

원격 다중 참여자 3차원 그래픽스 게임 개발

Development of Multi User Games using 3D Graphics and Multicasting

김종성*, 박지현**

*JongSeong Kim, **JiHun Park

*아시아나항공(주) 금호서울시스템팀, **홍익대학교 정보대학원 전자계산학과

요 약

본 논문에서는 원격 다중 참여자 3차원 그래픽스 게임 개발에 대하여 연구 하였다. 3차원 컴퓨터 그래픽스를 이용하여 표현된 가상의 공간에서, 컴퓨터 통신을 사용하여 원격지에 떨어져 있는 여러 사용자가 서로 상호작용 하면서, 여러 사용자가 서로 3차원 그래픽스로 표현된 물체를 공유할 수 있는 게임 시스템 구축의 연구에 목적을 두었다.

3차원 그래픽스 물체를 표현하기 위하여 OpenInventor와 C++를 사용하였다. 통신을 위하여 멀티캐스트 통신 프로토콜을 사용하였으며, 입력 장치로서 3차원 위치 방향 인식/입력기인 Polhemus Fastrak을 사용하였다. 한가지 예로, 분산된 네트워크에서 여러 사용자가 함께 피아노를 연주하는 효과를 얻을 수 있는 게임을 구현하여 제시한다.

Abstract

In this paper we describe a development of a multi user game using 3D graphics and multicasting. First, we represent a virtual reality world by 3D graphics technology. Next, we use computer communication technology for the purpose of interaction among multi users. As a result, multi users share 3D graphics objects and play games. The goal of this paper is a development of a system that enables multi users to play games sharing a virtual world represented by 3D graphics.

We use the OpenInventor library and C++ language for 3D graphics scene representation, and IP multicasting protocols for computer communication. The system allows to use Polhemus Fastrak 3D input device that recognizes the 3D position and 3D orientation. As an example we present a game that demonstrates the effect of piano concert by multi piano players who are distributed in a computer network.

1. 서론

컴퓨터 그래픽스는 눈부신 발전을 하고 있다. 3차원 컴퓨터 그래픽스는 공간의 복잡한 물체의 동작을 현실감있게 표현하고 있다. 인터넷과 같은 컴퓨터 통신망은 지구상의 거의 모든 시스템을 연결해 놓았다. 3차원 그래픽스 물체의 움직임을 실제 물체와 같이 부드럽고 자연

스럽게 표현하기 위한 여러 연구들이 진행되고 있다. 또한, 컴퓨터 그래픽스 알고리즘을 이용하여 공간상의 빛의 효과와 물체의 움직임을 현실감있게 보여주는 그래픽스 라이브러리들도 많이 개발되었다. 사용자가 시스템과 서로 상호작용 하면서 가상의 세계를 체험하도록 구축된 시스템의 예를 많이 접할 수 있다. 가상현실 시스템이란 3차원 컴퓨터 그래픽스를 통하여 표

현된 가상의 현실세계를 사용자가 체험하는 것 같은 느낌을 유도하는 시스템이라 할 수 있다.

본 논문에서는 여러 사용자들이 분산된 네트워크에서 3차원 그래픽스로 표현된 물체들을 공유하면서, 사용자들 상호간에 자신의 그래픽스 물체에 대한 움직임까지도 주고 받을 수 있는 게임 개발에 대하여 연구 하였다.

본 연구에서는 한가지 사례로, 분산된 네트워크에서 여러 사용자가 함께 피아노 연주를 하는 효과를 얻을 수 있는 피아노 연주 게임 시스템을 구현하여 제시한다. 구현 기술로서는 오픈인벤터 그래픽스 라이브러리와 멀티캐스트 통신 프로토콜을 사용하였으며, Polhemus Fastrak 3차원 위치 방향 인식/입력기를 사용할 수 있도록 하였다. 구현된 피아노 연주 게임 시스템에서는 피아노 연주 게임을 하는 여러 사용자가 공통의 움직임 장면을 볼 수 있게 하여, 결과적으로 함께 피아노를 연주하는 효과를 얻을 수 있는 원격 다중 참여자 3차원 그래픽스 게임이 되도록 하였다.

2. 관련연구

2.1 가상환경 네트워크 구조

고려대학교 시스템연구실에서는 다중 사용자 가상환경을 위한 네트워크 구조에 대한 연구가 있었는데, 전체 데이터와 네트워크에 관련된 정보를 하나의 노드인 서버에서 관리하고, 사용자는 이 서버에 의해서 관리되는 그룹에 접속하여 3차원 가상 공유 공간에 접근하는 방식을 소개하였다. 이 방식에서는 한 번에 하나의 목적지에 전송하는 유니캐스팅 방식으로 시스템을 구축하였으며, 그룹 단위로의 유니캐스팅을 함으로써 소프트웨어적으로 멀티캐스팅을 제공할 수 있는 방법을 제안하였다.[16]

2.2 삼차원 가상환경 시스템

한국과학기술연구원 CAD/CAM 연구팀에서는, 초고속 통신망에서 가상환경과 상호작용을 공유하기 위한 연구로서, 상호작용이 가능한 3차원 가상세계표현은 멀티유저 환경을 지원하도록 확장한 VRML을 기반으로 하고, 네트워크 모델은 IP 멀티캐스트 환경하에서 멀티유저와 상호작용을 지원하면서, 3차원 가상 세계를 공유하도록 하는 연구가 있었다.[15]

2.3 가상 전시장 시스템

시스템공학연구소 시스템응용연구부에서는 전시 공간이 필요없는 가상 전시장 시스템 개발에 대한 연구가 있었는데, DataGlove등의 장치를 이용하여 가상현실에서 인간의 감각에 인공적인 자극을 제공함으로써 가상의 환경 안에 있다는 느낌을 유도하는 시도가 있었다.[14]

2.4 SGI의 Slotcar 게임

SGI(Silicon Graphics, Inc.)에서는 Slotcar 라는 게임을 개발하였는데, 3차원 그래픽스를 표현하기 위하여 오픈인벤터 라이브러리를 사용하였고, 분산된 네트워크에서 여러명의 사용자가 함께 자동차 경주를 할 수 있도록 멀티캐

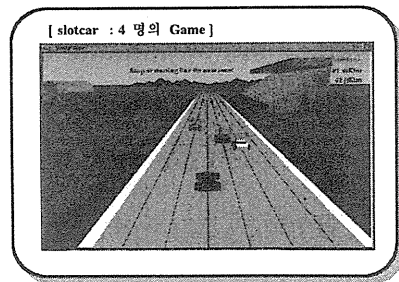


그림 1. SGI의 Slotcar 게임.

