK (dhd-3.4.0) for retrieving the information about haptic for user can receive the information which has given the strength information using three motors tactile sensation.

The haptic technology has been studied about how feel the nearest to reality by departmentalized the tactile sensation and how fast within a short time receive the feedback process [4].

Therefore, we proposed a gaming framework applying the haptic device for realistic game and then implement an interactive game through efficiently convey the sensory feedback and describe about the part of tactile sensational effect when the collision has been occurred in the game.

3 Implementation

3.1 Framework

We propose the efficient scheme for applying the haptic device to game using the OpenFrameworks on the OpenGL. For applying the haptic device on the game framework, we need separated threads. They have to receive the coordinates of the haptic devices independent arithmetic in the main thread, and for that, we use the thread object of the OpenFrameworks.

There are variables for independent coordinates which has been received the haptic handle in the class, On /Off switch for haptic device, haptic device number, and timer variable which is for the measure about interaction time between the game and haptic device.

```java
class hapticThread : public ofThread
```

```java
int hapticNumber;
```

```java
void setup()
{
    hapticNumber = 0;
    // Other setup code
}
```

```java
void update()
{
    // Update haptic device
    if (hapticNumber == 0)
    {
        // Haptic device 1
    }
    else if (hapticNumber == 1)
    {
        // Haptic device 2
    }
    // Other update code
}
```

```java
void draw()
{
    // Draw haptic device
    if (hapticNumber == 0)
    {
        // Draw haptic device 1
    }
    else if (hapticNumber == 1)
    {
        // Draw haptic device 2
    }
    // Other draw code
}
```
The haptic thread is started as soon as the object has been created in the main function and it has been received the coordination constantly when the haptic has been handled. The below shows how haptic coordinate values are processed to apply to the game. In the main loop, we consider that the trajectory of sphere has been moving and avoiding some objects and process of the procedures. The spheres are 3D object and the remains consisted of planes.[5]

The entire object in the game has created each of the method in the of3dprimitives class, this class is established world coordinate and local coordinate of inherit of the class, ofnode.

When the collision between the objects in the game is detected, the feedback is processed by call the Collision function which is independent function in the haptic class.

3.2 Haptic Function

To implement the game, we use the functions as Table 1

Table 1. Used SDK Functions

<table>
<thead>
<tr>
<th>Type</th>
<th>Function name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int_SDK</td>
<td>dhdEnableForce(uchar val, char ID=-1)</td>
<td>Receive device id</td>
</tr>
<tr>
<td>Int_SDK</td>
<td>dhdGetPosition(double *px, double *py, double *pz, char ID=-1)</td>
<td>Get the coordinate of device handle</td>
</tr>
<tr>
<td>Int_SDK</td>
<td>dhdSetForce(double fx, double fy, double fz, char ID=-1)</td>
<td>Apply the device to calculated strength</td>
</tr>
<tr>
<td>Int_SDK</td>
<td>dhdSetForceAndTorqueAndGripperTorque(double fx, double fy, double fz, double ta, double tb, double tg, double t, char ID=-1)</td>
<td>Apply the strength and acceleration</td>
</tr>
</tbody>
</table>

The following functions are used for the re-replace method for each function within the class of HapticThread as Table 1.

4 Performance Analysis

We compared the rate of reaction of our game framework, applied with the haptic device, with that of without our game framework. We compared the time of reaction speed of two university students because the reaction speed can be different individually. We tested university students A, B to exam the reaction speed of applied the haptic device with our game framework and without our game framework and it is shown in the Table 2.

Table 2. Interaction measurement

<table>
<thead>
<tr>
<th>Button/Student</th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Haptic button</td>
<td>0.00488281</td>
<td>0.00585938</td>
</tr>
</tbody>
</table>
The measure method of reaction speed is the difference time that the player A and B press the button of haptic device or keyboard when the character of player appeared after the game start. For the objectively measurement, we print the “Collision!” message in the separated log window when the obstacles are touched, and the user press the key when the use the keyboard.

For the objectively measurement, we design that print the “Collision!” message in the separated log window when the obstacles are touched for haptic device user but user press any key by look at the message when the user use the keyboard. It is not necessary to see such a message because it receives feedback by the device itself in the haptic device. In the result of the measurement, it is faster reaction time when the user plays using the haptic device than when the user depends on the sight.

5 Conclusion

We propose a game framework that applies the haptic device to the game which is to efficiently convey the tactile sensation, is based on the input and output of the 3D and the kinesthetic feedback on the game framework, and implement a realistic game in serious gaming using this proposed framework. Then we verify the efficiency of our framework by examining the satisfaction of the user in this game. We compared the rate of reaction of our game framework, applied with the haptic device, with that of without our game framework. In the result of the measurement, there is a faster reaction time when the user plays using the haptic device than when the user plays depending on the sight. Therefore we conjecture that our framework is better applied to an FPS (First Person Shooter) game due to the fast response required by FPS games.
Acknowledgments. This work was supported by the National Research Foundation of Korea Grant funded by the Korean Government [NRF-2014R1A1A2059902]. This work was supported by 2014 Hongik University Research Fund.

References