Experience Virtual College Campus Tour with Generative AI-Assisted Avatar in VR Mobile Application

Indra Kishor	Akshay Sharma	Abhay Pratap Singh Chauhan	Dheeraj Thagriya
Assistant Professor	Student	Student	Student
Poornima Institute of Engineering & Technology, Jaipur	Poornima Institute of Engineering & Technology, Jaipur	Poornima Institute of Engineering & Technology, Jaipur	Poornima Institute of Engineering & Technology, Jaipur

Abstract

Virtual reality emerges as an revolutionizing technology for immersive experience of world that is limited to Computer generated simulations. Due to its immersive real life like simulations , it has attracted big tech giants to get their hands dirty in the business of computer based simulations. In this paper , we discussed the concepts of embedding an Avatar into the virtual environment and wrapping it into a mobile application making the VR experience platform independent and to make the virtual tour of a college with an Avatar assisting the explorer by walking with them and interacting with them by real time communication possible on mobile application that will enhance the user experience and make the virtual exploration more accessible and engaging for a diverse audience As existing virtual tours are static and pre-recorded and require VR headset, our project makes the virtual tours more dynamic and interactive with an embedded digital element, an Interactive Navigation Panel allowing users to visit specific areas of their interest whether it is the classrooms, labs, library, hostel, etc. on a mobile application.

Keywords: Avatar, Language Processing, Lip-sync, Phonetics, Mobile application.

Introduction

A.   Introduction to Virtual Reality

Our understanding of the world is shaped by our

sensory and perception systems. While we commonly recognize five primary senses—taste, touch, smell, sight, and hearing—our sensory capabilities extend beyond these. In reality, humans possess numerous additional senses, including our sense of balance, among others. Alongside these senses, our brains engage in intricate processing of sensory information, resulting in a complex and diverse flow of information from our environment to our minds. Through the replication of sensory experiences, virtual reality (VR) technology immerses users in simulated surroundings, fostering a sensation of presence and involvement. VR simulates an alternate reality by utilizing a variety of sensory inputs, including sight, sound, and occasionally touch, based on the idea of sensory perception. Through the use of specialized technology, such as controllers and headsets, virtual reality (VR) allows users to interact and experience computer-generated surroundings as though they were real.

B.   Introduction to Avatar

An avatar is digital human element which is a personalized graphical illustration of a human. We can treat them as humans of the metaverse where user can explore the virtual world through this digital element.

C.   Introducing Lip-Sync

As avatar more like humans of metaverse they should speak and while speaking their lips should make movements or gestures just like human lips. Hence, Lip sync is a term in 3D-animation which make the animation sequences that

mimics human lips. Making the avatar more interactive and real.

D.   Introduction to 3-D Animation
3D animation is a process of making moving 3D objects.3D animation is the basis of the trendy words like Meta-verse. Lip Sync is a also a 3D animation technique.

E.   Introduction to Generative AI
Generative artificial intelligence refers to a group of
AI techniques and models designed to generate new information that is inspired by or similar to preexisting data. A wide range of media, including text, photos, audio, video, and more, can be used for this. To identify the basic frameworks and patterns in the data, these models are trained on large-scale datasets. This allows for the creation of new material that follows these pre-existing patterns. In our project , we are using OpenAI's pre-trained model gpt-3.5-turbo-instruct on which we just need to feed our data and give the model feedback about kind of responses we are expecting.

Literature Survey

Our survey, it reveals several noteworthy research works and innovations related to College Virtual Tours. These studies have explored several different methodologies, technologies and application in the domain of Virtual College Tours.

According to Triranna Widiyaningtyas (2018) presented Adaptive Campus Virtual Tour using Location-Based Services. This feature monitor user's location and find relevant places easily. They used K Nearest Neighbor (KNN) algorithm to map the nearest location in a campus. In order to provide more useful information, this application also associates the user's location with the nearest event being implemented on campus. However they have generated this virtual tour using the images moreover making it static in nature.

