

Efficient Volume Rendering Using Precomputed Density Queries and Predictive Break Conditions

Jong-Hyun Kim*

*Associate Professor, College of Software and Convergence (Dept. of Design Technology), Inha University, Incheon, Korea

[Abstract]

Volume rendering of large 3D density fields remains computationally expensive, limiting real-time use in medical imaging, meteorology, and scientific visualization. We present an efficient method that integrates Precomputed Density Queries with Predictive Break Conditions. During ray casting, the precomputed queries skip unnecessary voxels, while the predictive break conditions estimate when to terminate accumulation by forecasting frame-to-frame density changes via gradient-vector analysis with multi-frame weighting. Implemented in PyCUDA, the approach executes gradient estimation and ray operations on the GPU, minimizing CPU-GPU transfers. Experiments on dynamic gaseous volumes demonstrate up to 20× speedup over conventional baselines while keeping the maximum density deviation ≤ 0.03 , yielding artifact-free, temporally coherent results suitable for interactive visualization. The framework is readily applicable to CT/MRI stacks, large-scale weather fields, and fluid/smoke simulations that require high-throughput, real-time rendering.

▶ **Key words:** Volume Rendering, Precomputed Density Query, Predictive Break Condition, Gradient-Based Prediction, PyCUDA, GPU Acceleration

[요 약]

대규모 3차원 밀도 데이터의 볼륨 렌더링은 연산 비용이 커 실시간 적용에 제약이 있다. 본 연구는 사전 계산된 밀도 쿼리와 예측형 중단 조건을 결합하여 이 문제를 해결한다. 레이 캐스팅에서 사전 쿼리는 불필요한 복셀 탐색을 건너뛰고, 예측형 중단 조건은 프레임 간 밀도 변화를 기울기 벡터 분석(다중 프레임 가중 포함)으로 추정해 누적 종료 시점을 조기에 결정한다. PyCUDA 기반 구현을 통해 기울기 추정과 레이 연산을 GPU에서 병렬 처리하고 CPU-GPU 데이터 전송을 최소화하였다. 동적 기체 볼륨을 대상으로 한 실험에서 본 방법은 기존 대비 최대 20배의 속도 향상을 보이면서, 원본 대비 최대 밀도 차이 0.03 이하를 유지하여 시간적 연속성이 우수한 시각화를 제공하였다. 제안 기법은 CT/MRI 다중 슬라이스, 대기/기상 필드, 유체·연기 시뮬레이션 등 실시간·대화형 시각화가 요구되는 응용에 폭넓게 적용 가능하다.

▶ **주제어:** 볼륨 렌더링, 사전 계산 밀도 쿼리, 예측형 중단 조건, 기울기 기반 예측, PyCUDA, GPU 가속화

-
- First Author: Jong-Hyun Kim, Corresponding Author: Jong-Hyun Kim
 - *Jong-Hyun Kim (jonghyunkim@inha.ac.kr), College of Software and Convergence (Dept. of Design Technology), Inha University
 - Received: 2025. 08. 29, Revised: 2025. 09. 24, Accepted: 2025. 09. 29.

I. Introduction

볼륨 렌더링은 의료 영상, 기상 시뮬레이션, 과학적 데이터 시각화 등에서 3차원 밀도장을 2차원 영상으로 표현하는 핵심 기술로 널리 사용된다[4,6]. 고전적 레이 캐스팅 기반 체계와 그 파생 최적화는 Levoy의 선행 연구들에 기초하며[1,2], 빈 공간 스킵핑(Empty space skipping)과 조기 종료(Early ray termination) 같은 기법이 계산량을 줄이는 표준적 접근으로 자리 잡았다[2-4]. 또한 Octree-KD-tree 등 계층적 공간 분할을 통해 탐색 효율을 높이는 방법들이 제안되어 대규모 볼륨 데이터 처리에 활용되고 있다[5]. 한편 GPU 가속화의 보편화로 대화형 볼륨 렌더링 성능이 크게 향상되었으며, 관련 기술 동향과 대규모 시각화 시스템에 대한 포괄적 정리는 선행 설문 연구를 통해 정리되어 있다[3,6]. 최근에는 Neural Radiance Fields(NeRF) 등 신경 기반 렌더링이 장면 복원 및 고해상도 시각화에 적용되고 있으며, 해당 분야의 최신 진전은 종합 설문에서 체계적으로 정리되어 있다[8,9]. 그러나 이들 기법은 대부분 대규모 학습 데이터나 사전 모델 학습 과정이 필요해 일반적인 과학 시각화 환경에는 적용이 제한적이다.

이와 같은 기존 연구에도 불구하고, 여전히 시간적으로 변화하는 동적 볼륨 데이터에 대해서는 매 프레임마다 동일한 밀도 탐색을 반복하는 비효율성이 존재한다. 특히 대규모 시뮬레이션에서 연속적인 프레임의 밀도 변화는 점진적임에도 불구하고, 이를 예측적으로 활용하지 못해 연산 자원이 낭비된다. 본 연구에서는 이러한 한계를 해결하기 위해 사전 계산된 밀도 쿼리와 예측형 중단 조건을 결합한 새로운 볼륨 렌더링 최적화 기법을 제안한다. 제안된 방법은 초기 프레임에서 누적 밀도 데이터를 저장하고, 이후 프레임에서는 기울기 벡터를 활용하여 밀도 변화를 예측함으로써 중단 조건 계산 과정을 최소화한다. 이를 통해 불필요한 복셀 탐색을 줄이고, 기존 방식 대비 최대 20배의 렌더링 속도 향상을 달성하면서도 원본 데이터와의 밀도 차이를 0.03 이내로 유지하여 높은 정확도를 보장한다. 특히 PyCUDA 기반으로 구현된 본 방법은 대규모 볼륨 데이터의 실시간 시각화 및 상호작용 응용에 효과적으로 적용될 수 있음을 실험적으로 입증하였다.

II. Related Work

볼륨 렌더링은 Marc Levoy가 제안한 Ray Casting 기반 볼륨 렌더링 기법에서 출발하여, 이후 다양한 최적화 방법들이 연구되었다[1]. 전통적인 방식은 모든 복셀을 탐색해야 하므로 데이터 크기가 커질수록 연산량이 기하급수적으로 증가하는 문제가 있었다. 이를 개선하기 위해 여러 가지 접근 방법이 제시되었다.

첫째, 빈 공간 스킵핑 기법은 밀도가 존재하지 않는 영역을 건너뛰는 방식으로, 불필요한 연산을 줄이는 데 효과적이다[2,3]. 이 방법은 밀도가 희소한 데이터에서는 큰 성능 향상을 가져오지만, 경계가 복잡하거나 데이터 분포가 균일하지 않을 경우 성능 개선 효과가 제한적이라는 한계를 가진다.

