VR-Based Visualization for Biological Networks

[1]Ahmed Hisham, [2]Basim Mahmood

[1]Department of Computer Science, University of Mosul, Mosul, 41002, Iraq
[2]Bio Complex Laboratory, Exeter, UK

Abstract - **The current revolution in technology forces most daily life systems to generate large-scale data. Understanding such data is not an easy task due to its size and complexity. Therefore, data visualization has played a key role in understanding data more clearly. Moreover, most of the available approaches for data visualization are considered traditional that deal with data in a 2D way. Furthermore, one of the most modern methods for data visualization is converting data into network models that include nodes and edges. This kind of visualization has been proven as an efficient method of analyzing large-scale data. On the other hand, with the advent of virtual reality technology and the metaverse concept, data can be visualized and analyzed in an interactive and 3D way. However, there is a lack in the literature providing enough information on how to generate such 3D models. This thesis comes to investigate the integration of virtual reality technology and complex networks, and how this integration can generate interactive data visualization models. To this end, a 3D VR-based interactive biological network model is designed and implemented to be adoptable by the metaverse platform. The biological network represents the gene-gene interactions of the sex chromosomes. The developed model enables analysts to dive into data and interact with its objects, which lead to a more professional, deep, and accurate analysis. The model proved its efficiency in terms of usability when tested by experts.**

Keywords: Complex Networks, Biological Networks, Virtual Reality, Data Visualization, Interactive Visualization, Metaverse.

I. INTRODUCTION

The current technological era has witnessed a noticeable revolution in developing systems and applications that generate large-scale data. Visualizing data techniques vary depending on the data itself and the depth of analysis. The analysis of some applications requires only descriptive statistics. On the other hand, other applications need more advanced analysis such that the use of machine learning, deep learning, or other advanced statistical analysis techniques. Moreover, the analysis of data is usually performed based on visualizing data in particular forms such as curves, histograms, boxplots, and pie charts, to mention a few [1]. These forms can be interpreted by data analysts or people of interest and can be used to present facts about a particular phenomenon.

Furthermore, one of the modernist techniques used in visualizing data is converting the data into a network model [2]. This kind of technique is based on concepts inspired by complex networks. A complex network is represented as a graph (G) with vertices (V) and edges (E). The generated model can be visualized using particular layouts of interest (e.g., isometric, circles, geo-layout, etc.). The analysis can also be performed using network science metrics. These metrics play a significant role in deeply understanding network models.

According to [3][4], complex network models are considered a powerful technique for analyzing big and complex structures of data. Therefore, the field of complex networks has thrived and attracted worldwide researchers. The regular techniques for visualizing network models usually performed in two-dimensional (2D) graphs. This kind of visualization is useful because a lot of information can be encoded in the network (e.g., nodes, edges, or both). For instance, node size can be used as an indicator of a particular fact about the data. Moreover, node color can be used to encode information about the data (e.g., representing particular communities). Likewise, Network edges in terms of color and thickness (weights) can be involved to encode information. The labels of nodes and edges represent another dimension that can encode information in it. Figure 1 depicts a network model that encodes a variety of information within its structure. The figure demonstrates nodes of different sizes and colors. The edges also have different weights and colors. These indicators enable researchers to understand their data clearly. In addition, network experts are able to explain the whole network model in almost one sight.
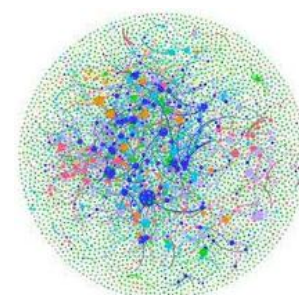

Figure 1: A regular network model that includes nodes and edges

The network model shown in Figure is displayed in a 2D dimension. This approach is frequently used in visualizing a wide range of applications such as biological networks [5], social networks [6], political networks [7], and transportation networks [8]. The problem with such a visualization technique is that when having giant network models in terms of the number of nodes and edges, the 2D visualization makes it difficult to understand the complex and dense connections among nodes. Therefore, researchers and data analysts try to use 3D visualization techniques [9]. As known, adding more dimensions to a shape, more information can be obtained about that shape. As a result, the 3D techniques enable network analysts to see more details about the data model (see Figure 2).
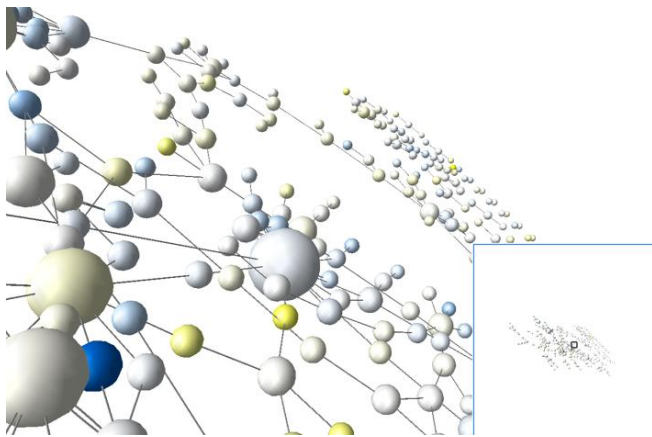
Figure 2: A 3D network model

Furthermore, with the advent of Virtual Reality (VR) technology, 3D visualization has attracted more attention from researchers [10]. This is because VR technology enables researchers, analysts, and designers to interact with their data. This important feature is highly useful in the case of models with high complexity. Yet, VR technology has been recently proven an efficient tool for data visualization in the literature [11],[12].

Many other fields have utilized the features of VR technology (e.g., education). The work of [13] investigated the differences between web-based and VR-based applications in education. Their study proved that the VR technology showed its efficiency during the learning process more than the other learning applications because of the interactive features that VR technology provides. Generally, the use of VR technology has recently thrived in implementing real-world problems such as in physics [14], military [15], astronomy [16], video gaming [17], tourism [18], architecture engineering [19].

1.1 Literature Review

Complex network visualization in 2D has a problem in understanding the intricate relations among data objects.

Therefore, it is feasible to visualize such networks using VR technology. The 3D visualization of complex networks can support analysis and answer questions about the intricate relations among nodes within the network. In this regard, one of the earliest studies that used VR technology for network visualization was [20], they visualized and explored social networks using VR tools. They showed that VR visualization is significantly more efficient than typical 2D visualization especially when it comes to social networks. A few years later, the idea of visualizing complex networks using VR technology was adopted in biological networks. For instance, [21] implemented the connections of the human brain in a VR environment. Their approach enabled researchers to interact with the model and investigated/tracked the connections of the human brain.

Furthermore, biological networks represent the most complicated models due to the complex nature of their data. For instance, gene-gene interaction networks, disease-disease interaction networks, and brain connections networks have a high-dense structure and it is not an easy task to analyze such network models. Therefore, some works in the literature tried to use VR technology aiming at mitigating the difficulties of the analysis process. The work of [22] developed an approach to visualizing genome networks using VR technology. The network they build consisted of nodes that are connected by edges. These edges are created when an interaction exists between two nodes. Moreover, the authors used the Unity software and coding in designing the VR system with the support of genome experts. Brain networks can also be visualized using VR technology.

Another work was performed by [23] who suggested a VR system for the gamification of chemistry tasks. They allowed researchers to track and interact with chemical reactions in real time. The VR system was tested by many users to ensure its efficiency. The study of [24] developed an interactive VR-based visualization of neural networks. Their visualization enabled researchers to control the parameters used in a neural network. The approach allowed users to use virtual hands to manipulate network parameters aiming to provide real-time evaluation of the output. The model made users able to virtually manipulate the input layer, hidden layer 1, hidden layer 2, or the output layer. Such controllability may support researchers in better adjusting their neural network models. However, the work is considered simple since some advanced neural network models (e.g., deep learning) need more nodes and more layers, which is more challenging. The visualization was performed using Python programming language and Unity.

In the literature, one of the distinguished examples was a system proposed by [25]. They represented brain regions as

nodes and the connections between the regions are considered the edges. The 3D visualization of the brain network took into account the anatomical context, which was important in the analysis of the model. The visualization imitated the real shape of the brain. The authors also involved eight expert participants to evaluate the system to prove its efficiency. In our point of view, when designing a VR-based system, it is crucial to involve participants who are experts in the application. This step represents a validation of the model and can improve the designed VR system based on the feedback. [26] used VR technology in visualizing a co-citation network. The VR model was interactive and enables users in interacting with the model. The interactions make it more efficient in understanding the model. [27] Designed a novel VR model for neuron connections in the brain that was user-friendly to support the learning process. The model was designed to be easy to interact with and provided high educational and learning values. Also, the model provided a lot of features that enabled learners to retrieve information in an easy and efficient way.

1.2 Problem Statement and Contribution

According to the literature, few studies consider the integration of complex networks and VR technology, which leads to a lack in the literature. The main reason behind this lack is that VR devices and software have not yet become popular in academic settings. Another lack in the literature that has been observed, there is no clear research work or guides on how to generate interactive VR-based network models, especially for the gene-gene interactions network. Although few works have been performed in the literature, they still have not publicly available and it is difficult to be accessed by researchers and developers. Furthermore, with the advent of the metaverse concept, the tendency of developers in building applications in such virtual worlds is not yet thrived. Therefore, it is required to strengthen this tendency and go through this contemporary topic and contribute to it and eventually enrich the literature with more resources and guides. This thesis arises the following questions:

1) Can we convert a 2D network model into a 3D network model that enables data analysts to interact with the data?
2) How efficient are the 3D VR-based network models in data analysis?
3) What are the main challenges in developing metaverse applications?

According to the problem statement, this research is an attempt to investigate the aforementioned lacks and feed the literature. This work also tries to guide researchers on how to generate interactive data visualization models. Hence, the contribution of this thesis can be summarized as follows:

1) Develop a 3D VR-based interactive biological network model for gene-gene interactions that work in the metaverse, which is considered a novel investigation.
2) Provide a novel guide that contains the required resources for the metaverse.

This article is organized as follows, the research methodology including the dataset and the details of the proposed model design are described in the next section while the results are shown in Section 3. This work is concluded in Section 4.

II. RESEARCH METHOD

In this chapter, the dataset used in this thesis and the software requirements including the feasibility study and design phase are explained in detail. Moreover, the method and tools used for visualizing the biological data in a 3D way as well as the design of the VR environment are explained. Also, the method of making the model interactive in terms of the actions that users can perform in the virtual environment is explained. The chapter also describes all the codes in the form of flowcharts and how to adjust the settings of the environment for making it compatible with the network model.

The general flowchart of the whole work performed in this thesis is described in Figure (3). The work starts with performing the analysis of the model including the feasibility study and defining the model requirements. Then, the dataset is brought and formalized to be adequate for visualization. The 2D model is, then, built and the network (nodes and edges) is visualized. After that, the 3D model is designed and the network is also visualized in a 3D method. The designed 3D model in this thesis is called "BioVerse". Then, the VR environment is built and the BioVerse model is adjusted in the designed environment. A test is performed to show how smooth the network can work in the environment; the whole model is tested until reaching a smooth and compatible flow between the environment and the BioVerse model. After completing the work, a usability test was performed by a variety of users aiming to show the differences between the 2D and the 3D models and show which one is more efficient.

