

CHAPTER

4

GESTURE INTERACTION IN 3D HOLOGRAM

Ajune Wanis Ismail, Fazliaty Edora Fadzli, and Muhammad Akma Iman

4.1 INTRODUCTION

Holograms provide an interesting way to display and convey information. Holograms have been improved each year to enhance user interaction. This is because interaction is important as it provides benefits to the user's interaction with virtual objects. Gesture interaction has become a recent topic, as it allows a user to use their bare hands to directly interact with a hologram. Therefore, there remains an issue regarding how real hand gestures can be well-suited to a hologram application. Therefore, this chapter discusses the development process and implementation of three-dimensional (3D) object manipulation using natural hand gestures in a hologram. It describes the design and development process for a hologram application and integrates it with real hand gesture interaction.

Hologram involves storage, recording, and subsequent reconstruction of both the phase and amplitude of the light field scattered by the object (Bernardo et al., 2018). In general, a hologram helps to convey project information in the form of light. Hologram elements such as mirrors, lenses, and directional diffusers can implement lights on a single material based on the frequency characteristic. Thus,

holograms can be used in various types of media or devices, such as hologram memory, hologram projection screens, head-mounted displays (HMDs), holographic printers, and more.

A hand gesture is a complex and non-rigid target with some dynamic components. Therefore, fingertips, thumbs, or palms are usually sub-resolution limits of a conventional device. Hence, the conventional device must be expensive, large, or complex to recognize the small motions. However, despite the hardware's ability to track minuscule motions, the computational limitations of the tracking algorithms hinder a gesture recognition application. Command signals from the user's hand is transmitted to an external device, such as a robotic arm, in a remote plant. Users only use a hologram that is representative of the physical object, with less direct interaction with the hologram object. Furthermore, a conventional input gesture, such as a keyboard in the hologram, is used to manipulate the images. This type of gesture is non-intuitive. Hence, this chapter intends to explore the implementation of hand gestures in a hologram so users can easily command the 3D object on the hologram with a natural sense

4.2 ARCHITECTURE FRAMEWORK

This section explains the system overview for the development stage. Unity has been used to develop the application. The system consists of a hologram screen, 3D object data in .fbx format, and a tracking technique for hand gestures. The tracking technology is based on a motion-based device using Leap Motion.

This system overview, as shown in Figure 4.1, provides a list of interactions such as "Selection", "Spawning", "Rotation", "Translation", and "Release". The model we used underwent the modeling process, and some of the other models required textures. The following subsections detail the system overview.

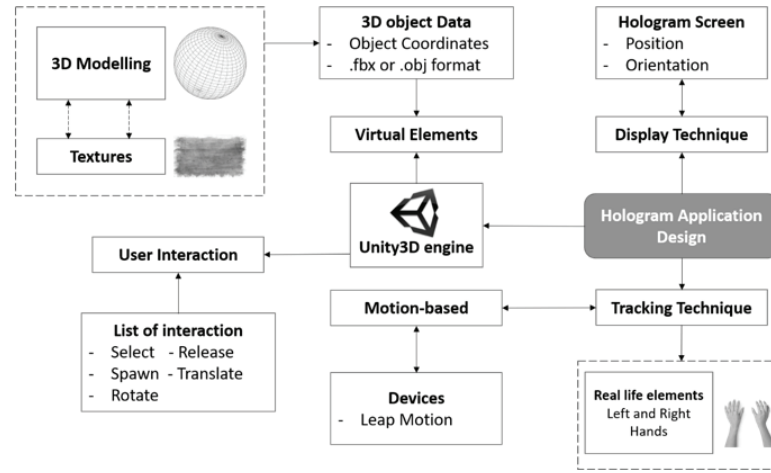


Figure 4.1 System component and overview

Figure 4.2 illustrates the architecture framework that consists of three phases. It is a structured approach, progressing through the subsequent stages. The first phase involved studying how natural hand gestures could be integrated into holographic interactions. This included investigating manipulation strategies for virtual objects using hand gestures and evaluating user interaction. The Leap Motion Device, along with its software development kit (SDK), was used to implement gesture control. The SDK provides features that enable accurate tracking of real hand movements. This phase explores the process of integrating natural hand gestures in hologram-based applications.

Building upon the findings from Phase 1, this phase focused on developing interactive 3D object manipulation techniques such as selection, translation, and rotation. Gesture metaphors were designed for intuitive interactions—such as tapping, pinching, and rotating—to enable users to engage naturally with virtual objects. The Leap Motion Controller captured real-time hand movements and provided depth data through its motion tracking system. Unity3D was selected as the development platform due to its compatibility with Leap Motion and its robust feature set. This phase successfully enabled users to perform interactions such as selecting, translating, rotating, spawning, and releasing virtual objects using hand gestures.