

TangibleScreen

객체중심 프로젝션을 통한 상호작용성 향상

신선형*, 김정현
포항공과대학교 컴퓨터공학과
gkim@postech.ac.kr

TangibleScreen : Enhancing Interactivity through Object-centric Projection

Seonhyung Shin*, Joungyun Gerard, Kim
Virtual Reality Laboratory
Dept. of Computer Science and Engineering
Pohang Univ. of Science and Technology (POSTECH)

Abstract

Most interaction schemes in virtual environment are indirect in one way or another. In particular, without a haptic device (which introduces its own problems due to its cumbersomeness), users must rely on visual (or/and aural) feedback, and can not directly appreciate the 3Dness of the interaction object even with stereoscopy. This causes a drop in object presence because people are used to, for instance, observing objects in one's hand, rotating and manipulating them with physical contact. To alleviate this problem, this paper proposes a hand-held cubic screen, named TangibleScreen, on which the appearance of the target interaction object is projected. We choose the Relief Texture Mapping as the rendering method to correctly generate the viewer dependent textures to be projected on the non-planar surfaces of the TangibleScreen.

1. 서론

대부분의 가상 환경 내에서의 상호작용 방법은 직접적이지 못한 면이 있다. 실생활에서의 상호작용이 촉감적인 느낌을 동반하는 것에 반하여 자연 제스처 인식, 포인팅 기법 등과 같은 가상 환경 내에서의 상호작용 방법은 사용자가 상호작용 하려는 대상 객체와 직접적으로 닿지 않기 때문에 허공에서 상호작용 하는 듯한 느낌을 받을 수 있다. 특히 촉감적인 피드백을 받을 수 있는 장비(Haptic Device)가 없는 대부분의 가상 환경에서, 사용자는 시각적인 피드백에만 의존하여 상호작용을 해야 한다. 이러한 경우 입체 영상(Stereoscopy)을 제공하는 등 높은 시각적인 효과를 내는 가상 환경에서도 사용자는 상호작용 하려는 대상 객체의 입체감을 느끼기 어렵다. 또한 실생활에서 사람들은 물리적인 접촉을 하면서 객체와 상호작용 하는 것에 익숙해져 있기

때문에, 이러한 문제는 가상 객체의 실재감(Presence)을 떨어뜨리는 원인이 된다.

본 연구는 일반적인 가상 환경 내에서의 상호작용 방법이 촉감적인 피드백을 제공하지 않기 때문에 사용자가 상호작용 하려는 대상 객체의 입체감을 느낄 수 없고, 따라서 가상 객체의 실재감이 떨어지는 문제를 완화시키는 방법에 대하여 논의한다.

프로젝터의 프로젝션 가능 범위 안에서 자유롭게 움직일 수 있는 사용자가 손에 들고 있는 소형 입체 스크린을 통하여 촉감적인 피드백을 받으면서 원하는 가상 객체를 올바른 영상으로 관찰할 수 있고 상호작용 할 수 있도록 한다. 또한 영상 기반 렌더링 방법을 사용하여 일정 범위의 프레임 레이트(Frame Rate) 유지가 가능하도록 한다. 이를 위해서는 효과적으로 디스플레이 할 수 있는 프로젝션 시스템 정의, 알맞은 영상 기반 렌더링 방법 선택, 적절한 스크린 모양 결정, 사용자 시점

움 고려하여 생성한 영상을 평면이 아닌 스크린에 바르게 디스플레이 할 수 있는 방법의 제안, 원하는 위치에 원하는 크기로 프로젝션 하는 문제 해결 등을 해주어야 한다.

제안된 방법을 통하여 사용자는 원하는 위치에서 가상 객체를 관찰할 수 있고, 시점에 맞는 영상을 볼 수 있으며, 손에 들고 있는 스크린을 통하여 촉각적인 피드백을 받으며 가상 객체의 실제감을 느끼면서 상호작용할 수 있다.

2. 관련 연구

2.1 평면이 아닌 면으로의 프로젝션

91년 Dorsey 등은 무대 디자인 측면에서 극장의 곡선 모양의 배경막에 프로젝션 되는 영상이 관객에게 왜곡 없이 올바르게 보이도록 하기 위하여 프로젝션 하려는 영상을 사전에 왜곡하는 방법을 제안하였다[1].

그러나 앞의 연구들은 움직이지 않는 고정된 위치에 있는 사용자를 가정하거나, 많은 계산량으로 인하여 실시간 처리가 어렵다는 제한을 가지고 있다. 이 같은 문제를 해결하고자 98년 Raskar 등은 움직이는 사용자가 올바른 결과를 볼 수 있도록 임의의 3차원 모양의 스크린에 프로젝션 하는 실시간 방법을 제안하였다[2]. 이 논문에서는 프로젝티브 텍스처 매핑(Projective Texture Mapping)을 이용한 2단계 렌더링을 사용하였다.