According to Li Wang (2019) presented Design and Implementation of Three-dimensional Virtual Tour Guide Training System Based on Unity 3D they have developed a Virtual training system which provides a training environment for the trainers. However this training environment lacks of a 3D Mentor that can guide trainers.

According to N. Sripriya (2020) presented Speech-Based Virtual Travel Assistant For Visually Impaired developed a virtual guide that can assist visually impaired peoples to explore the world around them freely without any kind of help and assistance. It is useful more often when user may find textual interaction uncomfortable like when driving user finds textual interaction difficult. There future scope is that the user can be provided with a more personalized experience with implementation of deep learning techniques. Use of web scrapping techniques can provide more enhanced information about places. The bot can also be enabled to understand a more advanced queries which involve two nouns and provide the required result. For example, if a user asks for: "Find me a temple near a beach" - the bot should understand the intent as not just a temple search but also a beach nearby.

According to Ronmar E. Calingasan presented Into the Campus: Creating an Android-Based Interactive Virtual Tour of La Verdad Christian College – Apalit using Unity 3D has developed a Virtual Tour for there college that allow its user to experience the environment of the college. For this Virtual Tour the have developed a mobile application. There Future Recommendations are addition of a mini map in the Virtual Tour Part of the application, make the main character wear a uniform in the Virtual Tour, improvement of the collision detection, panning in the

3D Map and flexibility of the App to be deployed in iOS for Apple devices.

According to Indra Kishor (2023) presented Virtual Tour With Voice Assistant using Extended Reality has developed a Virtual Tour for the College Campus with real voice assistant service that is based on IoT concept. They had addressed the problem in hectic schedules, colleges are facing difficulties during visiting the college to the parents because all faculty members have their other work to do. With this problem, they have developed a Virtual Tour with unique voice assistant facility on the concept of IoT technology. There future recommendations is the addition of an avatar of faculty members. That will be added in labs, classrooms and faculty rooms. When any person will go to any of classroom, lab or faculty room, then the Avatar will interact and answer the question which will be asked by the person who tours college virtually, and the avatar will work on Self Learning feedback mechanism.

Proposed Work

In this paper, we are focusing on integrating an avatar into pre-built virtual environment in our case which is going to be a college campus. An avatar has features just like a human e.g. like lip movement while speaking, it also mirrors human body language during conversations enabling the college tour to be more interactive . Giving the tourist a feeling of a tourist guide assisting them. The tourist can engage the avatar in a conversation which is just like humans do with the help of Generative-AI. The conversation mainly includes information about the college like number of buildings, labs, faculties, college timings, placement statistics and various societies operating in college for holistic development of a student. In short, presenting the characteristics of college that contribute to its academic and social environment without any human presenting them.

Methodology

In this section , we will discuss the approach to achieve our proposed objectives. The brief of the approach is shown in figure given below.
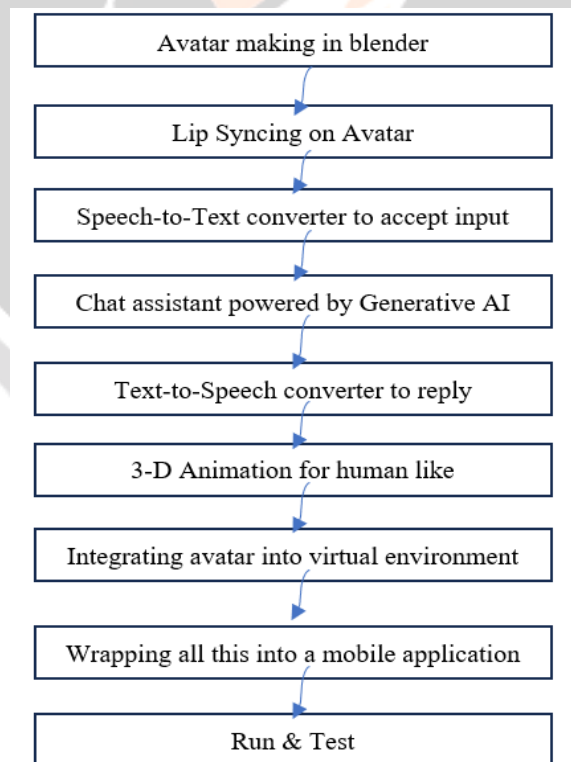

