ADVANCED WHACK A MOLE VR

Tal Pilo, Or Gitli and Mirit Alush

TABLE OF CONTENTS

Introduction

We developed a VR game based on the common arcade game "Whack a mole". The game includes two levels:

- The first one is a classic version of the common game "Whack A Mole", well known from the 90's arcade rooms. The impact is with a hammer that appears when the player make a fist with his right hand.
- The second one is an advanced version of "Whack A Mole" includes horizontal view of about 120 degrees. The player stand in front of seven pillars, and ghosts come throw from them. The goal is to hit the ghosts with his fists.

The both levels are against the clock, and the goal is to hit as many moles or ghosts as possible.

The thought behind creating two different levels was to give the user an opportunity to experience the VR in several ways and in an optimal manner.

Development Environment

We developed the project using Unity 2017.03 – graphic engine, scripted with C# in Visual Studio 2017.

Unity is a cross- platform game engine developed by Unity Technologies.

Unity gives users the ability to create games in both 2D and 3D, and the engine offers a primary scripting API in C#, for both the Unity editor in the form of plugins, and games themselves, as well as drag and drop functionality.

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms.

The HTC Vive is a virtual reality headset developed by HTC and Valve Corporation. The headset uses "room scale" tracking technology, allowing the user to move in 3D space and use motion-tracked handheld controllers to interact with the environment

Manus VR is the first virtual reality glove Input Device created specifically for general consumers.

Manus VR users an assortment of sensors to track hand movement in real time and use the captured data to faithfully reproduced the movement in virtual reality. It operates completely wirelessly and comes with an open-source SDK that developers can use to integrate the hand-tracking functionality into their applications and games.

Some relevant links:

https://www.unity3d.com

https://www.vive.com/us/product/vive-virtual-reality-system

https://manus-vr.com/hardware

Application Overview

The application consist of three scenes for the user to navigate:

1. Menu
2. Classic scene
3. Advanced scene

1. Menu

The main menu scene allow the player to choose between two possible levels and read the instructions for each level. Each level design and gameplay is different. The main menu environment designed with elements that reminds the classic arcade. Each choice in the menu has the feature of the level gameplay, that means the first level choice option is design as a mole coming from a hole and the second level choice is design as a pillar that coming out from the ground with a ghost standing next to the pillar. The instruction allow the player to understand both of the games rules. It explains about the scoring system, time and every secondary features you can use in the second level. It also explains technical use of the HTC VIVE, Manus VR gloves and trackers. The player can navigate through the menu options with laser beam that comes out of the Manus VR right glove. The beam is created when making a fist with the glove, it points to the wanted direction.
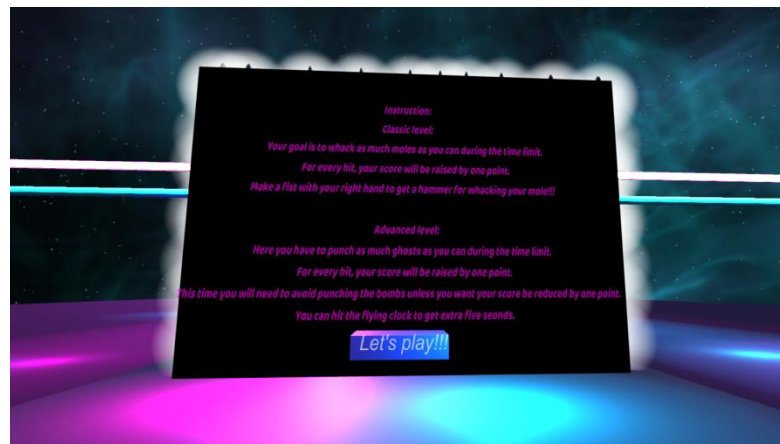
2. First level- classic whack a mole

