BIOMATH
https://biomath.math.bas.bg/biomath/index.php/biomath
Biomath Forum
RESEARCH COMMUNICATION

Integrating mixed reality technologies in genomic data visualization and analysis for bioinformatics research

Tereza Trencheva[1], Ivan Trenchev[1,*], Iglika Getova[1], Miglena Trencheva[2]

[1]University of Library Studies and Information Technologies, Bulgaria
t.trencheva@unibit.bg 0000-0002-7527-1070
i.trenchev@unibit.bg 0000-0002-8726-4775
i.getova@unibit.bg 0000-0002-8512-7814

[2]South-West University "Neofit Rilski", Bulgaria
megy_tr2001@swu.bg 0000-0003-3170-8821

Received: August 30, 2023, Accepted: March 12, 2024, Published: May 15, 2024

Abstract: **With the advancement of Mixed Reality (MR) technologies and bioinformatics, researchers are exploring new ways to enhance the visualization and analysis of genomic data. The integration of MR technologies in bioinformatics research has the potential to revolutionize the way scientists interpret complex biological information. This article discusses the application of MR in genomic data visualization and analysis, highlighting its advantages in facilitating a more immersive and interactive experience. In particular, we will present case studies related to the implementation of the Unreal Engine in MR for bioinformatics research.**

As part of the research, the role of intellectual property in bioinformatics will be analyzed, providing insights into its significance and implications in the field. The integration of MR can improve collaboration among researchers and assist in the understanding of intricate patterns within genomic datasets. Furthermore, the article examines the challenges faced in implementing MR technologies in bioinformatics and addresses possible solutions to overcome these obstacles.

Overall, the integration of MR in bioinformatics research has the potential to reshape the field and drive innovation in genomic data analysis.

Keywords: **mixed reality, genomic data visualization, style, styling**

I. Introduction

With the rapid progress of Mixed Reality (MR) technologies and the field of bioinformatics, researchers have been actively exploring novel avenues to enhance the visualization and analysis of genomic data. Genomics, which involves studying the complete set of genes in an organism, generates vast amounts of complex biological information that can be challenging to interpret and analyze effectively. The integration of MR technologies in bioinformatics research has emerged as a promising solution with the potential to revolutionize the way scientists interact with and understand genomic data. This article aims to delve into the application of MR in the visualization and analysis of genomic data, highlighting its advantages and potential for facilitating a more immersive and interactive experience.

MR, which combines elements of virtual reality (VR) and augmented reality (AR), enables users to perceive and interact with both virtual and real-world objects in a merged environment. By leveraging MR, researchers can visualize genomic data in three-dimensional (3D) space, providing a more intuitive representation of complex genetic structures and patterns.

*Corresponding author

Citation: Tereza Trencheva, Ivan Trenchev, Iglika Getova, Miglena Trencheva,
Integrating mixed reality technologies in genomic data visualization and analysis for bioinformatics research,
Biomath 13 (2024), 2403126, https://doi.org/10.55630/j.biomath.2024.03.126

One significant advantage of MR in bioinformatics research is its potential to improve collaboration and data interpretation among researchers. MR technologies enable multiple users to share a virtual environment and collaborate in real-time, irrespective of their physical locations. This collaborative aspect of MR can greatly enhance the exchange of ideas, foster interdisciplinary collaborations, and facilitate the collective understanding of intricate genomic datasets [1].

The implementation of MR technologies, such as the utilization of the powerful Unreal Engine, holds great promise in bioinformatics research. The Unreal Engine, known for its capabilities in rendering high-fidelity virtual environments, can be leveraged to create visually stunning and realistic representations of genomic data. This integration enables researchers to explore genomic structures, analyze gene interactions, and gain valuable insights into genetic variations in an immersive and interactive manner.

Moreover, this research also addresses the role of intellectual property in the context of bioinformatics. It analyzes the significance of protecting intellectual property rights in the field, considering the rapid advancements and potential commercialization of MR technologies in bioinformatics. Understanding the implications of intellectual property is crucial for fostering innovation, incentivizing research, and ensuring ethical and responsible use of MR technologies in the bioinformatics domain.

