

Development of virtual reality application for centrifugal pump impeller–machining process for training purpose

J. Kanchana^a, S. Sindhya^a

^aPSG College of Technology, Coimbatore, India

(Communicated by Madjid Eshaghi Gordji)

Abstract

The manual training of the new employees requires more time and labour work hours. Virtual reality and modern digital manufacturing techniques can be used to reduce the time required in this, thereby reducing the product cycle time. Virtual Reality (VR) provides a completely immersive and interactive experience of a simulated environment to the user. It helps in better understanding of the manufacturing process without using much of the employees working hours. The study aims at developing a Virtual Reality application for the machining process of a centrifugal pump impeller. This application would help the trainees in understanding the machining process involved and the process parameters. The factory setup and the impeller are modelled using Solidworks. The Virtual Reality application is developed using Unity 3D and played using Android smartphones and VR glasses. The application can be used as an alternative to the traditional training processes in industries.

Keywords: Virtual reality, Unity, Solidworks, VR glasses, Android smartphone

1. Introduction

The biggest asset of any industry are their trainees. They undergo an industrial training period of nearly two to three months initially. These training are provided by the well experienced or best employees of the industries. Though this training aims at providing complete knowledge of the industrial process to the trainees, they also result in the loss of the productive working hours of the employees. Virtual Reality and Modern Digital Manufacturing Techniques helps in the reduction of

Email addresses: kan@psgias.ac.in (Kanchana J), sindhya_psg@gmail.com (Sindhya S)

Received: August 2021 *Accepted:* November 2021

these losses. VR provides a complete immersive and interactive experience of a simulated environment to the user. The three main characteristics of VR are immersion, interaction and imagination. VR helps in better understanding of the process by the user with less or no help from others. The users can also view the process information repeatedly until they have understood the operations thoroughly.

The VR is widely used for training in many industrial and critical environments. Yu [9] explains about the application and importance of VR in various fields and the light baking, unwrapping, the materials and the textures used in his virtual application development. Wang [8] has used AutoCAD and 3ds Max for modelling and imported the models to the Unity 3D. The use of MySQL database helps the user to view the updated information in the virtual environment. Gupta [5] has integrated the Unity with Google VR SDK for the mobile VR application. The GvrViewermain, GvrEventSystem, GvrReticle, the use of Ray Caster and the Event triggers are used in the application for the syncing of the Unity with Google VR. Carlos [6] has explained the use of the User Interface for a better interaction with the user in the virtual environment. The author provides training in two steps, one is becoming familiar with the instruments used in the industry and secondly giving training in the required instrument through voice instructions. Andrea [4] explains the use of the prefab and the importance of them in developing a virtual environment. The sound effects and the rendering methods improves the user interaction. Cotelli [7] has used virtual characters (avatars) in the virtual application developed and they are used for the navigation in the virtual environment. David [3] has used a main menu scene in the application showing the different difficulty levels in the application developed. Zhou [10] has developed an education application for assembly and has used the scripts for the rotation and selection of the objects and the changing perspective. Raul [2] explains about the collision free motion and how they are established using coding for their VR application. Ahmad [1] has developed a virtual application for airplane turbine engine manufacturing and assembly process. This work aims at developing a virtual application for training the new employees in the machining processes involved in the centrifugal pump impeller using MobileVR. The application developed incorporates all the features like audio assistance, UI, teleport present in an advanced VR application into a more economic and less interactive application on a mobile.

2. Hardware and software systems

A. Hardware

The work presented here uses VR glasses in conjunction with the Windows 10 operating system. The VR glasses is used with mobile and no special sensors are required for this. The hardware requirements are listed in the Table 1.

Table 1: Hardware requirements for the VR application

Components	Specifications
CPU	Intel Core i5-4590
GPU	GeForce GTX 1060 or greater
RAM	16GB

B. Software

The VR application is developed using the Unity 3D game engine. The Unity version of 2018.4.13 has been used as it is found to be more stable than the previous versions. Unity Asset store is used for the download of the textures and the sound effects. The application is developed as a mobile VR, so an

android smartpone of version 5 or higher is preferred. The Google VR SDK is installed for android functionalities and features. The application developed is build as an apk file and installed in the smartpone. This apk file once build is a stand-alone application and does not require any additional software support, it can be installed in any android smartpones that meets the requirement.

