Physics of the Earth and Planetary Interiors 298 (2020) 106312

Contents lists available at ScienceDirect

Physics of the Earth and Planetary Interiors

journal homepage: www.elsevier.com/locate/pepi

GeoVReality: A computational interactive virtual reality visualization framework and workflow for geophysical research

Xianying Wang[a], Cong Guo[b], David A. Yuen[c,d], Gang Luo[e,f,*]

[a] Key Laboratory of Computational Geodynamics, College of Earth and Planetary Sciences, University of Chinese Academy of Sciences, Beijing 100049, China
[b] IDSSE—NOITOM Lab for Development of VR Technology in Deep-Sea Science, Institute of Deep-sea Science and Engineering, Chinese Academy of Sciences, Sanya 572000, China
[c] Dept. of Applied Physics and Applied Mathematics, Columbia University, New York, NY 10026, USA
[d] Dept. of Big Data, School of Computer Science, China University of Geosciences, Wuhan 430074, China
[e] School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China
[f] Key Laboratory of Geospace Environment and Geodesy, Ministry of Education, Wuhan University, Wuhan 430079, China

ARTICLE INFO

Keywords:
Virtual reality
Geophysical model
Interactive visualization
Unreal engine
Unity 3D
Big data

ABSTRACT

We present a new interactive computational virtual reality (VR) visualization framework for geophysical Big Data and models for the development of immersive collaborative virtual reality applications with a focus on targeted processing and interaction of Big Data. The framework includes a high-performance scalable persistent storage solution for the spatial analysis of Geospatial Information System (GIS), which uses an engine based on efficient in-memory computing. To more effectively visualize and interact in a VR environment, a machine learning algorithm library is used for compressing and extracting visual data. The framework supports mainstream rendering engines and VR hardware. The framework is extensible, customizable, cross-platform, and it is based only on open source tools. A workflow was introduced, and the geophysical data visualization and interaction effects were demonstrated by taking the abyss data of the Mariana Trench as example.

1. Introduction

Virtual reality is a new medium that offers a lot of potential for data visualization. By immersing ourselves in the data, we can take advantage of the greater space on offer, more natural interactions, and analyze multi-dimensional data viscerally. The best understanding of the concept of virtual reality stems from what it is looking for – a complete immersion. Total immersion means a true sensory experience, and we forget that it is a virtual artificial environment in which people interact naturally with the physical world.

Since most of the large-scale geological and geophysical simulations today are performed in Three-Dimensional (3D) computational space, their data analysis inevitably requires a 3D or stereoscopic display apparatus. We also need a sophisticated user-interface to control the viewpoint. The standard mouse controller for a Personal Computer (PC) is inconvenient enough to choose, set, and adjust the camera position and direction when we want to intuitively observe highly complicated visualization objects in the stereoscopic display system (Fig. 1); meanwhile, data has become so vast and unwieldy that we've upgraded its status to 'Big Data'. The visualization of this data therefore plays an increasingly important role to help us break down its complexity.

VR is different from 3D space and model display on a Two-Dimensional (2D) screen. There are several reasons for using virtual reality for data visualization: (1) Reduce distraction (immersion). By focusing on the entire field of view, you can focus on the goal, whether it helps archaeologists visualize the location of objects of interest during the excavation process (Benko et al., 2004). By "rendering" in the data, one can get a true sense of scale, which is difficult to see when viewing data on a desktop screen. (2) More space. With 360-degree space, there is more room to display data, such as Bloomberg's proof of concept for its virtual reality trading terminal (Seward, 2014). (3) Multi-dimensional data analysis. We mainly use our sight to analyze and interpret the data, but what if we can also use our hearing? Through the data-audio relationship, we can understand the importance, theme, and location of specific data points by the loudness, type, and direction of the sound. By using multiple senses, we can enhance our ability to process more dimensional data. While it may be a bit radical to discuss taste and smell in a data visualization environment, it is not outside the scope of the possibility of "feeling" data (Washburn and Jones, 2004; Choi et al., 2010). This is technically possible now through tactile

* Corresponding author at: Key Laboratory of Geospace Environment and Geodesy, Ministry of Education, Wuhan University, Wuhan 430079, China.
E-mail addresses: gluo@sgg.whu.edu.cn, gangluo66@gmail.com (G. Luo).

https://doi.org/10.1016/j.pepi.2019.106312
Received 26 November 2018; Received in revised form 22 August 2019; Accepted 6 September 2019
Available online 10 September 2019
0031-9201/

Fig. 1. VR with motion capture system (Project Alice by Noitom™ Inc.)

feedback gloves. (4) Greater bandwidth for processing data (Thomas, 2014). Much like a computer, our optic nerve is capable of transferring information at about 1 MB/s. When we simply read words on a screen, we're only using 0.1% of this capacity. Naturally, this would have improved with visualization techniques that have been developed over the years, but at the end of the day this is still about reading information from a 2D screen. Virtual reality immerses you in a stimulating 3D world that engages your brain and enables you to fully utilize your optic nerve's bandwidth. (5) More natural interaction. In the real world we interact with objects directly with our hands. This allows us to connect with the environment around us and get a better idea of the objects we're dealing with. For a long time, we've used keyboards and mice as conduits for this interaction. Through virtual reality, we can return to a more natural way of interacting—by physically pushing buttons, moving windows around and manipulating data streams (such as in this VR assisted biological specimen analysis (Smichrissoft, 2018)). That is in addition to walking around and through these data worlds.

Through these benefits, we can improve employee efficiency, conduct a deeper analysis of data more easily, and make faster decisions. Combining existing software and hardware in a useful way, we elaborate a flexible workflow which can save your development efforts.

