

영상 기반 렌더링을 지원하는 가속기 구조에 관한 연구

정우남^o 이승기 박우찬 한탁돈
연세대학교 컴퓨터과학과

(wnjung^o, sklee, chan, hantack)^o@kurene.yonsei.ac.kr

Study of Accelerator Architecture to Support Image-Based Rendering

Woo-Nam Jung^o, Seung-Gi Lee, Woo-Chan Park, Tack-Don Han
Dept. of Computer Science, Yonsei University

요 약

현재 실시간의 실감 영상을 위한 다양한 기법들에 대한 활발한 연구가 진행되고 있다. 영상 기반 렌더링은 새롭게 주목 받고 있는 렌더링 방법으로 기존의 기하학 기반 렌더링과는 다르게 모델을 작성하는데 쉽게 사용될 뿐 아니라, 실감 영상을 만들어 내는 것에도 탁월한 성능을 나타내고 있다. 현재 사용하는 그래픽 가속기는 기하학 기반 렌더링의 방법을 위주로 설계되고 있는 추세이다. 이에 영상 기반 렌더링을 지원하는 구조의 제안을 통해서 실시간 영상의 생성을 가능하게 하였다. 또한 기존의 그래픽 가속기와의 통합을 통해 하드웨어 비용을 절감하며 효율적으로 두 가지 기법을 지원하는 구조를 제안하였다.

1. 서론

3차원 그래픽스는 컴퓨터를 이용하여 영상을 생성하는 분야이다. 지금까지 컴퓨터 그래픽은 다각형을 기반으로 한 방법을 이용하였고, 아직까지도 이 방법을 이용하여 영상을 생성하고 있다. 기하학 기반 렌더링의 방법으로 생성된 영상은 질적인 면에서 많은 발전을 이루어, 다양한 기법을 통해서 보다 현실감 있는 영상을 생성하여왔다. 더욱이 컴퓨터 하드웨어의 발전으로 개인용 컴퓨터에서도 현실감 있는 영상을 직접 접할 수 있게 되었다. 하지만 컴퓨터로 생성된 그래픽은 역시 인공적으로 만든 영상이기 때문에 실제 세계의 영상을 모두 표현하기 어려울 뿐 아니라, 표현한다 하더라도 컴퓨터로 실시간 처리하기에는 어려움이 많다. 또한 실제 세계를 정확하게 모델로 작성하는 것도 어려운 문제이다. 이런 것을 해결하고자 제안된 방법이 바로 영상 기반 렌더링이다. 이 방법을 이용한 방법은 다각형을 이용한 방법에 비해서 더 현실과 가까운 영상을 만들 수 있고 비슷한 영상을 만드는 경우 상대적으로 적은 연산으로 영상이 생성될 수 있다는 장점을 가지고 있다. 또한 모델로 만드는 방법에서도 실제 이미지를 사용하기 때문에 보다 복잡한 영상의 생성에 있어서 효과적으로 적용이 가능하다.[1]

영상 기반 렌더링은 화소당 처리를 요구하기 때문에 이를 처리하기 위해서는 많은 연산을 필요로 한다. 이에 제안한 구조는 4개의 와핑 결과를 동시에 얻을 수 있어 효율적인 와핑 처리가 가능하다. 또한 간단한 덧셈기와 쉬프트만을 첨가하여 낮은 비용으로 처리가 가능하다. 또한 재구성 구조에서는 OpenGL의 구조를 지원하는 매핑 구조의 텍스처 주소 생성과 매핑과의 유사점을 이용하여 통합된 구조를 제안한다. 이렇게 제안한 구조는 기하학 기반 렌더링과 동시에 영상 기반 렌더링을 지원할 수 있도록 한다.

본 논문에서는 2장에서는 영상 기반 렌더링의 기본 알고리즘과 하드웨어로 구성된 관련 연구를 알아 본다. 3장에서는 영상 기반 렌더링을 위해 제안한 영상/기하학 렌더링 혼합 하드웨어 구조에 대하여 논하고, 끝으로 4장에서 결론을 맺는다.