The first level was created to simulate "Whack a mole" gameplay in a virtual retro arcade. We have based our designed on some photos we got from "Google images" so that the experience will be realistic as possible. The main feature in the first level were created in "Blender", an open-source 3D computer graphics software. We used it to make the moles, room and the "Whack a mole" arcade machine which contains the moles. At first the player stand in front the "Whack a mole" game, in the middle of an arcade. After a brief countdown, the moles are starting to come out randomly from the nine holes that on top of the game machine. At first the moles will come out in a slow pace and the interval between each mole that come out is quite long. As the time decrease, the more often the moles will comes out and much faster. The player goal is to hit with the hammer as much as moles he can within the time limit. Whenever the player flattened his hand, he see the Manus VR gloves in their default design. By making a fist, the player replace his glove with a hammer, with this hammer you can achieve point by hitting the moles whenever they comes out. As the countdown reaches to zero, the moles stop to come out and the score is displayed on screen.
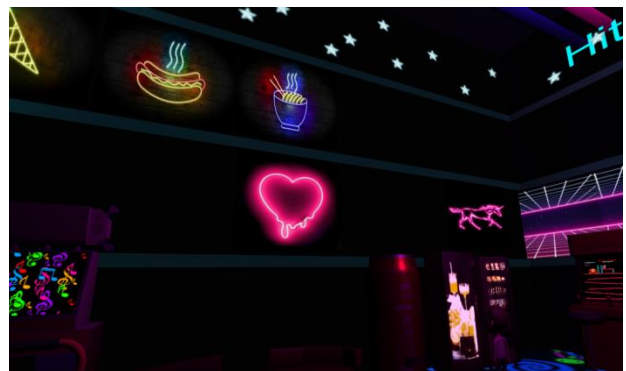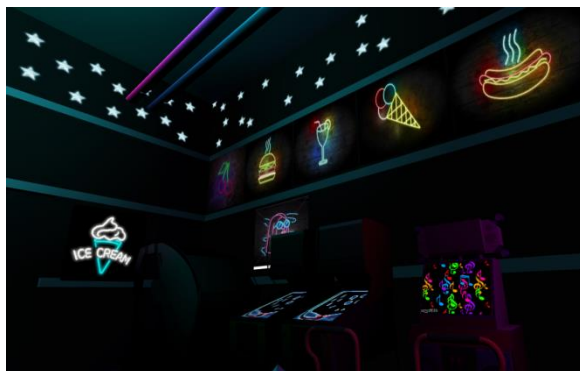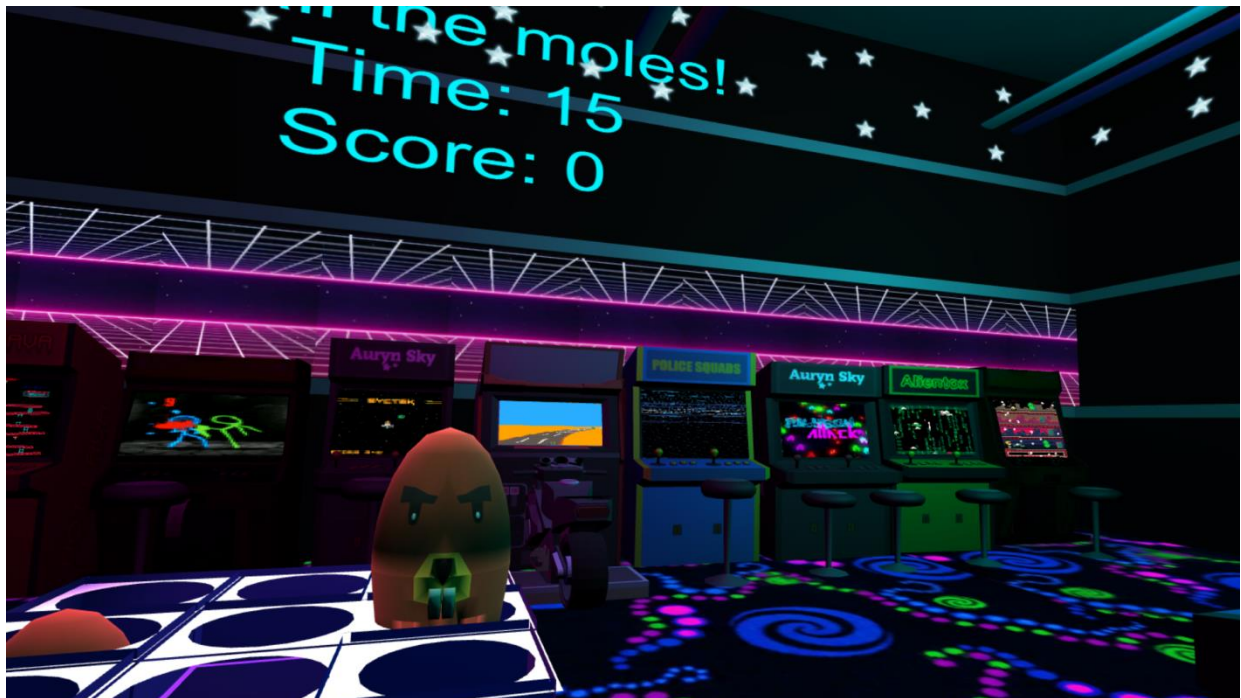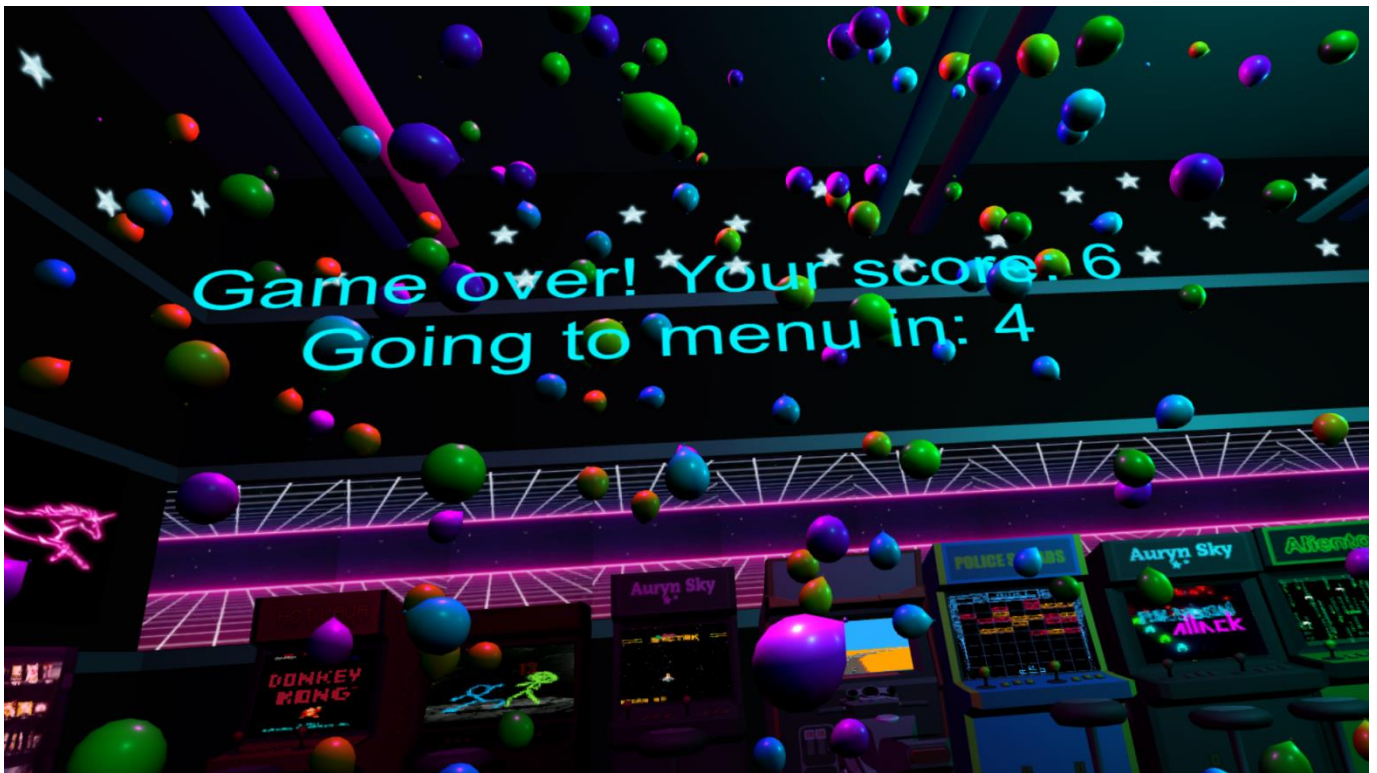
3. Second level- advanced