Fig. 1. Major steps to achieve proposed work

A. Creating Avatar in Blender 3D

Firstly, we enable "Rigging:Rigify" add on in blender. Click Edit > Preferences, search "Rigify," enable, and save to activate Rigify in Blender. Add human armature by Click Add>Armature>Human (Meta-Rig).Now, we select a Mesh, An object's defined shape is defined by its vertex, edge, and face collections, collectively referred to as a mesh. Every vertex in three dimensions is a point, and when they are joined by edges to form polygons—mostly triangles or quadrilaterals—the polygons comprise the object's surface. By Click Add> Mesh> Cube.

By selecting Cube in Sculpt Mode then Click Modifier> Add Modifier > Multiresolution > Subdivide so that we can make even minute changes as per our need. Place the mesh on the Human armature as shown.
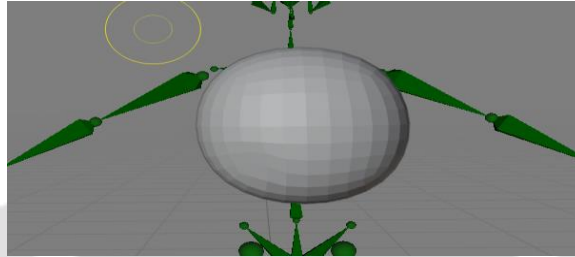

Fig. 2. Placing a mesh at the centre of armature

With the help of Grab brush in sculpt mode stretch the mesh so that encompassing the arms, legs and neck like this as shown in figure. Making the body of an avatar. Similarly, we can make fingers of hand and toes .
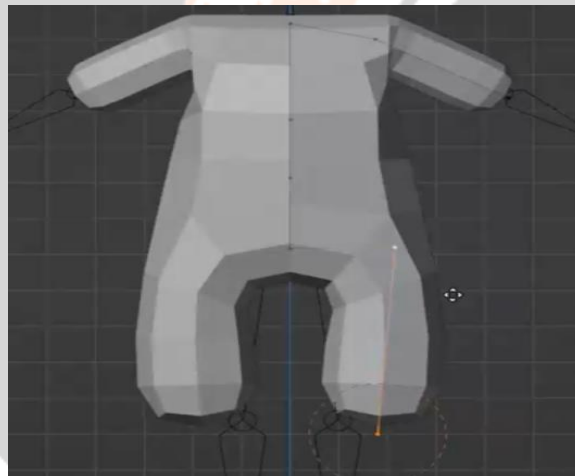

Fig. 3. Stretching a mesh to make arms and legs

To model face repeat the above steps till placing the mesh on the armature's head. Make the major landmarks of a face like nose , ears , eyes and lips. As nose and ears come out of the body we can make them with the help of Grab brush just like we have done when we made body of an avatar. For eyes we need to deform the mesh with the help of Grab brush pushing inward. For lips use Scrape brush and then smoothen it to make a lip. For a particular face we need to adjust the mesh accordingly.
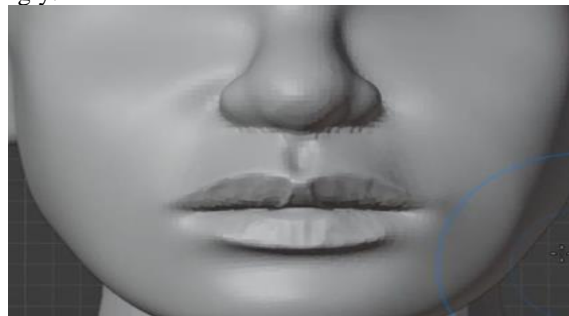