이 방법은 일반적인 3차원 모델 렌더링과 비교하였을 때, 첫번째 단계의 결과 영상을 텍스처 메모리로 옮기는 과정과 프로젝티브 텍스처 매핑을 한 스크린 모델을 렌더링 하는 과정이 추가적으로 필요하다. 그러나 프로젝티브 텍스처 매핑은 OpenGL[3]이 지원하는 그래픽스 표준으로 그래픽스 하드웨어의 지원을 받아 실행되므로 전통적인 텍스처 매핑과 비교하였을 때 추가 연산 시간이 필요하지 않다. 따라서 이 방법은 두 번째 단계의 텍스처가 붙여진 스크린의 기하 모델을 렌더링 하는 데 걸리는 시간 따라 추가 소요 비용이 결정된다. 다시 말해서 알고리즘 수행 시간이 스크린 기하 모델의 복잡도에 따라 크게 좌우된다는 문제점을 가지고 있다.

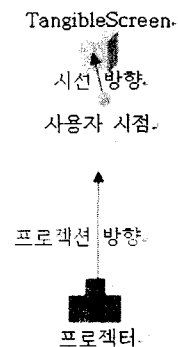
2.2 프로젝션을 이용한 가상/증강 현실 객체

프로젝션은 원하는 영상을 실제 물체의 표면에 직접 비추는 방법으로 증강 현실(Augmented Reality)에 많이 사용되어 지고 있다. Raskar는 앞에서 살펴본 2단계 렌더링을 이용하여 대상 가상 객체와 같은 모양의 실제 물체 표면에 가상 객체의 영상을 프로젝션 하고, 이를 이용하여 실제 물체와 상호작용 하고 있는 듯한 느낌을 받으면서 가상 객체와 상호작용 할 수 있는 방법을 제안하였다[4]. Kijima 등은 사용자가 소형 프로젝터를 휴대하는 개념의 Projection Head Mounted Display (PHMD)를 제안하고, 이를 이용하여 사용자가 움직이면서 원하는 위치에 영상을 프로젝션 하여 증강 현실을 구성하였다[5].

3. TangibleScreen

3.1 TangibleScreen의 개요

본 연구에서는 사용자가 휴대하는 소형 입체 스크린에 원하는 가상 객체의 영상을 프로젝션 하고, 이를 통하여 가상 객체를 관찰하고 직접적으로 상호 작용할 수 있는 인터페이스를 제안한다. 이 때 사용자가 휴대하는 소형 스크린을 TangibleScreen이라 명한다. TangibleScreen이 프로젝터의 프로젝션 범위 안에 있고, 프로젝터의 프로젝션 방향과 사용자의 시선 방향이 큰 차이가 나지 않을 경우 사용자는 손에 들고 있는 TangibleScreen을 통하여 원하는 가상 객체를 관찰하면서 자유롭게 이동하며 가상 환경을 둘러볼 수 있다.



[그림3-1] TangibleScreen의 개념도.

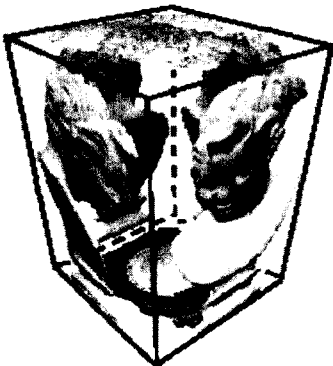
TangibleScreen에 프로젝션 해주는 영상은 사용자

시점을 고려하여 생성한다. 사용자는 기존의 2차원 평면으로 표현되던 가상 객체를 손 안에 들고 있는 TangibleScreen을 통하여 촉감적인 피드백을 느끼면서 관찰할 수 있다. 따라서 제안된 TangibleScreen은 이동 가능하고(Movable), 만져서 느낄 수 있는(Tangible), 직관적인(Intuitive) 인터페이스로 가상 환경에서의 상호작용성을 높일 수 있다.

3.2 TangibleScreen을 위한 영상 기반 렌더링 방법

Relief Texture Mapping은 [8] McMillan과 Bishop의 3차원 영상 와핑 방정식(3-D Image Warping Equation)[7]을 사전 와핑(Pre-warping) 과정과 일반적인 텍스처 매핑 과정으로 나누어 놓은 것이다. 와핑된 텍스처를 생성하고, 그래픽스 하드웨어의 지원을 받아 빠르게 텍스처 매핑을 함으로써 전체적인 와핑을 수행한다. 따라서 일반적인 3차원 영상 와핑 방법에 비교하였을 때, 같은 연산을 수행하는 것이지만 실행 시간을 단축시킨다는 장점을 지닌다.