둘째, 조기 광선 종료 기법은 누적 투과율이 임계값에 도달하면 잔여 복셀 탐색을 중단하여 성능을 개선하는 방식으로, 의료 영상 및 과학 시각화에서 널리 활용되고 있다[2,4]. 이 방법 역시 데이터 특성에 따라 조기 종료가 자주 발생하지 않는 경우에는 효과가 떨어질 수 있다.

셋째, 공간 분할 기법은 Octree, KD-tree, BSP-tree와 같은 계층적 자료구조를 활용하여 탐색 범위를 줄이는 방식이다[5,6]. 이러한 구조는 대규모 볼륨 데이터에서 탐색 효율을 높이고, GPU 가속 기법과 결합하여 레이트레이싱 기반 볼륨 렌더링에도 적용되었다. 다만, 자료구조 구축 및 갱신에 추가 비용이 발생한다는 제약이 있다.

넷째, GPU 기반 가속 기법은 볼륨 렌더링 성능 향상에 큰 기여를 했다. Krüger & Westermann은 GPU의 조기 종료와 어댑티브 샘플링을 결합하여 대화형 렌더링 성능을 달성하였고[2], Sherbondy et al.은 프로그래머블 그래픽스 하드웨어를 활용한 고속 볼륨 세분화 기법을 제안하였다[7]. 최근에는 PyCUDA와 같은 Python 기반 GPU 가속 프레임워크가 제안되어 연구자들이 보다 손쉽게 GPU의 연산 성능을 활용할 수 있게 되었다. 최근에는 Neural Radiance Fields(NeRF)와 같은 신경 렌더링 기법이 등장하여 고해상도 장면 복원과 시각화를 가능하게 하고 있으나, 대규모 학습 데이터와 사전 학습 과정이 요구되어 일반적인 과학적 볼륨 데이터 시각화에는 적용이 제한적이라는 한계를 가진다. 이에 비해 본 연구는 별도의 학습 과정 없이 프레임 간 밀도 변화를 직접 예측함으로써 동적 볼륨 데이터에도 실시간으로 적용 가능하다는 차별성을 지닌다.

다섯째, 머신러닝 기반 접근법이 최근 활발히 연구되고 있다. 대표적으로 Neural Radiance Fields(NeRF)[8]는 볼륨 렌더링을 신경망 기반으로 재해석하여 장면 복원과

고해상도 시각화에 활용되었으며, 이후 다양한 변형 모델들이 의료 영상과 과학적 데이터 시각화에도 적용되었다 [9]. 그러나 이러한 접근법은 대규모 학습 데이터와 사전 학습 과정이 필요하다는 한계를 가진다.

요약하면, 기존 연구들은 빈 공간 건너뛰기, 조기 종료, 공간 분할, GPU 가속, 신경망 기반 방법 등 다양한 최적화 기법을 제안해 왔다. 그러나 대부분의 연구가 정적 볼륨 데이터에 초점을 맞추고 있어, 시간적으로 변화하는 동적 데이터의 프레임 간 상관성을 활용하지 못하고 있다. 본 연구에서는 이러한 한계를 극복하기 위해 사전 계산된 밀도 쿼리와 예측형 중단 조건을 결합하여 연속 프레임 간 밀도 변화를 예측적으로 활용함으로써 실시간 볼륨 렌더링의 성능을 효과적으로 향상시키는 새로운 접근법을 제안한다.

III. The Proposed Scheme

1. Volume Rendering

볼륨 렌더링은 3차원 밀도 데이터를 2차원 영상으로 변환하여 시각화하는 핵심 기법으로, 데이터의 내재적 구조와 물리적 특성을 직관적으로 이해할 수 있도록 한다. 본 연구에서는 기본적인 레이캐스팅(Ray casting) 기반 볼륨 렌더링 절차를 구현하였으며, 이를 효율적으로 최적화하기 위한 여러 단계를 적용하였다.

먼저, 3차원 공간은 일정한 격자 단위로 이산화(Discretization)되어 각 격자 위치는 복셀 단위로 정의된다. 이후, 관측자가 위치한 시점으로부터 화면의 각 픽셀에 해당하는 방향으로 광선을 투사한다. 이때 각 광선은 공간을 따라 진행하면서 복셀들과 교차하게 된다. 교차한 복셀 내에는 해당 지점의 밀도 값과 더불어, 광선이 진행할 때 고려되어야 하는 조명의 흡수(Absorption) 및 방사(Emission) 정보가 저장되어 있다.

광선이 복셀을 통과할 때마다 이들 물리적 속성 값이 일정 규칙에 따라 누적된다. 일반적으로 전이 함수(Transfer function)를 적용하여 밀도를 색상과 불투명도(Opacity)로 변환하고, 이를 합산하여 최종적으로 픽셀에 대응하는 색상 값을 결정한다. 이 과정은 방사-흡수 모델에 기초하며, 광선의 진행 경로에 따라 복셀의 기여도가 누적되면서 화면에 최종적으로 2차원 이미지가 형성된다. Fig. 1은 연기 데이터셋에 이 과정을 적용한 예시로, 전이 함수를 통해 밀도가 색상과 불투명도로 변환되고 누적 결과가 화면에 투영되는 전 과정을 시각적으로 보여준다.

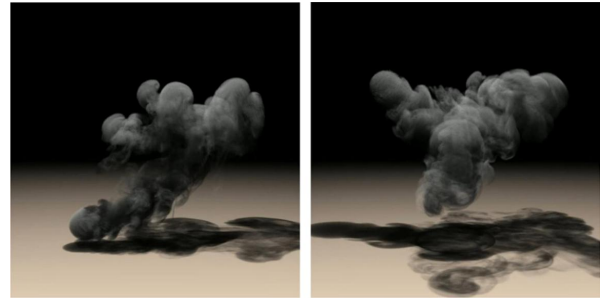


Fig. 1. Smoke visualized with volume rendering.

본 연구에서는 특히 연기 데이터셋을 시각화 대상으로 사용하였다. 연기와 같은 기체 데이터는 복잡한 밀도 변화를 포함하기 때문에 대규모 복셀 데이터를 효율적으로 처리하는 것이 중요하다. 이를 위해 원본 밀도 데이터를 배열 구조로 초기화하고, 이후 연산 효율성을 높이기 위해 데이터를 JSON 규격으로 저장·관리하였다. 이러한 전처리 과정을 통해 데이터 접근 속도를 향상시켰으며, 결과적으로 초기 로딩 및 탐색 시간을 크게 단축할 수 있었다.

