RESEARCH ARTICLE OPEN ACCESS

Exploring the Lunar Landscape : VR Walkthrough using Mobile VR Headset

Lokesh Parab*,Lavanya Reddy**,Rohit Yadav***,Siddhesh Shirdhankar****, Prof.Mohan Kumar*****

*(Department of Electronics & Telecommunication, Mumbai University, Malad(W)–400095,Maharashtra, lokeshparab4545@gmail.com)

**(Department of Electronics & Telecommunication, Mumbai University, Malad(W)–400095,Maharashtra, lavanyareddy2310@gmail.com)

***(Department of Electronics & Telecommunication, Mumbai University, Malad(W)–400095,Maharashtra,, cmrohityadav23@gmail.com)

****(Department of Electronics & Telecommunication, Mumbai University, Malad(W)–400095,Maharashtra, sid.shirdhankar18@gmail.com)

*****(Department of Electronics & Telecommunication, Mumbai University, Malad(W)–400095,Maharashtra, mohankumar@atharvacoe.ac.in)

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

Abstract:

The advent of virtual reality technology has revolutionized the way we experience and interact with digital environments, offering unparalleled opportunities for immersive exploration. Among the myriad applications of VR, simulating extra-terrestrial landscapes stands out as an especially captivating endeavour. In this paper, we delve into the development and implementation of a VR walkthrough on the moon, specifically tailored for mobile headsets, utilizing the powerful combination of Unity and Blender. By leveraging the cutting-edge capabilities of these software platforms, we aim to provide users with a realistic and enthralling experience of traversing the lunar surface, offering insights into both the technical intricacies and the creative process involved in crafting such immersive virtual environments. This paper not only showcases the potential of VR as a tool for education and exploration but also underscores the significance of interdisciplinary collaboration between computer graphics, and scientific visualization in pushing the boundaries of immersive story telling.

Keywords-**VR technology, Mobile VR headset, Walkthrough, Unity, Blender**

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

INTRODUCTION

In recent years, virtual reality (VR) technology has emerged as a powerful tool for creating immersive digital experiences. Among the various applications of VR, the simulation of extra-terrestrial landscapes holds particular fascination. Our research focuses on developing a VR walkthrough set on the moon, a project that integrates the capabilities of Unity and Blender. Unity, a versatile game engine, provides the framework for creating interactive VR environments, while Blender, a robust 3D modelling software, enables the detailed design of lunar terrain, lighting and assets. By combining these tools, we aim to offer users a captivating experience of exploring the lunar surface in a mobile VR headset, allowing them to interactively navigate and discover its unique features.The development process involves several key components. Firstly, meticulous attention is paid to recreating the lunar landscape with precision and accuracy. This includes sculpting the terrain, texturing surfaces, and integrating realistic lighting to emulate the distinctive look and feel of the moon's surface. Secondly, spatial audio is implemented to enhance immersion, with sound effects meticulously designed to reflect the environment and contribute to the sense of presence. Finally, user interaction and navigation are carefully crafted to provide a seamless and intuitive experience, allowing participants to freely explore the virtual lunar environment at their own pace.

Fig. 1: Mobile VR Headset

Beyond education, our research underscores the broader implications of VR technology in scientific visualization and public engagement with space exploration. By leveraging the immersive capabilities of VR, we can inspire curiosity, spark interest in science, and foster a deeper appreciation for the wonders of the cosmos. Moreover, the interdisciplinary collaboration between computer graphics, game development, and scientific expertise highlights the importance of synergy between different fields in pushing the boundaries of immersive storytelling and interactive experiences. Through our exploration of the moon in VR, we aim to showcase the potential of this technology to not only entertain but also educate and inspire audiences worldwide.

