



# International Journal for Innovative Engineering and Management Research

A Peer Reviewed Open Access International Journal

www.ijiemr.org

## COPY RIGHT



**ELSEVIER**  
**SSRN**

**2023 IJIEMR.** Personal use of this material is permitted. Permission from IJIEMR must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. No Reprint should be done to this paper, all copy right is authenticated to Paper Authors

IJIEMR Transactions, online available on 11<sup>th</sup> Jan 2023. Link

[:http://www.ijiemr.org/downloads.php?vol=Volume-12&issue=Issue 01](http://www.ijiemr.org/downloads.php?vol=Volume-12&issue=Issue 01)

**DOI: 10.48047/IJIEMR/V12/ISSUE 01/77**

Title **Design of a Simulator to Study Driving Vehicle Performane using Unity Game Engine**

Volume 12, ISSUE 01, Pages: 830-839

Paper Authors

**Boyanapalli Aakash, Dr. M. Narayana, Dr. Abdul Ahad, P Deepak**



USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per **UGC Guidelines** We Are Providing A Electronic Bar Code

## Design of a Simulator to Study Driving Vehicle Performane using Unity Game Engine

Boyanapalli Aakash<sup>1</sup>, Dr. M. Narayana<sup>2</sup>, Dr. Abdul Ahad<sup>3</sup>, P Deepak<sup>4</sup>

<sup>1</sup> UG Student, Department of AI, Anurag University, Hyderabad, Telangana, India.

Mail-Id: aakashkmt29@gmail.com

<sup>2</sup> Processor, Department of ECE, Anurag University, Hyderabad, Telangana, India.

Mail-Id: narayanaece@anurag.edu.in

<sup>3</sup> Associate Professor, Department of AI, Anurag University, Hyderabad, Telangana, India.

Mail-Id: ahadbabu@gmail.com

<sup>4</sup> UG Student, Department of AI, Anurag University, Hyderabad, Telangana, India.

Mail-Id: [deepakpuram3@gmail.com](mailto:deepakpuram3@gmail.com)

### Abstract

I present a driving simulator app developed using unity to improve driving skills of learners. The app consists of realistic environment so that user can get driving skills. The simulator app is developed using Unity game engine, it allows applications and games to be created and exported to many different platforms. The game scenario consists of different roads, traffic boards, houses, gas stations and mountains etc. During simulation the traffic boards, houses, gas stations make the simulation more realistic. The driving skills can be studied by testing the user using the simulator. The simulator is reviewed by all users in play store as it is similar to real driving and useful for learning driving and understand traffic rules.

**Keywords** - unity game engine, ncrb, car driving simulation, driving skills, adas

### Introduction

It is possible to create very realistic environments that let user to immerse in the simulator and achieve best results in their use by using advanced technologies. These types of simulators can speed up the process of acquiring basic skills and improves the confidence levels of the users to drive vehicle perfectly

In Particular, driving simulators improve driver's driving skills by including different ways and situations in which safety can be neglected depending on the driver's behavior. The simulator allows future drivers to get used to many situations that rarely happen but could lead to traffic accidents, such as landslides,

animals in the middle of the road, or a previous accident. The development of simulated environments allows drivers to recover data that can be analyzed not only for their use but also to study different factors that have an impact on traffic safety [1]. Simulators can help to reduce the accidents and also helps to users to boost up the confidence in driving.

A comparative analysis of the National Crime Records Bureau (NCRB) data from 2017 to 2020 shows that the number of lives lost per 100 road crashes was 35 in 2019, 34 in 2018 and 32 in 2017. This is also defined as “severity of accidents” and an indicator of how safe or unsafe roads are. Users can make the environment safe by learning with these simulators.

### **Problem Statement**

There are many other simulators which are very expensive. In driving simulators, the relationship between them and the results generated as a function of them can be included to analyze the level of driving efficiency and safety. When a person sees himself in a real driving environment like a city with heavy traffic or a complex junction, he may think that he is not up to the challenge. In particular, there are people who suffer from amaxophobia, a fear of vehicles or traveling in a vehicle.

Driving simulators allow users to experience many of these challenges in virtual environment without endangering the user or other drivers [2]. Simulators enable users to acquire driving skills to encounter and overcome these problems, so that they can apply these skills when they encounter similar situations in a real environment.