2. 영상 기반 렌더링

영상 기반 렌더링에서 입력 이미지를 이용하여 이미지를 처리

하는 방법을 두 가지 단계로 나눌 수 있다. 2차원의 이미지를 3차원의 형태로 변형을 시키는 와핑의 단계와 부드러운 표면을 구성하기 위한 재구성 단계가 바로 그것이다. 와핑은 차원의 변화로 인해 원본 영상이 입력 영상과는 다른, 3차원의 원하는 영상으로 변형이 된다. 이렇게 변형된 영상은 부드러운 표면으로 다시 만들어져야 한다. 와핑만으로는 구멍과 같은 부분이 생성되고, 이를 처리하는 단계가 필요하다.

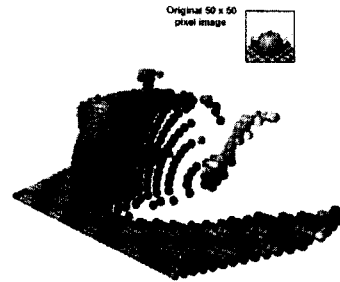


그림 0 와핑과 재구성

2.1 와핑

그림 1 에서 보이는 봐와 같이 원본 이미지는 와핑을 통해서 2차원의 영상이 3차원으로 변형된다. McMillan[1]은 3차원 와핑 방정식을 제안하여 시점의 변화를 통해서 원본 영상에서 원하는 영상으로 변형을 할 수 있도록 하였다. 수식 1은 제안된 와핑 방정식이다.

$$u_2 = \frac{w_{11}u_1 + w_{12}v_1 + w_{13} + w_{14}\delta(u_1, v_1)}{w_{31}u_1 + w_{32}v_1 + w_{33} + w_{34}\delta(u_1, v_1)}$$
$$v_2 = \frac{w_{12}u_1 + w_{22}v_1 + w_{23} + w_{24}\delta(u_1, v_1)}{w_{31}u_1 + w_{32}v_1 + w_{33} + w_{34}\delta(u_1, v_1)}$$

수식 0 와핑 방정식

u_2, v_2 는 원하는 영상의 좌표이고, u_1, v_1 는 원본 영상의 좌

표이다. w_{nm} 는 원본 영상과 원하는 영상 사이의 카메라 시점 매개 변수의 변화 상수이다. $\delta(u_i, v_i)$ 는 원본 영상 평면과 시점 사이의 거리의 비율로 정의 된다.

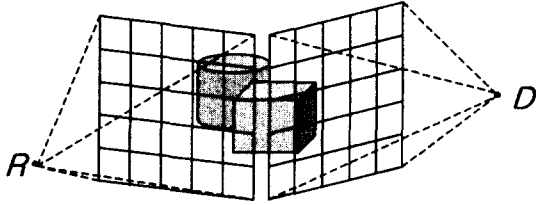


그림 0 와핑의 시점 변환

수식 1은 그림 2에서 보는 바와 같이 원본 영상과 원하는 영상의 각각의 시점과의 관계를 통해서 얻어진다. 따라서 R의 시점에서 D의 시점으로 변화하였을 때의 좌표를 통해서 새로운 3차원의 영상을 얻어 내는 것이다.

2.2 재구성

앞 절에서 설명된 방법을 통해서 얻어지는 좌표는 텍스처 매핑과의 유사한 방법으로 색깔 값을 매핑시켜 원하는 영상의 색깔을 구성한다. 하지만 이 방법만으로는 3차원의 영상을 얻을 수는 있지만 완전 완전한 영상을 얻을 수 없다. 입력되는 원본 영상 자체가 2차원의 제한된 영상이므로 와핑으로 구멍이 발생 할 수 밖에 없다. 이런 부분을 처리할 수 있는 것이 필요하다. 이런 문제는 재구성 단계를 통해서 해결을 한다. 이 단계에서는 원본 영상의 시점을 벗어나는 부분의 가려진 영상도 처리할 수 있도록 한다.

