Visualization of human-scale blood flow simulation using Intel OSPRay Studio on SuperMUC-NG

Mayer, E.[1], McCullough, J.W.S.[2], Gunther, J.[3], Cielo, S.[1], Coveney, P.V.[2,4]

[1]Leibniz Supercomputing Centre (LRZ)
[2]University College London (UCL)
[3]Intel
[4]University of Amsterdam

1. Background

Significant effort from many research groups is currently directed towards creating a virtual human – a personalized digital representation of a human's biophysical processes – to help optimize healthcare for a given individual's needs. Achieving this goal will require input from scientific, computational and medical specialists alike. The input of all the parties is essential to ensure that the virtual human can capture an individual's physiological characteristics accurately, that the simulations constructing it can be conducted efficiently and that the data generated can be stored and analysed effectively. By design, human-scale simulations can generate significant quantities of data and this, in turn, demands tools that can provide high-impact visualizations for the ultimate end-users: clinicians and patients.

As part of our efforts in moving towards virtual human simulations, the team members at UCL have been utilising the HemeLB 3D blood flow solver to develop human-scale simulations of arterial and venous geometries. For such simulations to accurately resolve flow features, geometries consisting of many millions, if not billions, of data locations are required. This tool can rapidly generate large quantities of data which can be challenging to post-process. In this presentation, we will discuss how a collaboration with team members from LRZ and Intel has developed a workflow that can efficiently generate near real-time visualisations of the data generated by HemeLB by utilising the same high-performance computing resources used to conduct the simulations. With this workflow, it is possible to efficiently produce 2D images and videos but also 3D and immersive visualisations that are used for virtual reality (VR) experiences.

2. Methodology

The open-source HemeLB code [1, 2] has been developed with a specific focus on the study of three-dimensional vascular flows at high-resolution and at extreme scales. This is enabled through the use of the lattice Boltzmann method (LBM), a non-traditional approach to solving the Navier-Stokes equations for fluid flow that possesses many benefits in high performance computing. The localised computational kernel of the LBM permits two particular advantages for blood flow simulation. The first is that it is a straightforward algorithm to parallelise leading to good computational performance on both small and large computational scales. The second is that it enables complex geometric shapes to be accommodated using local assignation of the simulation site to its particular category – e.g. fluid, wall, inlet, outlet. HemeLB has been specifically optimised to enable efficient performance on the sparse geometries that are

characteristic of vascular domains. We have already demonstrated this performance to large scale on some of the largest and fastest supercomputers currently available [3].

To generate a proof-of-concept dataset for visualisation, we utilised our self-coupled implementation of HemeLB [2, 3] to run a simulation of blood flow through the arteries and veins of the left forearm. This was run for the period of a single heartbeat, and we output a total of 64 frames of data at a rate of approximately 7.4GB per step. This simulation was conducted on LRZ's SuperMUC-NG [4] supercomputer. The dataset chosen provided a good basis for developing a visualization workflow in terms of data quantity and complexity in advance of full human-scale studies.

In parallel to the simulation development, team members at LRZ and Intel made use of Intel® OSPRay Studio [5] to develop a workflow for visualizing the HemeLB data directly on the production machine. We adopt a custom input plugin to map the simulation data to the highly memory-efficient VDB volumes of Intel Open Volume Kernel Library for delivering visualization at native resolution and interactive frame rates, eliminating complex data pre-processing steps. Additionally, we have developed tools to enable visualisation of the HemeLB data as part of a stereographic 360° VR experience.

3. Results
The post-processing of the HemeLB data on SuperMUC-NG allows visualizations to be viewed in near real time. These are highly customizable to permit different viewpoints, rendering and focus points. The use of the stereographic virtual reality rendering will allow the end-user to be immersed within the visualization and move to areas of interest. We provide example images from our workflow in Figure 1, whilst Figure 2 demonstrates an example 360° image in non-stereo form. We will provide video examples of our workflow and its use through our presentation.

