

Immersive isosurface visualisation for engineering datasets

Kadek Ananta Satriadi*, Kingsley Stephens*, Callum Atkinson†, Maxime Cordeil*, and Tobias Czauderna*

*Faculty of Information Technology
Monash University
Melbourne, Australia

Email: kasat2@student.monash.edu, {kingsley.stephens,
max.cordeil, tobias.czauderna}@monash.edu

†Laboratory for Turbulence Research in Aerospace and Combustion,
Department of Mechanical and Aerospace Engineering,

Monash University
Melbourne, Australia

Email: callum.atkinson@monash.edu

Abstract—The visualisation of isosurfaces is an important step towards the understanding of many physical phenomena. Physical simulations produce large amounts of quantitative numerical information which need to be interpreted by domain experts. Traditionally, researchers use standard tools to create isosurface visualisations from fluid simulations and extract knowledge. With the recent evolution of display technology, it is now possible to create *immersive* visualisations of such data. Benefits include a higher bandwidth of communication between the data and the user, e.g., by using stereoscopic display technology and head tracking. Since the data is intrinsically 3D spatial, the use of the immersive environment supports natural visualisation, exploration, and analysis of isosurfaces. In this paper, we present the study and the design of a method and a workflow to display isosurfaces in heterogeneous immersive environments. In particular, we address the problem of scale and structure inherent to this type of data. We provide a set of tools to process large volume datasets to eventually display it in 3D immersive environments. We demonstrate the workflow of tools with a usage scenario that involves a dataset from direct numerical simulation of a wall-bounded turbulent flow.

I. INTRODUCTION

Scientific experiments and simulations produce large amounts of data that need to be presented and analysed through visualisation. Data visualisation helps researchers to visually explore and understand clusters, trends and other forms of emerging information lying in the data. In this paper, we set our focus on the visualisation of three-dimensional scalar field data, which is one of the data types that are commonly used in the scientific and engineering domain, such as fluid dynamics simulation and three-dimensional experimental measurements [1], [2].

One of the ways to visualise three-dimensional scalar fields is by isosurface extraction and visualisation. Scientists commonly use visualisation software such as Paraview¹ and ImageVis3D² to perform isosurface visualisation. Paraview and ImageVis3D offer various data analytics and 3D visualisation tools that are very useful for visualising isosurfaces. Isosurfaces extracted from an engineering dataset created by direct numerical simulation (DNS) of turbulent flows contain

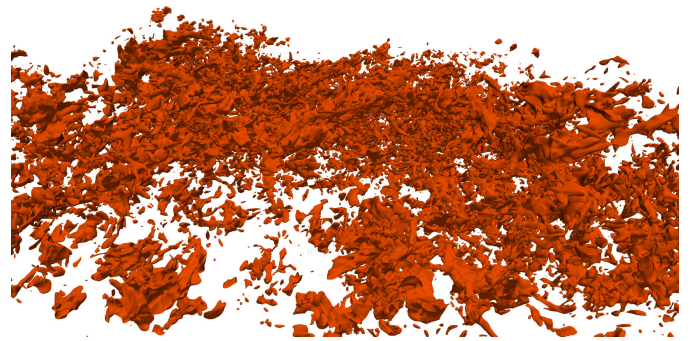


Fig. 1. Visualisation of isosurfaces of coherent flow structures from direct numerical simulation of a turbulent boundary layer [3], created in Paraview.

a significant number of three-dimensional multiscale coherent flow structures (see Fig. 1). Exploring such complex and dense structures on a conventional 2D display can be a challenging task. For instance, it is hard to maintain the orientation as the user navigates the visualisation due to isosurface structures' density and the 2D screen's limited field of view. Furthermore, traditional 2D screens present serious perception and cognition limitations to visualise and interact with 3D data.

The progression of virtual reality technology has led to a new way of visualising data. Virtual reality environments provide an immersive experience which could help users in exploring large datasets. Recently, a new research thrust called *Immersive Analytics* [4] has emerged and aims at studying how *immersion* can support reasoning and decision making through new display and interaction technology.

Currently, there is a wide range of immersive displays available for consumers. Head-mounted displays such as the Oculus Rift and the HTC Vive provide a virtual reality experience for single users for a small budget. For a larger audience more expensive display systems such as the CAVE2 [5] is a potential solution for collaborative data visualisation and analysis, and research in this domain is active [6], [7]. Mobile VR has also become more interactive. With Google Daydream³ or Samsung

¹<https://www.paraview.org/>

²<https://www.sci.utah.edu/software/imagevis3d.html>

³<https://vr.google.com/daydream/>

Gear VR⁴, users are now able to interact with objects in the virtual environment by using a remote controller. Despite the high variety of immersive visualisation displays, bringing scientific visualisations into virtual reality is still a challenging task. A lack of virtual reality applications that are designed for data visualisation makes the process of, e. g., extracting and visualising isosurfaces in a virtual reality environment rather complex.

Nowadays, visualisation software such as Paraview provide a feature to transfer the visualisation to a virtual reality display. However, this feature can only be activated with a custom build of Paraview from source code. Moreover, at the current stage, the lack of interaction in the virtual environment is also a significant disadvantage. For example, when dealing with large and complex isosurface structures, the extent of the interactivity might go beyond basic six degrees of freedom (6 DoF) navigation. Researchers might need to perform additional tasks such as measuring the volume of a structure of interest. In addition to interactivity, an efficient workflow to extract volume data from the dataset before entering the visualisation phase is also an important aspect.

