
테이블 탑 디스플레이용 LED기반 객체 컨트롤 시스템

증강현실을 위한 테이블 탑 디스플레이용 입력장치

Object Control System Based on LED for Table Top Display

오진식*, Jinsik Oh, 정종문**, Jongmun Jeong, 김정식**, Jeongsik Kim,
양형정***, Hyungjeong Yang, 이칠우****, Chilwoo Lee

요약 본 논문에서는 테이블 탑 디스플레이의 사용에서 증강현실을 구현하기 위해 현실 세계에서의 객체를 통한 가상세계의 객체에 영향을 줄 수 있는 객체 컨트롤 입력장치를 제안한다. 이를 위해 본 논문에서는 테이블 탑 인터페이스의 LED 기반 객체 컨트롤 입력장치를 사용하여 네트워트상에서 상호 화면의 제어를 가능하게 하고 입력이 능동적이고 자연스럽도록 하였다. 객체 컨트롤 장치는 적외선 LED를 이용하여 테이블 탑 디스플레이에서 인식이 되도록 하였다. 테이블 탑에서 이와 같은 기능들의 구현을 본 논문에서는 에어하키 게임의 구현을 통해 보인다. 에어 하키는 테이블 위에 퍽을 놓고 라켓으로 퍽을 쳐서 상대방의 골문에 넣어 점수를 얻는 게임이다. 본 논문에서 제안한 객체 컨트롤 장치를 이용함으로써 사용자는 증강현실에서 실제 객체의 이동으로 가상 세계 객체에 영향을 줄 수 있는 보다 실감나는 인터페이스를 느낄 수 있다.

Abstract In this paper, we propose an object control system on table top display to support augmented reality in which objects in real-world control computer-generated data. We developed LED-based object control system over table top display so that it is capable of controlling networked table top display. The LED-based object control system supports more natural interactive interface. In this paper, infrared LED is used to implement the object control device to be recognised by table top display. We applied LED-based object control system in air hockey game where a team has a point if their puck goes to opponent's net on a table. Since users can control objects in cyber world with the proposed object control device, it is realized augmented reality.

핵심어: 증강현실, 객체 컨트롤, 테이블탑 디스플레이, 물리엔진

Keywords: Augmented Reality, Object Control, Table Top Display, Physics Engine

본 논문은 문화관광부 및 한국문화콘텐츠진흥원이 주관하는 전남대학교 문화콘텐츠기술연구소 육성사업의 연구비 지원에 의해 수행되었음.

*주저자 : 전남대학교 전자컴퓨터공학부 학사 e-mail: js21cfriend@hanmail.net

**공동저자1 : 전남대학교 전자컴퓨터공학부 학사 e-mail: amiris@nate.com

**공동저자2 : 전남대학교 전자컴퓨터공학부 석사 e-mail: wind1105@lycos.co.kr

***교신저자 : 전남대학교 전자컴퓨터공학부 교수; e-mail: hjyang@chonnam.ac.kr

****교신저자 : 전남대학교 전자컴퓨터공학부 교수; e-mail: leecw@chonnam.ac.k

1. 서론

최근에 인간과 컴퓨터간의 자연스러운 상호작용을 위하여 시각을 기반으로 한 사용자 의도 및 행위를 인식하기 위한 연구가 활발히 진행되어 왔다. 그 중에서도 손을 이용한 제스처 인식은 시각 기반 인식 분야에서 핵심 기술 분야로 계속 연구되어 왔으며, 이를 이용한 인간과 컴퓨터의 상호작용에 관한 연구가 활발하게 진행되고 있다.

테이블은 인간이 커뮤니케이션을 하는 장소로 많이 이용되고 있다. 이러한 테이블 위에서 컴퓨터와 상호작용을 할 수 있다면 좀 더 자연스럽게 컴퓨터를 이용할 수 있을 것이다. 테이블 탑 디스플레이는 인간의 직관적인 도구인 손을 이용하여 쉽게 컴퓨터와 상호작용을 할 수 있도록 지원한 시스템이다. 즉, 손의 터치 포인트를 인식하여 입력을 받아들임으로써 컴퓨터와 상호작용이 가능하게 한다. 그러나 손의 터치 포인트를 입력으로 이용하는 것은 여러 테이블 탑 응용에서 실감나는 상호작용이 어렵게 하는 요인으로 작용한다.

