Acta Mechanica Slovaca **29 (4): 50 - 53,** December **2025**
https://doi.org/10.21496/ams.2025.029

Acta Mechanica Slovaca
ISSN 1335-2393
www.actamechanica.sk

Implementation of Audio Guidance in Unity VR Simulations

Erik Varjú [1,*], Juraj Kováč [1]

[1] Technical University of Košice, Faculty of Mechanical Engineering, Department of Industrial and Digital Engineering, Park Komenského 9, 042 00 Košice, Slovak Republic

Abstract: This article presents a guided virtual reality (VR) simulation for the assembly of a robotic arm, developed in Unity 3D using the VR Builder framework. The simulation is designed for educational and training purposes, targeting engineers and technical professionals seeking an interactive understanding of mechanical assembly processes. It runs on the Meta Quest 3 headset in full VR mode, using controllers for object interaction. The primary aim is to demonstrate how immersive environments can enhance comprehension of part functionality and assembly order through intuitive interaction and audio guidance. The simulation consists of a short assembly sequence involving the basic components of a robotic arm. As the user progresses through the steps, audio instructions are triggered automatically to explain each part's name, purpose, and placement. These instructions are generated using text-to-speech tools, ensuring clarity and consistency. Although the simulation covers a simplified process, it serves as a proof of concept for more complex systems with larger assemblies. By combining spatial interaction with contextual audio cues, the simulation offers an engaging training method that can be expanded for industrial or educational use. The project also explores the practical potential of VR-based instruction for environments where hands-on access to real equipment is limited. Future development will focus on integrating hand-tracking to provide a more natural and accessible user experience.

Keywords: virtual reality; unity 3D; audio guidance; technical training

1. Introduction

Virtual reality (VR) has become an increasingly important tool in industrial training, offering safe, cost-effective, and immersive environments where operators can practice assembly, maintenance, or troubleshooting tasks without real-world risks. While many VR simulations focus on accurate 3D modeling and interactive functionality, an equally critical element for effective training is audio guidance. Clear and well-structured instructions can significantly improve the learning experience by providing immediate feedback, guiding users step by step, and reinforcing the meaning and purpose of individual components [1].

The importance of audio guidance becomes even more apparent when considering the diversity of trainees. Not all operators share the same background knowledge, learning speed, or familiarity with the machinery. Visual instructions alone may be insufficient, especially in complex assemblies where understanding both procedural steps and the functional roles of parts is essential. Audio guidance helps overcome these barriers by combining instruction with explanation, ensuring that users not only know what to do but also why each step matters [2].

Despite these advantages, implementing audio instructions in VR simulations is not always straightforward. Developers must choose between generating speech dynamically, which ensures flexibility but may be limited in voice quality, or preparing

*** Corresponding author:** Erik Varjú, ***E-mail address:*** *rerik.varju@tuke.sk*

pre-recorded audio, which allows for greater customization but increases production time. This article presents two practical approaches to integrating audio guidance into Unity VR projects, using a robotic arm assembly simulation as a case study. The goal is to demonstrate how structured audio can enhance training effectiveness and serve as a foundation for larger, more complex applications in the future [3].

2. Methods of Implementation

The integration of audio guidance into VR simulations requires a clear design strategy to ensure that instructions are delivered at the right time and in the right format. In Unity, there are multiple ways to achieve this, but in practice, two methods stand out as practical and accessible for developers: generating audio dynamically with the operating system's built-in text-to-speech engine, or pre-generating audio files externally and importing them into the project.

Both methods follow a similar overall workflow: a step or event is completed within the simulation, which triggers the playback of the corresponding audio instruction. The difference lies in how the audio is created and managed. The built-in text-to-speech option prioritizes speed and flexibility, while the external MP3-based approach prioritizes quality and customization.

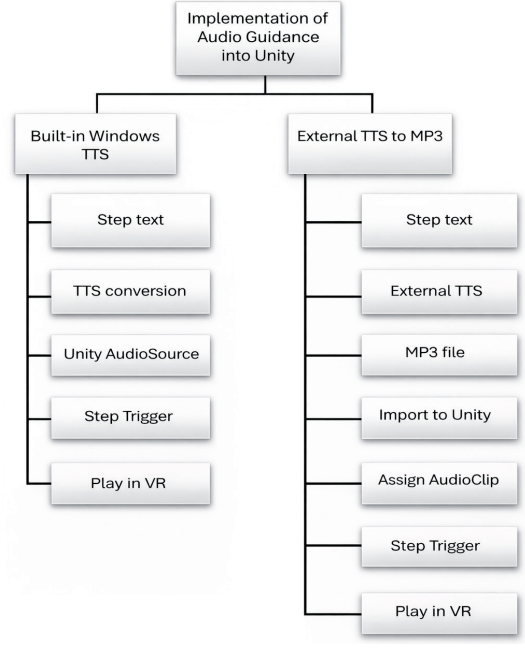

Figure 1: Workflow of audio guidance implementation methods

To illustrate the decision-making process and workflow, Fig. 1 presents a flowchart comparing the two implementation strategies. This diagram highlights the parallel steps and the points at which the two approaches differ.

