

석사학위논문

효율적인 최대 취소 가시화
알고리즘과 알고리즘 선택

2019년

한 성 대 학 교 대 학 원

정 보 시 스템 공 학 과

정 보 시 스템 공 학 전 공

한 철 희

석사학위논문
지도교수 계희원

효율적인 최대 휘소 가시화 알고리즘과 알고리즘 선택

Efficient maximum intensity projection algorithm
and algorithm selection

2018년 12월 일

한성대학교 대학원

정보시스템공학과

정보시스템공학전공

한 철 희

석사학위논문
지도교수 계희원

효율적인 최대 휘소 가시화 알고리즘과 알고리즘 선택

Efficient maximum intensity projection algorithm
and algorithm selection

위 논문을 공학 석사학위 논문으로 제출함

2018년 12월 일

한 성 대 학 교 대 학 원

정 보 시 스템 공 학 과

정 보 시 스템 공 학 전 공

한 철 희

한철희의 공학 석사학위 논문을 인준함

2018년 12월 일

심사위원장 이 지 은 (인)

심사위원 서 화 정 (인)

심사위원 계 희 원 (인)

국 문 초 록

효율적인 최대 휘소 가시화 알고리즘과 알고리즘 선택

한 성 대 학 교 일 반 대 학 원
정 보 시 스템 공 학 과
정 보 시 스템 공 학 전 공
한 철 희

최대 휘소 투영(Maximum intensity projection, MIP)는 의료 영상 처리에서 일반적인 시각화 기술 중 하나이다. 최대 휘소 투영을 활용한 연구는 다양하게 진행되며, 그 중 대표적으로 가속화와 화질개선에 대한 연구가 대표적이다. 본 연구는 최대 휘소 투영에서 가속화 방법 중 하나인 빈 공간 도약을 이용하여 가속화하는 알고리즘을 구현한다. 또한 빈 공간 도약 알고리즘을 개선하여 효과적으로 향상시키는 방법에 대해 연구하였다. 기존의 빈 공간 도약을 이용하여 새로운 방법들을 제시하고, 새로운 방법들을 효과적으로 선택하는 방법에 대해 소개한다. 빈 공간 도약은 볼륨 데이터에서 불필요한 부분을 연산하지 않고 건너 뛰어 연산량을 최소화하는 알고리즘인데, 본 연구에서는 불필요한 부분과 영상뿐만 아니라 큰 영향을 미치지 않는 부분까지 포함하여 건너뛰는 방법을 소개함으로써 효과적인 가속화 알고리즘을 제시한다. 빈 공간 도약 알고리즘에서 개선된 새로운 방법들은 개선된 속도와 다양한 이미지 품질을 보여주며, 본 연구는 이를 비교하여 최상의 이미지를 선택하는

방법을 제시한다. 실험 결과는 다양한 공차 함수를 이용하여 사용자가 요구하는 속도에 적합한 가속화된 영상을 보여준다. 제안된 방법을 효과적으로 선택하기 위해 최대 휘소 투영을 여러 개의 의료 데이터를 통해 실험하고, 짧은 시간에 비교적 높은 품질의 이미지를 선택하는 방법을 제시한다.

【주요어】 의료 가시화, 볼륨 가시화, 최대 휘소 투영, 빈 공간 도약

목 차

제 1 장 서 론	1
제 1 절 볼륨 가시화	1
제 2 절 연구 내용	2
제 3 절 논문 구성	3
제 2 장 관련 연구	4
제 1 절 최대 휘소 투영	4
제 2 절 빈 공간 도약	4
제 3 장 효율적 빈 공간 도약	7
제 1 절 동적 판단을 이용한 빈 공간 검사	7
제 2 절 공차(公差) 함수의 제작	9
1) 상수 공차 함수	9
2) 중앙으로부터 거리를 독립변수로 하는 공차 함수	11
3) 복셀값을 독립변수로 하는 공차 함수	12
제 3 절 공차 함수의 다양성	13
제 4 장 최적 공차 함수 선택법	15
제 1 절 데이터 세분화	15
1) 고밀도 골격형 데이터	16
2) 저밀도 골격형 데이터	16
제 2 절 최적의 공차 함수	17
제 5 장 실험	19

제 1 절 실험 환경	19
제 2 절 히스토그램 누적 합 곡선을 이용한 데이터 세분화	19
1) 히스토그램	20
2) 히스토그램 누적 합 백분율에 따른 데이터 세분화 분석	22
제 3 절 데이터 세분화에 따른 속도 및 화질 비교	23
제 4 절 데이터 세분화에 따른 영상 비교	25
제 5 절 데이터를 이용한 선택법 검증	31
제 6 장 결론 및 향후 개선 방안	35
참 고 문 헌	37
ABSTRACT	39

표 목 차

[표 3-1] 하나의 광선에 대해 공차 값을 더하는 알고리즘 예시	9
[표 5-1] 볼륨 데이터 세트	19
[표 5-2] 히스토그램 누적 합 백분율에 따른 밀도값	22

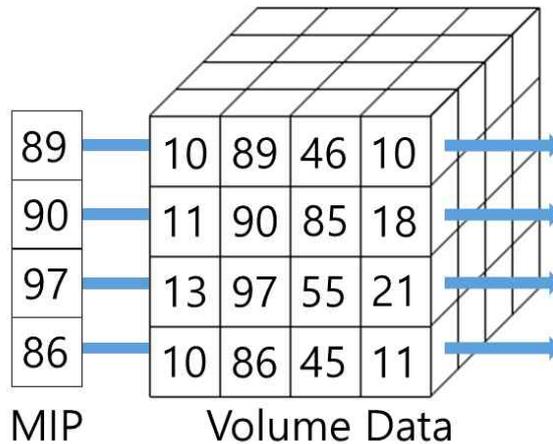
그림 목 차

[그림 1-1] 광선 추적법 예시	1
[그림 2-1] 빈 공간 도약 예시	5
[그림 3-1] 최댓값을 비교하여 블록을 건너뛰는 예시	7
[그림 3-2] 상수 공차 함수의 그래프	10
[그림 3-3] 중앙으로부터 거리를 독립변수로 하는 공차 함수의 그래프	12
[그림 3-4] 복셀값을 독립변수로 하는 공차 함수의 그래프	13
[그림 4-1] 혈관의 밀도에 따른 예시 영상	15
[그림 5-1] lower 데이터의 히스토그램	20
[그림 5-2] head 데이터의 히스토그램	20
[그림 5-3] liver1 데이터의 히스토그램	21
[그림 5-4] liver2 데이터의 히스토그램	21
[그림 5-5] 시간-신호 대 잡음비 그래프	24
[그림 5-6] lower 데이터의 가시화 영상	26
[그림 5-7] head 데이터의 가시화 영상	27
[그림 5-8] liver1 데이터의 가시화 영상	29
[그림 5-9] liver2 데이터의 가시화 영상	30
[그림 5-10] 흉부 데이터의 히스토그램	31
[그림 5-11] 흉부 데이터의 가시화 영상	33

제 1 장 서론

제 1 절 볼륨 가시화

볼륨 가시화는 일반적으로 CT나 MR 등으로 얻은 의료 데이터를 이용하여, 진단에 필요한 의료 영상을 생성하는 가시화 방법이다(Sabella, P., 1988). 본 연구에서 다루는 최대 휘소 투영(Maximum Intensity Projection, MIP)(Schreiner, S., 1993; Mroz, L., 1999)은 대표적인 볼륨 가시화 방법 중 하나로, 볼륨 데이터에서 관찰자가 바라보는 방향으로 최대값을 누적하여 화면에 출력하는 방법이다.



[그림 1-1] 광선 추적법 예시

최대 휘소 투영은 골격이나 조영된 혈관, 피부층, 근육조직과 같이 높은 밝기의 조직을 주변 장기의 가림 없이 출력하기 때문에 의료 진단에 다양하게 활용된다. 최대 휘소 투영 영상은 일반적으로 쉬어-왓 분해(Fang, L., 2002; Kye, H., 2017)와 광선 추적법(Levoy, M., 1988)이 대표적으로 사용되며, 높은 화질과 확장성 측면에서 현대에는 쉬어-왓 분해보다는 광선 추적법을 사용하는 추세이다. 광선 추적법은 [그림 1-1]과 같이, 영

상을 구성하는 각 복셀 데이터에서 관찰자가 바라보는 방향에서 가상의 광선을 발사하여, 볼륨 데이터 내부의 각 복셀 데이터에서 밀도값을 얻고, 이들을 비교하며 최댓값을 누적하여 화면에 출력하는 방법이다. 광선 추적법은 연산량이 많은 기법으로 시간이 많이 소요되기 때문에, GPU를 사용(Kiefer, G., 2006; Kye, H., 2011; Kye, H., 2012; Mroz, L., 2000)하거나 데이터를 사전 정렬(Mroz, L., 2000)하는 등 다양한 방법으로 가속화가 연구되었다. 그 중 빈 공간 도약 알고리즘(Levoy, M., 1990; Kye, H., 2015; Kwon, O., 2015)은 일반적으로 사용되는 가속화 방법이다.

제 2 절 연구 내용

최대 휘소 투영 영상의 생성 과정은 누적과 비교 연산에서 많은 시간을 소요한다. 최대 휘소 투영의 출력 픽셀은 가상의 광선이 진행하면서 비교한 데이터에서의 최댓값 하나로 결정되므로 가상의 광선이 진행 중 얻은 다른 값들은 최종 영상에 반영되지 않는다. 따라서 다른 값들은 최종 영상에 반영되지 않으므로 불필요한 값으로 분류가 가능하다. 이에 착안한 빈 공간 도약은 광선의 불필요한 값들을 연산하지 않고 건너뛰어 속도를 향상한다. 그러나 기존의 빈 공간 도약은 공간 도약의 효율이 높지 않은데, 그 이유는 어떤 영역이 가시화에 불필요한지 판단할 때 판단 기준이 지나치게 엄격하기 때문이다.

본 연구는 먼저 빈 공간 도약의 효율을 향상하기 위해, 볼륨 각 영역의 생략 여부를 낙관적으로 예측하는 몇 가지 새로운 방법을 제안한다. 제안 방법은 더 많은 공간을 도약하여 불필요한 값들에 대해서 관대하게 도약하는 방법으로, 가시화 속도가 우수하지만 경우에 따라 화질 손실이 발생할 수 있다. 따라서 본 연구는 제안 방법들을 화질과 속도의 관점에서 서로 비교하였다. 제안 방법의 상대적 우수성은 사용자가 원하는 화질의 수준에 따라 달라진다. 본 연구는 이를 이용하여 사용자가 원하는 화질 수준에 따라 적합한 가속 방법을 선택하도록 방법을 제시한다. 제안 방법의 상대적 우수성은 사용자가 원하는 화질의 수준에 따라 달라지며, 일정 화질 손실을 감안한다면 매우

우수한 속도로 연산이 가능하다. 이를 이용하여 본 연구는 사용자가 원하는 화질 수준에 따라 적합한 가속 방법을 선택하도록 유도하며, 고속 가시화하는 방법을 제안한다. 본 연구의 공헌은, 낙관적으로 최대 휘소 투영 빈 공간 도약을 수행하는 새로운 방법을 제안하며, 제안한 가속화 방법의 성능을 비교하고, 사용자 입력에 따라 자동 선택 되는 고속 최대 휘소 투영 가시화 방법을 제시하는 것이다.

제 3 절 논문 구성

본 논문의 구성은 2장에서 기존 연구인 빈 공간 도약 알고리즘에 대해 소개한다. 3장에서는 제안한 알고리즘의 구현 방법에 대해 설명한다. 4장에서 제안한 알고리즘을 선택하는 방법을 제안한다. 5장은 실험 결과로 제안한 알고리즘의 속도와 화질의 차이를 보이고 6장에서 결론과 향후과제를 제시한다.

