# IMPLEMENTATION OF VIRTUAL REALITY IN ANDROID GAMING

**Abhilash G. Joshi[1], Ashwini S.Dabhade[2], Ashwini S.Borse[3]**

[1] BE, Department of Computer Engineering, GESRHSCOE, Nashik, Maharashtra, India

[1] BE, Department of Computer Engineering, GESRHSCOE, Nashik, Maharashtra, India

[1] BE, Department of Computer Engineering, GESRHSCOE, Nashik, Maharashtra, India

---------------------------------------------------------------------------***---------------------------------------------------------------------------

**Abstract** - *In today's scenario, Virtual Reality which can be referred to as immersive multimedia or computer simulated life, replicates an environment that simulates physical presence in places in the real world or imagined worlds and lets the user interact in that world. Virtual reality artificially creates sensory experiences, which can include sight, hearing, touch, smell, and taste. Virtual reality gaming is where a person can experiences being in three dimensional environment and interact with that environment during a game. Based on Virtual Reality (VR) technology, we provide a more interesting and convenient way for people to play virtual reality game on Google Cardboard. In this project, we propose a Virtual Reality in Gaming for on Android platform through Google Card board (Wearable Device). The game is rendered when player aims using his/her eye sight at the specific marker. The players can view the virtual scenario through the lenses of Google Cardboard. Player moves the device to control the game. The experiment results show that the proposed game system can work effectively and provide winner result to the player.*

***Key Words***:  *Virtual Reality, 3-D game, Google Cardboard, Android Device.*

## 1. INTRODUCTION

The purpose of our project is to design and implement a 3-D game system in Virtual Reality using Google Cardboard game, which intended to replace the real world environment with the digital one and the human senses are immersed in the virtual environment. Based on Virtual Reality environment, the players may need to use their phones as well as wearable device to view the virtual world. Player only has to put android device in Cardboard and play the game. Our main goal is to improve immersion in shooter games, which are already known to be immersive especially amongst young generation. However, and due to time constrains, equipment availability and other limitations, this project turns out to be more like an exploration of possibilities and a challenge to difficulties in order to reach such a goal. In a matter of fact, the game play scenario was not our major concern. On this project we focused on finding a way to develop a virtual reality system, and on to learn how to solve problems in this field.

## 1.1 MOTIVATION

Motivations Real world and computer games have their own distinct strengths. By allowing us to combine these strengths, we can use VR to improve existing game styles and produce new ones. For discussion, we consider a player's gaming experience as consisting of four parts; physical, social, mental and emotional. Research into VR gaming serves another purpose beyond the improvement of gaming styles and the development of new ones, as gaming environments are well suited for exploratory research. In this section we examine the strengths of real world and computer games, the role of VR in combining and extending them, and the value of gaming as an exploratory research area. We strongly believe that Virtual Reality is a promising idea, which can enhance the player's gaming experience by providing exciting new ways to control their actions, through motions and 3D movement.



**Fig -1**: Requirements of gamming application

## 1.2 LITRATURE SURVEY
**Primitive Game Engines:**

Before game engines, games were typically written as singular entities: a game for the Atari 2600, for example, had to be designed from the bottom up to make optimal use of the display hardware this core display routine is today called the kernel by retro developers. Other platforms had more leeway, but even when the display was not a concern, memory constraints usually sabotaged attempts to create the data-heavy design that an engine needs. Even on more accommodating platforms, very little could be reused between games. The rapid advance of arcade hardware which was the leading edge of the market at the time meant that most of the code would have to be thrown out afterwards anyway, as later generations of games would use completely different game designs that took advantage of extra resources. Thus most game designs through the 1980s were designed through a hard-coded rule set with a small number of levels and graphics data.

