무안경식 다시점 입체 디스플레이를 위한 GPU기반 렌더링 기법

A GPU based Rendering Method for Multiple-view Autostereoscopic Display

안종길, Jong-gil Ahn*, 김진욱, Jinwook Kim**

요약 최근 3차원 입체 디스플레이에 대한 관심이 증가하고 있다. 무안경식 다시점 입체 디스플레이(Multiple-view Autostereoscopic Display)는 관찰자가 셔터 글래스(Shutter glasses)또는 HMD(Head Mounted Display)와 같은 특별한 장치를 착용하지 않고도 여러 시점(Multiple-view)에서 입체 영상을 볼 수 있어 가상현실(Virtual Reality), 모바일, 3D TV 등의 분야에서 각광 받고 있다. 하지만, 디스플레이 설계 시 정해진 거리와 시역 내에 관찰자가 위치해 있어야만 입체 영상을 볼 수 있다는 문제점이 있다.
본 연구에서는 무안경식 다시점 입체 디스플레이 시스템을 구축하여 이 시스템에서 관찰자의 시점 변경에도 실시간으로 자연스러운 3차원 입체 영상을 보여 주기 위한 효율적인 GPU기반 렌더링 기법을 제안한다.

Abstract 3D stereo display systems gain more interests recently. Multiple-view autostereoscopic display system enables observers to watch stereo image from multiple viewpoints not wearing specific devices such as shutter glasses or HMD. Therefore, the Multiple-view autostereoscopic display is being spotlighted in the field of virtual reality, mobile, 3D TV and so on. However, one of the critical disadvantages of the system is that observer can enjoy the system only in a small designated area where the system is designed to work properly.
This research provides an effective way of GPU based rendering technique to present seamless 3D stereo experiences from an arbitrary observer's view position.

핵심어: *Autostereoscopic Display System, Parallax Barrier System, GPU based Rendering Method, Multiple-view Autostereoscopic Display System*

본 논문은 문화체육관광부의 스포츠산업 기술개발사업(가상현실 기반 실감형 스포츠 시스템 개발)의 일환으로 수행되었음.

*주저자 : 한국과학기술연구원 영상미디어센터 학생연구원; e-mail : hide989@imrc.kist.re.kr

**교신저자 : 한국과학기술연구원 영상미디어센터 선임연구원; e-mail : jwkim@imrc.kist.re.kr

1. 서론

디스플레이 기술의 눈부신 발전, 컴퓨터 기술의 고도화, 단순 2차원 평면 영상이 아닌, 조금 더 실제에 가까운 3차원 입체영상을 보고 싶어 하는 사용자들의 요구 등에 힘입어 3차원 입체영상 기술에 대한 관심이 증가하고 있고 관련 연구도 활발히 이루어지고 있다.

3차원 입체영상 디스플레이 기술은 실제 물체의 산란 정보를 이용한 홀로 그래픽(Holographic) 방식과 양안 시차(Binocular Disparity)를 이용한 스테레오 스코픽(Stereoscopic) 방식으로 분류된다. 홀로 그래픽 방식은 물체에서 산란된 빛을 이용해 자연스러운 입체영상을 재현하는 궁극적인 3차원 입출력 방식이지만, 방대한 양의 3차원 정보 및 관련 소자의 한계로 현재는 실시간(Real-Time)의 구현이 불가능한 한계를 갖고 있다. 인간의 시각 시스템(HVS:Human Visual System)구조를 모방해 좌, 우 2개의 2차원 영상을 기반으로 3차원 영상을 입출력 해주는 스테레오 스코픽 방식은 현재의 기술로 구현 가능한 입체영상 기술이며, 이 기술은 3D TV와 같이 평면적인 디스플레이 장치에서 입체감을 생성하여 3차원을 표현하는데 적용되고 있다.