### **Proposed System**

We propose a car simulator to learn driving skills in a unique manner. And these simulators having many unique features comparing with other simulators. Naturally, simulators allow users to repeat simulations in one instance and encounter certain events multiple times. Moreover, it is possible to simulate real situations that are practically impossible to do voluntarily in a real environment. Events such as a vehicle stopping in the middle of the road or a pedestrian accidentally crossing the road are events that can easily be replicated in the simulator. This enables simulator users to better prepare for unusual situations that may be encountered in the future.

Simulators can also be used to conduct studies on how certain agents, such as a driver's fatigue, physical or mental state, medication, drugs or alcohol, or age, affect driving ability [3]. Moreover, they can be

used to study Advanced Driver Assistance Systems (ADAS) or the presence of children in the vehicle on driving performance easily and without risk.

Finally, it is important to mention that simulators can collect data in a simple way. Thus, users can access information on completed simulations and observe their evolution. It allows drivers to see their mistakes and committed traffic offenses and what aspects they need to improve to avoid them in the future.

## Methodology

In this work, a driving simulation scenario with different road sections and events and a vehicle model are presented to analyze driving efficiency and safety. Scenery and vehicle modelling were developed with Unity Game Engine [4]. Moreover, it is possible to export applications to many different platforms including PC, mobile and consoles. A notable feature of the latest version of Unity is its integration with virtual reality platforms such as Oculus.

A driving simulator integrates scenarios and displays them in real time, which is used to analyze the user's driving performance, which is used to analyze the different simulations fulfilled by the driver, so that conclusions can be drawn about their driving ability. The rest of the

paper is organized as follows. Section two presents the problem statement and simulators. Next, Section 3 describes the design of the driving simulation followed by the implementation of the user vehicle followed by Section 4 and Section 5 shows the experimental details and finally Section 6 presents the main conclusions about the presented work.

## Simulator Design

Creating a platform where the users can learn to drive safely and efficiently. There are very less simulators to learn driving in a cheaper way, to solve this problem I have developed this car simulator. A driving simulator is software that simulates the behaviour of a particular process. In this process, real situations are replaced by other artificially created ones that serve to acquire competencies that can later be transferred in an effective way to a real-life situation. In the field of education, simulators are meant not only for building concepts and enhancing knowledge, but also for applying them to new contexts that, for many reasons, are inaccessible to people from the methodological context in which their learning was developed. Simulators have educational advantages such as providing open learning environments based on real models, and those users adopt an active role by

transforming themselves into builders who learn from their own experience. All of the above obviously applies to driving simulators.

Consequently, simulators for learning and predicting safe and efficient driving should be carefully designed to reinforce optimal driving behaviours by providing feedback [5]. Over the years, driving simulators have been used in many research studies for a variety of purposes. Their suitability has been shown as a tool for assessing drivers' behaviour and learning safe and efficient driving skills.

### **Implementation of Driving Simulation Scenario**

I have developed a 3D driving simulation scenario using the Unity game engine, with the aim of covering a wide range of environments and situations that drivers need to know in order to react properly. I have also designed a consumer vehicle. The development of the game started by designing the terrain (It is an object which has terrain collide, a physics component which is use to collide with a game object to place an object on the terrain to replicate real world terrain). Then more terrains are placed using the 'create neighbour terrain' button in inspector while the initial terrain is selected. In my simulator there are nine terrains placed in a square shape. Then these terrains are given

a texture which is a grass colour which is taken from an asset of unity asset store called Grass Flowers. Then the terrain is sculpted using the raise or lower terrain option present in the inspector while the particular terrain is selected, and also various brushes are used from unity assets to sculpt the terrain to have hills.

Then the terrain is added with various trees, grass, flowers placed in the terrain. The tree assets are conifers BOTD and the flowers and grass assets are from Grass Flowers asset. These assets are from unity assets store which are free assets. The assets can also be designed in blender and then those models can be used in unity but in this simulator, we used the free assets that are in unity asset store considering the time aspect and optimal way to develop the simulator. Then the directional light is adjusted, the event system is generated and then the roads are placed in on the terrain after sculpting the road way in the terrain. The road ways are painted with mud texture first using paint terrain option in inspector then the roads are placed as game objects.