2. Key technologies and challenges

2.1. Key elements of a virtual reality experience

Immersive virtual reality aims to create a non-physical world that makes people perceive it as a real physical world. The sense of existence that immersive virtual reality brings to the brain is letting people thought to be somewhere, but it does not exist and is achieved through simple psychological cues and/or physical means (for example holding something in hands or feeling the wind and hot). When a person's various sensations are sufficiently activated to produce perceptions that exist in the non-physical world, the person begins to fully enter the immersive state. To achieve immersive awareness, you need to meet the following elements:

(1) Sensory feedback

Virtual reality needs to be as relevant as possible to our sensory experience. These perceptions include sight, hearing, touch, and so on. Positive feedback and awareness are obtained through the system's hardware and software (input and output) using the senses. Key components of these virtual reality systems include head-mounted displays (Such as HTC Vive (see https://www.vive.com) and Facebook Oculus Rift (see https://www.oculus.com)), special gloves or body accessories (Such as Noitom Hi5Glove (see https://www.noitom.com.cn/hi5/194.html) and CaptoGlove (see https://www.captoglove.com)), and manual controls (Such as Samsung VR Controller (see https://www.samsung.com/global/galaxy/gear-vr) and HTC Vive Tracker (see https://www.vive.com/us/vive-tracker)). In this paper for application, we use HTC Vive for head-mounted displays and trackable controller developed by Noitom Inc.

(2) Interactivity

The vital part of the interactive virtual reality experience requires a comfortable and natural virtual environment for the user. If the virtual environment responds to the user's actions naturally, it is easy to maintain the user's excitement and immersion. If the virtual environment does not respond quickly enough, the human brain will notice quickly, and the immersion will diminish. The response of the virtual environment to the interaction may include changes in the way the participants move or the perspective.

2.2. Challenges in visualization of geophysical data

Traditional data visualization uses computer image processing and graphics technology to transform various types of data into readable and understandable graphic images, helping observers to process and identify data features and patterns. The use of VR technology is to achieve the leap of geophysical results display, breaking through the limitations of the human eye in the real world of recognition and understanding, enabling observers to immerse in the virtual environment and interact with the display results in real time. Thus, how to make massive, multi-dimensional, heterogeneous, dynamic geophysical data in the virtual environment, and then deeper expression of the inherent scientific laws, is a real problem that needs to be solved urgently.

Since complex geophysical data needs to be well explained, geologists and geophysicists will work together. In the process, various geological, geochemical, and geophysical data are used to form a self-consistent model; however, the data itself contains a variety of parameters and attributes, which need to be continuously adjusted. This requires the visualization framework to meet different levels of needs.

First, to dynamically access these data in real time, you need to store them in an efficient database. Second, you may use logical or algebraic calculations when using these data, and you need a real-time computational engine. Third, geoscience data is spatially sensitive and requires a precise coordinate system to regulate it to avoid misunderstandings. Fourth, interactivity, which involves multi-user interactions, also involves user-data interaction. A detailed description is in the architecture introduction.

3. Architecture and workflow

Based on the above functional and technical requirements, we designed a unified geophysical interactive virtual reality visualization process framework (Fig. 2).

3.1. Software modules

3.1.1. MongoDB

MongoDB (MongoDB, Inc., 2019) is a NoSQL database, named “humongous”, is an open source, high performance, scalable, schemaless, document-oriented database (Chodorow and Dirolf, 2010). The internal storage is a kind of JSON-like structured data. The most functional and most relational database in non-relational databases, supporting secondary indexes, supporting redundancy, supporting data fragmentation, and providing good support for massive data. MongoDB provides a set of indexing and query mechanisms to handle geospatial information.

The data structure supported by MongoDB is a loose BSON format. Like binary JSON, data is stored as Key-Values pairs. The Key is used as an index. It can be a file name or an easily distinguishable number. The Value corresponds to the binary of the model. The storage data format is < NameID, Data > .

3.1.2. Apache spark

Apache Spark (The Apache Software Foundation, 2018) is a Big Data parallel computing framework based on In-Memory computing. In-Memory computing improves the real-time performance of data processing in Big Data environment, while ensuring high fault tolerance and high scalability, and allowing users to deploy Apache Spark in a cluster based on cheap hardware (Zaharia et al., 2016). Spark is an alternative to MapReduce and compatible with distributed storage layers such as Hadoop Distributed File System (HDFS) (Borthakur, 2008) and Apache Hive (Thusoo et al., 2009). It can be integrated into the Hadoop ecosystem to make up for the lack of MapReduce.

3.1.3. Geospark

Geospark (https://www.geospark.io) is mainly used for geometric and spatial queries (indexation, position, aggregation, join) of 3D datasets (Yu et al., 2015). GeoSpark is an open source memory cluster computing system for processing large-scale spatial data. It is a combination of traditional geographic information system (GIS) and Spark. GeoSpark extends Resilient Distributed Dataset (RDD) to form Spatial Resilient Distributed Dataset (SRDD) and efficiently partitions SRDD data elements across machines and introduces novel parallelization spaces (geometric operations, following the Open Geospatial Consortium (OGC) standard) transformations and operations (for SRDD), providing a more intuitive interface for users to write spatial data analysis programs. GeoSpark extends the SRDD layer to perform spatial queries (e.g., range queries, K-Nearest Neighbors (KNN) queries, and join queries) on large-scale spatial data sets. After retrieving the geometry object in the SRDD layer, the user can invoke the spatial query processing operations provided in GeoSpark's spatial query processing layer.

