

확장된 사용자 인터랙션을 이용한 VRML 기반 운전 시뮬레이션

이선민, 김명희
이화여자대학교 컴퓨터학과
e-mail : {blue,mhkim}@mm.ewha.ac.kr

Driving Simulation based on VRML with Extended User Interaction

Seon-Min Rhee and Myoung-Hee Kim
Dept. of Computer Science & Engineering, Ewha Womans University

요 약

VRML(Virtual Reality Modeling Language)은 비/반몰입형 가상현실을 구축하기 위하여 널리 사용되는 언어이다. 그러나 이 언어는 키보드와 마우스 위주의 단순한 형태의 사용자 인터랙션만을 제공하기 때문에 다양하고 현실감 있는 가상환경을 구축하기엔 많은 제약이 따른다. 특히, 운전 시뮬레이션과 같이 확장된 사용자 인터랙션을 필요로 하는 가상 환경을 구축하기 위해서는 핸들/페달 장치를 이용할 수 있도록 하여 사용자에게 실제 상황과 흡사한 환경을 제공할 수 있어야 한다. 따라서 본 논문에서는 사용자 인터랙션을 확장하여 핸들/페달 장치를 이용할 수 있는 VRML 기반 운전 시뮬레이션을 구현하였다.

1. 서론

가상현실(Virtual Reality)이란, 3 차원 컴퓨터 인터페이스를 이용하여 사용자의 몸짓, 음성 등을 감지하여 사용자의 의도를 추출하고 이에 따른 적절한 응답을 음향이나 영상 등으로 출력하여, 사용자로 하여금 컴퓨터가 생성하는 가상환경을 직접 체험할 수 있도록 만들어 주는 제반 기술을 가리킨다[1, 2, 3, 4]. 가상현실의 궁극적인 목적은 컴퓨터가 만들어 낸 가상세계에 사용자를 완전히 몰입 시키는 것이지만, 이를 구현하기 위해서는 고가의 장비가 필요하다. 반면, 비/반몰입형 가상현실은 몰입형 가상현실에 비해 높은 해상도의 디스플레이 환경을 제공해 주고, 비교적 쉽고 효율적으로 가상공간을 구축할 수 있다. 더욱이 인터넷의 급속한 확산과 더불어 웹에서 이용 가능한 비/반몰입형 가상 현실의 필요성이 대두되고 있으며, 이를 구현하기 위한 언어로써 많이 사용되고 있는 VRML(Virtual Reality Modeling Language)의 중요성이 또한 커

지고 있다. 그러나 VRML 은 키보드 또는 마우스를 중심으로 하는 단순한 형태의 사용자 인터랙션만을 제공하기 때문에 보다 현실감 있는 가상환경을 구축하는데 있어 많은 제약이 따른다. 따라서 본 논문에서는 이와 같은 단점을 개선하여 확장된 사용자 인터랙션을 이용한 VRML 기반 운전 시뮬레이션을 구현하고자 한다.

본 논문의 구성은 다음과 같다. 2 장에서는 기존의 운전 시뮬레이션에 대하여 살펴보고, 3 장에서는 확장된 인터랙션을 이용한 운전 시뮬레이션의 설계 및 구현에 대하여 설명한다. 4 장에서는 구현 결과를 보여주며, 마지막으로 5 장에서 본 논문의 결론을 맺는다.

2. 관련 연구

운전 시뮬레이션은 복잡하고 정교한 구조를 가진 자동차 장치를 포함한 차량 시뮬레이터와 이의 조작을 통한 실제 운전 환경을 제공하여 실험자에게 현실성을 부여하고 이를 다각적으로 활용할 수 있도록 해주는 기능을 수행한다[5]. 운전자가 자동차를 운전하

본 연구는 정보통신부 정보통신연구센터 지원사업(ITRC)과 과학기술부 국가지정연구실 사업에 의해 지원 받았습니다.

는 동안 수행하는 핸들의 동작, 가속 페달 조작 등을 통해 야기되는 차량의 운동을 실시간으로 시뮬레이션 함으로써 그 움직임을 예측하고 그 결과를 운동, 시각 및 음향을 통해 운전자에게 피드백하여 차량 시뮬레이터에 탑승한 운전자가 실제로 자동차를 운전하고 있다는 느낌을 갖게 해 준다[6].