While the integration of MR in bioinformatics research offers tremendous opportunities, it is not without its challenges. This article acknowledges the obstacles encountered in implementing MR technologies, such as the need for robust computational infrastructure, data integration and processing issues, and the requirement for user-friendly interfaces.

II. Methodology

The methodology described in this research focuses on the integration of Unreal Engine, MATLAB, and NAMD (Nanoscale Molecular Dynamics) for genomic data visualization and analysis. Here is an overview of the methodology:

Preprocessing and analysis in MATLAB: The process begins with preprocessing and analyzing genomic data using MATLAB. This involves employing various bioinformatics tools and algorithms to extract relevant data and insights from the genomic datasets

Data preparation for visualization: MATLAB is utilized to generate the necessary data and parameters for visualizing the genomic data. This includes generating coordinate information for atoms, defining molecular structures, and extracting other relevant attributes required for the visualization.

Interfacing with NAMD: NAMD is a powerful molecular dynamics simulation software. In this methodology, NAMD serves as the engine for simulating the dynamic behavior of molecules based on the data generated in MATLAB. The data prepared in MATLAB is fed into NAMD for real-time simulation and analysis.

Integration with Unreal Engine: Unreal Engine, a robust game development engine, is used for creating immersive and interactive visualizations of the genomic data. The integration is achieved by establishing a connection between Unreal Engine and NAMD. This connection enables the real-time transfer of simulation data from NAMD to Unreal Engine, allowing for dynamic visualization of molecular structures and interactions.

Visualization and interaction in Unreal Engine: Researchers can develop interactive and visually compelling representations of the genomic data, using Unreal Engine's visual scripting system (Blueprint) and potentially supplementing it with C++ programming. This includes rendering 3D models of molecules, visualizing atom-level details, and incorporating additional visual effects to enhance the understanding of complex biological structures.

Real-time analysis and exploration: With the integration of NAMD and Unreal Engine, researchers can analyze the simulated molecular dynamics in real time. They can interactively explore the molecular structures, study molecular interactions, and gain insights into genetic variations and patterns within the genomic data. The system allows for dynamic adjustments and real-time feedback during the analysis process.

Evaluation and validation: The generated visualizations and analysis results can be evaluated and validated through comparative analysis with existing visualization methods, benchmark datasets, or expert feedback. The accuracy, performance, and utility of the integrated system can be assessed to ensure its effectiveness in supporting bioinformatics research.

By combining the capabilities of Unreal Engine, MATLAB, and NAMD, this methodology enables researchers to integrate dynamic simulation, advanced visualization, and interactive exploration for genomic data analysis.

The integration of Unreal Engine and MATLAB facilitates a seamless workflow from data processing to real-time visualization, while NAMD empowers the system with molecular dynamics simulation capabilities. This integrated approach holds promise in