2.1 Dataset Collection

The data used in building the network models is biological data for human genes of chromosome X. The main source of these genes is the National Center for Biological Information (NCBI). The dataset consists of 732 genes that exist in chromosome X and 572 interactions with the other human chromosomes. These interactions show the impact of two or more genes in causing a particular known disease. This dataset will be modeled to be visualized in the form of 2D and

interactive 3D VR-based methods that enable researchers to investigate the relations of genes in chromosome X with the other genes that exist in the other chromosomes and eventually extract facts and knowledge about a particular disease (see Table 1). Furthermore, there are many sources that contain biological datasets. However, regardless of the type of biological dataset, this thesis tries to model any biological dataset that includes entities with relations and interactions between them. Therefore, the proposed model accepts any biological data inputs in a particular format and models its corresponding network.

Table 1: Illustrates an example of the genes of chromosome X and the genes of other chromosomes that associate with a disease

Gene in Chromosome X	Genes in other Chromosomes	Associated Disease
UBQLN	UBE3A (Chromosome 15)	Angelman syndrome
NR3C4	RB1 (Chromosome 13)	Retinoblastoma
IL2RG	JAK3 (Chromosome 19)	Severe Combined Immunodeficiency
BRCC3	BRCA2 (Chromosome 13)	Breast Cancer
FLNA	VHL (Chromosome 3)	Von Hippel-Lindau disease
BRCC3	BRCA1 (Chromosome 17)	Breast Cancer

The dataset includes information about each gene, such as an identification number, the scientific name, and the chromosome associated with it. Also, other information such as diseases associated with each gene and its interactions with other genes is also included in the dataset. The dataset is formulated to be in two files; the first file includes the nodes (genes) and their attributes, and the second file includes the edges that represent the relations between the nodes (genes).
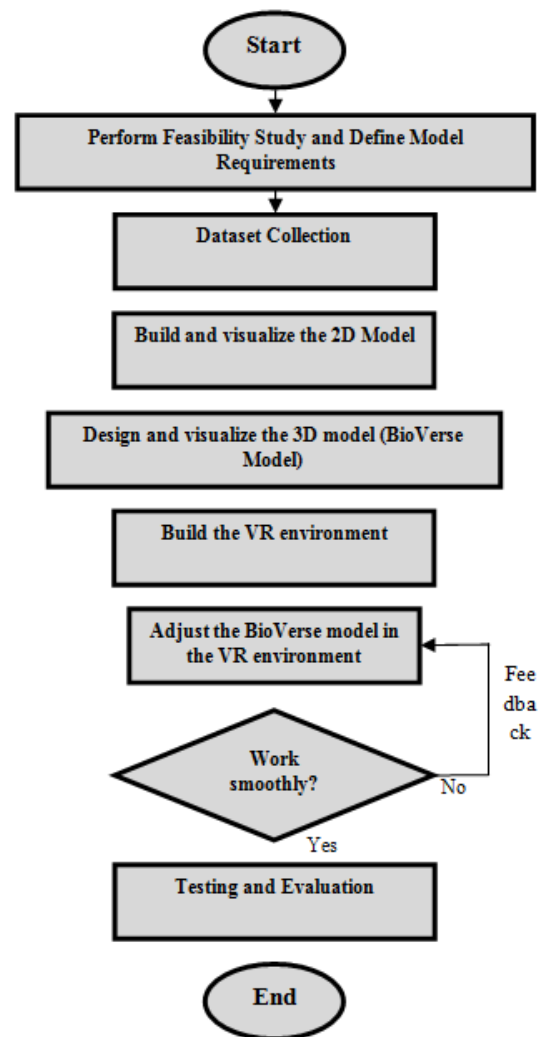
Figure 3: Workflow Diagram

The generated files are used to generate both network models; the 2D and the interactive 3D. The biological network models consist of nodes, which represent the genes, and the edges connect the nodes to determine the pathogens between the genes. Moreover, each of the generated network models (2D and 3D) is represented as a graph (G) with vertices (V) and edges (E). It can be visualized using particular layouts of interest (e.g., clusters, circles, geo-layout, etc.). The analysis can also be supported using network metrics. These metrics play a significant role in deeply understanding the network model. Furthermore, the strategy followed in building the network models can be summarized based on the following example:

Given that we have two genes in the same or different chromosomes. These genes are represented as two nodes, an edge is created between them if and only if they have a disease in common or there is an interaction between them. Considering the genes and diseases presented in Table 3.1, the network can be visualized as shown in Figure 4.

According to the figure, the type of chromosomes is encoded by node color. For instance, the pink color distinguishes genes of chromosome X, and the green color distinguishes chromosome 13, and so on. Also, the size of nodes encodes the frequency of diseases that a gene has with other genes in the same or different chromosomes. Gene BRCC3 which exists in chromosome X has two diseases (breast cancer and Von Hippel Lindau) in common with BRCA1 (chromosome 17) and BRCA2 (chromosome 13). The label of the edges encodes the disease associated with two genes. Practically, the network model shown in the figure is a simple example and part of a huge network model.
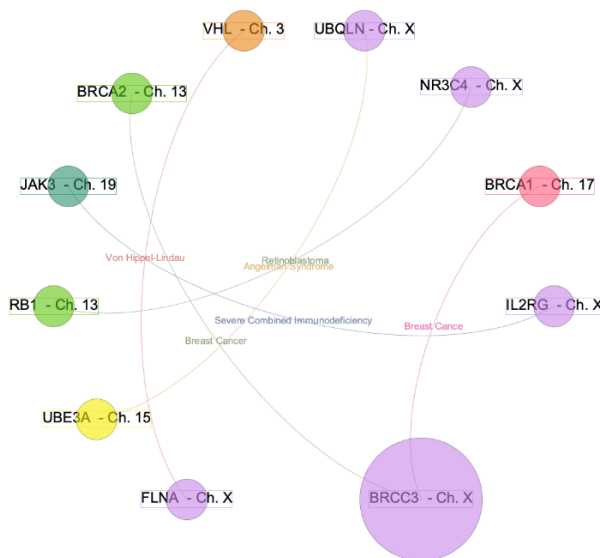
Figure 4: An example of how a network model is represented

2.2 2D Visualization

The dataset described in the previous section is used to visualize the 2D network model using the Gephi software, which displays the nodes and edges of large data in a superficial way. The size of the nodes can be used as an indicator of a certain fact about the data. Moreover, the color of the nodes can be used to encode information about the data, therefore, the color of a particular node represents a specific fact about that node. Similarly, network edges in terms of color and thickness (weights) can be used to encode information. The node and edge labels are another dimension in which it can encode information. Table 2 presents the design characteristics of the 2D network model and the next chapter shows the visualization and the main characteristics of the model.

Table 2: The main design characteristics of the 2D network model

Feature	Description	2nd Level requirements
# of Nodes	732	-
# of Edges	572	-

2D visualization Layout	Circle Pack Layout	Hierarchy 1 - Chromosome
Expansion Layout	3 steps	-
Nodes Size Ranking	50-150	Degree Centrality
Nodes Color Ranking	27 colors	Chromosome
Edges Weights	0.1 – 2.0	Weight
Labels	True	Labels of Nodes and Edges
Nodes Opacity	50%	-
Nodes Opacity	100%	-

2.3 Interactive 3D VR-Based Model

The interactive 3D VR-based model is considered software that passes through the software life cycle (see Figure 5). Therefore, the following sections include the steps that are followed in the field of software engineering when designing any software. The first step starts with defining the requirements including the software and hardware requirements as well as the feasibility study of the software. The next step is to design the model, followed by the third step, which is coding the design and then testing it, and finally performing maintenance as required.

Figure 5: Software Development Life Cycle (SDLC).

2.3.1 BioVerse Feasibility and Requirements

For the purpose of realizing that the proposed model is feasible, an analytical study was performed when surveying the literature. The feasibility study of BioVerse was carried out based on a different point of view as follows.

- *Scientific point of view*: scientifically, adding more dimensions to any shape can increase the information gathered from that shape. From this point, BioVerse will be designed to be more efficient than the 2D model in investigating and analyzing biological networks as well as easier to interact with. Therefore, the proposed model is supposed to be "*scientifically feasible*".
- *Economic point of view*: with the advent of VR technology that enables people to remotely interact with

each other in virtual worlds, BioVerse should be "*economically feasible*" since it secures a private room for virtual meetings and enables more researchers to interact with the same model without the need to travel.

- *Technical point of view*: BioVerse is planned to be designed using sophisticated programming languages and tools, which enables developers to continue adding features as required. Also, the BioVerse is supposed to be open-source and used the recent versions of VR devices; therefore, BioVerse should be "*technically feasible*".

- *Operational point of view*: the ease-of-use is an important feature of any software. BioVerse is planned to be designed in a way that is considered easy to use by any person and smooth when moving around the environment. Therefore, BioVerse is supposed to be "*operationally feasible*".

2.3.2 Hardware and Software Requirements

To create an interactive VR-based 3D visualization, it is required to use a powerful computer, a VR headset such as Oculus Rift or HTC Vive, and 3D software such as Unity or Unreal Engine. The software can be used to create detailed 3D models, interactions, textures, animations, and other special effects. Figure 6 depicts the architecture of the model. Once the model is created, it can be uploaded and installed in the headset hardware.

Creating a 3D visualization is a complex process that requires extensive knowledge of 3D modeling, animation, and high programming skills. However, with the right tools and skills, developers can create a stunning 3D visualization that can bring their product or service to life.

Figure 6: Represents the architecture of the model

In this paper, to build the **BioVerse-VR** model, it was needed to select the most adequate software and hardware tools that enable the production of a compatible, reliable, smooth, and efficient model. In virtual reality, which includes the final model to visualize and interact with it through a headset and joysticks for the right hand, left hand, and other interactions, the device used is called Meta Quest2, as it is considered a multi-capacity device in which applications can be run independently without linking it to a computer system. Hence, the hardware devices used are as follows:

- A Lenovo Game3 laptop computer that has the following specifications:

  CPU: 12th Gen Intel(R) Core (TM) i5-12450H 2.00 GHz

  RAM: 8GB

  GPU: Nvidia RTX 3050 – 4GB

  USB: USB3, Type-C

  OS: Windows 11

- Meta Quest 2 headset with 2 joysticks for both hands as shown in Figure 7.