**Unity Survey Area:**

3 Key points for each engine

- **I.** Usability(UI, how easy it was to learn and develop with)
- **II.** Functionality (what exactly the engine can do)
- **III.** Price point (speaks for itself)

Table 1 : Unity Survey Area

| PROPERTIES | UNITY | UNREAL ENGINE 4 | GAME MAKER | FLASH PRO | GAME SALAD |
|---|---|---|---|---|---|
| FULL 3D SUPPORT | YES | YES | NO | NO | NO |
| FULL 2D SUPPORT | YES | NO | YES | YES | YES |
| FREE VERSION | YES | NO | YES | YES | YES |
| PAID VERSION | YES | YES | YES | YES | YES |
| PC EXPORT | YES | YES | YES | YES | YES |
| PLAY STATION EXPORT | YES | YES | YES | NO | NO |
| XBOX EXPORT | YES | YES | NO | NO | NO |
| NINETENDO EXPORT | YES | NO | NO | NO | NO |
| IOS | YES | YES | YES | NO | YES |
| ANDROID | YES | YES | YES | NO | YES |
| WPS | YES | YES | YES | NO | YES |
| MAC | YES | YES | YES | YES | YES |
| AVAILABLE ON MAC OS | YES | YES | YES | YES | YES |
| AVAILABLE ON WINDOWS | YES | YES | YES | YES | YES |

## 2. PROPOSED SYSTEM

In our project we proposed a game system which can be played by two ways:

1. Endless game- After selecting the endless game player gets started with his game. Player has to fight with the enemies to finish the game. As the game is endless therefore there will be no as such a finish for the game. Player have to earn more and more points to increase his level. After getting increasing level game difficulty will increase and also player will be having advance weapons and different spaceships to fight with.

2. Level based game- Level based game will be divided into some number of levels. After completing one level player will get next level unlocked also his power will be more. But the difficulty level will be increasing as player gets into next levels. This game will end after completing the level and finish point for the game will be after completing all the levels
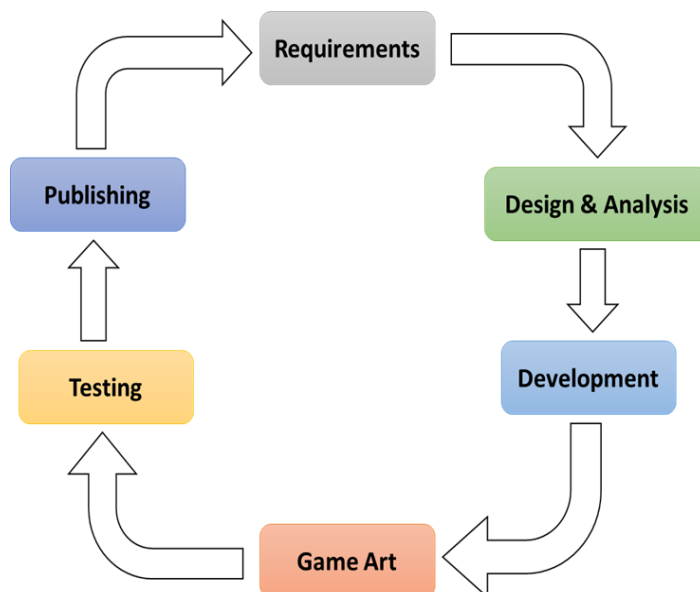


## 3. COMPONENTS

1. Gyroscope:

Gyroscope is the inbuilt function in the android or iOS devices. It is used for the rotation of the view in the screen in 360 degree rotation. The working of Gyroscope is as shown in the figure. By using the Gyroscope player can see in every direction i. e. Life, Right, Up, Down, Front and Back. A gyroscope is a device that uses Earth's gravity to help determine orientation. Its design consists of a freely-

rotating disk called a rotor, mounted onto a spinning axis in the center of a larger and more stable wheel. As the axis turns, the rotor remains stationary to indicate the central gravitational pull, and thus which way is "down. "The gyroscope maintains its level of effectiveness by being able to measure the rate of rotation around a particular axis. When gauging the rate of rotation around the roll axis of an aircraft, it identifies an actual value until the object stabilizes out. Using the key principles of angular momentum, the gyroscope helps indicate orientation. In comparison, the accelerometer measures linear acceleration based on vibration.

