

미리 계산된 밀도 쿼리 가속화를 이용한 PyCUDA 기반의 최적화된 볼륨 렌더링

이수호^o, 김종현^{*}

^o강남대학교 소프트웨어응용학부,

^{*}강남대학교 소프트웨어응용학부

e-mail: jonghyunkim@kangnam.ac.kr

Optimized Volume Rendering Based on PyCUDA with Precomputed Density Query Acceleration

SooHo Lee^o, JongHyun Kim^{*}

^oSchool of Software Application, Kangnam University,

^{*}School of Software Application, Kangnam University

● 요약 ●

볼륨 렌더링은 3D 밀도 데이터를 가시화 할 때 활용되는 기술로써 이 알고리즘에서 중요한 것은 렌더링 시간 단축이며, 본 논문에서는 이 계산시간을 효율적으로 개선시킬 수 있는 방법을 제시한다. 렌더링의 처리 시간은 탐색하는 횟수에 따라 결과 차이가 발생하지만, 탐색 횟수가 적을 경우 렌더링의 품질이 저하되고 반대로 경우에는 화질의 표현력은 높으나 많은 처리시간이 소요된다. 따라서 화질이 떨어지지 않는 최소의 탐색 방법이 요구되므로 본 논문에서는 밀도의 탐색 최적화와 시간별 밀도가 존재하는 위치를 예측하여 계산을 효율적으로 처리 할 수 있는 PyCUDA 프레임워크에 대해서 소개한다.

키워드: 볼륨 렌더링(Volume rendering), 밀도 쿼리(Density query),
파이쿠다(PyCUDA), 가속화(Acceleration)

I. Introduction

볼륨 렌더링은 3차원 볼륨 데이터를 2차원 이미지로 가시화하는 방법이다[1]. 밀도를 색상 데이터와 매핑하여 수집된 데이터양에 따라 색상의 높낮이를 2차원 화면에 표시한다. 대표적으로 의료영상에 많이 쓰이며 3차원 밀도 데이터가 사용되는 기상 상황, 기체 영상, 게임 등 볼륨 데이터를 가시화할 때 일반적으로 사용된다. 정밀한 표현력이라는 장점을 가지지만 많은 양의 볼륨 데이터로 인한 처리시간이 오래 걸린다는 단점이 있다. 이를 해결하기 위해서 볼륨 렌더링 최적화는 필수적으로 필요한 과제이다.

본 논문에서는 중단 조건이라는 방법을 적용하여 밀도를 최소한으로 저장하고 저장된 중단 조건으로 밀도의 위치를 예측하여 렌더링 시간을 최소화하는 최적화 기법을 소개한다.

II. Preliminaries

1. Related works

볼륨 렌더링 계산 시에 큰 데이터 계산이라는 단점 때문에 시간을 단축하려는 기법들이 제안되어왔다. 볼륨 데이터에서 밀도가 없는 빈 공간을 건너뛰어 밀도가 존재하는 부분만 렌더링하는 기법이 있다[2]. 일반적인 밀도가 존재하는 복셀(Voxel)에 전이함수가 적용되지 않는 좌표는 빈 공간으로 취급하여 렌더링을 수행하지 않는 방법으로 최적화를 진행하였다. 이 외에 최소한의 밀도만 저장하는 방법도 연구되었다[3]. 이 방식은 2차원 스크린의 각 픽셀로 광선을 쏘면 광선이 관통하는 복셀을 누적하여 이 픽셀이 누적한 복셀의 양이 충분해지면 광선을 단절시키는 방법이다.

III. The Proposed Scheme

1. Volume rendering

이번 장에서는 볼륨 렌더링 기법의 구현 방법을 간단하게 설명한다. 공간이 격자로 이산화된 각 위치에서 볼륨 데이터의 방향으로 광선을 쏜다. 이후 광선이 진행되는 방향에 위치하여 광선과 관통하는 복셀에 저장되어 있는 볼륨 데이터의 밀도, 조명의 흡수, 방사에 대한 값을 수집한다. 수집된 복셀 데이터들을 누적하며 한 좌표의 값을 저장한다. 광선에 따라 좌표 시점까지 도착하여 저장된 값을 색상 데이터로 변환하여 렌더링한다 (Fig. 1 참조). 본 논문에서는 연기 데이터 장면을 사용했으며 밀도 데이터의 양이 방대하여 데이터들을 배열에 초기화하여 JSON 규격으로 저장하여 진행하였으며, 결과적으로 빠르게 전처리 과정을 수행하였다.

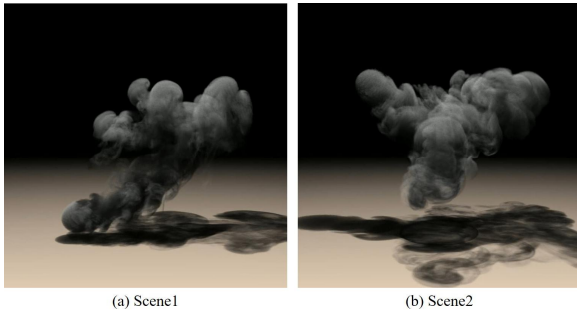


Fig. 1. Visualizing smoke with volume rendering.