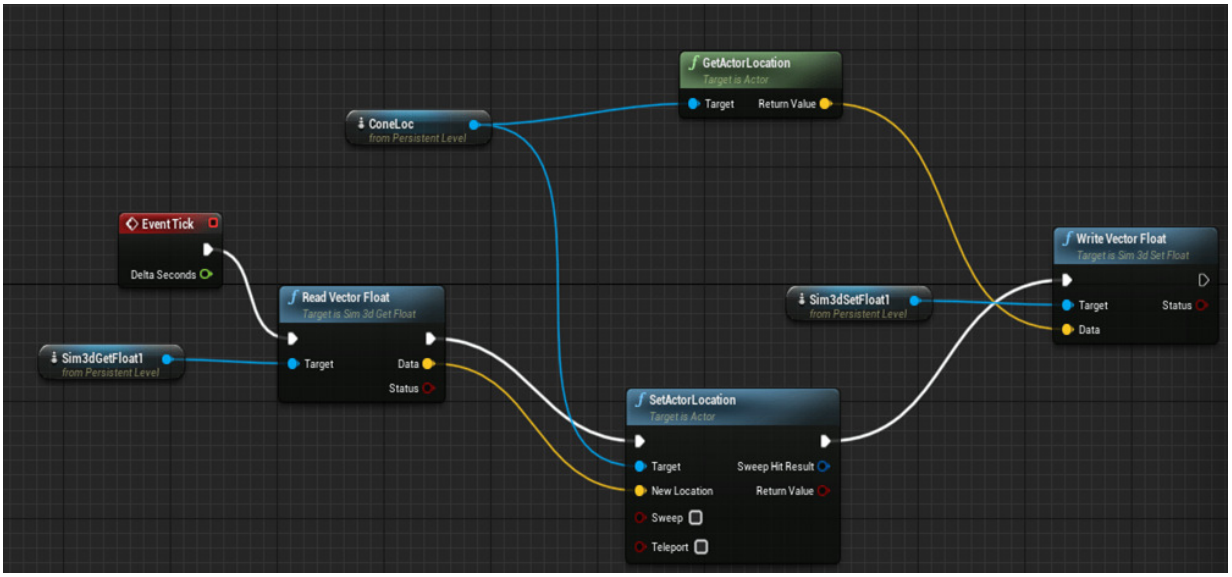

Fig. 1: Connection between MATLAB and Unreal Engine using Blueprint

advancing bioinformatics research and enhancing the understanding of complex genomic phenomena.

III. Results

In the rapidly evolving realm of computer-assisted quantum chemistry and molecular biophysics, there's an imperative need for methodologies that cohesively amalgamate both analytical and visualization data processing into an uninterrupted system. In this investigation, we introduce a conceptual framework capitalizing on the functional capabilities of NAMD, MATLAB, Unreal Engine, and the adaptive Natural Language Processing (NLP) AI language model, thereby constructing a synergistic platform for quantum structure interpretation.

NAMD, through its sophisticated molecular dynamics algorithms, yields nuanced multi-dimensional datasets of atomic trajectories and their interactions. MATLAB transforms these datasets, applying quantum-statistical and topological methods for their intricate deconstruction. Unreal Engine metamorphoses these outcomes into intuitive yet detailed three-dimensional visualizations. Culminating the process, the adaptive NLP AI language model, which learns in real-time and self-corrects from each interpretation error, conducts a semantic analysis of the visualizations, producing an amplified and enriched scientific comprehension.

Within the purview of computer-assisted quantum chemistry and molecular biophysics, this study unveils a pioneering integration between MATLAB and Unreal Engine [2]. This nexus underscores distinct methodologies formulated by the authors, with a quintessential outcome manifested in visualizations from Figures 1 through 3. These depictions elucidate the potential and innovations stemming from this alliance, showcasing an avant-garde approach in quantum-mechanical analysis and its data representation (see Fig. 1-3). The integration can be actualized in the following manner [3]:

Using the file system: One of the simplest ways to connect the two software is by using the file system. MATLAB can generate or process the necessary data and save it in a file format that Unreal Engine can read. Then, Unreal Engine can utilize this data for visualization or other purposes.

Using TCP/IP communication: Utilizing TCP/IP communication provides a more advanced way of connecting MATLAB and Unreal Engine. You can use TCP/IP sockets in each environment and establish communication between them by sending and receiving data over the network. This requires implementing code to create TCP/IP clients or servers in each environment.

Using specialized tools and plugins: There are tools and plugins available that can facilitate the connection between MATLAB and Unreal Engine. For example, you can explore the MATLAB Engine API, which allows executing MATLAB code in other programming languages, including C++, the language of Unreal Engine. This enables you to execute MATLAB code directly from Unreal Engine and communicate with the results.