따라서 본 논문에서는 테이블 탑 디스플레이의 사용에서 실감나는 상호 작용[1]을 지원하기 위해 현실 세계의 객체를 통한 가상세계의 객체에 영향을 줄 수 있는 객체 컨트롤 입력장치를 제안하여 증강 현실을 가능하게 한다. 이를 위해 본 논문에서는 테이블 탑 인터페이스의 LED 기반 객체 컨트롤 입력장치를 사용하여 네트워트상에서 상호 화면의 제어를 가능하게 하고 입력이 능동적이고 자연스럽게 하였다. 객체 컨트롤 장치는 적외선 LED를 이용하여 테이블 탑 디스플레이에서 인식이 되도록 하였다.

본 논문에서는 테이블 탑에서 이와 같은 기능들의 구현을 에어 하키 게임을 통해 보인다. 에어 하키는 테이블 위에 퍽을 놓고 라켓으로 퍽을 쳐서 상대방의 골문에 넣어 점수를 얻는 게임이다. 본 논문에서 제안한 물리엔진을 이용함으로써 사용자는 보다 실감나는 인터페이스를 느낄 수 있다.

2. 관련연구

협동적 상호작용이란 공동 작업을 하는 구성원들이 직간접적인 의사소통을 통하여 어떤 현상을 일으키거나 시스템을 조작하는 것을 말한다. 테이블 탑 디스플레이의 상호작용 방법은 크게 두 가지로 나눌 수 있다[2][3].

2.1 정전 용량 방식을 이용한 상호 작용

정전 용량 방식을 이용한 상호 작용 스크린 밑에 Force-Sensitive-Resistors의 정전 용량의 변화량을 측정할 수 있는 장치를 장착하여 손의 위치나 접촉정보를 감지할 수 있도록 한다. 정전 용량 기반 방식은 시스템 복잡도를 크게 낮추는 효과가 있으며, 스크린에 손의 접촉여부와 더불어 얼마나 접

근했는지의 접근성 여부를 판별할 수 있다는 장점이 있다. 이러한 장점을 활용하여 최종적으로 상호작용이 가능한 표면을 현실 세계로 만들 수 있다. 그러나 이 시스템들은 스크린 속에 안테나 및 센서 등의 전자적 장치가 내장되어 있으므로 거의 대부분이 상부투영(top-down projection) 방식을 채택하고 있다. 따라서 손과 투영된 이미지가 겹치는 현상이 자주 발생하여 사용자는 겹치는 부분과 스크린 위에서 손이 움직이는 것을 자주 응시하게 되어 주의 집중을 방해하는 결과를 초래한다. 또한 높은 해상도를 구현하기 위해서는 아주 많은 센서 연결을 필요로 하므로 제작비용이 매우 증가한다.

2.2 시각 기반 방법을 이용한 상호 작용

시각 기반 감지 방법은 비디오 카메라를 이용하여 스크린 위에 놓여진 객체를 검출 및 인식하여 입력신호를 만드는 방법이다. 이러한 방법은 저렴한 비용으로 시스템을 구성하고 높은 해상도의 데이터를 얻을 수 있으나 카메라만을 이용하여 인식해야 하기 때문에 정확도나 인식률이 떨어진다.

논문에서 사용한 테이블 탑은 시각기반 방식 중 FTIR원리를 이용하여 제작되었다(그림 1). 테이블 탑 디스플레이의 구조는 [그림 2]와 같다. 테이블 안쪽에 프로젝터와 적외선 카메라가 설치되어 있고, 테이블 위는 특수 제작된 아크릴 판으로 제작되어 있다. 이 아크릴판 모서리에는 적외선 LED가 부착되어 있어서 아크릴판에 적외선을 비추게 되면 아크릴판 내에서 전반사 되게 된다. 아크릴판 위에 손가락을 대게 되면 아크릴판 내에서 전반사 되던 적외선이 Frustrated 현상 때문에 수직으로 빛이 분산되어 나온다. 분산되어 나온 적외선을 적외선 카메라가 촬영하게 되고 촬영된 영상을 분석하여 컴퓨터 화면의 어느 위치에 손가락 신호를 보낼지 정하게 된다. 이와 같은 원리로 기존 터치스크린과는 다르게 여러 개의 터치정보도 분석하여 컴퓨터에 입력을 줄 수 있다.