Fig. 4. Lips making using Scrape brush

Now, as our avatar body has been created we need to give the avatar body color. Enter edit mode, use vertex select or edge or face select mode to faces of the mesh to color. In the material properties tab , we can create a new material and will adjust the material setting such as base color, specular roughness to achieve the desired color. For giving colors for hairs and beared , we need to UV-unwrap our mesh to properly map the textures onto it.

We will be using some pre-built assets for clothes, shoes, jaw with teeth and tongue, eyes and adjust them to their position. Hence ,our avatar has been created.

Fig. 5. Final Avatar after color and texturing

B. Lip Sync on Avatar in Blender 3D

To make our avatar ready for lip sync we need to rig it first meaning we need to assign bones to the mesh which will allow us to pose and animate the avatar.

Rigging a Character

We have already inserted a human armature at the time of body modelling . Now we need to adjust the bones to align perfectly with our model. More specifically, we need to align the facial bones and bones of fingers and toes. In order to align them, three basic operations are move the bone, rotate around one end of bone and extension of one bone and the compression of the second connected bone.

Fig. 6. Rotation, Movement and extension of bones

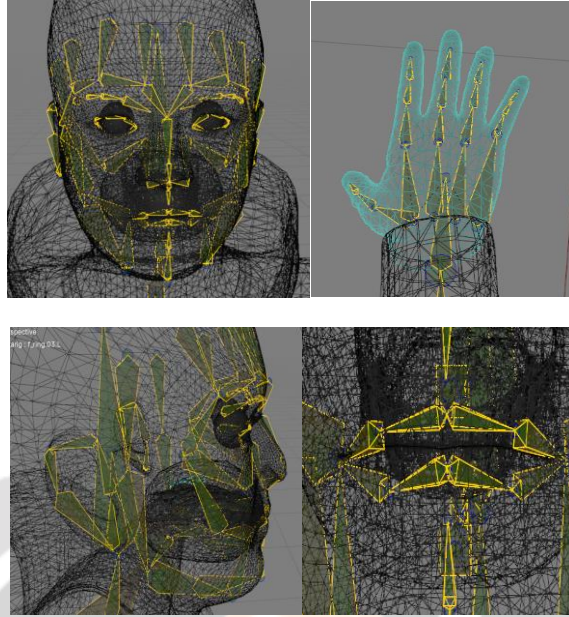With the help of these basic operations we can align bones of our avatar.

Fig. 7. Alignment of facial, hand and lip bones

Now, select avatar mesh and with shift pressing select the armature and CTRL+P and Choose "with automatic weights". This will assign the bones to the vertex group based on the position. Now , select the armature and go to Data and Click on "Upgrade Face Rig" button and wait for it and then Click on "Generate Rig" button. This will create a rig which we will use for animation. The white lines around an avatar that we are seeing are controls for the movement of body parts.

Fig. 8. Character Rigging

Lip Sync Animation

For creating the Lip Sync animation, we need to manually set poses for some very common phonemes like a ,ee ,er ,f ,g ,h ,i ,k ,l ,m ,n ,ng ,nk ,o ,oo ,or ,pr ,s ,x ,sh ,ch ,t ,th ,u ,v ,w ,y ,z ,zh ,j ,wh ,qu.

