

# PC클러스터를 이용한 스테레오 서라운드 디스플레이

최정단<sup>0</sup>, 변기중, 장병태  
한국전자통신연구원 공간정보기술센터  
{jdchoi<sup>0</sup>, kbyun, jbt}@etri.re.kr

## A Stereo Surround Display Using PC Clusters

Jeongdan Choi<sup>0</sup>, Kijong Byun, Byungtae Jang

Electronics and Telecommunications Research Institute, Spatial Information Technology Center

### 요 약

본 논문은 온라인 VR 게임 엔진의 일환으로 몰입시각화 방법의 구현에 관한 것이다. 몰입영상을 제공하기 위한 방법으로 광역의 시야각을 지원하는 프로젝션 디스플레이 구현과 멀티 채널 디스플레이의 스테레오 서라운드를 지원하기 위한 여러 대의 PC 렌더링 시스템, 그리고 이를 지원하는 애니메이션 동기화 방법에 대한 구현상의 방법을 소개 하고자 한다. 새로운 VR 기술들이 점차 시스템의 저가화에 따라 사용자에게 새로운 몰입형 아케이드 게임 환경을 지원하기에 이르렀다. 본 논문에서는 프로젝션 기법을 사용하여 스테레오 이미지 디스플레이가 가능한 방법과 멀티 채널 시스템을 이용한 멀티 모니터를 이용하는 방법에 대한 구현이다. 구현된 방법은 멀티 클러스터 시스템과 멀티 프로젝션 디스플레이 시스템, 스테레오를 구현, 그리고 애니메이션의 동기화를 위한 방법으로 나누어 설명하고자 한다.

### 1. 서 론

사람의 오감 중 다른 감각에 비해 시각은 외부로부터 받아들이는 정보 중의 70%를 차지한다[1]. 가상현실감을 전달 하고자 할 때 몰입형 시각 장치에 대한 고려는 올바른 방법으로서 반드시 고려되어야 한다. 기존 게임은 전통적으로 주 디스플레이 장치로서 하나의 모니터를 사용하고 있다. 하나의 모니터 디스플레이는 저렴하고 적용이 단순한 반면에 사용자에게 몰입감을 전달 하는 면에서는 부족한 감이 있다. 최근 들어 VR 게임의 성장이 가속화 됨에 따라 HMD 또는 대형 모니터 또는 프로젝터와 같은 다른 디스플레이 장치가 등장하고 있다. 이러한 디스플레이 장비들은 사용자들에게 VR 게임 환경에의 몰입감을 효과적으로 전달 하는 것으로 알려져 있다.

멀티 채널 디스플레이는 사용자에게 개인적인 공간으로 몰입감을 제공하는HMD와는 다른 형태로 광역의 FOV(Field Of View)를 제공한다. 이러한 멀티채널 디스플레이를 가능하게 하는 것은 저가의 PC 시스템을 클러스터로 연결하여 렌더링의 분산처리가 가능하여 기존의 고가의 그래픽 전용 워크스테이션에서나 가능한 작업을 수행하게 하는 것이다.

뿐만 아니라, 스크린을 사용하는 프로젝션 시스템의 사용은 이미 고가의 비주얼라이제이션 시스템에서 널리 사용되고 있다. 이러한 장비의 예로는 CAVE[1][2] 또는 DOME 등이 있으며, 이러한 세트 장비들의 디스플

레이 면의 연결과 적당한 각도의 위치를 잡도록 하기 위하여 많은 노력과 시간을 소모하게 된다.

