

PC 클러스터 기반의 가상현실 시스템의 설계 및 구현

박창훈⁰ 고희동 김태윤
고려대학교
(minam, ko)@kist.re.kr, tykim@netlab.korea.ac.kr

Design and implementation of virtual reality system based on PC cluster

Chang-Hoon Park⁰, Hee-Dong Ko, Tai-Yun Kim
Dept. of Computer Science, Korea University

요약

본 연구는 저가의 가상현실 시스템을 구현하기 위하여 LAN에 연결된 다수의 PC들로 구성된 가상현실 시스템을 제시한다. 이때, PC들은 고유 기능에 따라 3차원 그래픽을 실시간으로 랜더링 하는 VR 서버, 가상공간의 운영을 위하여 GUI 인터페이스를 지원하는 제어 서버, 상호작용을 위하여 입출력 장치가 연결된 장치 서버로 분류한다. 제안하는 시스템의 핵심 기능은 분산된 서버들을 단일 가상공간에 통합하는 것이다. 우리는 서로 다른 특징을 가지는 서버들을 통합하기 위하여 동기 마스터/슬레이브, 이벤트 관리자, 상호 작용 관리자를 설계하고 이를 구현하였다.

1. 서 론

가상현실은 컴퓨터를 이용하여 참여자에게 실시간 상호작용이 가능한 3차원 그래픽 공간을 제공하여 몰입감을 일으킨다. 이러한 가상현실 기술은 군사 훈련, 오락, 협동 작업, 의료 등 다양한 분야에 응용되고 있으며 차세대 쌍방향 미디어로 주목을 받고 있다 [1].

지금까지 가상현실이 요구하는 고성능의 실시간 처리를 위하여 고가의 워크스테이션을 필요로 하였다. 특히, 수천만 이상의 폴리곤을 가지는 3차원 모델을 랜더링하기 위하여 SGI사의 그래픽 전용 워크스테이션이 주로 이용되었다. 그런데, 최근 PC의 계산 능력과 3차원 그래픽 가속기의 성능이 급속하게 향상되면서 PC를 기반으로 하는 가상현실 시스템의 등장이 가능해졌다 [2].

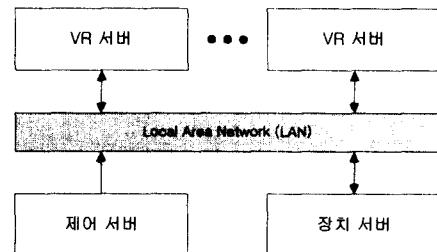
하지만, 단일 PC가 그래픽 전용 워크스테이션에서 제공하는 모든 기능을 제공하기 어려운 점들이 있다. 특히, 둘 이상의 채널을 출력하는 다중 디스플레이를 단일 PC를 이용하여 제공하는 것이 그렇다. 최근 두개의 모니터 출력력을 지원하는 그래픽 카드가 개발되었지만 높은 해상도와 실시간 처리를 함께 지원하는 것에 문제가 있으며 둘 이상으로 확장하는 것도 불가능하다.

본 연구는 이를 극복하기 위하여 다수의 PC를 이용하여 다중 디스플레이를 구현한다. 이때 모니터 상의 채널들을 동기화 하는 메커니즘이 필요하다. 그리고, 가상공간은 3차원의 그래픽 랜더링 뿐만 아니라 다양한 입출력 장치와 가상공간의 융통성 있는 제어를 지원해야 한다. 따라서, 우리는 다중 디스플레이와 가상공간의 효율적인 운영을 위하여 PC 클러스터를 기반으로 하는 가상현실 시스템을 제안 한다.

2. 시스템 개요

제안하는 시스템은 LAN에 연결된 복수 대의 개인용 컴퓨터들을 이용하여 참여자와 실시간 상호작용 하는 가상공간을 생성한다. 우리는 각각의 컴퓨터들을 고유의 기능에 따라 다음과 같이 분류 한다.