스트 통신 프로토콜을 사용하였다. [2]

Slotcar에서는 게임 구분번호를 사용자가 선택하여 게임을 실행하여, 특정한 그룹의 사용자들과 트랙 경주를 할 수 있게 하였다. 또 사용자들간에 통신을 하기 위하여, 패킷의 종류를 CarInfo, CarCrash, RaceStartInfo, CarCollision, CarJoin, CarExit, CarQuery, StartQuery, FinishQuery, CarFinish와 같이 클래스 객체로 설계하여, 같은 멀티캐스트 그룹에 속한 사용자간에 알릴 수 있게 하였다. [2]

2.5 SICS의 Dive 게임

SICS(Swedish Institute of Computer Science)에서는 Dive(Distributed Interactive Virtual Environment)를 개발하였는데, 분산 네트워크 환경에서 3차원 그래픽스를 표현하여 다중 사용자가 하나의 가상공간을 공유하도록 개발된 시스템이다.[1] 그래픽스를 표현하기 위하여 OpenGL등의 라이브러리를 사용하였고, 통신에는 IP 멀티캐스트를 사용하였는데, 분산환경의 시스템을 지원하기 위하여 MINI-Isis 툴킷을 사용하였다. [1]

아이시스(Isis) 툴킷은 프로세스 그룹들을 관리하거나, 그룹 기반의 소프트웨어를 개발할

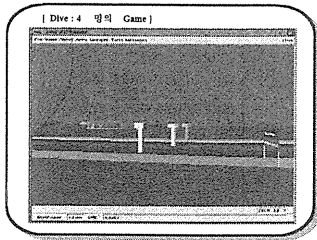


그림 2. SICS의 Dive 게임.

수 있는 상위레벨 기법의 여러 방법들을 제공해 주며[7], 가상 동기화 프로세스 그룹들과 그룹 통신을 위해 분산 프로그래밍 환경을 제공한다.[7]

Dive에서는 다중 사용자를 위하여 Diveserver를 두고 있다. 그리고 Vishnu라는 브라우저에 HMD(Head Mounted Display), Dataglove 등의 장치를 사용하여 3차원 그래픽스 물체의 선택이나 잡기를 할수있게 하였으며, VRML 파일(*.vr)을 새로이 읽어들이거나, 새로운 URL(Uniform Resource Location)을 선택하여 변경할 수 있게 하였다. 또 동일한 공간에 있는 사용자간에는 동일한 오디오 컨퍼런스를 공유하도록 하였다. [1]

2.6 멀티캐스트

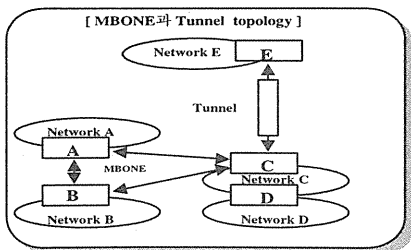


그림 3. MBONE 및 터널 연결 그림.

멀티캐스트(Multicast)는 TCP/IP 프로토콜군에 속하며, UDP 층(Layer)을 소켓으로 이용하고 있다. 멀티캐스트는 D클래스에 속하는 IP 주소 (224.0.0.0 - 239.255.255.255) 를 사용하며, 멀티캐스트 그룹에 속한 호스트들에게 패킷을 전달하는 통신 프로토콜이다.[11] 패킷은 TTL(TimeToLive)값 이내의 라우터를 통과하여 다른 네트워크에 전달될 수 있다.[8] [11]

서로 다른 네트워크에서 멀티캐스트 통신을 하기 위해서는 MBONE 을 이용하는 방법, 터널 프로그램을 이용하는 방법의 두 가지가 있을 수 있다. MBONE은 멀티캐스트 패킷을 다른 네트워크로 라우팅해주는 기능을 가진 라우터로 연결된 네트워크의 집합이다. MBONE에 연결되어 있지 않은 네트워크는, 네트워크와 네트워크 사이에 터널 프로그램을 설치하여 멀티캐스트 통신을 할 수 있다.[12]

2.7 멀티캐스트를 지원하는 운영체제

멀티캐스트를 커널에서 지원하는 운영체제에는 Solaris(2.1+), BSD/386(1.1+), DEC OSF/1(2.0+), IRIX(4.0+), AIX(4.1+) 등이 있다. IRIX 의 경우 v4.0이라도 커널 및 mrouted가 불안정하게 동작하므로 URL (ftp://sgi.com/sgi/ipmcast) 에서 프로그램을 가져와 패치 작업을 해야 한다. v5.x에서는 정상적으로 동작하며, 새 버전의 mrouted를 URL에서 구해 사용할 수도 있다.[8]

2.8 터널

터널(Tunnel)은, 멀티캐스트 패킷의 IP 헤더에 유니캐스트의 목적지 주소값을 갖는 IP 헤더를 덧붙여 패킷을 전송한 다음, 목적지 네트워크에서 덧붙여진 IP 헤더를 제거해주는 통신 프로토콜이다. 결과적으로 터널 시작 네트워크에서 유니캐스트 패킷으로 전송되어 터널 끝 네트워크에서 멀티캐스트 패킷이 된다. 네트워크의 라우터가 멀티캐스트 라우팅을 지원하지 않을 때 터널 프로그램을 설치하여 서로 다른 네트워크에서 멀티캐스트 통신을 할 수 있다. 그림3에서 라우터 E와 C사이에 연결된 라우터가 멀티캐스트 라우팅을 지원하지 않더라도, 멀티캐스트 터널을 설치하면, 네트워크 E와 C 사이에서도 서로 멀티캐스트 통신을 할 수 있게 된다.[12]

2.9 MBONE

IP 멀티캐스트 라우팅을 지원하는 라우터들로 연결된 인터넷 네트워크의 서브셋을 MBONE 이라고 한다. 그림3에서 라우터 A,B,C,D가 서로 멀티캐스트 라우팅을 지원해 줄 때, 네트워크 A,B,C,D는 서로 멀티캐스트 통신을 할 수 있으며, MBONE 이라고 할 수 있다. MBONE은 T1 속도 이상으로 연결한다.[8] [12]

2.10 전세계 MBONE 망

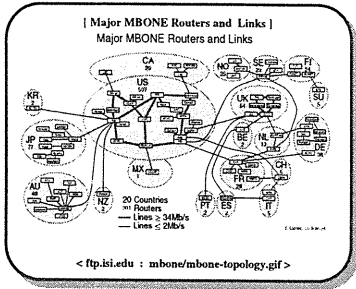


그림 4. 주요 MBONE 지도.