3.1.4. Scikit-learn

Scikit-learn (Pedregosa et al., 2011; Buitinck et al., 2013) is an open source Python machine learning library based on Numpy and Scipy (Oliphant, 2006; Jones et al., 2014). It provides many tools for data mining and analysis, including data preprocessing, cross validation, and visualization algorithms. The basic functions of Scikit-learn are mainly divided into six parts, classification, regression, clustering, data dimensionality reduction, model selection, data preprocessing.

3.1.5. ParaView

ParaView (see https://www.paraview.org/) is a program for analyzing and visualizing 2D and 3D data, both as an application

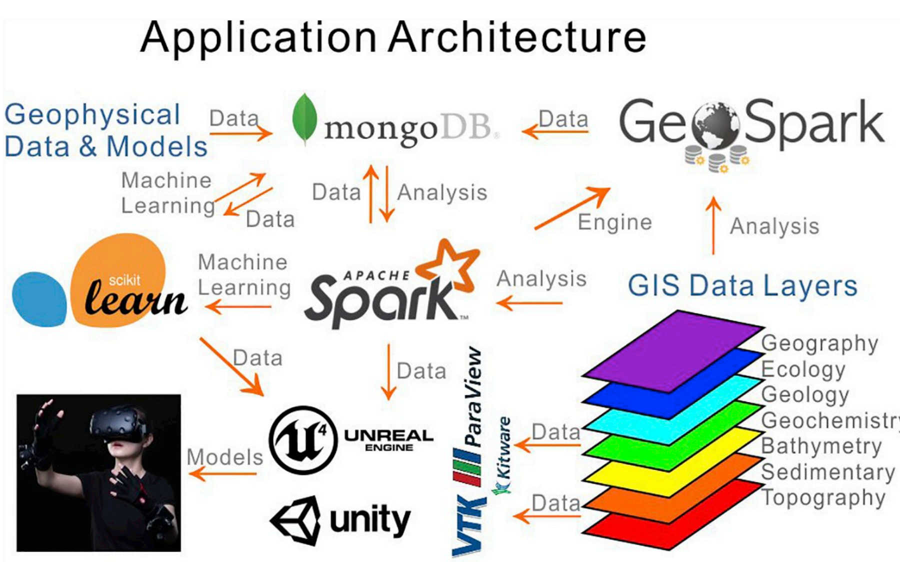

Fig. 2. Application architecture.

framework and as a turn-key solution. ParaView supports parallel computation, can run on a single-processor workstation or on a sever with distributed memory. ParaView is coded in C++, based on VTK (Visual ToolKit). Its batch processing power allows "data mining" in 3D space interactively.

3.1.6. Unity 3D

Unity 3D (https://unity.com) is a multi-platform integrated game development tool developed by Unity Technologies that allows players to easily create interactive content such as 3D video games, architectural visualization, real-time 3D animation, etc. (Creighton, 2010). It is a fully integrated professional game engine. Its editor runs on Windows and Mac OS X and can publish games to Windows, Mac, Wii, iPhone, WebGL (requires HTML5), Windows Phone 8 and Android. You can also use the Unity web player plugin to publish web games and support web browsing for Mac and Windows. Its web player is also supported by Mac widgets.

3.1.7. Unreal engine 4

Unreal Engine 4 (UE4) (Epic Games, Inc, 2018) is a new game engine that Epic Games just released. It is a follow-on version of UDK (Mittring, 2012). UE4 has some great graphics processing capabilities, including advanced dynamic lighting, a new particle system that can process millions of particles simultaneously. Until now, UE4 supports platforms such as PC, Mac, iOS, Android, Xbox One and PlayStation 4.

3.2. Hardware modules

At present, the conventional VR device set consists of the following four types: head-mounted display device; host system; tracking system; controller (Figs. 3; 4).

3.2.1. Head-mounted display device (HMD)

This is the most familiar to everyone, commonly known as virtual reality glasses. It is a kind of hardware device that is placed in front of the user to let the user see the VR effect. Such as HTC Vive (HTC Corporation, 2019) and Facebook Oculus Rift (Facebook Technologies, LLC, 2019). HMD hardware consists of display, processor, sensor, camera, wireless, storage, battery, and lens.

3.2.2. Host system

The host system refers to devices that provide various functions for HMD, such as smartphones and PCs. The host system determines the level of intelligence and automation of the HMD device.

3.2.3. Tracking system

The tracking system is generally used as a peripheral of the HMD, and of course can be integrated into the device. It typically includes built-in sensors, gyroscopes, and magnetometers. The tracking system creates an immersive experience by capturing user movements. For example, if you look up with an HMD device, the screen can be converted into sky by receiving signals from the tracking system.

3.2.4. Controller

Generally, it appears as a handheld device, through which users can track their movements and gestures. Such as Oculus Touch (Facebook Technologies, LLC, 2019), Samsung's Gear VR Rink (Samsung Electronics, 2019).

In this paper for application, we use the integrate project ALICE (see https://www.noitom.com.cn/alice/197.html) developed by Noitom Inc.

3.3. Workflow

3.3.1. Persistent storage and indexing

The MongoDB database comes with fragmentation technology, which divides the data set into multiple subsets. Each subset is stored on a fragmentation server. The Mongos routing process communicates response data release requests between the client and the data server and receives and returns the client. One only need to increase the fragmentation server when the data increases. The fragmentation cluster includes: a fragmentation server, a configuration server, and a routing server. The fragmentation server includes a data replication set. The configuration server records the fragmentation data location record, directly locates the fragment where the data is located during data query and improves retrieval efficiency; the routing server is responsible for receiving the client requests and receives the returned result. In order to quickly locate data in the database, a data indexing mechanism is created. MongoDB indexing mechanism is same as that in a relational data database, support multi-level index. We define the data type as a primary index, mainly to determine the collection of request data. For inside the collection, the ID index (the file name index) is used, and the index structure is used for rapid localization of data.