2. Break condition

앞에서 언급했듯이 밀도를 수집하는 과정에서 방대한 볼륨 데이터의 양 때문에 광선의 방향에 존재하는 모든 밀도를 수집하게 되면 큰 연산시간이 발생하게 된다. 밀도를 다 탐색하였음에도 광선의 탐색 범위가 끝나지 않았다면 불필요한 탐색이 이루어질 수도 있다. 따라서 본 연구에서는 중단 조건이라는 기법을 제안하여 적용함으로써 이를 개선한다.

일반적으로 볼륨 렌더링은 다회 차 확인 시 빠른 렌더링 처리를 요구하기 때문에 처음 실행 시 누적된 밀도 데이터 수치를 저장하여 다음번에 복셀 탐색에서 사용할 수 있도록 하였다. 저장되었던 좌표별 밀도 누적 데이터만큼 해당 좌표에 누적이 되어 수치만큼 저장되었다면 광선을 조기 종료하도록 계산하였다. 또한, 밀도의 누적이 일정 수치 이상이 되면 적정량으로 밀도를 수집하였다고 판단하여 광선을 조기 종료시켜 밀도를 최소한으로 저장하여 계산을 줄이고자 하였다.

3. Prediction of break condition

중단 조건을 저장하는 것에는 큰 오류가 하나 있다. 매년 중단 조건을 계산하는 점과 이 과정을 영상의 모든 프레임에 대하여 진행한다는 점이다. 중단조건을 저장할 한번은 저장해야하므로 기존 기법보다는 연산시간이 빠르지만 효과적이지 않다. 그래서 중단 조건을 저장하되, delta 타임별로 다음 프레임에 대한 중단 조건 값을 예측하여

중단 조건을 저장하는 과정을 단축했다. 중단 조건을 예측하는 방법으로 밀도의 변화량에 따른 기울기 벡터를 프레임의 좌표별로 계산하였다.

$$\begin{aligned} x_g &: (x + 1) - (x - 1) \quad (1) \\ y_g &: (y + 1) - (y - 1) \\ &\Delta (x_g, y_g) \end{aligned}$$

사용되는 변수 x_g 는 x축의 밀도에 대한 변화량, y_g 는 y축의 밀도에 대한 변화량을 의미한다. 다음의 공식을 통하여 x와 y의 밀도의 변화량에 따른 기울기 벡터를 각 좌표별로 모두 계산한다.

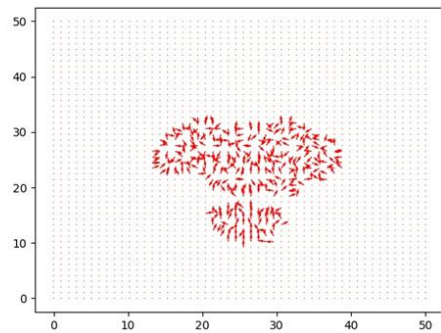


Fig. 2. Scene 3 frame 138, By Coordinate Gradient

위 과정에서 계산된 기울기 벡터를 기존 좌표에 적용하여 현재 좌표의 밀도 데이터를 기울기 벡터로 계산된 예측 좌표로 delta 타임에 비례하여 값을 이동시킨다.

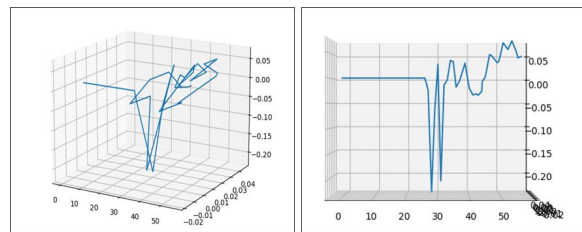


Fig. 3. Frame one coordinate Gradient

프레임 좌표별로 기울기 벡터를 계산하여 현재 좌표가 예측되면서 저장되어있던 밀도의 수치가 변화하는 것을 표현한 결과이다.

예측할 때 주의할 점으로는 두 가지가 있다. 첫 번째로 단일 프레임으로 예측이 진행된다면 예측으로 계산이 이루어지는 만큼 밀도의 위치가 자연스럽게 않고 결과가 불안전해 보여 단일 프레임 예측값만으로는 보완할 수 없다. 그러므로 본 연구에서는 다중 프레임으로 계산하여 이어지는 변화를 자연스럽게 하고자 4개의 프레임을 범위로 잡고 이 프레임들에 기여도 차이를 두어 예측값을 계산하여 중단 조건을 예측하고자 하였다. 이 예측되는 범위는 예측한 값을 포함하여 다음 프레임의 값을 계산하고, 저장해두었던 중단 조건을 사용하지 않으며 예측을 할 수 있도록 했다.