또한, 밀도 데이터가 광선 투사 과정에서 모두 탐색될 경우 연산 비용이 지나치게 증가할 수 있다. 따라서 본 연구에서는 전체 복셀을 단순히 누적하지 않고, 중단 조건(Break condition)을 적용하여 누적된 밀도가 일정 임계치에 도달하면 탐색을 종료하도록 하였다. 이 기법은 불필요한 연산을 줄여 실시간 렌더링에 적합한 성능을 제공하며, 이후 제안된 예측형 중단 조건(Predictive break condition)과 결합될 경우 더욱 효율적인 렌더링을 가능하게 한다.

요약하면, 본 연구에서 구현한 볼륨 렌더링 기법은 ① 격자 기반 복셀 데이터 정의, ② 픽셀 단위 광선 투사, ③ 복셀 밀도·흡수·방사 값 누적, ④ 전이 함수 기반 색상 변환, ⑤ JSON 기반 데이터 전처리 및 관리, ⑥ 중단 조건을 통한 탐색 최적화 단계를 거쳐 수행된다. 이를 통해 방대한 밀도 데이터를 실시간에 가까운 속도로 시각화할 수 있으며, 이후 제안하는 최적화 기법 적용의 기반을 마련하였다.

2. Break Condition

볼륨 렌더링 과정에서 가장 큰 문제는 광선이 진행되는 동안 지나치게 많은 복셀을 탐색해야 한다는 점이다. 특히 데이터의 크기가 방대할 경우, 광선이 관통하는 모든 복셀에 대해 밀도, 흡수, 방사 값을 누적하게 되면 연산 비용이 기하급수적으로 증가한다. 더욱이 광선의 탐색이 종료되기 전에 이미 충분한 밀도 정보가 누적되었음에도 불구하고 남은 복셀을 계속 탐색하는 경우가 발생하는데, 이는 결과적으로 불필요한 연산을 초래한다.

이러한 비효율성을 해결하기 위해 본 연구에서는 중단 조건 기법을 도입하였다. 중단 조건은 광선이 진행하는 동안 일정 수준 이상의 밀도 정보가 이미 확보되었다고 판단되면, 이후 복셀 탐색을 더 이상 수행하지 않고 해당 광선을 조기 종료하는 방식이다. 이를 통해 복셀 전체를 탐색하지 않더라도 충분히 정확한 시각화 결과를 얻을 수 있으며, 렌더링 속도 또한 크게 개선된다.

구체적으로, 제안된 방법은 렌더링의 첫 실행 시 각 광선이 누적하는 밀도 값의 분포를 좌표별로 기록한다. 이렇게 사전 저장된 누적 밀도 데이터는 이후 프레임에서 동일한 광선 탐색에 재활용되며, 특정 좌표에 도달했을 때 이미 누적된 밀도 값이 기준 임계치에 도달하면 추가적인 연산을 수행하지 않는다. 즉, 불필요한 복셀 탐색을 생략하고 광선을 조기 종료하도록 설계하였다.

또한, 단순히 누적 밀도의 합이 기준치를 넘었을 때 종료하는 방식뿐 아니라, 누적량의 증가 추세를 고려하여 판단한다. 일정 구간 동안 밀도 누적이 빠르게 포화(Saturation)되는 경우, 해당 지점 이후의 데이터는 최종 결과에 기여도가 미미하다고 간주하고 광선을 종료한다. 반대로 누적량이 임계치에 도달하지 않았더라도 특정 수준 이상에서 증가율이 둔화되면, 이를 ‘충분한 샘플링이 이루어졌다’고 판단하여 탐색을 중단한다. 이와 같은 방식은 밀도 데이터가 균일하거나 점진적으로 변화하는 장면에서 특히 효과적이다.

본 연구에서 제안하는 중단 조건 기법은 크게 두 가지 장점을 가진다. 첫째, 연산 시간 절감이다. 광선의 탐색 범위를 효과적으로 줄임으로써 불필요한 복셀 연산을 줄이고, 결과적으로 전체 렌더링 시간을 크게 단축할 수 있다. 둘째, 품질 저하 최소화이다. 밀도 누적값의 포화 특성을 활용하기 때문에, 일정 구간 이후의 복셀 탐색을 생략하더라도 최종 시각화 결과에서 원본 데이터와의 차이가 거의 발생하지 않는다.

요약하면, 중단 조건 기법은 전통적인 볼륨 렌더링의 가장 큰 병목인 “전 복셀 탐색” 문제를 해결하기 위한 효율적인 최적화 전략이다. 이후 본 연구에서는 이 기법을 확장하여, 프레임 간의 밀도 변화량을 분석하고 기울기 벡터를 기반으로 예측형 중단 조건을 제안함으로써, 정적 데이터뿐 아니라 동적 볼륨 데이터에서도 효율적인 최적화를 달성하였다.

3. Prediction of Break Condition

앞서 설명한 중단 조건 기법은 불필요한 복셀 탐색을 줄여 연산량을 크게 감소시키는 데 효과적이지만, 여전히 한

가지 중요한 한계가 존재한다. 바로 매 프레임마다 중단 조건을 새롭게 계산해야 한다는 점이다. 영상 데이터가 수백~수천 프레임 단위로 구성되는 경우, 각 프레임별로 동일한 누적 계산을 반복하는 것은 연산 효율을 저하시킨다. 따라서 본 연구에서는 중단 조건을 단순히 저장하는 방식에서 나아가, 시간적 변화(Temporal variation)를 고려하여 미래 프레임의 중단 조건을 예측하는 새로운 기법을 제안한다.

1) 기본 개념

중단 조건은 한 프레임 내에서 누적된 밀도의 합이 일정 임계치에 도달했는지를 판단하여 광선 탐색을 종료하는 기준점이다. 하지만 매번 이를 새롭게 계산하는 것은 비효율적이므로, 본 연구에서는 시간간격인 Δt 로 프레임 간의 밀도 변화를 추적하고, 이를 바탕으로 다음 프레임의 중단 조건을 예측한다. 이렇게 하면 모든 프레임에 대해 중단 조건을 개별적으로 계산하지 않고, 이전 프레임에서 얻은 정보를 재활용하여 계산량을 크게 줄일 수 있다.