II.THEORY

Process for designing a 3D model using Blender:

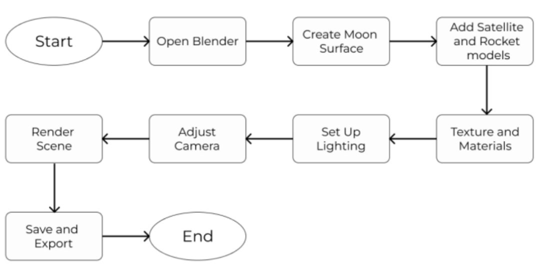

Fig. 2: Steps for designing a 3D model

Building the Foundation:

The initial step involves selecting appropriate geometric shapes from Blender's primitive library. Cubes are ideal for constructing satellites. Blender offers various tools to manipulate these primitives. The user can adjust the size of each shape using the scaling tool to achieve the desired dimensions.

Visual Refinement:

Flat surfaces come alive with the application of textures in Blender. These textures, which can be obtained from within Blender's library or imported from external sources, add visual interest and mimic the appearance of real-world materials like wood grain, brick patterns, or fabric weaves. Assigning materials to objects further enhances visual realism [7]. Blender offers a vast library of materials like wood, metal, glass, or fabric, each with customizable properties like colour, roughness, and reflectivity. Selecting the appropriate material for each object significantly impacts the overall aesthetic of the environment [15].

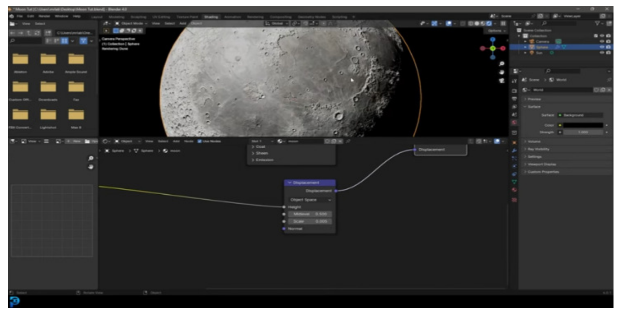

Fig. 3: Adding textures to the Moon Surface

Lighting and Mood:

Lighting plays a vital role in establishing the mood and atmosphere within the 3D environment. Blender offers a variety of light types, including point lights that cast concentrated beams, spotlights that illuminate specific areas, and area lights that bathe the scene in soft overall illumination [4]. The user strategically positions these lights to achieve the desired effect. By strategically placing spotlights or adjusting the intensity of specific lights, users can draw attention to particular elements within the environment, creating a sense of focus and guiding the viewer's eye [2].

Iterative Process:

A core principle of this theory is the iterative nature of the workflow. After completing each stage (modeling, adding details, texturing, lighting), the user has the opportunity to evaluate the outcome. If something feels off – the proportions are wrong, a detail is missing, or the lighting isn't creating the desired mood – the user can revisit the corresponding stage and make adjustments before progressing further.

Final Rendering:
Translating the Virtual into Reality: Once all aspects have been meticulously crafted, the final stage involves rendering the scene. Rendering translates the 3D scene into a high-quality image or animation. This serves as the final output, a tangible representation of the user's creative effort.

Finally, the designing of 3D Models is completed, Now the focus shifts towards the development of a space exploration for mobile virtual reality (VR) headsets.

III. WORKING

A. Program:

The provided code snippet illustrates a fundamental aspect of player input and character movement implementation within the Unity game development environment. Through the Unity Input system, the code captures player input from the horizontal and vertical axes, typically corresponding to keyboard or controller inputs for left/right and forward/backward movement, respectively.

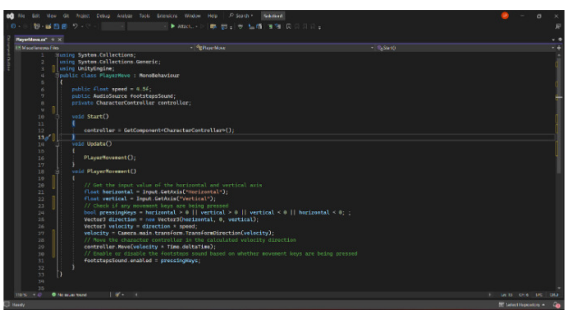

Fig. 4: Program for Player Movement

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PlayerMove : MonoBehaviour
{
    public float speed = 3.5f;
    public float gravity = 10f;
    private CharacterController controller;