- VR 서버 : 가상공간을 위하여 동적으로 생성하는 3차원 모델 데이터를 실시간 제시
- 제어 서버 : 가상공간의 임의의 속성 또는 상태를 GUI 인터페이스를 통하여 비정기적인 커맨드 형태로 생성
- 장치 서버 : 상호작용을 위하여 입출력 장치를 통하여 가상공간을 연속적으로 생성



<그림 1> 시스템 구성도

이와 같은 접근은 첫째, 요구하는 기능을 병렬로 수행하면서 가상현실을 응용하고자 하는 목적에 따라 서버들을 자유롭게 구성할 수 있다. 즉, 요구되는 기능을 만족하면서 최소의 비용으로 성능의 최적화를 지원한

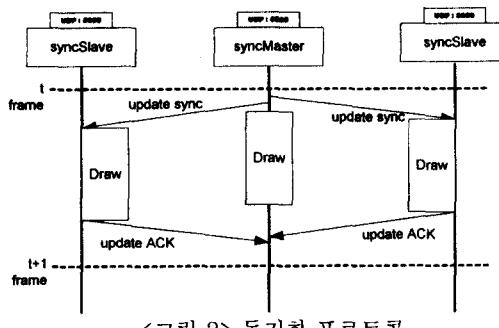
다. 둘째, LAN으로 연결된 서버들 사이에 TCP 또는 UDP와 같은 표준 통신 프로토콜을 이용하기 때문에 서로 다른 운영체계를 탑재한 이질의 호스트들을 이용할 수 있다. 예를 들어, 본 논문과 관련하여 3D 랜더링 서버는 운영 체제 자체를 위하여 요구되는 부담이 적은 리눅스를 기반으로 하며 제어 서버와 장치 서버는 GUI 개발과 기존의 저가 인터페이스 장치의 도입이 유리한 윈도우를 기반으로 하였다.셋째, 가상현실은 매우 다양한 기술을 도입하여 통합하기 때문에 독립적인 개발의 지원이 필요하다. 제안하는 구조는 3차원 음향, 모션 다이나믹, 비전 등의 통합을 고려한 것이다. 이 외에도 플러그-인을 통하여 데이터 또는 제어를 외부 프로그램과 교환 할 수 있는 기존의 상용 프로그램과 통합 할 수 있는 여지를 남겨두었다.

3. 핵심 구성 요소

3.1 다중 채널 동기화

제안하는 시스템은 복수대의 독립적인 호스트들을 이용하여 넓은 시야각의 가상공간을 제시하도록 다중 채널을 지원한다. 즉, 다중 채널이 마치 단일 호스트에서 동작하는 것과 같은 효과를 제공하기 위하여 단일 채널을 제공하는 3D 랜더링 호스트들 사이에 소프트웨어적인 동기화 메커니즘을 도입한다. 이 메커니즘은 프레임이 갱신 할마다 모든 호스트들의 시점(Viewpoint)을 일관되게 유지하도록 한다.

동기화 메커니즘은 아래 그림과 같이 하나의 마스터와 복수개의 슬레이브로 이루어진다. 마스터는 시점이 갱신될 때 나머지 슬레이브들에게 update sync를 통하여 이를 전달한 후에 자신의 가상공간에 반영한다. 그리고, 나머지 슬레이브들은 전달 받은 시점을 그들의 가상공간에 반영한 후에 마스터에게 update ACK를 전달한다. 이때, 마스터는 나머지 모든 슬레이브들로부터 update ACK가 도착할 때 가지 다음 프레임으로 진행하지 않고 기다린다.



<그림 2> 동기화 프로토콜

이와 같은 동기화가 매 프레임마다 이루어 지면 시스템은 전체 프레임 갱신률을 가장 낮은 프레임 갱신률을 가지는 호스트에 맞추게 된다. 결국, 각각의 호스트가 서로 다른 시점에서 가상공간을 제시할 때 Draw 시간이 동일하지 않을 수 있기 때문에 이를 고려한 것이다.