C. Programming

The interaction required for the application are developed using the *C#* scripts written in Visual Studio Code/ visual studio code community version. The Unity is interfaced with the Visual studio as the coding environment and the *C#* libraries provided by the *C#*. NET framework are used while developing the codes.

3. Building virtual environment

A. Data Collection

The machining process of the centrifugal pump and the parameter details are gathered from PSG Industrial Institute. The impeller raw material is bought casted from suppliers and the minor operations are done in the institute. Factory model similar to that of the industry is to be developed and the layout details are collected.

B. 3D Model

The factory environment is the preliminary requirement of the virtual environment. The model of the factory is built using Solidworks. For initial operations only lathe is taken into consideration and so the lathe model is also built. The impeller is also modelled using the Solidworks. These models are saved as sldprt (Solidworks part files).

C. Importing models to Unity

The models saved as sldprt are to be imported into Unity as fbx (filmbox) files. The 3ds Max application is used for the conversion of the files from sldprt to fbx. The fbx files are imported as asset in unity.

D. Application development in unity

The Unity builds the project as a Desktop VR by default and so the build settings has to be changed initially. The android smartpone version to be used is specified in the player settings. The models once imported in unity are adjusted for its position and scale. The Unity scene on default has only directional light and a main camera. The lighting, material and texture settings are found to be essential for better user immersion. The default directional light provides shady effects at many areas of the model and therefore addition of lighting becomes essential in this case. A directional light is used at the other side of the default one followed by four point light sources for lighting within the factory as in the fig 1. The materials and the texture required for the walls and floor are collected from the unity asset store. The project window has four folders namely 3d models, prefabs, scripts and materials. The materials and textures required for the entire development is placed in the materials folder.

The lathe models are also positioned within the factory and for the lighting within the factory four point light sources are used. The impeller model is placed in the factory, positioned on a table. The VR glasses used does not have the controller, so a timer is used for selection. This timer is executed using a *C#* script. The script collects the pointer location and selects the object on completion of a 360degree timer completion. The script for selection of the event is:

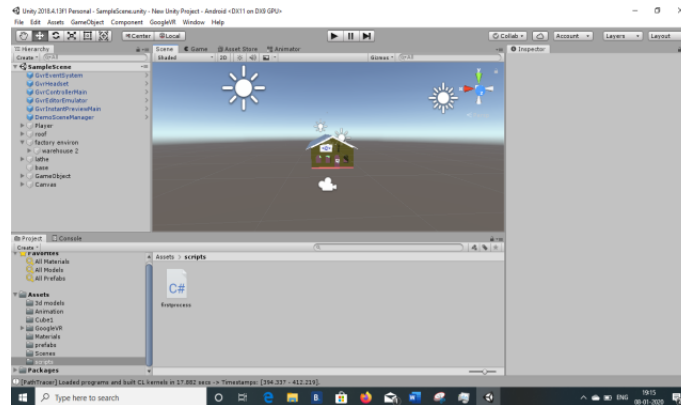


Figure 1: Factory model with lighting and textures in unity

```

pointerEventData = eventData as PointerEventData;
if (pointerEventData.pointerEnter != null)
    currentLookAtHandler =
ExecuteEvents.GetEventHandler<IPointerClickHandler>(pointerEventData.pointerEnter);

```

The Teleport option transports the user from his original position to the focussed position in the Virtual Environment. Four cylinders are created on the floor of the factory at different positions for teleporting as shown in the fig 2. Teleport is possible only among these four cylinders. Teleport script is referenced by these cylinders. The input for the script is the timer and the object pointed. If the timer has fully completed 360 degrees, then the current position of the user is shifted to the referenced teleport position only. The script

```
Player.position=currentLookAtHandler.transform.position +new Vector3(0,0.3568f,0);
```

Player.position=currentLookAtHandler.transform.position +new Vector3(0,0.3568f,0); moves the user from his current position to new position whose vector3 or position along 3 axis is given in the script.