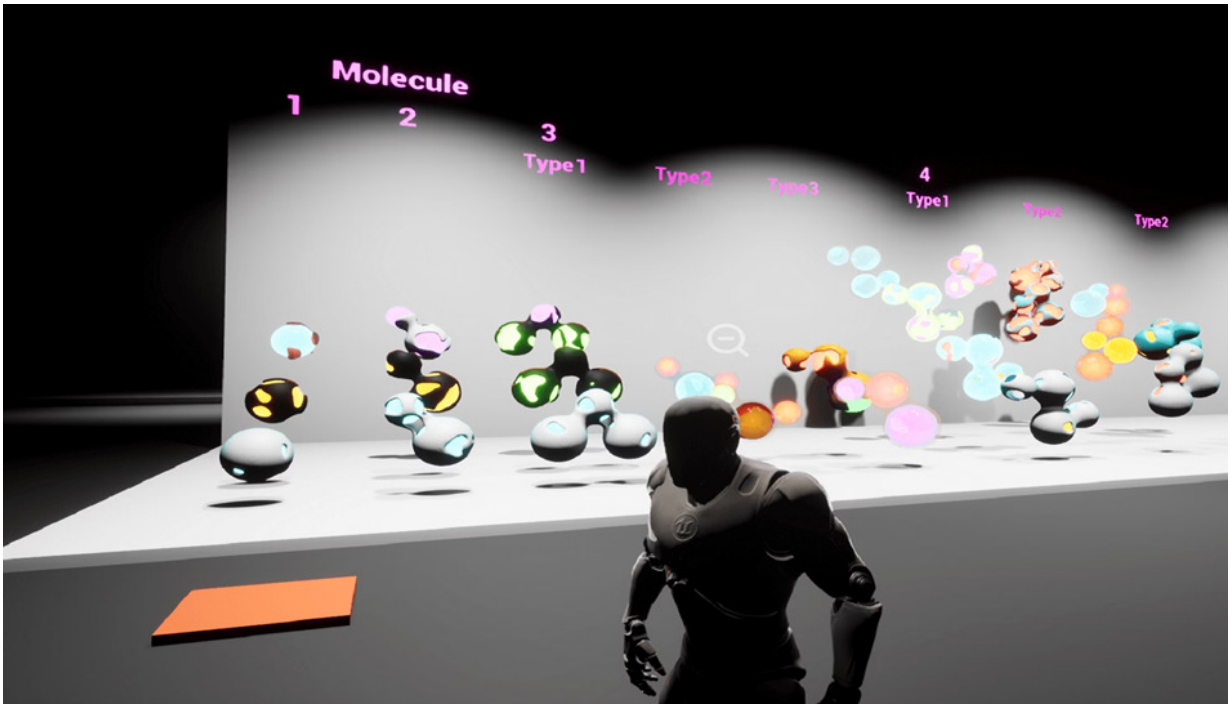

Fig. 2: Using MR for vitalization a biological data

Using external data exchange methods: You can utilize external tools or formats for data exchange between MATLAB and Unreal Engine. For instance, you can use JSON or XML formatting to represent and exchange data between the two environments. MATLAB can generate or process data in the respective format, and Unreal Engine can interpret and use it for visualization or other purposes.

Using middleware solutions: Middleware solutions exist that provide integration between different software platforms. These solutions can be used to connect MATLAB and Unreal Engine. For example, you can explore options like ZeroMQ, RabbitMQ, or ROS (Robot Operating System) as potential middleware tools that can be used for communication between the two software.

Using MEX functions: In MATLAB, you have the ability to create MEX functions (C/C++ functions) that can be called directly from MATLAB. This allows you to create MEX functions that execute C++ code and interact with Unreal Engine. Thus, you can execute MATLAB code that communicates with Unreal Engine through MEX functions.

These are some of the methods for connecting MATLAB and Unreal Engine. The choice of approach depends on your specific requirements and implementation capabilities. It is important to maintain data compatibility and ensure the proper transmission and interpretation of information between the software components.

In the overarching endeavor to amalgamate quantum mechanics and bioinformatics, our research undertook the formidable task of bridging NAMD and MATLAB, further enriched by the inclusion of open-source NLP AI potentialities. The goal was a symbiotic blend of the computational prowess of these systems to harness the expansive realm of quantum biochemical interactions [4].