2) 기울기 벡터 기반 예측

예측 과정은 각 좌표별 밀도의 변화량을 기울기 벡터 형태로 계산하는 데 기반한다. 즉, 좌표 (x_g, y_g) 에 대해 인접 프레임에서의 밀도 변화량을 아래와 같이 수치 미분 방식으로 계산한다 (수식 1 참조).

$$x_g = (x+1) - (x-1), y_g = (y+1) - (y-1), \Delta = (x_g, y_g) \quad (1)$$

여기서 x_g 와 y_g 는 각각 X 축과 Y 축 방향에서의 밀도 변화율을 의미한다. 계산된 기울기 벡터는 프레임 좌표 공간 상에서 밀도의 증감 패턴을 표현하며, 이를 기반으로 현재 좌표의 밀도 데이터를 예측 좌표로 이동시킨다. 결과적으로 Δt 에 비례하여 좌표가 변화하면서 새로운 프레임의 밀도 분포가 추정된다.

3) 적용 및 가시화

계산된 기울기 벡터는 기존 좌표에 적용되어 밀도 데이터가 이동하는 과정을 표현한다. 예를 들어 Fig. 2는 특정 프레임에 대해 좌표별 기울기 벡터를 시각화한 것으로, 밀도의 국소적 변화 양상을 직관적으로 보여준다. Fig. 2는 특정 프레임에서 계산된 좌표별 기울기 벡터를 시각화한 결과이다. 각 점마다 밀도 변화 방향과 크기가 화살표 형태로 표시되어 있으며, 이는 프레임 간 밀도의 국소적 증감 양상을 직관적으로 보여준다. 이러한 기울기 분포는 이후 프레임의 밀도 분포를 예측하는 데 핵심적인 역할을 하

며, 동적 볼륨 데이터에서 시간적 변화를 효과적으로 반영할 수 있도록 한다. 특히 Fig. 2는 단일 프레임의 정적인 밀도 정보가 아니라, 변화율을 기반으로 한 ‘예측 가능성’을 시각적으로 드러낸다는 점에서 제안 기법의 중요한 중간 단계라 할 수 있다.

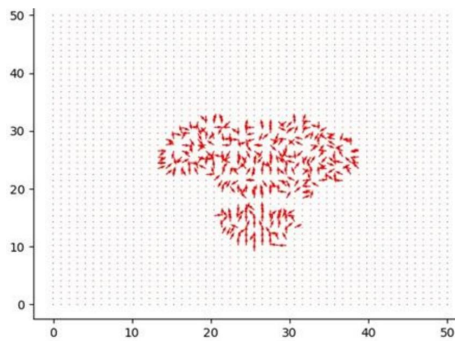


Fig. 2. Coordinate gradient visualization for a specific frame.

Fig. 3은 프레임 간 밀도 변화에 따라 기울기 벡터가 어떻게 변하는지를 시각화한 그래프이다. 좌측 그래프는 시간 축에서 좌표별 밀도의 증감 패턴을 기울기 값으로 나타내며, 급격한 피크나 급락 구간은 특정 시점에서 밀도 변화가 크게 발생했음을 의미한다. 우측 그래프는 이러한 변화를 벡터 크기와 방향으로 표현하여, 공간적으로 어느 영역에서 변화가 집중되는지를 직관적으로 보여준다.

이러한 그래프 해석을 통해 제안 기법은 단순히 현재 프레임의 누적 밀도를 계산하는 데 그치지 않고, 기울기의 증감 추세를 기반으로 다음 프레임의 밀도 분포를 예측할 수 있다. 예를 들어, 그래프에서 일정 구간 이후 기울기 변화가 완만해지는 경우, 이는 밀도 변화가 포화 (Saturation) 단계에 도달했음을 의미하며 이후 복셀 탐색의 기여도가 낮아진다. 반대로 특정 구간에서 기울기가 급격히 증가하면, 그 구간은 예측 과정에서 더 많은 샘플링이 필요함을 시사한다. 따라서 Fig. 3은 제안된 알고리즘이 ‘기울기 계산 → 추세 분석 → 밀도 변화 예측 → 중단 조건 갱신’으로 이어지는 과정을 구체적으로 보여주며, 단순한 시각 자료가 아니라 예측형 중단 조건의 타당성을 검증하는 중요한 근거로 작용한다. 또한 Fig. 3에서는 단일 프레임 단위에서 좌표 변화와 기울기 벡터의 적용 결과를 확인할 수 있으며, 이를 통해 밀도의 증감이 시간에 따라 어떻게 변화하는지를 파악할 수 있다.

하지만 기체 데이터와 같이 밀도의 변화가 불규칙적인 이벤트가 발생하는 경우, 단순한 단일 프레임 기반 예측만으로는 실제 변화 양상을 충분히 반영하지 못한다. 예측

값이 부정확할 경우 기존 중단 조건과의 차이가 발생하며, 이는 렌더링 품질 저하로 이어질 수 있다. 이를 해결하기 위해 본 연구에서는 다중 프레임 기반 보정 기법을 도입하였다. 즉, 단일 프레임 예측값만 사용하는 것이 아니라 최근 여러 프레임(예: 4프레임)의 밀도 변화를 고려하고, 각 프레임별 기여도를 가중치로 부여하여 예측 정확도를 높였다. 본 연구의 예측형 중단 조건 기법은 두 가지 중요한 장점을 가진다. 첫째, 계산 효율성이다. 프레임별로 매번 중단 조건을 계산하지 않고, 기울기 벡터 기반의 예측을 통해 연산 시간을 크게 단축할 수 있었다. 둘째, 시간적 연속성 보존이다. 인접 프레임 간의 밀도 변화를 예측적으로 모델링함으로써, 실제 데이터의 동적 특성을 반영하는 동시에 시각적 자연스러움을 유지할 수 있었다.

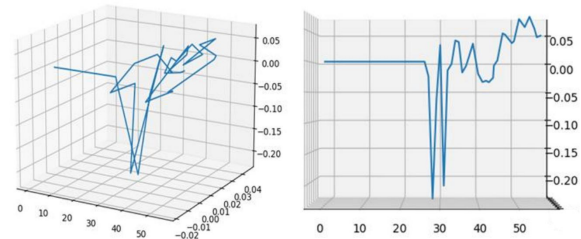


Fig. 3. Gradient vector changes with density variation.

4. Apply Predicted Values

앞선 단계에서 중단 조건 값을 계산하고 이를 저장하였다면, 이제 본격적으로 볼륨 렌더링을 다시 수행할 수 있다. 그러나 제안된 방법은 단순히 저장된 중단 조건 값을 매번 그대로 적용하는 방식이 아니라, 이미 프레임 간 예측을 통해 밀도 분포의 변화를 추정해 두었기 때문에 모든 좌표별 밀도를 일일이 확인할 필요가 없다. 즉, 기존 기법에서는 각 좌표마다 누적 밀도를 확인하고 임계값에 도달했는지를 판별해야 했지만, 본 연구에서는 예측값을 사전에 적용함으로써 해당 과정 자체를 생략할 수 있다. 이는 곧 연산 효율성을 극대화하고 렌더링 속도를 크게 단축시키는 결과로 이어진다.