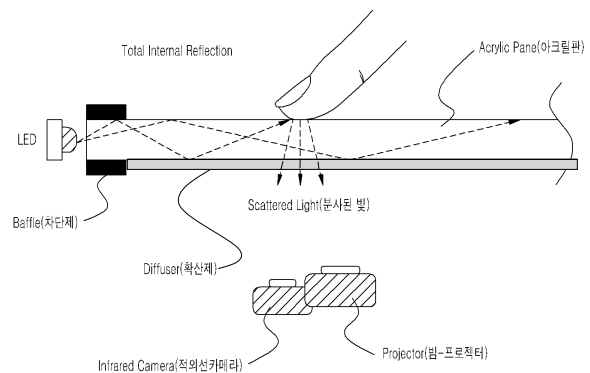


그림 1. FTIR원리를 이용한 테이블 탑 디스플레이

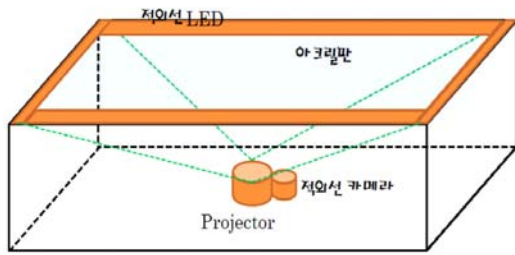


그림 2. 테이블 탑 디스플레이 구조

3. 네트워크를 통한 객체의 컨트롤

3.1 객체의 상호작용 표현

평면인 화면의 2차원 좌표 값만으로는 3차원 공간의 객체를 선택하는 것은 쉽지 않다. 이 문제를 해결하기 위하여 2D인 좌표 값을 3D로 변환하는 과정이 필요하다[4].

화면의 마우스 좌표를 이용하여 실제 3D공간의 좌표를 구하기 위해서는 OpenGL에서 제공하는 [그림 3]의 gluUnproject 라는 함수를 사용한다. gluUnproject 함수[5]로 두 점을 구한 후 두 점을 지나는 직선의 방정식을 구한다. 두 점의 좌표를 각각 (x_1, y_1, z_1) , (x_2, y_2, z_2) 라고 하면 공간에서 두 점이 만드는 직선의 방정식은 다음의 수식(1)과 같다.

$$(objx, objy, objz) = (x_1, y_1, z_1) + t(x_2, y_2, z_2) \quad \text{①}$$

이 직선과 첫 번째로 만나는 3D 공간내의 객체가 평면인 화면에서 선택한 좌표에 그려진 3D 공간의 객체이다.

```
GLint gluUnproject( GLdouble winX,
                   GLdouble winY,
                   GLdouble winZ,
                   const GLdouble *model,
                   const GLdouble *proj,
                   const GLint *view,
                   GLdouble* objX,
                   GLdouble* objY,
                   GLdouble* objZ )
winX, winY - 윈도우 좌표
winz - 0.0일 경우 첫 번째 스크린의 좌표,
1.0이면 두 번째 스크린의 좌표
model - 모델뷰 매트릭스 (glGetDoublev 함수를 이용)
proj - 프로젝트 매트릭스 (glGetDoublev 함수를 이용)
view - 뷰포트 매트릭스 (glGetIntegerv 함수를 이용.)
objX, objY, objZ - 구한 좌표 값
```

그림 3. 변환된 두 점을 찾는 함수

3.2. 물리현상 표현

객체 충돌 표현에서 보여 지는 물리현상은 크게 두 가지가

있는데 이동과 충돌이다.

이동은 테이블위에서 퍽이 움직이는 현상이다. 퍽은 속도벡터의 방향과 크기에 의해서 이동하게 되는데 이때 테이블 위를 이동하면서 수식(2)와 같이 테이블과 퍽의 마찰력에 의해 속도가 줄어들게 된다.

$$V_2 = V_1 \times \mu \quad (V_1: \text{현재속도}, V_2: \text{나중속도}, \mu : \text{마찰계수}) \quad \text{②}$$

충돌은 먼저 객체와 객체의 위치를 확인하여 충돌여부를 파악해야한다. 충돌하였을 경우 각 객체의 상태에 따라 속도 벡터를 변화시킨다. 충돌은 크게 객체와 테이블 벽 사이, 객체와 객체 사이의 충돌이 있다.