제 2 장 관련연구

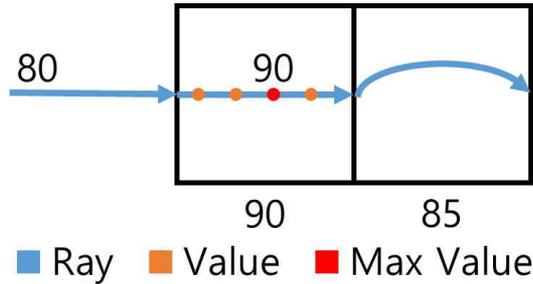
제 1 절 최대 휘소 투영

최대 휘소 투영(Maximum Intensity Projection, MIP)(Schreiner, S., 1993; Mroz, L., 1999)은 볼륨 가시화 방법 중 대표적인 방법으로 관찰자가 바라보는 방향에서 볼륨 데이터의 밀도값을 비교하며 최댓값을 누적하는 기법이다. 볼륨 가시화에 비해 연산량이 적으므로 데이터가 커서 연산량이 많은 의료 데이터에서는 많은 장점을 얻을 수 있다. 대표적인 최대 휘소 투영은 2가지 방법이 있는데 쉬어-왁 분해(Fang, L., 2002; Kye, H., 2017)와 광선 추적법(Levoy, M., 1988)이 대표적이다. 현재에는 광선 추적법을 주로 사용하며, 광선 추적법의 장점은 쉬어-왁 분해에 비해 높은 화질을 갖는다. 반면에 연산량이 쉬어-왁 분해에 비해 많다는 단점이 있다. 광선 추적법은 2가지로 구분되는데 관찰자가 바라보는 방향에서 출력 영상을 가져오는 방법과 관찰자가 바라보는 방향 반대편에서 출력 영상을 맺히는 방법이 있다.

제 2 절 빈 공간 도약

볼륨 데이터에서 출력 영상에 반영되지 않는 불필요한 부분을 제거할 수 있다면 연산량이 줄어 가시화 속도는 향상된다(M. Levoy, 1990). 빈 공간 도약 방법(Levoy, M., 1990; Kye, H., 2015; Kwon, O., 2015)은 어떤 영역에 속한 데이터의 값이 작다면, 그 영역은 최댓값을 구하는 최대 휘소 투영 결과에 반영되지 않으므로, 해당 영역을 건너뛰어 연산을 생략할 수 있음에 착안한 것이다(Mora, B., 2005). 빈 공간 도약을 수행하기 위해, 볼륨 데이터를 같은 크기의 영역, 즉 블록(block)의 모음으로 구성한다. 구체적으로, 각 블록 영역에 속한 복셀값들을 비교하여 최댓값을 계

산하고 별도의 공간에 저장하여 놓는다(Kye, H., 2008). 이 과정은 전체 단계에서 한 번만 일어나며, 다양한 관찰 방향에서 가시화가 필요할 때 블록 정보는 재사용 된다.



[그림 2-1] 빈 공간 도약 예시

가시화 과정에서는 블록 정보를 이용하여 빈 공간 도약을 수행한다(Kye, H., 2011). 광선은 관찰방향으로 진행하면서 현재 위치까지 누적된 최댓값을 갱신한다. [그림 2-1]과 같이 광선이 현재까지 누적된 최댓값이 80이라고 가정한다. 만약 왼쪽 블록의 최댓값이 90이라면 블록 내부를 탐색하며 최댓값을 갱신할지 탐색한다. 왼쪽 블록의 연산 결과 최댓값이 90으로 변경되며, 오른쪽 블록은 최댓값이 85이므로 무시하고 연산을 건너뛰어도 최댓값 90은 변화가 없다. 이렇게 오른쪽 블록처럼 블록을 건너뛰면 화질 손실 없이 불필요한 값에 대한 연산이 줄어 가속화를 수행한다.

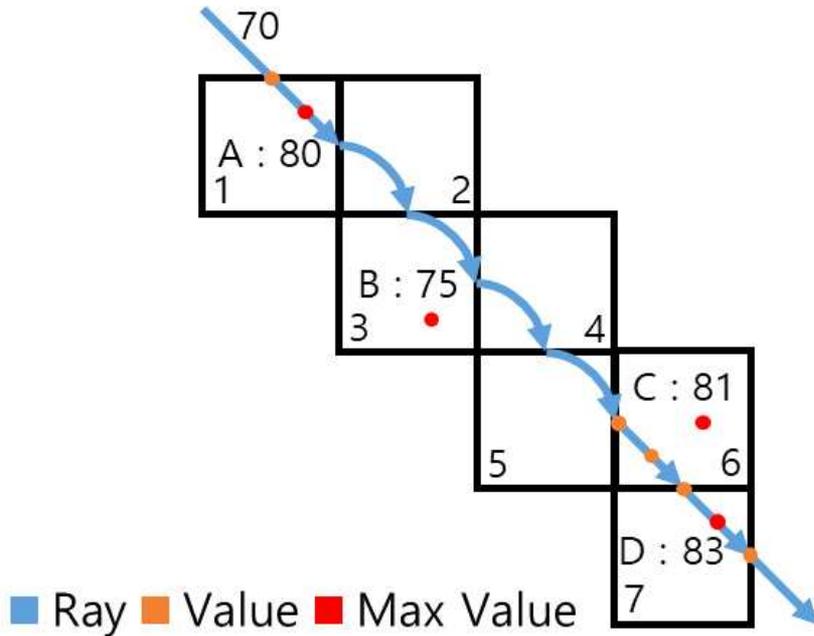
블록의 크기는 다양하게 결정할 수 있으며 그에 따라 장단점이 있다(Pekar, V., 2003). 예를 들어 블록의 크기가 $16 \times 16 \times 16$ 라면 같이 블록 하나 당 복셀의 수가 4096개가 포함될 수 있다. 만약 최대 취소 투영 연산 중에 어떤 $16 \times 16 \times 16$ 블록에 소속된 단 하나의 값이 매우 큰 경우 그 블록은 생략할 수 없어 4096개의 복셀 중 광선이 탐색을 해야 한다. 따라서 블록의 크기가 크면, 블록을 건너뛸 확률이 감소하여 연산량이 증가한다. 이는 당연하게도 4096개의 복셀 중 큰 밀도값을 갖고 있는 복셀이 존재할 확률이 높기 때문이다. 반대로 블록의 크기가 작은 $4 \times 4 \times 4$ 블록의 경우, 복셀은 64개가 존재한다. 블록의 크기가 작기 때문에 블록을 건너뛸 확률은 올라간다. 한편, 메모리 사용량 관점에서 비교해보면, 블록 크기가

작은 경우 생성되는 블록의 개수가 많아지므로 메모리 소모량이 많아진다.

본 연구의 경우에는 메모리가 충분하다는 가정하에 최대한 많은 블록을 건너뛰도록 가속화에 중점을 둔다. $4 \times 4 \times 4$ 블록의 경우 블록의 메모리가 지나치게 많아 불필요하므로 상대적으로 작은 크기의 $8 \times 8 \times 8$ 블록을 사용한다.

제 3 장 효율적 빈 공간 도약

제 1 절 동적 판단을 이용한 빈 공간 검사



[그림 3-1] 최댓값을 비교하여 블록을 건너뛰는 예시

[그림 3-1]은 최대 취소 투영에서 가상의 광선이 볼륨 데이터를 통과하면서 밀도값을 비교하여 빈 공간 도약을 수행하는 예를 보인다. [그림 3-1]에서 점 A, B, C, D 는 볼륨 데이터에서의 밀도값이며 각각 80, 75, 81, 83의 값을 갖는다. 또한 이 점들이 속하는 1번, 3번, 6번, 7번 블록에서의 최댓값이 점 A, B, C, D라 가정한다. 즉, 1번 블록의 최댓값은 점 A의 밀도값인 80, 3번 블록의 최댓값은 점 B의 밀도값인 75, 6번 블록의 최댓값은 점 C의 밀도값인 81, 7번 블록의 최댓값은 점 D의 밀도값인 83이다. 나머지 2번, 4번, 5번 블록의 경우의 최댓값은 40이다. 광선이 1번 블록에 진입하기 전까지 누적된 최댓값은 70으로 가정한다.

현재 광선이 1번 블록을 만날 때, 1번 블록의 최댓값이 80이므로 블록을 건너뛰지 못하고 블록 내부에서 연산이 진행된다. 광선이 많은 샘플링을 통해 진행하면서 점 A를 만나게 되고 광선의 최댓값은 80으로 갱신된다. 80으로 갱신된 광선이 최댓값이 40인 2번 블록을 만나면, 빈 공간 도약이 가능하여 2번 블록을 건너뛰어 3번 블록으로 넘어간다. 3번 블록의 최댓값이 75이고 현재 광선의 최댓값이 80으로 갱신되었으므로 2번 블록과 마찬가지로 3번 블록도 건너뛸 수 있다. 이와 마찬가지로 4번, 5번 블록의 최댓값이 40이므로 4번, 5번 블록도 건너뛰어 연산을 생략한다.

다음은 6번 블록인데, [그림 3-1]에서 광선은 점 C를 통과하지 않지만 그 정보는 미리 알 수가 없다. 6번 블록의 최댓값이 81인 것만 알려져 있으므로 6번 블록을 건너뛰지 못하고 광선이 진행하면서 비교하는 방법 밖에 없다. 광선이 진행하면서 점 C를 만나지 않기 때문에, 최댓값이 갱신되지 않은 채 불필요한 광선 진행과 많은 샘플링을 수행한다. 최대 취소 투영에서의 전체 가시화 과정에서 6번 블록과 같은 경우가 비일비재하다. 따라서 많은 광선이 불필요한 샘플링을 수행하게 되므로 연산이 많아진다.

이후 6번 블록의 모든 복셀 값을 비교한 광선은 7번 블록을 만나게 되며 7번 블록의 최댓값이 83이므로 마찬가지로 통과하지 못하고 6번 블록의 모든 밀도값을 비교하며 광선이 진행한다. 점 D, 최댓값 83을 만나 광선은 83으로 최댓값을 갱신하고 진행하게 된다.

본 연구는 6번 블록에서 점 C와 같은 경우에 착안하여, 광선의 누적 값과 블록의 최댓값의 차이가 작은 6번 블록과 같은 경우를 건너뛰어 샘플링을 줄여 효율적인 가시화를 수행하려고 한다. 본 연구는 새롭게 공차(公差, tolerance) 값을 두어, 블록과 광선에 누적된 값의 차이에 따라 블록을 건너뛸지 판단한다. 예를 들어 6번 블록에 광선이 닿을 때 광선에 누적된 값 80과 6번 블록의 최댓값 81이 유사하므로, 누적된 값이 작더라도 건너뛰도록 한다.

7번 블록의 경우, 현재까지 누적된 값 80과 블록 값 83이 유사하다고 판단하는 공차 값을 갖게 된다면, 7번 블록도 건너뛰게 된다. 이 때 속도는 향상되지만 실제 광선이 통과해야 할 값인 점 D, 83을 무시하였으므로 화질 손실이 발생한다. 만약 화질 손실을 방지하고자 공차 값을 줄인다면, 불필요한

6번 블록 또한 건너뛴 수 없게 된다. 즉, 공차를 크게 하면 속도는 향상되지만 화질 손실의 확률이 증가하므로, 사용자가 원하는 성능과 화질에 따라 공차 값을 결정해야 한다.

