

□ 신기술 해설 □

Virtual Studio 기술 소개

고 회 동 †

◆ 목 차 ◆

- | | |
|--|-------------------------|
| 1. Virtual Studio 개념 | 4. 3차원 그래픽 배경 이미지 생성 과정 |
| 2. Chroma-keying에서 Virtual Studio로 발전 과정 | 5. 가상 세트 구축 |
| 3. 카메라 인터페이스 | 6. 맺는말 |

1. Virtual Studio 개념

인간과 컴퓨터의 원활한 인터페이스, 다양한 정보의 전달 및 표시, 3차원 개체의 용이한 처리를 위하여 다수의 어플리케이션들이 급격히 3차원 그래픽 기능을 수용하고 있는 추세이다. 따라서 이러한 3차원 컴퓨터 그래픽 기능은 이제 단순한 기하학적 형태를 다루는 것에 그치지 않고, 3차원 영상 매체의 한 분야로 발전되어 인간에게 더 큰 시각적 사실감을 부여함으로써 생산성을 향상시킬 수 있는 다양한 응용 분야로 파급되고 있다.

3차원 영상 매체로써 3차원 그래픽의 다양한 응용 분야의 예는 가상 현실(Virtual Reality), 시뮬레이션, 게임, 방송 및 영상, 설계, 자료 시각화 등등 다양한 분야를 들 수 있다. 이러한 다양한 응용을 위해서 컴퓨터 그래픽은 단순한 기하학적 형태의 표현 뿐만 아니라, 효율적인 그래픽 개체

의 저장 및 운영, 실시간(Real-Time) 처리, 각종 주변 장치와의 인터페이스 및 연동, 실세계와의 자연스러운 접목을 위한 더욱 발전적인 기능을 갖추어야 한다.

현재 가상 현실 기술을 방송, 영상물 제작에 적용 그 실효성을 인정 받고 있는 가상 스튜디오(Virtual Studio, Virtual Set) 기술은 카메라로 촬영된 실제 영상과 컴퓨터 그래픽에 의해 생성된 가상 배경 및 가상 개체를 자연스럽게 그리고 동적으로 합성함으로써 실제 촬영된 인물이나 물체가 마치 가상의 세계에 존재하는 것처럼 시청자에게 보여지도록 하는 기술이다. 이러한 가상 스튜디오 기술을 방송 및 영상물 제작에 사용함으로써 실제 촬영 세트를 만들 필요가 없게 되고 특히 실제 세트로는 구현할 수 없는 환상적인 다양한 장면을 연출할 수 있는 장점이 있다. 따라서 더욱 경제적이면서도 신속하게 동적이고 다양한 분위기의 영상물을 제작하여 방송할 수 있다.

현재 이러한 이점 때문에 세계적으로 가상 스튜디오를 실제 방송에 도입하는 추세이고, 국내에

† 정회원 : 한국과학기술연구원 책임연구원

서도 각 방송사들이 '96년 국회의원 선거 방송을 기점으로 '97년 대선 방송에서도 경쟁적으로 그 도입을 시도하고 있다. 하지만 현재의 기술 수준은 매우 초보적인 수준으로 단순한 배경 처리 정도에 그치고 있다. KIST에서는 KIST 2000 과제의 세부 과제로 이러한 현재의 기술 수준을 극복하고 더욱 다양하고 효과적인 가상 스튜디오의 운영, 새로운 특수 효과, 정보의 효과적 전달을 위한 인터랙티브한 가상 개체 및 3차원 그래픽 차트/문자 기능 등을 갖춘 새로운 차원의 가상 스튜디오 시스템의 개발에 그 역점을 두고 있다.

이를 위해서 본 연구에서는 최소 30Hz 의 Frame Rate 가 보장되어야 하는 가상 배경의 그래픽 생성 속도를 지원하기 위한 최적 실시간 그래픽 생성 기술을 개발하고 특히, 더욱 효과적인 속도 향상을 위하여 Multiprocessing 기술을 적용하였다.

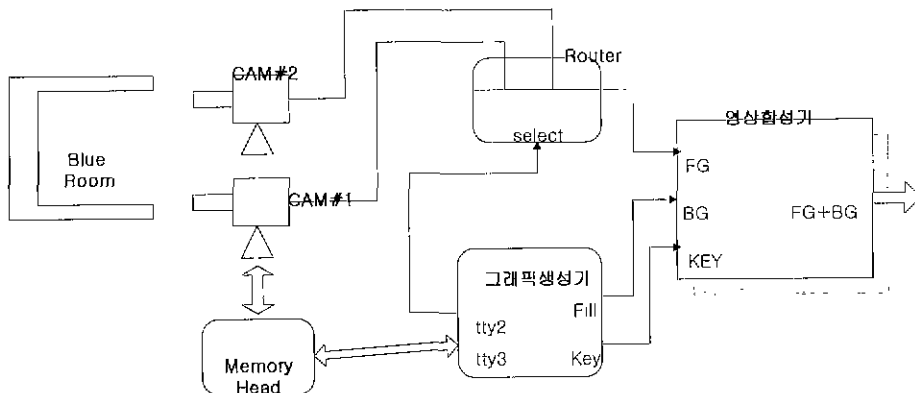
또한, 속도, 확장성, 일관성을 유지한 그래픽 및 가상 스튜디오의 운영 정보를 수용할 수 있는 자료 구조를 개발하고 개체 지향적 인터렉션 기능을 통해 실제 영상, 가상 배경/개체 들이 서로 상호작용하고, 스케줄, 메시지(Message), Animation Sequences 에 의해 손쉽게 다양하게 동작 될 수 있도록 하였다.

