Project Documentation

Project Title :- Hole in the Wall

Team Name :- Blink

Team Members :- Ravi Kumar, Sai Sravan , Shubham Yadav , Ashish Kumar

Team Mentors :- Atri bhattacharya , Anirudh agarwal

Aim Of The Project:

To create a game " HOLE IN THE WALL " in which human posture on the wall is matched with the pose of player in front of kinect camera by taking skeleton input obtained by kinect.

Motivation:

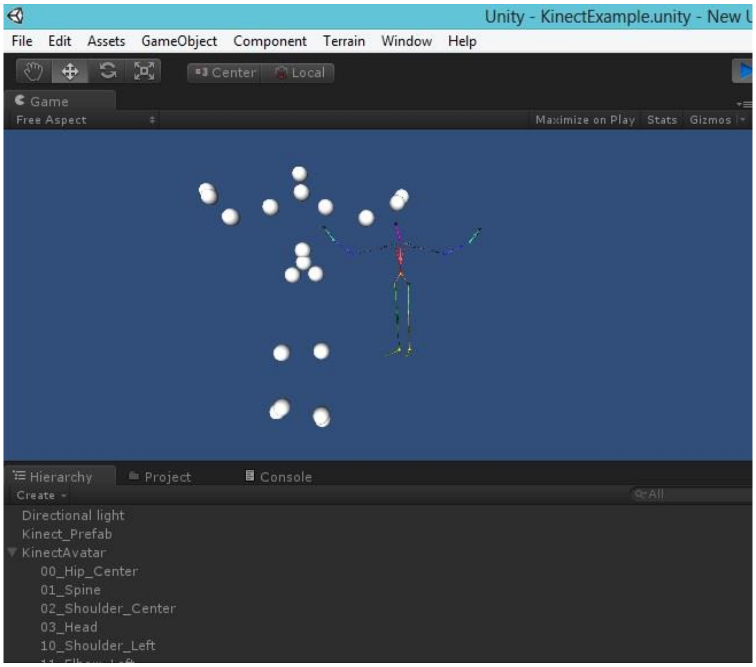
Motivated from the television show "HOLE IN THE WALL" . These days people spent much time on playing computer games because of this there arises many health and fitness related problems . We decided to create a game that replaces computer games and also helps in physical fitness and entertainment.

Theory :

There are two elements to the Kinect depth camera. The first is the Infra-Red projector that projects a field of dots onto the scene in front of the sensor bar. The second element is an Infra-Red camera that views the dots in the scene. The position of the dots in the scene the camera views depends on the position in the scene from which they were reflected.

Above you can see two images of the same scene. The upper image shows what the Kinect video camera sees. In lower part you can see the pattern of dots that the Infra-Red camera sees. Some of the dots are bright “alignment” dots; others are used for more precise positioning.

Kinect Skeleton Tracking **:**



Directional light
Kinect_Prefab
KinectAvatar
00_Hip_Center
01_Spine
02_Shoulder_Center
03_Head
10_Shoulder_Left

Kinect SDK :

A program can use the depth information from the sensor to detect and track the shape of the human body. The Kinect SDK will do this for our programs and provide skeletal position information that can be used in games and other program.

The skeletal tracking in the Kinect SDK can track six skeletons at the same time. For four of the bodies only simple location is provided but two will be tracked in detail. For those two bodies the SDK will provide the position in 3D space of 20 joint positions. The software will infer the position of joints if it is not able to view their position exactly, for example of the user is wearing clothing such as a skirt that obscures their limbs.

Software can be developed on Windows PC that uses the Kinect sensor by installing the

Kinect for Windows Software Development Kit (SDK). It contains drivers for both managed

(C# and Visual Basic .NET) and unmanaged (C++) applications that want to use the sensor.

Event handlers can be defined for events such as ColorFrameReady, DepthFrameReady,SkeletonFrameReady, and AllFramesReady ,for functions which are executed when the program gets new input from the color camera and depth camera.One can also get Data on the depth of every pixel,and whether a human player is present on that pixel or not.