한편, 공차 값 개념을 개발에 적용할 때 고려할 사항이 많다. 광선에 누적된 값과 블록에 저장된 최댓값을 비교할 때, 단순히 공차 값 차이를 계산할 것인지 상대적인 비율을 계산할 것인지 고려해야 한다. 그리고 모든 광선에 대해 동일한 공차 값을 적용할지, 광선의 진행에 따라 변화된 값을 적용할지도 고려해야 한다.

제 2 절 공차(公差) 함수의 제작

1) 상수 공차 함수

광선에 누적된 최댓값과 블록의 최댓값을 비교할 때 가장 간단한 전략은 두 최댓값의 절대적 차이를 계산하는 것이다. 블록의 최댓값이 더 작다면 그 블록은 명백하게 불필요하며 따라서 건너뛰면 된다. 블록의 최댓값이 크더라도 그 차이가 공차 값 이하라면 불필요한 블록으로 판단한다.

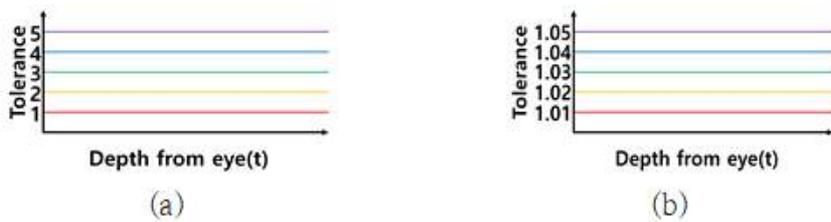
[표 3-1] 하나의 광선에 대해 공차 값을 더하는 알고리즘 예시

하나의 광선에 대한 광선 추적
1: (tMin, tMax) := RayVolumeIntersect(RayStart, Direction)
2: MaxValue := 0
3: for t := tMin to tMax
4: Block := FindCurrentBlock(t)
5: if(Block.Max <= MaxValue + tolerance)
6: t := t + SkipLength(Block)
7: else
8: position := RayStart + Direction * t
9: if(MaxValue < Density(position))
10: MaxValue := Density(position)
11: t := t + 1
12: endfor

[표 3-1]은 블록에 동일한 공차 값을 더해서 비교하는 알고리즘에 대한 예시이다. [표 3-1]에서 제시된 알고리즘은 일반적인 최대 취소 투영 알고리즘에서 빈 공간 도약과 매우 유사하다. 먼저 가상의 광선 하나의 시작 위치(RayStart)와 관찰자의 방향(Direction)으로 광선의 진행 범위(tMin, tMax)를 얻는다(line 1). 다음은 현재 속해있는 블록(FindCurrentBlock)의 최댓값을 얻는다(line 4). 블록의 최댓값이 광선의 최댓값보다 작다면, 그 블록 영역만큼 도약하고(line 6) 값이 크다면 현재 좌표에서 샘플링과 누적을 수행(line9-11)한 후 단위 길이만큼 진행한다(line 11).

만약 공차 값(tolerance)이 0이라면 빈 공간 도약(M. Levoy, 1990; H. Kye, 2008)과 완전히 동일하며, 공차 값을 증가시키면 광선의 최댓값이 증가되어 블록의 최댓값보다 커지게 되어 더 많은 블록을 건너뛴다.

[그림 3-1]의 6번 블록을 예로 들면 공차가 0일 경우, 현재 광선의 최댓값이 80이고 다음 블록이 6번 블록이라면 블록의 최댓값이 81이므로 블록을 건너뛰지 못한다. 하지만 공차가 1이라면 $\text{Block.Max} (=81) \leq \text{MaxValue} (=80) + \text{tolerance} (=1)$ 이 되어 최댓값이 81인 블록은 건너뛴다. 마찬가지로 [그림 3-1]의 7번 블록을 광선이 만난다면 공차 값이 1이고 광선이 81의 최댓값을 갖고 있으므로 $\text{Block.Max} (=81) \leq \text{MaxValue} (=80) + \text{tolerance} (=1)$ 이 되어 블록을 건너뛰지 않고 블록을 탐색하게 된다.



[그림 3-2] 상수 공차 함수의 그래프

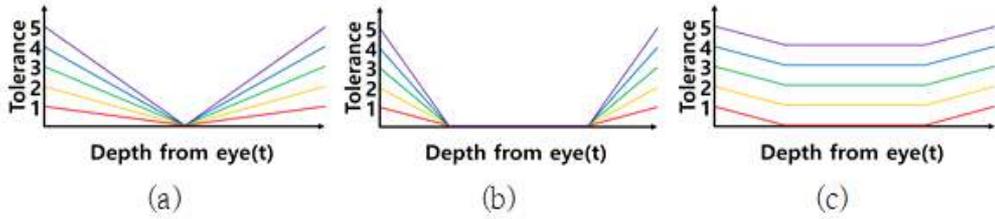
[그림 3-2]의 (a)는 덧셈을 이용한 공차 함수의 예시로 이를 공차 함수 A로 지칭한다. [그림 3-2]의 (a)공차 함수 A에서는 공차 값이 1, 2, 3, 4, 5로

예시를 보인다. 각각은 모두 상수 함수이며, 사용자가 선택에 따라 공차 값이 변화하며 공차 값에 따라 화질과 속도가 변한다. 화질과 속도의 변화는 제 5장에서 확인 가능하다.

[그림 3-2]의 (b)에 공차 함수는 공차 함수 B로 지칭한다. 공차 함수 B는 [표 3-1]의 덧셈 연산(line 5) 대신 곱셈으로 바꾸고 공차 값을 비율에 따라 적용시킨 예이다. 공차 함수 B에서는 공차 값이 1.01배, 1.02배, 1.03배, 1.04배, 1.05배로 예시를 보인다. 공차 값이 1.01배로 설정되면, 1%의 차이는 건너뛸 수 있다. 예를 들어 현재 광선에 누적된 값이 100이라면 최댓값이 101이하인 블록을 건너뛸 수 있으며, 광선에 누적된 값이 200이라면, 최댓값이 202이하인 블록을 건너뛸 수 있다. 공차 함수 B의 경우는 현재 누적된 값을 클 때, 그에 비례하여 더 큰 값을 건너뛸 수 있도록 하는 전략이다. 이 역시 공차 함수 A와 마찬가지로 상수 함수이며, 사용자가 선택한 공차 값에 대해 화질과 속도가 변한다.

2) 중앙으로부터 거리를 독립변수로 하는 공차 함수

일반적인 의료 영상의 경우 중요 관심부위는 영상의 중앙에 위치하는 경우가 대다수이다. 따라서 데이터의 중심부는 정밀하게 연산하고, 외곽 부분은 공차 값을 주어, 여유 있게 판단하면 중앙부에 위치한 데이터의 화질 손실을 줄여 효율적인 가시화를 기대할 수 있다. 가상의 광선이 볼륨 데이터의 시작과 끝([표 3-1], tMin, tMax 참고)을 알고 있으므로 볼륨 데이터 중심의 위치를 계산할 수 있다. 따라서 거리에 따라 공차 값을 주는 공차 함수를 구현할 수 있다.

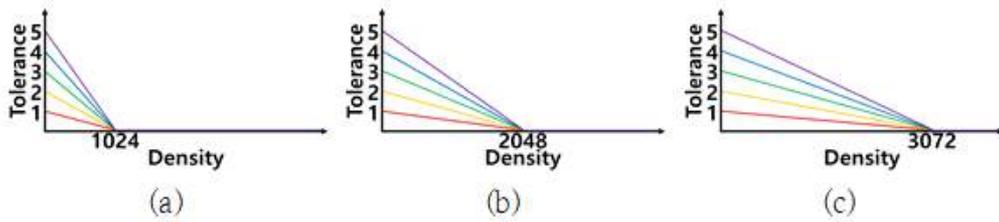


[그림 3-3] 중앙으로부터 거리를 독립변수로 하는 공차 함수의 그래프

[그림 3-3]의 (a)는 중심부의 공차는 항상 0으로 결정하는 방법이다. [그림 3-3]의 (a)를 공차 함수 C라 지칭한다. [그림 3-3]의 (b)는 공차 함수 D라 지칭하며, 깊이 영역을 4분할하여 중심부 영역, 즉 1/4지점부터 3/4지점까지의 공차 값을 0으로 두어 결정하는 방법이다. [그림 3-3]의 (c)는 공차 함수 D와 마찬가지로 깊이 영역을 4분할하여 중심부 영역은 상수 공차 값을 주고, 이외의 영역에서는 거리에 따라 더 큰 공차 값을 주도록 설계하였다. [그림 3-3]의 (c)는 공차 함수 E라 지칭한다.

3) 복셀값을 독립변수로 하는 공차 함수

최대 휘소 투영의 특성상 광선에 현재까지 누적된 값이 작으면 광선이 진행함에 따라 큰 값에 의해 갱신될 가능성이 높다. 따라서 현재까지 누적된 작은 값은 중요도가 낮다고 볼 수 있다. 이에 착안하여, 현재까지 누적된 밝기 값이 작을수록 큰 공차 값을 갖도록 하는 공차 함수를 설계하였다. 일반적인 CT데이터의 경우 12bit의 정밀도를 사용한다. 따라서 이에 맞게 0-4095로 대응하여, 다양한 모형을 설계하였다. 본 연구는 이미 정해진 밀도값, 다시 말해 복셀 데이터가 가지고 있는 값을 1024, 2048, 3072, 즉 밀도값 1024의 배수를 이용하여 공차함수를 생성하였다.



[그림 3-4] 복셀값을 독립변수로 하는 공차 함수의 그래프

[그림 3-4] (a)는 밀도값이 1024 이상일 경우는 공차 값을 0으로, 반대의 경우는 밀도값이 작을수록 크게 주어 낮은 밀도값의 경우 최대한 도약하게 하는 공차 함수이다. 이를 공차 함수 F라 지칭한다. [그림 3-4]의 (b)는 공차 함수 F와 유사하지만 2048의 밀도값을 기준으로 하며, 공차 함수 G라 지칭한다. 마찬가지로 [그림 3-4]의 (c)는 공차 함수 H로 지칭하며 3072의 밀도값을 기준으로 공차 값을 크게 주는 공차 함수이다.

제 3 절 공차 함수의 다양성

본 논문에서는 상수 공차 함수, 중앙으로부터 거리를 독립변수로 하는 공차 함수, 복셀값을 독립변수로 하는 공차 함수 총 3가지를 소개하였다. 이는 의료 데이터의 특성에 따라 설계된 함수로 의료 데이터에서 최적화된 효과를 지닌다. 의료 데이터가 아닌 데이터의 경우에는 다른 공차 함수가 더 좋은 결과를 가져올 수 있다.

