

PC용 가상현실 입출력 인터페이스의 설계 및 구현

서 정 태, 정 문 렬

송실대학교 정보과학대학 멀티미디어 연구실

서울시 동작구 상도5동 1-1

seojt@indigo.soongsil.ac.kr, moon@computing.soongsil.ac.kr

Design and Implementation of Virtual Reality Input-Output Interface for PC

Jung-Tae Seo, Moon-Ryul Jung

Multimedia Lab. Collage of Information Science. Soongsil Univ.

1-1, SangDo 5Dong, DongJakKu, Seoul South Korea

seojt@indigo.soongsil.ac.kr, moon@computing.soongsil.ac.kr

요약

몇 년 전에 개인용 컴퓨터에서 가상현실을 구현하려고 했다면 컴퓨터의 성능 때문에 그 구현은 불가능했을 것이다. 한가지 예로 사람의 인지 능력의 80%를 차지 한다는 시야를 형성하기 위해 필요한 삼차원 가상공간을 만드는 것만 하더라도 실시간 표현을 해야 하므로 삼차원 가속기도 없는 느린 중앙처리장치 때문에 거의 불가능했다. 하지만 근래에 나온 개인용 컴퓨터를 보면 워크스테이션에 가까울 만큼의 성능을 가지고 있고 삼차원 가속보드는 그 성능이 더 낫아 졌음에도 가격이 저렴하다. 또 요즘 인텔의 일단 저가형 삼차원 가속기를 생산 할 것이라는 발표나 국내 기업의 저가형 HMD^[2,10](Head Mounted Display) 발표 등을 미루어 보아 개인용 컴퓨터에서의 가상현실이 실용화 될 단계에 와 있다고 본다.

때라서 본 논문은 실제로 개인용 컴퓨터를 이용해 가상현실을 구현하는데 있어 필요한 장비나 그 장비를 사용하는데 따른 문제점 등을 논했고 또 가상현실 기법을 이용해 응용소프트웨어를 개발하기 위한 인터페이스 방법을 소개했다.

1 서론

가상현실 시스템이 갖추어야 할 요소 중 가장 중요한 것은 인간의 감각 기관 중 가장 큰 부분을 차지하는 시각적인 요소이다. 따라서 가상현실 시스템에서 HMD는 빼놓을 수 없는 장비중 하나이다. 그런데 HMD는 그 특성상 사람의 시야를 가리게 되고 따라서 HMD를 착용한 사용자는 키보드나 마우스를

사용할 수 없게 된다. 물론 사용자가 키보드나 마우스를 전혀 사용할 수 없는 것은 아니지만 사용자가 가상공간에서는 보이지 않는 실세계에 있는 입출력 장비를 이용할 경우 사용자는 실세계와 가상공간을 동시에 생각해야 한다. 따라서 상용자의 몰입 감을 저하시키는 결과를 낳게 되므로 사용자에게 키보드나 마우스를 사용하게 하는 것은 좋지 않다. 그래서 우리는 가상공간에서 입출력을 할 수 있고 또한 사용자의 몰입 감을 높일 수 있는 장갑과 이 장갑의 위치를 감지할 수 있는 Position Tracker를 사용했다. 또 지금까지는 기술적으로 난해한 문제 때문에 사용하지 않았던 음성인식을 통한 인터페이스를 사용했는데 이 부분은 마이크로 소프트사의 SAPI^[7](Speech Application Programming Interface)를 사용했다.