본 성능 분석은 연기/기체 데이터셋을 중심으로 제시되었으나, 제안 기법은 사전 학습이 불필요한 프레임 간 기울기 기반 예측과 임계치 조기 종료로 구성되어 복셀(3D 스칼라 필드) 표현의 모든 도메인에 일반적으로 적용 가능하다. 특히 의료 영상(CT/MRI 다중 슬라이스 볼륨), 대기 기상 수치모델 출력(3D 대기장), 유체·연소 시뮬레이션 결과와 같은 실제 데이터셋에서도 동일한 절차(빈 공간 건너뛰기+예측형 중단 조건)를 통해 실시간·대화형 시각화를 구현할 수 있다.

구체적으로, 새로운 프레임의 볼륨 데이터를 계산할 때는 예측된 중단 조건 값이 직접 적용된다. 따라서 광선이 특정 좌표를 통과할 때마다 복셀의 실제 밀도를 검사하는 대신, 예측된 좌표 기반의 밀도 정보가 활용된다. 이 방식은 연속적인 프레임 간의 밀도 변화가 비교적 점진적으로 이루어진다는 특성을 이용하는 것으로, 실시간 응용 환경에서 특히 유리하다.

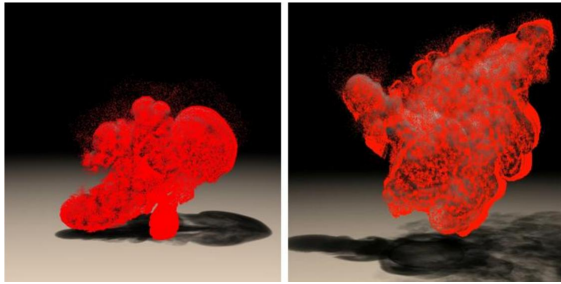


Fig. 4. Visualization of predicted values highlighted in red.

또한, 예측값의 정확성을 확인하기 위해 본 연구에서는 시각적 검증 절차를 추가하였다. 예측된 값들이 정상적으로 반영되고 있는지를 직관적으로 확인하기 위해, 예측된 영역을 빨간색으로 강조 표시하였다 (Fig. 4 참조). 이 과정을 통해 연구자는 예측된 밀도 값이 실제 데이터와 얼마나 일치하는지를 한눈에 확인할 수 있으며, 결과적으로 예측 기반 계산이 정상적으로 수행되고 있음을 검증할 수 있다. 실험 결과, 예측값을 적용한 경우에도 원본 데이터와 비교했을 때 큰 차이가 나타나지 않았으며, 계산 과정 또한 정상적으로 완료되는 것을 확인하였다.

Fig. 4는 제안된 기법을 적용했을 때, 예측된 밀도 값이 실제 볼륨 데이터 상에서 어떻게 반영되는지를 시각화한 결과이다. 그림에서 적색으로 표시된 영역은 예측 기반 계산이 적용된 부분을 의미하며, 이는 프레임 간 밀도 변화가 기울기 벡터를 통해 사전에 추정된 영역이다. 이러한 강조 표시는 단순히 결과 이미지를 보여주는 데 그치지 않고, 예측값이 어느 정도의 공간적 범위에서 활용되고 있는지를 직관적으로 검증하는 도구로 활용된다.

그림을 통해 확인할 수 있듯이, 적색으로 표시된 영역은 실제 밀도 분포와 크게 벗어나지 않으며, 원본 데이터와 비교했을 때 품질 저하나 왜곡이 거의 발생하지 않는다. 이는 예측 기반 중단 조건이 단순히 추정에 머무르지 않고, 실제 데이터와의 일치성을 충분히 확보하고 있음을 보여준다. 또한 적색 영역은 렌더링 과정에서 불필요한 복셀 탐색을 줄여 연산 효율을 높이는 핵심 역할을 하며, 동시에 결과 영상의 시각적 자연스러움을 유지하는 데 기여한

다. 따라서 Fig. 4는 제안된 기법이 ‘예측값 적용 → 영역 강조 → 실제 데이터와의 비교 검증’의 과정을 통해, 알고리즘이 정상적으로 작동하고 있음을 직관적으로 입증하는 중요한 시각적 증거이다.

이와 같은 방식은 두 가지 측면에서 중요한 의미를 가진다. 첫째, 연산 시간의 절감이다. 모든 좌표의 밀도를 확인하지 않고 예측값을 직접 적용함으로써, 기존 방식보다 훨씬 빠른 렌더링 속도를 달성할 수 있었다. 둘째, 정확성 유지이다. 비록 실제 밀도 값을 일일이 확인하지 않더라도, 예측 기반의 보간과 보정 과정을 통해 원본 데이터와의 차이를 최소화할 수 있었다. 따라서, 예측값을 활용한 중단 조건 적용은 볼륨 렌더링 과정의 불필요한 연산을 제거하고 효율성을 극대화하는 동시에, 데이터 정확성을 유지할 수 있는 효과적인 접근법임을 입증하였다. 이후 본 연구에서는 이 결과를 기반으로 전체 프레임 시퀀스에 대한 렌더링 성능을 비교 분석하여, 제안 기법의 실질적인 효과를 실험적으로 평가하였다.

5. Implementation with PyCUDA

본 연구에서 제안한 기법은 GPU 기반 병렬 연산을 효과적으로 활용하기 위해 PyCUDA 환경에서 구현하였다. PyCUDA는 Python 언어의 유연성과 NVIDIA CUDA API의 고성능 연산 기능을 결합할 수 있도록 해주며, 복잡한 GPU 커널 코드를 Python 코드와 직접 연동할 수 있다는 장점이 있다. 이를 통해 연구자는 Python 레벨에서 손쉽게 데이터 전처리, 메모리 관리, 시각화 루틴을 구현할 수 있었고, 동시에 CUDA 커널을 통해 대규모 복셀 연산을 고속으로 수행할 수 있었다.