Relief Texture Mapping은 영상 기반 객체(Image-based Object)를 위한 방법으로, 가상 환경 자체가 아닌 가상 환경을 이루고 있는 가상 객체들을 독립적으로 나타낼 수 있다. 따라서 Relief Texture Mapping을 이용하여 구축된 가상 환경은 영상 기반 렌더링의 장점인 일정한 렌더링 시간을 유지하면서, 3차원 기하 모델로 구성된 가상 환경과 같이 가상 객체와 쉽게 상호작용할 수 있다는 장점이 있다. 따라서, 본 논문에서는 Relief Texture Mapping을 기본 렌더링 방법으로 선택한다.



[그림3-2] 객체 둘러싸는 여섯 개의 Bounding 면[8].

3.3 Relief Texture Mapping

Relief Texture Mapping은 사전에 표현하고자 하는 가상 객체의 바운딩 박스(Bounding Box)를 구성하는 여섯 면에서 직교 투영(Orthogonal Projection)을 통해 영상 정보와 깊이 정보를 얻고, 변화하는 사용자 시점이 주어지면 바운딩 박스의 여섯 면에 붙일 새로운 텍스처를 계산하여 3차원 객체를 렌더링 하는 알고리즘이다 [8].

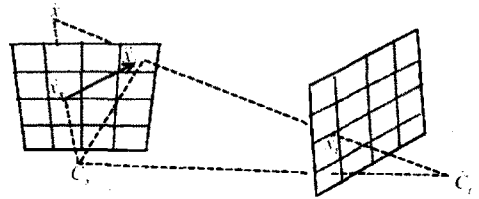
Relief Texture technique은 대응되는 3차원 점과 새로운 시점을 이용하여 와핑된 텍스처 좌표를 계산한다. 최종 연산식은 다음과 같이 표현된다[8].

$$u_i = \frac{u_s + k_1 displ(u_s, v_s)}{1 + k_3 displ(u_s, v_s)}$$

$$v_i = \frac{v_s + k_2 displ(u_s, v_s)}{1 + k_3 displ(u_s, v_s)}$$

그리고,

$$k_1 = \frac{\vec{f} \cdot (\vec{b} \times \vec{c})}{\vec{a} \cdot (\vec{b} \times \vec{c})}, k_2 = \frac{\vec{f} \cdot (\vec{c} \times \vec{a})}{\vec{a} \cdot (\vec{b} \times \vec{c})} \text{ and } k_3 = \frac{1}{\vec{c} \cdot \vec{f}}$$



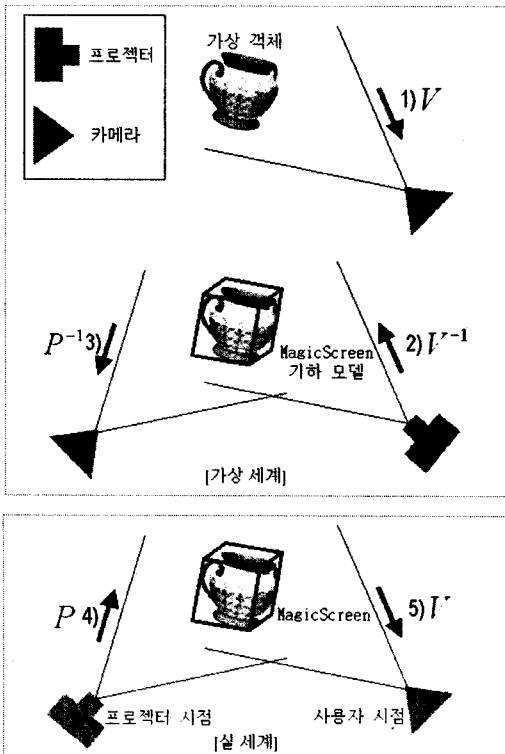
[그림3-3] Relief Texture Mapping 연산수행 과정[8].

육면체를 이루는 각 변의 길이는 가로, 세로, 높이 모두 최소한 사용자의 양안 사이의 거리 IPD(Interpupillary Distance) 보다 커야 한다. 그렇지 않을 경우, 사람은 양안을 통하여 물체를 보기 때문에 한 순간 TangibleScreen의 내 면이 보이는 문제가 생긴다.

