



**HAL**  
open science

# Towards Scalable Out-of-Core Volume Rendering for High-Performance Visualization of 3D Temporal Multivariate Gas and Fluid Data

Antoine Thebault, Stéphanie Prévost, Laurent Lucas, Leonardo Brenner

## ► To cite this version:

Antoine Thebault, Stéphanie Prévost, Laurent Lucas, Leonardo Brenner. Towards Scalable Out-of-Core Volume Rendering for High-Performance Visualization of 3D Temporal Multivariate Gas and Fluid Data. EuroVis 2025 - Posters, 2025, <10.2312/evp.20251135>. <hal-05141659>

**HAL Id: hal-05141659**

**<https://hal.science/hal-05141659v1>**

Submitted on 3 Jul 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY 4.0 - Attribution - International License

# Towards Scalable Out-of-Core Volume Rendering for High-Performance Visualization of 3D Temporal Multivariate Gas and Fluid Data

A. Thebault<sup>1</sup> , S. Prevost<sup>1</sup> , L.Lucas<sup>1</sup>  and L.Brenner<sup>1</sup> , <sup>†</sup>

<sup>1</sup>Universite de Reims Champagne-Ardenne, CEA, LRC DIGIT, LICHS, chaire OSMIUM, Reims, France

## Abstract

Scientific visualization enables a deeper insight into the data structure and properties. For Gas and fluid simulations, particularly those resulting from real-world gas captures or Computational Fluid Dynamics (CFD) models, this requires the analysis and integration of multivariate attributes alongside dynamic temporal variations. We present an ongoing work to develop a scalable, Out-of-core volume rendering application designed for massive 3D temporal multivariate datasets. Our rendering framework, called FRIGAS (Fluid Rendering Infrastructure for Gas and Atmospheric Simulations), introduces a novel pipeline which leverages improvements in Direct Volume Rendering (DVR) techniques for large scale visualizations, addressing constraints regarding temporal structured data and multivariate analysis for interactive spatial and temporal navigation without HPC resources.

## CCS Concepts

• **Computing methodologies** → Ray tracing; Shared memory algorithms; • **Software and its engineering** → Software design engineering;

## 1. Introduction

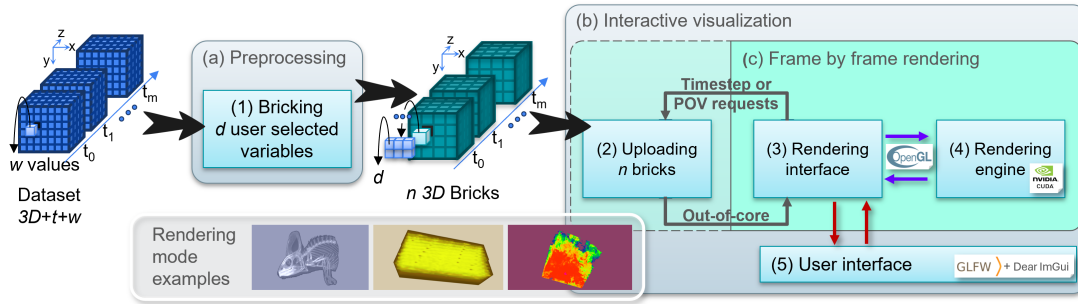
Nowadays, scientific visualization tools, particularly those for volume rendering, are outpaced the exponential growing size of data. In the context of computational fluid dynamics (CFD), tools like Paraview [Aya15] and VTK [SML06] remain widely used, offering intuitive UIs and robust statistical analysis. However, their rendering engines lack scalability without HPC resources, making them less suited for 3D temporal multivariate datasets or simultaneous visualization of large-scale correlated variables on one machine.

Recent works [WBDM23, HHK\*24, ABDS\*23] have explored improving accessibility to graphical information without compromises, even on consumer-grade hardware. This poster presents our ongoing work on developing a scalable, Out-of-core visualization pipeline designed for large-scale 3D temporal multivariate gas and fluid datasets. Our approach aims to enhance both visualization and interactivity, ensuring efficient handling of complex datasets while maintaining high-performance rendering.

