Fusion Engineering and Design 146 (2019) 551–554

Contents lists available at ScienceDirect

Fusion Engineering and Design

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

Real-time volumetric rendering of radiation fields using 3D textures

Ronan Kelly[a], Robert Skilton[a], Jonathan Naish[b]

[a] RACE, UKAEA, Culham Science Centre, Abingdon, OX14 3DB, United Kingdom
[b] CCFE, UKAEA, Culham Science Centre, Abingdon, OX14 3DB, United Kingdom

ARTICLE INFO

Keywords:
Dose map
Data visualisation
JET
Virtual reality

ABSTRACT

The remote maintenance of nuclear fusion facilities must be carefully planned and monitored due to the potential damage to equipment caused by the ionising radiation that results from neutron activation of facility components. We present a method for visualising three-dimensional radiation dose fields in real-time. An interactive volumetric representation is achieved using view-dependent parallel ray casting of a scalar field in three dimensions. This enhances the planning of remote handling operations by allowing motion paths to be better optimised for the minimisation of dose to vital components, prolonging the lifetime of remote handling equipment.

1. Introduction

Remote handling technology is used for repairing components in tokamaks that have been damaged or activated by energetic particles produced by nuclear fusion reactions [1]. Simulation software such as the Monte-Carlo N-Particle Transport Code [2] (MCNP) is used regularly for planning such activities [3]. Engineers use the results to forecast and therefore minimise radiation doses to material, equipment and personnel involved in operation and maintenance of fusion devices such as the Joint European Torus (JET) [4]. We expect this process to become operations-critical for future high-yield fusion facilities such as ITER and DEMO.

Visualising the results of such simulations is essential for operations engineers to develop remote handling tasks that minimise the dose to equipment. A typical output of the simulation process is a cell-based data format such as Visual ToolKit (VTK) which can be used for offline expert analysis or plotting. The results are tightly related to the CAD geometry used in the simulation, and hence are often visualised as an overlay [5].

In this contribution a new method for the real-time visualisation of simulation results as a dense volume is presented – see example results in Fig. 1. Although sample data stored as ASCII VTK has been used for this paper, the software architecture is generic to allow future extension to other file formats.

We achieve interactive real-time performance by exploiting parallel processing on the graphics processing unit (GPU), with scalar field data uploaded as three-dimensional (3D) textures. Lack of consumer hardware support for true 3D texturing is mitigated with software indexing of two-dimensional textures containing multiple XY slices through the volume. Integration with a commercial-off-the-shelf open source software framework, Unreal Engine 4 (UE4), allows fusing with context geometry derived from computer-aided design (CAD) engineering models and viewing with virtual reality headsets to support remote handling task development.

Readers should note that all figures are purely for illustration, and source data has not been verified or approved for use in any engineering assessments.

2. Radiation dose maps

Remote handling operations engineers at facilities such as JET plan operations in a process known as task development, using real-time simulations of remote handling equipment such as VR4Robots to check for collisions and develop optimal motion sequences [6]. Simulation results are typically presented to remote handling operations engineers for use in task development as dose maps in a static plot (see Fig. 2). Even when this plot is 3D, it is difficult and error-prone to reconcile with a remote handling simulation which may contain different context geometry, unit scale and colour coding. This motivated the development of our method, which allows full integration with remote handling simulations in UE4.

Previous work by the UK Atomic Energy Authority developed the VORTEX software package [7] for representing radiation as a 3D overlay of contours onto CAD geometry for use with VR headsets. VORTEX is optimised for tracing the dose received by human operators in a volume, whereas our method is focused on expert visual assessment of the dose to remote handling equipment. Rather than showing only isosurfaces to show the contour of dose fields, we have developed a way to simulate a dense volume of varying density analogous to a cloud of water vapour. Similar functionality is available in the open-source scientific data visualisation tool ParaView [8], but is not suited for integration into existing remote handling simulation tools.

https://doi.org/10.1016/j.fusengdes.2019.01.020
Received 8 October 2018; Received in revised form 4 January 2019; Accepted 4 January 2019
Available online 11 January 2019
0920-3796/