3.4 렌더링 및 프로젝션

Relief Texture Mapping은 기본적으로 물체를 둘러싸는 육면체를 만들고, 여섯 개의 각 면에 붙여질 텍스처

를 계산한 후, 텍스처 매핑을 하는 방법이다. 따라서 기본 렌더링 방법으로 이 방법을 사용하면 텍스처 자체가 실제로 프로젝션 하기 원하는 스크린 모양인 육면체를 기준으로 계산되므로, 원하는 객체를 TangibleScreen에 프로젝션 하기 위해서 Relief Texture Mapping 수행 과정 이외의 추가 연산 과정은 필요하지 않다. 즉, Raskar의 연구처럼 사용자 시점에서 원하는 객체를 렌더링 한 다음, 이를 프로젝션 하기 원하는 스크린의 기하 모델에 다시 프로젝션(Back Projection) 시키는 단계를 거치지 않아도 된다. 단지 사용자 시점에서 Relief Texture Mapping을 이용하여 새로운 텍스처를 계산하고, 이 결과를 프로젝터의 시점에서 렌더링 하여 주면 된다. 따라서 Raskar의 방법과 비교하였을 때 두 번의 렌더링이 아닌 한 번의 렌더링으로 사용자 시점을 고려한 올바른 영상을 생성할 수 있다. 이를 [그림3-4]를 통해 다시 설명하였다.



[그림3-4] 사용자 시점과 프로젝터의 위치가 다른 경우, 사용자의 시점에 맞는 올바른 영상 생성하기.

그림에서 V 는 관측 변환(Viewing Transformation)을, P 는 프로젝션 변환(Projection Transformation)을 나타낸다. 그림에서 볼 수 있는 것과 같이 Raskar의 경우 1) 가상 객체를 트래킹되는 사용자 시점에서 렌더링 한 후, 2) 그 결과 영상을 사용하여 스크린의 기하 모델에 재 프로젝션 하는 과정을 거친다. 그러나 Relief Texture Mapping을 사용하면 1), 2)의 과정이 Relief Texture Mapping으로 대체될 수 있다. 그 후 3) 프로젝터의 시점에서 가상 객체를 렌더링 하고, 4) 생성된 영상을 프로젝터를 통해 프로젝션을 한다. 5) 최종적으로 프로젝션 된 결과를 사용자가 보면 같은 결과 영상을 얻을 수 있다.

본 논문에서 제안한 방법은 단순히 Relief Texture Mapping 과정을 거쳐 텍스처가 입혀진 객체의 바운딩 박스를 렌더링 하는 것으로 항상 일정한 렌더링 시간을 보장할 수 있다.

3.5 가상 객체의 관찰

본 논문에서 제안한 TangibleScreen은 이동이 가능한 인터페이스(Movable Interface)로, 사용자는 넓은 가상 환경에서 자유롭게 이동하며 원하는 위치에서 손에 들고 있는 TangibleScreen을 통하여 가상 객체를 쉽게 관찰할 수 있다.

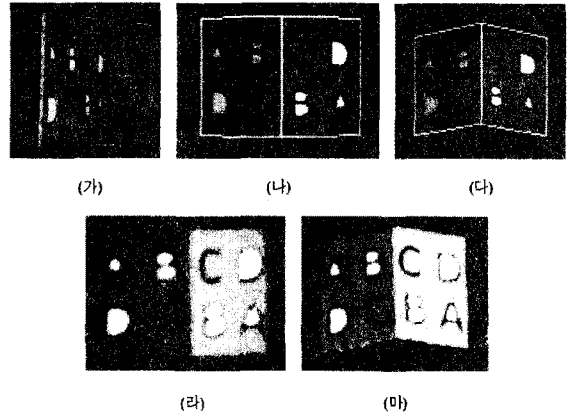
TangibleScreen은 자연스럽게(Natural) 직관적으로(Intuitive) 상호작용 할 수 TangibleScreen의 위치와 방향을 트래킹 하여 변화하면, 그 변화만큼 대응되는 가상 객체의 위치와 방향을 조정하여 렌더링 한다. 이렇게 함으로써 사용자는 가상 객체를 손에 직접 들고 상호작용 하는 듯한 효과를 얻을 수 있다. 따라서 좀 더 자연스럽게 직관적인 상호작용이 이루어질 수 있다. 또한 자연 체스처 인식이나 [6]에서와 같은 포인팅(Pointing)을 이용한 가상 환경에서의 상호작용 형태들은 아무런 촉감적인 피드백(Tactile Feedback)이 없다. 이는 허공에서 상호작용 한다는 느낌을 사용자에게 주어 실재감(Presence)이나 몰입감(Immersion)을 떨어뜨리는 원인이 된다. 그러나 본 연구에서 제안한 TangibleScreen은 손에 직접 들고 있는 물체를 통하여 가상 객체와 상호작용 할 수 있어, 만져서 느낄 수 있는 인터페이스(Tangible Interface)로서의 역할도 수행한다.