스테레오 스코픽 기술은 3차원 입체영상을 관찰하기 위해 관찰자가 특수한 안경을 착용해야 하는 안경식 입체영상 디스플레이와 특수한 안경을 착용하지 않고서도 3차원 입체영상을 관찰할 수 있는 무안경식 입체영상 디스플레이 방식으로 다시 나눌 수 있다. 안경식 3차원 입체영상 디스플레이는 입체영상을 보기 위해 관찰자가 셔터 글래스(Shutter glasses)나 HMD(Head Mounted Display)와 같은 특수한 장비를 착용해야 한다는 불편함을 갖고 있다. 또한, 가상환경(Virtual Environment)에서 다양한 장치들의 조작과 움직임을 통한 사용자 인터랙션(User Interaction)이 필요한 경우 셔터 글래스, HMD와 같은 장비들은 사용자 인터랙션에 제약을 줄 수 있다.

안경식 입체영상 디스플레이가 갖고 있는 이러한 문제의 개선을 위한 노력과 90년대 이후 평판 디스플레이 기술의 개발 및 발전으로 특수 장비를 사용하지 않고도 입체영상을 볼 수 있는 무안경식 입체영상 디스플레이(Autostereoscopic Display)[6][17]가 주목받고 있고, 이와 더불어 무안경식 입체영상 디스플레이라는 특수한 디스플레이 시스템 환경에서 실시간 응용 프로그램을 구동하기 위한 소프트웨어 관련 연구가 활발히 진행되고 있다.

하지만 무안경식 입체 디스플레이는 입체영상을 보기 위해 관찰자가 디스플레이 설계 시 정해진 시점에 위치해야만 하고, 무안경식 입체 디스플레이라고 하는 이 특수한 디스플레이 장치에서 응용 프로그램을 구동하고자 할 때, 이 특수한 디스플레이에 적합한 렌더링 알고리즘이 필요하다는 문제점이 있다.

본 논문에서는 무안경식 다시점 입체 디스플레이 시스템을 구축하고, 관찰자의 좌, 우 위치변화에도 자연스러운 입체영상을 볼 수 있는, 효율적이고 빠른 GPU기반 렌더링 기법에 대해 소개한다.

2. 관련연구

입체영상을 보기위해 셔터 글래스나 HMD와 같은 특수한 장비를 착용하는 것은 사용자에게 불편함을 야기 시켰고, 이러한 불편함을 해결하기 위한 노력으로 무안경식 입체 디스플레이가 주목받게 되었다. 특히 가상현실 분야에서 과거 CAVE(CAVE Automatic Virtual Environment)[7][16]와 같은 사용자 몰입형 가상현실 환경 시스템의 대체수단으로 주목받고 있다[2][3]. 이와 관련된 대표적 연구 활동으로 University of illinois at Chicago의 Electronic Visualization Lab에서는 35개의 무안경식 입체 디스플레이를 연결한 사용자 몰입형 가상현실 시스템을 소개하였다[11]. 또한 데스크탑 PC 사용자를 위한 무안경식 입체 디스플레이에 적외선 카메라 트래킹이 결합된 개인용 무안경식 입체 디스플레이 가상현실 시스템을 소개하였다[12]. 가상현실 분야 외에도 무안경식 입체 디스플레이는 최근, 3D TV[8], 휴대용 모바일 디스플레이 장치[9]로의 적용에 대한 연구도 활발히 이루어지고 있다.

그림 1. 무안경식 입체 디스플레이 기반 사용자 몰입형 가상환경 시스템

입체 디스플레이 시스템에 대한 연구뿐만 아니라, 무안경식 입체 디스플레이라는 특수한 하드웨어 환경에서의 실시간 렌더링에 관한 다양한 연구들 또한 활발히 진행되고 있다. Robert L. Koomia는 Autostereo Combiner address라고 하는 GPU기반 렌더링 알고리즘을 제안하였다[1]. Francois de Sorbier는 최신기술인 지오메트리 쉐이더[18]를 적용한 알고리즘을 제안하였다[10]. 이 알고리즘은 지오메트리 쉐이더를 이

용하여 지오메트리 데이터를 변형하는 대신, 시점영상을 만들어내기 위한 렌더링 횟수를 줄여 속도향상을 도모하였다.