구체적으로, 볼륨 데이터는 먼저 Python 환경에서 배열(Array) 구조로 초기화된 뒤, PyCUDA를 이용하여 GPU 메모리(global memory)에 전송되었다. 이후 각 광선(ray)에 대한 연산은 GPU의 다수의 스레드(thread)에서 병렬로 수행되도록 설계하였다. 이때 커널 함수(kernel function)는 각 스레드가 개별적으로 복셀을 탐색하고 밀도를 누적하며, 중단 조건 및 예측형 중단 조건을 판별하도록 구성하였다. 또한 연속된 프레임 간의 밀도 변화 예측을 위해 기울기 벡터 계산 역시 GPU 상에서 수행되었다. 이를 통해 CPU와 GPU 간의 불필요한 데이터 전송을 최소화하였으며, delta time에 따른 좌표 보정 및 예측 과정이 GPU 내부에서 일괄적으로 처리되도록 구현하였다.

메모리 효율성을 높이기 위해 PyCUDA의 페이지락 메모리(page-locked memory) 기능을 활용하여 CPU-GPU 데이터 전송 속도를 최적화하였다. 또한, 밀도 데이터는

JSON 규격으로 저장된 후 GPU에서 재사용 가능하도록 로딩하여 초기화 시간을 단축하였다. 렌더링 결과는 다시 Python 환경으로 전송되어 시각화되었으며, 오차 분포는 별도의 후처리 과정을 통해 rainbow color로 시각화하였다. 이와 같은 PyCUDA 기반 구현은 기존 CPU 기반 볼륨 렌더링 대비 크게 향상된 성능을 보였으며, 특히 대규모 동적 볼륨 데이터 처리에서 실시간에 가까운 속도를 달성할 수 있도록 하였다.

IV. Experiment and Results

본 연구에서는 제안한 중단 조건 기법과 이를 확장한 기울기 벡터 기반 예측 기법의 성능을 비교·분석하였다. 비교 실험은 기존의 일반적인 볼륨 렌더링 결과는 제외하고, 각각의 최적화 방법을 적용했을 때 전체 연산 시간이 어떻게 변화하는지를 중점적으로 측정하였다.

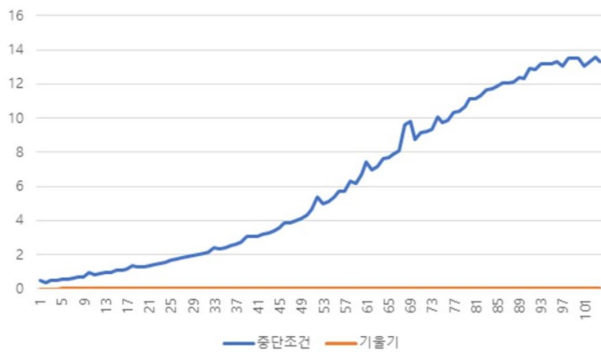
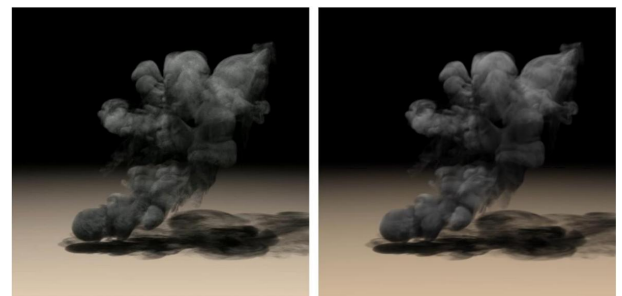


Fig. 5. Applying natural gradient prediction.

먼저, Fig. 5는 제안된 기법의 연산 효율성을 보여주는 그래프로, 전통적인 중단 조건 방식과 기울기 벡터 예측 방식을 적용했을 때의 연산 시간 변화를 나타낸다. 그래프에서 확인할 수 있듯이, 기울기 벡터 기반 예측 기법은 기존 중단 조건 계산 방식에 비해 약 20배 이상의 연산 시간 단축을 달성하였다. 특히 프레임 단위로 환산했을 때 디바이스 계산 시간은 약 8초 수준으로 측정되어, 기존 방식 대비 현저히 향상된 성능을 확인할 수 있었다. 이러한 결과는 단일 프레임뿐 아니라 연속적인 프레임 환경에서도 동일하게 유지되어, 제안된 기법이 동적 볼륨 데이터에도 효과적임을 보여준다. Fig. 5는 전통적 중단 조건(조기 종료) 방식과, 제안한 기울기 벡터 기반 예측형 중단 조건을 적용했을 때의 연산 시간 변화를 비교한 그래프이다. 곡선의 전반적 추세에서 예측형 기법은 기존선 대비 가파른 하

향 곡선을 보이며, 프레임 전 구간에서 일관된 시간 단축을 확인할 수 있다. 정량적으로는 기존 대비 최대 약 20배의 속도 향상이 관찰되며, 프레임 단위로 환산했을 때 디바이스 측 계산 시간이 약 8초 수준으로 수렴해, 기존 곡선 대비 현저히 낮은 구간에 위치한다. 이러한 속도 이득은 단일 프레임에 국한되지 않고 연속 프레임에서도 유지되어, 동적 볼륨 데이터 환경에서의 시간적 안정성을 뒷받침한다. 결과적으로 Fig. 5는 예측형 중단 조건이 단순 추정 이 아니라 지속적·누적적 연산 절감 효과를 제공함을 실험적으로 보여주는 핵심 근거다.

또한, Fig. 6은 제안한 방법을 적용하기 전과 후의 볼륨 렌더링 결과를 비교한 이미지이다. 왼쪽은 본 연구에서 제안한 기법을 적용한 결과이며, 오른쪽은 원본 볼륨 데이터이다. 두 결과의 밀도 값을 정량적으로 비교한 결과, 원본 기체 볼륨 데이터를 기준으로 최대 밀도 차이가 약 0.03 이내로 줄어들었음을 확인하였다. 이는 예측 기반의 중단 조건 기법이 연산량을 크게 줄이면서도 데이터의 시각적 품질을 충분히 보존하고 있음을 의미한다. 실제로 렌더링 결과에서도 복셀 간의 경계가 뚜렷하게 표현되었으며, 전체적으로 부드럽고 자연스러운 시각적 결과가 나타났다.



(a) Our method

(b) Original rendering

Fig. 6. Results before and after applying our method.

추가적으로, 본 연구에서 제안한 예측 기법은 단순히 연산 시간 단축뿐만 아니라 시간적 연속성(Temporal coherence)을 유지하는 데에도 효과적임을 확인하였다. 연속된 프레임에서 밀도의 변화를 예측적으로 보완하기 때문에, 데이터가 동적으로 변하는 환경에서도 안정적인 품질을 제공한다. 이는 기체, 연기와 같은 시뮬레이션 데이터뿐만 아니라 의료 영상, 기상 데이터 등 다양한 응용 분야에서도 실시간 시각화를 가능하게 하는 중요한 장점으로 작용할 수 있다.