3.6 결과

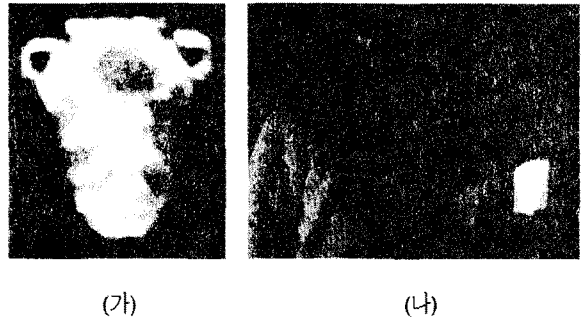
앞에서 제시한 방법들을 사용하여 TangibleScreen의 기하 모델을 결정하고, 원하는 가상 객체를 렌더링 한 후, 그 결과를 TangibleScreen에 프로젝션 하였다.

일반적인 성인의 경우 양안 사이의 거리는 5.8cm에서 7.8cm이다. TangibleScreen의 한 변의 길이가 최소한 사람의 양안 사이의 간격보다는 넓어야 한다는 조건과 사용자가 손에 무리 없이 쥘 수 있는 크기를 고려하여 한 변의 길이가 12cm인 정육면체로 실제 TangibleScreen을 제작하였다[그림 3-5](가).

앞에서 제시된 알고리즘에 따라 사용자 시점에 맞는 영상을 생성하여 TangibleScreen에 프로젝션 한 결과를 [그림 3-6]에 나타내었다. (가)는 정육면체의 사선으로 알파벳 문자가 평행으로 쓰인 3차원 모델이다. (나)는 (가)의 3차원 모델을 y 축 방향으로 -45도 회전한 후 사용자 시점을 고려하여 Relief Texture Mapping 하여 나온 결과 영상이고, (다)는 (나)를 프로젝터의 시점에서 바라 본 영상이다. (라)는 (다)의 결과가 프로젝터를 통하여 실제 TangibleScreen에 프로젝션 된 결과를 나타낸다. (나)와 같이 평행한 알파벳 문자를 볼 수 있다. 반면에 (마)는 (다)의 영상을 TangibleScreen이 아닌 평면 스크린에 프로젝션 시킨 결과로, 육면체를 기준으로 텍스처를 생성하는 Relief Texture Mapping으로 생성된 결과이기 때문에 평면에 프로젝션 할 경우 올바른 결과 영상을 볼 수 없다는 사실을 보여준다.



[그림3-6] TangibleScreen에 프로젝션 된 결과.

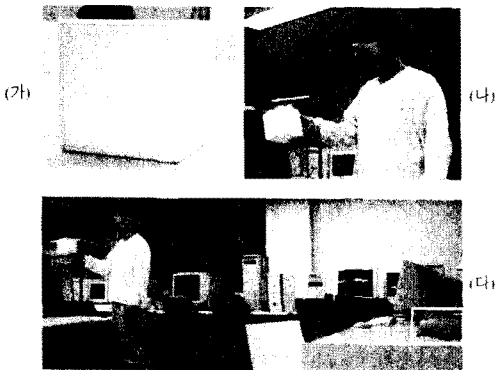


[그림 3-7] 다른 가상 객체의 영상이 TangibleScreen에 프로젝션 된 결과: (가) TangibleScreen에 영상이 프로젝션 된 모습, (나) TangibleScreen을 통하여 가상 객체를 관찰하는 모습.

4. TangibleScreen의 개선

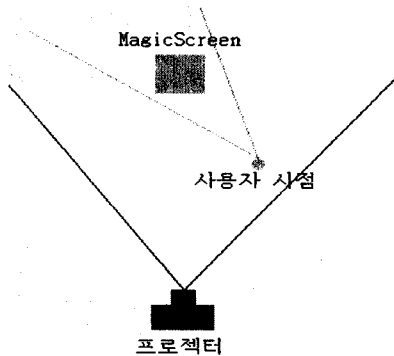
4.1 TangibleScreen의 한계점

TangibleScreen은 프로젝터에서 보이는 면과, 사용자에게 보이는 면이 동일해야 유효하다. 여기서 유효하다는 것은 사용자가 제대로 된 결과를 볼 수 있다는 것을 의미한다. 예를 들어 [그림 4-1]의 경우 프로젝터에서는 TangibleScreen의 정면만 보이지만, 사용자에게는 정면과 오른쪽 면이 보인다. 이와 같이 프로젝터에서 보이는 면과 사용자에게 보이는 면이 다를 경우, 사용자에게 보이는 일부 면이 프로젝션 되는 영상을 받지 못하므로 제대로 된 결과 영상을 볼 수 없다.



[그림 3-5] TangibleScreen 시스템: (가) TangibleScreen. (나) 사용자가 TangibleScreen을 통하여 가상 객체를 관찰하는 모습. (다) 전체적인 시스템 모습 (사각형은 프로젝터의 위치를 가리킨다).