The second level idea derived from the first level but spookier and much more VR oriented. At first, the room is similar to the first design, then the room become like a room from horror movie, with a lot of frightening features. It start with the frightening room design change, in this phase the player has no control at all. When the introduction is over, the countdown start and the game begin. In this level, opposed to the classic one, we have control on both of the gloves. The room has seven pillars, each pillar has three holes. Ghosts come out randomly from the holes toward the player direction. The player need to punch the ghost as much as possible within the time limit and each hit grant him one point. At random times bombs and clocks are coming out from the holes. The player need to ignore the bombs- if he hit one, 3 points is reduced from the score. If the player hit a clock, 5 seconds is added to the time limit. In the end, as the classic level, the player goal is to get as much point as he can until the countdown reaches to zero.
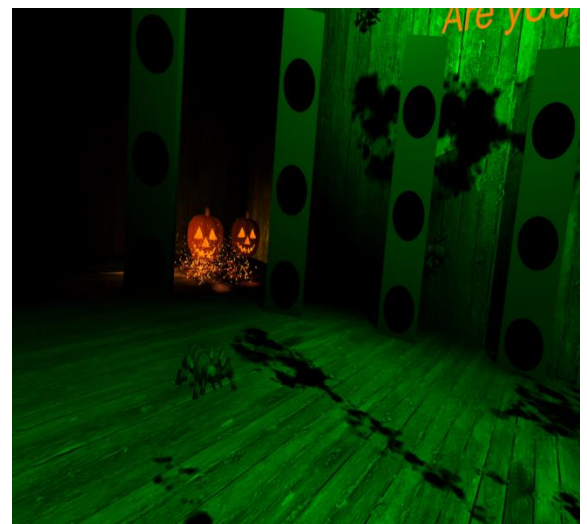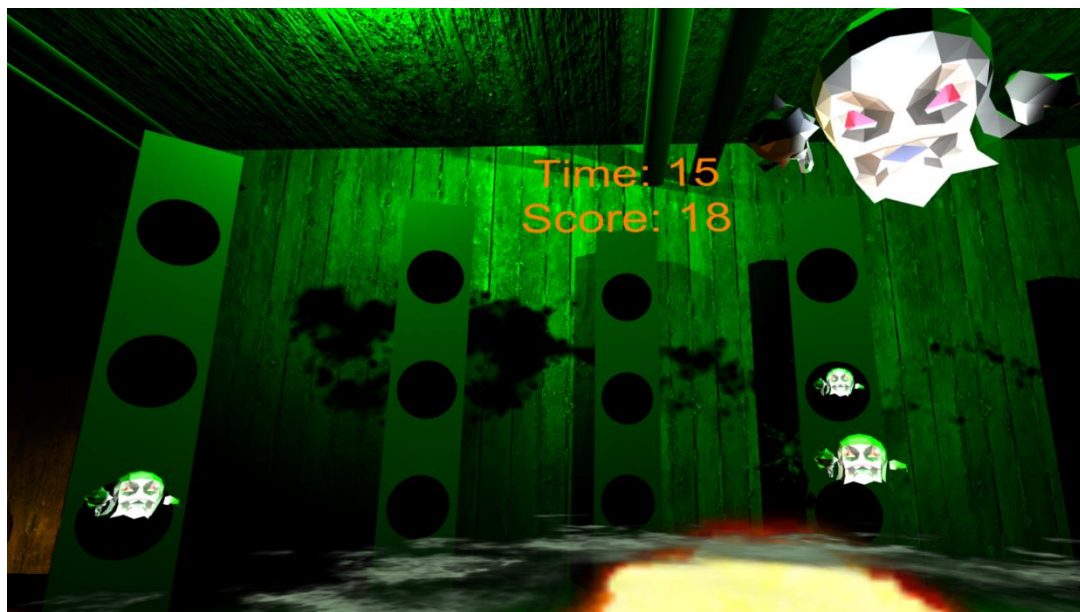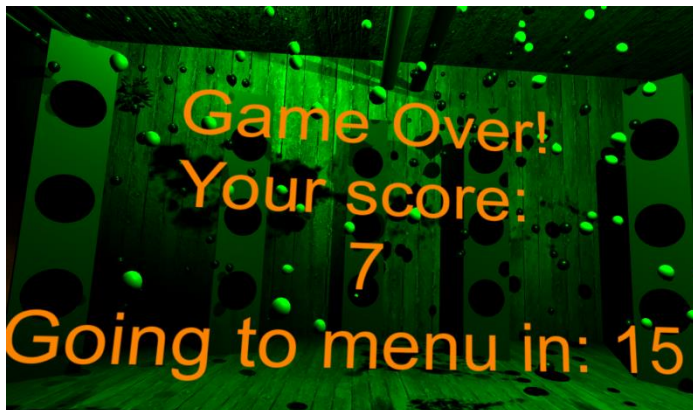
▌▌ Development Process