2 구현상 문제점과 해결책

2.1 3D Engine

많이 알려진 삼차원 엔진은 여러 가지가 있는데 여기에서는 마이크로 소프트사가 내놓은 DirectX 3D^[3]와 SGI의 OpenGL^[9]을 비교해보자. 이 두 엔진은 서로 지원하는 가속기가 다르기 때문에 둘 중 하나를 선택해야 한다. 간혹 가다 가속기 판매 회사 측의 광고를 보면 두 엔진을 모두 완벽하게 지원하는 것처럼 쓰여있는 것을 자주 보게 되는데 보통 한가지만을 제대로 가속을 해주고 나머지는 하드웨어 가속이라고 보기에 너무 느린 성능을 지원하지 않거나 삼차원 엔진에 있는 모든 기능을 지원하지 않는 경우가 대부분이다. OpenGL의 경우 가속기 가

격이나 화면 질 등 여러 가지 따져 봤을 때 DirectX에 비해 가격은 조금 비싸지만 성능이나 화질은 더 좋다. 또 OpenGL의 경우 고가의 가속기를 사용할 경우 더 낮은 성능을 보장하고 또 워크스테이션에서 실행 해야할 경우 등을 고려한다면 OpenGL은 많은 이점을 가지고 있다. 하지만 본 논문에서는 가상현실이라는 조금은 특수한 시스템에 적용해야 하는데 OpenGL의 경우 HMD를 사용하려면 스테레오 스크린을 지원하는 고가의 가속기를 사용해야 한다. DirectX의 경우는 2D와 3D의 가속기능을 동시에 지원하므로 가상공간을 스테레오 스크린으로 표시하기 위해 두 개(가상공간 아바타의 두 눈에 의해 만들어진 스테레오 스크린)의 3D화면을 2D함수를 통해 편집해 HMD에 표시 할 수 있다. 따라서 2D기능을 지원하지 않는 OpenGL에 비해 저렴한 가격에 좀더 리플래쉬 타임을 줄인 스테레오 스크린을 구현 할 수 있다.

2.2 음성인식

음성인식기술은 가상현실 인터페이스에서 몰입감을 주는데 빼놓을 수 없는 요소중의 하나 이나 기술적으로 어려운 부분이 많고 또한 음성 인식이 저조해 대부분의 응용소프트웨어에서 사용되지 못했던 것이 사실이다. 그러나 마이크로소프트사의 SAPI(Speech Application Programming Interface)는 개발자에게 음성 인식의 복잡한 부분을 숨겨주어 쉽게 음성인식이나 표현을 사용할 수 있도록 해준다.

사용방법은 음성 인식의 경우 우선 음성을 인식하기 위해 Microsoft Voice^[7]를 실행한다. 이 프로그램은 음성을 인식해 이 음성을 윈도우 명령으로 변화시키는 역할을 한다. 사용자의 음성 인식을 높이기 위해 음성 명령을 여러번 반복해서 들려주거나 특정 명령의 인식률을 높이기 위해 한 명령만 여러번 반복하여 들려줄 수 있다. 또 명령을 추가 할 수 있으며 추가하는 명령 중에는 인터넷 홈페이지로 바로 가는 명령, 어떤 사람에게 전화를 거는 명령, 특정 프로그램은 실행하는 명령, 단순히 문자열을 만들어내는 명령 등을 추가 할 수 있다.

2.3 문자입력

가상현실장비를 착용하고 있을 경우 키보드를 사용할 수 없기 때문에 문자나 숫자를 입력하는 것이 큰 문제가 된다. 가장 좋은 방법은 말로 하는 것이지만 본 논문에서 사용한 음성 인식기는 정해진 음성명령을 받아들일 때 60%정도의 인식률만을 갖는다. 따라서 음성 명령으로 문자를 입력하는 것은 입력 단

어에 한 계가 있고 또 단어 하나 하나를 부른다 하더라도 정확히 전달이 되지 않을 경우가 많아 음성인식을 사용하는 것은 문제가 많다. 따라서 우리는 가상공간에서 사용자가 문자나 숫자를 입력 할 수 있도록 가상공간에 떠있는 키보드가 나타나도록 했고, 수치가 어떤 물체의 크기나 모양을 변화시키는 경우에는 원형으로 생긴 볼륨조절 단자를 두어 사용자가 검지로 회전을 시킬 때 숫자 값이 변하고 결과가 선택된 물체에 적용되는 것을 관찰 할 수 있게 했다.