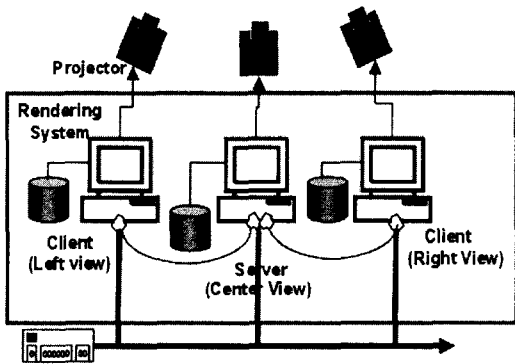
이러한 단점을 보완하고 광역의 FOV 출력의 요구에 부응하기 위하여 본 논문에서는 PC 클러스터링 방식에 의하여 각각의 PC가 할당된 영역만을 렌더링하며 디스플레이 결과를 스크린 표면을 이용하여 프로젝션 시키는 방식을 도입하고자 한다. PC 그래픽스 시스템은 과거 몇 년에 걸쳐 가격대 성능면에 있어서 급속한 발전을 거듭하고 있으며 전통적인 고가의 렌더링 시스템 성장을 앞지르고 있는 실정이다. 그러므로 다수의 PC 시스템을 이용하는 그래픽스 작업은 협동작업이 요구되는 CAD (Computer-Aided Design), 가상화 작업, 메디컬 또는 엔터테인먼트 비주얼라이제이션 같은 응용 분야에서 그 중요성이 증대되고 있다[3]. 고화질의 디스플레이는 매우 상세한 데이터 셋을 가시화 할 수 있을 뿐만 아니라 향상된 몰입감 제공과 사용자의 보다 나은 3차원 오브젝트의 인지를 위하여 필수적인 요건이다.

광역의 FOV 렌더링을 위한 노력으로 많은 연구들이 듀얼 헤더 디스플레이 형태를 시뮬레이션 하는 다양한 기술개발로 이루어졌다. 다수의 디스플레이 제품들은 하나의 렌더링 시스템에 연결되어 광역의 FOV를 시뮬레이션 해 보였다. 그러나, 현대의 PC 는 여전히 광역의 FOV를 계산해 내기에는 역부족이었으며 이로 인하여 몰입감을 제공하는데 실패 하였다. 최근에 이와 같은 연구들은 클러스터링된 PC 를 이용하여 멀티 채널

을 이용하는 방법으로 이루어지고 있다[3]. 클러스터로 구성된 각각의 PC들은 자신 고유의 할당된 시점의 영역만을 렌더링하여 분리된 디스플레이에 렌더링 하며, 또한 네트워크 프로토콜을 이용하여 동기화를 맞추게 된다. 그러나 여전히 서라운드 및 파노라마 디스플레이를 지원하기 위하여 다양한 동기화 지원 문제를 해결하여야 한다. 본 논문에서는 이러한 문제점 해결을 위하여 Direct3D[6]를 이용한 뷰 매트릭스의 구성 방법과 애니메이션의 동기화를 구현한 방법을 설명한다. 본 논문의 구성은 다음과 같다. 첫 번째로 PC 클러스터 시스템 구성 방법과 멀티채널 디스플레이의 구현 방법, 애니메이션 동기화 방법을 설명한다. 그리고 프로젝터를 사용하여 스크린간의 에지 블렌딩 방법과 스테레오 방법에 관한 방법을 설명한다. 마지막으로 현재 수행 중인 연구 방법의 결과와 결론을 맺도록 한다.

### 2. PC 클러스터 시스템

현재 진행 중인 연구에서는 다양한 사용자가 참여하는 온라인 VR 게임을 목표로 한다. 본 시스템은 네트워크로 연결된 클러스터 PC들과 이들의 애니메이션 비롯한 데이터와 이벤트의 동기화를 위한 방법, 그리고 모니터들과 프로젝션 디스플레이 시스템으로 구성된다. 그림 1은 이러한 시스템 구성도를 나타낸다.



(그림 1) PC 클러스터 시스템 구성도

### 3. 멀티 채널 디스플레이

멀티채널 디스플레이 시스템의 목적은 광역의 FOV를 통하여 가상현실 온라인 게임을 디스플레이 한다. 클러스터링된 각각의 PC 시스템은 제어하는 PC로부터 전달 받은 메시지를 가지고 자신의 고유한 영역만을 계산하여 렌더링하며, 역시 제어 하는 PC 로부터 전달 받은 렌더링 메시지에 따라 자신의 디스플레이를 이용하여 나타낸다. 이러한 제어 PC와 피 제어 PC간의 프로토콜을 위하여 TCP/IP 또는 IPX를 이용한다.