<sup>†</sup> The LRC DIGIT, a joint lab between CEA-DAM Ile de France and URCA, supports the OSMIUM CPJ and this research work. Authors would also like to thank D. Lima De Bem and R. Yuvaraj for the datasets.

## 2. Related Works

Volumetric rendering has evolved with advances in GPU computing [Nic07], enabling real-time visualization of large datasets [SZD\*23]. Hierarchical data structures, such as KD-Trees [VMD08, ZSL18] and octrees [LK11, HHK\*24], improve efficiency by subdividing voxel grids, while bricking techniques [SCRL20] optimize spatial partitioning. Another approach, Dynamic Scheduling [FBM\*15], leverages R-Trees [Gut84] to optimize data queries both before and during processing. Out-of-core rendering provides an effective strategy for managing large-scale data uploads using virtual page tables. Initially demonstrated for 3D datasets on GPUs [SCRL20], this method has since been adapted for Adaptive Mesh Refinement (AMR) data in  $3D + t$  contexts [ABDS\*23]. While these implementations are designed for single-variable datasets, they rely on a fixed 128-bit per voxel brick structure due to memory bandwidth constraints. We note that none of the previous state-of-the-art handle large temporal multivariate ( $3D + t + w$ ) datasets, i.e.  $3D$  ( $x, y, z$ ) grid with  $t$  timesteps, where each voxel is associated with  $w$  variables, which can store either scalar (e.g., temperature or pressure), vectors (e.g., velocity fields) or tensors. Our contribution aim at extending these approaches to support efficient simultaneous multivariate temporal access, enabling broader applications and analysis in scientific visualization.



**Figure 1:** Pipeline description from left to right ; In pre-processing (a), we brick the raw dataset in  $3D+t+w$ , with one brick per timestep and with  $w$  reduced according to  $d$  user selected variables. In visualization (b), the rendering engine (4) on GPU retrieves the corresponding bricks under the rendering interface (3) memory management on CPU. The render frame is updated on demand, by the user through the UI (5), with specific timestep or point of view (POV) requests. At the bottom, a few examples of rendering modes: MIDA, DVR, MIP with LUT.

### 3. Contributions

We propose an adaptive pipeline composed of independent modules for better flexibility, directly implemented into our application called FRIGAS. This approach allows us to contribute on the Out-of-core while supporting existing improvements made by [SCRL20]. Indeed, we provide a novel implementation with  $3D+t$  support on structured grids as in [ABDS\*23], as well as providing multivariate data support for an improved and robust Out-of-core structure. During our bricking process, the  $3D+t+w$  input data is subdivided into structured bricks, grouping voxels into fixed-size blocks ( $64^3$  as in [SCRL20, ABDS\*23]). For multivariate datasets, we added a step for the user to select  $d$  from the  $w$  variables to be visualized. However, the pipeline remains format-agnostic. Thus, bricks are normalized, converting values into a standardized range. A Rendering Interface then receives the preprocessed and normalized bricks, loaded onto the CPU and GPU for traversal by the Rendering Engine. Our pipeline is versatile due to its modular conception, through additional rendering modes variants and data preprocessing.

If the Rendering Engine takes advantage of a cache system, it still shall not directly manage the data. Instead, it receives tasks as specific rendering and memory requests handled by the CPU request manager. This distinction is crucial for GPU memory management as an added  $d$  dimension can quickly exceed VRAM capacity. The challenge is to join brick-based rendering with GPU-efficient data access, while maintaining independence from data structures and formats, as well as accommodating visualization modes. This abstraction layer also facilitates external connections, such as APIs or integration in other applications.