During the course of our experimentations, we sought to realize the implementation of MATLAB-centric algorithms tailored for quantum mechanical computations. These implementations particularly [5] leveraged an eclectic spectrum of wave functions within the ambit of Hybrid QM/MM modeling, as facilitated by NAMD [6]. For elucidative purposes, consider the integration paradigm where MATLAB's ode45 solver commingles with NAMD's Langevin dynamics, underpinned by adaptive employment of wave functions [7]:

$$\Phi(k,t) = \frac{1}{\sqrt{h}} \int \Psi(x,t) \mathrm{e}^{-ikx} dx$$
$$\Psi(x,t) = \frac{1}{\sqrt{h}} \int \Phi(k,t) \mathrm{e}^{ikx} dk$$

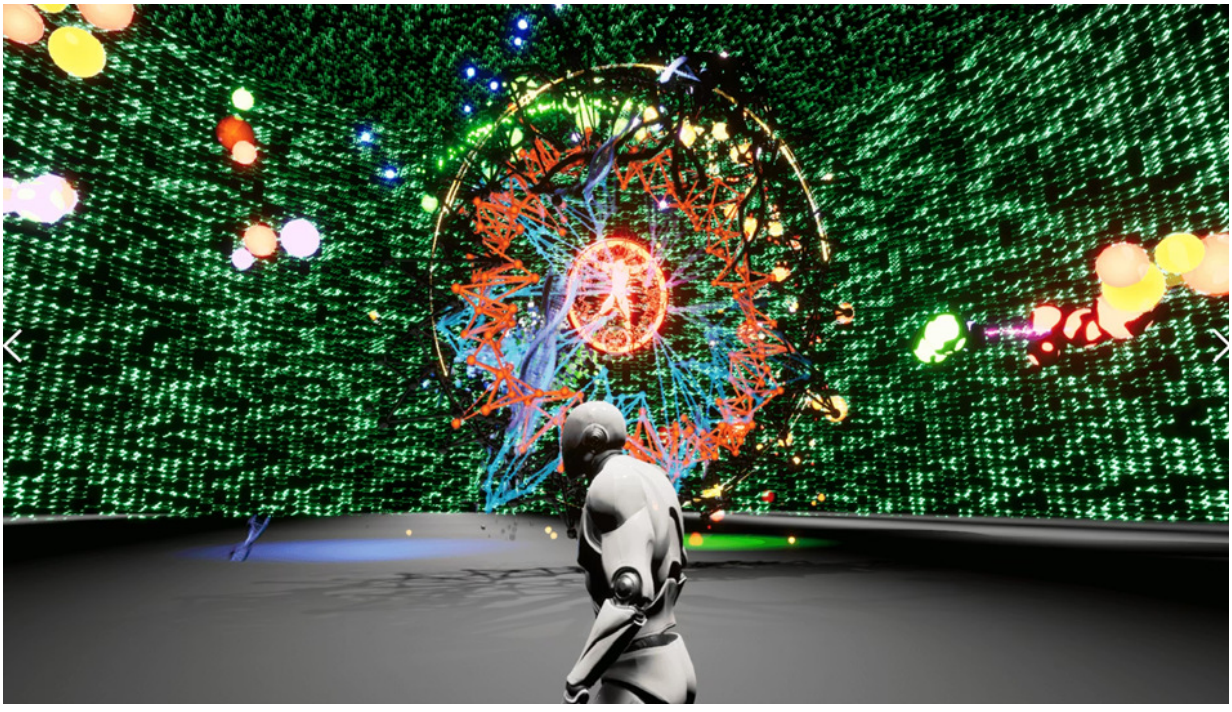

Fig. 3: Molecules and Organisms VFX

$$
\begin{aligned}
v_{\mathrm{g}} &= p/m, \;\; E = \hbar\omega \\
\langle f(t)\rangle &= \iiint \Psi^* f \Psi d^3V \\
\langle f_p(t)\rangle &= \iiint \Phi^* f \Phi d^3V_p \\
\langle f(t)\rangle &= \langle \Phi | f | \Phi \rangle \\
\langle \Phi | \Phi \rangle &= \langle \Psi | \Psi \rangle = 1
\end{aligned}
$$

In the field of Nanoscale Molecular Dynamics (NMD), wave functions play a critical role in describing the quantum behavior of particles at the nanoscale. These wave functions are solutions to the Schrödinger equation, which is the fundamental equation governing the dynamics of quantum systems [8].