실제 촬영된 영상과 가상 컴퓨터 그래픽 배경 및 개체들의 동적이고 자연스러운 합성을 위해서는 실제 카메라와 가상 카메라가 동기화 될 수 있도록 카메라의 Pan, Tilt, Zoom, Focus 측정 값의 전달 및 전처리 그리고 그래픽 Frame Rate 의 변화에 따른 지연(Delay) 에 대한 보상 처리 기능을 개발하였다

2. Chroma-keying에서 Virtual Studio 로 발전 과정

실제 세트는 파란 바탕의 배경 방 (Blue Room) 에서 촬영하지만 실제 세트 대신에 3 차원 가상 세트로 대체되어 진행되는 가상 세트에서 진행되는 것으로 보여진다. 여기서 가상 현실 기술은 가상 환경을 실시간으로 모사하고 가상 세트에서 진행자가 존재하는 것 같은 실시간으로 3 차원에서 실사 영상과 모사 영상을 합성 하는데 활용 된다. Virtual Studio 는 이렇게 실제 세트를 대체하는 가상 세트를 운영/저작하는 통합 시스템이다.

그림 1은 보편적으로 사용되는 Virtual Studio 시스템 구성 도로서 FG (foreground) 영상을 촬영하는 1 대 이상의 카메라와 카메라의 움직임, 시야



(그림 1) Virtual Studio 시스템 구성도

각 정보를 컴퓨터에 제공하는 Memory Head와 같은 추적 (Tracking) 시스템이 보편적으로 활용되고 있다. 실사 카메라와 같은 위치와 시야 각을 추적, 반영한 가상 카메라에서 BG (Background) 영상과 실사 촬영 화면을 동기화 하여 실시간으로 렌더링 하여야 한다. 여기서 보편적으로 사용되는 장비가 Onyx Reality Engine과 같은 그래픽 슈퍼 컴퓨터가 보편적으로 활용되고 있다. FG 영상과 BG 영상은 합성 (FG+BG)되어 후 처리 장비로 (믹서, Special Effect장비, 모니터, 리코더 등) 연결된다. 따라서 Virtual Studio 시스템은 추적 시스템, 렌더링 시스템, 영상 합성 시스템으로 구성 되어진다.

그림 1 과 같은 Virtual Studio 시스템에서는 Chroma-keying 영상 합성 기법이 필수적으로 사용된다. 기존의 Chroma-keying 기법은 Blue Screen 앞에서 촬영한 영상을 FG 이미지로 BG 이미지를 Chroma-keyer에 입력 합성하는 기법이다. Chroma-keying 기법은 방송국에서 일기 예보 등에서 이미 보편적으로 사용하고 있다. 그림 2처럼 아나운서가 일기 예보 지역의 지도와 일기 상태를 보여주는 도표 앞에서 방송한다. 여기서 아나운서는 Blue-screen 앞에서 실제 카메라가 촬영한 FG 이미지와 지도와 도표가 합성된 BG 이미지 화면을 각각 입력하여 Chroma-keyer에서 합성된 이미지를 방송하므로 아나운서가 지도와 도표 앞에서 방송하는 장면을 연출하는 것이다.

개념적으로 Chroma-keying 기법의 의미를 요약한다면 FG 이미지에서 특정한 색상 값 (Blue Screen 경우 파란색) 을 갖고 있는 부분만을 BG 이미지로 대체하는 기법이다. 여기서 파란색을 배경 색으로 활용하는 것은 피부 색과 대조를 잘 이루어 합성이 사회자의 경계가 잘 유지 되고 사회자가 FG 스튜디오에서 연기할 때에 선호도가 높기 때문이다. 그러나 배경 색상은 반드시 파란

색일 필요는 없다.

색상의 값을 대체하는 간단한 합성 기법에서 가장 중요한 요소는 조명이라고 할 수 있다. 기본적으로 파란 스튜디오 벽이 고른 밝기로 비추어 져야 된다. 조명이 고르되 너무 강하면 파란 스튜디오 벽에서 반사되는 파란 색상의 간접 조명이 사회자에게 비추어 지는 Blue Spill 현상이 발생한다. 영상 합성 과정에서 보정할 수는 있으나 배경 조명이 없어지므로 사회자에게 배경보다 강한 조명을 다시 때려줄 필요가 있으므로 FG 이미지와 합성되는 BG 이미지에서 조명의 밝기를 같은 정도로 유지하여야 합성 결과가 좋다.