2.4 한 화면에 표시할 수 없는 아이콘들의 표현

일반 응용소프트웨어에서 많이 쓰이는 인터페이스 중 한 가지로 지금까지 가상현실 인터페이스에서는 없거나 선택할 수 있는 아이콘의 개수를 제한했었다. 여기에서 말하는 아이콘은 메뉴나 파일 명어 쓰여진 버튼 박스 등을 말한다.

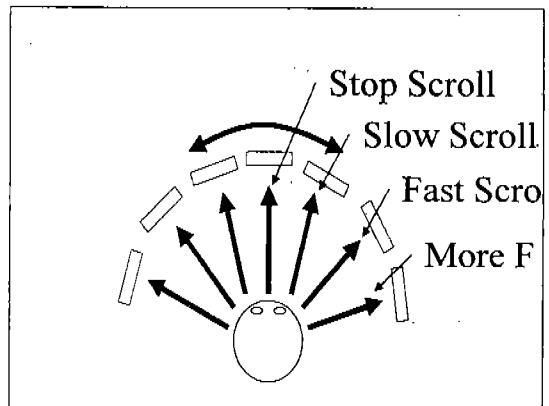


그림 1 3차원 공간 아이콘 배열 표현

아이콘의 개수가 많을 경우 2D인터페이스에서는 밑에 슬라이드바를 사용해 나머지 부분을 나타내게 했는데 이를 가상현실 인터페이스에 그대로 적용하기는 무리가 있다. 이유는 대부분의 HMD는 화소의 크기가 일반 모니터보다 좋지 않아 글자 크기를 모니터에 표현했을 때 보다 두 배 이상 크게 해야하고, 또 슬라이드바의 경우 검지로 눌러서 조종하려면 글러브로 부터 오는 데이터가 정밀하지 못하기 때문에 그 크기는 더 커야 한다. HMD의 해상도가 640*240인 것을 감안한다면 이것은 큰 문제가 된다. 이러한 문제를 해결하기 위해 우리는 HMD의 헤드 트래킹 정보를 사용하기로 했다.

우선 음성 명령으로 파일열기 모드로 들어간다. 이때 파일이 사용자의 앞에 반원모양의 배열로 나타나

고 사용자는 머리를 좌우로 움직여 파일들을 스크롤할 수 있다. 머리를 중앙에 둘 경우 스크롤이 멈추고 왼쪽이나 오른쪽으로 머리를 회전하는 정도에 따라 스크롤 하는 속도가 결정된다. 그리고 파일을 선택할 때는 선택할 파일아이콘이 화면 중앙 다시 말해 시선의 중앙에 놓는다. 이때 파일아이콘은 색깔이 밝은 색으로 변해 사용자가 알 수 있게 하고 읽어들일 경우 음성명령으로 읽어들인다.

3 전체 시스템 구성

사용한 장비는 HMD(I-glass), Glove(5th Glove), Position-Tracker(ISOTRACK II)를 사용했고, 모두 직렬포트를 사용해 컴퓨터와 입출력을 한다. 개인용 컴퓨터의 직렬포트는 1번부터 4번까지 4개가 있으나 1번과 3번 또 2번과 4번은 같은 인터럽트 번호를 사용하므로 두 포트는 동시에 사용할 수 없다. 따라서 마우스를 PS/2포트로 사용한다고 해도 PC는 두 개의 여유 포트만을 갖는다. 하지만 사용해야 할 포트는 3개이므로 컴퓨터 한 대만 가지고는 위에서 언급한 장비를 모두 동시에 사용할 수 없다. 따라서 네트워크드라이버를 만들어 두 개의 컴퓨터를 Lan으로 연결해 사용했다.