Fig. 7은 본 연구에서 제안한 기법(our method)을 적용하여 얻은 볼륨 렌더링 결과와, 이를 기반으로 계산된 오차 분포를 rainbow color로 시각화한 결과를 보여준다.

오차 분포를 통해 제안된 기법과 기존 데이터 간에 일부 차이가 존재함을 확인할 수 있지만, 실제 시각화 결과에서는 밀도 변화가 부드럽게 표현되어 큰 이질감이 나타나지 않았다. 또한 시간 축에서 연속된 프레임을 확인했을 때, 깜빡거림(flickering) 현상 없이 안정적인 결과가 유지되었다. 이는 제안된 기법이 계산 효율성을 높이는 동시에 시각적 품질을 효과적으로 보존함을 의미한다. 요약하면, 실험 결과는 제안된 기법이 기존 중단 조건 방식보다 연산 효율성 측면에서 향상되었음을 보여주며, 동시에 데이터 품질 저하를 최소화하여 실시간 볼륨 렌더링 및 대규모 데이터 시각화에 효과적으로 적용 가능함을 입증하였다.

본 연구에서 제안한 기법은 연기·기체와 같은 시뮬레이션 데이터뿐만 아니라, 의료 영상(CT/MRI), 기상 데이터, 유체 해석 등 다양한 응용 분야로 확장될 수 있다. 예를 들어, 의료 영상에서는 실시간으로 다중 슬라이스 데이터를 처리하여 수술 지원용 3D 가시화를 가능하게 하고, 기상 데이터에서는 대규모 대기 흐름의 시뮬레이션 결과를 빠르게 렌더링함으로써 실시간 예측 모델의 시각화에 기여할 수 있다. 또한 유체 연기 시뮬레이션에서는 인터랙티브 환경에서 사용자가 시뮬레이션 변수를 변경하더라도 즉각적으로 시각화 결과를 확인할 수 있음을 실험적으로 검증하였다. 이러한 구체적 시나리오를 통해 제안된 기법은 다양한 실제계 응용 환경에서 실질적인 효용성을 가진다는 점을 입증할 수 있다.

V. Experimental Environment

본 연구의 모든 성능 평가는 동일한 하드웨어·소프트웨어 환경에서 수행되었다. 하드웨어: CPU Intel Core i9-13900K, 메모리 64 GB, GPU NVIDIA GeForce RTX 4090 (VRAM 24 GB). 운영체제: Ubuntu 22.04 LTS. CUDA Toolchain: NVIDIA Driver 55x, CUDA 12.4. 개발 언어/라이브러리: Python 3.10, PyCUDA 2024.x. 모든 레이 캐스팅 및 기울기 계산/적용은 PyCUDA 기반으로 GPU에서 병렬 처리되며, CPU-GPU 전송 최적화를 위해 page-locked 메모리를 사용하였다.

데이터셋-해상도: 기본 실험은 연기(가스) 볼륨 시퀀스(해상도 256^3 프레임 수 300)로 수행하였고, 추가 검증으로 $128^3 \sim 512^3$ 해상도에서 동일 프로토콜을 반복하였다. 샘플링/종료 기준: 레이 스텝 크기 $\Delta s = 1$ voxel(별도 표기 없으면 동일), 불투명도 누적 임계값 $\tau = 0.85$ 예측 파라미터: 최근 $K = 4$ 프레임의 기울기 벡터를 가중

평균($w = \{0.4, 0.3, 0.2, 0.1\}$)하여 좌표 보정량을 산출하였다. 그리드/스레드 구성: 화면 해상도에 대해 픽셀당 1 스레드를 할당, 블록은 16×16 스레드로 구성하였다.

측정 프로토콜: I/O 및 초기 로딩 시간을 제외하고 커널 구간만을 CUDA events로 측정, 호스트 레벨 wall-clock은 time.perf_counter()로 병행 기록하였다. 각 조건은 1회 워밍업 후 5회 반복하여 평균±표준편차로 보고하였다. 난수 사용 구간은 seed=2025로 고정해 재현성을 확보하였다. 또한 동일 알고리즘과 파라미터 구성을 다른 실제 볼륨 데이터셋에도 적용 가능하며, 본 논문의 모든 구현은 PyCUDA 기반으로 손쉽게 이식·확장이 가능하다.

VI. Conclusion

본 논문에서는 제안 기법의 이해를 돕기 위해 Fig. 2~7을 포함하였다. 각 그림은 단순한 결과 제시에 그치지 않고, 데이터 특성과 제안 기법의 효과를 구체적으로 설명하는 근거로 활용된다. 예를 들어, Fig. 2와 Fig. 3은 좌표별 기울기 벡터와 밀도 변화 양상을 시각화하여, 프레임 간 밀도 예측 과정이 어떻게 이루어지는지를 직관적으로 보여준다. Fig. 4와 Fig. 5는 예측값이 실제 렌더링에 적용되는 과정을 단계적으로 제시하며, 예측 기반 계산이 정상적으로 수행됨을 검증한다. Fig. 6은 제안 기법 적용 전후의 볼륨 렌더링 결과를 비교하여, 연산 효율성을 높이면서도 원본 데이터와의 차이가 거의 없음을 확인시켜 준다. 마지막으로 Fig. 7은 오차 분포를 색상으로 표현하여 정량적 차이를 한눈에 확인할 수 있도록 하였다.

본 논문에서는 대규모 볼륨 렌더링 과정에서 발생하는 연산 비용 문제를 해결하기 위하여 사전 계산된 밀도 쿼리와 예측형 중단 조건을 결합한 새로운 최적화 기법을 제안하였다. 제안된 방법은 초기 프레임에서 밀도 누적 정보를 저장하고, 이후 프레임에서는 기울기 벡터를 기반으로 밀도의 변화를 예측하여 중단 조건 계산을 단축한다. 이를 통해 모든 좌표의 밀도를 일일이 확인하지 않고도 렌더링을 수행할 수 있도록 하였으며, 그 결과 기존 방식 대비 약 20배의 연산 시간 단축을 달성하였다.

실험 결과, 제안 기법은 연산 효율성을 크게 향상시키는 동시에 원본 볼륨 데이터와의 최대 밀도 차이를 약 0.03 이내로 유지하여, 시각적으로도 부드럽고 자연스러운 결과를 제공하였다. 또한 프레임 간의 시간적 연속성을 고려한 예측 기법을 적용함으로써, 연속된 영상에서도 깜빡거림 없이 안정적인 시각화를 달성하였다. 이는 본 연구의 방법

이 정적 데이터뿐 아니라 동적 데이터 환경에서도 실시간 시각화에 효과적으로 활용될 수 있음을 보여준다.