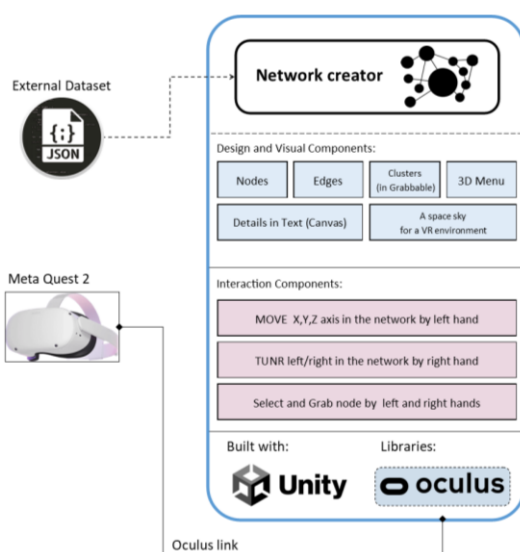**Figure 7: Meta Quest 2 VR device**

To build, visualize and interact with the **BioVerae-VR** model in a VR 3D environment, Unity version 2021.3.14f1 was used, which is a multi-platform game engine that uses a huge range of functions and features to develop in a 2D, 3D, VR environment or engineering solutions. The programming language used was C# that was connected with the Unity engine to develop the model and control the data, and interact with its objects more easily using specific scripts. These scripts represent code files containing functions and code that control program actions through its tools and put it on the object specified in the program, where the model was built by making nodes and edges to generate the biological network, as the nodes represent genes and edges are the links between genes. Genes were represented through a sphere and put some code scripts on it to take a specific shape, color, and movement according to what users interact.

www.irjiet.com   53

2.3.3 VR-Based Design and Coding

A virtual reality application called BioVerse is designed and implemented for interactively visualizing gene networks in a 3D environment. For the purpose of visualizing the biological data in the designed application, a 3D environment was designed and prepared. In the designed environment, users can easily, smoothly, and comfortably navigate the 3D world while interacting with the network as well as access thorough gene knowledge via a browser in virtual reality settings. Figure 8 is the use case diagram that illustrates the interactions between users and the BioVerse application.

BioVerse model was built to allow users to investigate and engage with the network using a VR headset and controllers this is because it was created using the multi-platform game engine, Unity, which has a broad variety of features, but it is particularly useful for creating 2D and 3D video games and virtual reality applications.

In addition to the use of the Virtual Reality Toolkit (VRTK) to create VR solutions in Unity, the C# programming language served as the primary programming tool for our applications. The Meta Quest 2 headset served as our virtual reality gear. The BioVerse model was implemented on a platform that allows it to operate on a PC with more processing power than a standalone headset using this kind of all-in-one Head Mounted Display (HMD). Moreover, by interacting with the data and adding movements and actions, we have used the Oculus toolkit in Unity to add the movements of the hands and their natural shape, as well as the additions of the Extended Reality (XR), which is the controller of the virtual reality that merged real, virtual, and human interactions in the environment.

In order to create an interactive browser that can be used to view additional gene information that is positioned above each chosen node, we used Zen Fulcrum browser (ZFBrowser), which is an external library that was added to Unity. The information from the approved sites for human genes is displayed on a particular Canvas on a screen-display with measurements and locations that are specified above each gene individually. For capturing the 3D world in virtual reality, we also used 3Skyboxes which is a collection of add-ons for Unity that contain images. To capture the environment appropriate for the biological data in space, a sky template was involved.
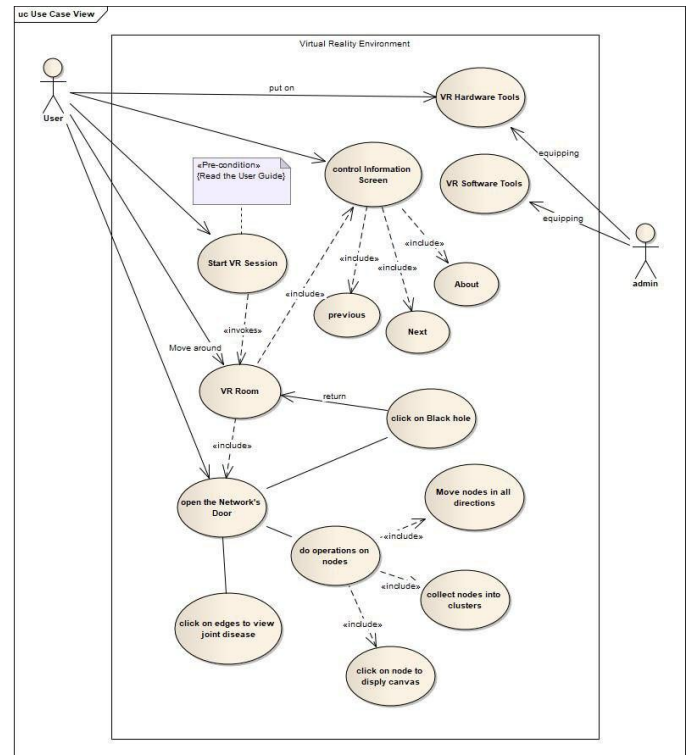
Figure 8: Use case diagram of the BioVerse application

2.3.4 Steps to build the BioVerse-VR app

We created a new project through Unity Hope, then, the appropriate environment was chosen for the work, the created project was a 3D-based environment. Then, through the Unity options, we performed the required steps to setting up virtual reality tools in a 3D environment that is known as XR, through which the project can be controlled after implementation. Using the headset and control panels, and in the following order, the steps to identify the XR and install the libraries for work are as follows:

- From the File tab, we choose Build Setting. We specify the platform as Android because the headset deals with applications with the .apk extension.
- From the Windows tab, we select Package Manager, then install the OpenXR Plugin, and then add Controller, and here we can use the control panels inside the bunty.
- From the Windows tab, we choose Package Manager, then install XR Plugin Management, and then add Examples of XR management implementation to configure the tools that we will deal with in the virtual reality environment.
- From the Windows tab, choose Package Manager, install XR Interaction Toolkit, and then add Starter Asset to start working with virtual reality tools.
- From the Edit tab, we choose project settings, then adjust the settings of the XR Interaction Toolkit by activating Oculus, and set openXR as Android to enable the output application further installed inside the headset.

- Adding work libraries through the Windows tab, select Asset Store and download the previously described libraries (Meta Interaction, Skyboxes3, etc.).

After completing the basic settings for creating a virtual reality within a 3D environment, we work and collect the tools and arrange them according to our needs in the Hierarchy window. Through *Scenes*, two scenes have been created Room Scene and Biological Network Scene.

A) Room Scene

Once the application starts, the scene of the room that can be toured is shown, which contains a screen to display information about the application and other information about the application. In the room, there is a door through which one can enter the next scene which is represented by the space of the complex biological network. To visualize and interact with the network model, a set of tools has been added to deal with and code it to achieve the goal of realizing and visualizing the scene. Moreover, the hierarchical list contains the following:

XR Interaction Manager: The "XR Interaction Manager" is a "Unity API" that enables interactions between "Interactors" and "Interactables" in XR applications. Interactors are objects in the scene that can initiate interactions. while Interactables are objects that can respond to interactions. The "Interaction Manager" acts as a mediator between the two, managing the interactions and ensuring consistency. Multiple "Interaction Managers" can exist in a scene, and "Interactors" and "Interactables" must register with a valid "Interaction Manager" for communication to occur. The "Interaction Manager" provides a set of methods that enable "Interactors" and "Interactables" to communicate with each other indirectly. Ensuring that both targets of an interaction event are updated consistently. By using the "XR Interaction Manager", developers can create immersive XR experiences that respond to user input in a natural and intuitive way. The "Interaction Manager" simplifies the process of managing interactions, allowing developers to focus on creating engaging content rather than worrying about the technical details of interaction management.

XROrigin: Is a "Unity API" component that provides a reference point for the tracking of XR devices such as VR headsets or motion controllers. It is responsible for implementing the concept of a camera rig in XR applications, providing a standardized way to manage the camera and device tracking. The "XROrigin" component includes several tracking origin mode options that enable developers to configure the reference frame for positions reported by the XR device. This is important for ensuring that the virtual world aligns with the physical world. The "XROrigin" component has several properties that allow developers to customize the

camera rig. Including "Origin" that specifies the position and orientation of the origin point for the tracking system. Also, "Camera Offset Object" which is used to offset the camera from the origin point. Finally, "Camera Object" that specifies the main camera that is used to render the scene (see Figure 9).To set up the "XR Origin" and input actions in Unity for tracked devices, developers can follow the general setup process, which involves creating an "XR Origin" camera rig.

This camera rig includes an "Origin GameObject", which serves as the reference point for device tracking, as well as two child "GameObjects" named "LeftHand Controller" and "RightHand Controller". Each of the controller "GameObjects" is equipped with an "XR Controller" component, which represents the motion controller for that hand.

By configuring the "XR Controller" components, developers can specify which input actions the controllers should respond to, such as grabbing or releasing objects. Following these steps, developers can create an "XR Origin" camera rig and input actions for tracked devices in their Unity project.

This standardizes the camera and device tracking, allowing developers to create immersive XR experiences that respond naturally to user input.

When assigning actions to "Action-based controllers" in Unity, it is recommended to use the "XR Controller (LeftHand) or XR Controller (RightHand)" binding paths. These binding paths correspond to the left and right motion controllers respectively, and provide a standardized way to map input actions to the controllers. This makes it easier to create a consistent user experience across different platforms and devices, while also reducing the amount of custom code that needs to be written.

Developers can include the "Input Action Manager" component in the "Interaction Manager GameObject". This component helps to manage input actions across different devices and controllers, ensuring a consistent experience for users. If developers have already installed the Starter Assets sample, they can add the "XRI Default Input Actions" from the "Samples folder to the Action Assets field of the Input Action Manager".

We evaluated the work of Stack Tech for the XR in Unity, as shown in the figure 9, which is based on the Oculus XR plugin, which represents the representations of the provider that flow into the framework consisting of a group of SDKs such as the display, inputs, the environment suitable for work, interfaces, recast, camera, object and images tracking, And Meshing the objects in the environment, and then add

what we need in the framework from the toolkit of the XR Interaction, which is classified under the developer tools, and then create the appropriate application to work as VR in Metaverse.

Camera Offset: Move the Camera to the desired height off the floor using a Game Object are the main camera coordinates associated with the headset and both the left and right joysticks.