또한 본 연구를 통해 공차 함수의 연산량 또한 중요하는 것을 확인하였다. 빈 공간 도약 알고리즘은 블록을 건너뛰어서 연산량을 줄임으로써 가속화하는 기법이므로, 개선된 빈 공간 도약 알고리즘 또한 블록을 건너뛰는 연산을 통해 가속화에 중점을 맞춰 개선되었다. 개선된 빈 공간 도약의 경우 공차 값이라는 연산을 추가적으로 하면서, 빈 공간 도약과는 달리 추가 연산이 포함된다. 이 때 추가 연산의 수식이 복잡하게 된다면, 일반적인 빈 공간 도약보다 더 많은 연산을 수행하여 가속화가 이루어지지 않을 수 있다. 따라서 개선된 빈 공간 도약 알고리즘에서 공차 함수를 올바르게 설계하기 위해서는 적

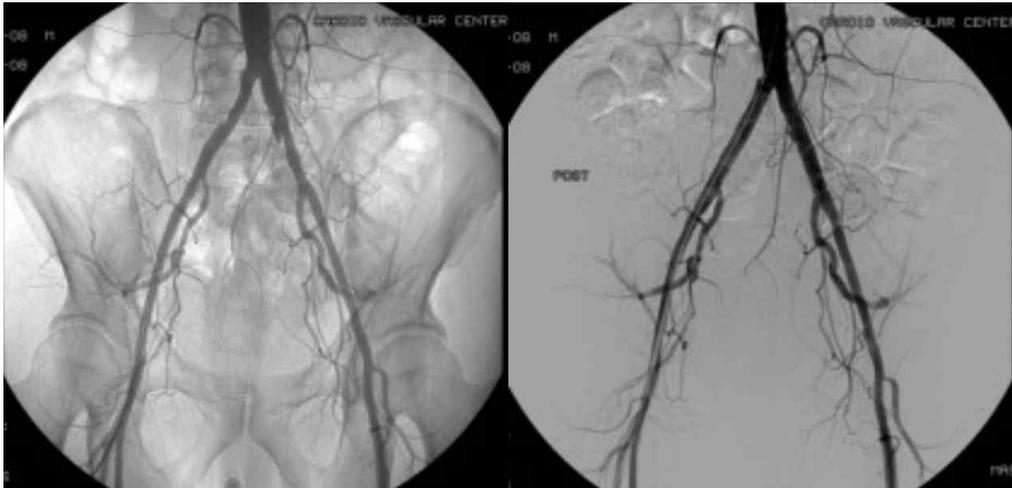
용시킬 데이터의 특징을 먼저 파악하고, 이에 따라 최대한 간결한 연산을 추가하여 블록을 건너뛰게 하여야 한다.

최대 취소 투영의 경우 최댓값을 이용한 볼륨 가시화 기법이다. 따라서 최댓값이 가장 중요한 자료임을 인지하고 추가적으로 공차 함수를 개선할 수 있다. 본 논문에서는 의료 데이터의 밀도값을 히스토그램으로 추출하여 특징을 선택하는 기법을 사용한다. 이는 실험을 통해 5장에서 검증하였다.

제 4 장 최적 공차 함수 선택법

제 1 절 데이터 세분화

이번 장에서는 3장에서 설계한 다양한 공차 함수들을 이용해 데이터를 세분화하였다. 공차 함수에 따라 속도와 화질에 있어 결과가 다르다. 제 5장 실험 결과에서 확인할 수 있듯이 데이터에 따라 다양한 공차 함수가 적용될 때 최적의 공차 함수가 다른 것을 확인 가능하다.



[그림 4-1] 혈관의 밀도에 따른 예시 영상

데이터를 판별하기 위해서는 데이터의 특징을 파악해야한다. 의료 데이터의 경우 피부나 근육층, 혈관, 골격부 등의 밀도값을 조절하여 [그림 4-1]과 같이 해당 혈관만 가시화하여 혈관이 막혔는지 또는 해당 골격만 가시화하여 골격이 부러졌는지 등을 판단한다. 이에 착안하여, 본 논문에서는 밀도값에 따른 특징을 찾고 밀도값에 따라 데이터를 세분화하는 방법을 제시한다. 의료 데이터에 있어 밀도값은 [그림 4-1]과 같이 큰 의미를 지니며, 이에 따라 영상의 결과 또한 큰 영향을 미친다.

데이터의 세분화는 밀도값을 이용한 히스토그램을 통해 세분화하였다. 히스토그램을 통해 세분화하는 이유는 데이터의 밀도값만 참조하므로 연산량이 불륨 가지화나 최대 휘소 투영에 비해 비교적 적다. 또한 밀도값의 분포를 수치화하여 확인 가능하며, 밀도값의 특징을 찾아내는데 용이하다.

히스토그램 수치화는 5장에서 확인 가능하다. 히스토그램의 분포도를 통해 데이터를 고밀도 골격형 데이터와 저밀도 골격형 데이터 2가지로 분류했다.

1) 고밀도 골격형 데이터

일반적인 CT데이터의 12bit로 경우 0~4095의 밀도값을 가진다. 본 논문에서는 일반적인 의료 데이터인 CT데이터를 통해 실험하였다. 1024이상의 값은 피부나 근조직, 혈관 등으로 구분되며, 1200이상의 값이 골격부로 구분된다. 1024이하의 밀도값은 매우 낮은 밀도값으로써 공기나 먼지 등이 이에 해당된다. 따라서 1024이하의 밀도값은 의료 데이터에 있어 불필요한 값들이 대부분이다.

본 논문에서 고밀도 골격형 데이터는 히스토그램 누적 합 곡선이 85%~95%에 있는 값들에서 1300이상의 값이 데이터의 10%이상 차지할 경우 고밀도 골격형 데이터로 판별한다. 1300이상의 값은 높은 밀도를 갖는 값이며, 이는 골밀도 기준으로 굉장히 단단한 골조직에 해당한다. 본 논문에서는 1300이상의 값이 10%이상 차지하는 부분을 히스토그램 누적 합 곡선을 통해 판별 가능하며, 이를 고밀도 골격형 데이터로 세분화 한다.

2) 저밀도 골격형 데이터

저밀도 골격형 데이터는 고밀도 골격형 데이터와는 달리, 밀도값이 1300이상의 값이 10%이하를 차지하는 데이터를 말한다. 이는 골조직이 비교적 약한 부분이 많이 포함되고, 밀도 값이 1024~1200인 피부나 근조직이 많은 데이터로 판단 가능하다. 제 5장의 히스토그램 누적 합 곡선을 통해 1300이상의 값이 10%이하인 데이터를 저밀도 골격형 데이터로 세분화 한다.

제 2 절 최적의 공차 함수

본 논문에서는 데이터를 고밀도 골격형 데이터와 저밀도 골격형 데이터로 분류한다. 공차 함수를 선택하는 방법은 실험을 통해 각 데이터의 특징을 발견하였다. 앞서 말한 히스토그램의 분포도를 통한 밀도값의 특징이 이에 해당한다. 공차 함수는 고밀도 골격형 데이터와 저밀도 골격형 데이터에서 최적의 공차 함수가 다르며, 이는 5장에서 실험을 통해 증명하였다.

본 논문에서 사용한 데이터는 의료 데이터로 국한되어 있다. 의료 데이터는 일반적으로 중앙부에 데이터가 밀집되어 있는데, 이는 촬영 방법에 밀접한 연관이 있다. 의료 데이터 촬영 시에 신체 일부 기관을 중점적으로 촬영한다. 따라서 촬영한 신체 일부가 항상 중앙부에 위치하는 특징을 지닌다. 의료 데이터가 중앙부에 위치한다는 특징에 따라 중앙부에 공차 값 낮게, 바깥쪽의 경우는 높게 사용한 공차 함수 C, D, E와 같은 알고리즘을 설계하였다.

다음은 모든 데이터에서 빠른 특징을 갖고 있는 알고리즘을 발견하였는데, 이는 공차 함수 A이다. 본 논문에서 소개한 공차 함수는 일반적인 빈 공간 도약에서 추가적으로 공차 값을 증가시켜 더 많은 블록을 건너뛰는 것이다. 더 많은 블록을 건너뛰는 것 또한 연산량을 줄이기 위함이었다. 실제로 의료 데이터는 많은 데이터를 참조하는데, 이는 공차 함수의 연산량에서도 영향을 미친다. 다시 말해 공차 함수 A의 경우 앞서 소개한 다른 공차 함수와 달리 연산이 간단하여 연산량이 작아 시간이 감소한다. 반면에 공차 함수 F, G, H는 의료 데이터의 특징에 따라 밀도값이 낮은 공기 층 부분을 최대한 건너뛰도록 공차 함수를 설계하였다. 하지만 밀도값을 참조해야하는 연산이 추가됨에 따라 연산량이 많아져 공차 함수 A보다 항상 느린 연산 결과를 확인하였다.

고밀도 골격형 데이터의 경우 골밀도가 높은 골조직을 많이 포함하는데, 골밀도가 높은 골조직은 의료 데이터 특성 상 중앙부에 많이 분포되어 있다. 이는 골조직이 피부나 근조직의 내부에 위치하기 때문이다. 고밀도 골격형 데이터의 경우 공차 함수 C에서 가장 높은 효율을 보인다. 공차 함수 C는 데

이터의 중앙부의 공차 값을 최소화하며, 데이터의 외면부로 갈수록 공차 값을 크게 주어 중앙부는 최대한 건너뛰지 않고, 외면부는 최대한 건너뛰도록 설계된 공차 함수이다. 따라서 고밀도 골격형 데이터의 경우 높은 골조직을 많이 포함하며, 이는 중앙부에 위치할 확률이 높다. 이에 따라 공차 함수 C가 가장 극적인 효과를 발휘하는 것을 확인하였다.

반면에 저밀도 골격형 데이터의 경우 공차 함수 A에서 가장 높은 효율을 보인다. 이는 공차 함수 F, G, H에서 오는 연산량의 한계에 따라 공차 함수 A의 연산량이 적어 가장 극적인 효과를 발휘한다. 따라서 골밀도가 높은 골조직이 많이 포함되어 있지 않은 경우나 중앙부에 골조직이 밀집되어 있지 않은 데이터의 경우가 저밀도 골격형 데이터에 해당되며, 공차 함수 A에서 최적의 공차 함수를 갖는다. 이는 제 5장에서 확인 가능하다.

제 5 장 실험

제 1 절 실험 환경

본 연구의 실험 CPU는 Intel i5-6600 @ 3.30GHz (4코어)와 GPU GeForce GTX 1060를 장착한 개인용 컴퓨터에서 실험하였다. 본 연구의 실험은 가시화 연산을 효율적으로 진행하기 위해, CUDA를 이용한 병렬 GPU 프로그래밍을 통해 제작하였으며, 개발 툴은 Visual Studio 2015, C++을 통해 구현하였다. CUDA는 9.0버전을 사용하여 제작하였으며, 최대 휘소 가시화와 빈 공간 도약 알고리즘, 그리고 빈 공간 도약 확장 부분에 적용시켰다. 실험에서 사용한 의료 데이터는 4종이며 [표 5-1]에 정리하였다. 데이터의 밀도 값은 2byte의 크기를 가지며 순차적으로 저장되어 있다.

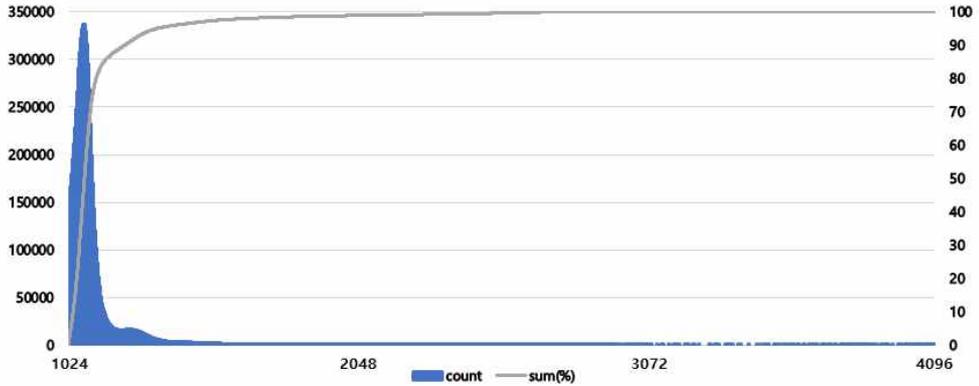
[표 5-1] 볼륨 데이터 세트

Dataset	Resolution	size(KB)
lower	512×512×1165	596,480
head	512×512×552	282,624
liver1	512×512×370	189,440
liver2	512×512×341	174,592

제 2 절 히스토그램 누적 합 곡선을 이용한 데이터 세분화

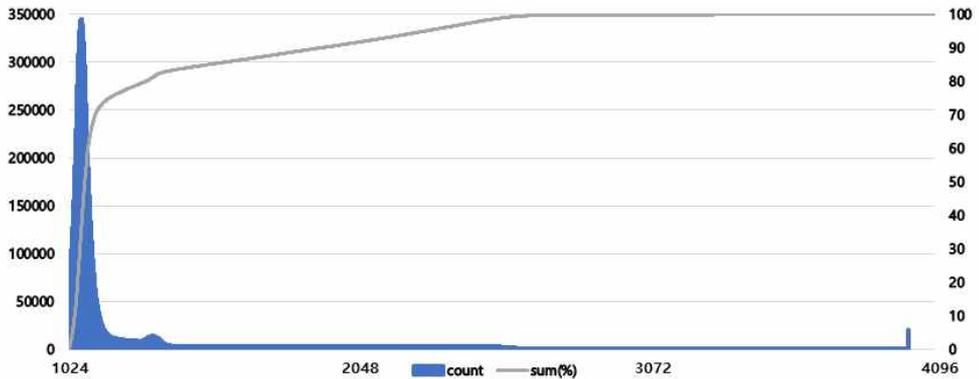
이번 절에서는 데이터를 세분화하기 위해 데이터의 밀도값을 히스토그램으로 출력하였다. 데이터에 따라 히스토그램을 도표하고, 고밀도 골격형 데이터와 저밀도 골격형 데이터를 분류하였다. 히스토그램은 공기층을 제외한 근조직과 피부, 골격에 대해서만 수행하였다. 공기층은 0~1024의 밀도값을 가진 데이터이다.