방송용 Chroma-keying 시스템은 사회자 경계선을 부드럽게 하고 FG 이미지 상에서 나타나는 그림자 같은 섬세한 부분을 합성된 영상에서도 볼 수 있도록 다양한 기법을 사용하고 있다. 만약, 사회자나 파란 색상의 실제 소품에 의하여 그림자가 오른쪽으로 지는데 합성될 BG 가상 세트의 소품은 오른쪽으로 그림자가 드리워지면 합성 결과가 부자연스럽게 보일 것이다. 따라서 가상 세트 설계 시에는 실제 FG 조명 사정을 반영하여 설계되어야 한다.

좋은 합성 화면을 위해서는 Chroma-keying 시스템 자체의 기능을 최대한 발휘할 수 있도록 FG 이미지 촬영 현장의 설계가 필수적이다. 특히, 가상 스튜디오에서 처럼 바탕 스튜디오가 벽 뿐만 아니라 바닥, 좌, 우면 까지 파란 색인 Blue Room일 경우 경계면에서 그림자가 지는 것을 방지하여야 한다. 이 경우 경계면을 곡면 처리하는 방법을 많이 사용한다.

기존의 Chroma-keying 시스템에서는 FG 이미지의 사회자를 촬영하는 카메라는 고정된 위치에서 화면을 잡도록 운영하여야 한다. 만약, 카메라가 좌,우로 움직이면 고정된 배경 화면에서 사회자는 우,좌측으로 미끄러지듯이 움직여 보인다. 또한,

카메라가 앞, 뒤로 움직이거나 렌즈 Zooming을 하면 사회자는 커지거나 작아진다. 즉, 지도와 도표 앞에 진행자가 서서 일기 예보하는 환상이 사라지고 두 개의 독립된 화상이 화면을 차지하고 있다는 느낌을 받게 된다. 따라서 카메라가 움직이면 FG 이미지와 BG 이미지 사이의 위치 관계성이 단절 되므로 FG 이미지를 촬영하는 카메라를 고정시키는 방법으로 Chroma-keying 시스템이 방송에 활용되었다.

FG 카메라를 움직이면서 BG, FG 이미지 소스 사이의 관계성을 유지하기 위해서 다양한 기법이 시도되었다. 일본 NHK에서 활용한 Synthesvision 시스템의 기법은 FG 카메라의 움직임, 초점 거리를 추적, BG 이미지를 변형한 후 Chroma-keyer에서 합성하였다. 93년 영국 BBC 방송국에서는 BG 이미지를 생성하는 가상 카메라의 움직임, 초점거리 값을 기록한 다음 그 순서대로 FG 카메라의 변수를 작동 시켜 FG 이미지와 BG 이미지를 순서대로 합성 시키는 방법으로 뉴스 프로그램에 실험하였다.

두 가지 기법 모두 추구하는 목표는 FG를 촬영하는 카메라의 변수와 BG를 화면으로 투영하는 가상 카메라 변수를 일치시키는 것이다. Virtual Studio 시스템은 위 두 가지 기법의 공통 분모 형태로 FG 카메라와 BG 가상 카메라의 동기화, 상호 제어를 통하여 실시간으로 두 카메라를 일치화 시키므로 FG 이미지와 BG 이미지 사이의 위치 관계성을 유지 시킬 수 있다.

영상 합성 장치는 Foreground 와 Background 로 각각 입력된 두 비디오 신호를 합성하게 된다. 이 두 장면을 합성할 때 Foreground 에 있는 파란색 신호는 제거되고 그 부분에 Background 의 비디오 신호가 대체 된다. 결과적으로 그래픽 배경 앞에 촬영 인물 이나 물체가 나타나게 되어 마치 그래픽 배경 속에 인물이나 물체가 존재하는 것

처럼 느껴지게 된다.

본 연구에서는 단순한 배경 처리 뿐만 아니라 그 배경 속에 나타나는 다양한 그래픽 물체, 차트, 문자 등등의 그래픽 개체를 다루고 있기 때문에 이러한 요소를 고려하여야 한다. 하지만 그래픽 개체를 단순히 그래픽 배경과 동일하게 Background 로 처리할 수는 없다. 왜냐하면 이러한 개체는 촬영 인물이나 물체보다 앞에 위치할 경우도 있기 때문이다.

특별한 지정을 하지 않을 경우 그래픽 개체도 동일한 그래픽 Rendering 결과가 비디오 신호로 변환되어 영상 합성 장치의 Background 입력으로 연결 되기 때문에 그래픽 배경과 구분이 없다. 따라서 그래픽 배경과 구분하여 그래픽 개체가 촬영 인물 또는 물체 보다 앞에 위치하도록 합성하기 위해서는 그래픽 개체가 그려지는 Pixel 부분의 Color 의 Alpha 값을 조절함으로써 영상 합성 장치가 해당 Pixel 부분만을 Foreground 입력 신호의 앞에 위치시켜 합성하도록 한다.

