

모바일을 기반으로 하는 적응적인 렌더링 영상 처리

주 헌 식[†] · 김 하 진^{††}

요 약

이 논문에서는 모바일을 기반으로 하는 사진 트리 영상 렌더링을 통하여 효율적인 PDA용 전자의무기록(EMR ; Electronic Medical Record) 차트를 제안하였다. 볼륨 렌더링을 위하여 최종 영상 공간 대신에 중간 영상 공간 공간 알고리즘을 사용하였고, 점 대 점 매핑으로 발생하는 Hole의 문제점을 해결하였다. 델타 트리 기반의 quad 트리를 사용함으로써 볼륨 표현에 효과적이었고, 압축 효과가 높았다. 볼륨 렌더링을 사용하여 렌더링의 시간을 줄일 수 있었고, 영상기반 렌더링(image-based rendering)을 이용한 PDA를 사용하여 질적 향상과 효율화를 얻게 되었다.

A Adaptive Rendering Image Processing for Based on the Mobile

Heon-Sik Joo[†] · Ha-Jine Kim^{††}

ABSTRACT

This paper presents an EMR(Electronic Medical Record) chart for efficient PDA through the quad tree image rendering based on the mobile. Using the intermediate image space algorithm instead of the final one for volume rendering, we have solved the problems of the holes coming from the point-to-point mapping. The quad-tree based on the delta-tree efficiently represents volume expressions and results in higher compression effects. With the volume rendering, we can decrease the rendering time and get a higher quality and efficiency for PDA through image based rendering.

키워드 : 모바일(Mobile), 영상기반 렌더링(Image-based Rendering), 이미지 프로세싱(Image-processing)

1. 서 론

최근 인터넷을 기반으로 하는 웹의 급속한 발전으로 다양한 정보들을 교환하고 공유 할 수 있게 되었는데 개방된 환경에 더 큰 가치를 지니게 되었다. 컴퓨터시스템의 흐름은 과거의 메인 프레임을 위주로 하는 중앙 집중 처리 방식에서 벗어나 데이터와 프로그램을 적절한 곳으로 분산 배치하여 처리하는 클라이언트/서버 컴퓨팅 방식으로 전환되어 왔다. 이제 인터넷의 등장과 함께 네트워크에 의한 이동 컴퓨팅 환경(mobile computing environment)으로 효과적인 정보시스템 구축이 필요 되었고 지역 네트워크를 기반으로 운영되어 오던 기존의 시스템들이 데이터 공유의 필요에 의해 개방형시스템으로 전환이 요구되고 있다. 정보화 사회에 들어선 후 가장 비약적인 발전을 이룬 분야는 인터넷과 모바일 통신 분야이다[14]. 초기 모바일 인터넷 서비스의 경우 단순히 텍스트를 기반으로 제공된 서비스 형태

에 불과하였다. 하지만 다양한 어플리케이션들을 실행시킬 수 있는 플랫폼 기반의 기술들이 개발되고 서비스 환경이 컬러 화되면서 고품질의 멀티미디어 콘텐츠들을 제공할 수 있게 되었다[21, 23]. 최근의 개인 정보 단말(PDA : Personal Digital Assistant)은 성능의 향상과 더불어 컴퓨터와 무선 통신 기술이 효과적으로 연계되면서 언제, 어디서든지 이동 환경에서 데이터의 교환 및 처리가 가능하게 되었다. PDA 기기에서 상용하기 위한 전문가 시스템 개발이 이루어지고 있다. 본 논문에서는 시각적인 영상 정보들을 영상렌더링(image-based rendering)[2, 5, 7-9]을 이용한 볼륨 데이터로부터 다양한 응용에 필요한 PAD용 전자의무기록(EMR, Electronic Medical Record) 차트를 사용함으로써 효율적인 질적 향상과 경영 효율화를 얻을 수 있다.

2. 관련 연구

시각적인 영상 정보들을 받아들여 볼륨 렌더링을 통하여 일정한 형태가 있는 오브젝트(object)의 2차원 단면 영상들을 기본으로 3차원 그래픽 이론들을 적용시켜 3차원 영상으로

[†] 정 회 원 : 삼육의병대학 컴퓨터정보과 교수

^{††} 정 회 원 : 아주대학교 정보통신대학 교수

논문접수 : 2003년 7월 21일, 심사완료 : 2003년 10월 6일

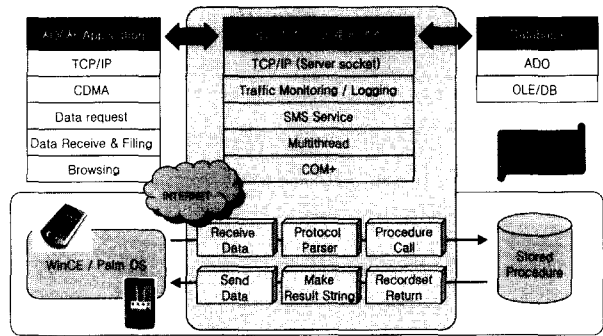
표현한다[10]. 볼륨 렌더링은 3차원 공간의 볼륨 데이터로부터 다양한 응용에 필요한 시각적인 영상정보를 추출하여 사용한다. 볼륨렌더링은 연속적인 2차원 슬라이스들을 기본으로 하기 때문에 슬라이스 영상들을 조합하는 식에 따라 여러 가지 방법으로 분류한다.