그림 2 전체 시스템 구성도

HMD와 Position Tracker는 로컬 컴퓨터에 연결하고 Glove는 다른 컴퓨터에 연결해 네트워크 드라이버를 만들었다. CORBA^[1]를 사용해 로컬함수를 사용하는 것처럼 함수를 호출하므로 보통 프로그램을 작성하는 것과 큰 차이는 없었다. CORBA는 OMG에서 제안한 네트워크 객체의 표준안으로 구현된 소프트웨어는 여러 가지가 있다. 이 소프트웨어는 인터페이스를 정의하고 이를 컴파일 하면 서버용 코드와 클라이언트용 코드를 만들어 주어 사용자는

실제 실행될 코드를 서버에 넣고 호출하는 부분은 클라이언트에서 보통 함수를 호출하듯이 호출할 수 있게 해준다. 따라서 장비와 드라이버를 서버에 설치하고 사용자가 사용하는 컴퓨터는 클라이언트가 되어 서버에 있는 장비를 사용하게 된다.

3.1 Glove

본 논문에서 사용한 Glove는 Fifth Dimension Technology에서 나온 5th Glove를 사용하였다. 이 글러브는 각각의 손가락에 하나의 광섬유가 있어 빛의 양을 감지해 구부러진 정도를 측정한다. 또 Pitch와 Roll을 측정할 수 있는 Sensor가 부착되어 있다. Pitch와 Roll값은 각각 180도를 측정할 수 있으며 모든 값은 0에서 255로 나타내어진다. 그리고 이 Glove는 약간의 문제가 있는데 Glove 자체의 문제인지 사용상의 부주의로 인한 건지는 알 수 없으나 몇 개월쯤 지난 후에는 손가락의 구부러진 정도의 측정값이 정확하게 나오지 않는다. 필터링을 하지 않으면 가상공간에서 손가락의 모양이 실세계에서 취한 동작과 다르게 되고 심한 경우 몇 개의 손가락은 거의 구부러진채 퍼지지 않는다.

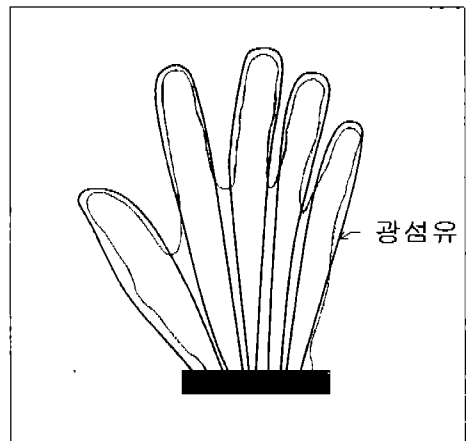


그림 3 The 5th Glove

글러브의 세부적인 사용방법을 보자, 속도는 19200 이고 글러브가 장착돼 있는지 확인하기 위해 직렬포트에 'A'를 보내면 글러브에서 'U'로 응답한다. 만약 응답이 없거나 다른 값이 온다면 직렬포트의 설정이 잘못되어 있거나 글러브가 장착되어 있지 않은 것이다. 다음은 'C'를 보내는데 이 값을 글러브에서 받으면 연속적인 데이터 값을 보낸다. 전송되어져 오는 데이터 값은 9바이트씩 연속적으로 오는데 첫 번째 값은 80H로 Header이고 나머지 7바이트는 데이터이고 나머지 한 바이트는 에러를 체크하기 위한 값이다. 7바이트의 데이터 값은 엄지부터 다섯 개의

손가락의 구부러진 정도를 나타내고 나머지 두 바이트는 Roll과 Pitch값을 나타낸다.

3.2 HMD

머리에 쓰고 두 눈앞에 각각 화면이 달려있어 입체 영상을 보게 하는 장비이다. 우리는 Virtual IO에서 나온 I-Glasses를 사용했다. 이는 최초의 PC용 HMD로 가격이 싸고 머리의 방위각을 감지할 수 있는 Tracker를 추가로 장착할 수 있으며 Stereo Screen을 만들기 위해 비디오카드에서 받은 신호를 라인단위로 홀짝 순서로 좌, 우 또는 우, 좌 순서로 Display하는 Interrace방식을 사용하고 있다.