Fig. 4. Apply natural Gradient prediction

두 번째로는 본 연구에서 사용되는 기체 데이터처럼 일정하게 진행되지 않고 다양한 이벤트가 밀도에 발생할 경우에 효과적으로 예측을 할 수 없다는 점이다. 이러한 경우에는 필히 기존 중단 조건과 밀도의 차이가 발생하게 되는데 이전 회차에서 프레임을 계산할 때 기존의 저장되었던 중단 조건 값을 불러와 이 값으로 다시 기울기 벡터를 계산하여 좌표별 밀도 데이터의 값의 증감을 계산해 해결하였다.

4. Apply predicted values

중단 조건 값을 계산하고 저장하였다면, 다시 볼륨 렌더링 계산을 시작할 때이다. 이때 계산을 진행할 때 중단 조건을 적용하여 계산하지만 이미 예측을 진행하였으므로 일일이 좌표별 밀도를 확인하지 않고 바로 계산을 완료한다.

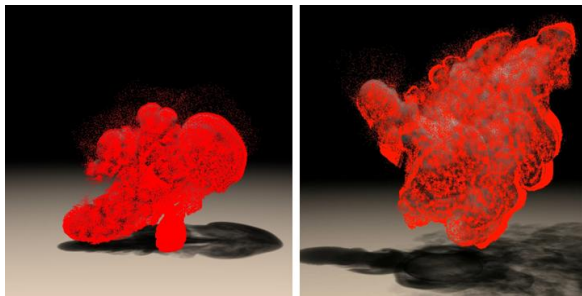


Fig. 5. the predicted part of red

예측값을 적용한 값들을 명확하게 확인하기 위하여 red 색상을 적용하여 확인해 보았고 정상적으로 계산되는 것을 확인할 수 있다.

5. Comparison

본 논문에서 진행한 중단 조건 기법과 중단 조건을 기울기로 예측하여 계산한 방법이 계산 시 유의미한 차이가 발생하는지 확인하기 위해 기존의 볼륨 렌더링 결과를 제외하고 중단 조건 기법만을 적용한 전체 연산시간과 중단 조건을 기울기벡터를 활용하여 예측하여 계산한 전체 연산시간을 비교하였다.

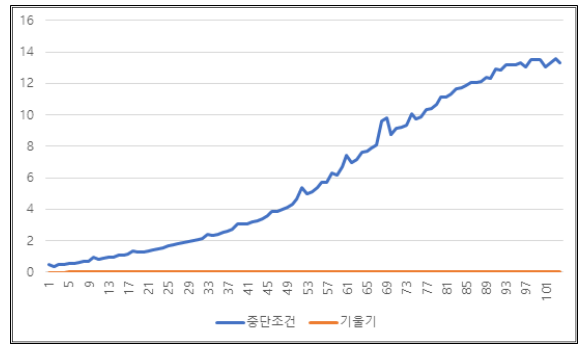
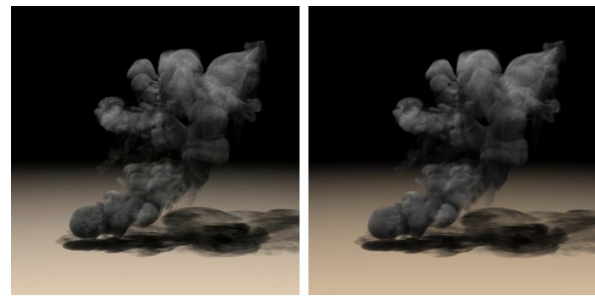


Fig. 6. Apply natural Gradient prediction

기울기벡터를 계산하여 예측한 중단 조건을 적용하여 렌더링을 진행하면 중단 조건을 계산하던 scene보다 20배 시간 단축을 확인할 수 있다.

frame 당 8sec의 디바이스 계산시간을 확인할 수 있다.



a) Calculate result

b) Original result

Fig. 7. Comparison of results

Fig7.에 왼쪽 이미지가 본 논문에서 제안한 기법으로 구현한 결과이다. 오른쪽의 원본 볼륨 데이터와 밀도의 수치 차이를 확인해 보았을 때, 원본 기체 볼륨 데이터 기준 최대 밀도 차이를 최대 0.03까지 줄였고, 부드러운 결과를 확인할 수 있다.

IV. Conclusions

본 논문에서는 중단 조건을 적용하여 밀도를 계산하므로 불필요한 계산을 줄였으며, 중단 조건을 계산하는 과정에서 밀도의 변화량에 따른 기울기를 예측하여 중단 조건 또한 최적화하였다. 이를 통해서 기존의 기법보다 밀도의 차이를 최소화하고 렌더링 시간을 단축할 수 있음을 확인하였다.

REFERENCES

- [1] Marc Levoy, "Display of Surfaces from Volume Data", IEEE CG&A, May 1988.
- [2] Sherbondy A, Houston M, Napel S. "Fast volume segmentation with simultaneous visualization using programmable graphics hardware", 8th World Congress on Intelligent Control and Automation, 2010
- [3] MARC LEVOY, "Efficient Ray Tracing of Volume Data", University of North Carolina, ACM Transactions on Graphics Vol.9, No.3, 1990
- [4] SooHo Lee, Jong-Hyun Kim, "Novel Kernel Design for Implementing Volume Rendering in the PyCUDA Framework", Computer Information Society of Korea, 2022