따라서 TangibleScreen 시스템 안에서 사용자는 위와 같은 문제가 일어나지 않는 특정 범위, 즉 프로젝터에서 보이는 면과 사용자에게 보이는 면이 동일한 범위 안에서만 제대로 된 결과를 볼 수 있다는 제한이 있다. 결국 사용자는 항상 프로젝터와 가까운 일정한 범위 내에 있어야 한다. 이러한 한계를 해결하기 위하여 두 가지 해결책을 제시하였다.



[그림 4-1] 프로젝션 되는 면과 사용자에게 보이는 면의 차이.

4.2 TangibleScreen 가이드라인(Guideline) 제시

4.2.1 가시성 테스트(Visibility Test)

먼저 TangibleScreen이 유효한 지, 아닌 지를 판단하기 위해 [알고리즘 4-1]을 사용하여 프로젝터와 사용자를 기준으로 TangibleScreen 각 면의 가시성 테스트를 한다.

TangibleScreen의 각 면에 대하여 다음의 과정을 반복한다.

1. 면의 중점 C 를 구한다.
2. 시점 E 에서 면의 중점 C 로 향하는 벡터 V 를 구한다.
3. 면의 법선 벡터(Normal Vector) N 을 구한다.
4. If (V dot product $N < 0$) then Visible
두 벡터 V 와 N 사이의 각이 90도 보다 크고 270보다 작으면, 그 면은 보이는 면이다.

[알고리즘 4-1] 가시성 테스트.

이 때 프로젝터에서 보이는 면과 사용자에게 보이는 면을 비교하여, 사용자에게 보이는 모든 면이 프로젝터

에서도 모두 보이면 TangibleScreen은 유효하다. 그렇지 않을 경우에는 TangibleScreen이 유효할 수 있도록 가이드라인을 제시한다.

4.2.2 가이드라인 알고리즘

TangibleScreen이 유효하도록 조정하는 방법에는 TangibleScreen의 위치나 방향을 변화시키거나, 사용자의 위치나 시선 방향을 바꾸는 방법이 있다. 즉 TangibleScreen의 위치, TangibleScreen의 방향, 사용자의 위치, 사용자의 시선 방향 등 네 개의 변수가 존재한다. 그러나 사용자의 시선 방향은 사용자의 위치와 TangibleScreen의 위치를 이용하여 나타낼 수 있으므로 고려 변수에서 제외하였다. 따라서 남은 3개의 변수를 사용하여 가이드라인 제시한다.

4.3 FrustumScreen

TangibleScreen의 한계를 극복하기 위해서 제안된 두 번째 방법은 TangibleScreen이 유효할 수 있는 범위를 넓히기 위하여, 프로젝션 받는 면의 수를 늘릴 수 있는 형태로 모양을 변화시키는 것이다. 이에 따라 본 논문에서는 TangibleScreen의 한 종류로서 사각뿔에서 수평 단면으로 윗부분을 잘라 내어 버린 사각뿔대의 형태를 가지는 FrustumScreen을 제안한다

가시성 테스트를 하여 TangibleScreen의 유효 여부를 판단한다.

If (TangibleScreen이 유효하지 않는가?)

If (사용자가 프로젝션 영역 밖에 있는가?)

 사용자를 프로젝션 영역 안으로 위치 시킨다.

Else

 For($i=vPlane$; $i>=0$ && !find; $i--$)

 // i : 현재 우선 순위

 // (최대값: $vPlane$ - 사용자에게 보이는 면의 수, 최소값: 0,

 {

 TangibleScreen의 위치나 방향 변화

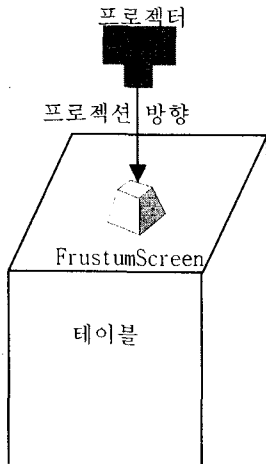
 If (우선 순위대로 i 개의 면이 유효한가?)

 find = true;

 }

[알고리즘 4-2] TangibleScreen 유효 범위 안내 알고리즘.