    // Start is called before the first frame update
    void Start()
    {
        controller = GetComponent<CharacterController>();
    }

// Update is called once per frame
    void Update()
    {
        PlayerMovement();
    }
    void PlayerMovement()
    {
        // Get the input value of the horizontal axis (e.g., A/D keys or left/right arrow keys)
float horizontal = Input.GetAxis("Horizontal");

// Get the input value of the vertical axis (e.g., W/S keys or up/down arrow keys)
float vertical = Input.GetAxis("Vertical");

// Check if any movement keys are being pressed
bool pressingKeys = horizontal > 0 || vertical > 0 || vertical < 0 || horizontal < 0;

// Create a new Vector3 to represent the direction of movement based on the input
Vector3 direction = new Vector3(horizontal, 0, vertical);

// Calculate the velocity vector by multiplying the direction vector by the speed
Vector3 velocity = direction * speed;

// Transform the velocity vector from local space to world space based on the camera's orientation
velocity = Camera.main.transform.TransformDirection(velocity);

// Move the character controller in the calculated velocity direction, scaled by Time.deltaTime for smooth movement
controller.Move(velocity * Time.deltaTime);

// Enable or disable the footsteps sound based on whether movement keys are being pressed
footstepsSound.enabled = pressingKeys;
    }
}
```

By aggregating these input values, the code determines whether any movement keys are actively being pressed. Subsequently, it constructs a direction vector in three-dimensional space based on the input values, facilitating the calculation of a velocity vector scaled by a specified speed parameter. Crucially, the velocity vector undergoes a transformation to align with the camera's orientation, ensuring consistent movement relative to the player's viewpoint. Leveraging Unity's character controller component, the code then applies this velocity to physically move the player character within the game world, accounting for frame rate fluctuations through multiplication by `Time.deltaTime`.

Additionally, the code manages the activation of a sound effect, presumably representing footsteps, contingent upon the player's engagement with movement controls. Overall, this code snippet exemplifies the foundational processes involved in handling player input and character movement within Unity [11].

B. Building a space exploration for a Mobile VR headset:

3D design development using Blender:

Utilizing Blender for model creation involves leveraging its robust suite of tools for environment design. Users can texture, and refine intricate details to create immersive environments. Blender offers flexibility in modelling techniques, allowing for the creation of different elements [12]. Its compatibility with various file formats ensures smooth integration with Unity for VR development. Through Blender, designers can achieve high-quality, visually appealing models optimized for VR experiences [5].

Unity Integration:

Unity serves as a central hub for integrating Blender models into VR applications, offering unparalleled capabilities for creating immersive environments. By leveraging the integration of Blender models, developers are afforded the opportunity to harness the sophisticated graphics rendering and lighting systems inherent to Unity [13]. This enables the manipulation and optimization of models with high-quality graphics and dynamic lighting, ensuring stunning visual fidelity within VR environments. [6].