3. 무안경식 다시점 입체 디스플레이 시스템

3.1 패럴랙스 베리어(Parallax Barrier) 방식

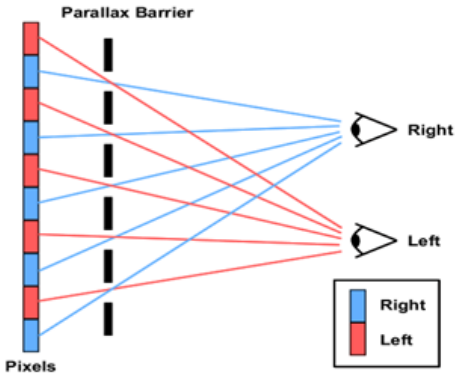

그림 2. 패럴랙스 베리어 방식

무안경식 입체 디스플레이는 입체 영상을 만들어내는 방식에 따라 렌티큘러 렌즈 방식과 패럴랙스 베리어 방식으로 나뉜다. 본 연구에서 사용된 패럴랙스 베리어 방식은 디스플레이 표면에 베리어(Barrier)라고 불리는 차폐막이 있어 이 차폐막이 현재 관찰자 위치에서 관찰자가 볼 수 있는 픽셀과 볼 수 없는 픽셀을 구분한다. 베리어에 의해 각각의 시점에서 관찰자 눈으로 서로 다른 픽셀들의 집합이 좌, 우 눈으로 들어오게 된다. 결국, 관찰자가 시역 내에서 입체영상을 보고 있다면, 이는 베리어 설계에 의해 베리어에 가려지지 않고 관찰자 두 눈으로 각각 들어와 입체영상을 만들어내는 픽셀들의 집합을 보고 있는 것이다.

본 연구에서는 기울어진 형태의 패럴랙스 베리어가 사용되었으며, 이 베리어에 의해 기울어진 서브픽셀 단위(Subpixel scale)의 R,G,B 채널이 하나의 픽셀로 관찰자 눈에 인식된다.

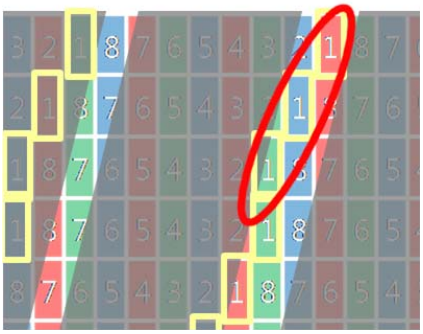

그림 3. 서브픽셀 단위의 픽셀구조

3.2 다시점(Multiple-view) 방식

다시점 방식은 특정 영역에서 입체영상을 볼 수 있는 방식으로 무안경식 입체 디스플레이 분야에서 가장 활발히 개발되고 있는 기술이다. 시점(Viewpoint)은 디스플레이 설계에 의해 정의된, 입체영상을 볼 수 있는 물리적 실제공간으로 다시점 입체 디스플레이에는 여러 개의 시점영상 정보를 포함하여 관찰자는 조금 더 넓은 시역 범위에서 입체영상을 볼 수 있게 된다. 시점의 수가 늘어날수록 관찰자가 입체영상을 볼 수 있는 시역의 폭이 넓어지지만, 최종 입체영상의 해상도가 늘어난 시점 수만큼 감소하는 문제점이 따른다.

본 연구에서는 총 8개의 시점영상 정보를 포함하는 8시점 입체 디스플레이를 사용하였다.

3.3 사용자 위치 추적

무안경식 입체 디스플레이는 디스플레이 설계 시 입체영상을 볼 수 있는 관찰자의 관찰 위치와 영역이 정해진다. 이는 관찰자가 고정된 위치에서만 입체영상을 봐야 하다는 문제점을 의미한다. 다시점 입체 디스플레이를 통해 입체 영상을 볼 수 있는 영역이 조금은 넓어졌지만, 이 역시 충분치 못하다. 관찰자가 이 영역을 벗어나는 경계 부분에 위치하게 되면 입체영상이 순간적으로 깨지게 되어 자연스럽지 않은 입체영상을 보게 된다.