3.3.2. Analysis of GIS data and 3D models

GeoSpark performs the following steps for all spatial RDDs:

(1) Load the original data set from the data source, convert the original data set to extract spatial information and store them in a

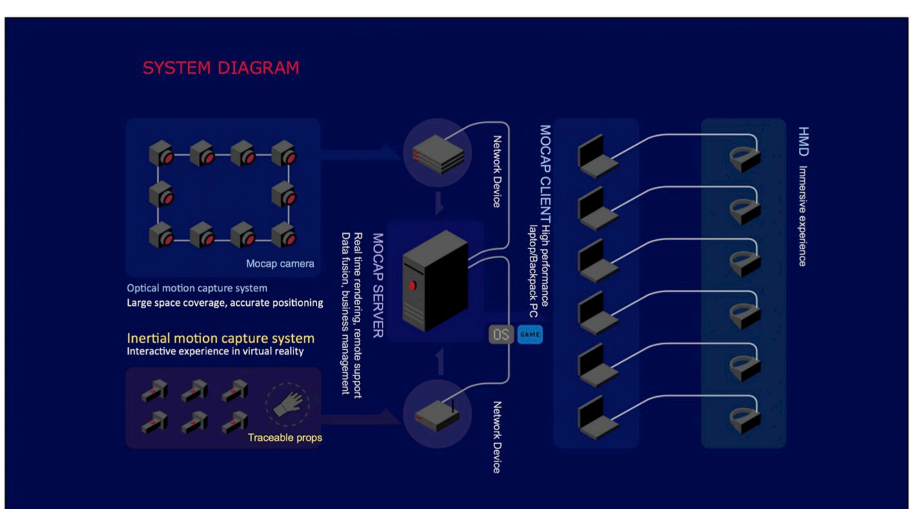

Fig. 3. VR hardware system architecture diagram.

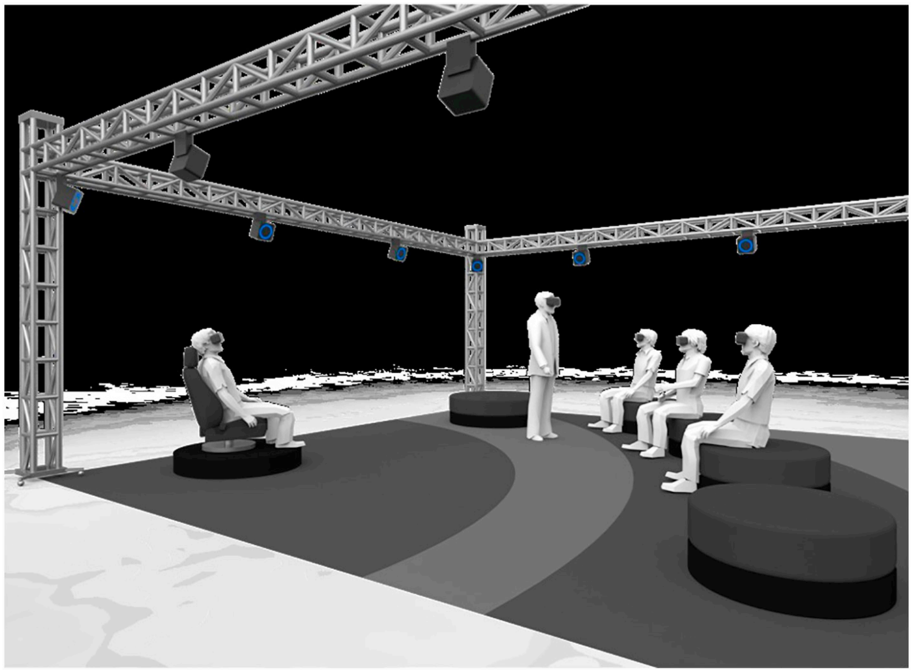

Fig. 4. interactive immersive environment concept diagram.

regular RDD. At the same time, the RDD is cached in memory for the next iteration of the job.

(2) Traverse the coordinates in the input RDDs multiple times to find their smallest bound rectangle.

(3) Calculate the grid file boundary (GLB), which is the intersection of two minimum bounding rectangles (MBR).

For 3D geological and geophysical models, first loading 3D point by using a GeoSpark customized data mapper, then change core K Nearest Neighbor (KNN) logic to make GeoSpark support 3D KNN search. For spatial colocalization (refers to the location of two or more species in adjacent relationships), Ripley's K function is often used to determine joint locations (Haase, 1995). It usually performs multiple times and forms a two-dimensional curve to observe. The calculation of the K function also requires an adjacent matrix between two types of objects. Adjacent matrices are the result of a join query. The process of finding this matrix in GeoSpark has the following steps:

(1) Call the GeoSpark point initialization method to store the two data sets in memory.

(2) Call the GeoSpark spatial connection query at one of the points. The first parameter is another point and the second parameter is the query distance. In this case, we assume that the query distance is 10 miles.

(3) Use a new instance of spatial pairing to store the result of step (2). Step (2) will return a new column with a new column specifying the adjacent elements of each tuple within 10 miles. The format is as follows: point coordinates, adjacent element 1 coordinates, adjacent element 2 coordinates.

(4) Call the persistence method in the Apache Spark layer to persist the results.

