

Virtual Skiing

Tomazic, Tomaz; Lai Chin Chung, Louis

University of Ljubljana, Faculty of informatics and computer science

Abstract

This paper describes a virtual skiing game using kinect and unity 3D.

Keywords: skiing, kinect, unity 3D, virtual

1. Introduction

The interactive installation "Virtual skiing" enables a visual immersion into the feelings of gliding on snow through a winter landscape. The computer rendered winter landscape is displayed over the entire wall in front of the skier. As on real skis you can regulate the speed of descent by changing the posture of your body so that the air resistance is decreased or increased. By shifting the weight of your body to the right or left ski you can make turns down the slope between the snow capped trees. The interface to the virtual world is implemented by computer vision techniques which capture the posture of the skier's body. [1]).

2. Software overview

2.1. Goal of the project

The goal of the project is to provide the user a virtual skiing experience. To do so, at least the following needs to be done:

- (i) track body movement of the user
- (ii) display a virtual 3D environment on a screen
- (iii) play game in this environment

2.2. Kinect

Overview. Kinect is the most convenient body movement tracking hardware we can find in the market, it thus becomes the tool for our project. The Kinect sensor has one RGB camera and two Infra-red sensors, allowing it to track movement to/from any direction in a 3D space.

Code. By examining the sample code, a skeleton of user is made using C#. A skeleton consists of nodes and lines, nodes representing the joints of the user, lines representing the limbs/body of the user. By comparing the positions of nodes, we can capture the user's movement and turn them into inputs in the

game. There are 2 main types of input in the skiing game, one controls the direction and the other controls the speed of skiing.

Directions. To control the direction of ski, user needs to bend their body left or right to turn, or stay in the middle to go straight. A simple method is used to extract this particular movement by the user, we simply compare the nodes on both shoulders. The data obtained is then thresholded into only 3 inputs, left, middle and right.

Speed. The user needs to squat to increase the speed, and detection of squatting is needed. When squatting, the position of the node on hip is changing obviously. So, the node on hip is used to determine whether the user is squatting or not.

2.3. Unity 3D

Unity is a development engine for the creation of 2D and 3D games and interactive content. It has multiplatform support and its own asset store where you can download scripts, objects, scenes.. [2]

Terrain. It has great support for creating terrain. In game we created simple terrain with mountains and put snow on it. For more flexible terrain we could create script or just few images of pre-defined heightmaps. Unity has option to generate terrain using heightmap.

Scripting. Unity supports C#, javascript and Boo scripting languages. It also supports applying different language script on same object. We used such ability for user movement in skiing game, using javascript and C# languages only.

3. Developing process

3.1. Planning

Mindmap. A mindmap is made to record all the ideas about the project. It gives the team some guide and direction to work on. This is also the step where we

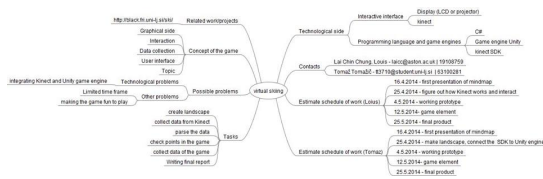


Figure 1. Mindmap

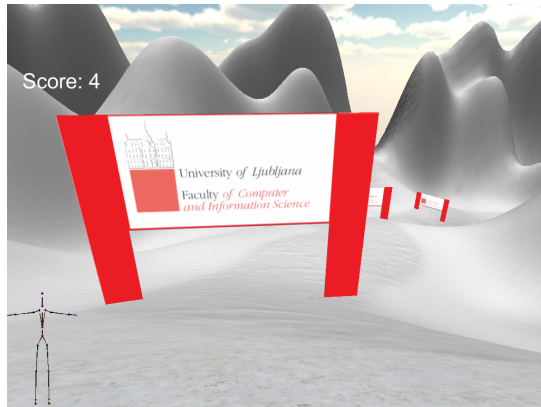


Figure 2. Screenshot from skiing game.

know each team member's strength and weakness, leads to better allocation of work later on.

3.2. Research

Kinect. At the beginning of the project, none of the team member knows anything about programming on

Kinect, it is the main focus of search. We found that Kinect SDK is essential in coding the sensor, it comes together with Kinect studio and a developer toolkit. Kinect studio is just for testing and the developer toolkit is a package of code examples.

Game engine. It took us almost no time to search for game engine, one of the team member knows how to use Unity 3D, certainly this becomes your choice of game engine.

Designing the system.

3.3. Work on the project

3.4. Replanning

3.5. Ending

4. Conclusions

4.1. Schedule (planned vs read)

a

sub

References

- [1] SOLINA FRANC and BATAGELJ BORUT (2005) <http://black.fri.uni-lj.si/ski/>
- [2] UNITY 3D (2014) *L^AT_EX*: <http://docs.unity3d.com/ScriptReference/index.html>
- [3] MITTELBACH F. and GOOSSENS M (2004) *The L^AT_EX Companion* (Addison-Wesley), 2nd ed.