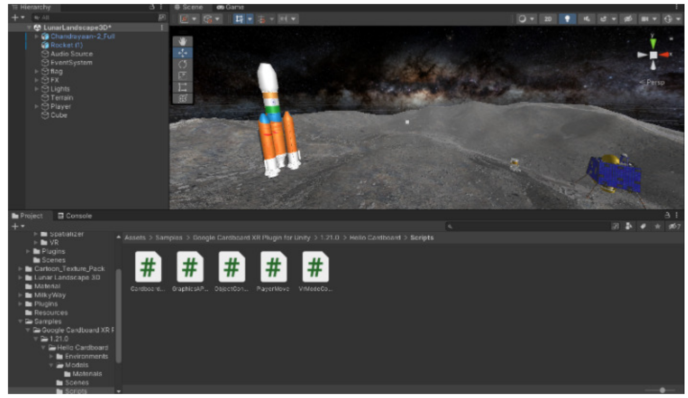

Fig. 5: Unity Interface

Bluetooth Controller Setup:

Integrating Bluetooth controller support within Unity enables users to interact with the VR environment intuitively. Through Unity's scripting capabilities, we can map controller inputs to specific actions within the VR scene. This includes configuring button triggers for navigation, interaction, and menu selection.

Fig. 6: Bluetooth controller

By supporting Bluetooth controllers, the VR experience becomes more accessible and engaging, offering users greater control and immersion. The joystick enables movement in four directions: forward, backward, left, and right, offering users to walk through the environment. In addition to movement, the trigger button on the joystick serves as a versatile tool for interacting with objects and models within the VR scene. By assigning specific actions to the trigger button, developers can create dynamic and interactive experiences that fully utilize the Bluetooth controller's capabilities [14].

Adding audio & sound effects:

In Unity, immersing users in a virtual reality environment can be greatly enhanced by incorporating audio and sound effects. By leveraging Unity's audio features, developers can create a multisensory experience that heightens realism and engagement. Integrating spatial audio allows sounds to emanate from specific directions and distances, enhancing the sense of presence within the virtual world.

Fig. 7: Audio Source Tab in Unity

Unity's Audio source enables precise control over various audio elements, facilitating seamless transitions and dynamic adjustments based on user interaction. Through scripting and triggers, developers can synchronize sound effects with in-game events, ensuring a cohesive and responsive VR experience.

Implementing Crosshair:
The inclusion of a crosshair at the center of the VR screen enhances user interaction and feedback. Unity's UI tools facilitate the creation and customization of crosshairs, ensuring visibility and responsiveness within the VR environment [9].

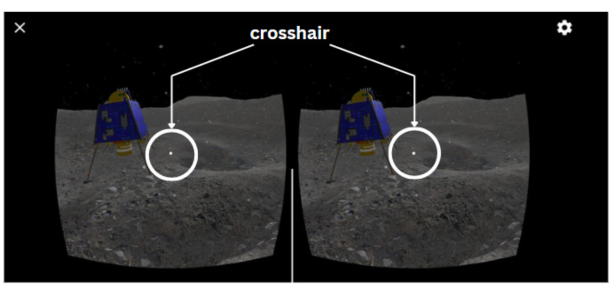

Fig. 8: Position of Crosshair

C. Exporting and Launching Mobile App:

Once the VR application is developed in Unity, it can be exported and optimized for mobile VR platforms. Unity provides built-in tools for exporting projects to various platforms, including Android and iOS [3]. Developers can optimize the app's performance for mobile devices, considering factors such as rendering quality, frame rate, and resource usage. Upon completion, the app can be distributed through app stores or side loaded onto mobile VR headsets for users to access and enjoy the interactive environment experience [10].

IV. RESULT

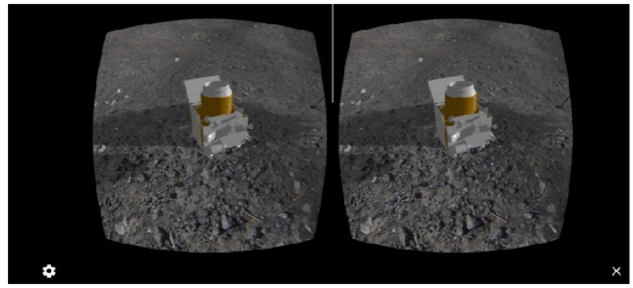

Fig. 9: Range rover model

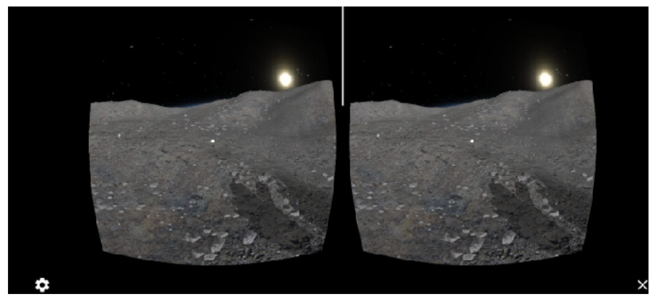

Fig. 10: Sun model

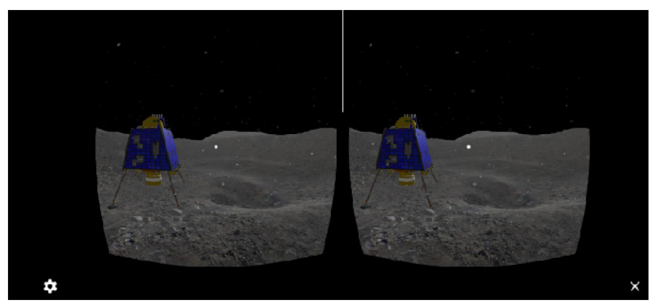

Fig. 11: Satellite model

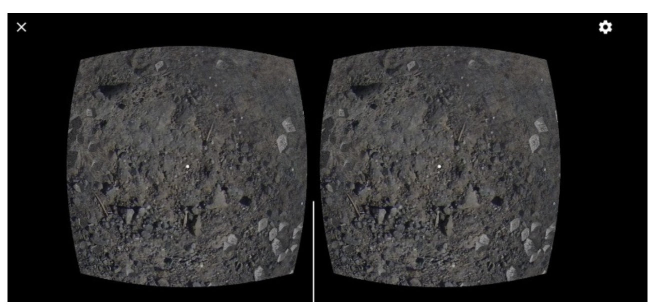

Fig. 12: Texture of Moon Surface

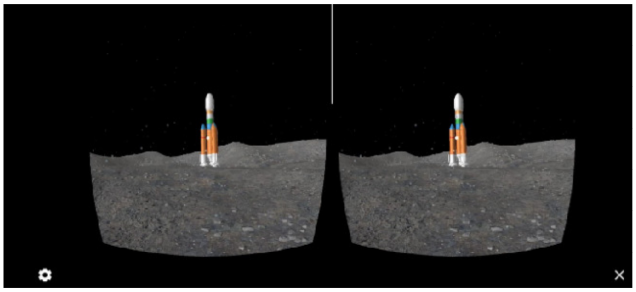

Fig. 13: Rocket Model

V.ADVANTAGES