Figure 2: Teleporting cylinders on the factory floor in unity

The impeller is placed on the table by adding features such as Mesh Renderer, Box Collider and Rigid Body in the Visual Inspector. The Mesh Renderer deals with the mesh filters, the material and textures that are to be added to the Impeller. The reflection probes and the shadow are also added using this feature. The Box Collider is a cube shaped, for collision detection that is, it has the

trigger event ability. The rigid body is used to provide the kinematic and gravity to the impeller. The mass and the drag details are provided in the rigid body feature for a realistic interaction. The Impeller has to be selected when the pointer is placed over it and it has to be picked from the table and placed on the chuck. These sequence of operation is done using the *C#* script. The script grabs the Impeller and when the impeller collides with the chuck of the Lathe, the Impeller is placed on the Chuck. Since a collision detection is necessary a Box Collider is added to the position of the Impeller on the Chuck. For easy operation a reference Impeller is placed on the chuck and this reference object is fitted with the Box Collider. The inputs for the script are the timer, current position of the gameobject and the final position of the gameobject. The selection of the gameobject is confirmed by the change of colour when gazed at. The grabbing of the object refers to completion of the timer and movement of the impeller along with the pointer. The placing of the object in the chuck is ensured by adding a name tag to the chuck.

Once the impeller has been fitted in the chuck, the operations are to be done. As per the details gathered from the industry, an outer diameter turning is done on the impeller. The turning operation is animated using the animator in unity. The user interface (UI) canvas is created with the start, Stop and OD turning text on the Lathe area for triggering the animation as shown in the fig3.

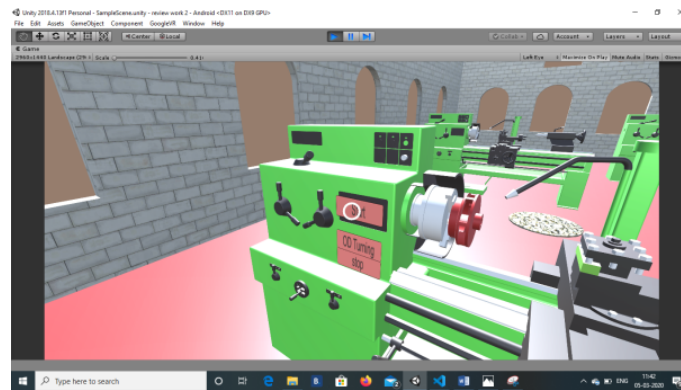


Figure 3: Game window displaying the UI on the Lathe

The UI is provided with the Timer for the selection of the buttons. The timer is set for one second. The Start button OnClick event activates the OD turning and the Stop Button. The OD turning Button OnClick event calls the Animation forlathe3 and also calls for a text which shows the machining parameters such as the speed, depth of cut and the number of passes. The Stop button shows the operation completion by reflecting the change on the outer diameter of the Impeller. The Impeller is split into two using the Solidworks to show the difference in the outer layer of the Impeller after the machining process. These split parts of the Impeller are again converted in FBX files before importing into the Unity. These split parts are assembled in the Unity and fit to the Lathe chuck as the reference position Impeller, which are made visible only on the completion of the animation and on the OnClick event of the Stop button.

The Audio is added for the easy guidance of the user. The Audio is added as a component from the Component Menu. The position of the Audio Source is set and the Audio clip needed for the audio are imported as a new asset into the Unity by creating a separate folder for Audio in the Project window. The initial Audio is entered in the Audio source as the Audio Clip.

A new scene is created in the project for showing the Main menu. In the main menu separate buttons for the Impeller machining and the lathe operation (Basic) are added as shown in the fig 4.

A UI canvas is created for the scene, the background is set for the scene. The UI has a text for the display of PSG pumps and then two buttons are added for the switching to the other scenes

created. The main menu with the button named as Impeller Machining takes the user to the impeller machining scene which has already been created.