### 4. FRIGAS: An application of our pipeline

FRIGAS (Fluid Rendering Infrastructure for Gas and Atmospheric Simulations) implements the aforementioned pipeline and allows us to experiment with both the Out-of-core architecture and visualization pipeline in Fig.1. The Out-of-core system proposes for now a set number of variables constrained by the 128 bits memory bandwidth limit to be loaded dynamically, thus either 4 *float32* or 8 *uint16*. Bricks and Page tables retain flags from [SCRL20] to

indicate their state in memory and description. It should be noted that Bricks are not constrained to a 128 bits limit and preprocessing uses its own configuration as an independent module. The current visualization system relies on various rendering modules to handle the variables. These modes are maximum (minimum, mean) intensity projection MIP (minIP, meanIP), MIDA [BG09], Volume Rendering (notably joint DVR with 1 to N simultaneous variables), IsoSurface and 3D properties related projections (Positions, Depth, Normals...). Frame-by-frame updates are on-demand, by user interactions or temporal requests for interactive temporal and spatial navigation.

Preprocessing supports custom 3D datasets (BVD, DAT), the NSDF format [Kla16] and NetCDF files with  $3D+t+w$  data. Dataset sizes range from GB to TB, with the number of variables and timesteps varying. Our first dataset consists of 10 (*float32*) variables per voxel, which are stored in a NetCDF format with 2005 timesteps on a  $\{61, 61, 81\}$  grid representing smoke evaporation through CFD simulations [BLS15]. A second dataset, derived from camera-LiDAR methane-based gas captures [YLC\*24], contains 8 (*float32*) gas concentration per voxel on a  $\{72, 32, 44\}$  grid with 488 timesteps. Our third dataset is a  $\{1536, 512, 512\}$  volume with 600 timesteps, with memory limits even without the multivariate analysis and up to 3 Gigabytes per variable per timestep.

### 5. Conclusion and On-going works

In this poster, we showcased our application FRIGAS with its pipeline. Through its conception, we aim for better flexibility and scalability in visualization of temporal multivariate large-scale data. At the moment, FRIGAS handles an Out-of-core temporal navigation of  $3D+t+w$  data processing and visualization, as well as  $3D+t$  and  $3D+w$ , albeit with a 128 bits limit for multivariate bricks. Building on [SCRL20], we are enhancing the multi-resolution page directory's (MRPD) structure as introduced in [ABDS\*23] with a new format that enables parallel brick loading with a shared *id* across  $d$  variables. Key areas of experimentation include loading performance and benchmarking using Out-of-core and CUDA Zero Copy [NVI00]. Finally, we actively collaborate on preprocessing support for data formats and welcome discussions on dataset integration.