그림4는 전 세계에 연결된 MBONE 망을 그린 것이다.[13] 전세계 MBONE 망에 연결하기 위하여 국내 단일 네트워크에서는 URL(mail://mbone-korea-request@cosmos.kais.t.ac.kr)에 메일을 보내서 터널 연결을 요청해야 한다. [8] [9]

2.11 관련연구들과 본 연구의 비교

관련연구들을 통하여 알아본 바와 같이 컴퓨터 그래픽스를 이용하여 3차원 물체들을 실감있게 표현하고, 통신을 통하여 여러 사용자가 함께 참여하는, 원격 다중 참여자 가상현실 시스템이나 게임 등의 연구가 많이 진행되고 있다. 이러한 연구는 3차원 그래픽스 물체를 쉽게 표현해주는 라이브러리의 개발, 분산 컴퓨팅에 대한 통신 프로토콜의 개발, 3차원 입력/출력 장치등의 개발 등에 대한 연구의 결과라 할 수 있다.

본 연구에서는 Slotcar게임과 같이 그래픽스 라이브러리로서 오픈인벤터를 사용하였다. 또한, 가상전시장시스템에서 데이터글러브를 사용했듯이 본 연구에서는 Polhemus Fastrak을 사용하였다. 또, Dive게임에서 Mini-Isis 통신 툴킷을 사용한것과 달리 삼차원 가상환경 시스템이나 Slotcar게임과 같이 멀티캐스트 통신을 사용하였으며, 서울 광주간 터널을 설치하여 분산된 네트워크에서 멀티캐스트 통신 게임이 가능함을 실제 보여 주었다.

3. 본론

3.1 연구의 목적 및 의의

본 연구에서는 원격 다중 참여자 3차원 그래픽스 게임을 실제 구현하여 효과적인 개발

사례를 제시해 보고, 이 구현 사례를 참고하여 향후 연구에서는 좀더 효과적인 시스템을 개발할 수 있도록 하는데 연구의 목적을 두었다.

즉, 3차원 그래픽스로 표현된 자신의 물체를 화면에 표시하고, 자신의 그래픽스 물체가 움직인 정보를 통신을 통하여 사용자들끼리 주고 받아, 모든 사람의 화면에 있는 그래픽스 물체들이 실시간으로 동시에 같은 움직임을 보여줄 수 있는 효과적인 게임을 구현해 보는데 연구의 의의를 두었다.

3.2 피아노 연주 게임의 개발

본 연구에서는 원격 다중 참여자 3차원 그래픽스

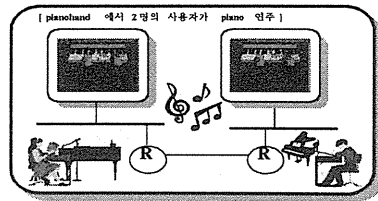


그림 5. 2명의 사용자가 피아노 연주 게임을 하는 장면.

게임의 한가지 사례로 피아노 연주 게임을 개발 하였다. 피아노 연주 게임은 여러명의 사용자가 서로 다른 네트워크에서도 함께 피아노 연주를 할 수 있는 게임이다. 피아노 연주 게임 시스템에서 여러 네트워크에서 게임을 하는 사용자수 만큼의 손이 화면에 나타나게 하여 다중 사용자를 표현하였다. 사용자의 추가시 모든 게임 사용자의 화면에는 추가된 사용자의 손(3차원 그래픽스 물체)이 나타나도록 하였다. 한 사람이 키보드를 치면 해당 음계의 피아노 소리를 내어 주고, 그래픽스 손이 피아노 건반을 누르는 움직임을 보여주도록 하였으며, 멀티캐스트 통신에의해 모든 사람의 화면에 공통의 움직임을 보여주도록 하였다. Polhemus Fastrak을 착용한 피아노 연주 지휘자의 팔의 움직임은 화면상에 관절로 표현된 3차원 그래픽스 지휘팔 물체가 움직여 모든 사용자들의 화면에 보여지도록 하였다. 결과적으로 사용자들은 함께 피아노를 연주하는 효과를 얻을 수 있으며, 원격 다중 참여자 3차원 그래픽스 게임 구현의 한 사례가 되도록 하였다.

3.3 시스템의 구성

시스템은 사용자와 호스트가 있으면 독립형(Stand Alone)으로 사용할 수 있다. 이때, 호스

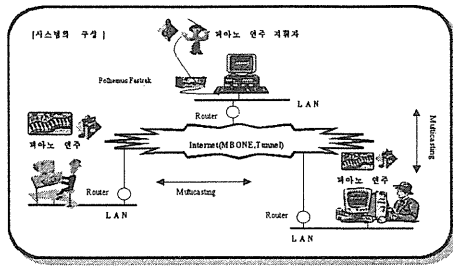


그림 6. 시스템의 구성.

트는 3차원 그래픽스를 표현해 주면서 멀티캐스트 통신을 지원해 주어야 한다. 물론 독립형에서 자신이 전송한 멀티캐스트 패킷은 루프백되어 자기 자신이 받아 처리한다.

호스트가 네트워크(LAN)에 연결되어 있을 때에는 같은 네트워크에 연결되어 있는 다른 사용자와 멀티캐스트 패킷을 주고 받으며 통신하면서, 함께 피아노 연주 게임을 즐길 수 있다. 추가로, Polhemus Fastrak을 피아노 연주 지휘자가 사용할 수 있도록 하였다. LAN이 MBONE 혹은 터널로 다른 네트워크에 연결되어 있으면, 서로 다른 네트워크의 여러 사용자와 피아노 연주 게임을 즐길 수 있도록 하였다.

그림5는 2명의 사용자가 피아노 연주 게임을 하는 장면이며, 그림6은 전세계에 연결되어 있는 MBONE 망에서 여러 사용자가 함께 피아노 연주 게임을 할 수 있음을 보인 것이다.