1) 히스토그램



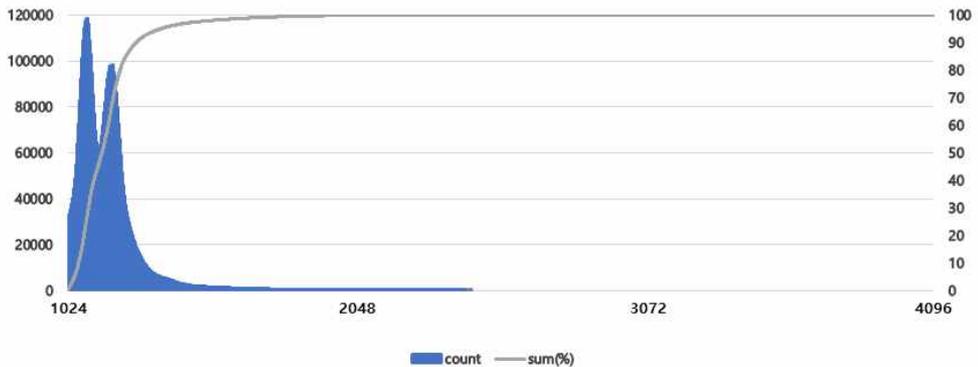
[그림 5-1] lower 데이터의 히스토그램

[그림 5-1]은 $512 \times 512 \times 1165$ 해상도의 다리데이터이다. 골격부 부근인 1200이상의 값들이 일정하게 4096까지 분포되어 있는 것이 특징이다. 따라서 히스토그램 누적 합의 값이 일정하게 증가하는 것을 보인다.



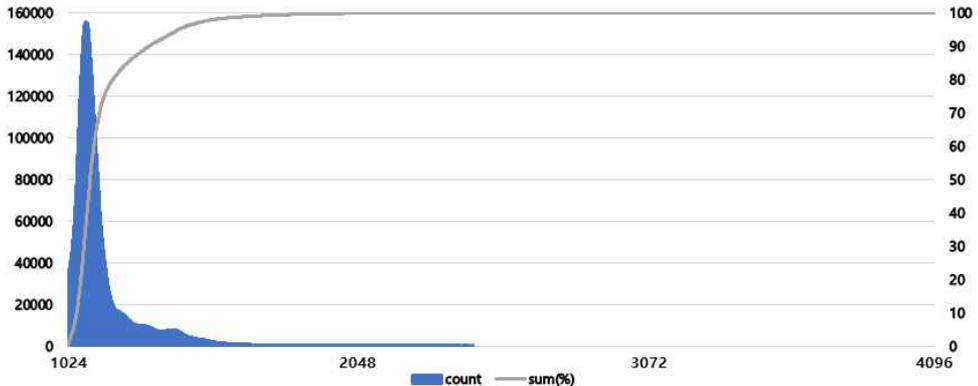
[그림 5-2] head 데이터의 히스토그램

[그림 5-2]은 $512 \times 512 \times 552$ 해상도의 머리 데이터이다. 다리 데이터와 마찬가지로 골격부 부근인 값들이 분포되어 있다. 다만 다리 데이터와는 달리 골격부 부근의 값들이 많이 분포하는 것을 확인 가능하다.



[그림 5-3] liver1 데이터의 히스토그램

[그림 5-3]는 512×512×370 해상도의 흉부 데이터이다. 흉부에서 간을 중심으로 촬영한 데이터이다. 다리 데이터와 머리 데이터와는 달리 골격부가 3000이상 부근에서는 존재하지 않는 것이 특징이다. 골격 구조의 강도가 비교적 약한 것을 확인 가능하다. 히스토그램 누적 합 그래프는 다리 데이터와 유사하다.



[그림 5-4] liver2 데이터의 히스토그램

[그림 5-4]는 512×512×341 해상도의 흉부 데이터이다. [그림 5-3]와 마찬가지로 흉부에서 간을 중심으로 촬영한 데이터이다. liver 데이터와 히스토그램 누적 합 그래프는 유사한 형태를 보인다.

4개의 데이터의 히스토그램에서 확인한 것은 의료데이터 모두 피부조직과 근조직인 1024이상의 값이 많이 분포된 것을 확인하였다. 따라서 의료데이터의 경우 골조직이 비교적 적은 것을 확인하였다. 머리 데이터의 경우 골조직이 30%정도를 포함하는 것을 제외하고 3개의 데이터는 20%이하인 것을 확

인하였다.

2) 히스토그램 누적 합 백분율에 따른 데이터 세분화 분석

[표 5-2] 히스토그램 누적 합 백분율에 따른 밀도값

백분율	lower	head	liver1	liver2
100%	4094	4000	2453	2456
99%	2126	2570	1681	1671
98%	1658	2490	1531	1540
97%	1498	2422	1440	1483
96%	1405	2355	1389	1444
95%	1339	2288	1355	1417
94%	1302	2221	1328	1396
93%	1278	2151	1307	1375
92%	1259	2077	1291	1353
91%	1242	1999	1278	1331
90%	1224	1920	1267	1312
89%	1206	1840	1258	1295
88%	1189	1761	1250	1279
87%	1173	1683	1243	1264
86%	1160	1602	1237	1249
85%	1150	1521	1231	1235
84%	1142	1442	1226	1224
83%	1136	1372	1222	1213
82%	1131	1338	1218	1203
81%	1127	1318	1214	1194
80%	1123	1297	1211	1185

[표 5-2]는 히스토그램 누적 합 백분율에 따른 밀도 값을 수치화한 표이다. 각 데이터에 따라 백분율이 도달하는 시점에서의 밀도 값을 확인 가능하다. 밀도 값이 1300이상인 부분이 데이터 따라 다르지만 백분율이 90% 이상일 경우 1300이상을 도달하였다.

head 데이터와 liver2 데이터는 밀도 값이 각각 81%와 90%에서 1300을 넘는 것을 확인하였다. 따라서 고밀도 골격형 데이터로 정의한다. Head 데이터의 경우는 기본 밀도 값 수치 자체가 높은 것을 확인할 수 있는데, 이는 인체의 머리 부분의 골밀도가 가장 높기 때문이다. liver2의 경우는 피부층과 근

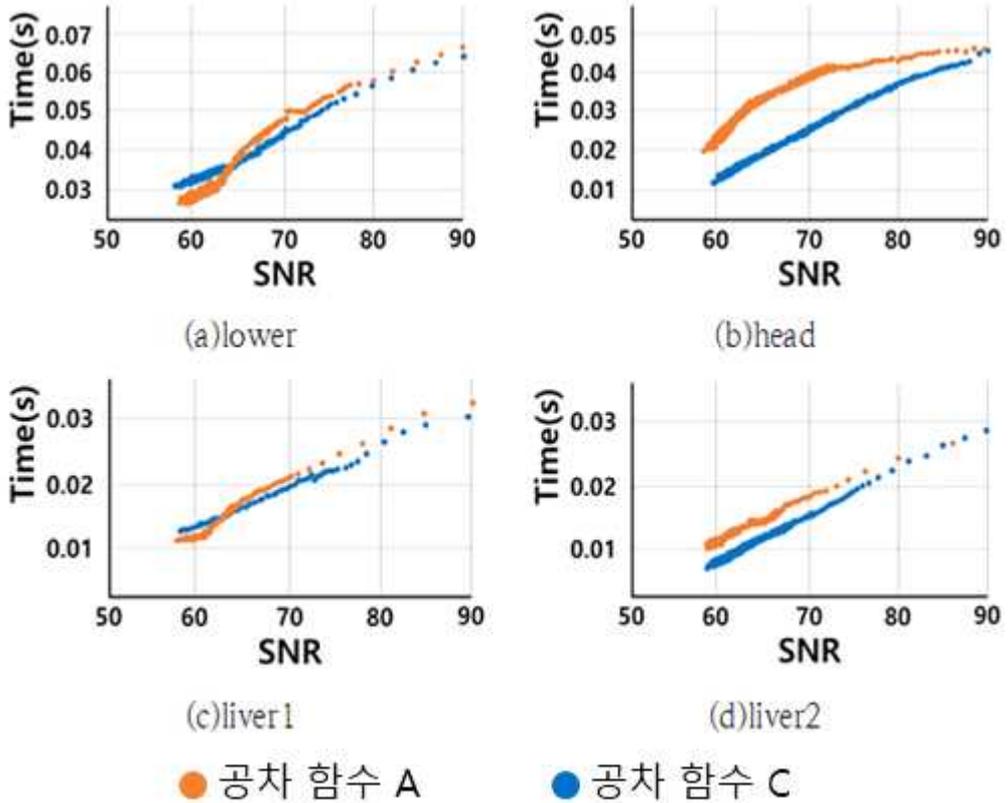
조직이 많이 분포되어 있지만 골밀도 또한 고루 분포되어 있다.

반면에 저밀도 골격형 데이터는 lower 데이터와 liver 데이터이다. Lower 데이터의 경우 신체의 가장 큰 부위이고, 히스토그램에서 확인했듯이 높은 값의 골밀도가 존재하지만 근조직과 피부조직이 대체로 많이 분포되어 있어 94% 지점에서 1300 값을 넘게 된다. Liver의 경우에는 골조직 분포가 뚜렷하게 많이 분포되어 있지만 1300이상의 높은 값의 골밀도는 93%에서 도달하였다.

제 3 절 데이터 세분화에 따른 속도 및 화질 비교

이번 절에서는 2절에서 확인한 데이터를 토대로 시간과 화질에 따라 비교하여 보았다. 공차 함수의 공차 값을 등차적으로 증가시키며 시간과 화질의 값을 비교하였다. 화질의 손상 수치 측정은 신호 대 잡음비(Signal-to-Noise Ratio, SNR)를 기준으로 실험하였다. 화질은 신호 대 잡음비 값 기준으로 60이 감안할 수 있는 수준의 영상이므로, 신호 대 잡음비 60을 기준으로 실험하였다. 이는 화질 손실을 감안할 수 있는 선까지 하고 시간측정을 할 경우 최적의 영상을 얻을 수 있기 때문이다.