4. Summary
Our approach allows the full domain simulated by HemeLB to be observed at a glance, a challenging task with existing techniques. In a clinical setting, such visualization capacity will allow improved comprehension of an individual's vasculature and blood flow characteristics for both medics and patients alike.
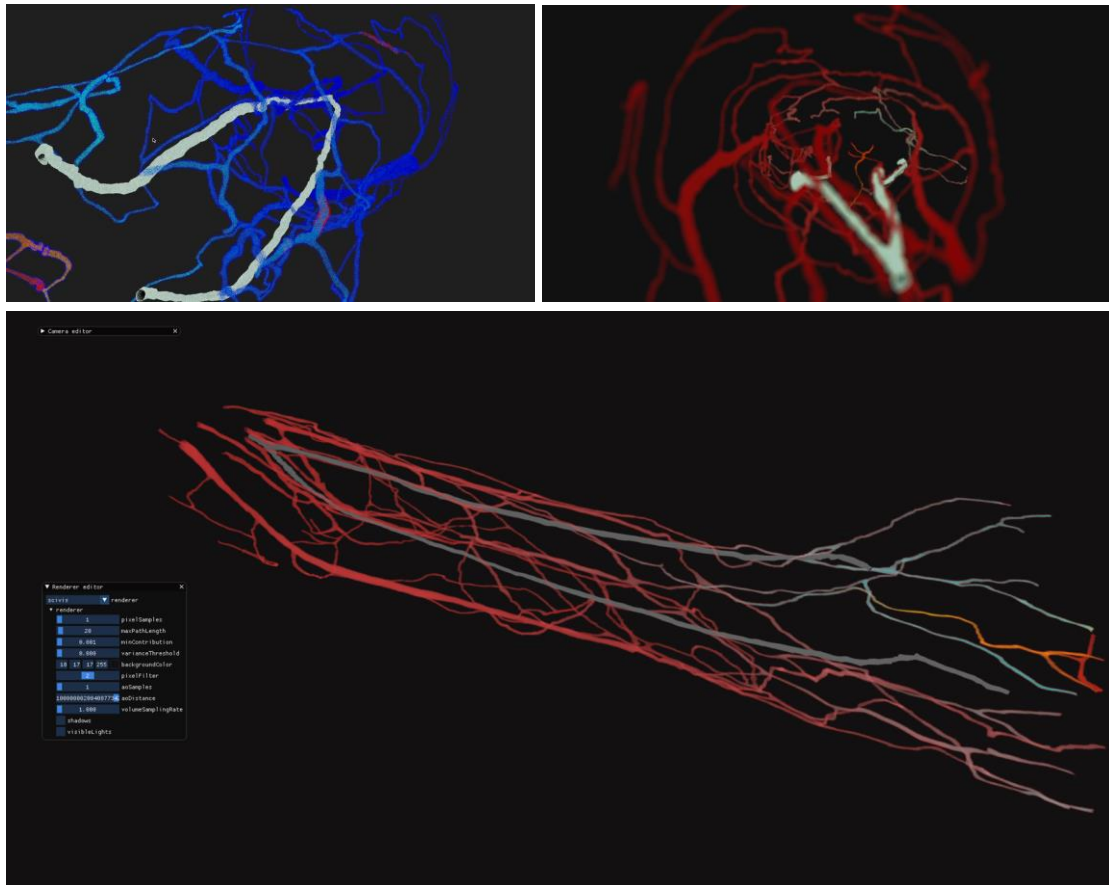
Figure 1 *Examples of data visualisation of the HemeLB arterial and venous simulations using our workflow developed on SuperMUC-NG. The visualisation options are extremely customizable in terms of focus, orientation and rendering options.*
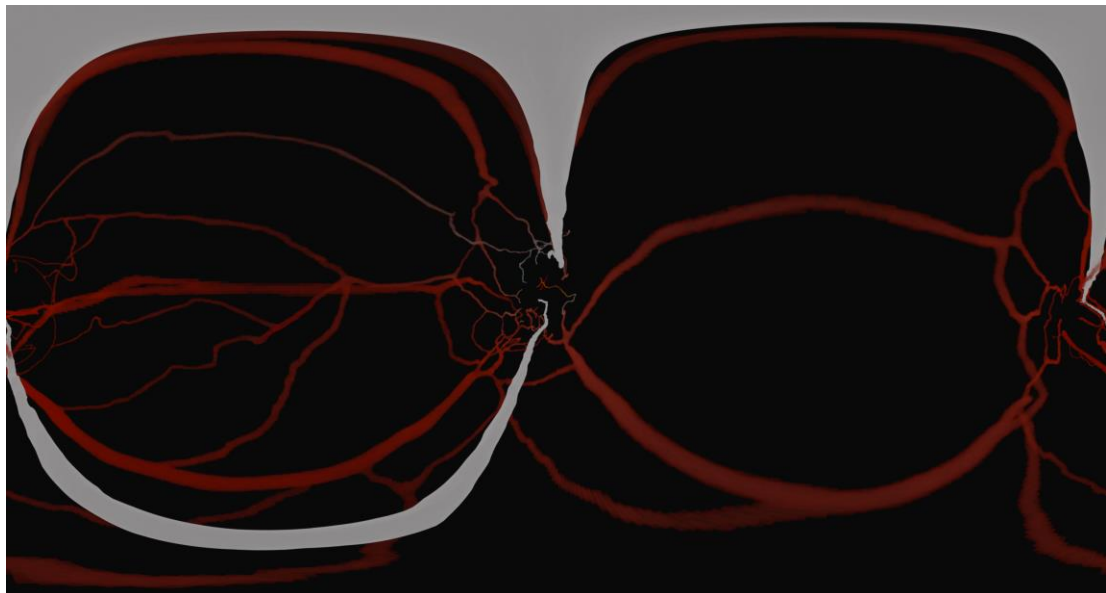
Figure 2 *Example of a non-stereo 360° image of the HemeLB data.*

5. References

1. HemeLB website – hemelb.org
2. HemeLB Code Repositories - https://github.com/hemelb-codes (main), https://github.com/UCL-CCS/HemePure, https://github.com/UCL-CCS/HemePure_SelfCoupled
3. J. W. S. McCullough, R. A. Richardson, A. Patronis, R. Halver, R. Marshall, M. Ruefenacht, B. J. N. Wylie, T. Odaker, M. Wiedemann, B. Lloyd, E. Neufeld, G. Sutmann, A. Skjellum, D. Kranzlmüller and P. V. Coveney, "Towards blood flow in the virtual human: efficient self-coupling of HemeLB", J R Soc Interface Focus11, 20190119 (2020), DOI:10.1098/rsfs.2019.0119
4. LRZ website: https://www.lrz.de/
5. Intel® OSPRay Studio: https://www.ospray.org/