본 연구에서 사용한 비디오 처리 장비인 SGI사의 SIRIUS VIDEO 는 그래픽 출력을 비디오 신호로 변환 시킬 때 RGB 뿐만 아니라 Alpha 신호도 함께 변환 시키므로 영상 합성 장치는 변환된 Alpha 신호를 수신할 수 있다. 본 연구에서 사용한 영상 합성 장치 Ultimatte 8 에서는 이렇게 전달된 Alpha 신호를 Matte 신호로 입력 받게 된다. 그렇게 되면 Ultimatte 8 은 이 Matte 신호를 기준으로 그 값이 1에 근접하면 해당 Pixel 을 Foreground 신호의 앞부분에, 그리고 0에 근접하면 뒷부분에 표시하게 된다. 결과적으로 특정 그래픽 개체들은 촬영 인물이나 물체 앞에 위치되어 더욱 입체적이고 사실적인 합성 장면을 연출할 수 있다.

요약하면 가상 스튜디오 시스템은 다음과 같이 구성 및 동작 된다. 카메라에 의해 촬영된 영상

은 Switcher 로 연결된다. 입력된 촬영 영상들 중 하나가 선택되어 영상 합성을 위해 Chroma Key 장비로 선택된 영상 신호가 전송된다. 측정된 카메라의 움직임 값은 RS232 로 그래픽 컴퓨터로 전송된다. 그래픽 컴퓨터는 수신된 카메라 움직임 값에 따라 그래픽 배경을 생성 시킨다. 그래픽 배경 내부에는 Video Input 을 통해 들어오는 비디오 영상이 Texture Mapping 되어 나타난다. 이렇게 생성된 컴퓨터 그래픽 배경은 합성을 위해 Video Out 을 통해 Chroma Key 장비로 연결된다. Chroma Key 장비는 촬영된 영상을 Foreground 로 컴퓨터 그래픽 배경을 Background 로 입력 받아 합성을 하고 그래픽 배경 중 각 Pixel 정보에 Alpha 값이 설정되어 있을 경우 그 부분을 Foreground 의 앞에 놓는다. 이렇게 합성이 완료된 영상은 최종적으로 출력 및 방송된다.

3. 카메라 인터페이스

실제 카메라의 움직임에 따라 가상 컴퓨터 그래픽 배경을 적절히 움직이기 위해 실제 카메라와 동일하게 움직이는 가상 카메라를 정의하여야 한다. 가상 카메라의 기본 요소는 Viewing Frustum 과 Viewpoint 로 구성 되어 있다. 이것은 각각 실제 카메라 렌즈의 Field-of-View 와 카메라 위치/움직임과 연관된다. 가상 카메라가 실제 카메라와 동일하게 연동되어 움직이도록 하기 위해서는 실제 카메라의 움직임을 추적할 수 있어야 한다.

추적 기법에는 광학식과 기전식으로 크게 두 가지 기법이 사용되고 있다. 광학식 추적 시스템에서는 Blue Room 벽에 새겨 놓은 기준 점이나 바둑판 모양의 선분의 패턴을 카메라에 잡힌 영상의 상대적인 변환 분석, 상대적 카메라의 위치, 움직임, Zooming값을 인식하는 기법으로 이동성

이 뛰어 나고 카메라의 초기 세팅 값을 정하는 과정이 비교적 간단하다. 여기서 다양한 패턴 인식 기법이 동원되는데 처리 시간이 비교적 많이 걸리므로 비디오 지연을 많이 추가하여야 하고 추출 변수의 정확도가 떨어지고 시간 축 상에서 부드럽게 연결시키기 어렵다. 또한, 카메라가 벽에 너무 가까이 다가가 Focus가 안 맞는 다든지 패턴이 너무 커져 벽의 기준 점이나 바둑판의 패턴을 잃으면 추적 시스템 전체가 쓸모 없이 될 수도 있으므로 돌발 사고를 미연에 방지하기 어렵다.

기전식 카메라 추적 장치는 위치 센서/모터로 구성되어 삼각 지지대, pedestal, boom, dolly 등 카메라를 장착 시키는 다양한 보조 장비에 장착하여 카메라의 위치, 움직임, 렌즈의 세팅을 추적한다. 이러한 전자 기계식 추적 시스템은 정확성이 높고 데이터의 연속성이 부드러운 반면에 카메라를 손에 들고 촬영하는 자유로운 이동성이 떨어진다. 또한, 정확한 변수의 calibration, 카메라의 초기 세팅 값을 정하는 alignment 과정이 복잡하고, 카메라 지지 대의 움직임의 가, 감속이 급속할 경우 구조적 떨림 현상이 나타날 수 있다.

또한, 카메라에는 렌즈의 특성에 따라 고유의 FOV(Field-Of-View) 가 있다. 그리고 Zoom 의 정도에 따라서 이 FOV 가 연속적으로 변화하게 된다. 따라서 컴퓨터 그래픽에 의해서 생성되는 배경은 카메라의 FOV 와 동일하게 맞추어져야 한다. 본 연구에서는 특정 측정 지점에 Marker 를 설치하고 Marker 와 카메라 기준점간의 거리와 카메라 높이를 측정하였다. 그리고 이에 맞추어 가상 카메라와 가상 Marker를 가상 세트 내에 그린 후 그것이 실사 이미지와 겹치도록 FOV를 조정하고, 이 측정을 Zoom 의 최대, 최소 범위 내에서 균일 간격으로 10회 정도 측정, 보간 해주어 측정 오차 및 Zoom 의 변화에 따른 오차를

보정하여 주었다.