The scenes are connected between them using the scene manager. A C# script is written to call the scene when the respective buttons are clicked.

```
SceneManager.LoadScene("SampleScene");
```

The scene named as 'SampleScene' is called using the scene management code.

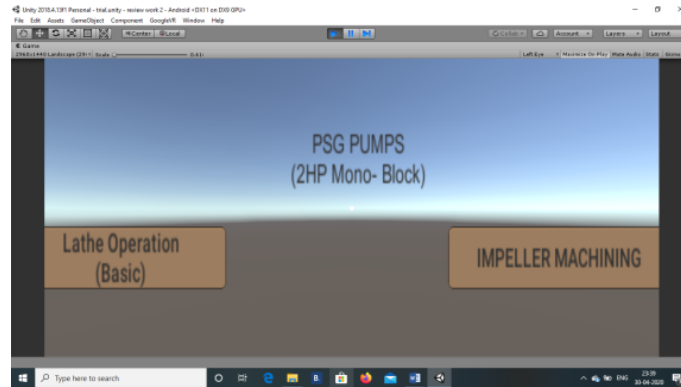


Figure 4: Main menu scene created in unity

Lathe basics a new scene is created within the project as shown in the fig 5. The scene only aims on educating the user regarding the basics of the lathe and how is it operated. The interaction is limited in this scene and an audio source is used. The user has to be directed back to the main menu scene after the current scene is viewed. This is done by using the Back to Main Menu button in both the scenes created. In Impeller machining scene, its placed on the Lathe UI and for Lathe Basic scene above the lathe.

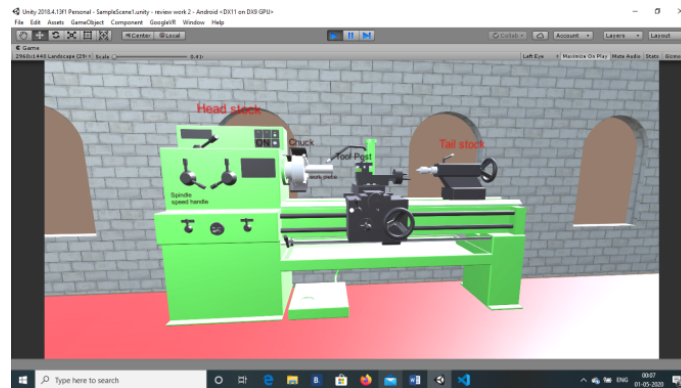


Figure 5: Lathe basic scene

The scenes are loaded based on their scene index as in the script:

```
SceneManager.LoadScene (sceneIndex - 1);
```

The different scenes created are to be added in the Build setting option in the same order of the Play as they are indexed accordingly. The scenes after creation are added in the Scenes to build option and the project is build. The build option now generates an apk file of the project which is saved in the desktop. This file is then sent to the mobile, where it is installed and viewed using the VR glasses.

4. Results and discussion

The main aim of the work was to develop a VR system. The VR application can be developed for different systems such as Windows OS, Android, Desktop, Web player using the 'build settings' option.

The fig 6 shows the result of the virtual reality application developed. Lightings and textures are adjusted for the real time effects.

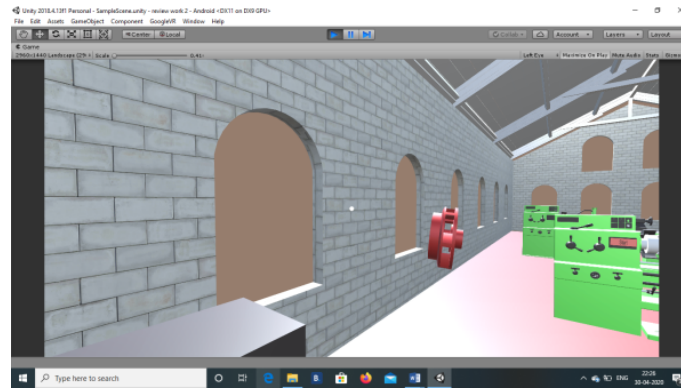


Figure 6: Impeller being picked and moved from its initial position

Upon opening the unity application on the mobile and viewed through the VR glasses, the user is presented with the main menu and two selections, one for the lathe basics and the other for the impeller machining. Since VR glasses does not have the controllers, the reticle pointers along with the timers are used for selection. Each scene is provided with a button to return back to the main menu scene. The lathe basics scene shows the major parts of the lathe and an audio is used to explain the user on the stepwise procedures to work on the lathe. This scene gives the trainee a basic idea of the lathe parts and the procedure in loading a work piece into the chuck, but here the interaction with the user is limited as only an audio source is used.