Locomotion System: The "Locomotion System" is a "MonoBehaviour" for creating immersive XR experiences that involve movement and navigation within a virtual environment. By acting as a middleman, it helps to streamline the development process and ensure a consistent user experience across different "locomotion Providers" and "XR Origins".
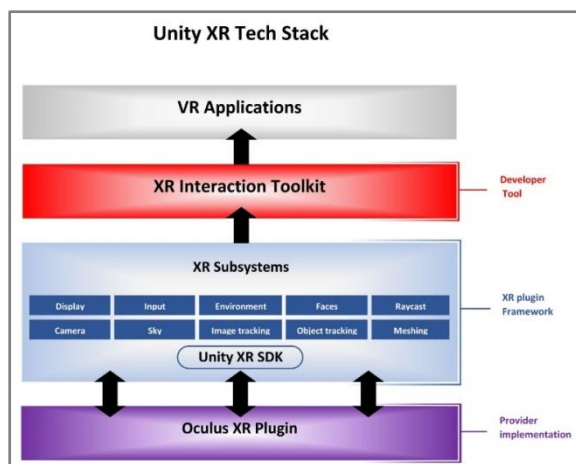
Figure 9: Unity XR Tech Stack

TM's Living room interior: It is a room purchased from the Unity asset store that contains a group of controllable objects, as the room was designed to be more realistic for working in and holding virtual meetings in the Metaverse application. A collection of personal images and indications have been added to the biological network section, and the TV screen has been programmed to display a collection of information related to the work and supervision of this model, as well as how to deal with the application (see Figure 10).
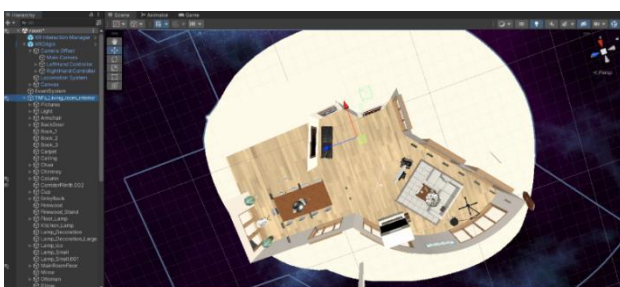
Figure 10: Living room interior design

A hidden cube was placed behind the biological network's door, and a software file called trigger.cs was placed on the cube, which moves to another scene when touched; when pressing the door handle, it will open by making a rotation animation; and when the door touches the cube, the person will be transferred to the biological network's other scene (see Figure 11).
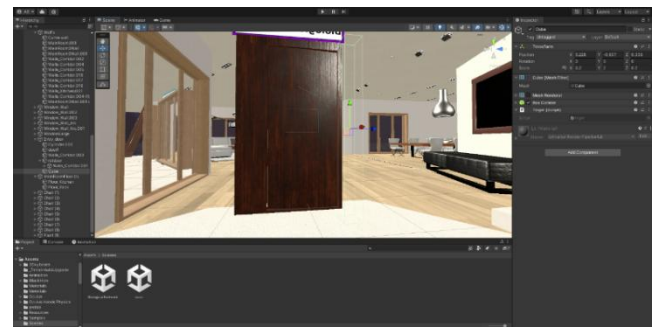

Figure 11: Collider cube behind the door to enter the biological network

EventSystem: Handles events such as input, ray casting, and sending. In a Unity scene, the EventSystem is in responsible for processing and handling events. There should only be one EventSystem in a scene. In Unity, the "EventSystem" interacts with several modules that handle different functionalities and maintain their respective states. The modules are typically overridable components that can be customized according to the specific requirements of the XR application. Upon launching, the EventSystem searches for any "BaseInputModules" that are attached to the same "GameObject" and adds them to its internal list. Each module receives an "UpdateModules" call when it is updated, enabling it to update its internal state accordingly. Once all the modules have been updated, the "Process call" is executed on the active module. This is where custom module processing can take place.

B) Biological Network Scene

It is the scene that represents the biological network, which contains nodes and edges that are linked to each other based on the previously collected data set information, and the biological network is visualized in a three-dimensional environment and interacted with via the controllers and the headset, and when clicking on any node, a browser appears above it with details. It displays the name of the gene and its chromosome in the environment space for this gene from the sites reliant on it, as well as for marking any node. Interaction It is possible to assemble a set of genes determined by network users in order to learn about and identify particular diseases. Figure 12 demonstrated the generated biological network in a 3D VR-based visualization. To make a clear visualization, the set of genes for each chromosome and its affiliates are encoded

in different colors and are located in distinct locations. The objects on which the network is constructed and the party to interact with are explained in the following hierarchy.
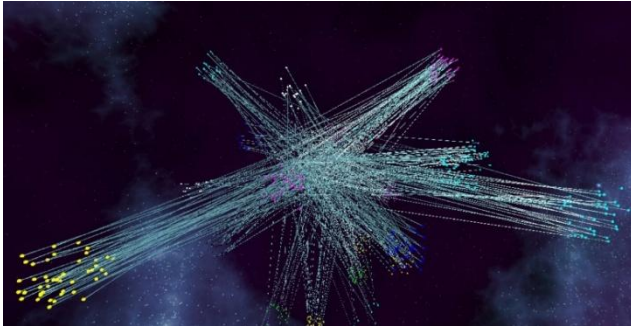

Figure 12: biological network in a 3D VR-based visualization

It is the scene that represents the biological network, which contains nodes and edges that are linked to each other based on the previously collected data set information, and the biological network is visualized in a three-dimensional environment and interacted with via the controllers and the headset, and when clicking on any node, a browser appears above it with details. It displays the name of the gene and its chromosome in the environment space for this gene from the sites reliant on it, as well as for marking any node. Interaction It is possible to assemble a set of genes determined by network users in order to learn about and identify particular diseases. Figure 12 demonstrated the generated biological network in a 3D VR-based visualization.

To make a clear visualization, the set of genes for each chromosome and its affiliates are encoded in different colors and are located in distinct locations. The objects on which the network is constructed and the party to interact with are explained in the following hierarchy (see Figure 13).
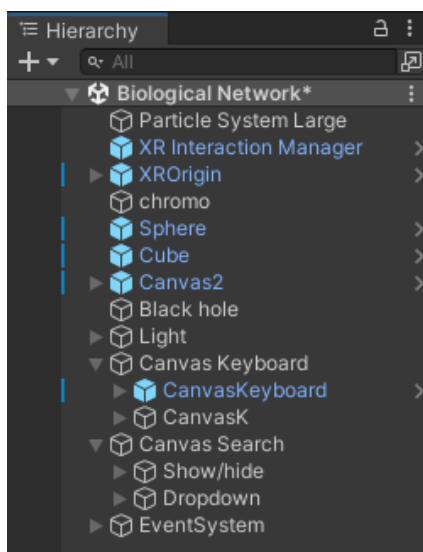

Figure 13: Hierarchy Biological Network Scene in Unity

For the XR Interaction manager, the XR Origin in this scene is the same action and settings that were set in the previous room scene. However, the difference in camera continuous turn speed and movement speed can be noted in the Locomotion System as depicted in Figure 14.
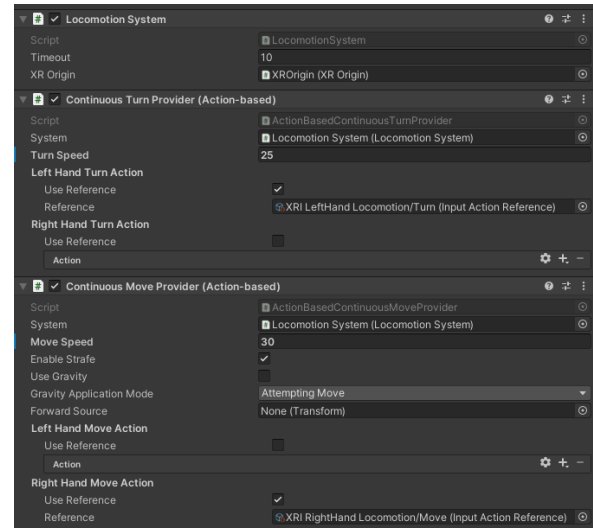

Figure 14: inspector Locomotion System Component

The continuous movement is limited to two axes: (X-, X+) and (Z-, Z+). As a result, the movement has been added to the top and bottom represented by the vector (Y-, Y+) by placing the **Hand Controll.cs** file in the XR Origin properties to enable it Ease of roaming in the virtual reality space and enjoying the three-dimensional network by reaching the highest and lowest node in the network (see Figure 15).
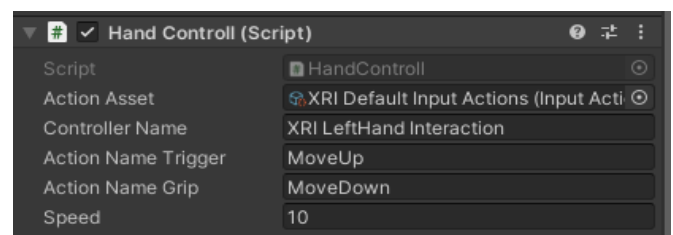

Figure 15: Inspector of Hand Control Script

Table 3: Description of the option for the Hand Control script

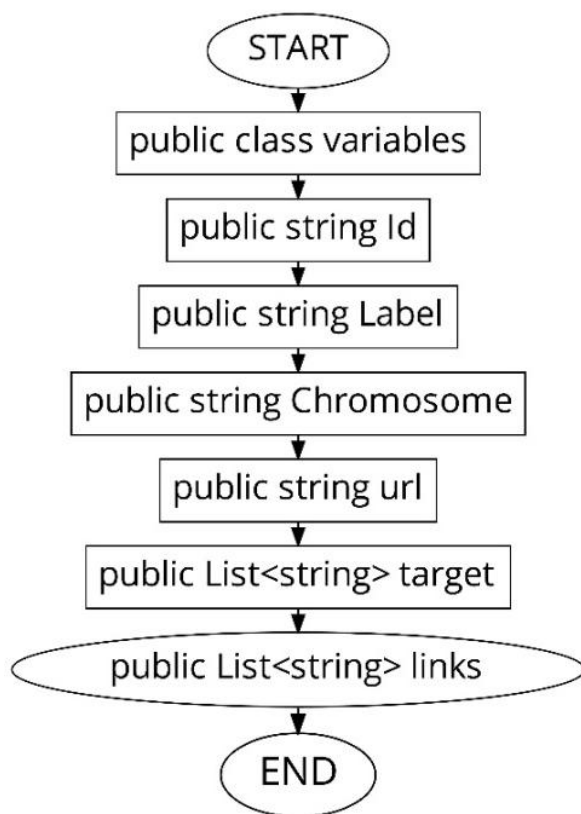| Property	Description
Action Asset	A group of tools that are added to control the buttons of the control tools through the joystick, and they are determined by adding the **XR Default input Action Manager.**
Controller Name	To determine which interaction, we deal with through its buttons, here we chose the left handle known as **XRI Left Hand Interaction.**
Action Name Trigger	Move to the top with the Trigger button, known as the X button, on the left joystick.
Action Name Grip	Move to the down with the Grip button, known as the Y button, on the left joystick.
Speed	Determine the speed of movement up or down and we set it 10.

Chromo: It is a GameObject in which we put a software code called **ReadJson.cs** and through Figure 17 it reads the dataset information from Block3.json as Textjson of type textasset as well as this data takes the form of Sphere defined in Chro and the formation of a matrix for the number of mynodes and a matrix for the number of links Mylink.

Figure 17: Code Flow define Variables for ReadJson.cs

Likewise, the locations of all genes belonging to a similar chromosome are close to each other, according to Loc, and the color of all genes of the chromosome is one color, according to Color (see Figure 18).

Figure 18: Code Flow for ReadJson.cs, Steps of diffusion of the network within boundaries and coloring of genes in groups

Sphere: It is a game object in the program that takes the shape of a ball, and we used it to form nodes represented by genes, where a single sphere was created, and the program me files that we need to configure the network and its tools (see Appendix: Figure 13), such as **variables.cs**, **links.cs**, and **Control.cs**, were created, which (Control.cs) is a general control on the network, and the files will be clarified code later in the code flow.