기존의 운전 시뮬레이션은 실제 자동차의 움직임과 정확하게 일치하도록 구축하고자 하였기 때문에 복잡하고 방대한 양의 계산이 필요하다. 따라서 이러한 작업을 수행하기 위하여 고성능, 고비용의 워크스테이션 위주의 연구가 진행되어 왔으며 이들 대부분이 C/C++, OpenGL 을 이용하여 구현 되었다.

가상 현실이나 컴퓨터 게임 등의 시각 위주 응용 분야에서 필요로 하는 운전 시뮬레이션은 복잡하고 정교한 모델보다는 단순화 된 모델을 필요로 하기 때문에 이전에 연구 되었던 복잡한 수식을 이용하는 운전 시뮬레이션을 사용하기에는 비용 부담이 크다. 따라서 이러한 분야에서, 일반적인 PC 를 이용한 비/반몰입형 가상 공간상에서의 운전 시뮬레이션을 구축할 수 있는 환경을 마련하는 것이 필요하다.

비/반몰입형 가상 공간 구축 시, 가상 객체와 사용자간의 인터랙션 기법을 제공해 주는 VRML 을 이용하면 쉽고 효율적으로 가상 공간을 구축할 수 있을 뿐만 아니라, 인터넷을 통한 다중 사용자의 참여가 가능해지기 때문에 보다 다양하고 현실감 있는 가상 공간을 구축 할 수 있을 것이다. 이를 위하여 마우스 이외의 핸들/페달 장치와 같은 다양한 장치를 통한 사용자 인터랙션이 가능하도록 사용자 인터랙션의 종류를 확장해야 한다.

3. 설계 및 구현

3.1 구현 환경

본 연구에서는, 확장된 사용자 인터랙션을 이용한 VRML 기반 운전 시뮬레이션을 구현하기 위하여 독일 ZGDV(Computer Graphics Center)[10] 연구소에서 개발한 Avalon 을 이용하였다[7, 8]. VRML 에서는 사용자 인터랙션으로써 마우스의 움직임을 추적하기 위한 TouchSensor 노드만을 제공한다. 이 노드를 통하여 마우스와 가상 객체와의 위치 관계를 파악할 수 있으며, 가상 객체의 선택 여부를 알 수 있다. 따라서 VRML 을 기반으로 하는 가상 환경에서 마우스 이외의 사용자 인터랙션을 확장하기 위해서는 기존의 VRML[9]에서 제공해 주는 모든 기능을 지원해 줌과 동시에 필요한 기능을 추가할 수 있도록 해 주는 구현 환경을 필요로 한다. Avalon 은 가상/증강현실 어플리케이션 개발을 위하여 만들어진 툴킷으로 ISO/IEC 14772 에 의해 표준으로 지정된 VRML97 을 기반으로 하며, 위에서 제시한 구현 환경을 제공해 준다. 즉, VRML97 에 정의된 모든 노드를 사용할 수 있을 뿐만 아니라 정의된 노드 이외의 새로운 노드의 생성이 가능하고, 기존 노드에 새로운 필드와 이벤트를 추가 함으로써 그 기능을 확장할 수 있다. 운전 시뮬레이션 구현에 필요한 핸들/페달 장치를 사용하기 위한

ScalarInputSensor 노드의 필드와 이벤트는 그림 1 과 같다.

```

ScalarInputSensor{
    field      SFString    label
    eventOut   SFFloat    value_changed
}

label
해당 장치에(본 연구의 경우, 핸들/페달장치)
부여된 이름

value_changed
본노드에 연결되어 있는 장치의 값이 변할 때
0.0-1.0 사이의 값을 내보낸다.
    
```

그림 1. ScalarInputSensor 노드의 필드와 이벤트

3.2. 자동차 동작 조작을 위한 물리적 접근

본 논문에서 구현한 자동차의 움직임은 크게 회전, 전진, 정지의 세 가지의 경우로 나누어 볼 수 있으며, 각 동작을 표현하기 위한 물리적 접근 방식 및 공식은 다음과 같다.