가상 카메라의 위치가 설정된 후 가상 카메라의 위치에 대한 상대적 수치로 그래픽 배경의 실제 비율을 결정하여야 한다. 그래픽 배경은 일단 사전에 모델링 되는 것이기 때문에 그 치수 단위는 실제와 다를 수 있다. 따라서 그 그래픽 모델이 사실적인 크기가 되도록 비율이 조정되어야 한다.

치수 조정이 완료된 후에는 카메라의 각 움직임에 대하여 가상 카메라가 동일한 비율로 움직이는지 확인하고 그 움직임이 정확하게 동기화 되도록 캘리브레이션 작업을 수행하여야 한다. 그렇게 하기 위해서는 특정 위치에 그 크기를 알 수 있는 실제 물체를 놓고 촬영하고 가상의 그래픽 배경에도 동일한 위치에 동일한 크기의 가상 물체를 위치시켜 두 장면을 합성시켜 두 물체가 일치하도록 가상 카메라의 위치 및 움직임 비율을 다시 조정한다. 본 과제에서는 각각 1미터 크기의 가상 Box를 모델링 된 스튜디오와 같이 합성하여 표시한 후 적절히 Box의 스케일(Scale)을 조절하면서 가상 스튜디오의 적절한 표시 비율을 찾아낸다.

또한 카메라를 움직여 위치를 바꾸어 다른 위치에서도 두 물체가 일치 하는지 확인하고 일치하지 않을 경우는 다시 비율을 조정한다. 각 변경 지점에서의 비율은 이론적으로는 동일하여야 하나 측정 오차 때문에 다르게 나타날 수 있다. 따라서 각 측정 지점에서의 비율을 입력하여 오차를 보상할 수 있도록 하여야 한다. 이렇게 되면 실제 카메라의 움직임에 따라 촬영 영상과 그래픽 영상은 자연스럽게 합성되어 보이게 된다.

본 연구에서는 기전식 추적 시스템인 ULTIMATTE사의 MEMORY HEAD라는 제품을 사용하였다. 메모리 헤드는 카메라와 함께 삼각대에 장착되어 카메라의 움직임(Pan, Tilt, Zoom, Focus)

에 관한 정보를 측정하여 그 결과를 출력하는 기능을 보유하고 있기 때문에 카메라의 움직임 측정을 위해 사용하였다. 카메라를 움직이기 위해서는 이 메모리 헤드의 손잡이를 좌우 위아래로 그리고 Zoom/Focus 스위치를 작동시켜 메모리 헤드를 움직임으로써 그 위에 장착된 카메라를 움직인다. 이러한 메모리 헤드의 움직임은 실제적으로는 손잡이에 의해 움직여 지는 것이 아니고 그 손잡이에 가해지는 압력을 감지하여 그 입력에 의해 모터를 구동, 움직이게 된다. 따라서 손잡이에 압력을 가한 시점과 그 입력에 대응하는 압력이 감지된 후 모터 구동 시까지의 지연 시간을 조절할 수 있다. 이것은 가상 카메라에 포착된 컴퓨터 그래픽 배경이 그려지는 시간과 실제 카메라에 의해 포착된 실사 영상이 포착되는 시간의 차이를 보상 하는데 사용될 수 있다.

본 연구에서는 메모리 헤드로부터 위치 정보를 그래픽 서버인 ONYX와 RS232로 연결하여 전송하였고 이 데이터는 ONYX에서 Pan 및 Tilt의 각도 그리고 Zoom에 대한 Field-Of-View 값으로 변환시켜 그래픽의 Viewpoint 및 Field-Of-View 값으로 전송, 그래픽 배경의 적절한 움직임을 디스플레이 하였다. 다음은 카메라의 위치 정보를 측정하고 전송하는 Ultimatte Memory Head와 시리얼 통신을 하기 위하여 시리얼 포트를 열고 명령을 Write하거나 측정값을 Read하기 위한 부분이다.

카메라 인터페이스 프로세스는 Ultimatte Memory Head로부터 측정값을 수신한 다음 시뮬레이션 Loop의 Application 프로세스에 전달하기 위해서 프로세스간의 통신을 위한 IPC(Inter-Process Communication)의 Messages를 사용하였다.

실제적인 운영 결과, 실사 이미지에 대하여 가상 배경이 자연스럽게 연동이 되기 위해서는 그래픽 디스플레이의 Frame Rate가 30 - 60 Hz가

되어야 하는 것으로 나타났다. 따라서 카메라 움직임 데이터의 전송 및 데이터의 전처리 과정에서 지연되는 시간을 보상하기 위해 카메라를 움직이기 위한 모터 구동 신호에 약 3 Frame (1 Frame : 30 Hz) 정도의 지연을 주었다.