The **Variables.cs** file contains the same information as the data set represented by the Block3.Json file, and it is read in order to extract each gene represented by its information node in the data set (see Figure 19).

Figure 19: Code Flow Variables.cs , The main variables defined in the dataset in the Block3.json

According to the information in the data set represented by the **Block3.Json** file, the **Link.cs** software file represents the edge between one node and another, and it was designed in a cubic shape to provide linkage in the event that a gene is linked to more than one gene, as Line Render helps to link an object with another object. Read it to extract the current gene Target and combine it with the other gene. The link file was placed on the Sphere object to link genes with each other according to the source and target parameters in the dataset. In the figure 20, which represents a diagram of the code of the link, we explain the steps for making this link and how to configure it programmatically.

In the beginning, we define **lfab, link, target** of the GameObject type, define an array called **g** of the GameObject type, and an array named **cc** of the Vector3 type, meaning three-dimensional. Adding colors to the matrix through linkk and then making an instantiate of the lfab and the actual link known to it, then we set a condition if the chromosome is not X we give colors to it and if it is an X chromosome just the name of linkk we add a tag /// with the iteration number and put it in the g matrix and so on The process until the count is equal to 0, and we can also, through this code, preserve the location of the link in the event that it is extracted to another place by calculating the offset and position.
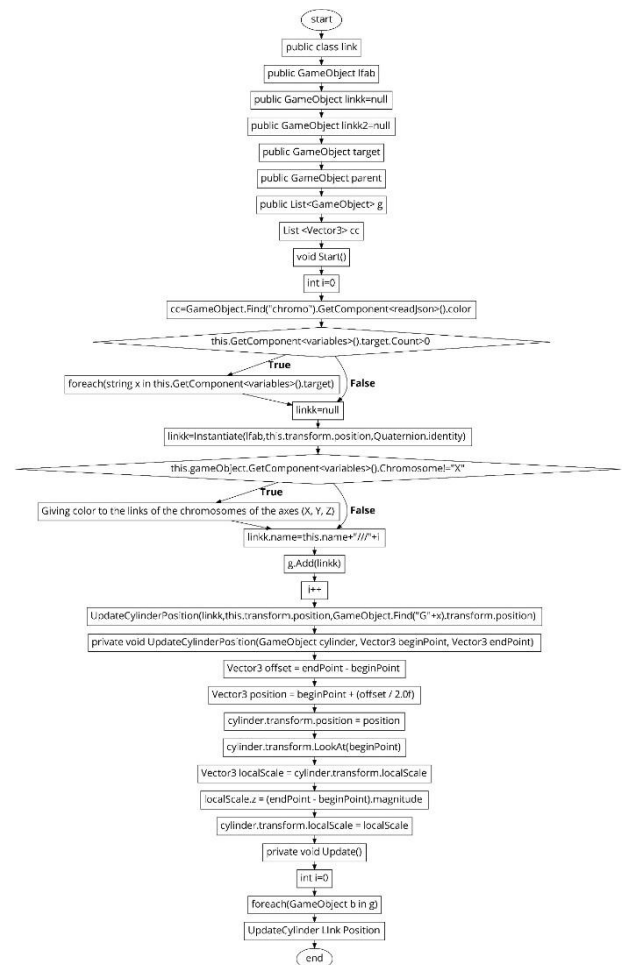
Figure 20: Code Flow Link.cs

The **Control.cs** software file includes a set of classes and functions that allow us to control the node selection. Its job here is to show and conceal the canvas above the node, which serves as a browser for this gene's information. The controller's name must be the same as the name linked with the Oculus, as shown in the Figure 21, and the XRI has been specified. Contact with the Left Hand Its class is actionAsset, and the action it employs with Sphere is disCanvas.
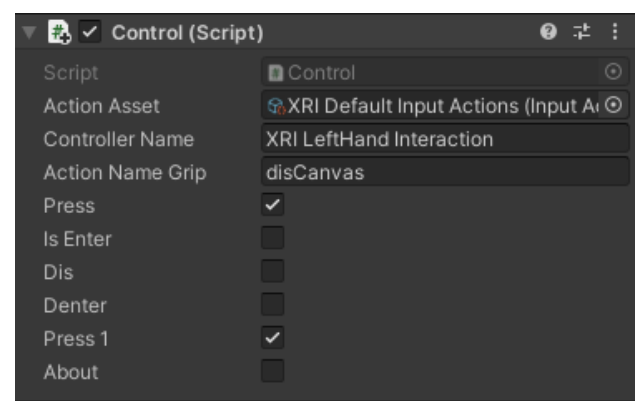
Figure 21: The Inspector of the Control.cs program file, which is located in the Sphere GameObject

And through the flow of the code in Figure 22, some variables necessary for the work were defined, and then after that we created an animation used to rotate the door in the first scene (room scene), and made a time for the browser that appears above each node, and we specified it with one second for each button press to show it in the OnEnable() function and hide it in the OnDisable() function.
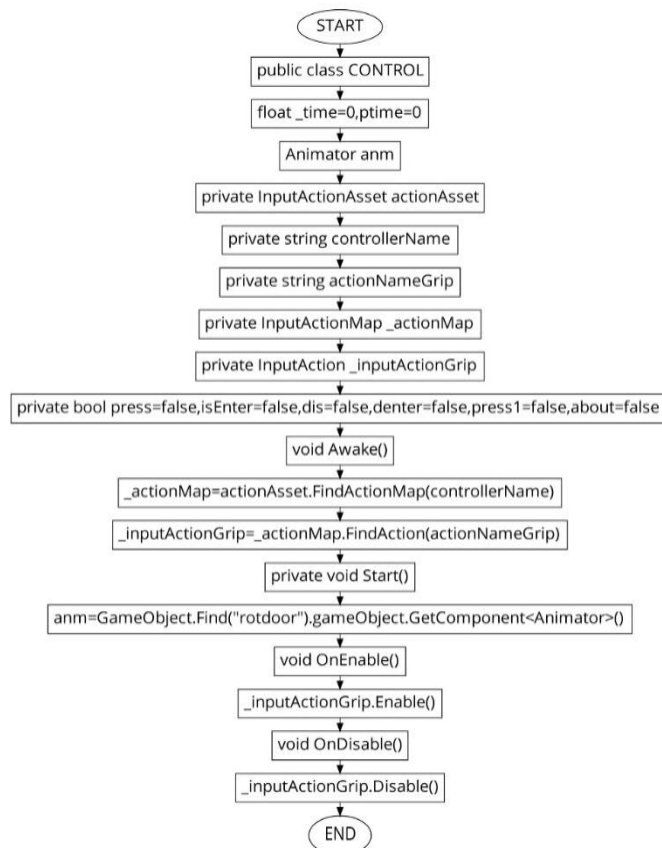

Figure 22: Code Flow control.cs Variables Definition

In the update function for all software procedures defined in Figure 22 previously, here is the work in a detailed way as shown in Figure 23, where each time a comparison is made first of the real time specified in one second between one button press and another and the time of turning the door to enter it into the biological network, when pressing the door handle in a function doorenter() by giving an integer value to Denter that is greater than zero with the press variable also activated and exiting the door through doorexit(). It is also used to show and hide the browser located on the canvas above the nodes, through the OnDiplay() function, the name of the gene and its chromosome can be shown in the canvas2 through Text1 and Text2 in the main camera, and this appears by placing the laser over any node in the network. This code can also be used in a presentation on the screen in the first scene, through the aboutin() function, and close it through aboutout().
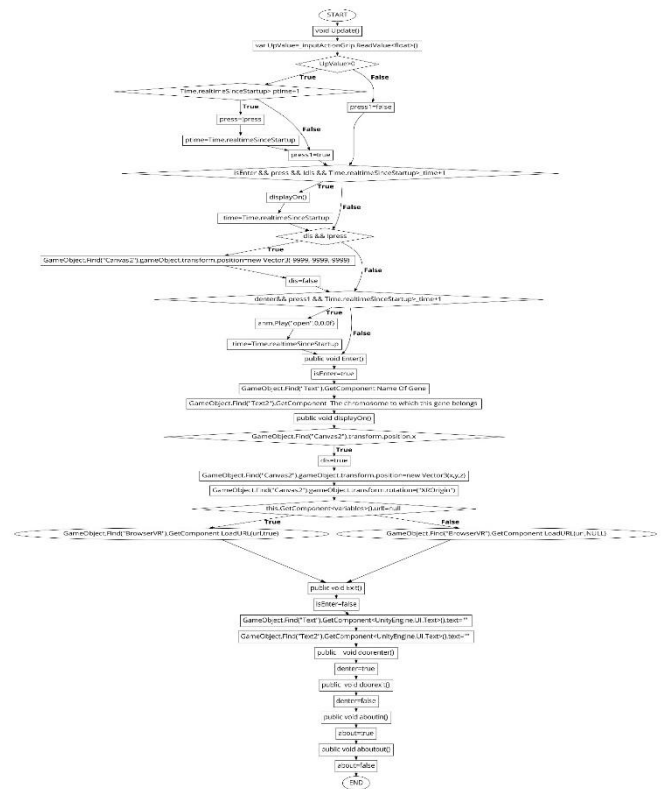
Figure 23: Code Flow control.cs Software steps to update the procedures required at work

Cube: It is a game object that was added to the work to benefit from it in making links between nodes, so we added some functions to it in its properties.

Through the above figure, we notice that the dimensions of the cube in the transform are relative to the (x, y, z) axis, and the mesh filter is of a cylinder type.

Canvas2: When we want to display our data, we need a canvas, which is considered a user interface UI, in which we display what we want in the form of texts or images, and in this work, we needed two canvases, the first was placed in XROrigin and contains two texts that display the name of the gene and its chromosome and move with movement The headset appears by simply hovering over any node and is located in the world space.

The other canvas is separate, it is located in the hierarchy of the scene, and it contains a browser to display information about each gene, and its location is above each gene, and it appears only by clicking on a selection button from the joystick.