3.4 피아노 연주 게임 어플리케이션의 구조

피아노 연주 게임은 C++의 클래스를 이용하여 객체지향적으로 구축하였다. 피아노 연주 게임은 MouseInputCB(), packetCB()등의 주요 프로시저어로 구성되는데, MouseInputCB()에서 서버의 기능을 두었고, packetCB()에 클라이언트의 기능을 두었다.

오픈인벤터(OpenInventor)는 실리콘 그래픽스(Silicon Graphics)에서 개발한 컴퓨터 그래픽스 라이브러리인데, 3차원 컴퓨터 그래픽스를 쉽게 표현할 수 있으며, 본 논문의 시스템 구축에 사용하여 게임의 장면을 표현하였다. 즉, 피아노, 손, 팔, 카메라와 같은 3차원 그래픽스 물체들을 설계하였으며, C++ 클래스로 오픈인벤터 라이브러리를 캡슐화하여 구현하였다.

네트워크(Network), 손추가(HandJoin), 손동작(HandAct), 손제거(HandExit)등의 오브젝트

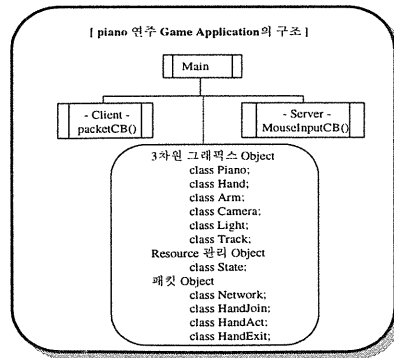


그림 7. 피아노 연주 게임 어플리케이션의 구조.

들을 설계하였으며, C++ 클래스로 멀티캐스트 통신 라이브러리를 캡슐화하여 구현하였다.

그림7은 피아노 연주 게임의 어플리케이션 구조와 오브젝트 종류를 보인 것이고, 그림15는 네트워크상에서 게임의 패킷 흐름과 수행 절차를 보인 것이다.

3.5 입력 장치의 설계

입력 장치는 사용자와 시스템간에 서로 상호작용을 도와주는 역할을 하게 된다. 특히 3차원 그래픽스 입출력 장치는 실제 공간에서의 물체의 움직임을 효과적으로 시스템에서 입력 받거나 보여줄 수 있어 사용자들에게 현실감을 높여주게 된다. 3차원 그래픽스를 위한 장치가 많이 개발되고 있다. 3차원 그래픽스 영상을 보여주는 안경, 3차원 공간의 위치 및 방향을 인식/입력해 주는 Polhemus Fastrak, 손가락의 움직임을 탐지해 주는 장갑(PowerGlove), 3방향 움직임을 입력해 주는 트랙볼 등의 3차원 그래픽스 입출력 장치들을 이용하면 3차원 그래픽스 게임을 하는 사용자들에게 더 재미와 흥미를 느낄수 있게 할 수 있다.

본 연구에서는 키보드, 마우스등을 이용하여 사용자의 입력을 받을 수 있도록 하였으며, 특히, Polhemus Fastrak도 사용할 수 있도록 하였다. 그림8은 키보드와 마우스의 역할을 표시한 것이고, 그림9는 Polhemus Fastrak을 사용했을 때 연주팔의 움직임을 표현한 장면이다.

3.5.1 키보드와 마우스

피아노 게임에서 키보드를 사용하여 피아노 건반 역할을 하도록 하였다. 키보드의 문자키를 이용하여 낮은 음의 소리에서 높은 음의 소리

까지 4옥타브의 음계를 소리낼 수 있도록 하였다.

키보드의 방향키 혹은 마우스로 카메라를 움직일 수 있도록 하였다. 카메라의 움직임에 따라 관점이 바뀌어 3차원 그래픽스 물체를 여

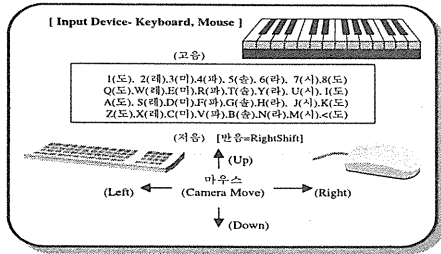


그림 8. 입력장치 - 키보드, 마우스.

러 각도에서 볼수 있게 된다.

사용자가 키보드로 피아노의 해당 음계를 누르면, 시스템에서는 전방향 기구학에 의하여 사용자 손과 손가락 관절을 변환시켜, 피아노의 해당 건반을 누르는 움직임을 보여주고 피아노 소리를 들려준다.

3.5.2 Polhemus Fastrak 입력기와 장치 드라이버

본 연구에서는 Polhemus Fastrak 3차원 위치 방향 인식/입력기를 사용하여 사람 팔의 움직임을 입력 받아서 게임에 사용하였다. 한 사람의 지휘자가 3차원 입력기를 팔에 차고 지휘

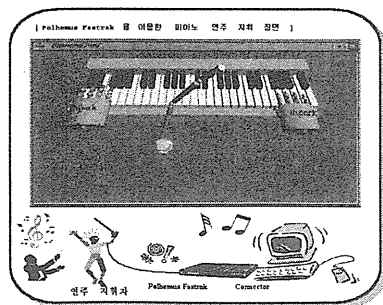


그림 9. Polhemus Fastrak을 이용한 피아노 연주 지휘 장면.

를 하면, 그 사람의 팔의 움직임이 화면에 나타나게 된다.

Polhemus Fastrak을 구동시키기 위한 디바이스 드라이버는 시리얼(Serial) 통신 기법을 사용하는데, 먼저 Fastrak을 초기화 시키고, 하

나의 읽기 명령을 보내면 일정한 지연 시간후에 Fastrak은 값을 되돌려 준다. 이 연구에서 Fastrak에서 받는 정보는 3차원 공간상의 위치와 쿼터니언(Quaternion)으로 표현된 방향값이다. 이 값을 받아서 사람의 움직임을 재생한다.