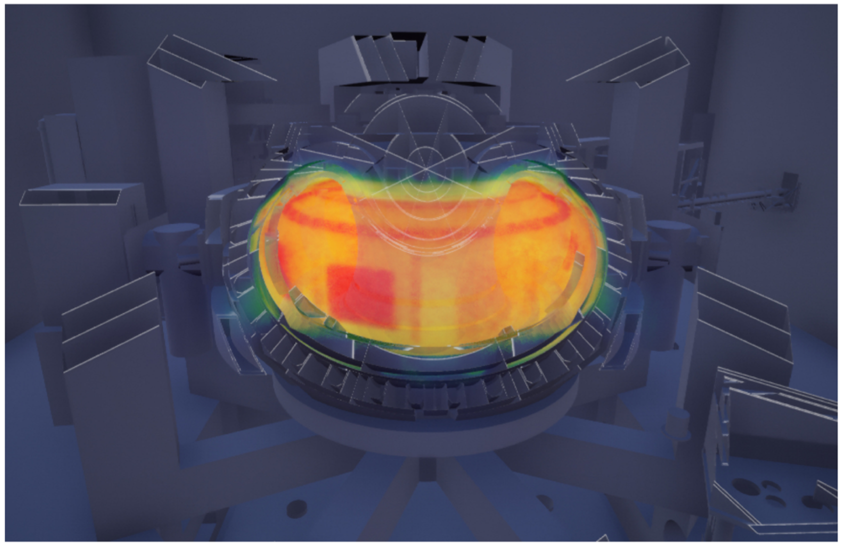

Fig. 1. Our method – a dose volume visualised with a cutaway model of the JET vessel.

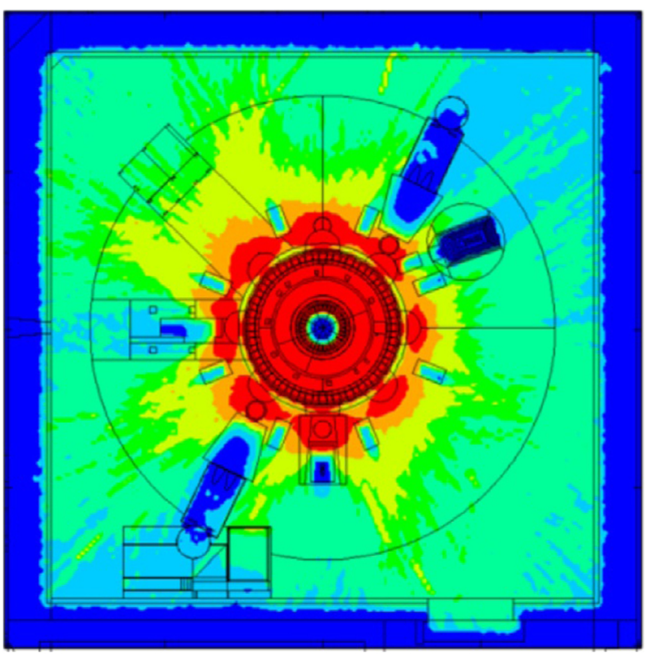

Fig. 2. Traditional 2D dose map (with axes removed). Reproduced with author's permission from [7].

3. Raymarching volumetric data

Our method introduces a ray-casting step into the rasterisation pipeline used traditionally for real-time graphics. A simple mesh such as a cube bounds the volume to be rendered, and is passed to the fragment shader where our method calculates each pixel colour. Due to the lack of commercial GPU support for 3D textures, volumetric data is pre-processed offline using a Python script integrated with UE4 to create a 2D RGBA texture (as seen in Fig. 3). Shader programs are generated from a higher-level representation (UE4 materials) and use a custom ray-marching function to sample the volume texture at multiple, regular intervals along the camera vector at each pixel, with the object-space position transformed into texture-space and decomposed to 2D coordinates to index into the texture. Sampled values are decoded from four 8-bit integers (RBGA) into a 32-bit floating point value.

In the default configuration the accumulated value along all sample points is used as input to Beer's Law of light attenuation to calculate final opacity, thus approximating a volume of varying density (see Fig. 4). Alternatively, a solid isosurface representation can be achieved by finding the shortest distance from the camera to a sample exceeding the configured threshold. This surface can then be shaded by calculating the surface normal by multisampling the volume along each axis (see Fig. 6). The maximum value sampled is then used as input to a heat-map function that indexes into a colour scale texture to give the final colour (see Fig. 5).

4. Evaluation

The rendering technique has been tested in a traditional desktop setup with consumer-grade hardware and rendered to stereo in a more immersive setting with virtual reality headsets.

The application has been tested on input VTK files of a maximum resolution of 363^3 cells, resulting in a texture map of 7260^2 32-bit pixels. The default maximum texture size supported by the Unreal Engine is 8192^2, which is also the limit imposed by most consumer GPU hardware. This equates to around 268.4MB of raw color data before compression. A typical consumer GPU in 2018 has 8GB of onboard memory, implying that data of over around 24× the tested resolution could be used without requiring specialist hardware. However the limitation of maximum texture sizes would mandate development of a technique for looking up data from an atlas of textures, a form of virtual texturing. A novel feature of this method over rasterising meshes of isosurfaces is that render latency scales with the screenspace resolution rather than the volume of data.