이때 가상현실의 일반적인 갱신률은 약 60Hz이다. 즉, 한 프레임을 Draw하기 위하여 요구되는 시간이 약 16ms 이

다. 동기화를 위하여 LAN으로 연결된 마스터가 슬레이브에게 update sync 또는 update ACK와 같은 메시지를 전달하는 소요 시간은 0.01ms 이다. 따라서, 메시지 교환으로 인한 네트워크의 자연이 전체 프레임의 동기화에 큰 영향을 미치지 않을 것이다.

3.2 상호작용 관리자

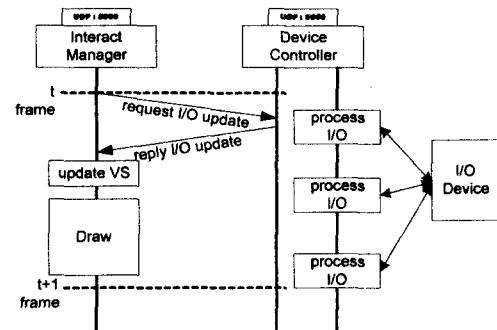
제안하는 시스템은 참여자와의 상호작용을 지원하기 위하여 장치 서버와 가상공간을 연결하기 위한 상호작용 관리자를 랜더링 서버에 도입한다[3].

가상공간의 상태는 상호작용에 의하여 계속하여 갱신한다. 가상공간은 상호작용을 위하여 참여자는 장치서버에 연결된 입출력 장치를 통하여 갱신할 수 있어야 한다. 상호작용 관리자는 입출력 장치의 갱신을 가상공간에서 아바타의 시점에 적용하는 네비게이션(Navigation) 또는 가상공간 내의 물체에 적용하는 매니퓰레이션(Manipulation)을 지원한다.

이때 장치 서버에서 입출력 장치를 제어하는 것과 랜더링 서버에서 프레임을 갱신하는 것이 다중 채널 동기화 때와 유사하게 서로 다른 갱신률을 가질 수 있다. 하지만, 자연스러운 상호작용을 보장하기 위하여 장치서버의 갱신률이 랜더링 서버보다 반드시 높아야 한다.

아래 그림과 같이 상호작용 관리자는 매 프레임이 시작할 때마다 장치 서버에 현재 입출력 장치의 갱신 값을 요구하여 이를 가상공간에 반영한다. 이 때 장치 서버는 상호작용 관리자 처리와 별도로 입출력 장치를 별도로 처리한다. 이때, 장치 서버에는 하나 이상의 장치를 제어할 수 있다.

이와 같은 접근은 각 장치의 서로 다른 갱신률과 관계 없이 랜더링 서버의 프레임 갱신에 의존하여 입출력 장치의 결과값을 가상공간에 반영시킬 수 있다.



<그림 3> 상호작용 프로토콜

3.3 이벤트 관리자

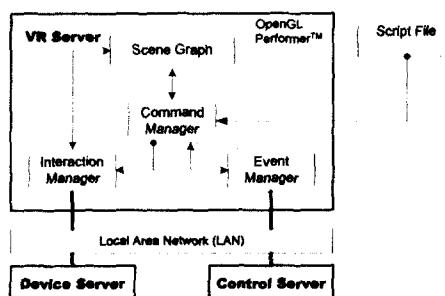
제안하는 시스템은 실시간 상호작용 뿐 아니라 비정기적으로 발생하는 가상공간의 갱신 요구를 처리할 수 있도록 이벤트 관리자를 도입한다. 본 연구와 관련하여 제어 서버의 GUI 인터페이스를 통하여 가상공간의 갱신 명령이 전달될 때 이벤트 관리자는 이를 받아서 처리한다.

이때 제어 서버는 가상공간의 개선 명령 뿐 아니라 가상공간의 입의의 속성에 대한 질의 명령을 이용할 수 있다. 즉, 가상공간이 입의의 상태에 만족할 때 이를 조명 또는 음향과 같은 실제 환경에 통합하는 것을 지원한다. 우리는 요구되는 명령을 수신하는 큐를 이용하도록 이벤트 관리자를 설계 한다 [4].