Initial encounters, however, were mired in complexities. The origins of these setbacks were multifaceted, ranging from compatibility conflicts, innate divergences in software philosophies, to the intrinsic intricacies of the deployed wave functions. These initial endeavors, regrettably, yielded catastrophic outcomes.

Nevertheless, the spirit of scientific inquiry demanded perseverance. Following rigorous code optimizations, rectifications of methodological lacunae, and incessant refinement of wave functions, a watershed moment was achieved. Our posterior analyses began to exhibit encouraging indicators, ushering a new horizon in the realm of integrative bioinformatics.

Listing 1: Integration of MATLAB with NAMD's Langevin dynamics

```
% Initialization of parameters and
    primordial wave functions
params = namd2matlab('config.namd');
psi_initial = waveFunctionInit(params);

% Deploying ode45 for resolution
[t, y, psi] = ode45(@namdFunc, [0 Tfinal
    ], y0, [], params, psi_initial);

function [dy, dpsi] = namdFunc(t, y,
    params, psi)
    % Extracting forces and updating wave
    functions via the NLP AI conduit
    [forces, new_psi] =
    getForcesAndWaveFunctionsNLP(y,
    params, psi);
    dy = NAMDIntegrate(y, forces, params);
    dpsi = waveFunctionEvolve(new_psi,
    params);
end
```

The Hartree-Fock method is another approach where the wave function is approximated as a Slater determinant. The method is computationally intensive but provides accurate results.

Examples of usage in NAMD [9]:

QM/MM simulations: Here, "QM" refers to quantum mechanical calculations, and "MM" refers to classical molecular dynamics calculations. In such simulations, the region of interest (e.g., the active site of an enzyme) is modeled using quantum mechanical methods, while the rest of the system is treated with classical molecular dynamics. This allows for accurate modeling of quantum effects where they are most significant without a substantial computational overhead [10].

Electronic structure calculations: In some cases, especially when studying chemical reactions or bindings, electronic structure calculations might be necessary to determine potential energy surfaces or electronic transitions. This typically requires the integration of NAMD with quantum chemical software packages.

It's worth noting that while NAMD has functionality for QM/MM simulations, detailed quantum chemical calculations are usually done using other specialized software packages.

To facilitate advancements in bioinformatics research, we embarked on the integration of voice commands within the Unreal Engine framework. Although our developments are still in their nascent stages, preliminary results are promising and indicate substantial potential for future academic and industrial applications.

By amalgamating the capabilities of MATLAB, NAMD, and Unreal Engine, we have devised a system that allows for real-time dynamic visualization and analysis of genomic structures. The voice command interface offers rapid navigation, manipulation, and interpretation of data.

To illustrate a segment of our work, we present the following Blueprint code:

```
Event BeginPlay
-> Initialize Voice Command Module
-> Bind Voice Commands [English/Bulgarian
    ]
-> Event Tick
   -> Listen for Voice Commands
      -> On Command Recognized
         -> Execute Associated Genomic
    Analysis Function
```

The integration of voice commands within the Unreal Engine for bioinformatics applications represents a complex scientific endeavor that encounters a series of computational and biological challenges. The primary issue in the realm of voice recognition is tied to semantic ambiguity, which frequently arises when interpreting domain-specific terms. To overcome this challenge, natural language processing algorithms enriched with bioinformatics-specific ontologies are employed [11].

There's also a challenge in integrating heterogeneous systems like MATLAB, NAMD, and Unreal Engine, as they possess distinct architectural characteristics. This issue is addressed by creating specialized intermediary interfaces and data exchange protocols.

Furthermore, the dynamic visualization of genomic structures demands high performance and minimal latency. This is achieved through the optimization of visualization algorithms and the utilization of parallel computations.

IV. Mathematical formulation of voice command integration in bioinformatics applications

A. Voice Recognition with Semantic Ambiguity

Let $V(s)$ be a function that interprets a voice signal s into a set of possible textual interpretations T. The objective is to determine:

$$t^* = \arg\min_{t \in T} \text{Ambiguity}(t) \tag{1}$$

where $\text{Ambiguity}(t)$ quantifies the semantic ambiguity of interpretation t.