본 논문에서는 영상기반 렌더링(image-based rendering)을 이용하여 특별한 하드웨어 없이 볼륨 데이터로부터 크게 효율적인 영상을 생성할 수 있는 알고리즘을 제안한다. 볼륨 렌더링은 크게 표면 렌더링(surface rendering)과 직접 볼륨 렌더링(direct volume rendering)으로 분류한다[1, 11, 12]. 표면 렌더링은 영상을 생성하기 위하여 볼륨으로부터 다각형(polygon)들을 추출한다. 직접 볼륨 렌더링은 볼륨 데이터를 다각형으로 변환하지 않고 직접 영상 공간에 투영하여 사용한다. 직접 볼륨 렌더링은 속도의 향상을 위하여 많은 방법들이 개발되었고, 영상의 효율을 높이기 위하여 다중프로세서, 그래픽 하드웨어 등의 특수 장비를 이용할 수 있다. 볼륨 내에 존재하는 일관성(coherence)을 팔진 트리와 같은 특수한 자료 구조에 저장하여 속도 향상을 시도하는 방법들도 있다. 또한 빠른 볼륨 렌더링을 위하여 최종 영상 공간(final image space) 대신에 중간 영상 공간(intermediate image space)를 사용하는 알고리즘도 제안되었다[15, 20]. 이러한 방법들을 이용하여 직접 볼륨렌더링에 걸리는 시간이 줄어들었지만 볼륨이 커지면 성능은 떨어진다. 일반적으로 직접 볼륨 렌더링을 수행할 때 한 화소에 여러 개의 복셀이 누적되어 계산된다. 영상 기반 렌더링은 임의의 위치에서 영상을 볼륨의 크기에 관계없이 일정한 시간 내에 얻을 수 있는 방법이다. 영상기반 렌더링이란 미리 얻어진 영상들의 집합으로부터 새로운 위치에서 영상을 생성해 내는 방법이다. 영상기반 렌더링은 다음과 같이 세 개의 영역으로 분류한다[17]. 근사 영상(images as approximations)[3, 19], 데이터베이스영상(images as databases)[12, 16], 모델영상(images as models)[4, 18]로 분류한다. 참조 영상(reference images)으로부터 원하는 영상을 생성할 수 있는 뷰 보간(view interpolation)알고리즘이 있다. 뷰 보간에서는 참조 영상들은 완전한 모델로 간주되는데, 문제는 중첩(overlap)과 구멍(hole)이다. 중첩을 해결하기 위해서는 뷰 보간에서는 지 버퍼(z-buffer)와 같은 가시면 결정(visible surface determination) 기법을 사용하면 된다. 영상에 존재하는 구멍(hole)에 대해서는 보간이나 포-코너 매핑(four-corner mapping)방법을 사용한다. 영상을 데이터베이스로 사용할 경우에 인접한 영상들 사이에는 많은 중복이 존재한다[6]. 이러한 중복을 없애고 효율적인 영상을 표현하기 위해서는 델타-트리(Delta-Tree)기법을 제안한다. 객체의 효율적인 표현은 공간의 절략뿐 아니라 렌더링 시간도 단축시킬 수 있다. 델타-트리의 각 노드들은 각 점에서 만들어

진 영상들을 저장한다. 그리고 각 노드들간에 중복되는 정보들을 삭제함으로써 압축 효과를 얻을 수 있다. 본 논문에서는 볼륨 렌더링 영상과 비슷한 영상을 만들어 낼 수 있는 효율적인 영상 기반 렌더링 알고리즘을 제시하였고, 이 알고리즘은 최종 영상 공간 대신에 중간 영상 공간을 사용하여 효율을 높였으며, 점 대 점 매핑(point to point mapping)으로부터 발생하는 구멍의 문제점을 해결하였다. 또한 효율적인 객체의 표현을 위하여 델타-트리에 기반을 둔 새로운 사진 트리(quad tree)를 제안한다.

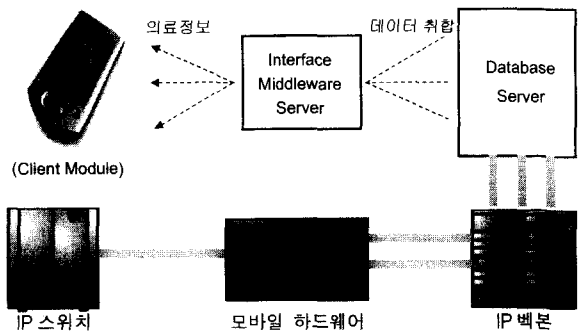
3. 시스템 구조

3.1 구조



(그림 1) EMR 시스템 구조

(그림 1)은 볼륨렌더링을 이용하기 위한 EMR 차트 시스템과 각 구성 컴포넌트들이다. 클라이언트에 해당하는 PDA는 무선 모바일이며 구성은 크게 3부분으로 구성되었는데 ① 사용자가 시스템을 사용할 수 있도록 하드웨어에 접속할 수 있는 사용자 Application 부분 ② 사용자 어플리케이션과 데이터베이스와 송수신을 할 수 있는 범용 통신 미들웨어 엔진(Sever) 부분 ③ 의무 기록 차트의 모든 것을 구성하여 지원하는 데이터베이스 부분으로 구성하고 있다. 사용자 Application 부분은 PDA를 가지고 직접 사용하는 부분으로 범용 통신 미들웨어 엔진과 송수신하며 어플리케이션으로 사용된다. 범용 통신 미들웨어 엔진 부분은 사용자 어플리케



(그림 2) Client Module

이전의 요청에 반응하여 데이터베이스에서 필요한 자료를 빠르게 송수신한다. 데이터베이스 부분은 EMR에 필요한 모든 데이터를 저장하여 어플리케이션의 요청에 범용통신 미들웨어 엔진을 통하여 요청한 자료를 송수신하며 지원한다.