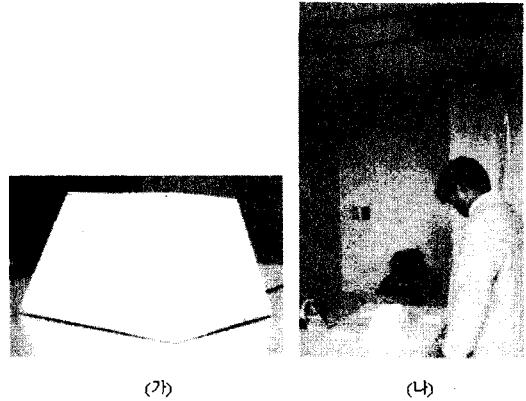
윗면과 옆면 사이의 각이 90°보다 크게 설계된 FrustumScreen은 일반적으로 TangibleScreen보다 더 많이 프로젝션 받을 수 있다. FrustumScreen이 최대의 효율을 얻을 수 있는 시스템 구조는 앞에서 예를 든 것과 같이 항상 밀면을 제외한 모든 면이 프로젝션 받을 수 있는 프로젝터와 FrustumScreen의 윗면이 서로 마주보는 형태이다. 따라서 이 논문에서는 [그림4-2]와 같은 테이블-탑(Table-top) 환경에 FrustumScreen을 적용하였다. 사용자는 테이블 주위를 자유롭게 이동하며 테이블 위에 올려진 FrustumScreen을 이용하여 TangibleScreen과 같은 방법으로 상호작용 할 수 있다. 이렇게 구성된 FrustumScreen 시스템은 FrustumScreen이 유효한 범위가 커질 뿐만 아니라, 사용자가 프로젝션 되는 빛을 가리는 경우도 적어지고, FrustumScreen의 이동 범위도 축소되므로 프로젝터로부터 나오는 원본 영상이 크게 확대되어 결과 영상의 화질이 떨어지는 현상도 줄어든다.



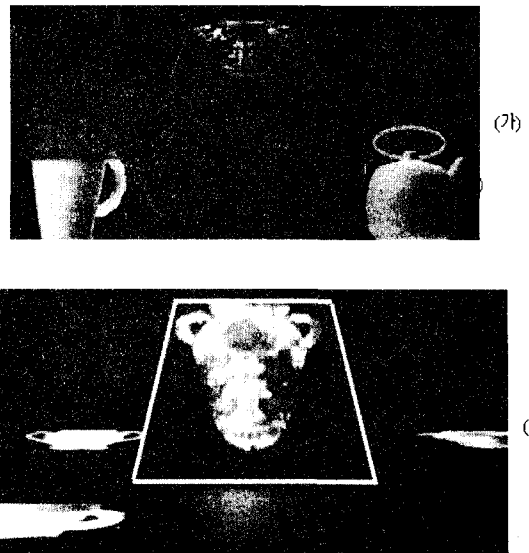
[그림4-2] FrustumScreen 시스템의 구조.

4.3.2 결과

FrustumScreen을 위한 Relief Texture Mapping을 제외하고는, 렌더링 및 프로젝션 등 모든 방법이 앞에서의 TangibleScreen과 동일하다. 프로젝터의 시야각을 이용하여 FrustumScreen을 제작하였고, 제작된 FrustumScreen을 적용할 수 있는 테이블-탑(Table-top) 환경을 구축하였다[그림 4-3].



[그림 4-3] FrustumScreen 시스템: (가) FrustumScreen, (나) 전체적인 시스템 모습.



[그림 4-4] FrustumScreen에 프로젝션 된 결과. (가) 프로젝터에서 바라 본 결과 영상, (나) 사용자 시점에서 바라 본 FrustumScreen에 프로젝션 된 결과. (FrustumScreen을 알아보기 쉽게 하기 위하여 외곽선을 그려 넣었다).

그림[4-4]는 생성된 영상을 FrustumScreen에 프로젝션 한 결과로, 프로젝터에서는 FrustumScreen의 옆면이 약간 납작하게 변형되어 보이지만, 이 영상이 FrustumScreen에 프로젝션 되면 사용자의 시점에서는 올바른 결과를 볼 수 있음을 알 수 있다.

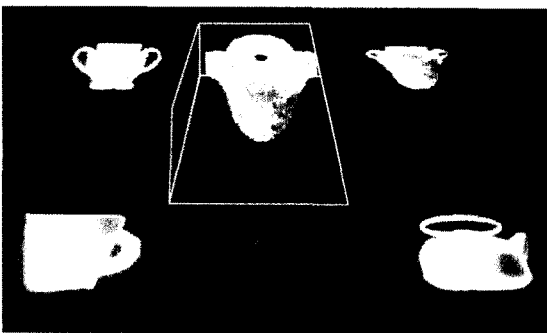
또한 FrustumScreen의 응용으로 가상 박물관을 구현해보았다. 테이블 위에 프로젝션 되는 2차원 영상 안에 놓여진 가상 객체 중의 하나를 사용자가 선택하면 FrustumScreen을 통하여 이를 관찰할 수 있다.