본 연구에서는 관찰자의 위치를 실시간으로 알아내어 관찰자의 위치에 맞게 시점영상들을 실시간으로 재생성하여 관찰자가 설계 시 정해진 시역 범위 밖으로 이동하여도 자연스러운 입체 영상을 볼 수 있도록 하였다. 현재 관찰자의 위치를 실시간으로 얻기 위해 얼굴추적(Face Tracking)을 사용하였다.

3.4 시스템 구성

본 연구에서 사용된 기본적인 시스템 구성은 다음과 같다. 무안경식 입체 영상 디스플레이를 위해 V3I사의 8시점 패럴랙스 베리어 기반 입체 디스플레이 장치를 사용하였다. 이 장치는 관찰자가 디스플레이로부터 4m 떨어진 위치를 기준으로 40cm(8개의 시점 * 시점거리 5cm) 범위 시역 내에서 입체영상을 볼 수 있도록 설계되었다. 관찰자의 위치를 실시간으로 알아내기 위해 디스플레이 장치 상단에 카메라를 설치하였다. 본 연구에서는 POINT GREY의 FLEA2 카메라를 사용하여 디스플레이로부터 4m 떨어진 지점에 있는 관찰자의 얼굴을 추적하였고, 카메라 오차보정(Calibration) 과정을 거쳐 얻어진 관찰자의 위치정보는 평균오차 0.68cm, 최대오차 1.3cm로 잘 작동하였다. 실시간으로 시점영상을 만들고 조합하는 그래픽 카드는 NVIDIA Geforce 8800GTX를

사용하였다.

3차원 렌더링 구현을 위해 OpenGL을 사용하였고, GLSL(OpenGL Shader Language)[5]을 사용하여 쉐이더를 작성하였다. 관찰자 얼굴추적을 위해 상용 라이브러리 FaceAPI[4]를 사용하였다.

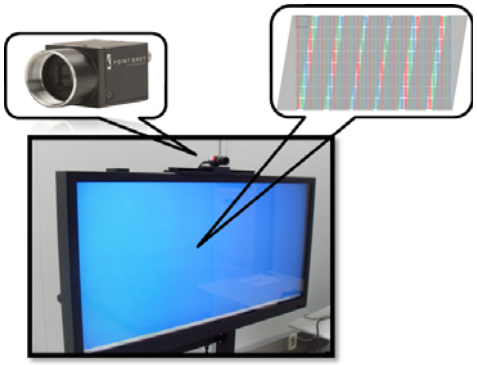

그림 4. 시스템 구성

그림 5. Wii Remote를 이용한 사용자 인터랙션

사용자 인터랙션으로 Wii Remote[13]를 이용한 사용자 인터랙션을 적용하였고, 인터랙티브 컨텐츠를 위해 실시간 물리 시뮬레이션 라이브러리인 Virtual Physics[14]가 사용되었다.

사용자 인터랙션으로 Wii Remote[13]를 이용한 사용자 인터랙션을 적용하였고, 인터랙티브 컨텐츠를 위해 실시간 물리 시뮬레이션 라이브러리인 Virtual Physics[14]가 사용되었다.

4. GPU기반 렌더링 기법

무안경식 다시점 입체 디스플레이는 서브픽셀 단위의 접근이 필요하여 고정 파이프라인을 사용할 수 없다. 따라서 이 하드웨어적 특성에 맞는 렌더링 기법이 필요하며, 이 렌더링 기법은 빠르고 안정적으로 입체영상을 보여줄 수 있어야한다.

지금부터 이러한 무안경식 다시점 입체 디스플레이 장치의 특수성에 부합하면서, 빠르고 안정적으로 입체영상을 보여주는 GPU기반 렌더링 기법에 대해 제안한다.

4.1 실제환경과 가상환경의 일치

관찰자가 위치한 실제 환경과 정확하게 일치하는 3차원 가상환경을 만든다. 정확하게 일치하는 환경을 만들기 위해 단위(Unit)를 일치시킨다. 예를 들어 실제 환경에서 관찰자 위치와 디스플레이간 거리가 4m일 경우 가상환경에서의 가상 카메라와 디스플레이 오브젝트간 거리도 4m로 일치시켜준다. 디스플레이와 관찰자간 거리 외에도 시점간격, 시점들이 모여 이루게 되는 시역, 입체 디스플레이의 가로, 세로 크기도 동일하게 일치시킨다.