4. 3 차원 그래픽 배경 이미지 생성 과정

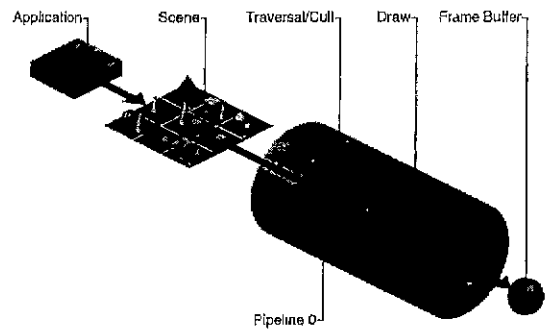
Virtual Studio에서 BG 이미지를 생성하는 기능은 가상 현실 용 그래픽 워크스테이션에서 실시간 비주얼 시뮬레이션 소프트웨어에서 수행한다. 가상 환경을 비주얼 DB 에 장면 그래프로 표현하고 그래픽으로 Dynamic한 화면을 표시하고 관련 주변 장치와의 인터페이스를 담당하는 것으로 실질적으로 가상 스튜디오 시스템을 운용하는 실행 부분이라고 할 수 있다. 비주얼 DB에서 지정된 노드의 모든 기능도 실질적으로 실시간 시뮬레이션 부분에서 처리된다고 할 수 있다.

실시간 시뮬레이션 부분은 사용자 처리, 그래픽 연산, 비주얼 데이터베이스 운용, 가상 배경 및 개체의 변경 및 애니메이션, 기타 각 노드의 정의된 기능 수행, 카메라 인터페이스 및 움직임 데이터 추출 작업을 동시에 수행하면서 실시간 이득을 최적화 시켜야 한다. 실시간 시뮬레이션 기능을 수행키 위해서는 작업을 나누어 분류하고, 이 기능들은 각각 다른 프로세스로서 실행되도록 하여, 각 프로세스를 다른 CPU 에 할당하여 다중 처리하여야 한다. 사용자 처리는 전체적인 제어, 각 Frame 의 Timing 제어, 각 노드들의 기능 구현, 기타 사용자 측면에서 고려하는 기능들이라고 할 수 있고, 이것은 APP 라고 정의한다.

카메라 위치 정보의 수신에 의해 가상 카메라의 Pan, Tilt, Zoom 값이 결정되면 이 값은 그래픽 View Point 에 설정된다. 이렇게 설정된 값에 따라 그래픽 컴퓨터는 해당 장면을 Rendering 하

게 된다. 본 연구에서는 기본적인 그래픽 기능은 SGI 사의 IRIS Performer 라는 그래픽 라이브러리를 사용하였다.

가상 카메라의 위치 정보에 따라 그래픽 장면을 변경 시키기 위해서는 우선 IRIS Performer 의 Rendering Pipeline 이 설정 되어야 한다. Rendering Pipeline 은 그래픽 영상 생성을 위한 전반적인 기능 수행을 나타내기 위해 정의되었다. Rendering Pipeline 은 화면상에 하나의 Geometry Pipeline 을 사용할 수 있는 윈도우를 그리고 사용자가 지정한 Scene 을 Rendering 에 적합하도록 처리한 후 Geometry Pipeline 을 구동 시켜 윈도우에 Rendering 결과를 표시하도록 한다. Geometry Pipeline 은 그래픽 컴퓨터 하드웨어 상에 존재하는 3차원 그래픽 처리 부분이다.



(그림 2) Rendering Pipeline

그래픽 채널은 Rendering Pipeline 에 표시되어야 할 장면의 여러 가지 조건들을 정의하고 있는 Rendering Viewport 로써, Eyepoint, View Direction, Field-Of-View, 그리고 Near/Far Clipping Planes 등과 같은 기본적인 시각 조건을 지정하고 구현하기 위해서 정의되었다.

따라서 본 연구에서는 카메라의 위치 정보를 그래픽 채널에서 설정하였다. 가상 카메라의 Pan 값은 그래픽 채널의 View Point, Heading 에 대입

하여 설정하였다. Tilt 값은 그래픽 채널의 View Point, Pitch에 대입하여 설정하였다. 그리고 Zoom 값은 그래픽 채널의 FOV 에 대입하여 설정 하였다.

상기와 같이 카메라의 위치 변화 정보를 시물레이션 Loop 내에서 채널의 Viewpoint 와 FOV를 설정 하므로써 원하는 위치에서 보여지는 그래픽 배경을 얻을 수 있다. 이렇게 생성된 그래픽 배경은 비디오 신호로 변환 된 후 영상 합성 장치의 Background 입력으로 연결된다.

그래픽 생성은 APP 에서 지정된 각종 그래픽 효과를 CULL 에서 선별된 그래픽 개체에 대응하는 그래픽 장면을 생성해내는 것으로써 기본적인 Rendering 및 Texture Mapping 기능을 직접 수행하고 이것은 직접적으로 그래픽 하드웨어를 제어하는 부분으로써 DRAW 라고 정의한다.