그 밖에 FrustumScreen의 여러 유용한 응용이 있을 수 있다. 예를 들어 여러 사람의 공동 작업을 위한 가상 테이블 환경에의 응용도 생각해 볼 수 있다. FrustumScreen의 밑면을 제외한 모든 면이 항상 프로젝션 되는 영상을 받을 수 있으므로, 스테레오 기능을 사용하여 각 사용자에게 맞는 영상을 보내주면, 어느 위치에 있는 사용자든지 보이는 모든 면은 프로젝션 받을 수 있으므로 FrustumScreen을 통하여 올바른 결과 영상을 볼 수 있다.

[그림 4-5] FrustumScreen을 이용한 가상 박물관. (FrustumScreen을 알아 보기 쉽게 하기 위하여 외곽선을 그려 넣었다).

5. 결론 및 향후 과제

본 연구에서는 촉감적인 피드백을 받으면서 실 생활에서와 같은 직관적인 방법으로 상호작용 할 수 있는, 사용자가 휴대하는 소형 입체 스크린인 TangibleScreen을 제안하였다. 제안된 TangibleScreen



시스템의 구현을 위해 적절한 프로젝션 시스템을 정의하고, 기존의 연구와 비교하여 이점을 가질 수 있는 Relief Texture Mapping을 기본 렌더링 방법으로 선택하였다. 사용자 시점에서 TangibleScreen을 통하여 올바른 결과 영상을 볼 수 있도록, 프로젝터의 위치와 다른 사용자 시점을 고려하여 생성한 영상을 평면이 아닌 스크린에 프로젝션 하는 문제를 해결하였다. 그 결과

사용자는 가상 환경을 이동하며 원하는 가상 객체를 TangibleScreen을 통하여 실 세계에서와 물체 관찰과 같은 직관적인 방법으로, 촉감적인 피드백을 느끼면서 관찰 할 수 있었다. 또한 TangibleScreen이 유효할 수 있는 범위가 제한적이라는 문제를 해결하기 위하여, TangibleScreen의 또 다른 종류로 프로젝터로부터 더 많은 면이 프로젝션 받을 수 있는 사각뿔대 형태인 FrustumScreen을 제안하였고, 그 결과 사용자의 자유로운 움직임에 대해 TangibleScreen보다 더 안정적인 결과를 얻을 수 있었다. 범위의 제한 없이 이러한 방법을 통하여 Tangible 상호작용을 하기 위하여서는 관련 연구에 언급되었던 Kijima 등의 들고 다니거나 HMD에 부착 할 수 있는 소형 프로젝터가 필요 할 것이며, 프로젝션의 방법은 이 논문에서 제안된 방법을 그대로 응용 할 수 있을 것이다. 본 논문에서 제안 된 방법은 실제 여러 가지 모양을 가진 객체의 추상적 Tangibility를 지향하고 있으므로, generic 한 육면체에만 적용 되었으나, 원통이나 그 밖의 복잡한 객체에도 쉽게 일반화가 가능하다.

향후 과제로는 사용자가 프로젝션 되는 영상을 가리는 문제를 해결할 수 있도록 여러 대의 프로젝터를 사용한 환경에 대한 적용도 시도해 볼 만 하다. 이 같은 경우, 사용자의 이동 가능 범위가 확대된다, 여러 대의 프로젝터에서 나오는 영상이 자연스럽게 이어지도록 하는 문제나, 영상의 밝기를 조정해 주는 것과 같은 문제에 대한 해결이 필요하다.

참고문헌

1. J. Dorsey, F. Sillion, and D. Greenberg, "Design and Simulation of Opera Lighting and Projection Effects," SIGGRAPH '91 Proceedings, July 1991.
2. R. Raskar, G. Welch, M. Cutts, A. Lake, L. Stesin, H. Fuchs, "The Office of the Future: A Unified Approach to Image-Based Modeling and Spatially Immersive Displays," SIGGRAPH '98 Proceedings, August 1998.
3. OpenGL, Available online: <http://www.opengl.org>.

4. R. Raskar, K. Low. "Interaction with Spatially Augmented Reality," AFRIGRAPH '2001 Proceedings, 2001.
5. R. Kijima, M. Hirose, "A Compound Virtual Environment Using the Projective Head Mounted Display," ICAT/VRST '95 Proceedings, 1995.
6. C. Kirstein, H. Muller, "Interaction with a Projection Screen using a Camera-tracked Laser Pointer," Proceedings of The International Conference on Multimedia Modeling, 1998.
7. Leonard McMillan Jr., "An Image-Based Approach to Three-Dimensional Computer Graphics," Ph.D. Dissertation, University of North Carolina at Chapel Hill, January 1997.
8. M. Oliveira, G. Bishop, D. McAllister, "Relief Texture Mapping", SIGGRAPH '2000 Proceedings, July 2000.