3.3.3. Analysis using machine learning

Scikit-learn can be used for spatial dimension reduction and feature learning. For example, whether it is raw data or data obtained through spatial analysis, geophysical models visualization process will face the problem of inefficiency due to too large data (such as getting stuck, which will hurt the user's immersion). Meanwhile, too many scatter points in space can make it difficult for users to understand a simple narrative. At this time, Scikit-learn and its DBSCAN (density-based spatial clustering of applications with noise) clustering algorithm (Birant and Kut, 2007) can be used to reduce the size of the spatial dataset. DBSCAN algorithm clusters spatial data sets based on two parameters: the physical distance and the minimum cluster size of each point.

3.3.4. Abstraction and representation

Data sets in geophysical research are usually very large, especially when it comes to 3D simulations. In order to handle such a large data set, the data size must be reduced in the first step (abstract): for example, by demotion sampling (space, time), selection of subsets, or selection of variables.

Create interfaces from 3D model visualization software (such as ParaView) to Unity 3D or Unreal Engine game engines for exporting generated geometry (e.g., streamlines, isosurfaces), including color scales, in the widely used Filmbox (FBX) format (Goldstone, 2011) And the value of the metadata such as the range (Bilke, 2014).

3.3.5. Interaction and presentation

In order to present visualizations in a virtual reality environment, a GUI that supports multiple displays is required. We use the Daikon Forge GUI (Daikon Forge, 2014) to implement elements of objects, time, cameras and overviews. To manage different data sets, design a hierarchy that can handle parallel display of multiple data sets with different time resolutions.

4. Visualization pipeline (Fig. 5) and application

Marine scientific research, resource survey and development, engineering construction and military activities need to accurately obtain information on seabed topography in the area of interest and use it as the basis for data and support. With the continuous improvement of deep-sea development technology, more and more research methods are used, such as manned submersibles, ROV, AUV, water gliders, deep tow systems, TV grabs and other detection equipment. The prerequisite for underwater detection of these detection equipment is to have a detailed understanding of the seabed topographic map of the detection area. It is especially important to visualize and analyze abyss data that has not been explored. Effects in VR glasses are shown in Fig. 6.

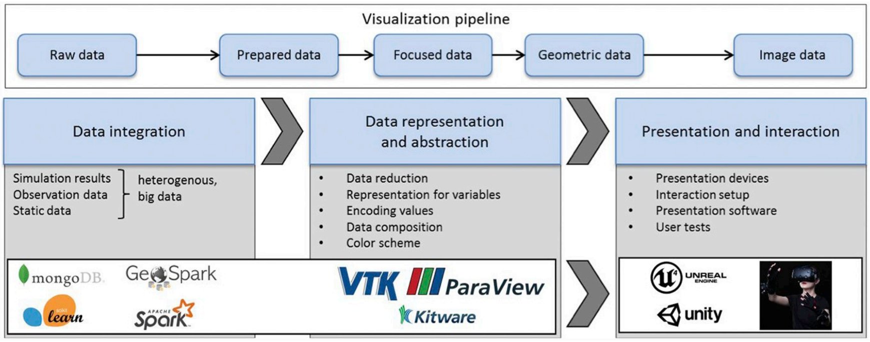

Fig. 5. Visualization pipeline.

4.1. Pre-built visual environment

A visual environment is the interface between users and information. To present a variety of geographic information, we created a virtual terrain to create a virtual world based on the real world. Unity terrain objects are powerful tools for presenting real-world environments with many features to support our visualization methods. In addition, the topographic relief of the terrain can be constructed from grayscale raw image files (RAW).

We built a global topographic model in the spherical coordinate system. In the model global geographic information consists of three-dimensional data, which contains two-dimensional coordinates (longitude and latitude) and other values on the coordinates, such as elevation, depth, and so on. Through mapping, the user's position in Unity 3D can be converted to geographic coordinates for data queries and reads. The mapping uses the Mercator projection method to define the degree of Unity units equivalent to longitude or latitude (Fig. 7). In this paper we used the abyss data of the Mariana Trench provided by the Institute of Deep-Sea Science and Engineering of the Chinese Academy of Sciences. The abyss data is stored in the GeoTagged Image File Format (GeoTIFF). GeoTIFF is converted to RAW files by QGIS project and image format converter. After importing RAW files into Unity terrain settings, one can then move over the real-world terrain by VR controller to view the geographic information, and the user's location returns the geographic coordinates through the mapping (Fig. 7).

4.2. Interaction with data

The application's user interface is used to interact directly with the virtual Earth and display parameter dialogs. In the desktop environment, the parameter dialog is displayed as a regular 2D window with the mouse cursor as the interactive device. However, in a VR environment, dialogs are projected into a plane that can be placed anywhere in the virtual space (Fig. 8).

For interaction, people can use the controller to control the virtual pick-up ray. The so-called pick-up ray emitted from the controller is used to point and manipulate the dialog elements, display data attributes and generate analytical data results by clicking or drawing lines (Figs. 9 and 10).

By implementing separate navigation and analysis modes, the user can switch using the buttons on the controller. For navigation, a trackball and free flight control are provided. On the planetary scale, a point on the planet is captured by the trackball and rotated to a new position, allowing quick navigation to the location of interest. On the other hand, free flight control is an immersive navigation method that can be used to check most linear features, seamounts and sea valleys. The direction of travel is determined by the direction in which the rays are picked up. With the pick-up ray, you can draw the curve directly onto the surface.