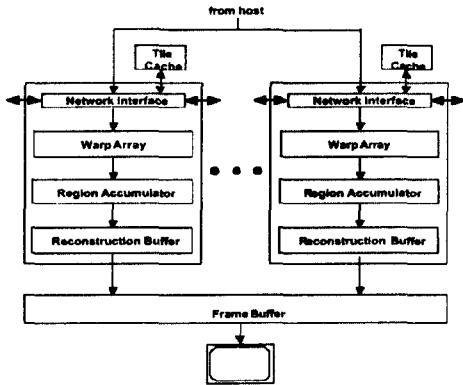


그림 3 WarpEngine 구조

2.3 관련연구 : WarpEngine

WarpEngine은 영상 기반 렌더링을 처리하기 위해 제안된 구조로 메모리 프로세서 집합 구조로 설계된 시스템이다. 특히 이 구조는 일반적인 텍스처 매핑의 방법과는 다르게 전 방향 렌더링 기법을 사용하고 있다. 또한 16x16의 샘플 타일을 렌더링의 기본 단위로 사용하고 있다. 다음 그림 3은 WarpEngine의 하드웨어 구조를 나타내고 있다. host에서 어떤 원본이미지의 타일을 사용할 것인지를 결정한다. 타일은 Warp Array의 입력이 된다. Warp Array에서는 와핑에 필요한 연산을 수행한다. 와핑 된 샘플들은 Region Accumulator로 전달되어 Warp Buffer에 모아지면, 이것을 Reconstruction Buffer에 보내어 최종 화소의 값을 계산한다. 결정된 값은 프레임 버퍼로 전달하여 출력을 한다

3. 영상/기하학 렌더링 혼합 하드웨어 구조

3.1 파이프 라인구조 비교

3차원 그래픽 가속기는 기하학처리 단계와 렌더링 단계로 분리되어 처리를 하고 있다. 그림 4의 (a)[4]에서 각각의 처리 단계를 보여주고 있다. 입력되는 점의 정보를 변화과정을 거쳐 원하는 위치를 결정하고 점을 다각형의 형태로 연결한다. 연결된 다각형의 내부를 렌더링의 과정을 통해서 채워서 영상을 생성한다. 영상 기반 렌더링의 경우 기하학 기반 렌더링과 같이 두 가지 단계로 구분이 가능하다. 2장에서 설명한 바와 같이 와핑과 재구성의 단계이다. McMillan이 제시한 영상 기반 렌더링의 파이프 라인은 그림 4의 (b)[1]와 같다. 와핑이 다른 부분에 비해 높은 비중의 작업을 수행한다. Clipping의 경우는 모델 데이터의 특성상 기하학 기반 렌더링에 비해 많은 작업을 수행하지 않는다. 이후의 작업은 다시 재구성을 위한 작업으로 이루어 진다. 원하는 영상의 색깔 값은 와핑 된 좌표에 맞게 처리된다.

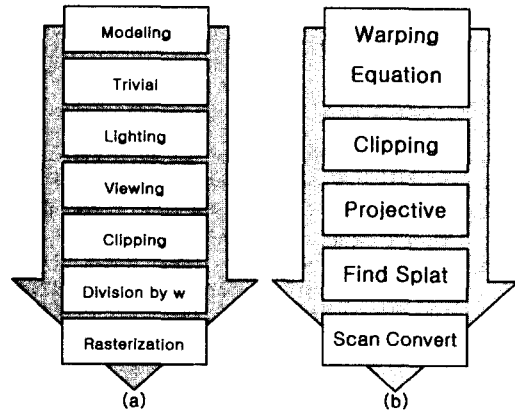


그림 4 그래픽 파이프 라인

3.2 와핑 구조

영상 기반 렌더링의 방법에서 와핑은 기하학 기반 렌더링에서 기하학처리 단계에서의 연산과 유사하다. 와핑의 방정식을 살펴보면 수식 2와 같이 행렬 연산이 가능하다. 기하학처리 단계에서 사용하는 각 점에 대한 변환 및 빛에 대한 행렬 연산과 유사함을 알 수 있다. 이러한 특성을 이용하여 기존에 그래픽 가속기의 기하학처리 단계의 하드웨어를 이용한다면 영상 기반 렌더링을 위한 구조의 통합이 가능하다.