1) 객체와 벽의 충돌

객체와 벽의 충돌 시 벽은 정지 상태이고 객체는 이동 방향을 바꾸게 된다. 이 경우에는 객체의 충돌 후 속도벡터만 구하면 된다. 충돌 후 속도벡터를 구하기 위해서 벽의 충돌 위치와 객체의 위치를 연결하는 축을 구한다. [그림 4]와 같이 이 축을 기준으로 하여 객체의 속도를 분할한다.

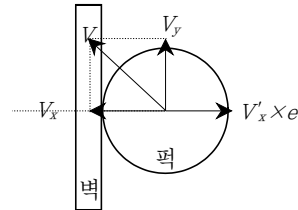


그림 4. 객체와 테이블 충돌

분할된 속도 중에 축과 평행한 속도벡터가 벽과의 충돌로 인해 방향이 반대방향이 된다. 충돌 후 반대 방향이 된 속도 벡터는 충돌 계수에 의하여 속도가 줄어들게 된다. 분할 후 계산한 속도를 다시 더하면 객체의 충돌 후 속도가 된다.

2) 움직이는 객체 간의 충돌

움직이는 객체끼리의 충돌 시에는 서로에게 힘을 가하게 되어 객체의 이동 방향이 바뀌게 된다. 객체끼리의 충돌여부 판단은 [그림 5]와 같이 객체의 중심위치간의 거리를 계산하여 두 객체의 반지름의 합보다 작을 경우 두 객체가 충돌했다고 판단한다. 충돌했으면 충돌후의 속도벡터를 계산하여 객체에 대입하면 객체는 충돌 후의 위치로 이동하게 될 것이다.

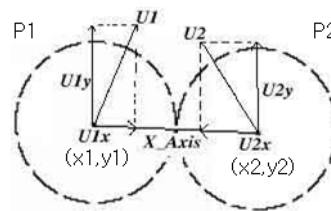


그림 5. 객체와 객체가 부딪친 경우

충돌 후의 속도벡터를 계산하기 위해서는 먼저 객체의 중심을 연결하는 축의 단위 벡터를 수식(3)과 같이 구한다.

$$X_{Axis} = (x_2 - x_1, y_2 - y_1) / (\text{두 점 사이의 거리}) \quad \text{⑧}$$

충돌하기 전의 속도벡터를 X_{Axis} 를 기준으로 분할을 한다. 객체의 중심을 이은 선을 x 축이라고 하면 객체의 속도벡터 U 이 x 축의 방향으로 어느 정도의 크기를 가지고 있는지 알기 위해 내적을 수식(4)와 같이 구한다[6].

$$U_{1x} = X_{Axis} \times (X_{Axis} \cdot U_1) \quad (4)$$

$$U_{1y} = U_1 - U_{1x}$$

$$U_{2x} = -X_{Axis} \times (-X_{Axis} \cdot U_2)$$

$$U_{2y} = U_2 - U_{2x}$$

크기를 구했으므로 x 축 단위 벡터를 곱하면 U_{1x} 가 구해지게 된다. U_{1y} 는 U_1 과 U_{1x} 의 차이를 계산하여 구할 수 있다.

분할한 속도 벡터와 객체의 질량을 이용하여 충돌후의 속도 벡터를 수식(5)와 같이 계산한다. 분할된 속도벡터 U_{1x} 와 U_{2x} 가 서로 충돌하였기 때문에 에너지 보존의 법칙과 운동량의 법칙에 의해서 수식(5)와 같이 계산되어지고 U_{1y} 와 U_{2y} 는 외부로부터의 힘의 변화를 받지 않았기 때문에 충돌 후에도 속도는 같다. 결론적으로 객체의 충돌 후의 속도 벡터는 $V_{1x}+V_{1y}$ 가 된다.