3.2.1 회전

자동차의 회전을 구현하기 위해서는 두 가지 과정을 거쳐야 한다. 우선, 페달에 의한 자동차의 속력과 핸들에 의한 회전 각도를 고려하여 X, Z 축 상으로의 이동 거리를 계산하고 자동차의 새로운 위치를 찾는다. 이에 따라 자동차를 이동 시킨 후, 자동차 차체를 회전 각도에 따라 적절하게 회전해 주어야 실제로 자동차가 회전한 것처럼 보여지게 된다. 그림 2 는 핸들/페달 값을 이용하여 회전을 구현하기 위한 과정이며, 그림 3 은 자동차 회전시의 상황을 보여준다.

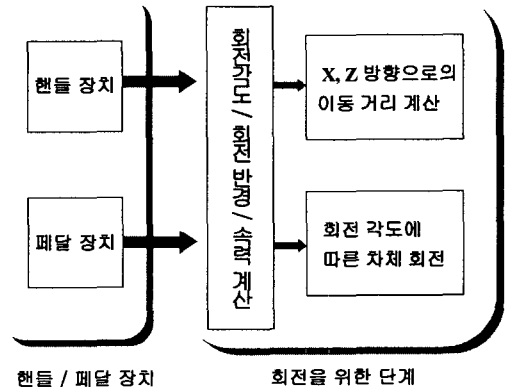


그림 2. 핸들/페달 장치에 의한 회전

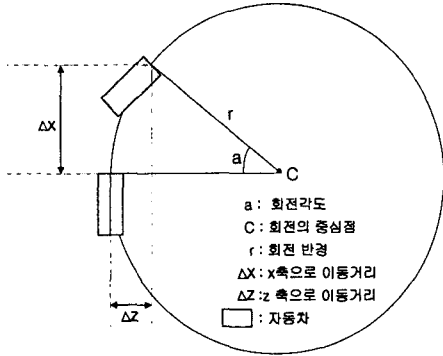


그림 3. 자동차의 회전

1) 회전 각도 및 회전 반경

자동차의 새로운 위치를 계산하기 위하여 자동차 회전 각도와 그에 따른 회전 반경을 구해야한다. 그림 4는 핸들의 회전 각도와 그에 따른 회전 반경 사이의 관계를 나타낸다.

회전 각도는(그림 4, a) 사용자가 핸들을 돌림으로써 정해지는 값으로 자동차의 회전 정도를 결정한다.

회전 반경(그림 4, r)은 회전 각도에 따라서 바뀌는 값이다. 회전 각도에 따라 자동차는 정해진 회전 반경을 반지름으로 하는 원을 그리며 회전 하게 되며 (식 1)과 같이 구할 수 있다.

$$r = d / \sin(a) \quad (식 1)$$

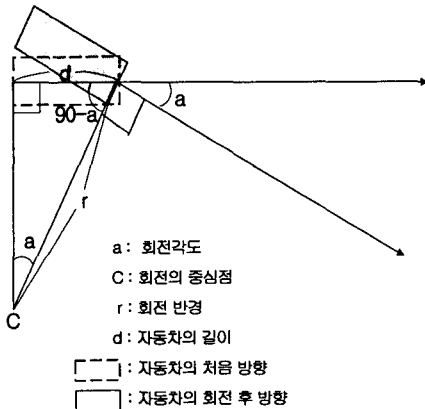


그림 4. 자동차의 회전 각도와 회전 반경 간의 관계

2) 회전에 따른 자동차 위치변화 (X, Z)

(식 2)와 (식 3)는 정해진 시간 프레임당 x, y 축으로의 이동거리를 구하는 공식이다. 여기에서, turnRadius는 회전 반경, Δa는 시간 프레임의 회전 각도를 나타낸다.