Fig. 9. Pose for phonemes (a) ee and (b) a

Map these phonemes with shape keys. As the audio will be generate from text which we will discuss in later section. We will use the C# script after importing the avatar into the virtual environment. Given below is the script to identify phonemes and then trigger the specific pose mapped with that phoneme.

```
using UnityEngine;
using System.Collections.Generic;
public class PhonemeAnimationController: MonoBehaviour {
    public Animator animator;
```

```csharp
    private Dictionary<string, string>      phonemeAnimationMap  =  new Dictionary<string, string>()
    {
      { "a", "Phoneme_A" },
      { "ee", "Phoneme_EE" },
      { "er", "Phoneme_ER" }
      // Add more phonemes and animation states as needed
    };
    public void TriggerAnimationClip(string phoneme)
    {
      if (phonemeAnimationMap.ContainsKey(phoneme))
      {
        string animationState = phonemeAnimationMap[phoneme];
        animator.Play(animationState);
      }
      else
      {
        Debug.LogWarning("No animation clip mapped for phoneme: " + phoneme);
      }
    }
    private void SimulatePhonemeDetection()
    {
      string detectedPhoneme = "a";
      TriggerAnimationClip(detectedPhoneme);
    }
    private void Start()
    {
      SimulatePhonemeDetection();
    }
}
```

C. Speech-to-Text Converter
```csharp
using System;
using System.Speech.Recognition;
class Program
{
  static void Main(string[] args)
  {
    SpeechRecognitionEngine recognizer = new SpeechRecognitionEngine();
    recognizer.SetInputToDefaultAudioDevice();
    recognizer.LoadGrammar(new DictationGrammar());
    recognizer.SpeechRecognized                              +=                              new
EventHandler<SpeechRecognizedEventArgs>(recognizer_SpeechRecognized);
    recognizer.SpeechRecognitionRejected                              +=                              new
EventHandler<SpeechRecognitionRejectedEventArgs>(recognizer_SpeechRecognitionRejected);
recognizer.RecognizeAsync(RecognizeMode.Multiple);

    Console.WriteLine("Listening for speech input. Press any key to exit...");
    Console.ReadKey();
    recognizer.Dispose();
  }

  static void recognizer_SpeechRecognized(object sender, SpeechRecognizedEventArgs e)
  {
    Console.WriteLine("Recognized speech: " + e.Result.Text);
  }
```

```
static void recognizer_SpeechRecognitionRejected(object sender, SpeechRecognitionRejectedEventArgs e)
   {
     Console.WriteLine("Speech recognition rejected.");
   }
}
```

D. Chat assistant powered by OpenAI API

Chat assistant powered by OpenAI means that we are using OpenAI models under the hood to generate human like responses on data which is provided by us. In our case , we will give information about college and it will generate responses on questions asked from the data that we have given. C# script to generate such responses is provided below.

```
using System;
using LangChain.DocumentLoaders;
using LangChain.Indexes;
using LangChain.LLMS;

class Program
{
   static void Main(string[] args)
   {
     Console.WriteLine("Enter your query:");
     string query = Console.ReadLine();
     string apiKey = Constants.APIKEY; Environment.SetEnvironmentVariable("OPENAI_API_KEY", apiKey);
TextLoader loader = new TextLoader("data.txt");
     VectorstoreIndexCreator indexCreator = new VectorstoreIndexCreator();
     var index = indexCreator.FromLoaders(new IDocumentLoader[] { loader });

     Console.WriteLine(index.Query(query));
   }
}
```

E. Text-to-Speech Converter
```
using System;
using System.Speech.Synthesis;
public class TextToSpeechConverter
{
   public static void TextToSpeech(string text, string outputFilePath)
   {
     SpeechSynthesizer synthesizer = new SpeechSynthesizer();
synthesizer.SelectVoiceByHints(VoiceGender.Female,                                        VoiceAge.Adult);
synthesizer.SetOutputToWaveFile(outputFilePath);
synthesizer.Speak(text);
synthesizer.Dispose();
Console.WriteLine("Text converted to speech and saved as: " + outputFilePath);
   }
 public static void Main(string[] args)
   {
     string textToConvert = "Hello, how are you today?";
     string outputAudioFile = "output.wav";
     TextToSpeech(textToConvert, outputAudioFile);
   }
}
```