2.1 Windows Built-In Text-to-Speech (TTS) Integration

The first approach uses the built-in text-to-speech functionality available in the Windows operating system. In this method, the text of the instruction is written directly in the Unity project (Fig. 2). When the user completes a step in the VR Builder framework, the text is sent to the Windows TTS engine, which generates speech dynamically. This audio is then played through an AudioSource in Unity [4].

– *Advantages: Quick to implement, flexible, no need to pre-record audio. Any changes to instructions can be updated instantly by editing text.*

– *Limitations: Limited voice quality and naturalness; availability of voices depends on the operating system; requires the application to run on Windows.*

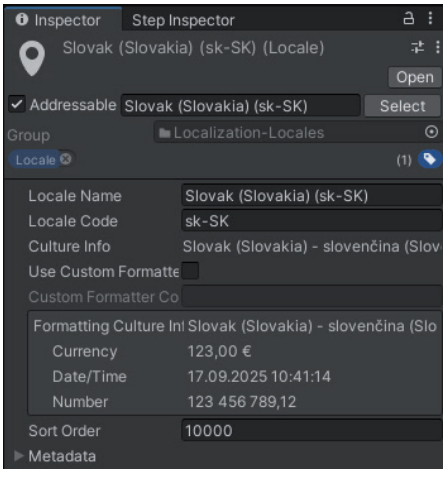

Figure 2: Text-to-Speech Settings in Unity

2.2 External TTS to MP3 Integration

The second approach uses external text-to-speech software or online services to generate speech from text. The resulting audio files are saved in MP3 format, named according to their corresponding steps, and imported into Unity (Fig. 3). Each MP3 file is linked to an AudioSource and triggered automatically when the step is completed in VR Builder [5].

– *Advantages: High-quality voices, customizable tone, speed, and language options. The resulting audio can be reused across projects.*

– *Limitations: Requires additional time to generate and manage audio files. Editing instructions involves regenerating audio files.*

Both methods are compatible with the step-and-chapter structure of VR Builder. The system ensures that once a step is completed, the corresponding audio instruction is triggered automatically, providing immediate guidance for the next action [6].

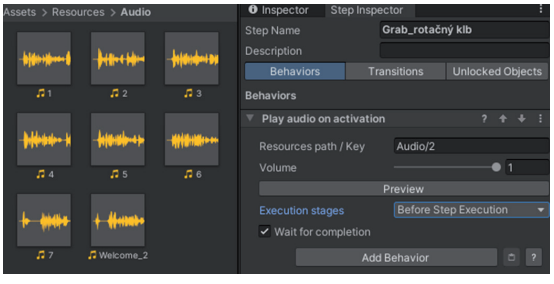

Figure 3: MP3 Integration into Unity

3. Practical Demonstration in Unity VR

To demonstrate the two approaches to audio implementation, a short VR simulation was created in Unity, focusing on the basic assembly of a robotic arm (Fig. 4). The project was developed using the VR Builder framework, which structures the simulation into steps and chapters. Each step represents a single action performed by the operator, and once the step is completed, the corresponding audio instruction is triggered automatically [7].

The virtual environment was designed for the Meta Quest 3 headset and is fully immersive, using controllers for interaction. At the start of the simulation, the operator teleports to the designated workspace, where all robotic arm components are placed on a table. Each part is labelled with a virtual name tag to support recognition and orientation. As the operator proceeds through the steps, audio

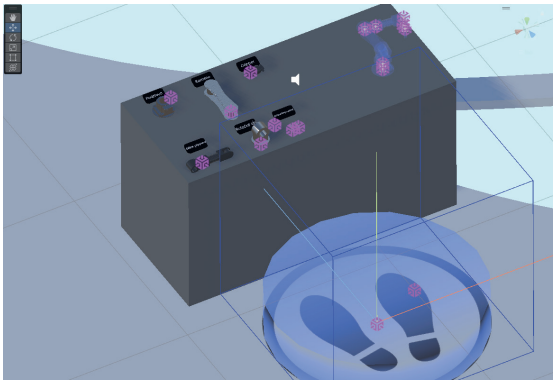

Figure 4: Basic Assembly of a Robotic Arm

instructions provide both procedural guidance (e.g., which part to pick up, where to place it) and conceptual explanations (e.g., the function of the part, why it is positioned at a specific location).