(그림 2)는 EMR 시스템을 Client Module를 나타낸 것으로 인터페이스 비들웨어 서버를 중심으로 클라이언트에 해당하는 모바일과 데이터베이스 서버 간의 필요한 영상 의료 정보들을 제공한다. 모바일 기술 구조는 영상정보들을 볼 수 있는 멀티미디어의 유연성과, 확장성을 부여하였다. application은 Access Point를 이용한 Wireless LAN(802.11)과 CDMA(IS-95C) 이동통신망을 동시에 지원한다[22].

따라서 장소의 제약 없이 실시간으로 의료정보에 쉽게 접근이 가능하다. 또한 모바일 플랫폼은 VM(Virtual Machine)을 적용하여 단말기 기종과 운영체제의 구애를 받지 않는 하드웨어 독립적인 형태로 간단히 소프트웨어만 수정하면 탑재가 가능한 것으로 C 언어 적용한 GVM 플랫폼을 적용하였다(그림 3).

무선 단말기 플랫폼 비교와 서비스 현황				
플랫폼	개발사	개발언어	제공방식	채택업체
BREM	퀵컴	C	바이너리	KTF
KVM	썬마이크로 시스템	자 바	바이코드	LGT
GVM	신지소프트	모바일 C	스크립트방식	SKT
MAP	모빌탑	C	바이너리	KTF
XVM (SKVM)	XCE	자 바	스크립트	SKT

(그림 3) GVM 플랫폼

EMR 시스템은 아래와 같은 시스템의 특성을 갖는다.

- 융통성(Flexibility)
 - 의무기록을 전자문서형식에 적합하게 설계
- 효율적 접근(Virtual Access)
 - 무선 랜(Wireless LAN), CDMA모듈을 이용한 동시 정보 처리
 - 허가받은 사람만이 접근 가능 및 거의 모든 지역 내에서의 정보 접근 가능
- 환자관리(Patient Management)
 - 현장진료수행에 따른 임상기록의 관리
- 정확한 결과물 (Clear Outcomes) 제공 및
 - 일관된 정확성 (Consistency Accuracy)
 - “현장중심의 적시성”에 기반을 둔 진료 정보의 효율적 관리
 - 모든 진료기록에 대한 최신 정보 실시간 제공
- 신속한 의사전달 및 신속한 접근(Communication/Access)
- 현장에서 신속 정확한 정보 제공뿐만 아니라 협의진료의

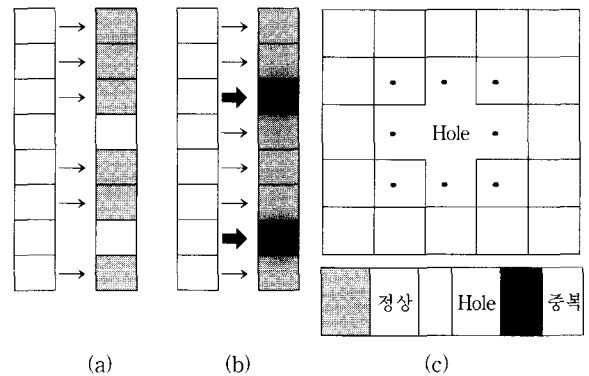
활성화

- 시간과 장소에 구애받지 않고 진료행위 가능
- 여러 사람의 동시 시스템 접속 가능
- 쉬운 정보 검색(Easy Retrieval) 및 시간 절약(Time Saving)
 - 모든 사용자에게 쉬운 정보 검색 및 정보 제공
 - 전자화 된 문서를 통해 차트자료 보관 및 조회하는 시간 절약

3. 2 매핑

3.2.1 중간 영상 공간(Intermediate Image Space)과 최종 영상 공간(Final Image Space)

기존의 영상-기반 렌더링 알고리즘들은 참조 영상들을 최종 영상 공간에서 생성하고, 최종 영상 공간으로 투영하였다. 최종 영상 공간 사이의 투영에는 두 가지 방법이 사용된다. 중첩은 영상이 줄어 들 때 발생 할 수 있다. 구멍은 영상이 팽창할 때 발생할 수 있다. 이러한 현상들은 화소의 방향이 서로 다른 경우에도 발생 할 수 있다. 점대 점 매핑(point-to-point mapping) 근원영상공간과 표적 영상 공간의 화소의 크기나 방향이 다를 경우에 만들어지는 영상에서 구멍과 중첩이 발생한다.