The road assets used are Easy Roads 3D from unity asset store. The rock models are placed using mass place option in inspector. The invisible walls placed at the edge of the terrains used glass texture to be



invisible. Then the houses and gas station models are placed on the terrain. The assets used to get the house and gas station models are UK terraced House FREE and Toons gas station from the unity assets store. The lake and the stunts are also placed on the terrain. The sun light is also adjusted. The rock models are also placed using mass place option in inspector [6]. Then all the game objects are added with mesh colliders so that the car will collide with these objects such as a real car collide with real world objects.

Finally, after removing all the unwanted trees, flowers, grass and rocks on the road by using left shift button and brush while the object is selected. The designing part of the game scene is completed. This is how the terrain looks after designing the scenario shown in Figure 1.

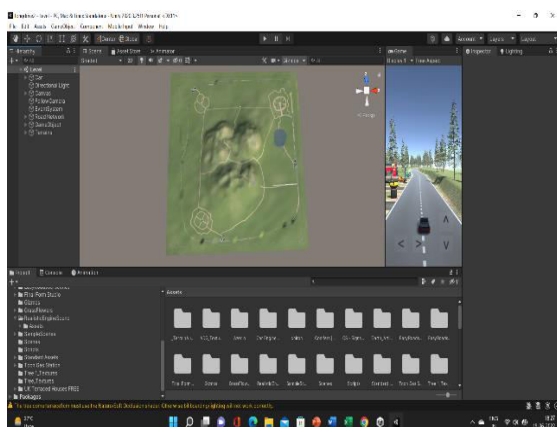


Fig. 1: The terrain looks after designing the scenario

## Implementation of users vehicle

Now coming to the development part of user's vehicle, the user's vehicle is a car model which is an asset taken from unity asset store which is further modified by adding more objects and textures to it. Then rigid body physics component is added to that car model to make the car model act like a rigid body. This gives the model some mass, drag, angular drag, gravity, which makes the model act as a real rigid body as in the real world. The car model has lights, helpers and car object of which the car object has wheels object and car body and spoiler [7]. The car model can be seen in the hierarchy as in Figure 1. Now the car model is designed but in order to move this object we need to add some functions to that object, we can do that by coding the inputs which are given to that object which gives an output. We should do that first by finding a way to give that input, for mobile it will be touch buttons and for pc's it will be keys. we have taken mobile input as I wanted to develop an application which can reach easily to the maximum number of people. The canvas is created then four buttons are added to it, one for acceleration, one for deceleration, one for turning left and one for right.

Now in order to take these inputs an input system script is created and coded in vs

code using c sharp(c#) language and then the script is added to the car model as a component. The input system has all these buttons declared and the start function has the get component function which enables to get input when a button is on pressed and that input is sent to the car controller script which is created. The car controller code has many declarations such as for torque, downforce, speed type, top speed, brake torque, steel angle, old rotation which are used in the script. The start function is for the initialization which has many functions used for brake torque, wheel colliders for rigid body, current torque, function to add a curved bias towards 1 for a value 0-1 range, unclamped version of lerp (to allow value to exceed the from-to range) and move function which is the key function for the movement of the car model [8]. Move function will take all the inputs such as accel, brake, left and right. In move function the inputs are clamped to move the car object in terms of 0 to 1 for accel, -1 to 1 for steering and -1 to 0 for brake. This move function is responsible for the movement of the car while all other functions mentioned earlier are simply used to make the movement of the car look more realistic otherwise the movement looks discrete which is not how a real car move. The car audio works by taking

inputs when on pressed a button, there are three types of engine audio used from standard unity car audio assets which are played according to which button is pressed by user.

For example, if user presses acceleration button, then the acceleration high mp3 is played. Now when we click the run button, we can see the car moving when the buttons are pressed but the wheels are not spinning, so now there will be added a script added to the wheel object called wheel effects. The wheel effect is a simple code which spins the wheel object in forward direction when accel button is clicked and spins backward when deceleration button is clicked. The wheel effect also rotates the front wheels to 30 degrees left when the left button is clicked to steer the car model and 30 degrees right when the right button is clicked to steer the car object. Finally, the development of the user's vehicle is completed which can be shown in Figure 2.