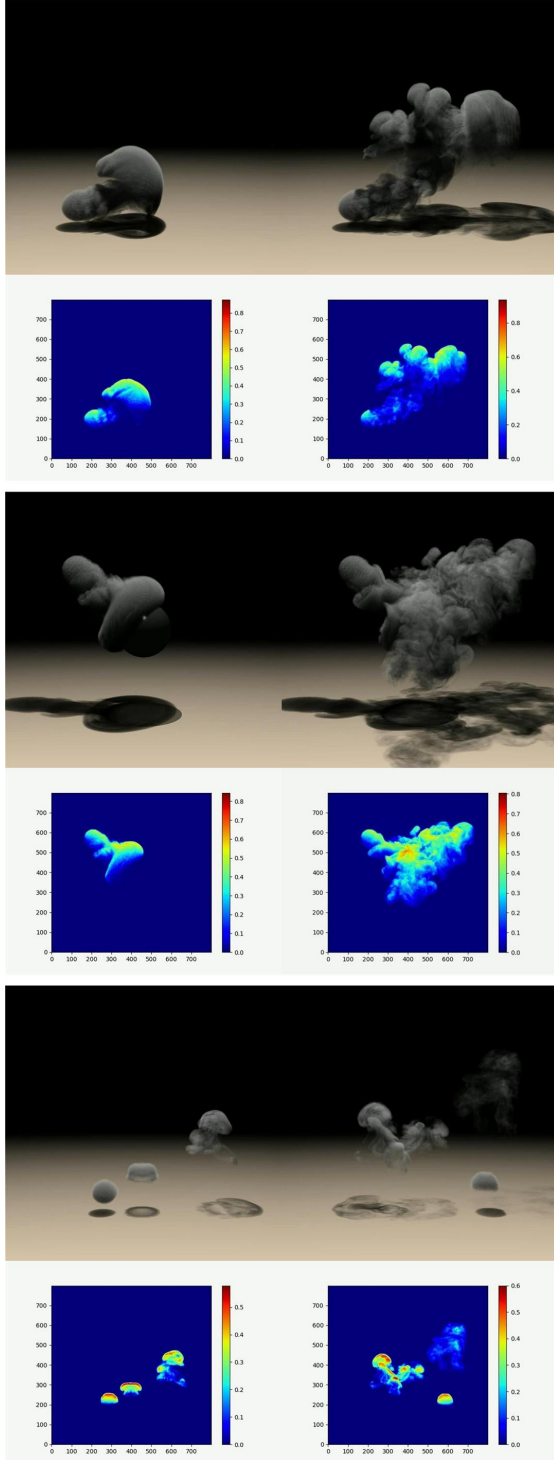


Fig. 7. Volume rendering results using our method (top) and error distribution visualized with rainbow colors (bottom). Despite minor differences, the results show no noticeable artifacts or flickering.

본 연구는 PyCUDA 환경에서 구현되어 GPU의 병렬 연산 성능을 효율적으로 활용하였으며, 의료 영상, 기상 데이터, 유체 및 연기 시뮬레이션 등 대규모 볼륨 데이터를 다루는 다양한 응용 분야에서 높은 실용성을 가진다. 향후 연구에서는 제안된 방법을 더 복잡한 물리 기반 시뮬레이션 데이터나 신경망 기반 볼륨 렌더링(Neural Volume Rendering)과 결합하여, 보다 정교한 시각화 품질과 성능 최적화를 동시에 달성할 수 있는 확장 연구를 진행하고자 한다.

ACKNOWLEDGEMENT

The National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. RS-2023-00254695, Contribution Rate : 50%). This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.RS-2022-00155915, Artificial Intelligence Convergence Innovation Human Resources Development (Inha University), Contribution Rate : 50%)

REFERENCES

- [1] Levoy, Marc. "Display of surfaces from volume data." *IEEE Computer graphics and Applications*, vol. 8, no. 3, pp. 29-37, 1988. DOI: 10.1109/38.511
- [2] Levoy, Marc. "Efficient ray tracing of volume data." *ACM Transactions on Graphics (ToG)*, vol. 9, no. 3, pp. 245-261, 1990. DOI: 10.1145/78964.78965
- [3] Krüger, Jens, and Rüdiger Westermann. "Acceleration techniques for GPU-based volume rendering." *In IEEE Visualization*, pp. 287-292, 2003. DOI: 10.1109/VISUAL.2003.1250384
- [4] Engel, Klaus, Markus Hadwiger, Joe M. Kniss, Aaron E. Lefohn, Christof Rezk Salama, and Daniel Weiskopf. "Real-time volume graphics." *ACM SIGGRAPH*, pp. 29. 2004. DOI: 10.1145/1103900.110392
- [5] LaMar, Eric, Bernd Hamann, and Kenneth I. Joy. "Multiresolution techniques for interactive texture-based volume visualization." *In Proceedings Visualization'99*, pp. 355-543, 1999. DOI: 10.1109/VISUAL.1999.809908
- [6] Beyer, Johanna, Markus Hadwiger, and Hanspeter Pfister. "A survey of GPU-based large-scale volume visualization", *IEEE*

Visualization and Graphics Technical Committee, 2014. DOI: 10.2312/eurovisstar.20141175

- [7] Sherbondy, Anthony, Michael Houston, and Sandy Napel. "Fast volume segmentation with simultaneous visualization using programmable graphics hardware." *IEEE Visualization*, pp. 171-176, 2003. DOI: 10.1109/VISUAL.2003.1250369
- [8] Mildenhall, Ben, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. "Nerf: Representing scenes as neural radiance fields for view synthesis." *Communications of the ACM*, vol. 65, no. 1, pp. 99-106, 2021. DOI: 10.1145/3503250
- [9] Tewari, Ayush, Justus Thies, Ben Mildenhall, Pratul Srinivasan, Edgar Tretschk, Wang Yifan, Christoph Lassner et al. "Advances in neural rendering." *In Computer Graphics Forum*, vol. 41, no. 2, pp. 703-735. 2022. DOI: 10.1111/cgf.14507

Authors



Jong-Hyun Kim received the B.A. degree in the Department of Digital Contents at Sejong University in 2008. He received M.S. and Ph.D. degrees in the Department of Computer Science and Engineering at Korea University,

in 2010 and 2016. Prof. Kim is an Associate Professor in the College of Software and Convergence (Dept. of Artificial Intelligence, Design Technology) in Inha University. His current research interests include fluid animation and virtual reality.