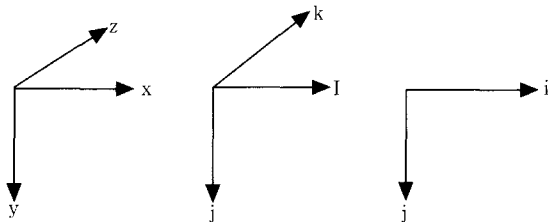


(그림 4) 중복과 Hole

(그림 4)(a)는 근원 영상 공간의 화소 크기가 표적 영상 공간 보다 큰 경우에 구멍이 발생한다. 반대의 경우에는(그림 4)(b)처럼 중첩이 발생한다. 두 공간의 화소가 서로 다른 경우에도 화소의 방향이 다른 경우에는 (그림 4)(c)에서 보는 바와 같이 구멍이 발생할 수 있다. 점대 점 매핑에서 발생하는 문제점들을 해결하기 위해서는 포-코너 매핑을 사용한다. 이 매핑은 근원 영상 공간의 한 화소를 표적 영상 공간의 임의의 사각형으로 변환을 한다. 그러나 이 매핑은 많은 계산 시간을 필요하다. 이 논문에서는 밀립객체 공간(sheared object space)에서 얻어지는 중간 영상 공간(intermediate image space)을 사용함으로써 포-코너 매핑과 같은 복잡한 계산을 하지도 않고, 좋은 질의 영상을 효과적으로 생성할 수 있다. 밀립객체 공간에서 모든 광선들

은 세 번째 좌표축(일반적인 x, y, z 좌표축에서는 z축)과 평행 한다. 객체 공간(object space)의 모든 단면(slice)들을 광선들은 평행이동 시킴으로써 복셀(voxel)들을 밀림 공간내의 2차원 중간 영상으로 투영한다. 밀림 객체 공간은 다음과 같은 성질을 갖는다.

- ① 중간 영상의 화소가 존재하는 스캔 라인과 불륨 데이터의 복셀들이 존재하는 스캔라인은 서로 평행하다.
- ② 주어진 복셀 단면에서 모든 복셀들은 같은 크기를 가진다.
- ③ 평행 투영(parallel projection)에서 모든 복셀 단면들은 같은 크기를 가지며 이를 임의로 선택할 수 있다. 이를 단위 크기로 선택하면 복셀 들과 중간 영상의 화소들은 일대일 대응이 된다.



(그림 5) 불륨공간, 밀림객체공간 및 중간영상공간

(그림 5)에서 불륨 공간(객체 공간), 밀림 객체 공간 및 중간 영상 공간의 관계를 볼 수 있다.

중간 영상 공간을 밀림 객체 공간에서 중간 영상이 위치하는 2차원 공간이라고 정의하면 (그림 5)와 같은 불륨 공간(객체 공간), 밀림 객체 공간 및 중간 영상 공간의 관계를 나타낼 수 있다. 밀림 객체 공간의 (성질 1)에 따라서 중간 영상 공간의 스캔 라인은 불륨 공간의 스캔 라인과 평행 한다. 또한 (성질 2)와 (성질 3)에 의하여 중간 영상의 모든 화소들은 같은 크기 및 방향을 가진다. 따라서 중간 영상 공간을 이용하여 포-코너 매핑을 사용하지 않고 점 대 점 매핑에서 만들어지는 구멍들을 없앨 수 있다. 최종 영상은 중간 영상 공간으로 워핑 하여 만든다.

3.2.2 중간 영상 공간 사이의 매핑

점 대 점 매핑에서 기인하는 중첩이나 구멍 같은 현상들을 없애기 위해서는 최종 영상 공간보다 중간 영상 공간이 효율적임을 설명하였다. 이 절에서는 중간 영상 공간 사이에 매핑이 어떻게 일어나는지 설명한다. 중간 영상 공간을 만들기 위해서 시각행렬을 다음과 같이 분해 할 수 있다.

$$M_{view} = M_{warp} \cdot M_{shear}$$

M_{shear} 는 불륨 공간의 한 복셀을 중간 영상공간의 한 화소로 변화시킨다. M_{shear} 는 s_i, s_j, t_i, t_j 와 같은 계수들로 표현이 되며, 각각은 다음과 같은 의미를 지닌다.

- s_i : i 축으로의 밀림 계수(shear coefficient to the i-axis)
- s_j : j 축으로의 밀림 계수(shear coefficient to the j-axis)
- t_i : I 축으로의 평행 이동계수(translation coefficient to the i-axis)
- t_j : j 축으로의 평행이동 계수(translation coefficient to the j-axis)

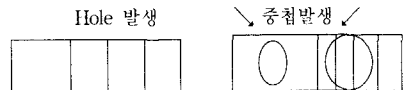
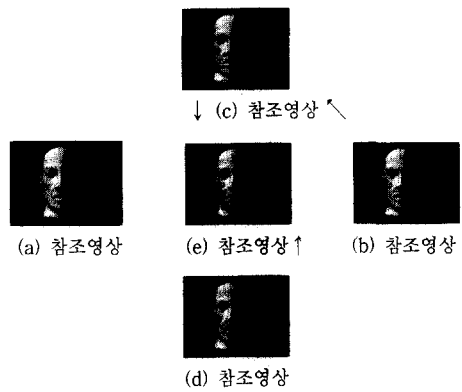
시각행렬로부터 분해된 M_{shear} 는 복셀(x, y, z)을 중간 영상 공간의($x + z \cdot s_i + t_i, y + z \cdot s_j + t_j$)로 투영된다. 시각행렬 Π_s 에서의 새로운 영상 I_s 를 만드는 경우를 생각해 보면, 시각 행렬 Π_1 에서 만들어진 참조영상 I_1 을 가진다. I_1 의 화소(x, y)의 깊이 값(depth value)이 z일 때 이 화소는 새로운 영상 I_s 의 점 p_s 로 다음과 같은 식을 통하여 투영된다.