2. Skybox:

Skybox is the mechanism for applying effect to the game surrounding. There are various types of skybox which contains the Dawn effect, Eric effect, Moonshine effect, Overcast Effect, Starry Night effect, Sunny effect. A skybox is a method of creating backgrounds to make a computer and video games level look bigger than it really is. When a skybox is used, the level is enclosed in a cuboid. The sky, distant mountains, distant buildings, and other unreachable objects are projected onto the cube's faces (using a technique called cube mapping), thus creating the illusion of distant three-dimensional surroundings.
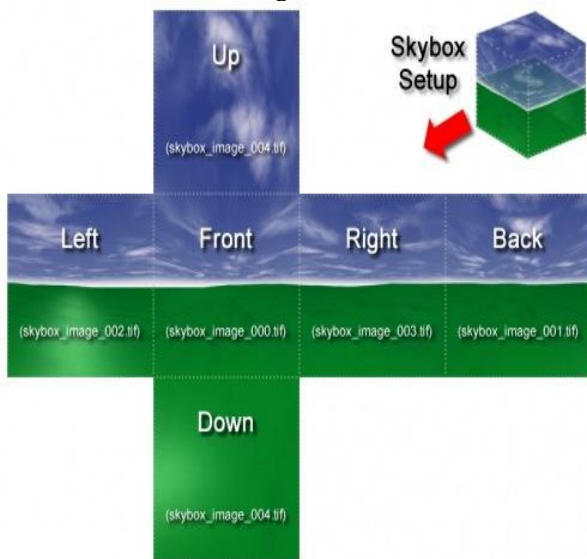


3. Random Generation of Different Objects:

The generation of the object using random spawning is carried by using the C script. Using this script the number of the generated object and its generation speed is maintained.

4. Target Image View with Camera:

Target image view is the marker used for targeting the object and destroying them. This target destroys the object.

5. Object Destroy on Sight Shoot:

A cube collider of specific size is attached to the target and is made invisible using Mesh Renderer. When the object come into the range of the collider it appears in front of the target then it gets destroyed.

6. Cardboard Screen Divide: Cardboard screen Divide is the mechanism done using adjustment of the size of the camera or by applying the Cardboard API which splits the main camera into two consecutive cameras.
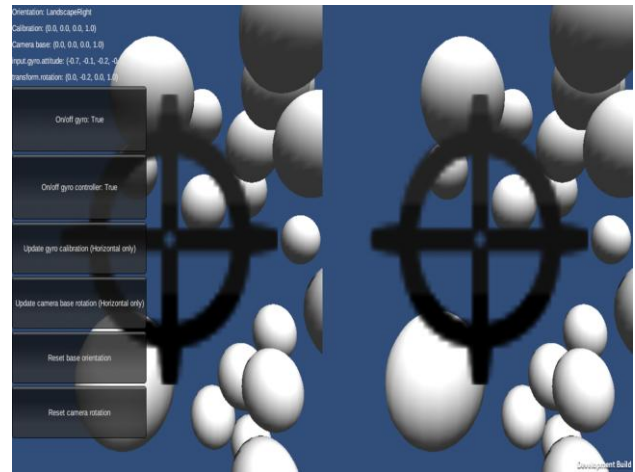


7. Particle System:

Particles are small, simple images or meshes that are displayed and moved in great numbers by a particle system. Each particle represents a small portion of a fluid or amorphous entity and the effect of all the particles together creates the impression of the complete entity. Using a smoke cloud as an example, each particle would have a small smoke texture resembling a tiny cloud in its own right. When many of these mini-clouds are arranged together in an area of the scene, the overall effect is of a larger, volume-filling cloud.
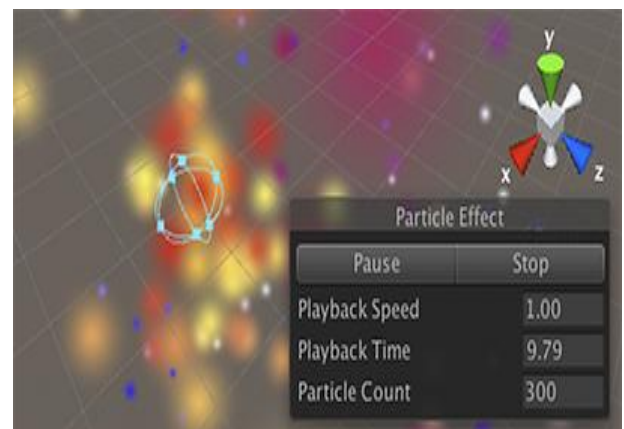
## 4. IMPLEMENTATION DETAILS