F. Walking Animation Sequence

The following will be the states of walking animation sequence and will repeat on loop while walking. In blender we have defined the sequences. When we import the avatar in unity these animations will also get imported. The script is

defined to trigger the animation , when arrow keys are pressed is also given below.
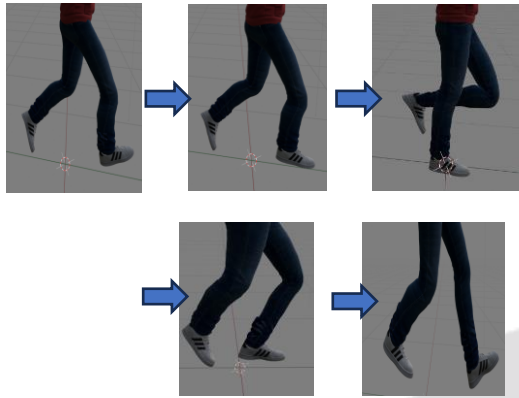

Fig. 10. Walking animation states

```
using UnityEngine;
public class CharacterController : MonoBehaviour
{
    private Animator animator;
    private void Start()
    {
        animator = GetComponent<Animator>();
    }
    private void Update()
    {
        float horizontalInput = Input.GetAxis("Horizontal");
        float verticalInput = Input.GetAxis("Vertical");
        if (verticalInput > 0)
        {
            PlayWalkingAnimation("Forward");
        }
        else if (verticalInput < 0)
        {
            PlayWalkingAnimation("Backward");
        }
        else if (horizontalInput < 0)
        {
            PlayWalkingAnimation("Left");
        }
        else if (horizontalInput > 0)
        {
            PlayWalkingAnimation("Right");
        }
        else
        {
            StopAnimations();
        }
    }
    private void PlayWalkingAnimation(string direction)
    {
        StopAnimations();
        string animationName = "Walk" + direction;
        animator.SetBool(animationName, true);
    }
```

```
    private void StopAnimations()
    {
       animator.SetBool("WalkForward", false);
       animator.SetBool("WalkBackward", false);
       animator.SetBool("WalkLeft", false);
       animator.SetBool("WalkRight", false);
    }
}
```

G. Import avatar into the virtual environment

Avatar is built in Blender 3D. Now we need to import in Unity. In Blender ,export your avatar as an FBX file make sure to check selected objects. In Unity, open the project and import this FBX file. Unity will import the avatar with its animation. They will be a part of avatar's Animation Controller. Configure the Rig to Humanoid. Position the avatar into the scene and Test

.

H. Wrapping into an application

The final step is to make a mobile application on which our college campus assisted with avatar can be accessed. For this we need to install the VR development tools for unity. Stream VR as we are using HTC vive headset. Optimize avatar's rig and animation for better performance in VR environment. Build VR application for the target platform.

Result and Discussion

We made a mobile application in which there is campus of a college in which Avatar will assist user with exploring the campus and resolve queries through speech. This project is an integration of various technologies which are different in principles but when implement to a specific objective doing wonders.

Fig. 11. Avatar in Library of College

Conclusion and Future Scope

We have made a VR application in which it provide the facility of touring a college campus virtually assisted with avatar as an instructor. This is a very small leap towards integrating VR technology to solve the interactivity of existing campus tours by integrating avatar assisted with Generative AI. The future scope of this project has a very broad spectrum it can be used in demonstration of infrastructural projects ,creating VR experiences, make learning possible by visualizing , reducing human workforce.

References

1.   T. Widiyaningtyas, D. D. Prasetya and A. P. Wibawa, "Adaptive Campus Virtual Tour using Location-Based Services," 2018 Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS), Batu, Indonesia, 2018, pp. 419-423, doi: 10.1109/EECCIS.2018.8692853.