$$p_s = \Pi_s \Pi_1^{-1}(x, y, z, 1)^T$$

$$I_1(x, y) \rightarrow I_s(x - z \cdot s_i - t_i + z \cdot s_{is} + t_{is},$$

$$y - z \cdot s_j - t_j + z \cdot s_{js} + t_{js})$$

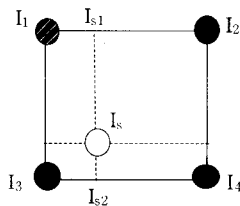
위의 식에 따라서 한 화소는 i 축으로 $z \cdot s_{is} + t_{is} - z \cdot s_{i1} - t_{i1}$ 만큼 움직이며 j 축으로 $z \cdot s_{js} + t_{js} - z \cdot s_{j1} - t_{j1}$ 만큼 움직인다. 따라서 식에 나타나는 밀림 계수들과 평행이동 계수들은 일정한 상수임으로 화소의 이동 거리는 오직 깊이 값에 만 의존한다. 즉 같은 깊이 값을 가지는 모든 화소들은 같은 거리만큼 움직이게 된다. 하나의 참조 영상만을 가지고 임의의 뷰 에서의 영상을 완벽히 복원할 수 없다. 두 개의 참조 영상이 있고, 이들 사이에 I 축으로의 밀림계수만 다른 경우를 가정해 보자. 앞 절에서 설명했듯이 중간영상 공간을 사용하기 때문에 영상에서 점 대 점 매핑에서 기인하는 구멍은 발생하지 않는다. 그러나 참조영상에서는 보이지 않던 부분이 새로운 영상에서는 보이게 됨에 따라 구멍이 발생한다.



(그림 6) 두 개의 참조영상들이 원하는 영상들

(그림 6)에서 두개의 참조 영상이 표적영상공간에 투영되는 과정을 볼 수 있다. 이때 두개의 참조 영상이 새로운 뷰에서 의 영상에 존재하는 모든 정보를 가지고 있다고 가정하여 다음과 같이 나타낼 수 있다. (그림 6)는 (그림 6)(a)의 I축으로 밀림계수는 (그림 6)(d)의 밀림계수보다 크다. (그림 6)(f)에서 볼 수 있듯이 밀림계수가 줄어들 때에는 중첩이 영상의 오른쪽에서 발생하며, 구멍은 영상의 왼쪽에서 발생한다. 같은 방법으로 (그림 6)(b)가 (그림 6)(c)로 투영될 때도 구멍과 중첩이 발생한다.

두 개의 영상을 합하게 되면, (그림 6)(e)와 같은 최종 영상을 구멍 없이 얻을 수 있게 된다. (그림 6) 주위의 네 개의 참조 영상으로부터 원하는 뷰에서 의영상을 구멍 없이 얻을 수 있다. 위의 생각을 2차원으로 확장시킬 수 있다. (그림 7)에서 I1과 I2(I3과 I4도 마찬가지)는 I축으로의 밀림 계수만이 다르다.



(그림 7) 네 개의 참조 영상으로부터 원하는 뷰에서 영상을 Hole 없이 만들

I1과 I3(I2와 I4)도 마찬가지)는 j축으로의 밀림 계수만이 다르다. 이 때 네 개의 전처리 영상 사이에 존재하는 임의의 뷰로부터 새로운 영상을 얻는 경우를 생각해 보자. 앞에서 언급한 바와 같이 구멍 없이 얻을 수 있다. 얻고자 하는 최종 영상으로부터 구멍 없이 만들어진다.