공차 함수 A, B, C, D, E, F, G, H를 모두 실험하였고, 고밀도 골격형 데이터는 공차 함수 C가 효과적이고, 저밀도 골격형 데이터는 공차 함수 A가 효과적인 것을 확인하였다. 공차 함수 F, G, H의 경우 모든 데이터에 대해 A보다 효과적이지 못했다. 밀도값의 비율을 기준으로 한 공차 함수 B의 경우가 가장 좋지 못한 결과를 가져왔다. 공차 함수 D와 E의 경우에는 연산량이 많아 공차 함수 C보다 모든 데이터에 대해 효과적이지 못했다.

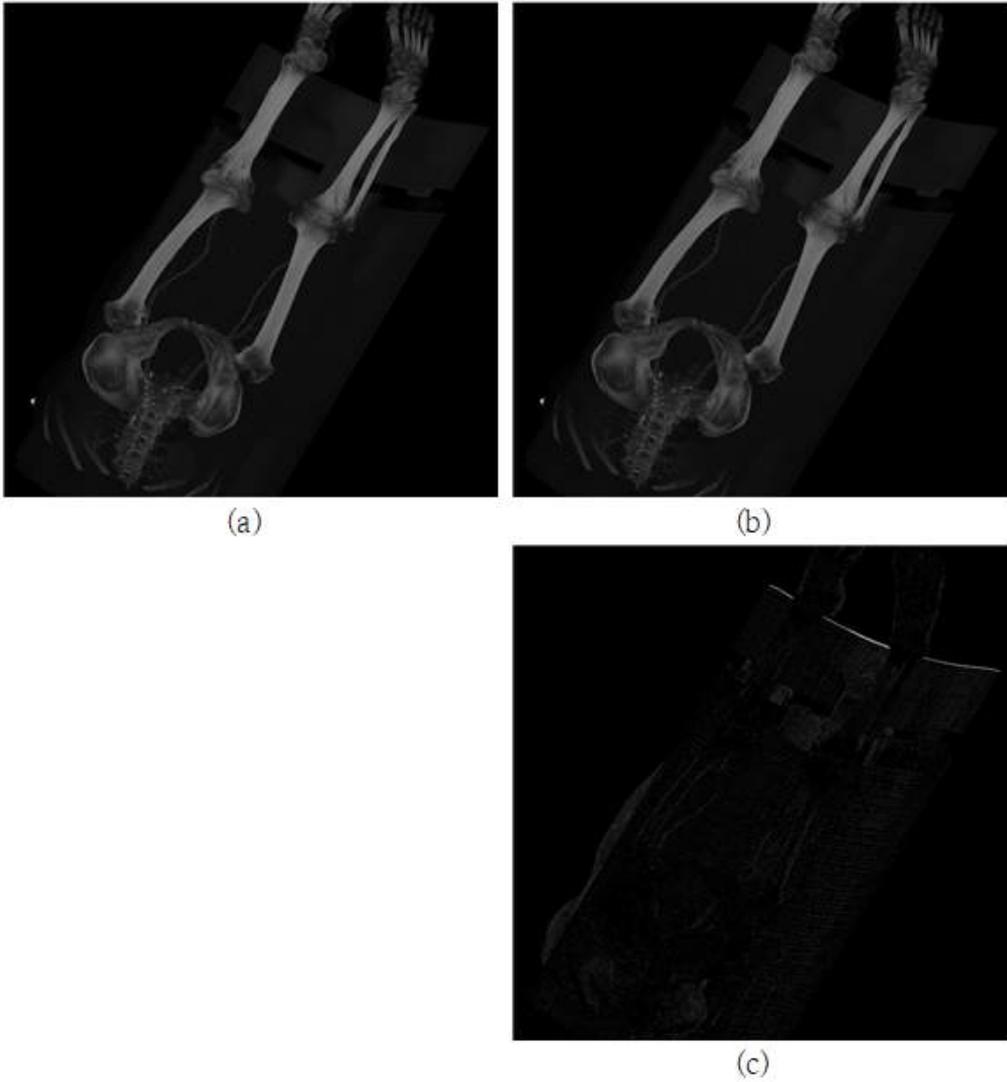


[그림 5-5] 시간-신호 대 잡음비 그래프

[그림5-5]는 시간에 따른 신호 대 잡음비를 그래프로 나타내었다. [그림 5-5]의 (a), (b), (c), (d)는 각각 lower, head, liver1, liver2 데이터이다. 신호 대 잡음비 60이상을 기준으로 고밀도 골격형 데이터인 head와 liver2는 공차 함수 C가 효율적인 것을 확인하였다. 저밀도 골격형 데이터인 lower와 liver1의 경우 공차 함수 A가 신호 대 잡음비 60에서 더 빠른 시간에 가시화가 가능하다. 신호 대 잡음비 70이상에서는 공차 함수 C가 더 빠른 것을 확인할 수 있는데, 이는 의료 데이터 특성상 모두 중앙에 분포되어 있어 일정 부분 연산적 우위를 갖지만, 신호 대 잡음비 60 부근에서는 공차 함수 A가 더 효과적인 것을 확인하였다. 따라서 고밀도 골격형 데이터는 공차 함수 C에서 최고의 효율을 보이고, 저밀도 골격형 데이터는 공차 함수 A에서 최고의 효율을 보인다.

제 4 절 데이터 세분화에 따른 영상 비교

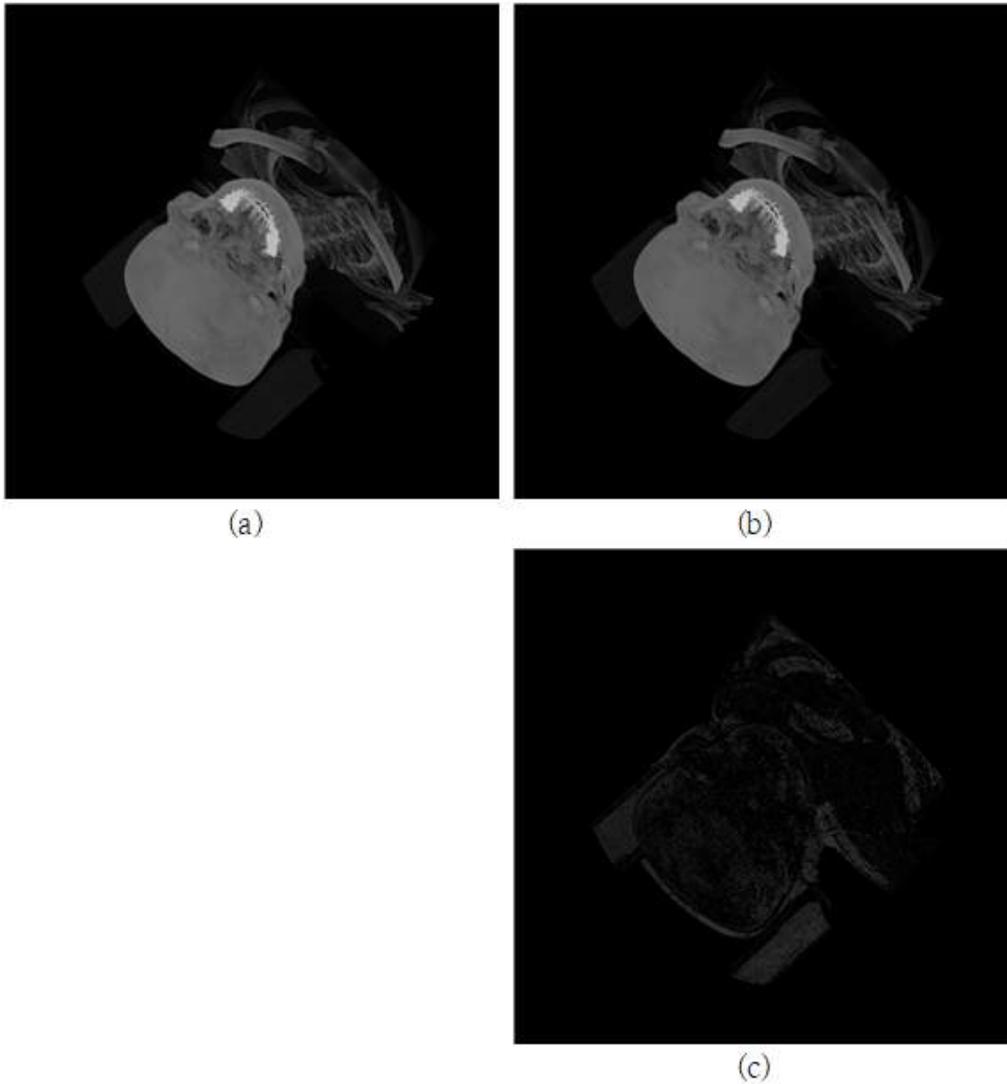
본 절에서는 빈 공간 도약을 실행한 영상과 공차 함수를 이용하여 개선된 빈 공간 도약을 실행한 영상에 대해 영상 차이를 확인하였다. 영상 가시화는 CUDA를 이용한 GPU 병렬처리를 사용하여 수행하였다. 영상의 크기는 1000×1000 픽셀의 출력 영상이다. 일반적인 CT데이터를 사용하여 0~4095의 밀도값을 갖는 데이터를 0~255크기의 값으로 출력하였다. 빈 공간 도약을 실행한 영상과 공차 함수를 이용한 개선된 빈 공간 도약 영상의 결과를 이용하여 차영상을 제작해 비교하였다.



[그림 5-6] lower 데이터의 가시화 영상

다리 데이터는 저밀도 골격형 데이터이다. 이는 2절과 3절에서 확인 가능하며 히스토그램 누적 합 94%에서 1302의 밀도값으로 1300을 넘게 된다. [그림 5-6] (a)는 빈 공간 도약만 실행한 영상이며, 가시화 시간은 0.3초이다. 저밀도 골격형 데이터를 앞 절에서 확인하여 공차 함수 A를 이용하여 [그림 5-6] (b)의 영상을 출력하였다. [그림 5-6] (b)의 영상의 경우 신호 대 잡음비 60에서 0.028초의 시간이 소요되었다. 약 10배 빠른 가시화를 확인할

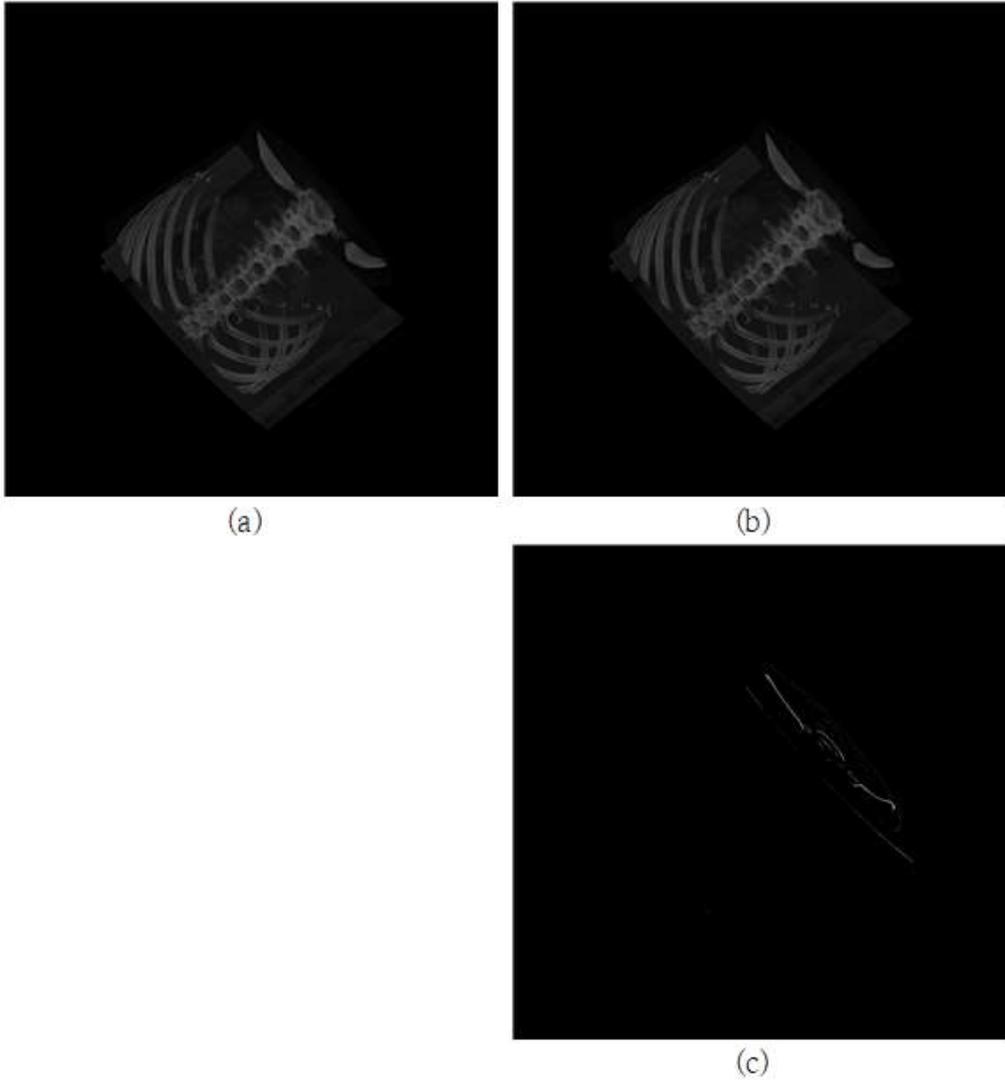
수 있다. [그림 5-6] (c)는 [그림 5-6] (a) 영상과 [그림 5-6] (b) 영상의 차 영상 결과이다. 차영상의 경우 $|a-b| \times 10$ 의 수식을 이용하여 차영상을 출력하였다.