이 HMD는 단점이 있는데 인터레이스방식을 사용하기 때문에 해상도가 640*480에서 640*240으로 반으로 줄고 또한 Tracker는 방위각만을 측정할 수 있고 도스에서 제공되는 Application을 사용할 때는 적어도 Yaw값의 경우 360도 회전각이 측정이 되나 윈도우에서 사용할 경우 Joystick Virtual Device로 설정해 Joystick API를 사용하는데 이 때는 Yaw값이 180도 밖에 측정되지 않는다.

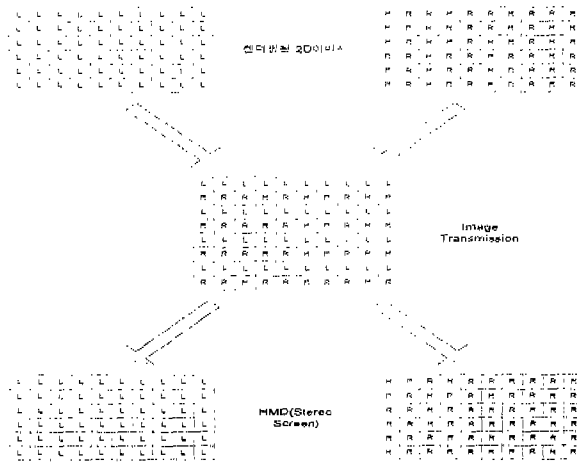


그림 4 Interrace 방법

3.3 Position Tracker

실제공간에서 팔이나 머리 등에 부착해 위치 및 방위각을 측정할 수 있는 장비로 여러 개를 사용할 경우 모션캡처가 가능하다. Polhemus ISOTRACK II는 Magnetic Position Tracker로 두 개의 수신기를 달수 있고 그중 한 개만 사용할 경우 초당 60번 두 개 모두 사용할 경우 30번 감지된 값을 받을 수 있다. 그리고 그 값이 수신기에서 컴퓨터까지 도달하는데 걸리는 시간은 20ms이다.

Tracker로부터 오는 데이터는 X, Y, Z, Yaw, Pitch, Roll로 구성되어 있는데 이중 X, Y, Z는 위치(Position)값을 나머지는 방위 값을 나타낸다. 방

위 값은 그림5에서 보는 바와 같이 Yaw는 Z축을 Pitch는 X축을 Roll은 바뀌어진 X축을 중심으로 회전하는 값을 나타낸다. 여기서 Roll값은 두 축 다 음에 적용하면 바뀌지 않은 X축을 중심으로 회전하므로 Roll값을 먼저 적용한 후에 나머지를 적용해야한다. 또 Pitch값은 Yaw값이 양의 값일 경우 부호가 바뀌므로 코딩할 때 이를 고려해야한다. 밑에 있는 코드는 위에서 말한 것들을 적용한 것이다.

```
float sign;
if(Isotrack[3] > 0) sign = -1; else sign = 1;
```

```
Hand->pHandChild->AddRotation(D3DRMCOMBINE_REPLACE, 1, 0, 0, Isotrack[5]);
Hand->pHandChild->AddRotation(D3DRMCOMBINE_AFTER, 0, 0, 1, Isotrack[3]);
Hand->pHandChild->AddRotation(D3DRMCOMBINE_AFTER, 1, 0, 0, sign*Isotrack[4]);
```