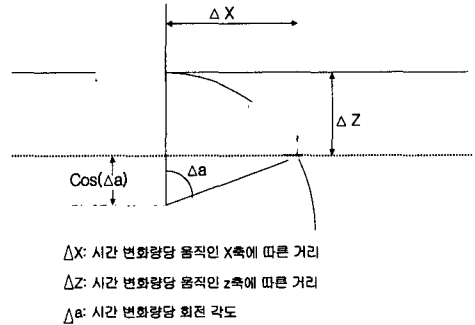


그림 5. 시간 변화량에 따른 X,Z 축으로의 이동 거리

$$\Delta X = \text{turnRadius} \times \sin(\Delta a) \quad (식 2)$$

$$\Delta Y = \text{turnRadius} \times \cos(\Delta a) \quad (식 3)$$

위의 식에서 구한 x,y 축으로의 이동거리를 가지고 현재의 자동차의 위치는 이동시키기 위한 공식은 (식 4)와 (식 5)와 같이 나타낸다. currentX, currentY는 현재 자동차의 x,y 축상의 위치를 의미하며, ΔX, ΔY는 (식 2),(식 3)에 의해 구한 값이다. 그림 5는 시간 변화량에 따른 x,z 축 상의 이동거리를 나타낸다.

$$\text{currentX} = \text{currentX} + \Delta X \quad (식 4)$$

$$\text{currentY} = \text{currentY} + \Delta Y \quad (식 5)$$

3) 회전 각도에 따른 자동차 차체 회전

자동차의 위치이동에 이어 차체를 회전 각도만큼 돌려줌으로써 회전을 완성한다. 본 논문에서는 오른손 좌표계를 이용하여 구성하였으므로, 자동차의 회전은 y 축을 중심으로 한다.

3.2.2 전진

자동차의 이동 거리를 S, 속력을 V, V의 속도로 움직인 시간을 T라고 했을 때, 이동 거리 S는 (식 6)과 같이 나타낼 수 있다. 그러나 자동차가 전진할 때 항상 일정한 속도로 움직이기 보다는 매 순간 다른 속력으로 움직이는 경우가 일반적이다. 따라서 시간을 적절한 간격으로 나눈 후 각 시간 간격에 따라 해당 속력을 적용 시키는 것이 바람직하다. (식 9)는 i+1 번째 시간 프레임에서의 이동 거리를 구하는 공식으로, i 번째 시간 프레임까지 움직인 거리와 i 번째 시간 프레임과 i+1 번째 시간 프레임 사이에 움직인 거리를 계산하여 더해줌으로써 구할 수 있다. Si+1는 자동차가 i+1 번째 시간 프레임까지의 이동 거리, Si는 i 번째 시간 프레임까지의 이동 거리이고 ΔS는 (식 8)에서와 같이 구할 수 있다. Vi+1는 i+1 번째 시간 프레임에 적용할 속력, ΔT는 i 번째 시간 프레임과 i+1 번째

째 시간 프레임 사이의 시간 간격이다. 여기에서 V_{i+1} 는 (식 7)와 나타낼 수 있다. A는 현재 페달의 밟힌 정도에 따라 자동차의 가속도로 바뀌준 값이다

$$S = V \times T \quad (\text{식 6})$$

$$V_{i+1} = V_i + A \times \Delta T \quad (\text{식 7})$$

$$\Delta S = V_{i+1} \times \Delta T \quad (\text{식 8})$$

$$S_{i+1} = S_i + \Delta S \quad (\text{식 9})$$

3.2.3 정지

사용자가 브레이크 페달을 밟으면 감속도가 커짐으로써 속력은 줄어들게 된다. 따라서 자동차의 정지는 속력의 감속을 이용하여 구현할 수 있다. 위에서 (식 7)을 (식 10)과 같이 수정함으로써 직진과 정지를 동시에 처리해 줄 수 있다. B는 현재의 감속도를 나타내는 값으로 핸들/페달 장치의 브레이크 페달을 밟았을 때 발생하는 값을 이용하여 구한다.

$$V_{i+1} = V_i + (A-B) \times \Delta T \quad (\text{식 10})$$

4. 구현 결과

회전, 전진, 정지의 기능은 핸들/페달 장치를 조작하여 수행할 수 있으며, 이를 자동차 모델에 적용시켜서 시각화 할 수 있다(그림 6). 이 경우, 사용자는 자동차 외부에서 그 움직임을 관찰할 수 있으며, 자동차의 움직임에 따른 외부 경관의 변화를 관찰 하기 위하여 구현 모듈을 고속도로 모델에 적용하였다(그림 7).

구현 환경은 PentiumII(Intel MMX) Processor, 256MB Main Memory이며, 핸들/페달 장치는 Microsoft Side Wider Precision Racing Wheel 을 사용하였다.