B. Integration of Heterogeneous Systems

Given systems $S_1, S_2, \ldots, S_n$, the integration function between any two systems S_i and S_j is given by:

$$I(S_i, S_j) : \text{Data}_{S_i} \rightarrow \text{Data}_{S_j} \tag{2}$$

where Data_{S_i} and Data_{S_j} represent the data structures of systems S_i and S_j respectively.

C. Dynamic Visualization

For genomic data D, the visualization function $V(D)$ should be optimized to minimize latency L:

$$\min V(D) \quad \text{subject to} \quad L(V) \leq \epsilon \tag{3}$$

where ϵ is a predefined acceptable latency threshold.

In the endeavor to integrate voice commands into bioinformatics applications, we confront the challenge of semantic ambiguity, encapsulated by the function $V(s)$. This function interprets a voice signal s into a set of potential textual interpretations T. Such ambiguity can lead to erroneous or unintended outcomes. To counteract this challenge, we employ advanced natural language processing algorithms fortified with bioinformatics ontologies. These ontologies offer a structured representation of knowledge, enabling the system to minimize the error function $E(I)$ and select the most accurate interpretation t^*.

Furthermore, we face the intricacy of integrating heterogeneous systems, represented by $S_1, S_2, \ldots, S_n$. These systems exhibit distinct architectural nuances, complicating their seamless integration. To surmount this obstacle, we devise the function $I(S_i, S_j)$ that amalgamates data between any two systems S_i and S_j.

Lastly, the dynamic visualization of genomic structures necessitates the optimization of the function $V(D)$ to ensure that the latency L is minimized. This is achieved through the deployment of parallel computations and advanced visualization algorithms.

The integration of tools such as MATLAB, NAMD, Unreal Engine 5, and the NLP technologies presents a sophisticated scientific inquiry necessitating a meticulous examination of the mathematical and algorithmic characteristics inherent to each of these systems.

When endeavoring to amalgamate these tools, potential discrepancies associated with the mathematical models employed by each system might emerge. Such incongruities could lead to instability within the integrated framework and result in analytical aberrations.

Furthermore, optimizing performance in such an integration might prove challenging due to the disparate computational demands of each tool. This could necessitate a reevaluation of the algorithms and data structures utilized by each system to achieve optimal efficiency.

Additionally, genomic data are inherently intricate and voluminous, potentially augmenting computational complexity when integrating diverse tools. Our current endeavors are centered on comprehending and addressing these challenges.

Modern scientific research increasingly relies on software solutions that facilitate the analysis, visualization, and interpretation of complex data. Open-source software offers invaluable advantages in this context, providing transparency, flexibility, and accessibility:

Transparency and Flexibility: One of the primary advantages of open-source software is its transparency, allowing researchers to delve into the code and understand exactly how a particular tool operates. This also offers the opportunity for modification and adaptation of the software to meet specific research needs [12].

Compatibility and Integration: Open-source software often offers greater compatibility with other tools. For instance, Unreal Engine 5 can be used for free and provides a vast array of visualization functions. This makes it an ideal tool for bioinformaticians who want to visualize complex biological structures in 3D space.

Reduced Costs: Open-source software reduces operational costs for researchers, especially when working in an academic environment with a limited budget. Octave, for example, serves as an excellent alternative to MATLAB and offers many similar functionalities without the associated licensing fees [13].

Community Support: Open-source software is often maintained by active communities that provide resources, guidelines, and patches. This ensures the software is regularly updated and responds to the new requirements and challenges in scientific research.

Open-source software plays a pivotal role in contemporary scientific research, offering solutions that are flexible, adaptive, and cost-effective. In the realm of bioinformatics, where the analysis and interpretation of large datasets are paramount, such tools can substantially contribute to progress and innovation [7].

V. Conclusion

The convergence of Mixed Reality (MR) technologies with bioinformatics presents a promising frontier for the enhancement of genomic data analysis. As elucidated in this article, MR provides researchers with a more immersive and intuitive way to interpret intricate biological information, enabling them to interact with and understand complex genomic patterns in unprecedented ways. The incorporation of platforms such as the Unreal Engine has not only showcased the feasibility of such integration but has also highlighted the potential to revolutionize the manner in which we conduct bioinformatics research.