본 연구에서는 미리 미터(mm)단위로 통일하였다. 실제 환경에서 디스플레이와 관찰자간 거리는 4m, 디스플레이의 가로 길이는 94cm, 세로 길이는 53cm, 시점간격은 5cm이다. 따라서 가상환경에서 디스플레이와 관찰자간 거리는 4000, 디스플레이 가로 길이는 940, 세로 길이는 530, 시점간격은 50으로 하였다.

그림 6. 관찰자의 실제 환경

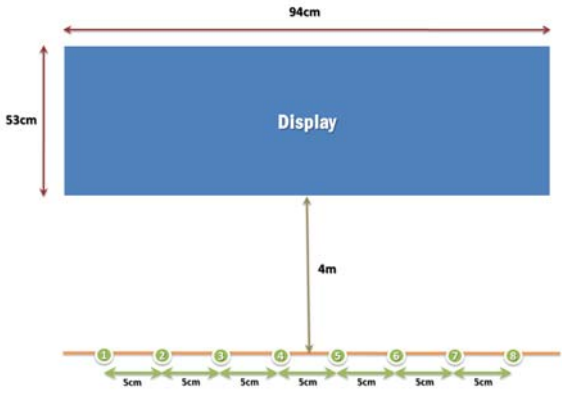

그림 7. 실제 환경과 동일한 가상 3차원 환경

4.2 디스플레이 픽셀 정보 저장

무안경식 다시점 입체 디스플레이는 관찰자의 위치에 따라 보이는 픽셀 분포가 달라진다. 이는 디스플레이 설계 시 결정된다. 따라서 최종 화면에 입체영상을 렌더링하기 위해서

는 입체영상 장치의 픽셀구조를 알고 있어야 한다. 본 연구에서는 이 픽셀구조의 정보를 디스플레이 장치 해상도와 동일한 크기의 이미지(본 연구에서는 인덱스 이미지라는 이름으로 정의)에 기록하였다. 본 연구에서 사용된 디스플레이 장치가 8시점 입체 디스플레이 이므로, 모든 서브픽셀에는 각각 0부터 7사이의 값이 기록된다. 기록되는 0부터 7까지의 숫자는 시점영상 정보이며, 이는 반복적인 특징이 있다. 시점영상 정보가 기록된 이 인덱스 이미지는 렌더링의 맨 마지막 단계인 시점영상 조합단계에서 사용된다.

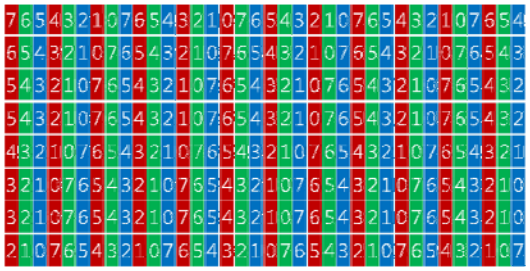

그림 8. 디스플레이 장치의 픽셀구조

4.3 시점영상 생성

다시점 입체 디스플레이 장치는 시점영상 정보의 조합을 통해 입체영상을 만들어 낸다. 이를 위해 각각의 시점에서 시점영상을 생성해야 한다. 시점영상이란 물리적 실제 공간인 시점에서 현재 바라보는 영상을 의미한다. 본 연구에서 사용된 다시점 입체 디스플레이는 8시점 입체 디스플레이로, 관찰자의 위치를 중심으로 8개의 시점영상(좌, 우 각각 4개)을 생성해야 한다. 이를 위해 가상의 3차원 공간에서 디스플레이 설계 시 정의된 시점의 위치에 가상의 카메라를 놓고 시점을 바꾸어가며 현재 화면에서 보여 지는 장면을 저장한다.