The point-and-click operation is the simplest to give the value of the position of the ray (Fig. 9). In Unity 3D, the normal way to do this is to use get requests through the WWW class. As the user moves over the terrain, the application asks the user's location and maps the location to geographic coordinates after a period of movement (about 0.2 Unity units). The WWW URL adds a get request with parameters, i.e. geographic coordinates, and assigns it to the WWW object. Then, we hand the WWW object to a coordinator waiting for the completion of the www request. In an appropriate network, the delay from starting the

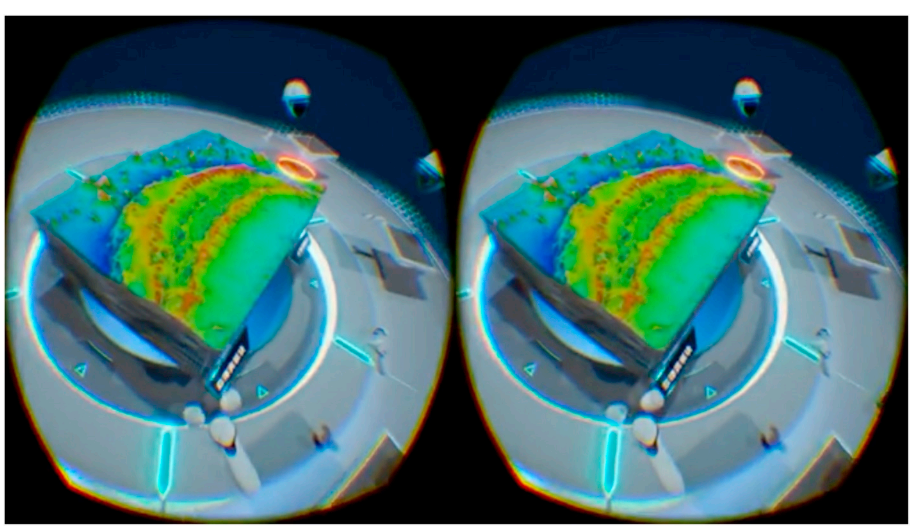

Fig. 6. VR perspective in a HMD.

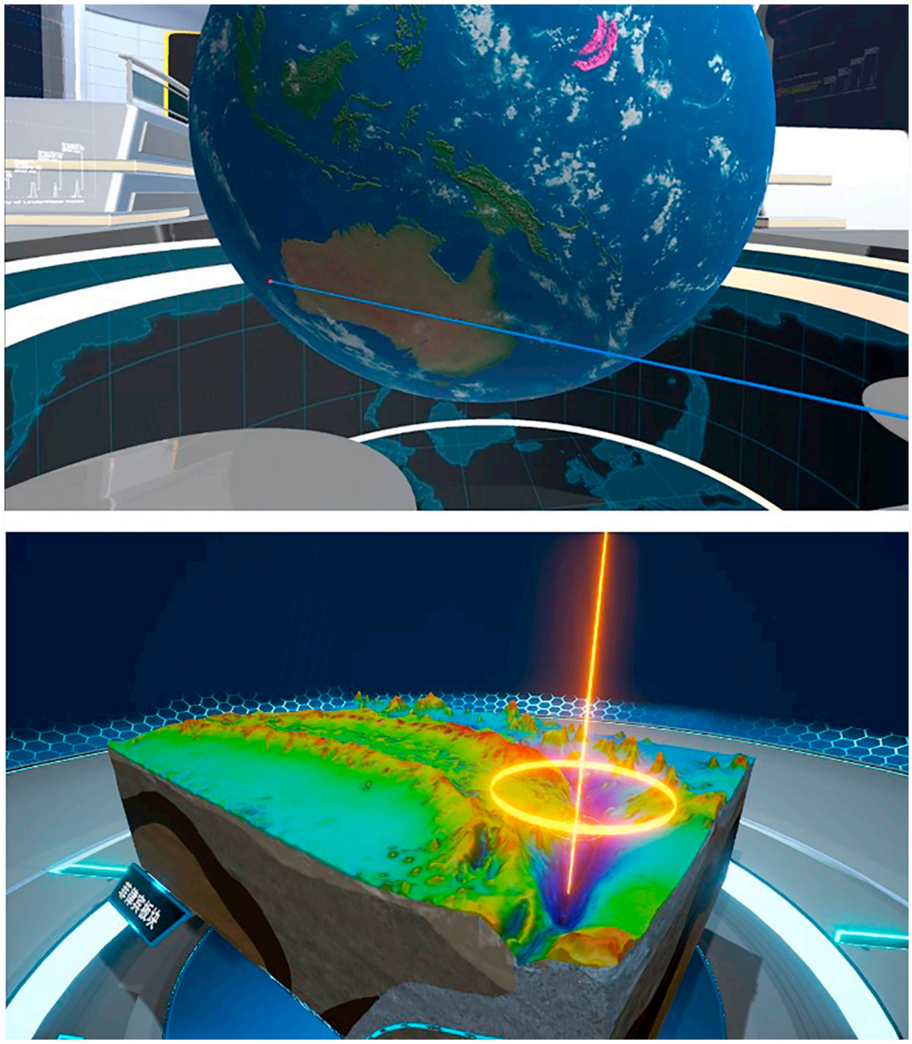

Fig. 7. Terrain data visualization in spherical coordinates.

