Gradient Traversal: Accelerating Real-Time Rendering of Unstructured Volumetric Data

Mehmet Oguz Derin
Morgenrot, Inc.
South Korea
oguz@morgenrot.net

Takahiro Harada
Advanced Micro Devices, Inc.
United States of America
Morgenrot, Inc.
United States of America
takahiro.harada@amd.com

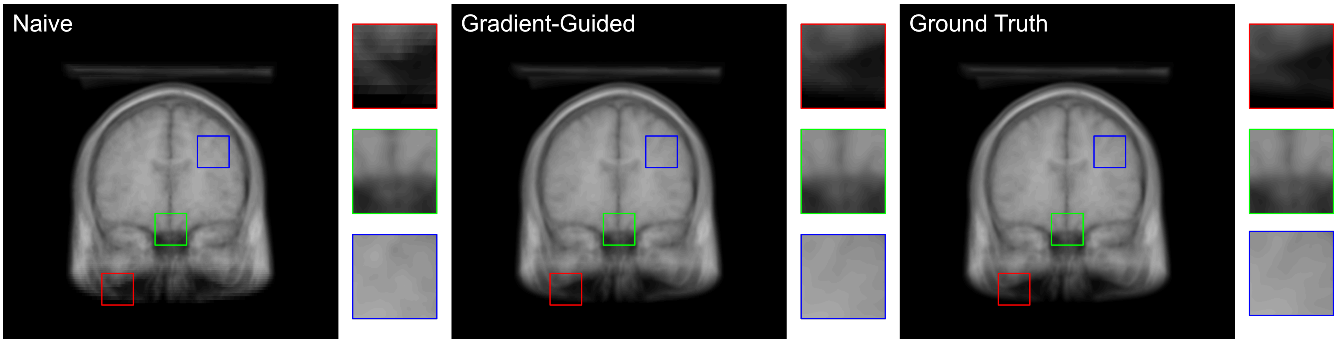

Figure 1: Left: Naive uniform sampling exhibits artifacts and loss of structural information due to undersampling. Middle: Our gradient traversal method, operating within a comparable computational budget, demonstrates improved fidelity in regions of high data variance. Right: Ground truth image for reference. All images represent single-iteration results without temporal accumulation.

Abstract

We present a novel gradient traversal algorithm for real-time volume rendering of large, unstructured, dense datasets. Our key contributions include a two-pass approach consisting of a gradient estimation pass with random offsetting and a divergent gradient traversal refinement pass, achieving significant improvements over traditional methods per traversal step. By leveraging modern GPU capabilities and maintaining uniform control flow, our method enables interactive exploration of complex, dynamic, unstructured volumetric data under real-time constraints, addressing a critical challenge in scientific visualization and medical imaging.

CCS Concepts

• **Computing methodologies** → **Rendering**.

Keywords

Volume Rendering, Live Imaging, Unstructured Data

ACM Reference Format:

Mehmet Oguz Derin and Takahiro Harada. 2024. Gradient Traversal: Accelerating Real-Time Rendering of Unstructured Volumetric Data. In *SIGGRAPH Asia 2024 Posters (SA Posters '24), December 03–06, 2024, Tokyo, Japan.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3681756.3697979

1 Introduction

Real-time volume rendering of large, unstructured, dense datasets remains a significant challenge in scientific visualization and medical imaging [Sarton et al. 2023]. Traditional approaches often rely on acceleration structures, which are challenging to update for dynamic data, provide limited acceleration when visualization requires traversal through the entire intersection segment of the ray and the volume, and can be memory-intensive [Sarton et al. 2020].

We present a novel compute-optimized gradient traversal algorithm to address these limitations and enable real-time rendering of complex volumetric data without acceleration structures. Our approach leverages modern GPU capabilities through workgroup optimization, maintaining uniform control flow to determine regions of interest collaboratively. Key advantages include:

- Elimination of memory overhead and preprocessing time
- Reduced execution divergence with expanded search space
- Efficient use of workgroup memory and atomic operations

2 Method

Our method consists of a two-pass approach: a uniform gradient estimation pass with random offsetting, followed by a divergent gradient traversal refinement pass.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SA Posters '24, December 03–06, 2024, Tokyo, Japan

© 2024 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-1138-1/24/12
https://doi.org/10.1145/3681756.3697979

Algorithm 1 Gradient Estimation and Traversal

```
$t \leftarrow$ RandomOffset()
for $i \leftarrow 1$ to num_initial_steps do
    $p \leftarrow$ eye_pos + ray $\cdot t$
    $\nabla f(p) \leftarrow$ EstimateGradient($p$)
    if $\|\nabla f(p)\| >$ threshold then
        AddROI($p$)
    end if
    $t \leftarrow t +$ step
end for
CollaborativeSortAndMergeROI()
$t \leftarrow 0$
color $\leftarrow 0$
while $t <$ max_dist and not_terminated do
    $p \leftarrow$ eye_pos + ray $\cdot t$
    if IsInROI($p$) then
        sample $\leftarrow$ SampleVolume($p$)
        AccumulateColor(sample, color)
    end if
    $t \leftarrow t +$ step
end while
return color
```