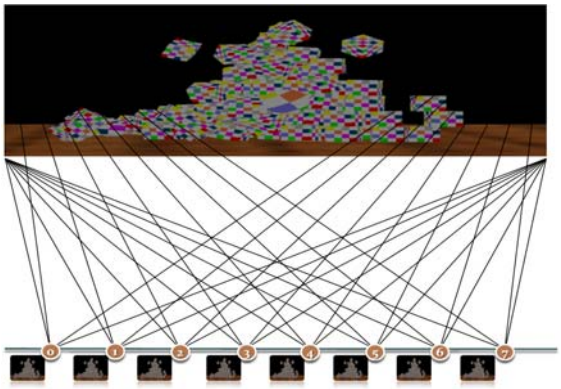

그림 9. 각각의 시점에서 시점영상 생성

시점영상을 얻기 위해 각각의 시점에서 현재 보여 지는 장면을 렌더링하게 되는데, 렌더링 된 시점영상은 FBO(Framebuffer Object)[15]에 의해 바인드 된, 비디오 메모리의 텍스쳐(Texture)에 저장된다. 비디오 메모리에 시점영상을 저장함으로써 속도 향상과 시점영상 조합의 편리함을 얻을 수 있다.

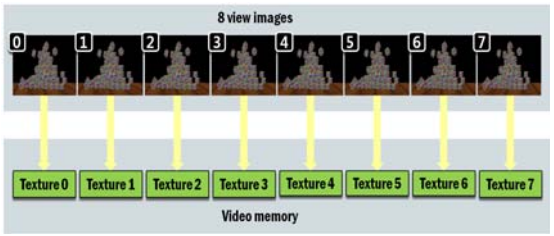

그림 10. 비디오 메모리에 시점영상 저장

4.3 시점영상 조합

시점영상 생성 단계를 통해 저장된 8개의 시점영상은 프래그먼트 쉐이더(Fragment Shader)에서 최종적으로 보여 지는 하나의 입체영상으로 조합된다. 시점영상의 조합은 디스플레이의 픽셀구조와 동일해야하므로, 이를 위해 디스플레이 픽셀구조 저장 단계에서 미리 저장해 놓은 인덱스 이미지를 사용하여 디스플레이 픽셀구조와 동일하게 조합한다.

프래그먼트 쉐이더에서 8개의 시점영상과 인덱스 이미지를 불러와 현재 픽셀의 시점정보를 인덱스 이미지로부터 확인한다. 인덱스 이미지에 기록된 시점정보는 서브픽셀 단위로 기록되어 있으므로 서브픽셀 단위로 시점정보를 확인하고, 이에 해당하는 시점영상의 서브픽셀 값을 가져와 출력하게 된다. 이 과정은 디스플레이 해상도에 맞게 수행되고, 디스플레이 픽셀구조에 맞게 조합된 영상은 최종적으로 실제 디스플레이 화면에 뿌려져 입체영상으로 보여 지게 된다.

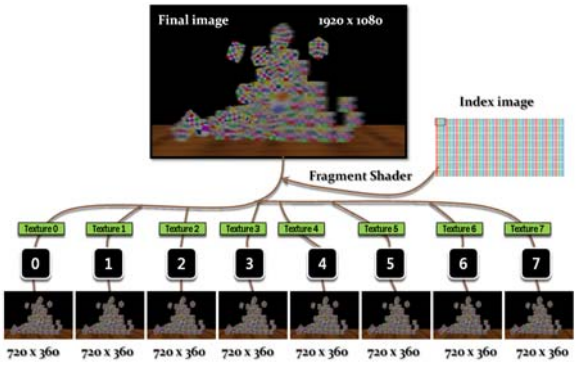

그림 11. 프래그먼트 쉐이더를 이용한 시점영상 조합

프래그먼트 쉐이더의 코드는 그림12와 같다. pixelFormat은 인덱스 이미지, 텍스쳐 배열로 선언된 view는 시점영상 배열을 의미한다. 텍스쳐에 저장된 인덱스 이미지(소스 코드 상의 pixelFormat)의 픽셀 값을 읽어 texID에 넣는다. 인덱스 이미지는 서브픽셀마다 시점정보가 기록되어 있으므로 texID에는 서브픽셀 R,G,B에 각각 시정정보들이 들어가게 된다. col[9]에는 각각의 시점영상의 현재 픽셀 값이 들어가게 되고, 최종적으로 화면에 출력될 gl_FragColor에는 인덱