Black hole: It is a game object in the form of a sphere that has been added to this scene in the event that the person using the network wants to return to the room scene, and this is done through a code called "**Back ToRoom.cs**" as Figure 24, Once approaching this hole, the person will move to the room.
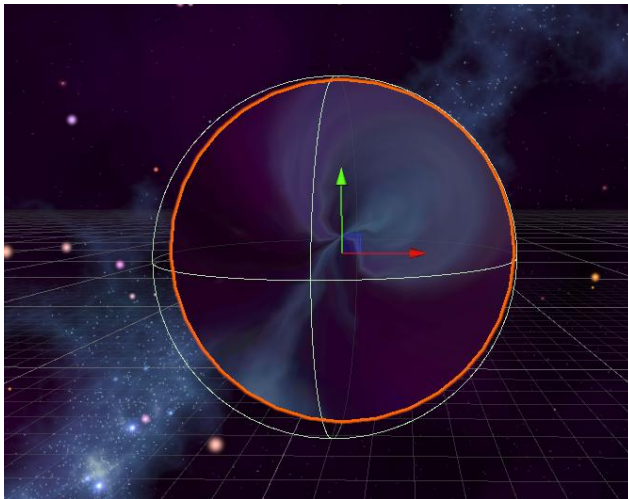
Figure 24: Black Hole

Figure 25 shows the software steps for returning to the room scene from the biological network scene by entering the black hole, where when touched by the collier surrounding it, it activates the BackToRoon.cs program file, which gives instructions to load the room scene.
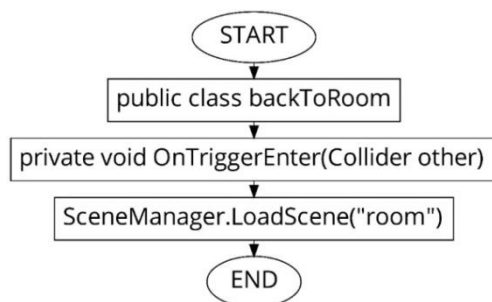

Figure 25: Code Flow backToRoom.cs

Canvas Keyboard: Through this canvas that contains a keyboard, a text box and a button, we search for any gene in the biological network by writing the correct scientific name of the gene in the text box field and then pressing the Search Gene button, in order to facilitate access to any gene quickly as well as roaming the network As in Figure 26.
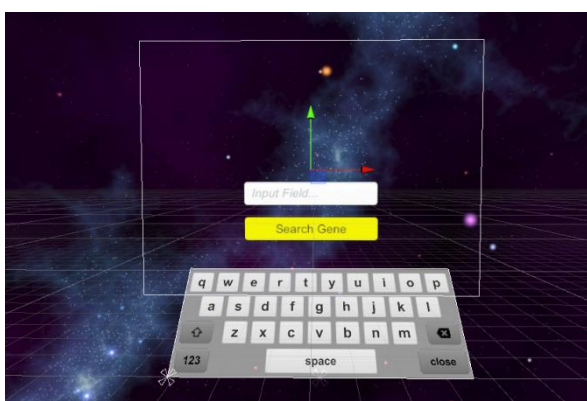
Figure 26: Canvas Keyboard to search for any Gene

Canvas Search: We can hide and show any link between the genes that we identify through the chromosome name by selecting it from the dropdown and pressing the show/hide button to simplify dealing with the linkage of genes and show only what we need and reduce the number of links as in Figure 27.
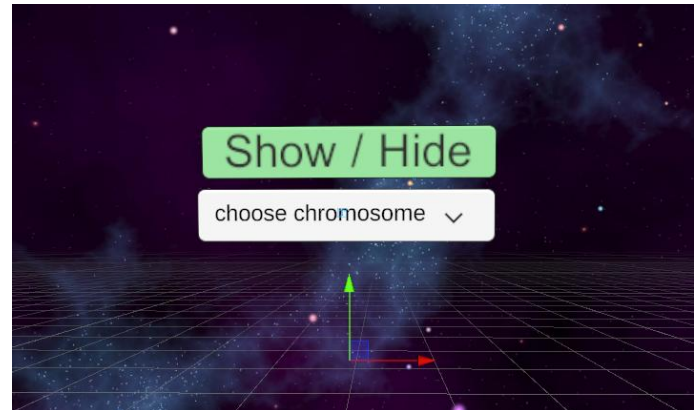
Figure 27: Canvas searchto show and hide links between genes

2.3.5 Network Interactions

Virtual reality headsets provide a unique experience in an interactive, adventurous and action-rich manner. Examples of what can be done in virtual reality, for example, moving, picking up objects and interacting with the environment using hands or the joystick, as well as through virtual tools such as pressing the buttons of the joystick to do a specific thing, 3D interfaces, and others. Here we will explain the technologies we have introduced to interact with and visualize the network and make the most of virtual reality. Interaction and visualization depends on the virtual reality technology used, as meta quest 2 was used in this project, and it is an integrated VR headset that does not need a computer or wires to run applications, and it comes with two joysticks, one for each hand, as the joystick contains inputs such as buttons and cursors that are used To activate programmed actions in the VR app, in particular we used some of these inputs available in the joystick in the BioVerse app and mapped them to different actions that allow the user to interact with the network and the environment.

In Figure 28 we explain some procedures that correspond to each entry of the joystick 1- It is a rotation in space and around the network in two directions, right and left, at an angle of 360 degrees, 2- It is the movement upward towards the Y + axis, 3- It is the movement downward towards the Y axis, 4- Display a browser above each node that is selected and click on it, 5- Grabbing and extracting the nodes from their place and dragging them to another place, 6- It is movement in space and navigating the network in the direction of X + forward, X- backward, Z + to the right and Z- to the left , 7- main oculus menu.
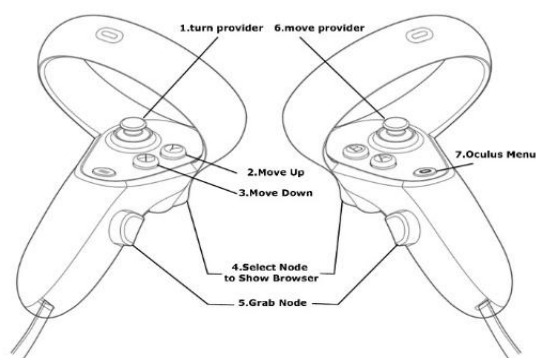
Figure 28: Oculus Quest 2 joystick buttons, and the function of each button is explained

III. RESULTS AND DISCUSSIONS

In this chapter, we will present the results of the biological network and visualize it in a virtual reality in a three-dimensional environment, how to interact with genes, ways to connect them with each other, and learn all the details about the gene through a web page that displays above each gene in the environment and how to extract the gene and transfer it to another place through special controllers with virtual reality devices. We also made a comparison to visualize this data in a two-dimensional environment and a three-dimensional environment by interacting with it and creating a questionnaire paper containing a set of questions to find out the obstacles to interaction with the network and the use of control devices and the difference between visualizing the biological network in a 2D and 3D environment through the experience of users in this field As a major in biological sciences and technicians in computer science.

3.1 Visualization and Interaction Results

The biological network can be visualized in a three-dimensional way in virtual reality, where it is possible to rotate in an endless space around the biological network as a whole, as well as genes can be seen in the form of nodes connected to each other through links, and all this is done through a joystick and a headset, as The rotation of the headset gives a complete view in all directions and according to the desire of the user, either through the joystick, it is possible to move forward, backward, go up and go down in the space surrounding the network to give a complete view of the network, and by pointing to any node, one can Know the name of the gene and its chromosome. Before entering the biological network, the (BioVerse-VR) application starts with a room in virtual reality, where it is possible through this room to visualize the existing objects where there is a meeting office for metaverses as in Figure 29 that provides an opportunity for users of this application to be in this room to discuss Details of the biological network and adding proposals for future work in this field to develop the model.

Figure 29: Meeting Office for Metaverses

And in an area of the room there is a screen with a set of images as in Figure 30 through which a presentation about the complex biological network and how to build, visualize and interact with it, as well as information about this application such as the version of the application, the year of its establishment, and the supervisor.

Figure 30: Screen showing information about the biological network

And on the opposite side of the screen there is a door as in Figure 31 through which it is possible to enter and visualize the biological network in a three-dimensional way in a second scene called (the biological network scene) in an open space without borders. The network can be visualized from every direction, and it is possible to return to the room through a black hole that exists in space, so when touching it, the user will return to the first room scene.

Figure 31: Door of Biological Network

After opening the door in the first scene (the room), the visualization in virtual reality moves to the second scene, which is called (the biological network) in an open, endless space where the complex biological network is located, as in Figure 32, which shows the network with all its nodes and links.
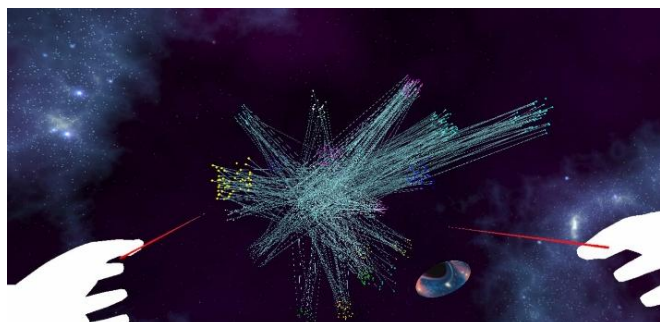

Figure 32: Shows the network with all its nodes and links

It is possible to interact with the nodes of this network by placing the laser of the joystick over any node without pressing on it, and information will appear on the screen, the name of the gene and the chromosome to which this gene belongs, as in the figure 33.
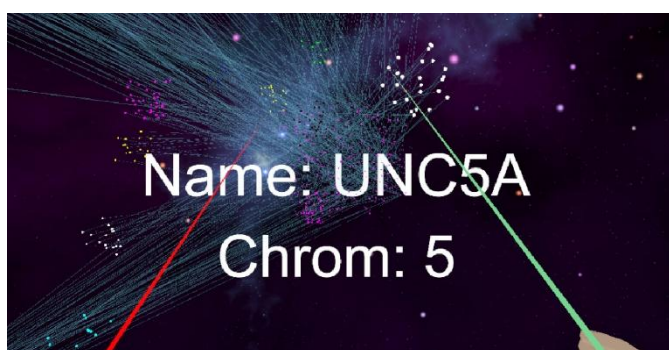
Figure 33: Information such as the name of the gene and its chromosome

When pointing the laser in the joystick on any node and clicking on it with the selection button, a browser will open to find out more information about this gene through approved global links in this field such as Wikipedia as in Figure 34.

Figure 34: Browser to see information about the gene

The node can also be pulled from its group and moved to another place through the grab button in the joystick. This method is very useful for collecting a group of genes in one place and conducting a study on it by specialists in biology, such as identifying or discovering new diseases, as in Figure 35.
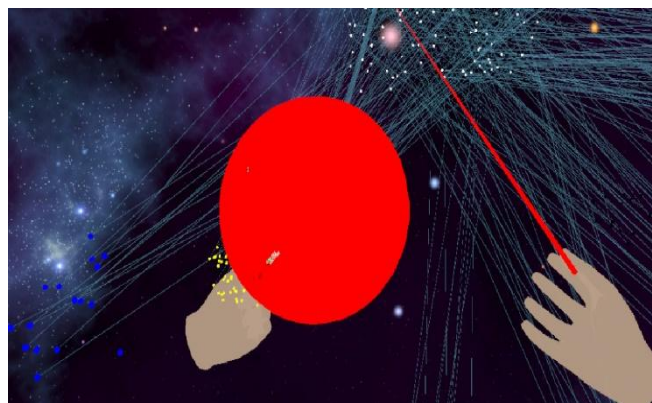
Figure 35: Grab the Gene from its Group to Another Location

Interacting with the network takes place by knowing the name of the gene by simply pointing at it, or knowing complete information by clicking on the node or dragging the node and holding it by hand and transferring it to another place, and it is also possible to cancel the links between nodes by simply selecting the chromosome through the dropdown and pressing the button (show / hide) located on the screen, and this gives greater interaction to the network and facilitates its study, that is, only the links that I want with the chromosome X can be shown, as in Figure 36.
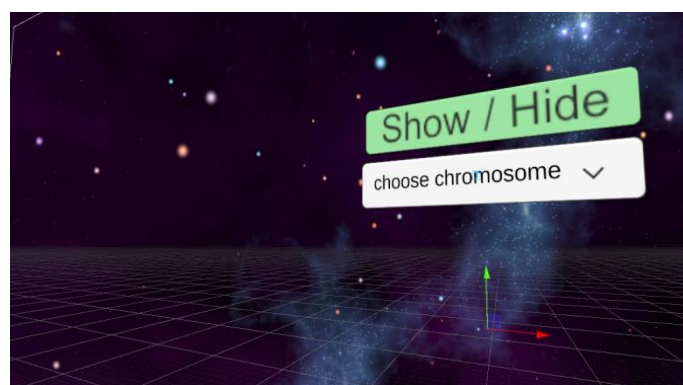

Figure 36: Show and hide links between nodes in a complex biological network

When we want to reach a specific node without wandering around in space and the complex network, in the gene search location we write the scientific name of the gene correctly in the text box field through the keyboard on the screen and pressing the search gene button, and this facilitates the time of accessing the genes as in the figure 37.