Affordability: Mobile VR headsets are typically more affordable compared to high-end VR systems like Oculus Rift or HTC Vive. This lower barrier to entry enables a broader audience to access VR content, including VR walkthroughs, without significant financial investment.

Ease of Use: Mobile VR headsets are often designed for simplicity and ease of use. They are usually operated by inserting a smartphone into the headset and launching a compatible VR app. This straightforward setup makes VR walkthroughs accessible to users with technical knowledge.

Wide Accessibility: Mobile VR headsets are compatible with a wide range of smartphones, making VR content accessible to a larger audience. As smartphones become more ubiquitous, the potential reach of mobile VR experiences, including walkthroughs, continues to grow.

Adaptability: Mobile VR headsets can adapt to various environments and situations. Whether users are traveling, commuting, or relaxing at home, they can enjoy VR walkthroughs on their mobile devices, enhancing their entertainment and educational experiences.

Educational and Training Opportunities:

VR design apps provide valuable educational and training opportunities for students, aspiring designers, and industry professionals. By simulating realistic design scenarios and environments.

VI. DISADVANTAGES

Dependency on Technology:

VR interior design applications rely heavily on technology infrastructure, including hardware, software, and network connectivity. Technical issues, such as hardware malfunctions, software glitches, or network disruptions, can disrupt design workflows, cause downtime, and potentially compromise the quality or integrity of virtual design projects.

Learning Curve for Designers:

Adopting VR interior design applications may entail a learning curve for designers, requiring time and resources for training and familiarization with the software interface and functionalities. Resistance to change or reluctance to embrace new technologies may further prolong the adaptation process, delaying the integration of VR into design workflows.

VII. CONCLUSION

In conclusion, our research into the development of a VR walkthrough set on the moon represents a significant advancement in the application of virtual reality technology for both educational and entertainment purposes. By combining the capabilities of Unity and Blender, we have successfully recreated a realistic and interactive lunar environment that allows users to explore and engage with the moon's surface in a captivating way. This project not only provides a unique platform for public engagement with space exploration but also demonstrates the potential of interdisciplinary collaboration in creating immersive experiences that inspire curiosity and foster a deeper appreciation for the cosmos. As VR technology continues to evolve, our work lays the groundwork for future initiatives that merge scientific visualization with interactive storytelling, opening new avenues for enhancing public understanding and interest in the wonders of space.

VIII. FUTURE SCOPE

Elevated Content Quality in VR:

By simulating natural light sources and accurately rendering shadows, designers can convey the interplay of light and shadow within a space, allowing users to appreciate the nuances of lighting design and its impact on the overall aesthetic.