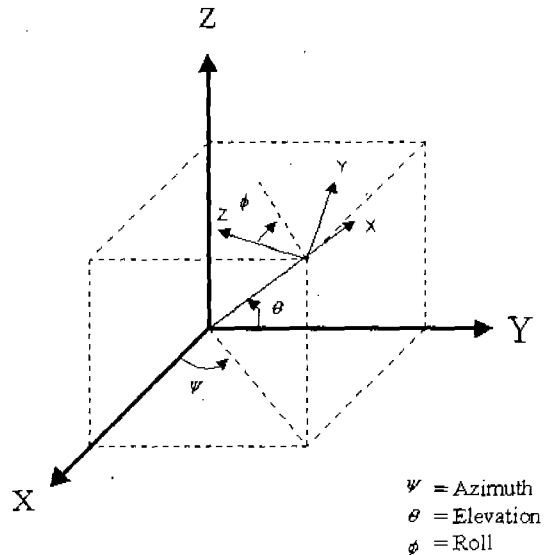


그림 5 Tracker로부터 오는 방위 값

4 결론

아이콘선택의 경우 사용자가 고개를 몇 번 좌우로 움직여보고 쉽게 사용방법을 알 수 있었으나 수치 입력에서 회전형의 입력 장치는 조절방법을 잘 알아 내지 못했다. 문자 입력에서는 초보자의 경우 키보드를 사용했을 때와 입력시간이 큰 변화가 없었으나 키보드에 익숙한 사용자는 키보드 자판을 알고 있는 초보자와 큰 차이가 없을 정도로 현저히 느려졌다. 이는 키보드가 일반 응용소프트웨어에서 차지하는 부분을 생각한다면 중요한 문제이다. 가상현실 기법

에서 이를 해결할 수 있는 방법은 음성인식률을 높이는 것인데 이 또한 쉬운 문제는 아니다. 가상현실 장비나 기법이 고급화되지 않는 한 이러한 문제는 계속 남아 있을 것이다.

Position-Tracker의 경우 진동이 심하고 방위각의 경우 정확하게 측정하지 못한다. 또 사용자의 행동이 화면에 나타나기까지의 지연시간이 길어 사용자가 몰입감을 느끼는데 장애 요소가 되었다. 지연시간의 경우 장비를 바꾸지 않는 한 수정할 수 없지만 진동이나 방위각의 경우에는 필터링을 하면 어느 정도 해결이 되리라고 본다. 또 개인용 컴퓨터에서 가상현실을 구현하기 위해서는 장비의 가격이 몇 십만원 이하이어야 하는데 현재의 Position Tracker는 그 가격이 너무 비싸기 때문에 큰 걸림돌이 된다. 따라서 트래커의 가격이 인하되거나 저렴한 가격에 위치나 방위각을 측정할 수 있는 다른 기술이 나와야 할 것이다.

6 참고 문헌

- [1] Robert, Dan Orfali Harkey, "Client/Server Programming with Java and CORBA", A Wiley-Interscience Publication, 1997
- [2] Grigore Burdea, Philippe Coiffet, "Virtual Reality Technology", A Wiley-Interscience Publication, 1994
- [3] Jason Kolb, "WIN32 Game Developer's Guide DirectX3", Waite Group Press, 1997
- [4] Mark R. Mine, Frederick P. Brooks Jr, Carlo H. Sequin, "Moving Object In Space: Exploiting Proprioception In Virtual-Environment Interaction", ACM SIGGRAPH Conference Proceedings, 1997
- [5] Popyrev, I., Billinghamurst M., Weghorst, S., Ichikawa, T. "Go-Go Interaction Technique: Non-Linear Mapping for Direct Manipulation in VR." In Proceedings of UIST, 79-80, 1996
- [6] Popyrev, I., Billinghamurst M., Weghorst, S., Ichikawa, T. "Go-Go Interaction Technique: Non-Linear Mapping for Direct Manipulation in VR." Technical Sketches, SIGGRAPH, 1996
- [7] Michael C. Amundsen, "MAPI, SAPI, & TAPI Developer's Guide", Sams Publication, 1996
- [8] SGI, "Comparing OpenGL and Direct 3D", <http://www.sgi.com/Technology/OpenGL/direct-3d.html>, 1995
- [9] Mason Woo, Jackie Neider, Tom Davis, "OpenGL Programming Guide Second Edition", Addison-Wesley, 1997
- [10] Roy S. Kalawsky, "The Science of Virtual Reality and Virtual Environment", Addison-Wesley, 1993
- [11] Polhemus, "3SPACE USER'S MANUAL", 1993
- [12] 심우섭, "PC 상에서 몰입형 가상현실 시스템 구현", 프로그래밍 세계 1996. 4