2.1 Gradient Estimation

We employ random offsetting to expand search space across spatially correlated rays. For each ray, we first generate a pseudo-random offset δ to jitter the initial sampling position. Then, the gradient at each sample point is estimated using central differences: $\nabla f(p) \approx \frac{f(p+h)-f(p-h)}{2h}$ where h is adaptively adjusted based on local volume properties.

2.2 Gradient Traversal

The refinement pass with gradient traversal focuses on regions of interest identified in the first pass. The traversal algorithm adjusts the step size based on the magnitude of the local gradient: step_size = base_step $\cdot (1 - \alpha\|\nabla f(p)\|)$ where α is a user-defined parameter controlling the adaptation strength.

3 Implementation

Our implementation leverages Vulkan in a C++20 codebase, with shaders written in GLSL. A single compute shader executes both passes, reducing host-device synchronization. We utilize GPU execution primitives such as subgroup operations, workgroup memory, and atomics to maximize utilization and minimize divergence. Key optimizations include:

- Data-independent loop structures for uniform control flow
- Workgroup collaboration using atomic operations, broadcasts, and efficient use of shared memory for gradient caching

4 Results

Experimental early evaluation shows that our method effectively balances quality and performance. The initial gradient estimation pass efficiently identifies regions of interest, while the refinement pass allocates computational resources to the most visually important areas. For example, when rendering a biological sample, our method preserves structures within a single rendering iteration while similar performance uniform stepping skips over details.

Our method exhibits strong consistency across different GPU architectures. Testing on laptop-class M1, RTX 3060, and 4060 GPUs demonstrated consistent quality improvements of 1.5x-1.7x PSNR over baseline uniform step sampling at only 1.1x compute budget.

Table 1: Comparison of PSNR values (dB) and Execution Times (ms) for `gradient traversal` and `naive` methods at different sample sizes.

Samples	Gradient Traversal		Naive	
	PSNR (dB)	Time (ms)	PSNR (dB)	Time (ms)
16	39.68	3.30	29.48	2.90
64	52.26	5.90	36.52	4.50

5 Conclusion and Future Work

We have presented a novel gradient traversal algorithm for the real-time rendering of large, unstructured volumetric datasets. Our method leverages modern GPU capabilities to increase image quality within the same computational budget, enabling interactive exploration of complex data in applications such as live medical imaging and scientific visualization.

Future work includes exploring integrating machine learning for better initialization [Weiss et al. 2022], trying to model uniform value fields, extending for iterative refinement, and dithering of collaborative workgroup region boundaries to improve perceived image quality spatiotemporally. Making the algorithm more accessible by integrating it into common open-source volume visualization packages is also an important direction [Wald et al. 2024].

Acknowledgments

We extend our sincere gratitude to OpenSPIM, the EMBL-EBI BioImage Archive, the Cell Image Library, and OpenMicroscopy for providing open-access repositories that were fundamental to our research. We also thank BrainWeb for supplying the Subject 04 Simulated T1 image used in Figure 1 1. We are grateful to Masaki Yamazaki for the initial discussions regarding the target data and application, which were instrumental in identifying the bottlenecks that led to the development of our method. We also thank Masamichi Nakamura and Hiroshi Ito for providing infrastructure support for test hardware. Part of this work was inspired during a TansaX project at the Japan Aerospace Exploration Agency (JAXA).

References

Jonathan Sarton, Nicolas Courilleau, Yannick Rémion, and Laurent Lucas. 2020. 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 (2020), 3008–3021.

Jonathan Sarton, Stefan Zellmann, Serkan Demirci, Ugur Güdükbay, Welcome Alexandre-Barff, Laurent Lucas, Jean-Michel Dischler, Stefan Wesner, and Ingo Wald. 2023. State-of-the-art in Large-Scale Volume Visualization Beyond Structured Data. In *Computer Graphics Forum*, Vol. 42. 491–515.

Ingo Wald, Stefan Zellmann, Jefferson Amstutz, Qi Wu, Kevin Griffin, Milan Jaros, and Stefan Werner. 2024. Standardized Data-Parallel Rendering Using ANARI. *arXiv preprint arXiv:2407.00179* (2024).

S. Weiss, P. Hermüller, and R. Westermann. 2022. Fast Neural Representations for Direct Volume Rendering. *Computer Graphics Forum* 41, 6 (2022), 196–211.