This paper aims to address several challenges for scientific visualisation in virtual reality environments. The objectives of this research cover several points as follows:

- 1) creating an effective way to extract isosurfaces from scalar field datasets which cannot be imported directly into off-the-shelf visualisation tools or virtual reality software frameworks,
- 2) investigating to what extent the extraction process can be performed interactively in terms of constructing isosurfaces from the input datasets during run-time,
- 3) presenting interactive isosurface visualisations on virtual reality displays, and
- 4) developing an efficient workflow for extracting information about individual structures.

The paper is organised as follows. Section II describes an example use case, followed by related work regarding existing solutions for scientific visualisation in virtual reality environments in Section III. Section IV discusses a proposed workflow for isosurface extraction and visualisation using immersive environments, Section V explains the implementation in detail and is followed by concluding remarks in Section VI.

II. USE CASE

Datasets from a direct numerical simulation (DNS) have been chosen as exemplary use case to demonstrate the implementation of our approach. A direct numerical simulation is performed in the fields of fluid mechanics and thermofluids to provide a high fidelity numerical solution to the non-linear partial differential Navier–Stokes equations which govern the motion of liquids and gases.

A. Datasets

The datasets used in this research were acquired from a direct numerical simulation undertaken in the Laboratory

⁴<http://www.samsung.com/global/galaxy/gear-vr/>

TABLE I
USE CASE DATASETS

Name	Dimensions [px]	Filesize
Dataset1.u	384 x 385 x 384	440MB
Dataset1.v	384 x 385 x 384	440MB
Dataset1.w	384 x 385 x 384	440MB
Dataset2	2048 x 400 x 1001	3.2GB
Dataset3	2048 x 500 x 1001	8GB

for Turbulence Research in Aerospace and Combustion at Monash University, Australia. The scalar field data is stored in HDF5 [8] files. The datasets are the result of a large-scale computation that can extend to 30 million CPU hours with a grid size of over 11 billion grid points [3]. The size of the files generated from the simulation can reach as much as 42GB for a single velocity component. Smaller scalar fields will be used as a proof of concept in this study. The dimensions of the scalar field and the size of each dataset used in this study can be seen in Table I.

Each of the datasets contains a scalar field which is compiled of several variables: scalar values, x-grids, y-grids, and z-grids. The scalar points are structured in a rectilinear grid. The first three datasets in Table I share the same volume grids with identical geometry but contain a different scalar value representing the three velocity components of the fluid. Each of the scalar values in the datasets represents the magnitudes of three-dimensional vectors in u , v , and w direction respectively.

To give more insight about the technical details of the datasets, it is worth discussing briefly the HDF5 file format. HDF5 is defined as a data model, library, and file format for storing and managing data [9]. It is a technology suite which not only provides a data storage model but also a library to maintain and develop HDF5 files in various programming languages. The HDF5 file is designed to be a portable, extensible and efficient data management model. HDF5 also supports parallel I/O which is suitable for distributed computing.

When it comes to visualisation, the HDF5 file model does not provide a straightforward method. To visualise the content of a HDF5 file, e. g., Paraview can be used by creating an XDMF⁵ file describing the data structure and data types of the HDF5 file. The topology and geometry of the HDF5 file must be specified explicitly in the XDMF file so that Paraview can read the data contained in the HDF5 file. Hereby, the XDMF approach often becomes a bottleneck in the process because it requires users to have knowledge about the XDMF concepts and syntax.

B. Design requirements

The following design requirements for the visualisation system have been identified based on the feedback from an expert in the fluid simulation domain:

- 1) The visualisation system has to be able to load relatively large isosurfaces (more than 2,000,000 points).
- 2) The visualisation system has to be interactive providing:

⁵<http://www.xdmf.org/index.php/>

- a) navigation in the visualisation,
 - b) means to select and measure the volume of the bounding box of isosurface structures,
 - c) storage of reference points including camera view and orientation for review, and
 - d) isolation of coherent structures for a more focused view.
- 3) The visualisation system has to be usable in different immersive environments such as head-mounted displays (HMDs), CAVEs, and 3D stereoscopic screens.

Requirement 2) are navigation and selection tasks. While these are standard visual analytics tasks, very few guidelines exist yet to support them in heterogeneous immersive environments.

C. Visualisation requirements

Currently, no visualisation has been implemented in a virtual reality environment for the used datasets (see Table I). The visualisation tasks have been conducted so far using Paraview using a conventional 2D computer screen. The process involves the manual creation of XDMF files. The workflow of data visualisation starts by extracting the isosurfaces based on different isovalues. The isosurface extraction creates a 3D object that has separated small structures representing regions with coherent flow value (see Fig. 1). The exploration process is performed by, e. g., locating and measuring the boundary dimensions of these structures.

Another task is to illustrate the relationship between two datasets by colouring the isosurface generated from one dataset with the scalar values from another datasets by colour mapping. The volume data with identical grids such as Dataset1.u, Dataset1.v, and Dataset1.w (see Table I) can be used for this particular task. As long as the two datasets share a common grid, any combination of isosurface and colour mapping is possible.

The visualisation of large datasets such as Dataset2 and Dataset3 could produce a significant number of regions (see Fig. 1). When the user zooms into a particular region, the surrounding regions are not visible due to limited computer screen field of view. This condition becomes a concern during the exploration of large isosurfaces where features of interest may be occluded in the "forest" of isosurfaces.