Furthermore, intellectual property emerges as a pivotal facet within this domain, emphasizing the importance of ensuring rightful ownership, fostering innovation, and facilitating the effective dissemination of knowledge. While the marriage of MR technologies and bioinformatics is not without its challenges, these hurdles are not insurmountable. With dedicated efforts towards problem-solving and collaboration, the impediments can be addressed effectively.

In the grand tapestry of scientific research, it is collaborations such as these - between advanced technology and intricate biology - that herald a new era of understanding and innovation. The potential of MR in bioinformatics not only reshapes our current landscape but also paves the way for future breakthroughs in genomic data analysis.

Acknowledgment

This research would not have been possible without the financial assistance of the following project: *A Conceptual Educational Model for Enhancing Information Literacy in an University Information Environment*,

financed by the National Science Fund of the Ministry of Education and Science of Bulgaria with Contract No KP-06-H35/10 from 18.12.2019, led by Prof. DSc Stoyan Denchev.

REFERENCES

[1] S. Holt, Virtual reality, augmented reality and mixed reality: For astronaut mental health; and space tourism, education and outreach, *Acta Astronautica*, 203:436–446, 2023.

[2] R. Calvo, R. P. Sartoris, O. R. Nascimento, M. Šedivý, A. Sojka, P. Neugebauer, V. T. Santana, Quantum phase transitions probed by EPR spectra in dimeric spin arrays with supramolecular couplings, *Coordination Chemistry Reviews*, 480:48–53, 2023.

[3] C. Jiang, S. Dhamankar, Z. Liu, G. Vinod, G. Shaver, J. Evans, C. Puryk, E. Anderson, D. DeLaurentis, Co-simulation of the Unreal Engine and MATLAB/Simulink for Automated Grain Offoading, *IFAC-PapersOnLine*, 55:379–384, 2022.

[4] J.-L. Lehners, Review of the no-boundary wave function, *Physics Reports*, 1022:1–12, 2023.

[5] M. El Beheiry, S. Doutreligne, C. Caporal, C. Ostertag, M. Dahan, J. B. Masson, Virtual Reality: Beyond Visualization, *Journal of Molecular Biology*, 431:1315–1321, 2019.

[6] L. Kent, C. Snider, J. Gopsill, B. Hicks, Mixed reality in design prototyping: A systematic review, *Design Studies*, 77:101046, 2021.

[7] K. A. Said, A. A. L. Beex, Non-asymptotic bounds for discrete prolate spheroidal wave functions analogous with prolate spheroidal wave function bounds, *Applied and Computational Harmonic Analysis*, 63:20–47, 2023.

[8] Z. O. Memar, M. Moosavi, A combined QTAIM, DFT and molecular dynamics study on the nanoscale dynamical and structural organization of imidazolium-based dicationic ionic liquids, *Journal of Molecular Liquids*, 365:120186, 2022.

[9] D. Liu, C. Zhang, J. Zhang, G. T. Xu, J. Zhang, Molecular pathogenesis of subretinal fibrosis in neovascular AMD focusing on epithelial-mesenchymal transformation of retinal pigment epithelium, *Neurobiology of Disease*, 185:106250, 2023.

[10] M. M. Aboelnga, J. W. Gauld, Comparative QM/MM study on the inhibition mechanism of beta-Hydroxynorvaline to Threonyl-tRNA synthetase, *Journal of Molecular Graphics and Modelling*, 115:108224, 2022.

[11] B. B. Ramos, *Unreal Engine 5 Shaders and Effects Cookbook: Over 50 recipes to help you create materials and utilize advanced shading techniques*, Second Edition, Packt Publishing, 2023.

[12] E. Chu, L. Zaman, Exploring alternatives with Unreal Engine's Blueprints Visual Scripting System, *Entertainment Computing*, 36:100388, 2021.

[13] S. Monji-Azad, M. Kinz, J. Hesser, N. Löw, SimTool: A toolset for soft body simulation using Flex and Unreal Engine, *Software Impacts*, 17:100521, 2023.