2.   L. Wang and L. Wang, "Design and Implementation of Three-Dimensional Virtual Tour Guide Training System Based on Unity3D," 2019 International Conference on Communications, Information System and Computer Engineering (CISCE), Haikou, China, 2019, pp. 203-205, doi: 10.1109/CISCE.2019.00054.

3.   N. Sripriya, S. Poornima, S. Mohanavalli, R. P. Bhaiya and V. Nikita, "Speech-Based Virtual Travel Assistant For Visually Impaired," 2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP), Chennai, India, 2020, pp. 1-7, doi: 10.1109/ICCCSP49186.2020.9315217.

4.   Calingasan, R. E., Ramos, J. B., Espinosa, E. J., Casipagan, B. C., Alviar, S. C., Labung, S. T., ... & Natividad, R. L. (2019). Into the Campus: Creating an Android-Based Interactive Virtual Tour of La Verdad Christian College–Apalit using Unity 3D. *Innovatus*, *2*(1), 1.

5.   Kishor, I., Kumar, K., Sharma, A., & Bansal, H. Virtual Tour With Voice Assistant using Extended Reality.

6.   Dr.Senthil Kumar.M, 2Vigneshwaran K, 3 Veeraragavan S, 4 Vasanth T, VIRTUAL TOUR FOR COLLEGE VISITORS.

7.   Handore, A., Suryawanshi, Y., Shaikh, A., Ahire, U. and Patil, C., Virtual Reality Based Virtual Tour of College Using Unity 3D

8.   Cabibil, Nizle& Jr, Antonio &Obero, Kezia& Arroyo, Jan Carlo &Delima, AllemarJhone. (2020). MuseoDabawenyo: An Interactive Virtual Reality Museum Application using Unity. International Journal of Advanced Trends in Computer Science and Engineering. 9. 3556-3562. 10.30534/ijatcse/2020/161932020.

9.   K.Medrano, R. Tejada and B. González, "Development of a computer platform for the enrichment of cultural environments using VR and AR," 2021 IEEE Latin American Conference on Computational Intelligence (LA-CCI), Temuco, Chile, 2021, pp. 1-6, doi: 10.1109/LA-CCI48322.2021.9769849.

10. L. F. Aguas, R. Coral, F. D. Morales and R. M. Toasa, "3D Modeling for virtual tour of a Higher Education Institution," 2022 17th Iberian Conference on Information Systems and Technologies (CISTI), Madrid, Spain, 2022, pp. 1-5, doi: 10.23919/CISTI54924.2022.9820366.

11. Zhan T, Yin K, Xiong J, He Z, Wu ST. Augmented Reality and Virtual Reality Displays: Perspectives and Challenges. iScience. 2020 Aug 21;23(8):101397. doi: 10.1016/j.isci.2020.101397. Epub 2020 Jul 22. PMID: 32759057; PMCID: PMC7404571.

12. Xiong, J., Hsiang, E.L., He, Z., Zhan, T. and Wu, S.T., 2021. Augmented reality and virtual reality displays: emerging technologies and future perspectives. Light: Science & Applications, 10(1), p.216.

13. N. Elmqaddem, "Augmented Reality and Virtual Reality in Education. Myth or Reality?", Int. J. Emerg. Technol. Learn., vol. 14, no. 03, pp. pp. 234–242, Feb. 2019

14. Rojas-Sánchez, M.A., Palos-Sánchez, P.R., & Folgado-Fernández, J.A. (2022). Systematic literature review and bibliometric analysis on virtual reality and education. Education and Information Technologies, 28, 155 – 192.

15. Hussain, Afzal & Shakeel, Haad & Hussain, Faizan & Uddin, Nasir & Ghouri, Turab. (2020). Unity Game Development Engine: A Technical Survey. University of Sindh Journal of Information and Communication Technology.

16. Cosmina, Cosmina. (2018). A Glance into Virtual Reality Development Using Unity. Informatica Economica. 22. 14-22. 10.12948/issn14531305/22.3.2018.02.