사람의 움직임을 측정하기 위하여 Fastrak 센서는 4개를 사용하였는데, 센서들을 어깨, 팔꿈치, 손목, 중지 끝(지휘봉 끝)에 두었다. 여기에서 위치 정보만을 고려하여 상대적인 좌표계 위치만을 계산하였다. 이 과정에서 사람의 팔의 각 관절의 자유도는 3으로 간주하였다. 팔의 길이는 고정되어 있으므로, 필요한 자유도는 2이므로, 회전축에 제약을 두어 입력받은 값으로부터 필요한 방향 벡터를 계산하였다. 이러한 사람의 움직임을 오픈인벤터의 타이머센서(TimerSensor)를 사용하여 일정한 시간 간격으로 읽어들인다. 입력값은 위에서 언급한 제약을 가진 쿼터니언 값으로 변환되어 이것을 멀티캐스팅 통신을 위한 패킷에 실어 보내며, 패킷을 받는 클라이언트들에서는 사람 움직임을 표현한 값들을 읽어들이어 화면에 수정 한다.

3.6 손가락 관절의 설계

피아노를 연주하는 손의 모든 손가락 관절은 자유도가 1이며, 전방향 기구학에 의하여 쉽게 해결된다. 또한 피아노 연주 지휘자의 팔의 관절은 자유도가 3이다. 그림10에서와 같이 X를 축으로 점A에서 물체를 회전하면(R1), 관절의 움직임과 같아진다. 피아노 건반을 누르는 손가락의 모양 변화는 손가락 마디를 축으로 회전한 정도에 따라 손가락이 굽어지게 되는데, 본 논문의 시스템에서는 각 손가락 관절을 구로 표현하여 회전축으로 사용하였으며, 기구학

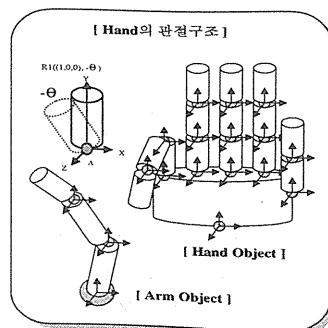


그림 10. 손과 팔의 관절구조.

에 따라서 1 자유도인 구 관절이 회전한 만큼 손가락 마디가 굽어지도록 하였다.

그림10은 손과 팔의 관절구조를 보여준 것이고, 그림13은 손의 장면그래프로서 손을 구성하는 손바닥, 손가락, 마디 등의 관련성을 표시한 것이다.

3.7 장면 그래프와 그래픽스 가상공간

그림12의 장면 그래프를 표현하면 그림11의 화면과 같은 장면을 볼 수 있게 된다.

모든 개개의 사용자는 자신의 화면에 표현되어 있는 모든 3차원 그래픽스 물체에 대한 자원을 자신의 지역 호스트에서 전부 가지고 있다. 즉, 자신의 화면에 표시되어 있는 피아노, 팔, 자신의 손, 다른 사용자의 손 등의 모든 물체에 대한 자원을 자신의 지역 호스트에 가지고 있고, 다른 사용자들도 또한 자신들의 지역

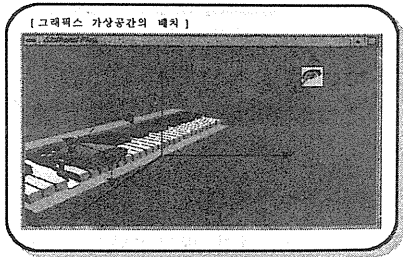


그림 11. 그래픽스 가상공간의 배치.

호스트에 모두 똑같이 가지고 있다. 그런데, 3차원 그래픽스 물체를 설계할 때, 물체의 움직임에 대한 정보(그래픽스 물체의 소유자 IP 주소, 눌러진 피아노 건반의 번호, 손가락 관절의 구부러진 각도, 팔 관절의 구부러진 각도 등)를 오브젝트 클래스의 멤버변수로 두었다. 그리하여, 그래픽스 물체가 움직이면 움직임에 대한 정보값을 다른 사용자들도 동일하게 수정할 수 있도록 멀티캐스트 통신을 이용하여 다른 사용자들에게 전송해 준다. 모든 사용자들은 또한 수정된 움직임에 대한 정보값에 의하여 자신의 화면에 표현되어 있는 그래픽스 물체를 다시 표현하게 되고, 모든 사용자가 똑같은 장면의 화면을 유지할 수 있게 된다. 즉, 이렇게 하여, 모든 사용자는 그림11과 같은 장면의 동일한 그래픽스 가상공간을 바라보고 있는 것 같은 느낌을 가지게 된다.

장면 그래프는 3차원 그래픽스로 표현된 물체의 구성을 나무 형태로 나타낸 것이다. 장면 그래프를 구성하는 물체들이 화면에 보여진다고 생각하면 쉽다. 3차원 그래픽스에서는, 작은

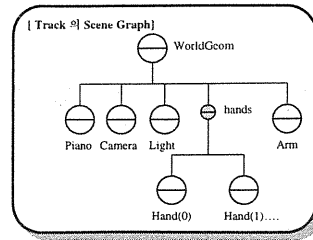


그림 12. 트랙의 장면 그래프.

단위의 3차원 그래픽스 물체가 모여 새로운 물체를 구성한다. 새로운 물체는 부모가 되고, 작은 단위의 물체들은 자식이 되는 나무를 구성하면, 뿌리 물체의 장면 그래프가 완성된다.

그래픽스 공간은 하나의 3차원 세계 좌표계(World Coordinator)로 표현되며, 뿌리로 시작되는 장면 그래프로 생각할 수 있다. 피아노 연주 게임에서는 세계 좌표계의 역할을 하는 트랙 오브젝트를 설계하였다. 그림12와 같이 트랙에 피아노, 카메라, 손, 팔 등의 3차원 그래픽스 물체를 추가 하였다. 손은 사용자의 수에 따라, 변동하는 사용자수 만큼의 손이 추가 및 제거될 수 있다. 그림12의 Hand(0), Hand(1)등과 같은 배열은 변동하는 사용자들의 손을 표현한다. 카메라의 위치는 응시점을 결정하게 되는데, 세계좌표계의 X,Y,Z축 방향으로 자유롭게 이동하

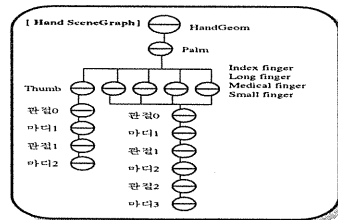


그림 13. 손의 장면 그래프.