III. RELATED WORK

The visualisation of scientific data in virtual reality environments is a relatively dynamic area. Several projects have been conducted on this topic focusing on different research problems. There is some literature available about successful application of virtual reality techniques for scientific visualisation, e. g., some research has proven that the virtual reality approach can enhance collaborative data exploration.

Dos Santos et al. [10] integrated virtual reality in a large-scale engineering project where users can investigate engineering data. The virtual reality approach used in the study showed a significant improvement in managing a large project. Other examples for collaboration in virtual reality environments for

scientific visualisations are [11], [12], and [13]. Applying the virtual reality approach, collaboration can involve participants from remote locations. One of the examples of remote collaboration in virtual reality research is [14]. The distributed virtual reality environment made it possible for the participants to meet in the virtual world and perform data exploration without being present in the same physical location. Virtual reality can also be used to virtually send a scientist to a remote research location. Research by Bednarz et al. [15] applied virtual reality techniques to perform a jaguar conservation study by enabling the scientist to see and feel the remote location. But the application of virtual reality techniques is not only useful for data exploration. It has also been used as a research communication media. The virtual skin exploration tools by Vazquez-Duchêne et al. [16] showed the accomplishment of the virtual reality approach in changing the perspective of people to the skin structure.

Despite a number of publications available about virtual reality applications for scientific visualisation, not many of them discuss specifically a visualisation pipeline or workflow. The development of virtual reality application is rather task-specific and time-consuming. It often also requires an advanced graphics programming knowledge to be able to visualise data in the virtual reality environment. Without visualisation tools, bringing visualisation into virtual reality environments can be a challenging and time-consuming process. Kageyama et al. [17] tried to address this issue by developing a vector field visualisation software for virtual reality environments. The focus of their research was on the development of a generic vector field virtual reality application. The result showed that the visualisation software worked well in CompleXcope [18], a CAVE-like display system. However, it did not demonstrate any capabilities to work in head-mounted displays. Another visualisation system called COVISE was able to perform volume rendering visualisation in a virtual environment that works on CAVE-like systems with various user interactions [19]. However, the issue of occlusion for dense isosurface structures from large volume data has not been addressed.

The most recent research in visualisation workflows for virtual reality was looking at a way to establish a workflow that can be performed in an office set-up [20]. The visualisation pipeline was built on top of Paraview and EnSight⁶. A Paraview plug-in was developed to enable the visualisation to be displayed on the zSpace⁷ with full VR interaction which is supported by either the Virtual Reality Peripheral Network (VRPN)⁸ for Windows or the Vrui VR Toolkit⁹ for Linux and MacOS. Although the workflow worked well within an office set-up with a relatively low cost of investment, it still missed some important aspects. The framework has been demonstrated to work on semi-immersive displays such as 3D stereoscopic screens. However, the extension to the head-mounted display was not covered. Consequently, the capability

⁶<https://www.ensight.com/>

⁷<https://zspace.com/>

⁸<https://github.com/vrpn/vrpn/wiki>

⁹<http://idav.ucdavis.edu/~okreyls/ResDev/Vrui/>

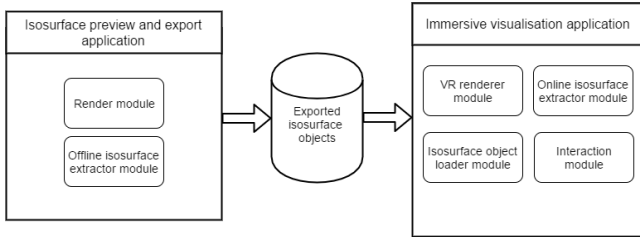


Fig. 2. Architecture of the visualisation tools.

of handling large isosurfaces was not clearly addressed and the visualisation is still limited by the size of the screen used.

Some previous research utilised game engines for visualisation and rendering. Nowadays, one of the industries where computer graphic capabilities are pushed to the limit is the gaming industry. Many game engines are equipped with latest computer graphics technology. Unity¹⁰ is one of the game engines that have been used in visualisation research, e. g., Marks et al. [21] conducted research on CAD data visualisation in a VR system by utilising the Unity game engine.

CAD models were transformed into the FBX format and loaded into Unity. The pre-processing stage was done using Blender¹¹ to reduce the complexity of the 3D models. The result shows that the system is robust enough to visualise CAD data in VR. However, the necessity to reduce the complexity of the models using Blender as well as missing interaction beyond six degrees of freedom (6 DoF) navigation, e. g., interactive isosurface extraction, are shortcomings of this approach.

IV. WORKFLOW

In Section II several design requirements have been identified to transform isosurface data contained in files into visual artefacts in heterogeneous immersive environments. To achieve this, a non-linear workflow is proposed that involves several data transformations and mappings.

A. Architecture

The architecture of the visualisation tools is divided into two primary applications (see Fig. 2) to handle two different isosurface extraction approaches (to be discussed below). The first application, *Isosurface preview and export*, provides an interactive isosurface visualisation on a conventional 2D display. This application also handles isosurface object export to storage such as hard disk. The other application is the *Immersive visualisation* application. Using this application a user can interactively explore and analyse isosurfaces visualisations in virtual reality environments.