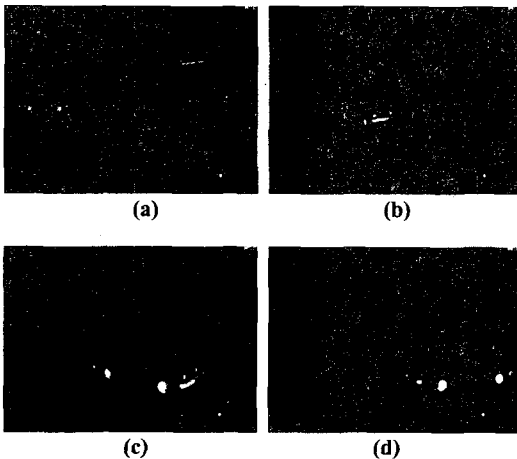


그림 6. 핸들/페달 장치 조작에 의한 자동차 전진, 정지, 회전의 사례 : (a)->(b)->(c)->(d)

5. 결론 및 향후 연구

본 논문에서는 VRML을 기반으로 하는 가상현실 속에서 보다 현실감 있는 운전 시뮬레이션을 구현하

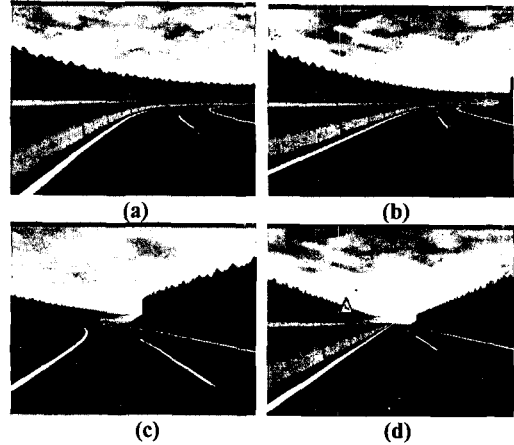


그림 7. 핸들/페달 장치 조작에 따라 변하는 외부 경관 사례 (고속도로 모델에 적용):(a)->(b)->(c)->(d)

기 위하여 키보드 또는 마우스 이외에도 핸들/페달 장치를 이용할 수 있도록 사용자 인터랙션을 확장하고 이를 자동차 모델과 고속도로 모델에 적용시켜 보았다.

향후 연구로는, VRML에서 핸들/페달 장치 이외에도 다양한 종류의 사용자 인터랙션이 가능하도록 그 기능을 확장하여 실제 환경과 흡사한 비/반몰입형 가상현실을 구축할 수 있도록 해야 할 것이다.

참고문헌

- [1] Dave Stampe, Bernie Roehl, John Eagan. "Virtual Reality Creations Explor, Manipulate, and Create Virtual Worlds on your PC", Waite Group Press Corte Madera California, 1993
- [2] Benford, Steve, Lennart Fahlen, "A Spatial Model of interaction in Large Virtual Environments", Proc. of 5th MultiG Workshop, 1992
- [3] Earnshaw, R.A., M.A. Gigante, H. Jones, "Virtual Reality System", Academic Press, 1993
- [4] 성운재, 원광연, "가상세계 저작도구의 설계 및 구현", 한국시뮬레이션학회 논문지 제 4 권, 제 1 호, 1995.6
- [5] Polydoropoulou, A. & Moshe, B.A., Travel simulators for data collection on driver behavior in the response of information, Transpn. Res.-C, Vol13, pp143-159, 1995
- [6] 조준희, 신준수, 이운성, "운전자 인자 연구를 위한 실시간 차량시뮬레이터" 1998
- [7] Johannes Behr, Adreas Frohlich, "AVALON, an Open VRML VR/AR System for Dynamic Application", Computer Graphics Topics, Vol.10, No.1 p28-30
- [8] "Avalon", www.zgdv.de/departments/z2/vc/projects/avalon/index.html
- [9] Jed Hartman, Josie Wernecke, "The VRML 2.0 Handbook-Building Moving Worlds on the Web", Addison Wesley Developers Press, 1998
- [10] Computer Graphics Center, www.zgdv.de