부가적으로 또 다른 하나의 프로세스를 카메라 움직임의 측정 및 처리 부분에 할당한다. 카메라의 움직임은 상기의 Viewing Frustum 과 연관되는 것으로, 카메라의 Pan, Tilt는 Eyepoint(View Point)의 방향에 해당되고 Zoom은 Field-Of-View (FOV)에 해당된다. 이렇게 구분된 프로세스들은 각각 다른 CPU에 할당될 수 있다. 각 프로세스들 중 대부분 처리 시간이 가장 많이 소요되는 것은 DRAW 부분이다. DRAW는 내부적으로 그래픽 하드웨어를 구동 시키는데, 하드웨어적으로 Rendering과 Texture Mapping을 위해 많은 연산을 수행하기 때문에 이 부분이 병목 현상을 일으키게 된다.

따라서 DRAW 를 하나의 CPU 에 할당하여 처리하고 다른 프로세스를 다른 부분에서 처리하게 된다면 최적의 속도를 확보하게 된다. 하지만 DRAW 자체가 과도한 연산을 수행하도록 다량의 그래픽 처리를 할 경우는 분산 처리를 통해서도 원하는 효과를 얻을 수는 없다. 이 경우는 더욱 많은 그래픽 처리 기능을 보유한 하드웨어를 사

용하거나 APP 또는 CULL 에서 선택적으로 처리하여야 할 연산량을 줄이도록 하여야 한다.

그래픽 처리 과정에서 화면의 단위는 다면체 삼각형이다. 즉, 물체의 표면을 삼각형으로 모델링하는 것이 일반적인 기법인데 컴퓨터 마다 실시간으로 처리할 수 있는 삼각형 수는 한계가 있다. 방송 프로그램의 경우 1초에 60개의 화면을 처리하여야 소위 방송 질 (Broadcast Quality)의 영상이 창출되는데 이 분야에서 가장 빠른 컴퓨터를 쓴다고 해도 5, 000 개에서 10, 000개 정도의 삼각형을 실시간으로 처리할 수 있는 정도이다. 그러나 실세계를 사진처럼 정교하게 삼각형으로 표시하기 위해서는 약 8, 000 만개의 삼각형이 필요하다. 따라서 현재 Virtual Studio 기술을 적용하는 분야는 Virtual Set가 그리 정교하지 않아도 방송이 가능한 분야에 적용하는 것이 선택과 인공적인 느낌을 전체 방송 프로그램에 조화 시키는 연출력이 필요하다.

5. 가상 세트 구축

본 연구에서는 가상 공간 저작에 필요한 3D 그래픽을 효과적으로 저작하고 종합적 정보를 기술하기 위해서 비주얼DB를 구성하였다. 그리고 그 내부는 가상 공간의 저작 및 운영을 위한 상태 정보와 기하학적 형태 정보를 담고 있는 것으로 그 전체를 scene이라고 정의하였다. 이 scene은 계층적인 그래프 구조로 노드(node)라는 단위 객체로 구성되어 비주얼DB를 구성하고 있다. 이러한 계층 구조의 연결관계는 가상 공간에서 표현하고자 하는 객체의 논리적 또는 공간적 구성에 관한 정보를 나타내게 되고, 상부에서 하부로 그리고 좌측에서 우측으로 탐색 한다. 또한 이들 자료 구조의 구성을 위해서는 객체 지향 프로그래밍 인터페이스가 제공되고, 비슷한 객체에 대해서

는 범용적으로 사용될 수 있도록 **template** 객체를 이용한다.

비주얼DB의 사용자를 통하여 실제 가상 공간을 구성하는 객체에 대한 속성을 정의할 수 있는 **Parameter Inspector**와 전체 scene의 구성을 알아볼 수 있는 **Scene Inspector**를 기본으로 한다. Studio를 저작할 것인지 아니면 가상 환경에 무관하게 움직이는 **Queue Sheet**를 저작할 것인지, 그리고 여러 개의 **Queue Sheet**가 존재하는 경우는 순서를 정의할 수 있다. 또한 **Frame**별로 Scene을 관찰할 수 있는 사용자 인터페이스가 있다.

비주얼 DB는 크게 두 가지로 나누어진다고 설명하였다. 가상 공간에 대한 정의를 하는 것과 그 속의 객체와 그의 **Frame**에 따른 동적인 속성을 부여하는 **QueueSheet**를 가지고 있다. 이렇게 구성함으로써 얻는 잇점은 가상 공간의 재사용성과 독립된 이벤트들을 나누어 제작할 수 있으므로 가상 공간 저작을 효율적으로 구성하도록 하기 위해서 이다.

(1) 가상 스튜디오

가상 스튜디오는 가상공간의 환경을 정의하는 것이다. 정적인 객체들뿐만 정의하는 것이 아니라 동적인 객체도 함께 정의가 가능하므로 효과적인 가상 공간의 구성이 가능하다.

(2) 큐 시트(Queue Sheet)

가상 공간의 환경과 무관하게 디스플레이 될 수 있는 객체들을 정의한다. 이렇게 함으로써 가상 공간의 환경 즉 배경에 대한 제작과 그 속에서의 객체들을 나누어서 개발을 할 수도 있다. 또한 하나의 가상 스튜디오 안에 여러 개의 큐 시트를 제작할 수 있게 함으로써 주제별로 가상 공간의 운용을 도모한다.