$$\begin{bmatrix} x \\ y \\ 1 \\ w \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ 0 & 0 & 0 & 1 \\ w_{31} & w_{32} & w_{33} & w_{34} \end{bmatrix} \begin{bmatrix} u_i \\ v_i \\ \delta(u_i, v_i) \\ 1 \end{bmatrix}$$

수식 0 행렬로 변환한 와핑 방정식

또한 데이터의 특성상 근접한 데이터의 경우 덧셈기와 뺄셈기를 이용하여 쉽게 처리 할 수 있다. ax_2 는 ax_1 에 a 를 더한 것과 같다 바로 인접하여 있는 좌표의 데이터이기 때문이다. $x_2 = x_1 + 1$ 와 같은 성질을 이용한 것이다. 특히 와핑의 경우 전방향의 재구성 기법을 사용하기 때문에 더욱 간단하게 사용이 가능하다. 더욱이 화소당 결정되어 있는 깊이 값도 역시

근접한 화소에서는 동일한 값의 적용이 가능하고, 이런 경우는 시점의 변화도 동일하게 사용되기 때문에 와핑 방정식의 w_{mn} 은 동일 하게 사용되어 그림 5의 구조와 같이 사용한다면 동시에 4개의 값을 뽑아 낼 수 있다. 그림 5에서 3개의 PE가 사용된 모습을 보여준다. 이는 수식 2에서 알 수 있듯이 3번째 행의 연산은 수행에 의미가 없기 때문이다. 세개의 PE에서 생성된 x, y 는 w 로 나누어서 원하는 와핑된 좌표 값을 얻는다. 그림 4의 기하학 기반 렌더링의 파이프라인 구조 division by w 단계의 작업과 유사한 수행으로 원하는 좌표로 와핑을 마무리함을 알 수 있다. 이런 특성을 이용하여 통합을 한다면 영상 기반 렌더링의 와핑 단계와 기하학 기반 렌더링의 기하학처리부의 통합이 가능하다.

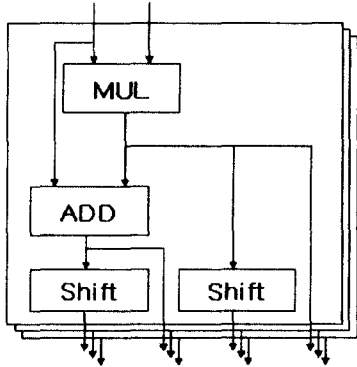


그림 5 와핑 PE의 구조

3.3 재구성 구조

재구성은 2.2절에서 설명한 바와 같이 색깔을 결정하는 단계이다. 특히 이 부분은 기존의 3차원 그래픽 가속기에서 사용되는 텍스처 매핑의 역할과 상당히 유사하다. 하지만 수행되는 작업 자체는 유사해도 텍스처 매핑의 경우는 역방향 매핑을 통해서 텍스처 맵의 색깔을 원하는 스크린 좌표로 가져오지만, 영상 기반 렌더링에서는 전방향 매핑을 통해서 원본 영상에서 원하는 영상을 만들어 내는 방법을 취하고 있다. 두 가지 방법이 서로 다르기는 하지만 유사한 기능의 장치들이 이용된다. 텍스처 어드레스를 계산하는 것과 색 보간을 하는 작업은 유사하게 사용할 수 있다. 재구성 단계에서도 기존의 그래픽 가속기에서 이용되는 구조를 다시 사용할 수 있다. 추가적으로 구멍을 제어하는 기능을 더하여 사용하면 영상 기반 렌더링을 지원하는 구조가 된다.