[그림 5-7] head 데이터의 가시화 영상

머리 데이터는 앞 절에서 확인한 결과 고밀도 골격형 데이터로 세분화 되었다. 히스토그램 누적 합이 81%에서 1312로 밀도값 1300을 넘게 된다. [그

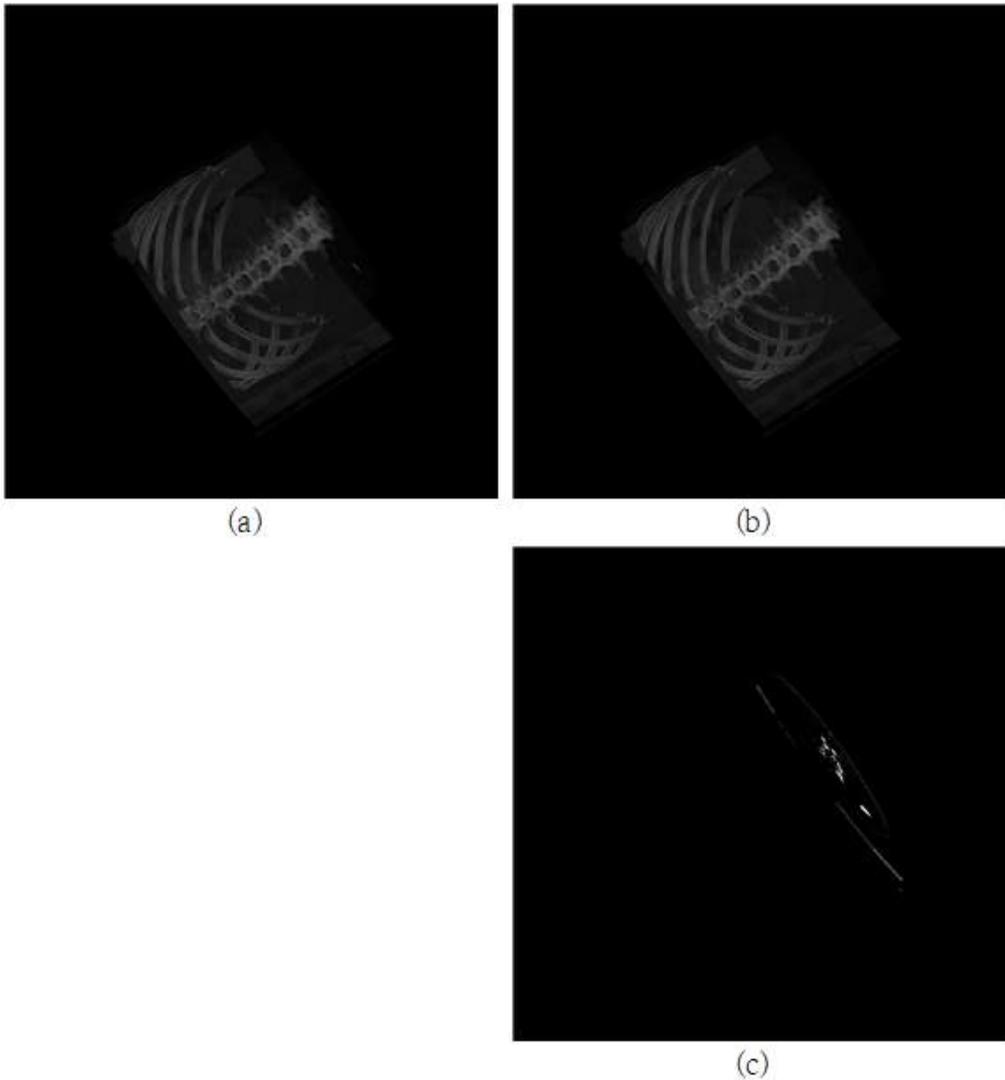
림 5-7] (a)는 [그림 5-6] (a)와 마찬가지로 빈 공간 도약만 실행한 영상이며 가시화 시간은 0.19초가 소요되었다. [그림 5-7]의 (b)의 경우는 공차 함수 C를 사용하였고, 신호 대 잡음비 60에서 0.012초가 소요되었다. 약 16배 이상의 빠른 가시화를 확인하였다. 머리 데이터의 경우 다리 데이터에 비해 중앙부에 더 집중되어 있는 데이터 구조를 갖고 있고, 공차 함수 C를 이용하여 더 극적인 효과를 얻은 것을 확인하였다. [그림 5-6]와 마찬가지로 (a)는 빈 공간 도약, (b)는 공차 함수 C를 이용한 신호 대 잡음비 60의 영상, (c)는 $|a-b| \times 10$ 의 수식을 이용한 차영상의 영상 출력 결과이다.



[그림 5-8] liver1 데이터의 가시화 영상

흉부 데이터의 출력 영상은 [그림 5-8]과 같다. 앞 절에서 확인할 수 있듯이 백분율 93%에서 1307의 밀도값이 도달하여 저밀도 골격형 데이터임을 확인하였다. [그림 5-6]와 마찬가지로 (a)는 빈 공간 도약, (b)는 공차 함수 A를 이용한 신호 대 잡음비 60의 영상, (c)는 $|a-b| \times 10$ 의 수식을 이용한 차영상의 영상 출력 결과이다. [그림 5-8] (a)의 경우 영상 출력에 소요된 시간은 0.14초이다. [그림 5-8] (b)는 영상 출력에 소요된 시간은 0.011s

이다. 약 12배 가량 빨라진 것을 확인하였다.



[그림 5-9] liver1 데이터의 가시화 영상

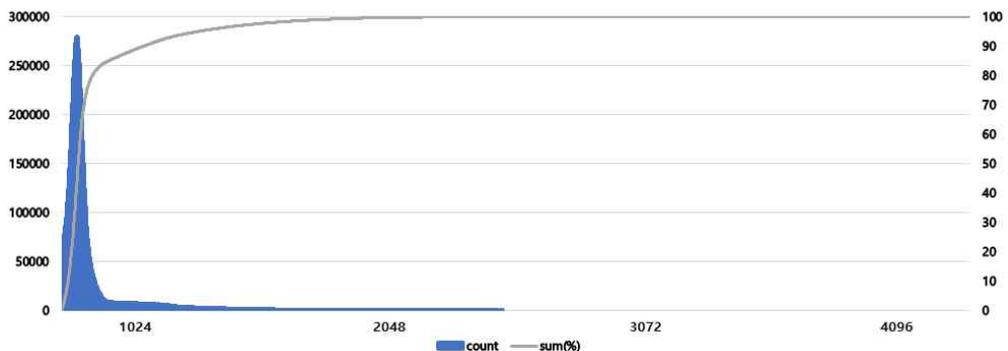
두 번째 흉부 데이터의 출력 영상은 [그림 5-9]과 같다. 백분을 90%에서 1312의 밀도값으로 고밀도 골격형 데이터임을 확인하였다. [그림 5-7]와 마찬가지로 (a)는 빈 공간 도약, (b)는 공차 함수 C를 이용한 신호 대 잡음비 60의 영상, (c)는 $|a-b| \times 10$ 의 수식을 이용한 차영상의 영상 출력 결과

과이다. [그림 5-9] (a)의 경우 소요 시간은 0.13초이고, [그림 5-9] (b)의 경우 0.008초로 약 16배 가량 빨라진 것을 확인하였다.

영상 출력 결과를 통해 알 수 있듯이 실제로 일반적인 빈 공간 도약의 출력 영상과 개선된 빈 공간 도약의 출력 영상은 육안으로는 판별이 불가능할 정도로 영상 화질의 손실이 거의 없다. 반면에 속도 측면에서는 약 10배~16배 가량 차이가 나는 것을 확인하였다. 이는 신호 대 잡음비 60의 경우이지만, 연산 속도가 10배 이상 차이나는 것은 낙관적인 결과이다. 물론 화질의 손실을 어느 정도 감안한다는 한계가 있지만, 데이터의 크기가 방대한 의료 데이터의 경우 영상 출력 시간을 10배 이상 향상 시킨 결과는 큰 이점으로 작용할 수 있다.

제 5 절 데이터를 이용한 선택법 검증

이 절에서는 본 연구의 선택법을 이용하여 새로운 데이터를 통해 고밀도 골격형 데이터와 저밀도 골격형 데이터를 분류하고, 이에 따른 영상 출력 결과를 확인하였다. 새로운 흉부 데이터는 $512 \times 512 \times 528$ 의 해상도를 가지고 있는 의료 데이터이다. 데이터의 크기는 270,336KB이다. 먼저 데이터를 분류하기 위해 히스토그램과 히스토그램 누적 합 그래프를 출력하였다.

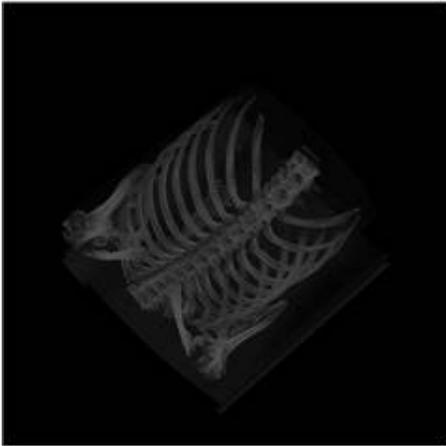


[그림 5-10] 흉부 데이터의 히스토그램

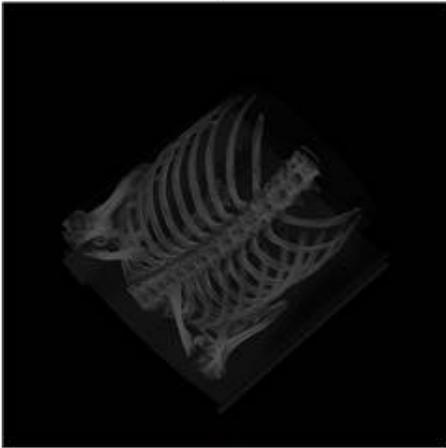
히스토그램은 흉부 데이터인 liver1, liver2데이터와 유사한 형태를 지닌

것을 확인하였다.