The robotic arm simulation is structured as a sequence of individual steps, each representing a specific action in the assembly process. Users must complete each step before the corresponding audio instruction is triggered, ensuring that guidance is delivered at the correct moment and reinforcing correct task execution. This behaviour is consistent across both audio implementation methods, whether using built-in Windows text-to-speech or external pre-recorded MP3 files.

Steps are organized in a logical order, with each one connected to the next, forming a continuous workflow. The step connections ensure that instructions are delivered sequentially, preventing users from skipping actions and supporting a clear learning path. Fig. 5 illustrates how the steps are arranged and linked within Unity using VR Builder, showing the automatic triggering of audio as user's progress through the simulation.

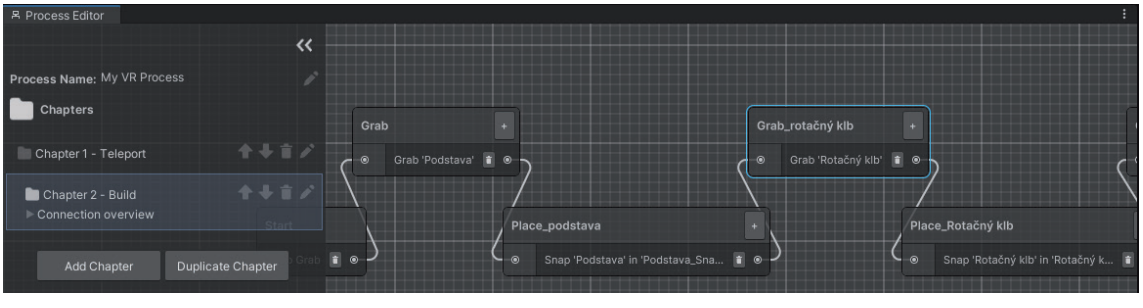

Figure 5: Sequence of Steps

4. Conclusions

This article presented two practical approaches for integrating audio guidance into Unity-based VR simulations, demonstrated through a robotic arm assembly scenario. Both methods—using built-in Windows text-to-speech and external pre-recorded MP3 files—enable stepwise audio instructions that provide users with procedural guidance and conceptual understanding of component functions. By linking audio playback to the completion of specific steps, the system ensures that trainees receive instructions at the right moment, enhancing learning efficiency and comprehension.

The case study illustrates that even a relatively simple simulation can serve as a scalable framework for more complex industrial applications, including multi-part machinery or larger training programs. The structured step-and-chapter approach, combined with automated audio, demonstrates how VR simulations can bridge the gap between visual interaction and conceptual learning.

Acknowledgments

This paper was supported by the projects APVV-17-0258, APVV-19-0418, VEGA 1/0508/22, VEGA 1/0383/25, KEGA 020TUKE-4/2023, KEGA 003TUKE-4/2024.

References

1. D. Scorgie, Z. Feng (2024). Virtual reality for safety training: A systematic literature review and meta-analysis, Safety Science, [Online] Available, from https://doi.org/10.1016/j.ssci.2023.106372
2. Kaplan, A. D., Cruit, J., (2021). The effects of virtual reality, augmented reality, and mixed reality as training enhancement methods: A meta-analysis. Human Factors, [Online] Available, from https://doi.org/10.1177/0018720820904229
3. Latoschik, M. E., Kern, F. (2019). Not Alone Here?! Scalability and User Experience of Embodied Ambient Crowds in Distributed Social Virtual Reality. IEEE Transactions on Visualization and Computer Graphics, [Online] Available, from https://doi.org/10.1109/TVCG.2019.2899250
4. Unity Technologies. (n.d.). AudioSource. Unity Scripting API. [Online] Available, from https://docs.unity3d.com/ScriptReference/AudioSource.html
5. Microsoft. (n.d.). System.Speech.Synthesis namespace. Microsoft Docs. [Online] Available from https://docs.microsoft.com/en-us/dotnet/api/system.speech.synthesis
6. MindPort. (n.d.). VR Builder: User manual. [Online] Available from https://www.mindport.co/vr-builder-manual
7. Unity Technologies. (n.d.). Introduction to audio. Unity Learn. [Online] Available, from https://learn.unity.com/tutorial/introduction-to-audio