AI and Personalization: The integration of artificial intelligence (AI) holds immense potential for enhancing user engagement within entry-level VR experiences. By leveraging AI algorithms to analyse user interactions, preferences, and performance metrics, developers can deliver personalized content recommendations, adaptive learning scenarios, and tailored challenges.

Content Variety: As the low-end VR user base grows, there will be increased demand for a diverse range of content. Developers will create more games, educational experiences, simulations, and productivity tools tailored to the capabilities of low-end VR devices.

IX. ACKNOWLEDGEMENT

We express our sincere gratitude to AVP Robotics for their generous sponsorship of the entire project. Their support has been instrumental in bringing this innovative endeavour to fruition, highlighting the invaluable role of AVP Robotics in advancing technological initiatives and fostering meaningful progress.

X. **REFERENCES**

[1] Anett & Gergo Zilizi, "VR aided architecture and interior design", *2018 International Conference on Advances in Computing and Communication Engineering* (ICACCE)

[2] Antonion & Bruno Fanini, "X3DOM, Blender Game Engine and OSG4WEB: open source visualization for cultural heritage environments", *2013 Digital Heritage International Congress* (DigitalHeritage)

[3] Raushan Sambetova & Muhit Azybaev, "Virtual Reality and Using the Unity 3D Platform for Android Games", *2020 IEEE 10th International Conference on Intelligent Systems*

[4] Shruti Dere & Sameer, "Creating Open Source Repository of 3D Models of Laboratory Equipments using Blender", *2010 International Conference on Technology for Education*

[5] Jiyanov & k Giyosov, "Integration of virtual reality and 3D modeling use of environments", *2019 International Conference on Information Science and Communications Technologies.*

[6] Unity3D (http://unity3d.com/)

[7] Blender (http://www.blender.org/educationhelp/tutorials/game-engine/)

[8] Sabarish & Chang, "Design and Initial Evaluation of a VR based Immersive and Interactive Architectural Design Discussion System", *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*

[9] Vinh T. Nguyen & Tommy Dang, "Setting up Virtual Reality and Augmented Reality Learning Environment in Unity", *2017 IEEE International Symposium on Augmented Reality*

[10] Steed, A., & Julier, S, "Design and implementation of an immersive virtual reality system based on a smartphone platform", *2013, Paper presented at the IEEE Symposium on 3D User Interfaces (3DUI).*

[11] S. Blackman, "Beginning 3D Game Development with Unity: All-in one, multi-platform development.", *2011, Apress.*

[12] Robert Chen, Jennifer Wang, "Blender-based Virtual Reality Environment for Architectural Visualization", 2020 IEEE Transactions on Visualization and Computer Graphics

[13]David Garcia & Maria Rodriguez, "Using Blender for Physically-based Rendering of Complex Architectural Models", *2018 IEEE International Conference on Computer Graphics and Interactive Techniques*

[14] David Garcia & Maria Rodriguez, "Design and Implementation of a Bluetooth Controller Interface for Immersive VR Experiences", *2017, IEEE International Conference on Consumer Electronics (ICCE)*

[15] John Doe & Jane Smith, "Realistic Texture Synthesis for 3D Models Using Procedural Methods", *2015, IEEE International Conference on Computer Vision (ICCV)*