To accomplish our final project, we used unity 2017.2.0f3. We searched for ways to adapt our project to the HTC vive and the Manus VR equipment. We downloaded and attached to our project the Manus VR physics and steamVR plugin.

We searched suitable assets for building "Whack a mole" game with an arcade room environment but could not find fitting assets for our game vision. Then we decided to design the moles, "whack a mole" machine and arcade room using "Blender", a free 3d models design tool. We had some problem importing the Blender's objects we designed to unity, but we managed to do so by building the interactive set of the game in unity, removed their mesh renderer and put the design on the interactive set. We also found only static arcade machines with no games playing on their screen, hence we downloaded GIF player plugin for unity, and downloaded arcade games GIFs and attached it to the screens.

We decided the moles to pop-up randomly from the holes in the machine using C# code, we make it pop up faster as the time goes by. We had some problem making the game realistic as possible, because the player hands can move in the direction of a solid object like the mole, what we did was to make the mole move down really fast when the player hit it and added sounds impact so the player can feel like he really hit the mole.

Next we wanted to make more advanced level, with more features and focus more on the gloves abilities. We searched for a creature that can fly toward the player, so we came up with ghosts. Hence the second level became with much more spooky vibes. We wanted this level to be more difficult so we created seven pillars with three holes for each pillar, hence we have 21 sources the ghosts can come through. We also added bombs the player need to avoid to raise the difficulty and clocks to give the player more time to play. We searched a lot for compatible sound to each gesture for making a more powerful and more realistic gaming experience.

Our main problem in the development process was a glitch with the Manus VR trackers, their rotation went wrong with no connection to our development. It happened with both set of the GIP trackers, and to all of the PCs. It stopped our progress for a lot of time, we tried a lot of options with the Manus VR plugin but nothing worked. Finally we managed to make it work by contact the Manus VR support and with their guidance we solved the problem.

We needed to build a main menu for the game, the problem was that every information about building menu online, used buttons feature on unity, but we couldn't use the buttons with the gloves, so we build touchable objects that simulate buttons.

We learned a lot about unity environment, C# and how to use the HTC vive/ Manus VR equipment, we enriched our knowledge about VR and gaming world.

For conclusion, our main challenges during the process was to make the game look as real as possible, especially during the gloves impact with the game objects. We tried to make the game interesting and intractable, by adding more features to the game and to adjust existing and new features to the VR experience.