$$V_{1x} = \frac{((M1 - e \times M2) \times U_{1x}) + ((1 + e) \times M2 \times U_{2x})}{M1 + M2} \quad (5)$$

$$V_{1y} = U_{1y}$$

$$V_{2x} = \frac{((M2 - e \times M1) \times U_{2x}) + ((1 + e) \times M1 \times U_{1x})}{M1 + M2}$$

$$V_{2y} = U_{2y}$$

(e : 충돌계수, $M1, M2$: 질량)

$$V1 = V_{1x} + V_{1y}$$

$$V2 = V_{2x} + V_{2y}$$

충돌 후의 속도벡터를 적용시키기 전에 객체사이의 위치를 조정해야 한다. 충돌 시 객체의 위치는 서로 겹쳐진 상태로 위치하게 된다. 이 위치에서 충돌후의 속도벡터를 적용하여 움직이면 꺾이 겹쳐진 상태를 벗어나지 못하여 다시 충돌했다고 인식하는 경우가 발생한다. 그래서 충돌후의 속도를 적용시키기 전에 꺾이 위치를 겹치지 않게 조정한다.

3.3. 다중 사용자 환경 구축

네트워크를 통해 다중 사용자가 상호작용[7]을 하기 위해 본 논문에서는 각 단말기에서 데이터의 주고, 받음을 [그림 6]과 같이 처리한다.

1번 시스템은 물리엔진이 구동하고, 2번 시스템은 물리엔진이 구동하지 않는다. 두 시스템이 각자 물리엔진을 구동 시킬 경우 두 시스템의 데이터가 서로 달라지는 현상을 막기 위해서 한쪽의 시스템만 물리엔진을 구동시킨다. 테이블 탑 디스플레이를 통해 사용자로부터 입력을 받으면 각각의 시스템은 입력 정보를 이용하여 객체를 생성 및 설정한다. 2번 시스템은 물리엔진이 구동하지 않기 때문에 객체의 위치정보 데이터를 1번 시스템에게 전송한다. 1번 시스템은 사용자의 입력의해 만들어진 객체와 네트워크를 통해 전송받은 객체, 그리고 가상공간 내의 객체 데이터를 이용해 물리엔진을 구동한다. 물리엔진을 통한 계산 결과는 2번 시스템에게 전송되고, 화면에 표현된다. 2번 시스템은 1번 시스템에게 받은 데이터를 화면에 표현한다.

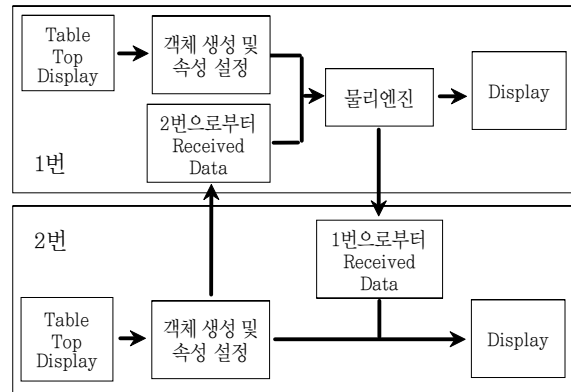


그림 6. 네트워크 기반 다중 사용자 시스템 구성도

4. 객체 컨트롤 장치

테이블 탑 디스플레이를 손가락 이용하여 터치하는 경우 마찰력 발생하여 객체의 움직임이 원활하지 못하게 된다. 본 논문에서는 마찰력을 최소한으로 줄여 움직임을 부드럽게 할 수 있는 LED기반 객체 컨트롤 장치를 제안한다.

4.1 적외선 LED 선택

테이블 탑 디스플레이는 적외선 빛을 감지하여 인식하기 때문에 적외선 LED를 입력 장치에 부착하여 테이블 탑 디스플레이가 인식할 수 있도록 하고, 입력장치의 표면에 마찰력이 적은 소재를 붙여 부드럽게 움직일 수 있도록 한다. 입력장치를 만들기 위해 적합한 적외선 LED의 선택이 요구된다. 적외선 LED의 빛이 옆으로 많이 퍼질 경우 테이블 탑 디스플레이에서 넓은 영역을 비추게 되어 세밀한 부분의 선택이 어려울 수 있다. 따라서 LED의 빛이 옆으로 많이 퍼지지 않고 직진성이 강해야 한다. 본 논문에서는 적외선 LED중에서 EL-1KL3를 사용한다. EL-1KL3는 [그림 7]과

같이 직진성이 강하기 때문에 입력장치의 적외선 LED로 사용하기에 적합하다.