는 변환을 수행하여, 화면상의 그래픽스 공간에 표현된 물체들을 여러 각도에서 볼 수 있도록 하였다.

장면 그래프에서 부모 물체의 변환은 자식 물체의 변환에 영향을 준다. 즉, 부모 물체가 변환을 하면, 자식은 부모의 구성체로서 같이 변환을 하게 된다. 즉, 그림13의 손의 장면 그래프에서 관절0이 회전하면 마디1, 관절1, 마디2, 관절2, 마디3등이 한 개의 구성체처럼 회전하게 된다.

3.8 클라이언트와 서버의 설계

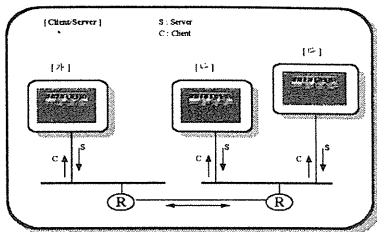


그림 14. 클라이언트와 서버.

개개의 피아노 연주 게임 사용자는 클라이언트의 역할과 서버의 역할을 모두 한다. 멀티캐스트 통신 프로토콜의 특성상, 패킷은 같은 멀티캐스트 그룹에 속한 호스트들에게 모두 전달되므로, 서버를 따로 두지 않고, 클라이언트와 서버를 하나의 피아노 연주 게임에 통합하여 구현하였다.

즉, 개개의 사용자는 클라이언트의 역할을 하면서, 자신의 그래픽스 물체에 대한 정보 전달을 하기 위하여 다른 서버에 의존하지 않고, 자신이 서버의 기능을 할 수 있기 때문에, 스스로 다른 모든 사용자들에게 자신의 그래픽스 물체에 대한 정보 전달을 한다. 개개의 피아노 연주 게임은 MouseInputCB에서 서버의 기능을 하며, packetCB에서 클라이언트의 기능을 한다.

3.9 다중 사용자의 그래픽스 물체 처리.

사용자의 입력은 그림15에서 화살표의 흐름과 같이 시스템에 전달되어, 네트워크에 손추가 (HandJoin), 손동작(HandAct), 손제거 (HandExit)와 같은 패킷의 형태로 전송되며, 멀티캐스트 통신 프로토콜에 의하여 모든 피아노 게임 사용자들에게 전달되며, 모든 사용자들은 공통의 화면을 볼 수 있게 된다. 결과적으로 분산된 네트워크의 원격지에 떨어져 있는 모든 피아노 연주 게임 사용자들은 자신의 화면에 표현된 3차원 그래픽스 오브젝트의 움직임 장면을 동시에 똑같이 표시하고, 실시간으로 서로 똑같은 장면의 화면을 바라보며 피아노 연주 게임을 하게 된다.

네트워크 오브젝트가 이런 패킷 오브젝트를 네트워크를 통하여 모든 사용자에게 전송해 주는 역할을 하게 된다. 이렇게 설계된 패킷 오브젝트를 사용하므로써, 자신의 그래픽스 물체에 대한 변경된 정보를 다른 사용자에게 알리고, 다른 사용자로부터 변경된 그래픽스 물체 정보를 전달받아 자신의 화면에 표시하는 방법을

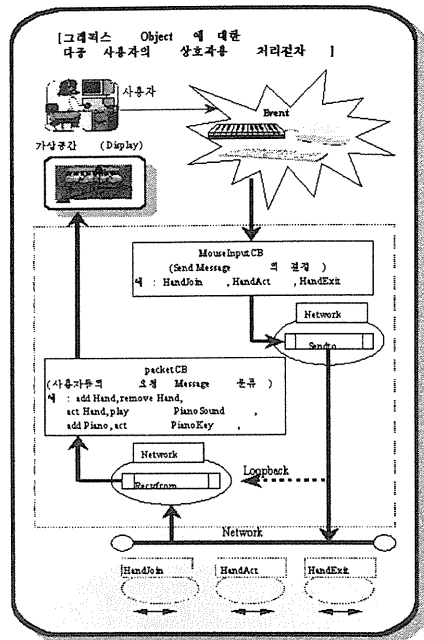


그림 15. 패킷흐름과 수행절차.

효과적으로 할 수 있었다.

그림16은 시스템의 메모리에서 그래픽스 물체 자원을 관리하는 상태를 표현한 것이며, 그림17은 시스템과 네트워크사이의 패킷의 경로를 표현한 것이다. 그림18은 시스템의 메모리에서 분산된 네트워크의 게임 사용자 그래픽스 물체 자원을 관리하는 방식을 표현한 것이다. 그림19는 새로운 게임 사용자가 추가되었을 때 패킷의 처리 방식을 표현하였으며, 그림20은 그래픽스 물체의 움직임을 다른 게임 사용자에게 알리기 위한 패킷의 처리 방식을 표현한 것이며, 그림21은 사용자의 종료시 패킷의 처리 방식을 표현한 것이다. 이와 같은 패킷의 처리와 수행 절차에 의하여 원격 다중 참여자 3차원 그래픽스 게임을 구현하게 되었다.

3.9.1 자원의 종합 관리

상태(State) 오브젝트는 자원을 종합적으로 관리하는데 사용한다. 즉, 상태 오브젝트에는 멤버변수로 피아노, Hands, 팔과 같은 그래픽스 물체와 네트워크와 같은 통신 오브젝트를 가리키는 포인터를 두고 있다.

그래픽스 물체의 움직임이 있을 경우, 움직임에 해당하는 정보(눌려진 피아노의 건반 번호, 손가락 관절의 각도, 팔 관절의 각도 등)를

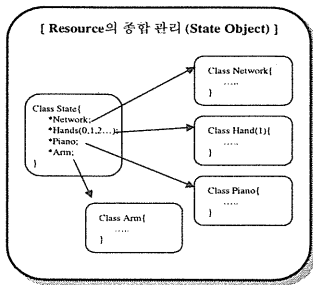


그림 16. 자원의 종합 관리 (상태 오브젝트).

물체의 멤버변수에 수정하여 보관하고, 필요시 그래픽스 물체의 멤버변수에 저장되어 있는 정보값을 탐색하는데 사용되어, 상태 오브젝트는 마치 데이터베이스(오브젝트의 멤버변수)의 자원(멤버변수의 값, = 그래픽스 물체의 움직임 정보)에 접근하는 경로(포인트)를 제공하는 역할을 한다.