One can specify various parameters for skeletal tracking as follows:

```
//Code Starts

kinectSensor.SkeletonStream.Enable (new TransformSmoothParameters ()//EDIT

SMOOTHING PARAMETERS OF SKELETON STREAM

{

Smoothing = 0.5f,

Correction = 0.5f,

Prediction = 0.5f,

JitterRadius = 0.05f,

MaxDeviationRadius = 0.03f

});

//These are slightly different from the default values.
```

The image frame from the camera is rendered as a 2D texture and displayed onto the screen.

Our first step

UNITY 3D :

.

our game needs better user processor interface hence we started to look for a game engine that best suits our needs.We used Unity 3D game engine which makes the task of making Video games, and in our case 2D and 3D objects quite easy.It can be coded in Visual C# Express and integrated with our Kinect functions. This is very convenient for us and we decided to pursue our project along these lines.We add the necessary references regarding Kinect SDK and XNA to the project to use predefinedclasses.

we decided to use unity which supports best user interface.we started to attach the game objects needed for our game .the basic idea of our gameplay was sorted out. we tried a basic model that can be controled by kinect and also analysed how to smoothen and neglect some of its random movements.

kinect model controlling:-

we imported a basic model with various joints and then attached the ones to be controlled by the kinect.we wrote a program that acts as an interface between the kinect and the model .

We then imported a complex model and attached the same program but the model started to show irregular motions at its lower body parts.Hence we wrote a program that smoothens these motions.

This model is rendered transparent for best view of the wall and the model.

comparison of poses :

we thought of various ideas or algorithms for comparing two poses.hence we sorted out the below algorithms.

1.Area overlapping

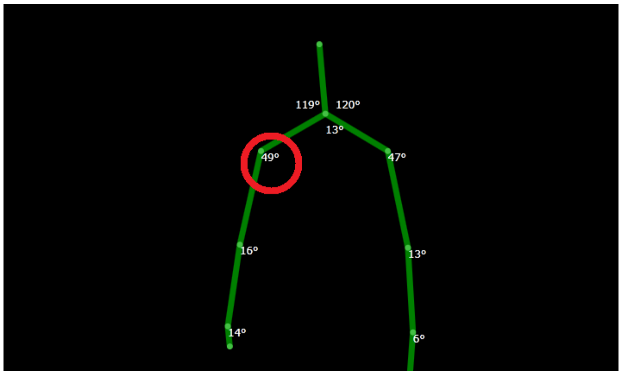
2.Three joint angle

AREA OVERLAPPING METHOD:-

In this method we overlap the areas of the two poses and compare the overlapped area with the area of the original pose.we calculate the areas by trapezoid method

which works the same way as a mathematical integration.But this method posed some controversies as it neglects the relative joint positions and only concentrates on the area in common.

Three joint angle:-



In this method we store the array of angle values of the original pose and compare them with those obtained from the current pose.This method was found to be ideal for the comparisons

.Hence we came upon to use this method.

Creating Diffenrent poses :

To create a human poses we used “ADOBE PHOTOSHOP”.

MAKING GAME PLAY:-

we added the walls and different lighting components to our game screen.The kinect model controlling program was keeping the bodycentre (hipcentre in our case)

at a fixed position so we modified it for rendering the model movable.

KINECT shows large errors when the joints are too near to be confused for wrong detection.Hence we collected some poses from previous made   that cause best detection and then added these on to the walls.

PROGRAMMING :-

A gamecontroller script is the main cpu of our game ,it collects all the data ,processes it and then displays the result.This script is made frame rate independent

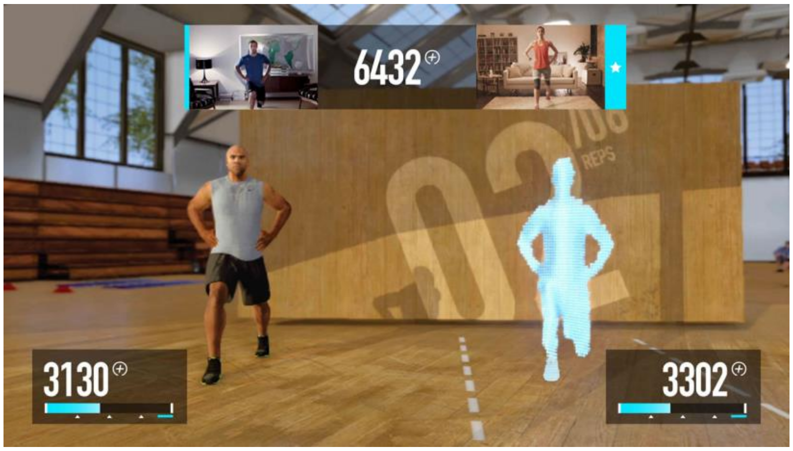
for uniform game play.GUI DISPLAY are used for score and other display.