Figure 37: Gene search in the biological network

Upon completion of work on the network, it is possible to return to the first scene (the room) by entering the black hole located near the network in space, as in Figure 38.
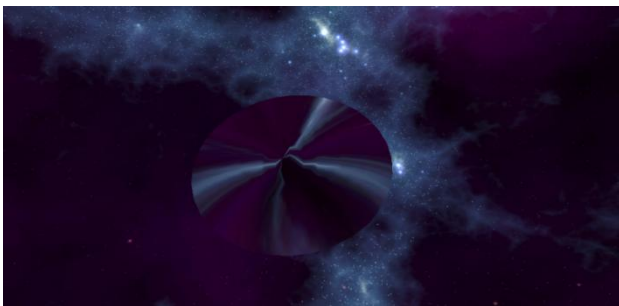

Figure 38: The black hole to return to the first scene (the room)

3.2 2D and 3D Visualization

This section presents the 2D and 3D visualizations of the biological network. The purpose of presenting both models is to show the differences between them and further use them in the usability test.

3.2.1 2D Visualization

The 2D visualization uses only flat graphics with X and Y axes. In Unity, they are called sprites and do not have 3D work geometry. It is visualized flat on the screen, and the camera has no perspective as in Figure 39.
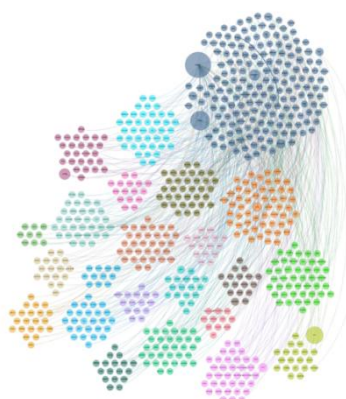
Figure 39: The biological network model in 2D visualization

3.2.2 3D Visualization

The 3D visualization of the biological network is shown in Figure 40. In our research, the data can be distributed and visualized in a more efficient way and clearer than the 2D model. Furthermore, through virtual reality technology, interaction with data, touching nodes and moving them are easier, and this is the basis of our work in this research.
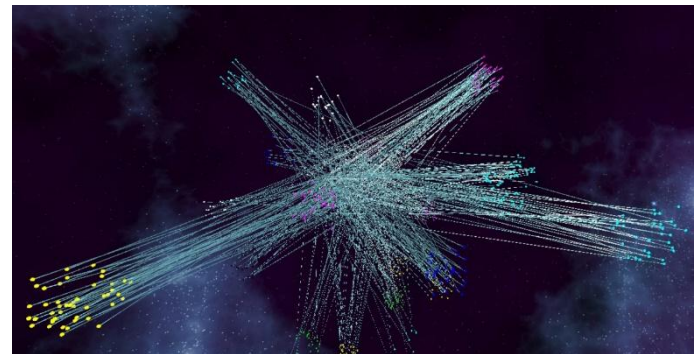

Figure 40: Three-Dimensional biological network

3.2.3 Usability Test

A questionnaire of 10 questions was designed and then distributed to 35 users of different specializations (see Table 4). The strategy followed in retrieving feedback was based on meeting each user individually and showing him/her the 2D network model, and then letting them experience and dive into the proposed BioVerse-VR 3D model using the required VR hardware. Thereafter, users were provided with the BioVerse-VR usability questionnaire to answer the questions. Each question includes a scale ranging from 1 to 5, where 5 = Strongly Agree, 4 = Agree, 3 = Neutral, 2 = Disagree, and 1 = Strongly Disagree.

Table 4: Applicants involved in testing the usability of the BioVerse application

#	Specialization	# of Applicants
1	Experts in Software Engineering	2
2	Experts in Biology	5
3	Experts in Computer Science	25
4	Experts in other specializations	3

Now, the feedback obtained is summarized for each question as follows:

- **Question 1:** "*When I used the devices and sensors, I did not find any difficulty in dealing with the network*". The feedback shows 57% of the applicants answered

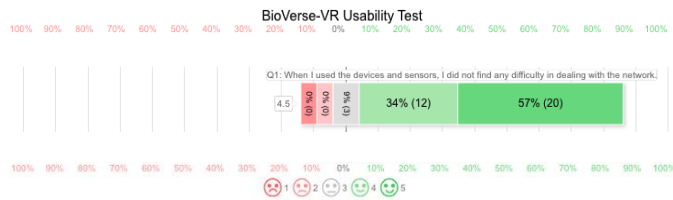"strongly agree" and 34% answered, "agree" (see Figure 41).

Figure 41: Question 1 Likert Chart of the responses

▪ **Question 2:** "*My interactions with the virtual environment seemed natural*". The feedback shows 37% of the applicants answered "strongly agree" and 46% answered, "agree" (see Figure 42).
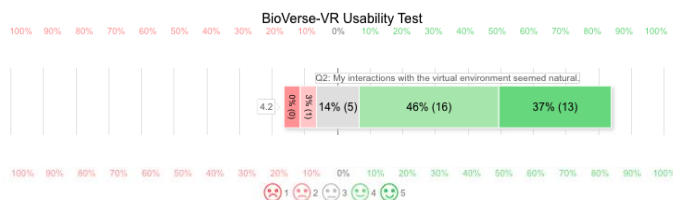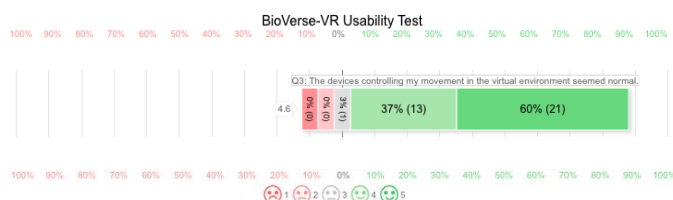

Figure 42: Question 2 Likert Chart of the responses

▪ **Question 3:** "*The devices controlling my movement in the virtual environment seemed normal*". The feedback shows 60% of the applicants answered "strongly agree" and 37% answered, "agree" (see Figure 43).

Figure 43: Question 3 Likert Chart of the responses

▪ **Question 4:** "*I was able to actively scan the virtual environment and network using vision and motion*". The feedback shows 60% of the applicants answered "strongly agree" and 34% answered, "agree" (see Figure 44).
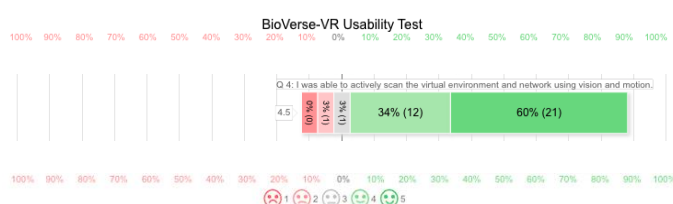

Figure 44: Question 4 Likert Chart of the responses

▪ **Question 5:** "*I was able to closely examine the network and interact with it*". The feedback shows 69% of the applicants answered "strongly agree" and 29% answered, "agree" (see Figure 45).
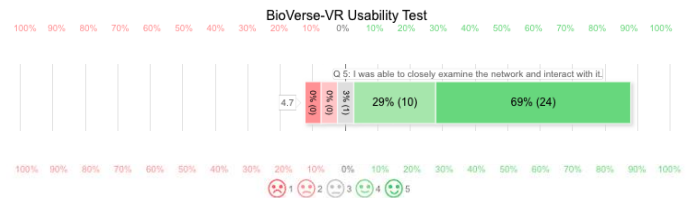

Figure 45: Question 5 Likert Chart of the responses

▪ **Question 6:** "*I can focus on tasks and interact with the network rather than on devices*". The feedback shows 37% of the applicants answered "strongly agree" and 54% answered, "agree" (see Figure 46).
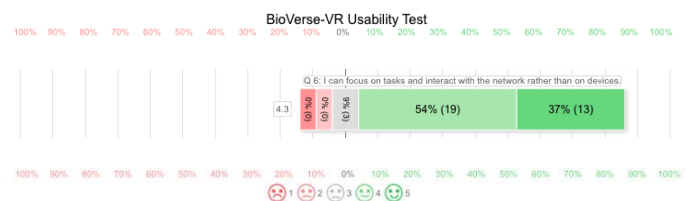

Figure 46: Question 6 Likert Chart of the responses

▪ **Question 7:** "*There is a noticeable difference between the two-dimensional network and the network in the virtual environment, which seemed clearer and more understandable*". The feedback shows 94% of the applicants answered "strongly agree" and 6% answered, "agree" (see Figure 47).
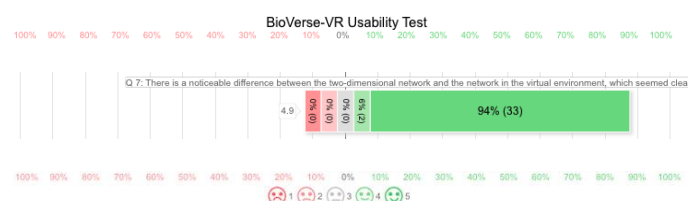

Figure 47: Question 7 Likert Chart of the responses

▪ **Question 8:** "*The amount of information you got from the grid in the virtual environment is much greater than what you got from the two-dimensional grid*". The feedback shows 51% of the applicants answered "strongly agree" and 40% answered, "agree" (see Figure 48).
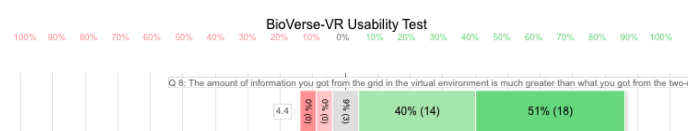
Figure 48: Question 8 Likert Chart of the responses

▪ **Question 9:** "*The design of the biological network was neat and clear to any user*". The feedback shows 46% of the applicants answered "strongly agree" and 49% answered, "agree" (see Figure 49).
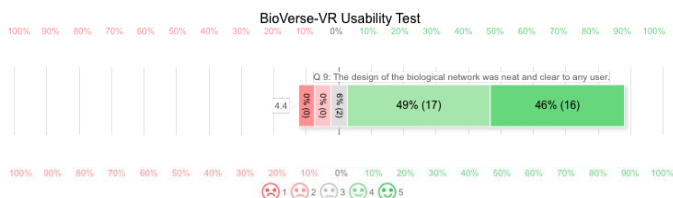

Figure 49: Question 9 Likert Chart of the responses

▪ **Question 10:** "*Interacting with the nodes and edges of the network was useful*". The feedback shows 51% of the applicants answered "strongly agree" and 40% answered, "agree" (see Figure 50).
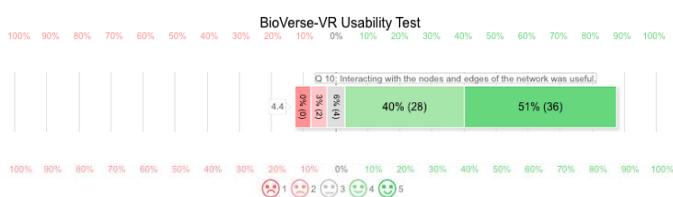

Figure 50: Question 10 Likert Chart of the responses

The results showed that most of the test results were positive and considered excellent. This means the BioVerse application is efficient and can be adopted by biologists for research purposes. In addition to the usability test, BioVerse is scalable since nodes and edges can be added smoothly. Also, the availability of the application objects (e.g., nodes, edges, and other objects) is always shown during the interactions with the model. Based on the questionnaire results, the proposed model is also reliable and valid because it meets most of the needs and expectations of users.