3.9.2 멀티캐스트 소켓

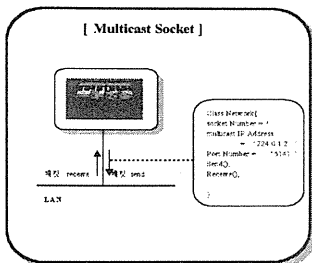


그림 17. 멀티캐스트 소켓.

게임의 시작시 네트워크 오브젝트를 생성하게 되며, 네트워크 오브젝트의 생성자(Constructor)에는 멀티캐스트 IP 주소와 포트번호로 멀티캐스트 소켓을 오픈하도록 하였다.

멀티캐스트 소켓은 패킷의 전송 및 수령에 사용된다. 분산된 네트워크상에서 사용자가 많아지더라도, 사용자가 전송한 하나의 패킷은 멀티캐스트 프로토콜의 특성상 같은 멀티캐스트 그룹에 속한 호스트들이 모두 함께 받게 되므로, 사용자수 만큼의 네트워크 패킷을 발생시키지 않고, 1개의 네트워크 패킷이 여러 사용자에게 전달되는 효과를 갖게 된다.

3.9.3 그래픽스 물체를 소유한 사용자의 구분

사용자의 IP 주소를 그래픽스 물체 클래스의 멤버변수로 두어, 분산된 네트워크에서 그래픽스 물체를 소유한 사용자를 판별하는데 사용한다. 또한, 그래픽스 물체 번호를 그래픽스 물

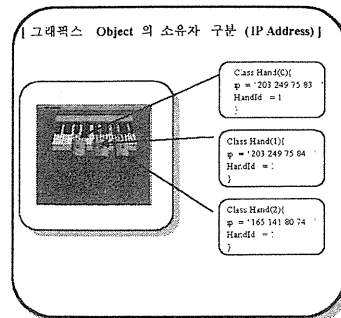


그림 18. 그래픽스 물체를 소유한 사용자의 구분(IP 주소).

체 클래스의 멤버변수로 두어, 한 사용자의 호스트에서 여러 피아노 연주 게임을 실행할 경우 어느 게임의 그래픽스 물체인지를 판별하는데 사용한다. 즉, 사용자의 IP 주소와 물체 번호의 2개 정보를 가지고, 그래픽스 물체가 어느 호스트의 몇번째 게임이 소유하고 있는지를 판별할 수 있게 된다.

예를들어, 다른 사용자의 손의 움직임이 있어서, 움직임 정보값을 수정했을 경우, 자신의 화면에 표현된 손의 IP 주소 멤버변수와 손소유자번호(HandId) 멤버변수를 검사하여, 움직임을 요청한 사용자의 게임이 소유하고 있는 손의 움직임을 처리한다.

3.9.4 사용자의 추가

사용자의 추가시, 추가된 사용자는 손추가(HandJoin) 패킷을 다른 사용자들에게 전송하여 자신의 출현을 알린다. 손추가 패킷에는 자신의 출현을 알릴 수 있는 정보(자신의 IP 주소, 손소유자번호 등)가 저장되어 있다.

손추가 패킷을 받은 사용자들은 추가된 사용자의 손을 자신의 화면에 추가하여 표시하고, 자신의 손에 대한 정보도 손추가 패킷에 저장하여 전송하여, 자신의 손도 추가된 사용자의 화면에 추가하여 표현할 수 있도록 한다.

3.9.5 사용자 물체의 움직임

사용자의 그래픽스 물체가 움직이면, 손동작(HandAct) 패킷을 다른 사용자들에게 전송하여 자신의 그래픽스 물체의 움직임을 알린다. 손동작 패킷에는 자신의 그래픽스 물체의 움직임을 알릴 수 있는 정보(눌려진 피아노 건반번호, 손물체의 굽혀진 손가락 관절의 각도, 팔 물체의 팔 관절의 각도 등)가 저장되어 있다. 손동작

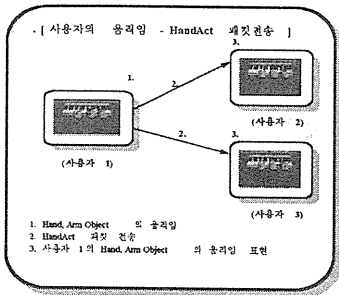


그림 20. 사용자 물체의 움직임 (HandAct 패킷 전송).

패킷을 받은 사용자들은, 손동작 패킷을 전송한 사용자의 그래픽스 물체의 움직임을 자신의 화면에서 다시 표현한다.

패킷의 전송주기는, 키보드나 마우스의 이벤트가 발생할 때 발생한 정보(피아노 연주)를 손동작 패킷 형태로 전송하고, 타이머센서에 의하여 일정시간 간격으로 Polhemus Fastrak에서 발생한 정보(피아노 연주 지휘)를 손동작 패킷 형태로 전송한다.

3.9.6 사용자의 종료

사용자의 종료시 손제거(HandExit) 패킷을 다른 사용자들에게 전송하여 자신의 종료를 알린다. 손제거 패킷을 받은 다른 사용자들은 종료된 사용자의 손을 자신의 화면에서 제거한다.

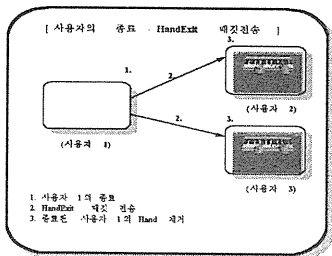


그림 21. 사용자의 종료 (HandExit 패킷 전송).

3.10 터널의 설치

IRIX-v.5.3에서는 URL(ftp://sgi.com/sgi/ipmcast)에서 mrouted를 구하여 터널을 설치할 수 있다. 본 논문에서는 홍익대학교 네트워크의 호스트(IRIX-v5.3)와 광

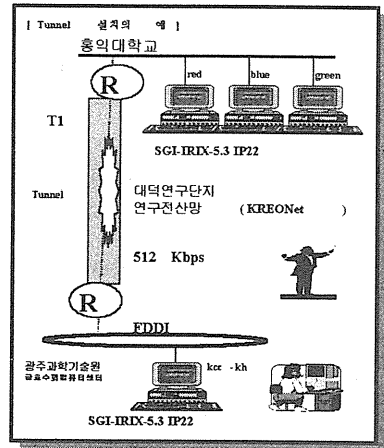


그림 22. 터널 설치의 예.