Now there are other buttons added in canvas such as restart, home, pause and music off buttons. The restart button is to restart the scene when the car is struck. The restart script is added to the restart button component and when the restart button is on pressed then the restart script code executes and the script has the load

scene function in the start function which makes to load the scene and start the scene again. Start audio is also added to the car model which only gets played at the start of the scene. The home button loads the home scene which is the first scene which gets opened when the application is opened which has the drive now button and the quit button. The drive now button loads the game scene [9]. The quit button has an application. quit function which will terminate the application. The pause function has a pause script which has conditional statement to control the time scale like if pause is clicked timescale = 1, else 0.

Finally, the music off button will off the background which is automatic and played on loop. Then the application is added with name, logo, splash screen and then while build process the home scene is given index as 0 so that it is opened first to give user a choice to drive or quit. The home scene can be seen in Figure 3. Then the game is built and tested on android. The figure 2 shows how the app is displayed on android device.

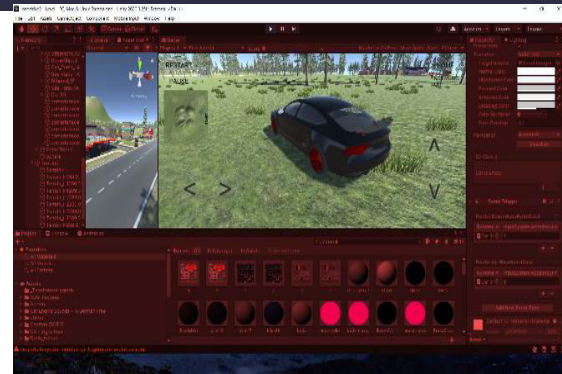


Fig. 2: How the App is displayed on Android device

Now this app is ready to test the driving performance of the users. When a user drives the car, he can experience the realistic movement of the car when the buttons are clicked. For a person who is driving a car for the first time in their life it is not recommended to give the real steering to their hands. It is better to teach then first on the simulation [10].

By this application many can learn the basics of driving which can avoid many traffic accidents while learning driving and also the simulation provide entertainment to the ones who want to drive a car sitting in their home.



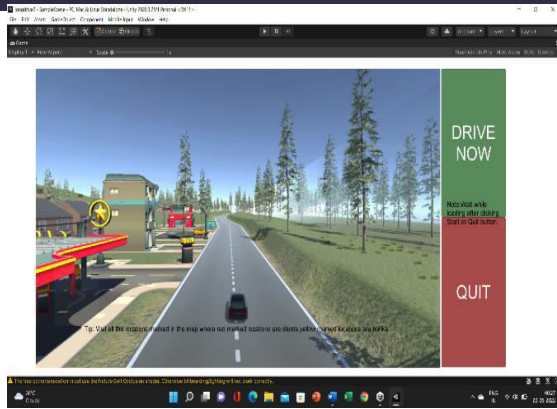


Fig. 3: The Home Screen on Android Device

### Experimental Results

While testing we took the experimental values of 6 people with an age ranging of 19-25 years old. First three users are first taught driving in the car simulator and then given a real car to learn driving and the next three users are directly given a real car to learn driving. All the users are given scores based on their performance by driving inspector while driving the real car. The scores are ranged from 1-10. The experimental results are shown the Figure 4. The results show that the users who learned driving in car simulator are prone to less mistakes and learnt driving fast, where as a user who drove a real car directly made many mistakes as they did not have any basic experience driving a car like how a car does turning its sensitivity and how to steer a car while going back.

This experience is very important for a user to avoid disasters on real road. The persons who played the car simulator later also continued to improve their driving skills, users also learnt how to do a perfect turn and drift. In this way a user can also learn new skills tested in car simulator which are dangerous to perform in real road. They might not get a real experience but the simulator provides information that what how to control a car while doing the skill in real road. And also, the third user have more score than first because the third users learnt after seeing how the first user drove as the other two users are also sitting in the same car with inspector.