기본적으로 기하학적 형태와 그들의 움직임과 상호작용의 효율적인 판리는 비주얼DB를 구성하는 단위인 노드로 정의하여 필요한 노드의 형태

를 정의하고 그 노드를 추가시킴으로써 원하는 기능이나 동작을 만들어 내었다. 노드 단위의 비주얼 DB구성으로 대부분의 원하는 동작을 만들어 낼 수 있음도 검증하였다. 효율적인 DB구성, 처리하기 이전에 디스플레이 시킬 필요가 없는 객체들을 제거할 수 있도록 DB를 구성한 것은 **Culling**기능을 쉽게 적용하고 정의된 기능을 **Traversal**함으로써 **Overhead**없이 가상공간 **Rendering**을 효과적으로 수행하여 속도 이득을 얻을 수 있었다.

본 연구에서 그래픽 가상 배경을 만들기 위해서는 **AutoCAD**사의 **3D Studio**를 사용하여 배경을 모델링 하였다. 이렇게 모델링된 배경을 단순히 비주얼 시뮬레이션 모듈에서 특정 **Light**를 지정하여 밝기를 조절 하면 그 실제성이 떨어진다. 왜냐하면 그래픽 하드웨어 및 그래픽 라이브러리에서 지원하는 **Light** 효과는 단순히 기하학적 형태의 **Vertex Normal**에 대한 **Light**의 상대적 방향에 따른 **Shading** 처리이기 때문이다.

실세계에서의 광원은 해당 물체와의 직접적인 관계 외에 물체에 반사되어 나타나는 간접광 효과가 있고 또한 실내의 경우는 많은 개수의 광원이 존재하기 때문에 그 과정이 매우 복잡하다. 따라서 실세계에서와 같은 효과를 내기 위해서는 일반적인 **Shading** 기법 외에 **Radiosity** 기법을 사용하여야 한다. 하지만 **Radiosity** 기법을 적용할 경우 실시간 그래픽 생성이 불가능 하다.

본 연구에서는 그 해결책을 위해 **Radiosity** 처리를 모델링 단계에서 미리 수행하여 그 결과로 나타난 정보를 그래픽 객체의 **Vertex Color**에 지정하여 조명에 의한 색깔 변화를 지정하여 두었다가 실제 가상 스튜디오의 그래픽 배경을 생성해내는 비주얼 시뮬레이션 단계에서는 그 **Vertex Color** 값만을 읽어 들여 랜더링 하는 방법을 채택 하므로써 문제점들을 해결하였다. **Radiosity**

처리를 위해서는 Lightscape Visualization System을 사용하였다.

6. 맺는말

물리적인 세트를 준비하기 위해서는 세트를 디자인하고 목공소에서 제작하고 그 세트를 방송국 스튜디오에 건설하는데 많은 시간, 비용, 인력을 소모하게 된다. 일단 세트에서 촬영이 끝나면 스튜디오에 건축한 세트는 다음 방송까지 분해하여 창고에 저장하게 된다. 다음 방송 때가 되면 다시 창고에서 옮겨 스튜디오에 건설하게 되고 정규 프로그램의 경우 이러한 분해와 건축은 계속 반복된다. 또한, 물리적인 세트의 경우 그 세트 규모가 스튜디오 공간에 따라 제약을 받게 된다. 유선 방송국과 같이 영세한 방송국에서는 스튜디오 공간 확보에 많은 비용이 소요되고 있다.

컴퓨터에서 디자인하고 컴퓨터가 창출한 가상 세트에서 연기자나 사회자를 촬영하는 가상 스튜디오는 공간에 큰 제약을 안 받아도 된다. 가상 스튜디오는 물리적인 세트에 비해 가상 세트를 컴퓨터 디스크 메모리에 저장하면 되고 연출자는 방송 때마다 세트 디자인을 마음대로 언제나 컴퓨터 디자인 툴을 사용하면 바꿀 수 있다. 따라서 가상 스튜디오 기술의 발전은 방송 매체의 일대 혁신을 가져올 첨단 기술분야로 각광을 받고 있다.

앞으로 컴퓨터의 그래픽 성능은 계속 향상할 것이고 따라서 가상 현실 기술의 발전도 가속화할 것으로 예상하며 가상 스튜디오 기술이 원숙한 형태로 보편화 될 날도 그리 멀지않을 것이다.

현재로서는 방송 분야에는 아날로그 방송 장비에서 디지털로, HDTV의 미래, 비디오 압축, 디지털 송수신 시스템으로, 공중파 방송에서 유선 방송으로, 방송망과 인터넷의 통합화 등 엄청난 변신을 요구하고 있다. 앞으로 기존의 스튜디오, 프로덕션 과정은 엄청난 변화를 경험하여야 할 것이고 Virtual Studio기술은 그 변화의 중요한 일부로서 확산될 것이다.



고 희 동

- 1982년 State University of New York at Albany (학사)
- 1989년 University of Illinois at Urbana-Champaign (박사)
- 1988년-1990년 George Mason University 객원교수

- 1990년-1997년 한국과학기술연구원 선임연구원
- 1997년-현재 한국과학기술연구원 책임연구원