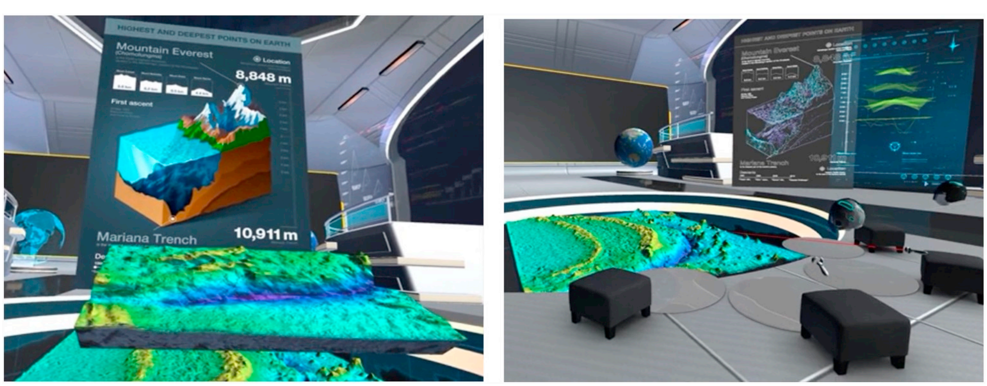

Fig. 8. Rich display of data information of Mariana Trench.

coroutine to returning data is about 0.45 s, which is almost real-time. The actual behavior of the selected object is a discrete event that occurs when the user presses a button. To model this, the button's continuous status report is converted to a discrete event. This is done using a conditional filter. In this case, the standard is the state of the button that must be "pressed", defined syntactically by symbolic constants.

To generate the depth profile, the water depth data can be sampled equidistantly along the line segment between two given geographic points. Although the endpoints of the depth profile are specified in geographic coordinates, spherical interpolation uses three-dimensional Euclidean coordinates.

5. Discussion and conclusion

5.1. Discussion

5.1.1. High-efficiency

The novelty of our approach is to provide a flexible workflow that

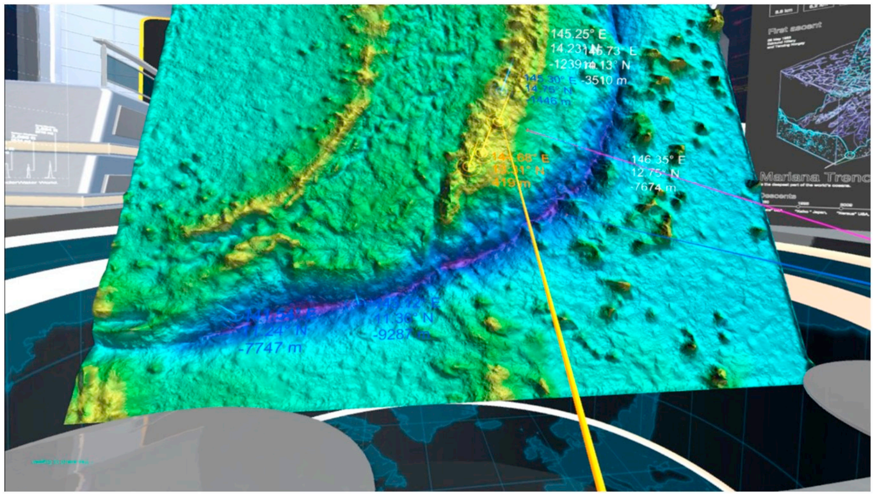

Fig. 9. Digital Labels - instantly and accurately query latitude and longitude and depth.

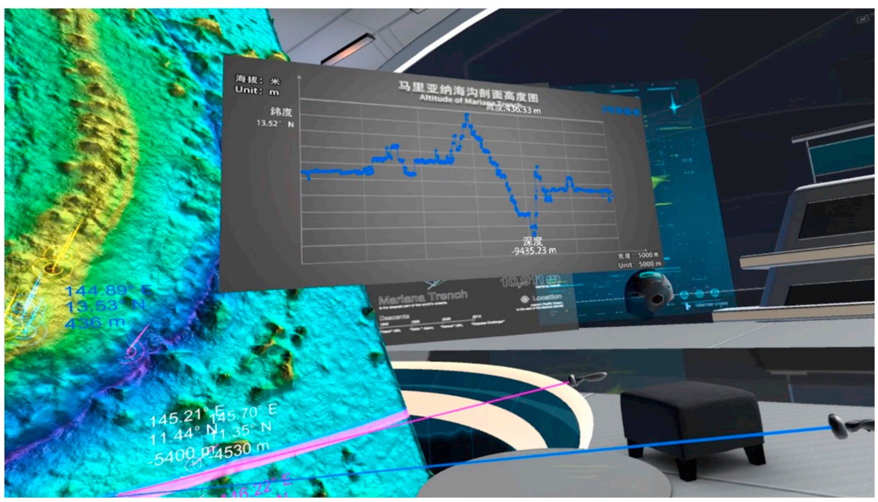

Fig. 10. Abyss data analysis - real-time collection of trench cross-section data.

consists of software components and supports researchers in different disciplines to generate meaningful visualizations. Our goal is not to develop new algorithms and techniques, but to provide a toolbox that combines these prior art techniques in a useful way. By using the generic parallel computing environment Spark and the visualization system ParaView as the initial part of our workflow, the framework can be easily applied to other scientific fields.

5.1.2. Ease-of-use

The importance of visualization in many scientific fields is increasing due to the increase in complex large data sets. As flexible workflows evolve, the foundation for generating visualizations for other application domains is provided. This computational VR visualization framework can be used on different devices. Thus, taking advantage of the framework it can save your development efforts. When enhanced by software and codes described in this paper, new and modified data types can even be easily processed for your own application.

5.2. Conclusion

In this paper, we present a computational interactive virtual reality visualization framework and workflow for geophysical models for exploring, analyzing, and presenting geoscience data from multiple sources and including numerous variables. To provide user interaction, we designed and implemented a modular and reusable interface for the game engines Unity 3D and Unreal Engine 4.