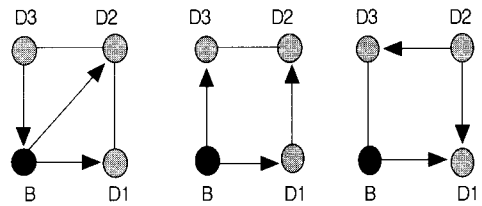
3.3 볼륨 표현

3.3.1 트리 구성

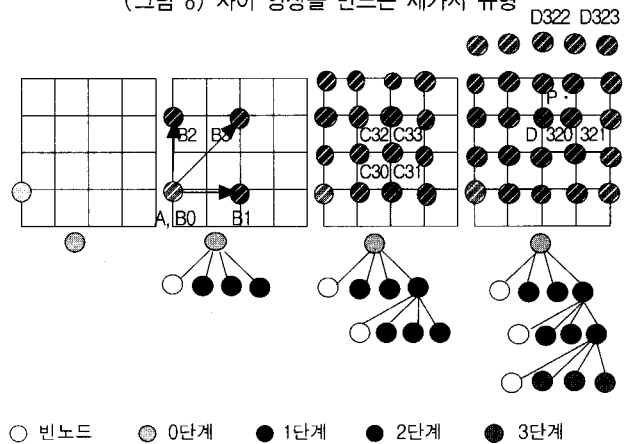
영상 기반 렌더링을 위한 데이터베이스를 구축하기 위해서 참조 영상들이 미리 정의되어진 뷰에서 만들어진다. 평행 투영인 경우에 하나의 뷰는 X-R(Reverse), X-N(Non-reverse), Y-R, Y-N, Z-R, Z-N의 6가지 중의 하나의 뷰-축(view-axis)을 갖는다. X-R(Reverse)의 의미는 뷰의 주축이 X축이고 볼륨의 단면이 뒤에서부터 앞으로 투영된다는 것을 의미한다. 하나의 뷰 축에 대해서 i축과 j축에 대하여 모두정수의 밀림 계수를 가지는 영상은 모두 9개이다. 볼륨의 표면 볼록하다면 임의의 뷰에서 만들어지는 영상들을 정수 밀림 계수를 가지는 9개의 영상으로부터 생성할수록 더 많은 구멍들을 제거할 수 있지만 저장 공간의 낭비가 따른다. 그러나 인접한 뷰에서의 영상은 비슷함으로 그들간의 관련성을 가지고, 압축 효과를 얻을 수 있다. 볼

륨데이터 베이스를 표현하기 위해서 사진 트리를 이용한다. 일반성 가지며 사각형(0, 0)¹⁾ - (1, 1)에서 깊이(depth)가 3인 트리를 만드는 과정을 예를 들어 설명한다. 트리를 구성하는 각 노드들은 자신이 가지는 화소들을 자신의 자식노드에 전달한다. 즉 한 노드는 그의 조상노드들이 가지는 모든 화소들을 가지게 되며 그 자신은 조상 노드들에 의해서 저장되지 않은 화소들만을 저장하면 된다. 이렇게 표현되는 영상을 차이 영상(difference image)라고 가정하여 단계(level)가 높은 노드일수록 적은 수의 화소를 가지게 된다 (그림 8).

- (a) $D1 = D1 - B, D2 = D2 - B, D3 = D3 - B$
- (b) $D1 = D1 - B, D2 = D2 - D1 - B, D3 = D3 - B$
- (c) $D1 = D1 - B, D2 = D2 - D1 - B, D3 = D3 - D2 - D1 - B$



(그림 8) 차이 영상을 만드는 세가지 유형



(그림 9) 트리 구성 과정

초기 기본 영상이 (0, 0)에서 시작된다. 다음단계에서 세개의 차이 영상들이 (0, 5, 0), (0, 5, 0, 5), (0, 0.5)에서 만들어진다. 차이 영상들은 (그림 7)와 같이 세 가지의 방법으로 만들어진다. 트리가 마지막 단계까지 확장되지 않았다면 (그림 8)(a)나 (그림 8)(b)가 사용된다. (그림 8)(a) 기본 모드에서 D2로 직접 가는 길이 있다는 장점이 있다. 어떤 길이 주어졌을 때 그 길이 D2를 통과 해야한다면 직접 가는 길이 있다는 것은 투영되는 화소의 개수가 줄어들음을 의미한다. 하지만 이 경우에는 D1과 D2사이 중복이 존재한다. 이러한 중복을 없애고 싶을 경우에는 (그림 8)(b)를 사용할 수 있다. 트리의 단계가 증가할 때마다

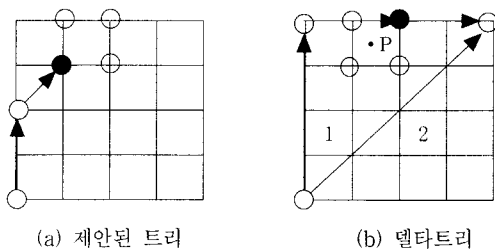
각각의 노드들은 자신의 자식 노드들을 생성한다. 예를 들어 단계 n의 어떤 노드(x, y)는 자신의 노드(x+(0.5)ⁿ⁻¹, y), (x, y+(0.5)ⁿ⁻¹), (x+(0.5)ⁿ⁻¹, y+(0.5)ⁿ⁻¹)들을 생성한다. 이러한 방법으로 트리의 자손들을 계속 생성하더라도 (그림 9) (c)에서 보는 것처럼 모든 노드들을 다 생성하지는 않는다. 따라서 완벽하게 트리를 생성하기 위해서는 (그림 6)(d)와 같이 맨 마지막 단계에서는 임의의 뷰를 포함하는 네 개의 참조 영상들을 모두 투영해야 하므로 (그림 6)(c)와 같은 형태를 사용하여 표현하는 것이 바람직하다. (그림 9)에서 3단계의 사진 트리를 형성하는 과정을 보여준다. 3단계 사진 트리의 경우에 1개의 0단계 영상, 3개의 1단계 영상, 12개의 2단계 영상과 36개의 3단계 영상들을 가진다.