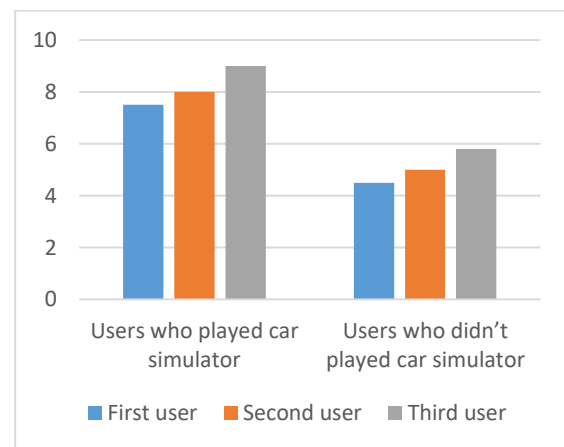


Fig. 4: Analysis of users scores for driving real car

In figure 5 the image we see is how the simulator looks on a real android device and we can see the car model, buttons, terrain and all other game objects. The apk of the car simulator is installed on the

android device to use the car simulator. The car simulator is also published in play store with a name of Long Drive Car Simulator, so that other users can also do self-assessment of their driving abilities by installing the app from play store.



Fig. 5: The Simulator on Android device

## conclusion

In this paper, Car driving simulation developed with Unity game engine. The simulator influences safety and efficiency through driver behaviour by incorporating different environment, user vehicle and vehicle controls by assessing the acquisition of driving skills and driver performance. Continuous technological advances have begun to develop low-cost simulators with realistic scenarios and events that would be impossible to have in a real driving environment without endangering people or vehicles. People who participated in simulation driving experiments highlighted the ease of interaction, the realistic experience and the

usefulness of learning to drive. The simulator can be extended by making the camera view into VR mode by using google cardboard VR plugin and make the car move by gyro inputs by which more realistic experience can be given to the users. And also, I am thinking to use photon cloud to make the simulator a multiplayer VR simulator so that I can develop metaverse for the improvement of communication in future. The car simulator can be installed by user having android device from play store. The simulator is named as Long Drive Car Simulator which has 5k+ downloads and still counting with 4.9 rating.

## References

- [1] S. Jamson, D. Hibberd, and A. Jamson, "Drivers' ability to learn eco-driving skills; effects on fuel efficient and safe driving behaviour. Transportation Research Part C: Emerging Technologies", IEEE International Conference, pp. 657-668, August 2015.
- [2] S. Yamin, A. Stinchcombe, and S. Gagnon, "Deficits in attention and visual processing but not global cognition predict simulated driving errors in drivers diagnosed with mild Alzheimer's disease", American Journal of Alzheimer's Disease and Other Dementias, pp. 351-360, November 2016.
- [3] S. Duan, S. Cong, Z. Zou, and Y. Song "modelling and simulation of the quantum ranging and positioning

- system”, International Journal of modelling and simulation, pp. 421-435. August 2019.
- [4] M. Quwaider, A. Alabed, and R. Duwairi, “Shooter video games for personality prediction using five factor model traits and machine learning”, Journal of simulation modelling practice and theory, January 2023. in press
- [5] Abdul Ahad and Marlene Grace Verghese, “The Substructure for estimation of miscellaneous data failures using distributed clustering techniques”, Proceeding of International Conference on Information Technology and Applications, Volume 350, April 2022.
- [6] S. Tamgade, V. Chakradhare, K. Sarve, S. Choudhari, and M. Wankhade, “Vibrometer to Measure Occlusion Force on Teeth Using Image Processing”, International Journal of Research in Engineering and Technology, pp. 26-27, November 2006.
- [7] Brubeck Lee, “a multi-point constraint unfitted finite element method”, Journal of advanced modelling and simulation in engineering sciences, September 2022.
- [8] M. Sailaja, Abdul Ahad, and Ali Hussain, “Machine Learning Medical Resources Allocation”, Journal of Physics: Conference Series, December 2021.
- [9] G. Grigoryan, S. Etermadidavan, and A. Collins, “computerized agents versus human agents in finding core coalition in glove games”, Sage Journal, May 2022.
- [10] S. Ward, W. Higinbotham, E. Duvelson, and A. Saciragic, “Inside the cloud-network communications basics for the relay engineers”, IEEE Annual Conference, pp. 273-303, April 2008.