Hand gesture recognisation:-

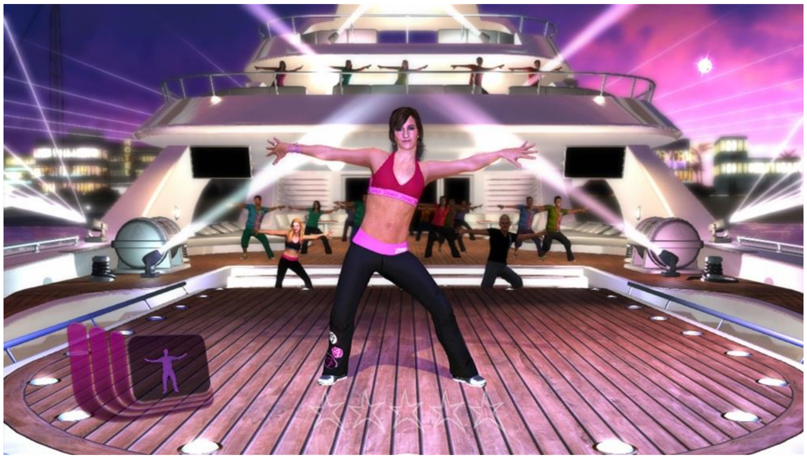
We programmed a pointer which is controlled by the movement of either hands. PLAY/PAUSE with hand motion recognition feature was also added to the game.

Its application and utility :

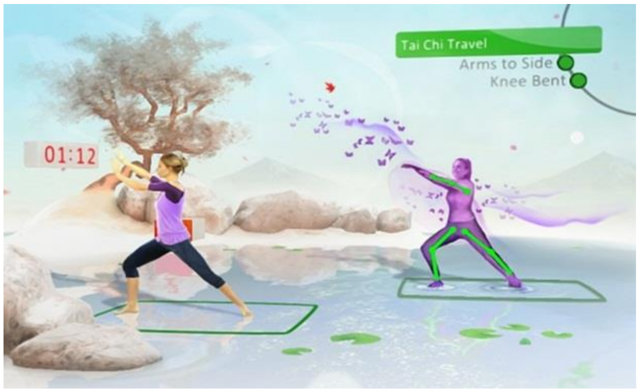
1. As physical trainer



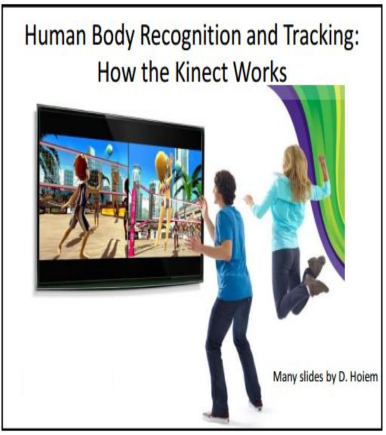
2. Dance instructer



3. Yoga instructor



4. Gaming :


Human Body Recognition and Tracking:
How the Kinect Works

Many slides by D. Hoiem

Further development And Future Scope:-

1. Game control through voice commands.

2. Making the game 3D pose supportive.

3. Rendering the game multi player supportive.

A word of thanks :

We would like to thank our mentors Atri bhattacharya and Anirudh agarwal along with Shivendu Bhushan , Sonu Agarwal and Swapnil Upadhyay for their guidance, patience, suggestions and their believe in us .
They inspired us to work on this exciting project and checked the progress of our project so sincerely that even if , sometimes, when we were like, frustrated and hopeless regarding our project they showed us the way.
We actually learned a new programming language and a set of new tools in it for creating exciting games.