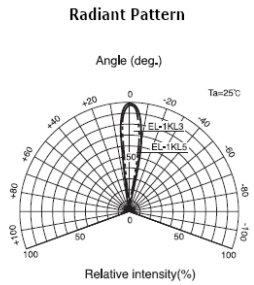


그림 7. 적외선 LED 빛의 퍼짐

4.2 적외선 LED 회로 구현

본 논문에서는 [그림 8]과 같이 전원, LED다이오드 저항으로 간단한 적외선 LED가 적외선을 발생시킬 수 있는 회로를 구현하였다.

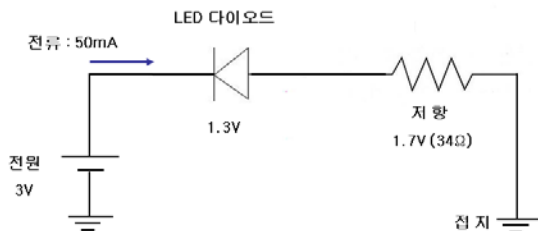


그림 8. 적외선 LED 회로도

전원의 전압은 일반 건전지로 만들 수 있는 3V로 설정한다. EL-1KL3은 [그림 7]과 같이 적외선이 $\pm 10^\circ$ 의 범위에서 빛이 나간다. 따라서 적외선LED에 흐르는 전류는 [그림 9]의 Radiant intensity와 Forward Current관계 그래프를 통해 알 수 있는 가장 적절한 값 50mA로 설정한다. [그림 10]의 Forward current와 Forward Voltage 관계 그래프를 참고하면 적외선 LED에 흐르는 전류가 50mA 인 경우 전압은 약 1.3V가 되는 것을 알 수 있다.

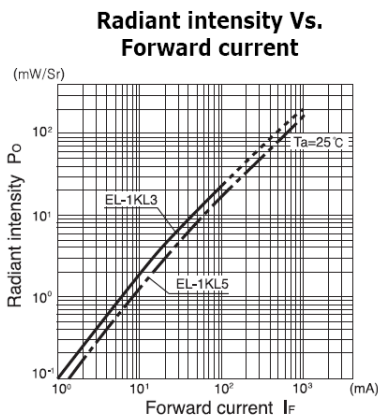


그림 9. Radiant intensity Vs. Forward current 그래프

Forward current Vs. Forward voltage

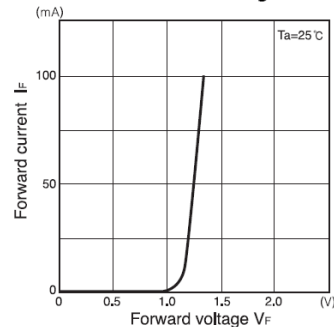


그림 10. Forward current Vs. Forward Voltage 그래프

전원, 저항, 적외선 LED는 직렬로 연결되어 있기 때문에 적외선LED에 1.3V가 걸리게 하기 위해서 저항은 34Ω이 되어야 한다.

4.3 입력장치 구현

테이블 탑 디스플레이의 입력장치는 [그림 11]과 같은 형태로 구현한다. 입력장치의 내부에 위에서 만든 회로를 장착하고 회로를 작동시킬 스위치를 바닥에 설치하여 테이블 탑 디스플레이 위에 놓아 스위치가 On이 되도록 한다. 입력장치의 표면에 부드러운 천을 감싸 테이블 탑 디스플레이와의 마찰력을 줄여 부드럽게 움직이도록 한다[그림 12].

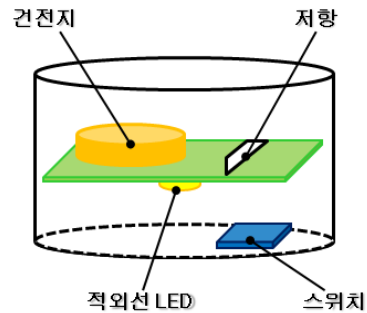


그림 11. 입력장치 구성도

입력 장치를 이용하여 테이블 탑 디스플레이에서 실행해본 결과 손으로 하는 것 보다 부드럽게 움직일 수 있었고, 실제 에어하키 게임과 비슷한 환경의 게임을 구현하게 되었다.