B. Data transformation

The data transformation stage is responsible for transforming the raw datasets into a visualisation-ready format. The datasets from the use case (see Section II) contain volume data

¹⁰<https://unity3d.com>

¹¹<https://www.blender.org/>

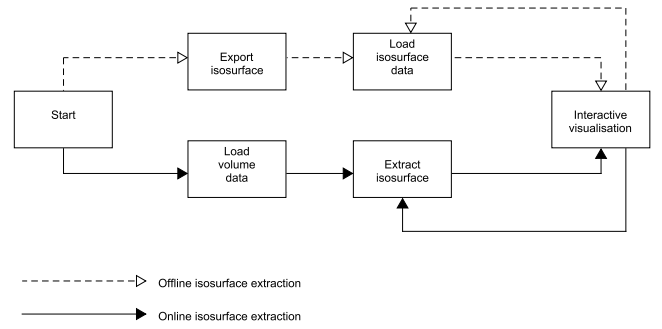


Fig. 3. Offline and online isosurface extraction.

which is stored in HDF5 files. Some other data, not required for the visualisation of isosurfaces, might also be stored in the same HDF5 files which contributes to an increased filesize. Moreover, as mentioned in Section II, performing the visualisation based on HDF5 files is rather complicated. Therefore, a data transformation stage is proposed where the elements (grids and scalar values) of the volume data is extracted from the HDF5 files and stored in a visualisation-ready format.

C. Visualisation entry point

Two workflows have been established reflecting the size of the datasets a user deals with. Due to technical choices, scale limitations exist that involve pre-specification of the isovalues to construct the isosurfaces *offline*. An *online* workflow has been provided which allows interactive construction of isosurfaces in the immersive visualisation application during run-time.

1) *Offline isosurface extraction*: This approach separates the isosurface extraction and isosurface visualisation/exploration (see Fig. 3). A large isosurface with millions of points might take a long time to be extracted. Consequently, extracting a large isosurface in the virtual reality environment is not desirable. In this case, the isosurfaces objects are exported to storage and loaded into the virtual reality environment when needed. This approach not only reduces the waiting time in the virtual reality environment but also improves the reusability and portability of the isosurfaces of interest.

2) *Online isosurface extraction*: The online isosurface extraction is opposite to the offline approach. In this approach, the isosurface extraction is performed in the virtual reality environment (see Fig. 3). However, the size of the volume data and the size of the resulting isosurface needs to be considered carefully to minimise the waiting time.

D. User interaction

1) *Navigation*: Navigation is supported by six degrees of freedom (6 DoF) transformation of the isosurface and markers revisit. The 6 DoF transformation is a basic navigation where the user can rotate and pan the isosurface object in the virtual reality environment. During exploration of a large structure,

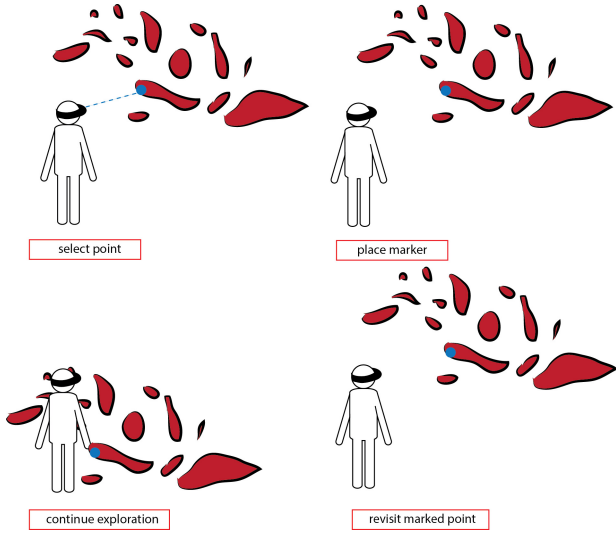


Fig. 4. Illustration of the placement of markers on structures of interest and revisit. During the exploration process, a user can place markers on structures. On request the application automatically navigates to a marked structure.

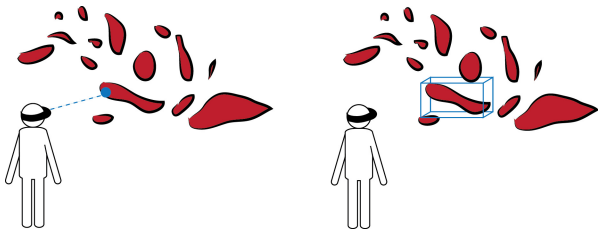


Fig. 5. Isosurface structure selection using the 3D pointer (left), isosurface bounding box measurement (right).

a user might want to store the location of the structures of interest. To accommodate this requirement, a waypoint mechanism is proposed which allows a user to place markers on the isosurface structures (see Fig. 4). Since a mouse pointer is not available in the virtual reality world, a 3D pointer is used to select a particular isosurface for marking. During the exploration, a user can return to a marked structure without manually moving or rotating the isosurface object using 6 DoF navigation.

2) *Selection and analytics*: For analytical purposes a bounding box for measurement of the structure of interest is proposed (see Fig. 5). A bounding box of a structure has annotations that indicate the dimensions of the box and the coordinates of each box point. The dimensions and coordinates of a bounding box can be exported to be used outside the visualisation application. A screen shot of the structure is also provided as a reference to the exported bounding box data.

When it comes to examination of dense structures, the bounding box of the structure of interest might be occluded by other structures. To overcome this issue, an isosurface isolation mechanism is proposed to help a user in maintaining the focus on the important structure. When a structure is isolated, the rest of the isosurface structures are hidden (see Fig. 6).