After the realization of data visualization in virtual reality, there are still some obstacles to overcome. The visual resolution needs to be increased, so the image is clear and easy to read, and dizziness and nausea are still a problem for some people. One of the key challenges is not just to build VR data visualizations - we need to design useful visualizations that take advantage of VR and provide an intuitive way to interact, analyze and manipulate data. As hardware evolves and our understanding of this new technology continues to improve every day, we believe that effective visualization of data in virtual reality is only a matter of time.

Acknowledgement

We thank the Institute of Deep-Sea Science and Engineering of the Chinese Academy of Sciences for providing abyss data of the Mariana Trench for visualization. We thank Beijing Noitom Technology Ltd., who provided support of fully immersive multi-user VR technologies. This research is supported by the National Key Research and Development Program of the Ministry of Science and Technology of China with the Project "2018YFC0603500 (2018~2020), 2016YFC0600310 (2016~2020), 2018YFC0308302 (2018~2020) and 2018YFC0308301 (2018~2021)". We thank Prof. Vernon Cormier (the

Editor) and two anonymous reviewers for their constructive comments and suggestions, which have improved our manuscript.

References

Benko H, Ishak E W, Feiner S, et al. Collaborative mixed reality visualization of an archaeological excavation[C]. International Symposium on Mixed and Augmented Reality, 2004: 132-140.
Bilke L (2014). VtkFbxConverter. https://github.com/ufz-vislab/. (Accessed 29 July 2014).
Birant, D., Kut, A., 2007. ST-DBSCAN: an algorithm for clustering spatial–temporal data [J]. Data Knowl. Eng. 60 (1), 208–221.
Borthakur, D., 2008. HDFS Architecture Guide[J].
Buitinck L, Louppe G, Blondel M, et al. API design for machine learning software: experiences from the scikit-learn project[J]. arXiv preprint arXiv:1309.0238, 2013.
Chodorow, K., Dirolf, M., 2010. MongoDB: The Definitive Guide[M]. O'Reilly Media, Inc.
Choi Y, Cheok A D, Halupka V, et al. Flavor visualization: Taste guidance in co-cooking system for coexistence[C]. International Symposium on Mixed and Augmented Reality, 2010: 53–60.
Creighton, R.H., 2010. Unity 3D game development by example: a seat-of-your-pants manual for building fun, groovy little games quickly[M]. Packt Publishing Ltd, 1-346.
Daikon Forge (2014). Daikon forge gui library. www.daikonforge.com/. .
Epic Games, Inc. (Version 4) [Software] Unreal Engine, Inc. Available from: https://www.unrealengine.com. 2018.
Facebook Technologies, LLC. [Hardware] Facebook Oculus Rift, https://www.oculus.com/us/. 2019.
Goldstone W (2011) Unity 3.x game development essentials. Birmingham: Packt Publishing, 2 edition, 1–488 pp.
Haase, P., 1995. Spatial pattern analysis in ecology based on Ripley's K-function: introduction and methods of edge correction[J]. J. Veg. Sci. 6 (4), 575–582.
HTC Corporation, [Hardware] HTC Vive, https://www.vive.com/us/. 2019.
Jones, E., Oliphant, T., Peterson, P., 2014. SciPy: Open Source Scientific Tools for Python[J].
Mittring, M., 2012. The technology behind the unreal engine 4 elemental demo[J]. In: Part of "Advances in Real-Time Rendering in 3D Graphics and Games," SIGGRAPH.
MongoDB, Inc., MongoDB. (Version 4.0) [Software] Available from: https://www.mongodb.com/what-is-mongodb. 2019.
Oliphant, T., 2006. Guide to NumPy[J].
Pedregosa, F., Varoquaux, G., Gramfort, A., et al., 2011. Scikit-learn: machine learning in Python[J]. J. Mach. Learn. Res. 2825–2830.
Samsung Electronics. [Hardware] Gear VR, https://www.samsung.com/us/support/owners/product/gear-vr-with-controller. [Accessed 17th August, 2019].
Seward. 2014 Virtual Reality Headset Oculus-rift meets the Bloomberg Terminal [Online] Available from: http://qz.com/218129/virtual-reality-headset-oculus-rift-meets-the-bloomberg-terminal/ [Accessed 5th July 2019].
Smichrissoft. Why VR Data Visualization Could Be Your Secret Weapon in 2018, [Online] Available from: https://medium.com/@smichrissoft/why-vr-data-visualization-could-be-your-secret-weapon-in-2018-8eed313e5965 [Accessed 5th Nov. 2018].
The Apache Software Foundation, 2018. Apache Spark. (Version 2.2.2) [Software]. Available from: http://spark.apache.org/news/spark-2-2-2-released.html.
Michael D. Thomas. 2014. Using virtual reality to understand big data. URL: http://www.sas.com/ro_ro/news/sascom/2014q1/virtual-reality-big-data.html. Accessed in August 2016.
Thusoo, A., Sarma, J.S., Jain, N., et al., 2009. Hive: a warehousing solution over a map-reduce framework[J]. Proceedings of the Vldb Endowment 2 (2), 1626–1629.
Washburn, D., Jones, L.M., 2004. Could olfactory displays improve data visualization[J]. Computing in Science and. Engineering 6 (6), 80–83.
Yu J, Wu J, Sarwat M. GeoSpark: a cluster computing framework for processing large-scale spatial data[C]//Sigspatial International Conference on Advances in Geographic Information Systems. ACM, 2015:1–4.
Zaharia, M., Xin, R.S., Wendell, P., et al., 2016. Apache spark: a unified engine for big data processing[J]. Commun. ACM 59 (11), 56–65.