The rendering technique has been tested in a traditional desktop setup with consumer-grade hardware and rendered to stereo in a more immersive setting with virtual reality headsets.

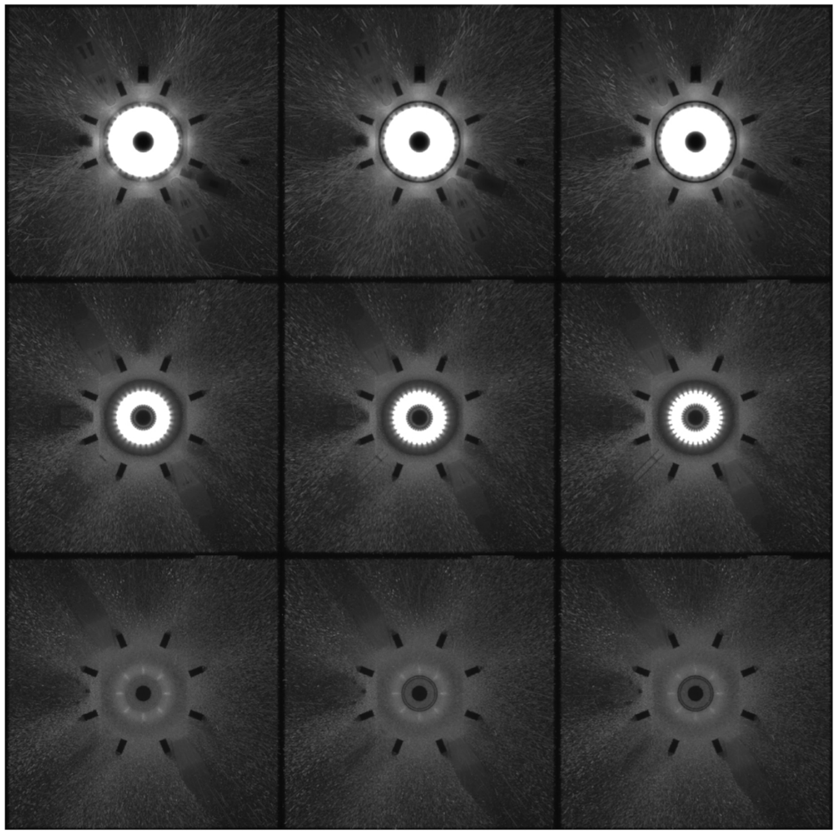

Fig. 3. Subsection of an example 2D 32-bit float texture, generated from JET gamma dose simulation results in VTK format.

Fig. 4. Simulated dose map of JET post DT2 campaign. Note lowered opacity around centre of the torus as an example of the density effect.

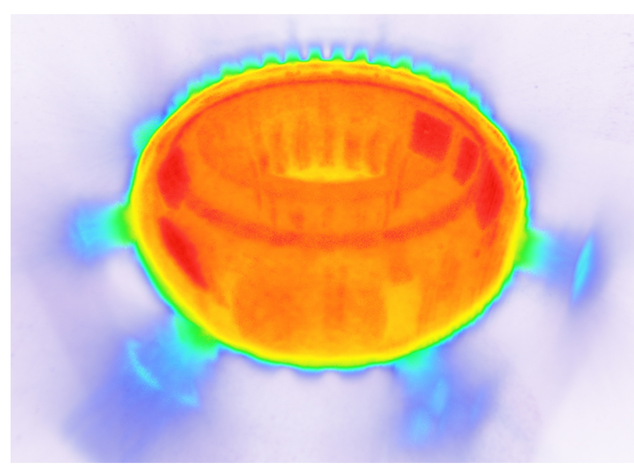

Fig. 5. Example of heatmap colouration, where warmer colours correspond to a higher dose.

5. Future work

While targeted at rendering radiation fields for remote maintenance planning in nuclear fusion facilities, our method inherently generalises to the visualisation of any scalar quantity that can be sampled in a three-dimensional field, such as neutron dose, plasma density, temperature or pressure. Visualisation of plasma density could be achieved at very high resolutions by exploiting the radial symmetry of density profiles in tokamaks, allowing a single 2D texture of a radial slice to be used for the entire volume. The method is currently restricted to use with data that has been pre-processed into a 2D texture. Extension to support online processing directly from VTK files or even live data streamed to the GPU would be feasible using UE4 render targets with significant further development. For radiation dose visualisation this may be of limited use as three-dimensional characterisation of dose fields in real-time is not currently possible. However for other data such as plasma density, real-time visualisation is plausible.