### 4.1 System Architecture
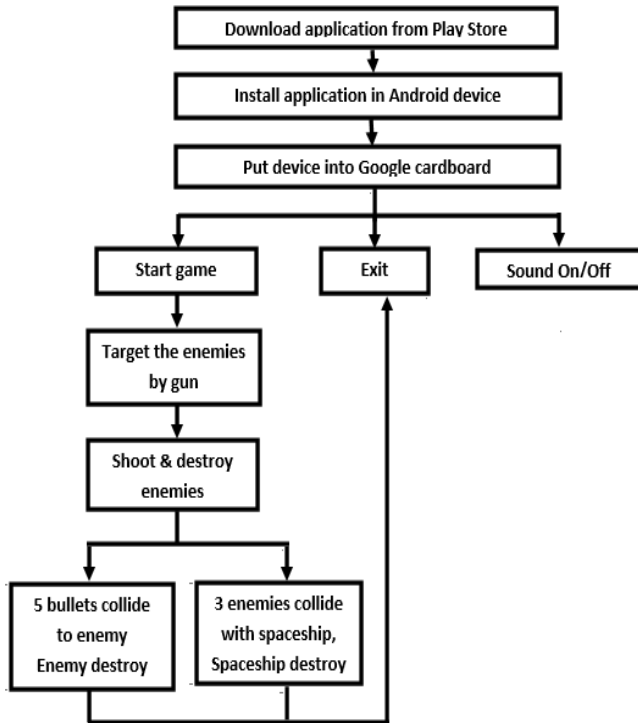


System architecture tells us that user has to download the game from Play Store and install into the android device. After completing the installation user has to launch the game application and has to put device into the google cardboard. Then the time comes when use can interact with device only using sensors. User has to use motion sensors to interact with the device. User starts the game by using motion sensors to reach toward the Start Game button and stare over it for few seconds. As soon as game gets started enemies starts to approach towards the player. Player has to target the enemy and kill them in 5 bullet shots. After 5 shots enemy destroys. Also the player get killed by enemies only after 3 enemies get collided with the target. After this next screen turn up showing either Restart or Exit the game.

### 4.2 Hardware and Software Specification

Hardware Specification:
1. Android Mobile Device with
2. Android OS 4.0 and above
3. 1.2 GHz Processor
4. Minimum 512 MB RAM
5. Gyroscope Feature supported
6. Google Cardboard (Wearable Device)

-Piece of cardboard
-45 mm focal length lenses
-Magnets or capacitive tape
-Hook or loop fastener

Software Specification:
1. Platform:
   Front end: Unity 4.5(Game Engine)
              3D max, MAYA (For Graphics).
   Back end: C script, JavaScript.
2. Operating System:
   Android
3. Tools:
   Unity 3D
   MonoDevelop
4. Programming Language:
   C sharp
   JavaScript

## 5. USER INTERFACE

### 5.1 Home Screen



### 5.2 Gameplay Screen

5.3 Game over Screen



## ACKNOWLEDGEMENT

## REFERENCES

[1] An Augmented Reality 3D Ping-Pong Game System on Android Mobile Platform ,Xin Gao, Jie Tian, Xiaoyuan Liang, Guiling Wang,2014.

[2] Augmented Reality, https://en.wikipedia.org/wiki/ Augmented reality/, 2014.

[3] A preliminary System framework for SilGam Book, Junhun Lee, YeongmiKim; Ryu, 2009.