주과학기술원내에 위치한 금호수퍼컴퓨터센터 네트워크의 호스트(IRIX-v.5.3)에 mrouted를 설치하여 멀티캐스트 터널 통신을 하였다. 터널로 연결된 네트워크에서 피아노 연주 게임을 실시하여, 분산된 네트워크에서 즐기는 원격 다중 참여자 3차원 그래픽스 게임의 사례를 보여 주었다.

4. 결론 및 향후 연구 과제

본 연구에서 구현하여 제시한 피아노 연주 게임을 통하여, 원격 다중 참여자 3차원 그래픽스 게임 개발의 사례를 보여 주었다. 피아노 연주 게임 구현 과정을 통하여 다음과 같은 결론을 내릴 수 있었다.

3차원 그래픽스 라이브러리인 오픈인벤터와 C++ 라이브러리, 멀티캐스트 통신 라이브러리 등을 캡슐화하여 새로운 클래스 오브젝트를 설계함으로써, 손과 같은, 사용자를 구분해주는 새로운 3차원 그래픽스 물체를 정의하여 사용할 수 있었고, 원격 다중 참여자 3차원 그래픽스 게임을 효율적으로 구현할 수 있었다. 손에는 IP 주소를 클래스의 멤버변수로 두어, 사용자를 구분하는 정보로 하였다. 멀티캐스트 통신의 사용은 패킷 오브젝트 설계를 간단히 할 수 있게 하였다. 즉, 손추가, 손동작, 손제거와 같은 패킷 오브젝트를 간단히 설계하여 사용함으로써, 3차원 그래픽스 물체의 움직임 정보를 다른 사용자에게 효과적으로 알릴 수 있었다. 멀티캐스트 통신은 클라이언트/서버 모델의 설계

를 간단히 할 수 있게 하였다. 멀티캐스트 통신은 하나의 메시지가 멀티캐스트 그룹에 속한 모든 호스트들에게 전달되므로, 서버를 따로 두지 않고도, 개개의 게임이 클라이언트와 서버 기능을 모두 할 수 있도록 설계하는데 편리성을 제공해 주었다. 터널을 실제 설치하여, 분산된 네트워크에서 할 수 있는 원격 다중 참여자 게임의 사례를 보여 주었다. Polhemus Fastrak을 사용할 수 있게 하여 3차원 입력 장치 사용의 좋은 예를 보여 주었다.

본 연구에서는 '3차원 컴퓨터 그래픽스 물체를, 여러 사용자가 원격지에 떨어져 있는 네트워크상에서 어떻게 서로 상호작용하면서 공유하여 볼 수 있을까?' 하는 물음에 대한 해답을 얻기 위하여 연구를 진행하였었는데, 피아노 연주 게임의 구현은 그러한 물음에 만족할 수 있는 좋은 사례였다고 생각된다. 이 연구는 원격 다중 참여자 3차원 그래픽스 게임의 다른 구현에 참고 할 수 있는 좋은 예가 될 것이다.

이 연구에서, 그래픽스 물체의 표현을 현실 세계의 물체와 같이 자연스럽게 표현해 주는 노력은 부족하였다. 또한 시간이 흐름에 따라 그래픽스 물체(지휘자의 팔)의 변한 모양을 표현하기 위하여, Polhemus Fastrak과 함께 사용한 오픈인벤터의 타이머센서 영향으로, 패킷의 전송이 많아지게 되었다. 타이머센서의 영향으로 발생한 패킷의 전송횟수 증가는 본 연구의 한계점으로 드러났다.

현재 Polhemus Fastrak과 파워글러브등의 3차원 입력 장치를 이용한 시스템을 구축중이며, 향후 연구과제로서 타이머센서 사용으로 인한 패킷의 전송횟수 증가에 대한 보완과, 실시간 동작 인식 시스템을 계속하여 연구할 계획이다.

* 본 연구는 1997년 정보통신부 산학연 연구과제 개발 지원으로 이루어 졌습니다.

참고문헌

- [1] C. Carlsson, O.Hagsand, "DIVE - A Multi user Virtual Reality System", IEEE VRAIS, URL(<http://www.sics.se/dive>), 1993.
- [2] Dave Immel, Gavin Bell, "slotcar Game", URL (<http://www.sgi.com/Technology/Inventor/Games/gamesExecutables.html>), 1994.
- [3] Donald Hearn, M. Pauline Baker, "Computer Graphics" 2nd Ed. Prentice Hall, 1986.
- [4] Jackie Neider, Tom Davis, Mason Woo, "OpenGL Programming Guide", Addison-Wesley, 1993.
- [5] John J. Craig, "Introduction To Robotics" 2nd Ed. Addison-Wesley, 1986.
- [6] Josie Wernecke, "The Inventor Mentor", Addison Wesley, 1994.
- [7] Kenneth P. Birman, Robbert van Renesse, "Reliable Distributed Computing with the Isis Toolkit", IEEE Computer Society Press, 1994.
- [8] Steve Casner, "Frequently Asked question (FAQ) on the Multicast Backbone (MBONE)", URL(<ftp://venera.isi.edu/mbone/faq.txt>), 1994.
- [9] Steve Casner, "Major MBONE Routers and Links", URL(<ftp://ftp.isi.edu/mbone/mbone-topology.ps>), 1994.
- [10] Tom Swan, "Tom Swan's C++ Primer", Sams, 1993.
- [11] W.Richard Stevens, "TCP/IP Illustrated, Volumel", Addison Wesley, 1994.
- [12] W.Richard Stevens, "TCP/IP Illustrated, Volume2", Addison Wesley, 1994.
- [13] W.Richard Stevens, "Unix Network Programming", Prentice Hall, 1991.
- [14] 김기호, 김동현, "전시 공간이 필요없는 가상 전시장 시스템 개발", 컴퓨터그래픽스 학회논문지 제1권 제1호, 1995.
- [15] 김주한, 김래현, 고희동, 최병균, "다자참여형 3차원 가상환경 시스템 구현", 한국컴퓨터그래픽스학회 춘계학술대회 논문지, 1997.
- [16] 최활란, 이희웅, "다중 사용자 가상환경을 위한 네트워크 구조", 한국컴퓨터그래픽스학회 춘계학술대회 논문지, 1997.