## References

- [ABDS\*23] ALEXANDRE-BARFF W., DELEAU H., SARTON J., LEDOUX F., LUCAS L.: A GPU-based Out-of-core Architecture for Interactive Visualization of AMR Time Series Data. In *Eurographics Symposium on Parallel Graphics and Visualization* (2023), Bujack R., Pugmire D., Reina G., (Eds.), The Eurographics Association, pp. 1–11. doi:10.2312/pgv.20231080. 1, 2
- [Aya15] AYACHIT U.: *The ParaView Guide: A Parallel Visualization Application*. Kitware, Inc., Clifton Park, NY, USA, 2015. 1
- [BG09] BRUCKNER S., GRÖLLER M. E.: Instant Volume Visualization using Maximum Intensity Difference Accumulation. *Computer Graphics Forum* 28, 3 (June 2009), 775–782. doi:10.1111/j.1467-8659.2009.01474.x. 2
- [BLS15] BOSILOVICH M. G., LUCCHESI R., SUAREZ M.: *MERRA-2: File Specification*. Tech. Rep. 9, GMAO Office, 2015. 2
- [FBM\*15] FLATKEN M., BERRER A., MERKEL J., HOTZ I., GERNDT A., HAGEN H.: Dynamic Scheduling for Progressive Large-Scale Visualization. In *Eurographics Conference on Visualization (EuroVis) - Short Papers* (2015), Bertini E., Kennedy J., Puppo E., (Eds.), The Eurographics Association, pp. 36–41. doi:10.2312/eurovisshort.20151122. 1
- [Gut84] GUTTMAN A.: R-trees: a dynamic index structure for spatial searching. In *Proceedings of the 1984 ACM SIGMOD International Conference on Management of Data* (New York, NY, USA, 1984), SIGMOD '84, Association for Computing Machinery, p. 47–57. doi:10.1145/602259.602266. 1
- [HHK\*24] HERZBERGER L., HADWIGER M., KRÜGER R., SORGER P., PFISTER H., GRÖLLER E., BEYER J.: Residency octree: A hybrid approach for scalable web-based multi-volume rendering. *IEEE Transactions on Visualization and Computer Graphics* 30, 1 (Jan. 2024), 1380–1390. doi:10.1109/TVCG.2023.3327193. 1
- [Kla16] KLACANSKY: OpenVis shared dataset. <https://klacansky.com/open-scivis-datasets/>, 2016. 2
- [LK11] LAINE S., KARRAS T.: Efficient sparse voxel octrees. *IEEE Transactions on Visualization and Computer Graphics* 17, 8 (2011), 1048–1059. doi:10.1109/TVCG.2010.240. 1
- [Nic07] NICKOLLS J.: GPU parallel computing architecture and CUDA programming model. In *2007 IEEE Hot Chips 19 Symposium (HCS)* (2007), pp. 1–12. doi:10.1109/HOTCHIPS.2007.7482491. 1
- [NVI00] NVIDIA: Cuda zero copy (documentation 12.6). <https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html?highlight=zero#zero-copy-memory>, 2000. 2
- [SCRL20] SARTON J., COURILLEAU N., REMION Y., LUCAS L.: Interactive visualization and on-demand processing of large volume data: A fully GPU-based Out-of-Core approach. *IEEE Transactions on Visualization and Computer Graphics* 26, 10 (2020), 3008–3021. doi:10.1109/TVCG.2019.2912752. 1, 2
- [SML06] SCHROEDER W., MARTIN K., LORENSEN B.: *The Visualization Toolkit (4th ed.)*. Kitware, 2006. 1
- [SZD\*23] SARTON J., ZELLMANN S., DEMIRCI S., GÜDÜKBAY U., ALEXANDRE-BARFF W., LUCAS L., DISCHLER J. M., WESNER S., WALD I.: State-of-the-art in large-scale volume visualization beyond structured data. *Computer Graphics Forum* 42, 3 (2023), 491–515. doi:10.1111/cgf.14857. 1
- [VMD08] VIDAL V., MEI X., DECAUDIN P.: Simple empty-space removal for interactive volume rendering. *J. Graphics Tools* 13 (01 2008), 21–36. doi:10.1080/2151237X.2008.10129258. 1
- [WBDM23] WU Q., BAUER D., DOYLE M. J., MA K.-L.: Interactive volume visualization via multi-resolution hash encoding based neural representation. *IEEE Transactions on Visualization and Computer Graphics* 30(8), 5404–5418. (2023). doi:10.1109/TVCG.2023.3293121. 1
- [YLC\*24] YUVARAJ R., LAUVAUX T., CIAIS P., BONNE J.-L., JOLY L., GROSHENRY A., BA A.: High-resolution modelling of methane plumes: validation and sensitivity experiments to explore UAV and satellite observations. In *EGU General Assembly Conference Abstracts* (Apr. 2024), p. 18958. doi:10.5194/egusphere-egu24-18958. 2
- [ZSL18] ZELLMANN S., SCHULZE J. P., LANG U.: Rapid k-d Tree Construction for Sparse Volume Data. In *Eurographics Symposium on Parallel Graphics and Visualization* (2018), Childs H., Cucchiatti F., (Eds.), The Eurographics Association, pp. 69–77. doi:10.2312/pgv.20181097. 1