그림 6에서 회색으로 표현된 부분이 기존의 가속기에 추가된 영상 기반 렌더링의 구조이다. 기존의 래스터라이저에서 텍스처 매핑을 하는 부분을 공유하고 메모리 관리에서도 유사한 특성을 이용한다. 3.2절에서 와핑된 데이터에 맵핑되는 색깔정보를 텍스처 매핑에서 주소를 생성해서 색깔 정보를 메모리에서 가져오는 부분과 유사하게 이용한다. 따라서 메모리도 텍스처 메모리와 역시 유사한 구조로 구성하고, 그대로 사용해도 무방하지만 역시 크기에 있어서는 차이를 보인다. 출력 영상 전체를 2차원 영상으로 만들기 때문에 텍스처와는 그 크기에서 차이가 발생한다. 색깔이 저장되는 구조도 역시 유사하다. 단지 기존의 구조와 다른, 전방향 매핑을 수행함에 따라서 새로운 영상기반 렌더링(IBR) 래스터라이저를 사용한다. 이 구조는 전방향의 특성으로 인해 프레임 버퍼와 밀접한 관계를 유지하며, 구멍이 생기고 겹쳐지는 부분을 처리한다. 겹이어서 겹쳐져서 보이지 않는 부분에 대해서는 색깔에 대한 정보를 메모리로부터 가져올 필요가 없다. 따라서 영상기반

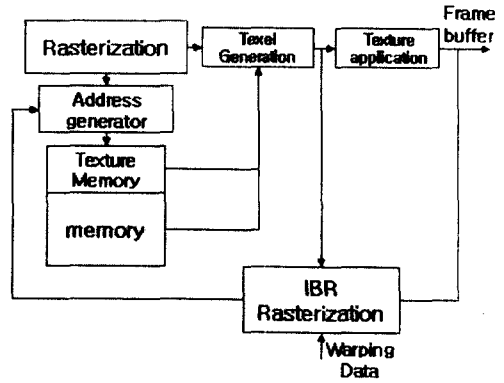


그림 6 래스터라이제이션 혼합구조

렌더링 래스터라이제이션 단계에서 Z-Test를 수행하여 메모리에 필요 없는 접근을 줄인다.

4. 결론

영상 기반 렌더링은 이미 다양한 분야에서 적용하여 사용하고 있다. 특히 높은 수준의 질적 효과를 보여주고 있어서 차세대 컴퓨터 그래픽의 기술로 점차 입지를 굳히고 있는 중이다. 이에 기존에 사용되는 개인용 컴퓨터의 그래픽 가속기에 영상 기반 렌더링을 지원할 수 있는 기능을 삽입하는 구조에 대하여 고찰하였다. 점의 변환과 점의 다각형화에 이은 다각형 내부의 색을 결정하는 기존의 그래픽 가속기의 구조의 작업 특성을 유지하면서 영상 기반 렌더링의 기능을 추가할 뿐 아니라, 역시 유사한 작업을 수행하는 장치에 대해서는 공유를 통해 하드웨어 비용에서도 많은 부담을 줄일 수 구조이다. 와핑의 방정식이 점의 변환과 유사한 특성이 있는 점을 토대로 기하학 처리부의 기능을 사용하는 와핑 구조의 사용 가능성을 살펴보았다. 또한 동시에 4개의 와핑을 수행하는 구조의 제한을 통해서 효율과 비용에 장점을 가지도록 하였다. 또한 래스터라이저의 텍스처 매핑의 구조가 영상 기반 렌더링의 구조와 유사하게 사용될 수 있는 점을 착안하여 통합된 구조를 제안하였다. 이 구조는 다각형과 영상을 기반으로 한 모든 기법을 모두 지원할 수 있는 장점을 가진다.

참고문헌

[1]McMillan, Leonard. *An Image-Based Approach to Three-Dimensional Computer Graphics*. Ph.D. Dissertation, University of North Carolina, April 1997.
 [2]Popescu, Voicu, *Forward Rasterization: A Reconstruction Algorithm for Image-Based Rendering*. Ph.D. Dissertation, University of North Carolina, January 17, 2001.
 [3]Foley, J.D., A. van Dam, S.K.Feiner, and J.F. Hughes. *Computer Graphics - Principles and Practice*. 2nd Edition Addison-Wesley Publishing Company, Reading, MA, 1990
 [4]Mark, William R. *Post-Rendering 3D Image Warping: Visibility, Reconstruction, and Performance for Depth-Image Warping*. Ph.D. Dissertation, University of North Carolina, April 21, 1999