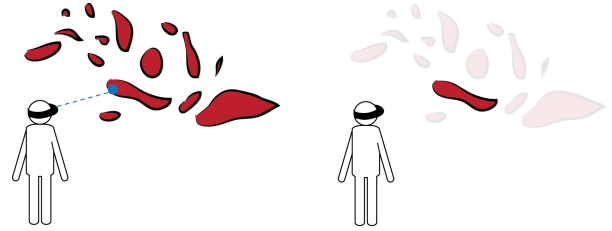


Fig. 6. Isosurface structure isolation.

V. IMPLEMENTATION

In this section the implementation of the proposed workflow is presented. The workflow is divided into two stages: the pre-processing stage and the visualisation stage. The pre-processing stage converts HDF5 files data into a visualisation-ready format. Then in the visualisation stage the isosurface extraction and visualisation is carried out.

A. Pre-processing the data

As the entry point to the visualisation stage the VTK [22] library's volume data format is used as input format. By using this format the volume data from HDF5 files can be visualised directly using off-the-shelf visualisation tools without involving the creation of XDMF files. The datasets can be loaded and manipulated in the visualisation tool which is discussed in the next section. To accommodate the conversion process a Java-based pre-processing tool has been developed that can be used to export volume data stored in HDF5 files into VTR files (file type to store VTK's `vtkRectilinearGrid` object). For large scalar fields (e. g., Dataset3), the VTR files are stored as partitions due to limitations of the VTK library's VTR file reader.

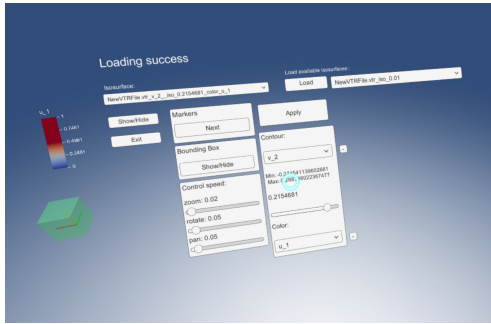
B. Visualisation

Unity has been used to build the visualisation application for several reasons:

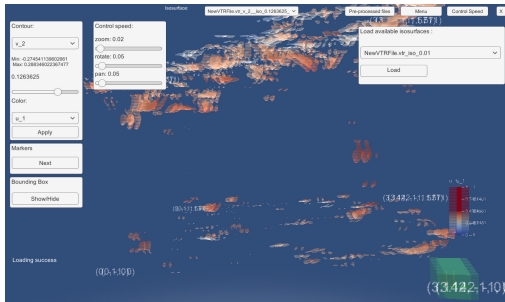
- 1) it has a good portability, Unity can build the application for almost all platforms and operating systems,
- 2) switching between rendering mode (virtual reality, stereoscopic 3D, or standard 2D) is done simply by changing the build options, and
- 3) all the virtual reality required system features such as stereoscopic rendering, graphical user interface, and interaction system are provided by the Unity game engine.

However, Unity limits the maximum number of vertices for a single mesh object to 65,535 which requires splitting of large isosurface structures into smaller parts.

The implementation of the offline isosurface extraction approach lets a user find the isosurface of interest by extracting isosurfaces with different isovalues prior to the visualisation phase. Once the desired isovalue or isosurface is found the isosurface mesh data is exported to OBJ files and an additional file is created containing metadata of the isosurface such as isovalue, scalar field dimensions, and scalar value ranges. A Java application has been developed based on the VTK



(a) 3D space GUI in virtual reality.



(b) 2D GUI on a 3D stereoscopic screen.

Fig. 7. The graphical user interface of the Unity visualisation application in virtual reality and 3D stereoscopic mode.

library for the isosurface export. The user can load up a single or multiple (partitions) VTR files, preview the isosurfaces, and then export OBJ files. The resulting isosurfaces of an extraction and export from Dataset2 can be seen in Fig. 10. The offline extraction of a large isosurface might take hours to finish. However, loading the isosurface object into the virtual reality scene is considerably fast.

The implementation of the online isosurface extraction approach provides the user with a graphical user interface (GUI) in the virtual reality scene. At the beginning of the visualisation process volume data (VTR file) is loaded into the application. Once the volume data is loaded the metadata of the volume is shown in the GUI. The GUI consists of two drop-down components where the user can specify the scalar field array to be used for the isosurface visualisation and the colour mapping. To specify the isovalue the user can use a slider. The scalar field range is also indicated on the GUI. To increase the portability of the visualisation application for heterogeneous immersive displays, the GUI has been implemented in two different rendering modes: 2D and 3D (see Fig. 7). In a fully immersive display such as a head-mounted display the GUI is shown as a 3D object whereas for semi-immersive displays such as a 3D stereoscopic screen the GUI elements are presented as 2D objects.

The online isosurface extraction is performed asynchronously in the background to minimise the lagging or view freezing due to a long running process. Figure 8 shows the isosurface of Dataset1.v which is coloured by the scalar values in Dataset1.u. The isosurface was extracted using the

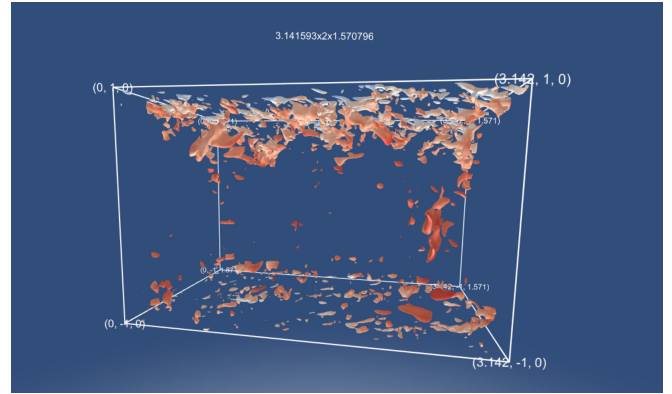


Fig. 8. The isosurface visualisation of Dataset1.v with colour mapping from Dataset1.u.