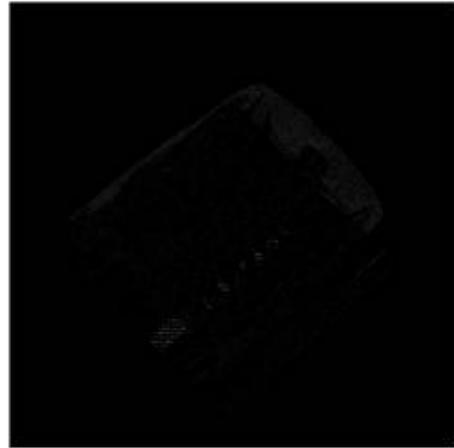
고밀도 골격형 데이터와 저밀도 골격형 데이터를 분류하기 위해 누적 합 백분율을 확인 결과 90%에서 1302의 밀도값에 도달하는 것을 확인하였고, 고밀도 골격형 데이터로 분류하였다. 90% 이상에서는 liver1, liver2와 비슷한 결과를 보였고, 2513의 밀도값에서 히스토그램 누적 합 백분율이 도달하였다. 4절에서와 마찬가지로 영상을 출력하여 비교하였다. 공차 함수 A와 C를 모두 적용시켰으며 신호 대 잡음비는 60인 영상을 출력하였다.



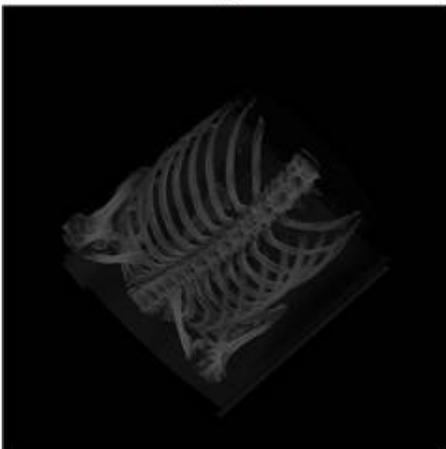
(a)



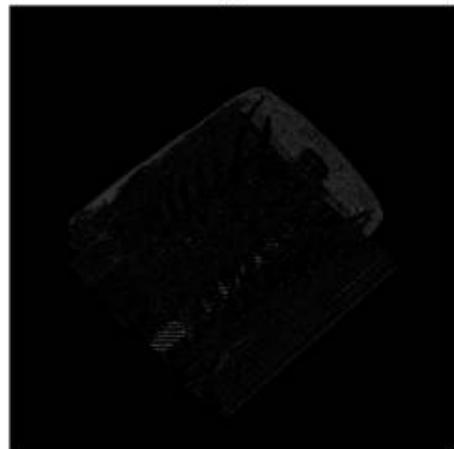
(b)



(c)



(d)



(e)

[그림 5-11] 흉부 데이터의 가시화 영상

[그림 5-11]의 (a)는 기존의 빈 공간 도약 알고리즘으로 출력하였고, 영상 출력에 소요된 시간은 0.18초였다. [그림 5-11]의 (b)는 공차 함수 C를 이용하였고, [그림 5-11]의 (c)는 $|a)-(b)| \times 10$ 수식을 이용한 차영상이다. [그림 5-11]의 (d)는 공차 함수 A를 이용하였고, [그림 5-11]의 (e)는 $|a)-(d)| \times 10$ 의 차영상 결과이다. 공차 함수 C를 이용한 [그림 5-11] (b)의 소요 시간 0.02초로 신호 대 잡음비는 64였다. [그림 5-11] (d)의 경우 같은 소요 시간인 0.02초에서 신호 대 잡음비는 60이었다. 이를 통해 고밀도 골격형 데이터는 공차 함수 C에서 가장 효과적인 것을 확인하였다.

제 6 장 결론 및 향후 개선 방안

본 논문은 최대 휘소 투영 볼륨 가시화를 위한 가속 방법을 제안함과 동시에 데이터를 분류하여 효율적인 가시화 영상을 얻는 방법을 제안하였다. 최대 휘소 투영 영상 출력을 가속화하기 위해 사용되던 기존의 빈 공간 도약 방법은 실제로 필요하지 않은 값들을 건너뛰지 못하는 한계를 포함하였다. 이에 본 논문에서 어느 정도의 화질 손실을 감안하고 가속화의 이점을 취하는 방법을 제안하였다. 5장의 여러 데이터의 실험을 통해 화질과 속도의 관계를 확인하였고, 올바른 공차 함수를 사용하는 것에 따라 얻는 이점을 최대화할 수 있다는 것을 확인하였다. 의료 영상의 경우 오차가 전혀 없는 고화질의 영상을 필요로 하는 경우가 많으나, 본 연구는 이를 감안하고 10배~16배 가량 빨라지는 알고리즘 설계하였으며, 적정 화질을 유지하는 것을 확인하였다. 또한 화질 손실의 차가 감안할 수 있는 선이라는 점에서 더 큰 이점이 있다. 화질 손실의 경우 차영상을 통해 확인하였으며 이는 육안으로는 확인 불가능할 정도의 차이이다. 육안으로는 확인할 수 없는 정도의 미미한 차이인 점을 감안했을 때 가시화 시간이 10배~16배 가량 빨라지는 가시화는 극적인 효과를 기대할 수 있는 가시화 이다.

비록 본 논문에서 제시한 공차 함수는 3종류로 적지만, 향후 공차 함수를 더 개발하여 비교가 가능하고, 개선된 공차 함수가 있다면 더욱 더 효과적인 가시화가 가능하다. 또한 데이터를 더 세분화하여, 많은 데이터에 따라 최적의 공차 함수를 개발한다면, 이는 볼륨 가시화에서 큰 효과를 가져다 줄 수 있다. 볼륨 가시화에 있어 빈 공간 도약 알고리즘과 개선된 빈 공간 도약 알고리즘은 모두 적용될 수 있다. 따라서 의료 데이터에 국한되지 않고 어떤 볼륨 데이터에서도 가속화 진행이 가능하다.

또한 본 논문에서는 제시한 데이터의 세분화와 선택법을 히스토그램을 통해 미리 분석하였다. 히스토그램을 통해 얻어진 데이터들을 통해 여러 데이터에서 추가적으로 특징을 찾는다면, 머신 러닝을 이용해 데이터를 자동으로 분류하는 것도 가능하다. 알맞은 공차 함수를 선택하는 방법 또한 무한히 개선

가능하다. 데이터의 특징을 찾고, 머신 러닝을 이용하여 볼륨 데이터들을 분류한다면, 볼륨 데이터 가속화에서 10배~16배가 아닌 더 극적인 효과를 가져올 수 있으며, 의료 데이터를 실시간으로 분석하는 것 또한 가능하다.

3장에서 소개한 공차 함수 다양성 또한 무한히 발전할 수 있다. 데이터에 특징을 올바르게 찾고 데이터에 따라 최적의 공차 함수를 선택한다면, 연산에 필요한 데이터를 최소화하여 더 극적인 효과를 발휘할 수 있다. 따라서 연산량이 많은 의료 영상이나, 볼륨 데이터를 토대로 한 영상에 있어 큰 이득을 취할 수 있다.

참 고 문 헌

1. 국외문헌

- Sabella, P., (1988). A Rendering Algorithm for Visualizing 3D Scalar Fields. 『ACM SIGGRAPH 1988 Proceedings of the 15th annual conference on Computer graphics and interactive techniques』, 51-58.
- Schreiner, S., Galloway, RL., (1993). A Fast Maximum-Intensity Projection Algorithm for Generating Magnetic Resonance Angiograms. 『IEEE Transactions on Medical Imaging』, 12(1), 50-57.
- Mroz, L., König, A., Gröller, E., (1999). Real-Time Maximum Intensity Projection. 『Data Visualization '99, Proceedings of the Joint EUROGRAPHICS - IEEE TCCG Symposium on Visualization』, 135-144.
- Fang, L., Wang, Y., Qiu, B., Qian, Y., (2002). Fast Maximum Intensity Projection Algorithm Using Shear Warp Factorization and Reduced Resampling. 『Magnetic Resonance in Medicine』, 47, 696-700.
- Kye, H., Lee, S., Lee, J., (2017). CPU-based real-time maximum intensity projection via fast matrix transposition using parallelization operations with the AVX instruction set. 『Multimedia Tools and Applications』, 77(12), 15971-15994.
- Levoy, M., (1988). Display of surfaces from volume data. 『IEEE Computer graphics and Applications』, 8(3), 29-37.
- Kiefer, G., Lehmann, H., Weese, J., (2006). Fast Maximum Intensity Projections of Large Medical Data Sets by Exploiting Hierarchical Memory Architectures. 『IEEE Transactions on Information Technology in Biomedicine』, 10(2), 385-394.
- Kye, H., Kim, J., (2011). Acceleration techniques for GPGPU-based Maximum Intensity Projection. 『Journal of Korea Multimedia Society』, 14(8), 981-991.

- Kye, H., Sohn, B., Lee, J., (2012) Interactive GPU-based maximum intensity projection of large medical data sets using visibility culling based on the initial occluder and the visible block classification. 『Computerized Medical Imaging and Graphics』, 36(5), 366–374.
- Mroz, L., Hauser, H., Gröller, E., (2000) Interactive High-Quality Maximum Intensity Projection. 『Computer Graphics forum』, 19(3), 341–350.
- Levoy, M., (1990). Efficient ray tracing of volume data. 『ACM Transactions on Graphics』, 9(3), 245–261.
- Kye, H., Jeong, D., (2008). Accelerated MIP based on GPU using block clipping and occlusion query. 『Computers & Graphics』, 32(3), 283–292.
- Kwon, O., Kang, S., Kim, S., Kim, Y., Shin, Y., (2015). Maximum intensity projection using bidirectional compositing with block skipping. 『Journal of X-ray science and technology』, 23(1), 33–44.
- Mora, B., Ebert, DS., (2005). Low-Complexity Maximum Intensity Projection. 『ACM Transactions on Graphics』, 24(4), 1392–1416.
- Pekar, V., Hempel, D., Kiefer, G., Busch, M., Weese, J., (2003). Efficient visualization of large medical image datasets on standard PC hardware. 『Proceedings of Joint EUROGRAPHICS – IEEE TCVG Symposium on Visualization』, 135–140.
- Kim, K., Park, H., (2001). A fast progressive method of maximum intensity projection. 『Computerized medical imaging and graphics』, 25(5), 433–441.

ABSTRACT

Efficient maximum intensity projection algorithm and algorithm selection

Han, Chul-Hee

Major in Information System Engineering

Dept. of Information System Engineering

The Graduate School

Hansung University

Maximum intensity projection (MIP) is one of the common visualization techniques in medical image processing. There are various researches using the maximum projective projection, among which representative studies are acceleration and improvement of image quality. This paper presents an algorithm for accelerating using the empty space leap, which is one of the acceleration methods in the maximum imaginary projection. We also studied how to improve the empty space leap algorithm effectively. We present new methods using existing void space leaps, and show how to select new methods effectively. In this paper, we propose an efficient acceleration algorithm by introducing a method to skip unnecessary part and image and also the part that does not have a big influence. present. Improved new methods in the empty space leap algorithm show improved speed and various image quality,

and this study presents a method to select the best image by comparing them. Experimental results show the accelerated images suitable for the speed required by the user using various tolerance functions. In order to select the proposed method effectively, we propose a method to select the image with relatively high quality in a short period of time by experimenting with the maximum imaginary projection through several medical data.

KEYWORD: Medical Image Rendering, Volume Rendering, Maximum intensity projection, block skipping