The impeller machining scene is provided with the audio assistance. The selection of the impeller from the table and the teleport option and placement of the impeller raw material in the chuck are done by the user itself under the guidance of the audio. The procedures can be repeated several times until the user is thorough with the operations. The process parameters are also displayed over the lathe when the machining animation plays. Here all the three characteristics of the VR, namely the immersion, interaction and imagination are used. This would help the trainees in better understanding the machining processes of the impeller.

5. Conclusion

Mobile VR applications has found its place in most of the industries because of its ease of operation compared to other VR applications. They are economical and best suitable for training purposes. The main focus of the work was to develop a Mobile VR application for the machining process of the centrifugal pump impeller on Unity 3D. The VR system is interactive and compatible with most of the Android mobiles. Since it is a game engine, the features like animation and modelling were in built without having the need to depend on other softwares. The animation and the audio added to the models give a more immersive effect in the virtual environment.

A. Scope for future work

The use of controllers would provide better interactive environment between the user and the simulated environment. The aesthetic and materials can also be added to the virtual models for a better visualization and feel of the parts. The physics of the models (kinematics, mass, gravity) can be understood better by the use of controllers.

Acknowledgement

We would like to acknowledge Mr. Saranraj, ASM technologies, Bangalore for his invaluable support and guidance in developing the application.

References

- [1] A. Ahmad, A. Rahman and M. Usman Aslam, *Virtual assembly of an airplane turbine engine*, IFAC paper online. 48(3) (2015) 1726–1731.
- [2] R. Crespo, R. Garcia and S.I Quiroz, *Virtual reality application for simulation and off-line programming of the Mitsubishi movemaster RV-M1 robot integrated with the oculus rift to improve students training*, Procedia Comput. Sci. 75(2015) 107–112.
- [3] D. David, E. Edwin, E. Arman, H. Hikari, N. Chandra and N. Nadia, *Development of escape room game using VR technology*, Procedia Comput. Sci. 157 (2019) 646–652.
- [4] A. de Giorgia, M. Romero, M. Onori and L. Wang, *Human machine collaboration in virtual reality for adaptive production engineering*, Procedia Manuf. 11 (2017) 1279–1287.
- [5] R. Gupta, R. Nawani and V. Talreja, *Building virtual reality applications with bluetooth controller interaction for google cardboard using unity 3D*, Int. J. Adv. Res. Comput. Sci. Eng. Inf. Technol. 2017.
- [6] C.A. Garcia, J.E. Naranjo and A. Ortiz, *An approach of virtual reality environment for technicians training in upstream sector*, IFAC-Pap. 52(9) (2019) 285–291.
- [7] D. Mol Machado, A. Cotelli, D. Galvao and A.C.A. Mol, *Use dosimetry virtual tool for security studies physics and nuclear*, Procedia Manuf. 3(2015) 1765–1771.
- [8] S. Wang, Zh. Mao, Ch. Zeng, H. Gong, Sh. Li and B. Chen, *A new method of virtual reality based on unity 3D*, 18th Int. Conf. Geoinformatics, 2010, pp. 1–5.
- [9] X. Yu, *Research and practise on application of virtual reality technology in virtual estate exhibition*, Procedia Eng. 15(2011) 1245–1250.
- [10] Y. Zhou, Sh. Ji, T. Xu and Zi Wang, *Promoting knowledge construction: A model for using virtual reality interaction to enhance learning*, Procedia Comput. Sci. 130(2018) 239–246.