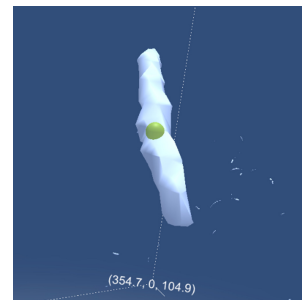


Fig. 9. The 3D pointer (green sphere) in 3D space.

online extraction approach. The online isosurface extraction works with less than 10 minutes waiting time when dealing with isosurfaces with up to 1 million points. As the number of points increases the waiting time can reach more than 10 minutes (e.g., it takes around 11 minutes to extract an isosurface with 1.1 million points).

C. Interactions

The first key interaction element besides the 6 DoF navigation is the 3D pointer. Similar to the pointer in 2D space, the 3D pointer is used to point to a particular object in the 3D space. In this particular case, the 3D pointer can be used to select a structure of interest on which an action should be applied. Figure 9 shows the 3D pointer which is represented as a green sphere. The pointer works by performing ray tracing in the scene and using the user's eyes centre point as the source point. When the ray collides with the isosurface structure the green sphere is shown at the collision point. The user is provided with a menu containing different actions such as *Show bounding box*, *Isolate structure*, and *Add a marker*.

The other user interaction elements in the scene are bounding box measurement, structure isolation and markers placement. To determine the bounding box of each isosurface structure closed isosurface structures or regions have been separated using the VTK library's `vtkConnectivityFilter` during the isosurface extraction process. Figure 10 shows the isosurface extracted from Dataset2. It can be seen from the figure that the surrounding structures occlude the bounding box of the

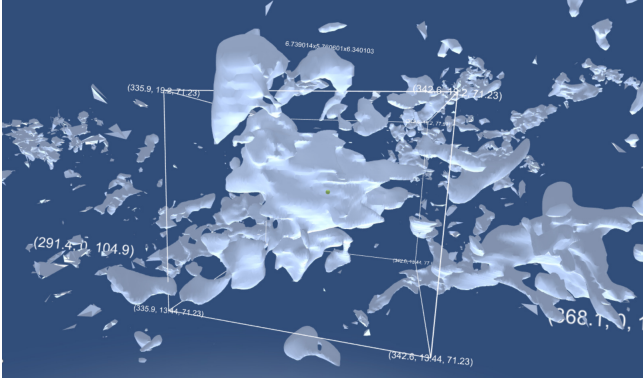


Fig. 10. Visualisation of the isosurface extracted from Dataset2 with isovalue 0.05. The structure's of interest bounding box is occluded by surrounding structures.

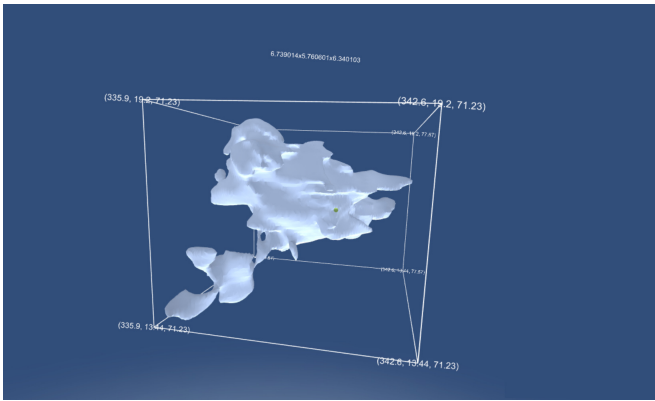


Fig. 11. Isolation of the structure of interest increases the structure's visibility (see also Fig. 10).

structure of interest. The solution to this problem is isolating the structure of interest as can be seen in the Fig. 11. In this figure the structure of interest as well as the bounding box dimensions and point locations are visible.

As mentioned before, the autonomous navigation to structures of interest is one of the key features during the exploration of a large isosurface in a virtual reality environment. The markers placement and revisiting interaction feature addresses this navigation challenge. The user can mark a structure by using the 3D pointer. The marker is indicated by a green sphere (see Fig. 12) and annotated with the point coordinate relative to the volume grid. The markers revisiting process is performed by changing the viewpoint of the scene to the viewpoint of the according marker. Markers are revisited in sequential order. As an illustration, Fig. 13 shows 5 markers in an isosurface that were placed sequentially from A to E. In this case, the markers revisit process will go from marker A to marker E and then eventually come back to marker A.

D. Extending the input file format

As a proof of concept for the flexibility of the workflow, the input file format has been extended beyond the use case. For this purpose, isosurfaces have been extracted from medical



Fig. 12. Visualisation of marked structures of an isosurface. The green spheres indicate the markers.