클러스터링된 PC를 이용하여 모니터 혹은 프로젝터를 이용한 스크린멀티채널 시스템을 구현하고자 할 때 공통으로 이용되는 뷰 매트릭스 조작에 관하여 설명한다.

### Camera Matrix Configuration

디스플레이 상에서 중앙에 해당하는 카메라에 대하여 왼쪽과 오른쪽 카메라는 이를 중심으로 회전을 시켜야 한다. 이때 모니터일 경우에는 중복을 피하여야 하며, 프로젝터의 경우에는 중복을 허용하여야 한다. 모니터에 대하여서는, D3DXMatrixRotationAxis 의 회전각 파라미터에 global aspect ratio로 나눈 값을 전달한다. 이렇게 하는 이유는 메모리 버퍼의 사이즈가 넓이와 높이의 비율로 결정되는 aspect ratio를 고려 하지 않았을 경우 회전에 의한 왼쪽 오른쪽의 카메라의 올바른 결과를 얻을 수 없다. 여기서 왼쪽의 모니터에는 각도의 플러스 회전 값을 전달하여야 하며, 오른쪽 카메라에 대하여 마이너스 회전 값 이다. 최종적으로 LookAt matrix 에 카메라 matrix를 곱하여 뷰 matrix 가 결정된다. 수식 1에서 3은 이에 대한 것이다.

$$M_{view} = M_{view} \times M_{fov} \quad (1)$$

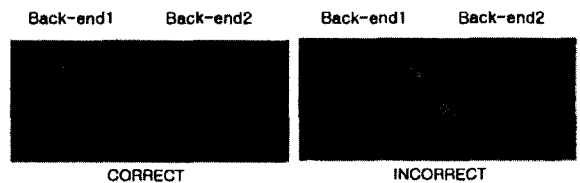
$M_{view}$  는 새롭게 수정된 매트릭스이며,  $M_{fov}$  는 멀티 채널 디스플레이를 위한 것이다. 또한 perspective 프로젝션을 위하여 Direct3D 에서는 다음과 같은 파라미터 셋팅이 요구된다.

$$h = \cos(fov/2) / \sin(fov/2), w = h / aspect \quad (2)$$

이 경우에 수평적으로 정렬된 세 개의 모니터인 경우,  $M_{fov}$  는 양 사이드의 모니터에 대한 FOV를 위한 것이며, 각 회전각 계산은 다음과 같이 이루어진다. 원래의 뷰 매트릭스  $M_{view}$  의 Up 벡터를 수식 (3)에서와 같이  $\theta$  값 만큼 회전하기 위한 것이다.

$$\theta = 2 \tan^{-1}(\tan(fov/2) \times aspect) \quad (3)$$

이러한 보정이 이루어진 결과는 다음 그림(2)와 같다.



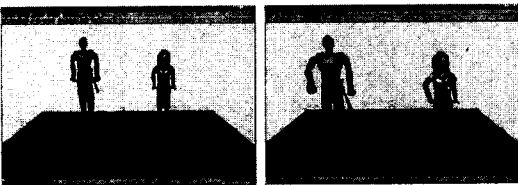
(그림 2) 인접 모니터 간의 보정된 멀티채널디스플레이

4. 캐릭터 애니메이션

모델링 데이터의 스킨 매쉬파일을 Direct3D 를 이용한 그래픽 라이브러리에서 사용하기 위하여 X파일로 저장한다. 이러한 방법은 기존의 데이터 활용을 위하여 아주 편리한 방법이며, 최근 DirectX 의 9.0 버전에서 제공된 X파일 컨버팅 프로그램은 텍스처어 부분이 깨어지는 등의 기존의 문제점이 많이 보완되었다. 뿐만 아니라, X파일은 3D studio Max와 같이 보편적인 모델러로부터 Biped 정보와 애니메이션 데이터를 포함 할 수 있어 확장 가능한 애니메이션 파일이다.