3.3.2 뷰 복원(View Reconstruction)

뷰 복원 과정을 (그림 9)를 가지고 설명한다. 임의의 뷰 p가 결정되면 시각 행렬이 만들어진다. 그 시각 행렬을 분해하여 밀립 계수 s_i, s_j와 평행이동 계수 t_i, t_j가 결정된다. 밀립계수 s_i, s_j를 가지고 뷰를 둘러싸고 있는 사각형(D320, D321, D322, D323)을 얻을 수 있다. 뷰 P에서 영상은 트리의 뿌리(root) 노드로부터 사각형까지 횡단하며, 각 노드들의 화소들을 투영함으로써 생성할 수 있다. (그림 9)의 예에서 영상을 생성하기 위한 길은 {A, B3,C32, D321, D322, D323}이다.

3.3.3 델타-트리와의 비교

델타 트리도 부모-자식간의 중복된 화소들을 제거함으로써 압축효과를 얻는다. 그렇지만 델타트리에서 몇몇 노드들은 사촌 노드에서 중복되게 표현된다.



(그림 10) 제안된 트리와 델타-트리와의 복원 경로 비교

뷰 p에서의 영상을 생성하기 위해서 두개의 트리 모두 동일한 길은 가진다. 그렇지만 (그림 8)의 검은 노드는 델타-트리에서는 두개의 길에 의해서 표현이 된다. 비록 (그림 10)(b)에 보이는 1, 2 두개의 길이 모두 같은 영상을 생성한다 해도 길 위에 있는 노드들은 각각 자신의 정보를 저장하고 있어야 한다. 하지만 본 논문에서 제시하는 트리는 맨 마지막 단계의 노드들만 제외하면 사촌들 간에 중복해서 노드들을 저장하지 않음으로써 저장 공간의 절약 효과를 얻을 수 있다.

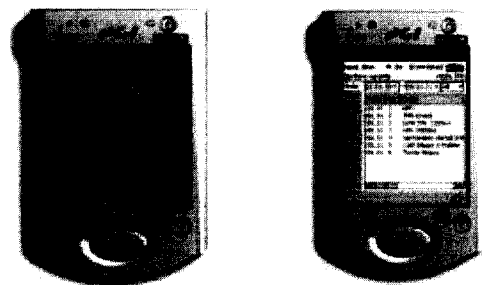
4. 실험 결과 및 고찰

quad 트리를 이용한 영상처리기법을 적용하여 의료정보 산업기술을 구현하였다. Data(자료)의 크기가 적을수록 그 결과 식이 개선되었다(그림 11).

데이터	크기	단계	뷰의 개수	중복 계수	중복계수 (델타트리)	압축률	압축률 (델타트리)
unc head	256*256*256	1	4	1.04	1.11	3.85 : 1	3.6 : 1
unc head	256*256*256	2	9	1.68	2	5.36 : 1	4.5 : 1
unc head	256*256*256	3	25	2.43	3.09	10.3 : 1	8.1 : 1
unc head	256*256*256	4	81	3.6	6.53	22.5 : 1	12.4 : 1
unc head	256*256*256	5	289	5.57	10.03	51.9 : 1	28.8 : 1
뇌	128*128*84	4	81	3.72	6.81	21.8 : 1	11.9 : 1
엔진	256*256*110	4	81	3.24	5.91	25	13.7 : 1

(그림 11) 트리의 공간 효율성

제시한 트리의 공간 효율성을 보여주고 있다. UNC head 256*256*256 실험을 했을 때 4단계 트리가 가장 좋은 것을 알 수 있다. 실험 결과 임의의 뷰에서의 영상을 생성하기 위하여 제시된 알고리즘은 대략 5~20ms의 투영 시간을 가진다. (0, 0)과 가까운 영상에서는 빠른 투영시간을 가지며, 멀어질수록 긴 투영시간을 가짐을 알 수 있다. 임의의 뷰를 복원하기 위한 투영시간은 투영화소의 개수에 비례한다. 압 시간을 포함하여 하나의 영상을 생성하기 위하여 약 25~40ms의 시간이 필요하다. 같은 질의 영상을 쉬어-왓 알고리즘을 만들었을 때 280~300ms 정도 걸리는 것과 비교하



(그림 12) unc head에 대한 영상 렌더링

면 제안된 알고리즘이 상당히 좋은 성능을 가지는 것을 알 수 있다. 또한 델타 트리로 임의의 뷰에서의 영상을 만드는 경우 1초에 10내·외의 영상을 만드는 것과 비교해 볼 때 제안된 트리가 좋은 성능을 보임을 알 수 있다. (그림 12)은 실험 결과를 나타낸다.

5. 결 론

본 논문에서 제안한 quad 트리 방식은 최종 영상공간대신에 중간 영상공간을 사용함으로써 효율이 높아졌으며, 점대 점 매핑 으로부터 발생하는 Hole의 문제점을 해결하였다. 델타 트리 기반의 quad 트리가 불투표현에 효과적이었으며, 압축 효과가 높았다. 본 논문에서 제안한 모바일 하는 기반으로 하는 사진 트리 영상 처리를 통하여 효율적인 PDA용 전자 의무 기록 차트를 제안하였고, 의료정보 시스템에 적용하여 의료 정보 서비스를 지원하고 보다 신속하며 효율적인 모바일 전자 의무 기록 차트로 사용을 제안하며, 향후 모바일 영상처리가 확대되어 관련한 진료 업무에 이용되어 보다 편리한 의료 영상 처리로서 유용하게 사용될 수 있는 연구가 필요 된다.