IV. CONCLUSION

This work involved the integration between the field of complex networks and virtual reality. A 3D VR-Based network model was developed to represent the gene-gene interactions. Also, the paper stated the hardware and software requirements that are used in implementing VR-Based network visualization. Furthermore, this work is considered a novel attempt that investigates such an important topic. It should be mentioned that this kind of work will be widely used especially with the advent of the metaverse concepts. It is also believed that VR technology will be the most frequent approach in the near future that solve real-life issues. This

paper answers the questions raised in Section 1 and can be concluded as follows: 1) It was possible to convert a 2D network model for gene-gene interactions into an interactive 3D VR-based network model. 2) The designed 3D VR-based network model provided the ability to interact with data in a comfortable way. 3) Developing a virtual world that contained the network model was important to make a big-enough space for analysts to move and interact freely in the environment. 4) The designed interactive features supported analysts in investigating the network model in an efficient way. 5) The virtual room in the model was necessary when involving more than one user in analyzing the network model. 6) It was crucial to generate a package that can be installed in any VR device and connect to the metaverse world. 7) The design questionnaires were highly useful in assessing the usability of the generated application. This step should be followed in such applications due to its role in improving the design and making it more comfortable for analysts especially when adopting such a modern technology. 8) The main challenges that developers may face during designing metaverse applications can be programming skills, 3D design skills, and the type of VR hardware used and its specifications. Each hardware has its own features that can be different from the features of other devices, which, in turn, affect the interaction features and compatibility of the application.

In addition to the previous conclusions, we state some recommendations that are considered the collective experiences obtained from this work. 1) Working on the topic of this paper needs to have a strong background in programming languages. 2) Understanding the framework of the VR applications is also considered a key role in designing efficient 3D VR-based interactive network models. 3) The first step of designing metaverse applications is to have well-defined goals and a complete perception of the planned application. 4) Testing the usability of the applications by experts is another aspect that should be given special attention by metaverse developers. 5) It is not recommended to use metaverse applications by visually impaired people.

For future work, a large-scale dataset of gene-gene interactions can be involved to have a huge network model that can be of interest to analysts. Also, more interactive features and 3D effects can be added aiming at having a more professional analysis of the network model. Finally, we strongly believe that this paper will open the horizon to metaverse developers in the near future.

ACKNOWLEDGEMENT

REFERENCES

[1] Park, Seungeun, et al. "Impact of data visualization on decision-making and its implications for public health practice: a systematic literature review." Informatics for Health and Social Care 47.2 (2022): 175-193.

[2] Barabási, Albert-László. "Network science." Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 371.1987 (2013): 20120375.

[3] Albert, R. and Barabási, A.L., 2002. Statistical mechanics of complex networks. Reviews of modern physics, 74(1), p.47.

[4] Newman, Mark EJ. "The structure and function of complex networks." SIAM review 45.2 (2003): 167-256.

[5] Barabási, A. L., Gulbahce, N., & Loscalzo, J. (2011). Network medicine: a network-based approach to human disease. Nature reviews genetics, 12(1), 56-68.

[6] Yang, R., Sala, F., & Bogdan, P. (2021). Hidden network generating rules from partially observed complex networks. Communications Physics, 4(1), 1-12.

[7] Mahmood, Basim, and Ronaldo Menezes. "United states congress relations according to liberal and conservative newspapers." 2013 IEEE 2nd Network Science Workshop (NSW). IEEE, 2013.

[8] Yu, Wei, Jun Chen, and Xingchen Yan. "Space–time evolution analysis of the Nanjing metro network based on a complex network." Sustainability 11.2 (2019): 523.

[9] Moreira-Filho, C. A., Bando, S. Y., Bertonha, F. B., Silva, F. N., &Fontoura Costa, L. D. (2022). Methods for Gene Co-expression Network Visualization and Analysis. In Transcriptomics in Health and Disease (pp. 143-163). Springer, Cham.

[10] Xiong, J., Hsiang, E. L., He, Z., Zhan, T., & Wu, S. T. (2021). Augmented reality and virtual reality displays: emerging technologies and future perspectives. Light: Science & Applications, 10(1), 1-30.

[11] El Beheiry, M., Doutreligne, S., Caporal, C., Ostertag, C., Dahan, M., & Masson, J. B. (2019). Virtual reality: beyond visualization. Journal of molecular biology, 431(7), 1315-1321.

[12] Cassidy, K. C., Šefčík, J., Raghav, Y., Chang, A., &Durrant, J. D. (2020). ProteinVR: Web-based molecular visualization in virtual reality. PLoS computational biology, 16(3), e1007747.

[13] Pirker, J., Kopf, J., Kainz, A., Dengel, A., &Buchbauer, B. (2021, March). The potential of virtual reality for computer science education-engaging students through immersive visualizations. In 2021 IEEE Conference on

Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW) (pp. 297-302). IEEE.

[14] Budi, A. S., Sumardani, D., Muliyati, D., Bakri, F., Chiu, P. S., Mutoharoh, M., &Siahaan, M. (2021). Virtual Reality Technology in Physics Learning: Possibility, Trend, and Tools. JurnalPenelitian&Pengembangan Pendidikan Fisika, 7(1), 23-34.

[15] Linssen, L., Landman, A., van Baardewijk, J. U., Bottenheft, C., & Binsch, O. (2022). Using accelerometry and heart rate data for real-time monitoring of soldiers' stress in a dynamic military virtual reality scenario. Multimedia Tools and Applications, 1-18.

[16] Cecotti, H. (2022, May). A Serious Game in Fully Immersive Virtual Reality for Teaching Astronomy Based on the Messier Catalog. In 2022 8th International Conference of the Immersive Learning Research Network (iLRN) (pp. 1-7). IEEE.

[17] Bowman, N. D., Rieger, D., & Lin, J. H. T. (2022). Social video gaming and well-being. Current Opinion in Psychology, 101316.

[18] Fan, X., Jiang, X., & Deng, N. (2022). Immersive technology: A meta-analysis of augmented/virtual reality applications and their impact on tourism experience. Tourism Management, 91, 104534.

[19] Safikhani, S., Keller, S., Schweiger, G., & Pirker, J. (2022). Immersive virtual reality for extending the potential of building information modeling in architecture, engineering, and construction sector: systematic review. International Journal of Digital Earth, 15(1), 503-526.

[20] Halpin, H., Zielinski, D. J., Brady, R., & Kelly, G. (2008, October). Exploring semantic social networks using virtual reality. In International Semantic Web Conference (pp. 599-614). Springer, Berlin, Heidelberg.

[21] Conte, G., Ye, A. Q., Forbes, A. G., Ajilore, O., & Leow, A. (2015, August). BRAINtrinsic: A virtual reality-compatible tool for exploring intrinsic topologies of the human brain connectome. In International Conference on Brain Informatics and Health (pp. 67-76). Springer, Cham.

[22] Kuznetsov, M., Elor, A., Kurniawan, S., Bosworth, C., Rosen, Y., Heyer, N., ... & Haussler, D. (2021, August). The Immersive Graph Genome Explorer: Navigating Genomics in Immersive Virtual Reality. In 2021 IEEE 9th International Conference on Serious Games and Applications for Health (SeGAH) (pp. 1-8). IEEE.

[23] Shannon, R. J., Deeks, H. M., Burfoot, E., Clark, E., Jones, A. J., Mulholland, A. J., &Glowacki, D. R.

(2021). Exploring human-guided strategies for reaction network exploration: Interactive molecular dynamics in virtual reality as a tool for citizen scientists. The Journal of Chemical Physics, 155(15), 154106.

[24] Lyu, Z., Li, J., & Wang, B. (2021, November). AIive: Interactive Visualization and Sonification of Neural Networks in Virtual Reality. In 2021 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR) (pp. 251-255). IEEE.

[25] Ganglberger, F., Wibmann, M., Wu, H. Y., Swoboda, N., Thum, A., Haubensak, W., & Buhler, K. (2022). Spatial-data-driven layouting for brain network visualization. Computers & Graphics, 105, 12-24.

[26] Cheng, Kun-Hung, Kai-Yu Tang, and Chin-Chung Tsai. "The mainstream and extension of contemporary virtual reality education research: Insights from a co-citation network analysis (2015–2020)." Educational technology research and development (2022): 1-16.

[27] Hellum, Owen, Christopher Steele, and Yiming Xiao. "SONIA: an immersive customizable virtual reality system for the education and exploration of brain networks." arXiv preprint arXiv: 2301.09772 (2023).

[28] W. M. Aioffi, C. A. Valle, G. R. Mateus, and A. S. da Cunha, "Balancing message delivery latency and network lifetime through an integrated model for clustering and routing in wireless sensor networks," *Comput. Netw.,* vol. 55, no. 13, pp. 2803–2820, Sep. 2011.

AUTHORS BIOGRAPHY

Ahmed Hisham, College of Computer Science and Mathematics, Computer Science Department, University of Mosul, Mosul, IRAQ. E-mail: Ahmed.21csp92@student.uomosul.edu.iq

Basim Mahmood, College of Computer Science and Mathematics, Computer Science Department, University of Mosul, Mosul, IRAQ. E-mail: bmahmood@uomosul.edu.iq

---

Citation of this Article:

Ahmed Hisham, Basim Mahmood, "VR-Based Visualization for Biological Networks" Published in *International Research Journal of Innovations in Engineering and Technology - IRJIET*, Volume 7, Issue 5, pp 48-68, May 2023. Article DOI https://doi.org/10.47001/IRJIET/2023.705007

****