본 논문에서 이용하고 있는 모션 데이터는 Biovision사의 Bvh 파일 포맷이며, 이를 스킨레톤으로 이루어진 캐릭터 파일에 적용하기 위하여 아래 수식의 방식을 거쳐게 된다. 그리고 그림3은 애니메이션 초기 프레임과 마지막 프레임의 다른 캐릭터에 대한 다른 모션이 적용된 결과를 보여준다.

(그림 3) 동일 모션이 다른 캐릭터에 적용되었을 때의 프레임 비교



Biped 데이터는 뼈의 기본 축으로서 x-axis을 이용하는 반면에 Bvh 파일은 임의의 축을 사용한다. 이러한 문제점을 해결하기 위하여 Bvh의 정규화된 bone-axis (bx, by, bz) 을 y-axis 으로 맞춘 후, 로컬 코디네이션 시스템을 ( x, y, z ) → ( y, z, x ) 와 같이 서클레이션 된다. 그래서 x-axis 이 뼈의 기본축과 맞출수 있다. 수식 4와 5에서 축 R<sub>axis</sub> 이 θ 을 이용하여 표현 되었다.

$$R_{axis} = (bx, by, bz) \times (0, 1, 0) = \begin{vmatrix} i & j & k \\ bx & by & bz \\ 0 & 1 & 0 \end{vmatrix} = (-bz, 0, bx) \quad (4)$$

$$\theta = \cos^{-1}[(bx, by, bz) \cdot (0, 1, 0)] = \cos^{-1} by \quad (5)$$

n번째 모션에 있는 i번째 세그먼트(본)을 렌더링 하기 위하여 실제계 좌표계인 W<sub>n</sub> 는 수식 6과 같은 매트릭스에 의하여 표현될 수 있다.

$$W_n = T_{n1} \cdot M_1 \cdot M_1^{-1} \dots T_{n(i-1)} \cdot M_{i-1} \cdot M_{i-1}^{-1} \cdot T_{ni} \cdot M_i \quad (6)$$

5. 스테레오 이미지 디스플레이

PC 모니터를 이용한 스테레오 뷰잉 방식은 인터페이스와 페이지 폴립 두 가지가 있다. 하드웨어 안경이 지원하는 방식에 따라서는 셔터 글래스와 애너굴리프 방식이 있다. 모니터의 요구조건으로는 CRT 방식이어야 하고, 그래픽 카드의 주파수가 120Hz 이상 지원을 권장하고 있다. 하지만 85Hz에서도 스테레오를 감상하기에 큰 불편함은 없다. 또한 안경의 스테레오를 가능하게 하는 소프트웨어 모듈이 있어야 한다.

한편, 프로젝션 시스템[4][5]의 경우에는 편광 필터를 이용하여 프로젝터 앞에 장착함으로써 가능해진다.

6. 구현결과

본 논문에서는 Pentium 4, CPU 2GHz 와 NVIDIA GeForce 3 그래픽 카드를 이용하였다. 3D 스테레오를 위한 드라이버로서는 nvidia stereo kit와 detonator XP 소프트웨어를 사용하였으며, 국내산 VRFX 안경으로 이용하여 구현하였다. 그림 4에서는 스테레오는 나타내지 못하였고, 애니메이션의 동기화 결과와 3개의 모니터를 이용한 멀티채널 시스템 결과를 나타내고 있다.



(그림 4) 멀티채널 디스플레이 시스템

참고문헌

[1]Smith, P.H and van Rosendale, J., eds., Data and Visualization Corridors(DVC) in report CACR-164, Sept. 1998

[2]Funkhouser, F and Li, K., Large-Format Displays, IEEE CG&A 20, 4, (July/August 2000), 20-21

[3]Brian, W, Constantine, P and Vasily, L, Scalable Rendering on PC Clusters, IEEE CG&A (July/August 2001), 62-70

[4]Ramesh. R and Paul B., A Self Correcting Projector, TR-2000-46, (Jan 2002),

[5]Zhang, Z., Flexible camera calibration by viewing a plane from unknown orientations, Computer Vision (1999), 666-673

[6]Microsoft Corporation. DirectX 8.1 programming reference, <http://www.microsoft.com/windows/direct>