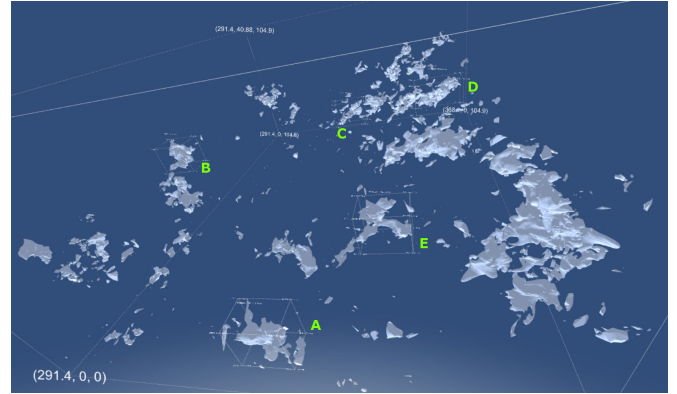


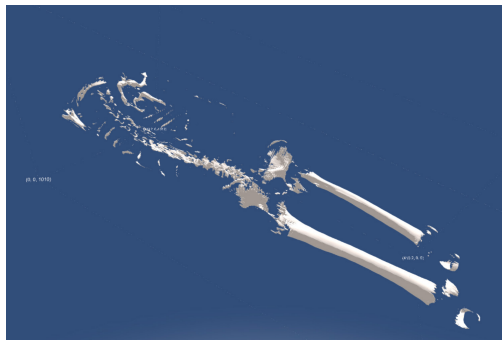
Fig. 13. Illustration of the markers revisit process: the revisit process will go from marker A to marker E and then eventually come back to marker A.

datasets. In medicine volume data or image data is usually generated by a CT scan or a MRI scan and stored in Digital Imaging and Communications in Medicine (DICOM) files. Figure 14 shows isosurfaces of a human skeleton and a human skull generated using the online isosurface extraction approach in a virtual reality environment. All of the interaction features work properly with this input data.

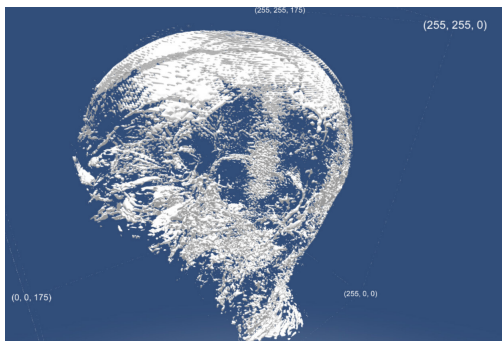
The CT scan and MRI scan data used here are in whole or part based upon data generated by the TCGA Research Network [23]. The human body data is from The Cancer Genome Atlas Sarcoma (TCGA-SARC) [24], and the human skull data is RIDER Neuro MRI [25] data.

VI. CONCLUSION

We proposed a workflow for interactive visualisation of large and complex isosurfaces in heterogeneous virtual reality environments. The workflow encompasses the pre-processing stage and the visualisation stage. In the pre-processing stage input files are transformed into a volume data format that can be visualised using off-the-shelf visualisation tools. The visualisation stage on the other hand handles the isosurface visualisation by providing an offline and an online isosurface extraction approach. For large isosurfaces with more than 1 million points using the offline approach is highly recommended. For smaller datasets the online isosurface extraction approach provides a more interactive way. The visualisation pipeline has also been demonstrated as being capable of providing isosurfaces visualisations for different types of input data.



(a) Human's skeleton.



(b) Human's skull.

Fig. 14. Isosurface visualisation of medical data in a virtual reality environment.