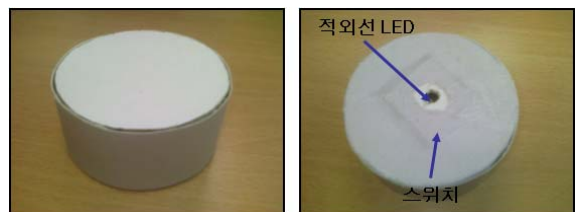


그림 12. 테이블 탑 디스플레이 입력장치

5. 결론

본 논문에서는 테이블 탁 평면상에서 객체의 선택을 위해 2차원 관점을 3차원으로 변환하였고, 객체의 이동과 객체와 객체의 충돌 시 나타나는 물리현상을 벡터연산을 통해 구현하였고, 이를 여러 대의 테이블 탁에서 이용할 수 있도록 네트워크를 설정하였다. 또한 테이블 탁 디스플레이가 인식할 수 있는 LED 기반 객체 컨트롤 장치를 제작하여 입력이 부드럽고 자연스럽게 하였다. 본 논문에서는 테이블 탁에서 이와 같은 기능들의 구현을 [그림 13] [그림 14]와 같이 에어하키 게임을 통해 보였다. 이 상호작용 물리엔진의 기대 효과는 실제와 같은 재미를 느낄 수 있으므로 흥미를 증가시킬 수 있었고, 테이블 탁 디스플레이의 인터페이스를 자연스럽게 받아들일 수 있었다는 것이다.

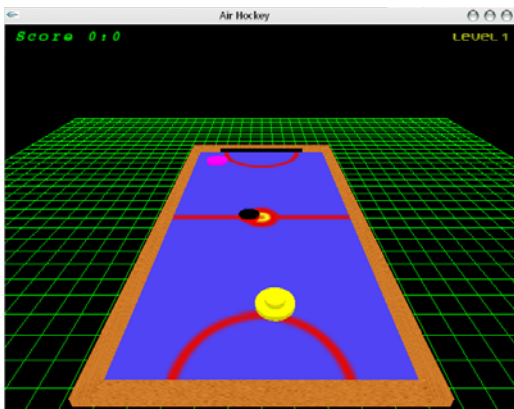


그림 13. 구현된 에어하키 화면

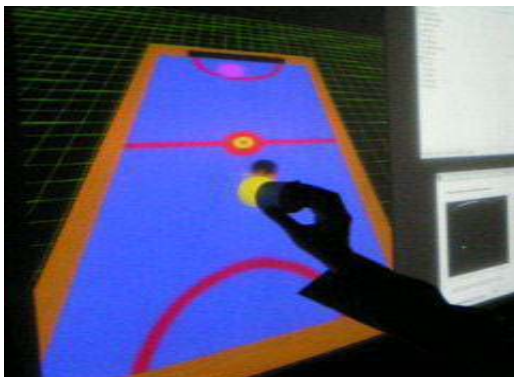


그림 14. 입력장치를 이용한 플레이

향후 연구 과제로는 실세계의 물리장치와 가상세계의 객체 충돌을 촉각적으로 느낄 수 있는 물리장치의 제작과 멀티 터치를 활용한 시스템 개발이 요구 된다.

참고문헌

- [1] 장한별, 김송국, 김장운, 이철우, “테이블 탁 디스플레이 기반 실감형 전자 요리 시스템”, 한국콘텐츠학회 종합학술대회 논문집 한국콘텐츠학회 2007 춘계 종합학술대회 논문집 제5권 제1호, 2007. 6, pp. 34-38.
- [2] 김송국, 이철우, “멀티터치를 위한 테이블-탁 디스플레이 기술 동향,” 한국 콘텐츠 학회 논문지 제7권 제2호, pp. 84-91, 2007.
- [3] 김송국, 이철우, “협력적인 상호작용을 위한 테이블 탁 디스플레이 기술 동향,” 한국콘텐츠학회 종합학술대회 논문집 한국콘텐츠학회 2006 추계 종합학술대회 논문집 제4권 제2호, 2006. 11, pp. 616-621.
- [4] Richard S. Wright, “OpenGL SuperBible,” 제3판, 정보문화사, 2005.
- [5] Dave Astle, Kevin Hawkins, Todor J. Fay, “Beginning OpenGL Game Programming,” Course Technology Ptr 2004.
- [6] <http://nehe.gamedev.net/>, NeHe Production
- [7] 김선우, “윈도우 네트워크 프로그래밍 : TCP/IP 소켓프로그래밍,” 한빛미디어, 2004.