참 고 문 헌

[1] Amin, M. B., Grama, A., Singhm, V., Fast Volume Rendering Using an Efficient Scalable Parallel Formulation of the Shear-Warp Algorithm 1995 Parallel Rendering Symposium, pp.7-14, October, 1995.

[2] Chen, S. E. and L. Williams. "View Interpolation for Image Synthesis," SIGGRAPH93, pp.279-288.

[3] Chen, S. E., Quicktime VR An Image-Based Approach to Virtual Environment Navigation Processings of SIGGRAPH 95, Computer Graphics pp.29-38, August, 1995.

[4] Chen, S. E., Williams, L., View Interpolation for Image Synthesis Proceedings of SIGGRAPH93, Computer Graphics, pp.279-288, August, 1993.

[5] Curless, B. and M. Levoy, "A Volumetric Method for Building Complex Models from Range Images," SIGGRAPH96, pp.303-312, 1996.

[6] Dally, J. W., McMillan, L., Bishop, G., Fuchs, H., The Delta-Tree : An Object-Centered Approach to Image-Based Rendering, MIT AI Lab Technical Memo 1604, May, 1996.

[7] Debevec, P., C. Taylor and J. Malik, "Modeling and Rendering Architecture from Photographs," SIGGRAPH96, pp. 11-20, 1996.

[8] Debevec, P. Modeling and Rendering Architecture from Photographs. Ph. D. Thesis, University of California at Ber-

keley, Computer Sciences Division, Berkeley, CA, 1996.

[9] Debevec, P. E., Y. Yu and G. D. Borshukov, "Efficient View Dependent Image-based Rendering with Projective Texture Mapping," Proc. of EGRW 1998, June, 1998.

[10] Faugera, O. Three-dimensional Computer Vision : A Geometric Viewpoint, MIT Press, 1993.

[11] Gelder, A. V., Kim, K. S., Direct Volume Rendering with Shading via Three-Dimensional Textures, 1996 Volume Rendering Symposium, pp.23-30, 1996.

[12] Gortler, J. S., Grzeszczuk, R., Szeliski, R., Cohen., F. M., The Lumigraph, Proceedings of SIGGRAPH96, Computer Graphics, pp.43-54, August, 1996.

[13] Hsu, W.M., Segmented Ray Casting for Data Parallel Volume Rendering 1993 Parallel Rendering Symposium, pp.7-14, October, 1993.

[17] Chen, S. E., Williams, L., View Interpolation for Image Synthesis Proceedings of SIGGRAPH93, Computer Graphs, pp.279-288, August, 1993.

[14] Imielinski, T. and Badrinath, B. "Mobile Wireless Computing : Challenges in Data Management," Technical Report, Dept. of Computer Science, University of Rutgers, Oct., 1994.

[15] Lacrou, P., Levoy, Mm., Fast Volume Rendering Using a Shear-Warp Factorization of the Viewing Transformation Proceedings of SIGGRAPH94, Computer Graphics, pp.451-458, July, 1994.

[16] Levoy, M., "Display of Surface from Volume Data," IEEE CG&A, Vol.8, No.5, pp.29-37, 1988.

[17] McMillan, L., An Image-Based Approach to Three-Dimensional Computer Graphics, Ph. D. Dissertation, University of North Carolina, April, 1997.

[18] McMillan, L., Bishop, G., Plenoptic Modeling : An Image-Based Synthesis Proceedings of SIGGRAPH93, Computer Graphics, pp.39-46, August, 1995.

[19] Regan, M., Pose, R., Szeliski, R., Cohen., F. M., The Lumigraphs Proceedings of SIGGRAPH 96, Computer Graphics, pp.43-54, August, 1996.

[20] Yagel, R., Kaufman, A., Template-based volume viewing in Eurographics 92, C-153-167, Cambridge, UK, September, 1992.

[21] 김기천 "모바일 서비스 기술 동향" 정보처리학회, 2002.

[22] 류지원, 권경환, 이궁해, "모바일 환경에서 기존 인터넷 콘텐츠 활용의 자동화를 위한 유무선 통합 인터넷 서비스 지원 시스템설계", 정보과학회 2003년 춘계 학술대회, Vol.30, No. 01, April, 2003.

[23] 최 성, 박석천 "모바일 콘텐츠 서비스 현황과 발전 방향", 정보처리학회, 2002.



주 헌 식

e-mail : hsjoo21014@syu.ac.kr

1992년 호서대학교 컴퓨터공학과(학사)

1994년 호서대학교 대학원 전자계산학과
(이학석사)

1998년 아주대학교 대학원 컴퓨터공학과
(박사수료)

1997년~현재 삼육의명대학 컴퓨터정보과 조교수
관심분야 : 컴퓨터그래픽스, 영상처리, 인터넷 응용



김 하 진

e-mail : hjkim@madang.ajou.ac.kr

1962년 서울대학교 물리과 대학 수학과
(이학사)

1978년 Grenoble 1 대학교 대학원 응용
수학과 D.EA(이학석사)

1980년 Saint-Entienne 대학교 대학원 응용
수학과(이학박사)

1984년~1985년 프랑스 INRIA 초빙교수

1989년~1992년 한국정보과학회부회장 및 회장

1993년~1995년 아주대학교 공과대학장

1974년~현재 아주대학교 정보통신대학 교수

관심분야 : 컴퓨터그래픽스, 수치해석