4. 구현

우리는 3대의 VR 서버와 1대의 제어 서버로 구성된 가상 현실 시스템을 구현하였다. VR 서버는 리눅스 운영 체제를 탑재하고 3차원의 그래픽 랜더링을 위하여 OpenGL Performer 라이브러리를 이용한다. OpenGL Performer는 SGI사의 워크스테이션에서 실시간 시뮬레이션을 지원하는 라이브러리로 최근 리눅스 기반에서 동작하는 라이브러리가 출시되었다. 제어 서버는 윈도우를 탑재하고 MFC 라이브러리를 이용하여 GUI 인터페이스를 개발하였다.

아래 그림은 VR 서버를 중심으로 앞에서 제안한 상호작용 메니저와 이벤트 메니저 그리고 장치 서버와 제어 서버와의 연결을 보여준다. 우리는 가상공간 자체와 다른 서버와의 효과적인 운영을 위하여 가상공간을 오프라인으로 제어하는 스크립트 파일과 수행 중에 조정하는 제어 서버로 구별하였다. 그리고, 이 명령을 처리하는 명령어 관리자를 새로 추가하였다.



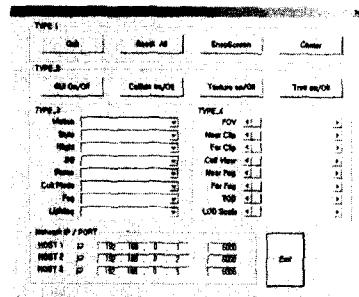
<그림 4> 구현 사례

아래 그림은 구현된 시스템으로 3채널을 제시한 결과를 보여준다. 모든 채널의 FOV(Field of View)는 45도이며 좌, 우 채널은 ±45도 만큼 회전된 시점을 가진다. 본 시스템은 초당 60 프레임의 개선을 보일 때 안정적으로 동작 했으며 모델의 복잡도에 따라 개선률이 변화할 때 모든 채널이 가장 낮은 채널의 개선률로 조정되는 것을 확인 할 수 있었다.



<그림 5> 3 채널 디스플레이

다음 그림은 가상공간의 속성을 제어 서버에서 조정할 수 있도록 제시되는 GUI 인터페이스이다. GUI의 형태는 명령의 특징에 따라 버튼, 토글 버튼, combo 박스, 스크롤 바로 분류된다. 이때 제어 서버의 명령은 VR 서버들 사이에 별도의 동기화 메커니즘 없이 단순히 모든 VR 서버에 전달한다.



<그림 6> 제어 서버의 GUI 인터페이스

5. 결론 및 향후 연구

제안하는 시스템은 저가의 PC 서버로 구성된 클러스터로 가상현실을 구현하도록 지원한다. 서버는 기능에 따라 VR 서버, 장치 서버 그리고 제어 서버로 분류한다. 이때 LAN으로 연결된 서버들은 UDP를 통하여 통신한다. 제안하는 시스템의 핵심 기능은 다중 디스플레이를 위한 채널 동기화, 외부 장치를 통한 가상공간의 연속적인 개선을 지원하는 상호작용 관리자 그리고 가상공간 개선 명령의 처리를 위한 이벤트 관리자에 의하여 구현된다.

향후 연구 과제는 다중 디스플레이와 함께 입체 영상을 제시하는 것이다. 채널 동기화는 VR 서버들 사이의 시점(Viewpoint)를 일치시킨다. 하지만, 입체 영상을 제시하기 위하여 각 디스플레이 장치의 비디오 개선의 일치를 보장해야 한다. 또한, 우리는 제안하는 시스템을 기반으로 4면의 CAVE를 구현할 것이다.

참고 문헌

- [1] Sandeep Singhal, Michael Zyda, "Networked Virtual Environments", ADDISON-WESLEY, 1999
- [2] 장상철, 박창훈, 서형준, "분산 시뮬레이션을 위한 기반기술 소개", '97 Seminar and Exhibition for EW & Simulation Systems, 1997
- [3] 박창훈, 박경동, 고희동 "가상현실 시스템을 위한 외부 모듈 인터페이스의 개발", HCI 학회, 1999
- [4] Benjamin Schaeffer, "A Software system of inexpensive VR via Graphics Clusters", ,