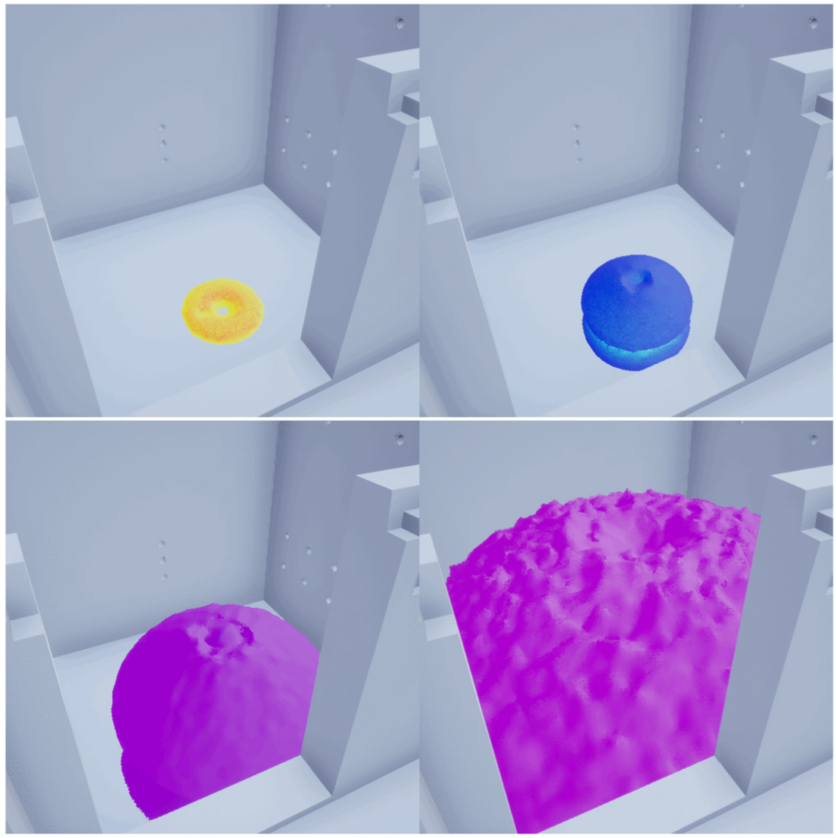

Fig. 6. Sequence illustrating isosurface representation, with threshold decreasing left to right. Note the surface shading provided by estimating the normal at each pixel.

References

[1] R. Buckingham, A. Loving, Remote-handling challenges in fusion research and beyond, Nature Physics 12 (5) (2016) 391.

[2] J.F. Briesmeister, MCNP: A General Monte Carlo Code for Neutron and Photon Transport. version 3a. revision 2, Tech. Rep. Los Alamos National Lab, 1986.

[3] R. Villari, P. Batistoni, S. Conroy, A. Manning, F. Moro, L. Petrizzi, S. Popovichev, D. Syme, J.E. Contributors, Shutdown dose rate benchmark experiment at jet to validate the three-dimensional advanced-d1s method, Fusion Engineering and Design 87 (7-8) (2012) 1095–1100.

[4] L. Snoj, I. Lengar, A. Čufar, B. Syme, S. Popovichev, S. Conroy, L. Meredith, J.E. Contributors, Calculations to support jet neutron yield calibration: modelling of the jet remote handling system, Nuclear Engineering and Design 261 (2013) 244–250.

[5] Y. Wu, F. Team, et al., Cad-based interface programs for fusion neutron transport simulation, Fusion Engineering and Design 84 (7-11) (2009) 1987–1992.

[6] N. Sykes, S. Collins, A. Loving, V. Ricardo, E. Villedieu, Design for high productivity remote handling, Fusion Engineering and Design 86 (9) (2011) 1843–1846, https://doi.org/10.1016/j.fusengdes.2011.04.004 Proceedings of the 26th Symposium of Fusion Technology (SOFT-26). URL: http://www.sciencedirect.com/science/article/pii/S092037961100398X.

[7] J. Naish, A. Burns, Minimising operator dose during jet shutdown using virtual, Fusion Engineering and Design 124 (2017) 1215–1218, https://doi.org/10.1016/j.fusengdes.2017.03.131 Proceedings of the 29th Symposium on Fusion Technology (SOFT-29). URL: http://www.sciencedirect.com/science/article/pii/S0920379617303496.

[8] KitWare Inc, The ParaView Guide, (2016).