REFERENCES

- [1] J. Soria and C. Atkinson, "Towards 3C-3D digital holographic fluid velocity vector field measurement tomographic digital holographic PIV (Tomo-HPIV)," *Measurement Science and Technology*, vol. 19, no. 7, p. 074002, 2008.
- [2] C. Atkinson, S. Coudert, J.-M. Foucaut, M. Stanislas, and J. Soria, "The accuracy of tomographic particle image velocimetry for measurements of a turbulent boundary layer," *Experiments in Fluids*, vol. 50, no. 4, pp. 1031–1056, 2011.
- [3] V. Kitsios, C. Atkinson, J. A. Sillero, G. Borrell, A. G. Gungor, J. Jiménez, and J. Soria, "Direct numerical simulation of a self-similar adverse pressure gradient turbulent boundary layer," *International Journal of Heat and Fluid Flow*, vol. 61, pp. 129–136, 2016.
- [4] T. Chandler, M. Cordeil, T. Czauderna, T. Dwyer, J. Glowacki, C. Goncu, M. Klapperstueck, K. Klein, K. Marriott, F. Schreiber, and E. Wilson, "Immersive Analytics," in *Proc. Big Data Visual Analytics (BDVA), 2015*. IEEE eXpress Conference Publishing, 2015, pp. 73–80.
- [5] A. Febretti, A. Nishimoto, T. Thigpen, J. Talandis, L. Long, J. D. Pirtle, T. Peterka, A. Verlo, M. Brown, D. Plepys, D. Sandin, L. Renambot, A. Johnson, and J. Leigh, "CAVE2: A Hybrid Reality Environment for Immersive Simulation and Information Analysis," in *Proc. IS&T / SPIE Electronic Imaging*, vol. 8649, 2013, pp. 864903.1–12.
- [6] M. Cordeil, T. Dwyer, K. Klein, B. Laha, K. Marriott, and B. H. Thomas, "Immersive Collaborative Analysis of Network Connectivity: CAVE-style or Head-Mounted Display?" *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 441–450, 2017.
- [7] M. Klapperstück, T. Czauderna, C. Goncu, J. Glowacki, T. Dwyer, F. Schreiber, and K. Marriott, "Contextuwall: Peer Collaboration Using (Large) Displays," in *Proc. Big Data Visual Analytics (BDVA), 2016*. IEEE eXpress Conference Publishing, 2016, pp. 7–14.
- [8] M. Folk, G. Heber, Q. Koziol, E. Pourmal, and D. Robinson, "An Overview of the HDF5 Technology Suite and Its Applications," in *Proc. of the EDBT/ICDT 2011 Workshop on Array Databases*. ACM, 2011, pp. 36–47.
- [9] HDF Group, "HDF5," <https://support.hdfgroup.org/HDF5/>, 2017, [last accessed 29/06/17].
- [10] I. H. F. dos Santos, L. P. Soares, F. Carvalho, and A. Raposo, "A Collaborative VR Visualization Environment for Offshore Engineering Projects," in *Proc. of the 10th International Conference on Virtual Reality Continuum and Its Applications in Industry*. ACM, 2011, pp. 179–186.
- [11] S. Zhang, C. Demiralp, D. F. Keefe, M. DaSilva, D. H. Laidlaw, B. D. Greenberg, P. J. Basser, C. Pierpaoli, E. A. Chiocca, and T. S. Deisboeck, "An Immersive Virtual Environment for DT-MRI Volume Visualization Applications: A Case Study," in *Proc. of the Conference on Visualization '01*. IEEE Computer Society, 2001, pp. 437–440.
- [12] D. Acevedo, E. Vote, D. H. Laidlaw, and M. S. Joukowsky, "Archaeological Data Visualization in VR: Analysis of Lamp Finds at the Great Temple of Petra, a Case Study," in *Proc. of the Conference on Visualization '01*. IEEE Computer Society, 2001, pp. 493–496.
- [13] T. Duval, C. Fleury, B. Nouailhas, and L. Aguerreche, "Collaborative Exploration of 3D Scientific Data," in *Proc. of the 2008 ACM Symposium on Virtual Reality Software and Technology*. ACM, 2008, pp. 303–304.
- [14] C. Fleury, T. Duval, V. Gouranton, and A. Steed, "Evaluation of Remote Collaborative Manipulation for Scientific Data Analysis," in *Proc. of the 18th ACM Symposium on Virtual Reality Software and Technology*. ACM, 2012, pp. 129–136.
- [15] T. Bednarz, J. Kim, R. Brown, A. James, K. Burrage, S. Clifford, J. Davis, K. Mengersen, E. Peterson, S. Psaltis, and J. Vercelloni, "Virtual Reality for Conservation," in *Proc. of the 21st International Conference on Web3D Technology*. ACM, 2016, pp. 177–178.
- [16] M.-D. Vazquez-Duchêne, O. Freis, A. Denis, C. Mion, C. Jeanmaire, S. Mine, and G. Pauly, "Virtual Reality for Skin Exploration," in *Proc. of the Virtual Reality International Conference: Laval Virtual*. ACM, 2013, pp. 5.1–4.
- [17] A. Kageyama, Y. Tamura, and T. Sato, "Visualization of Vector Field by Virtual Reality," *Progress of Theoretical Physics Supplement*, no. 138, pp. 665–673, 2000.
- [18] A. Kageyama, T. Hayashi, R. Horiuchi, and K. Watanabe, "Data Visualization by a Virtual Reality System," in *Proc. of 16th International Conference on the Numerical Simulation of Plasmas*, 1998, pp. 138–142.
- [19] J. Schulze-Döbold, U. Wössner, S. P. Walz, and U. Lang, "Volume Rendering in a Virtual Environment," in *Proc. of the Eurographics Workshop on Virtual Environments*, B. Fröhlich, J. Deisinger, and H.-J. Bullinger, Eds. The Eurographics Association, 2001, pp. 187–198.
- [20] S. Sua, A. Chaudhary, P. O'Leary, B. Geveci, W. Sherman, H. Nieto, and L. Francisco-Revilla, "Virtual reality enabled scientific visualization workflow," in *Proc. 2015 IEEE 1st Workshop on Everyday Virtual Reality (WEVR)*. IEEE, 2015, pp. 29–32.
- [21] S. Marks, J. E. Estevez, and A. M. Connor, "Towards the Holodeck," in *Proc. of the 29th International Conference on Image and Vision Computing New Zealand - IVCNZ '14*. ACM Press, 2014, pp. 42–47.
- [22] W. Schroeder, K. Martin, and B. Lorensen, *The Visualization Toolkit*. Kitware, 2006.
- [23] K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle, L. Tarbox, and F. Prior, "The Cancer Imaging Archive (TCIA): Maintaining and Operating a Public Information Repository," *Journal of Digital Imaging*, vol. 26, pp. 1045–1057, 2013.
- [24] C. Roche, E. Bonaccio, and J. Filippini, "Radiology Data from The Cancer Genome Atlas Sarcoma [TCGA-SARC] Collection," <http://doi.org/10.7937/K9/TCIA.2016.CX6YLSUX>, 2016, [last accessed 29/06/17].
- [25] D. Barboriak, "Data from RIDER NEURO MRI," <http://doi.org/10.7937/K9/TCIA.2015.VOSN3HN1>, 2015, [last accessed 29/06/17].