스 이미지로부터 얻은 시점정보에 맞게 시점영상의 해당 픽셀 값이 들어가게 된다.

```
#extension EXT_gpu_shader4 : enable

uniform isampler2D      pixelFormat;
uniform sampler2DArray  view;

void main()
{
    ivec4 texID = texture2D(pixelFormat, gl_TexCoord[0].st);

    vec4 col[9];

    for ( int i = 0; i < 9; i++ )
    {
        col[i] = texture2DArray(view, vec3(gl_TexCoord[0].st, i));
    }

    gl_FragColor = vec4(col[texID.r].r, col[texID.g].g, col[texID.b].b, 1.0);
}
```

그림 12. 시점영상 조합을 위한 프래그먼트 쉐이더 코드

6. 관찰자 위치변화에 따른 시역형성 방법

관찰자 위치변화에 의해 시점변화가 발생 할 경우, 시역 내에서의 관찰자 움직임에는 문제가 없으나, 관찰자의 위치가 시역을 벗어날 때, 순간적으로 입체영상이 깨지는 문제가 발생한다.

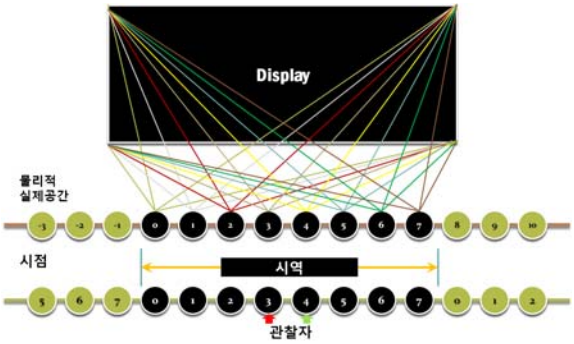

그림 13. 관찰자 위치 중심의 시역 형성

그림 13은 설계 시 정의된 시역 범위 안에 관찰자가 있는 경우이다. 관찰자가 위치하고 있는 물리적 실제 공간과 실제 공간에서 정의 된 시점의 위치들을 보기 쉽게 하기 위해 구역으로 나누어 번호를 표기하였다. 이 경우 관찰자는 디스플레이를 바라보고 있는 실제 공간의 0부터 7까지 범위 내에서 자연스러운 입체영상을 보게 된다. 하지만 관찰자 눈의 위치가 실제 공간에서 0보다 작아지는 순간, 또는 7보다 커지는 순간 입체영상이 깨지게 된다. 이는 관찰자가 위치한 실제공간에서 바라보는 영상과 획득한 시점영상의 차이 때문에 생기는 문제이다.

관찰자가 시역 범위를 벗어날 경우 생기는 이러한 문제는 관찰자의 위치를 중심으로 시점을 실시간 재구성 해 줌으로써 해결된다. 우리는 얼굴추적을 통해 현재 관찰자의 위치를 알 수 있고, 디스플레이의 픽셀구조와 픽셀구조가 반복된다는 성질을 알기 때문에 관찰자 위치 변화에도 새로 생성해야 할 시점영상 정보를 알아 낼 수 있다.

관찰자의 위치가 변경 되었을 때, 변경된 현재 위치에서 생성해야 할 시점들을 재구성한다. 앞서 언급했듯이 본 연구에서 사용된 디스플레이의 시점 간격이 5cm이므로 관찰자의 위치가 5cm변할 때마다 시점변경이 이루어진다. 시점들은 관찰자의 현재위치를 기준으로 항상 좌, 우 동일하게 생성한다. 이는 관찰자가 있는 현재의 위치보다 좌, 우측으로 동일하게 시점을 유지하여 관찰자가 시역을 벗어나는 순간이 없도록 유도하는 것이다. 시역이 유지된 상태에서 관찰자의 위치변화에 따르도록 한다. 그리고 이 시점들의 물리적 실제 공간에서 시점영상을 생성한다.

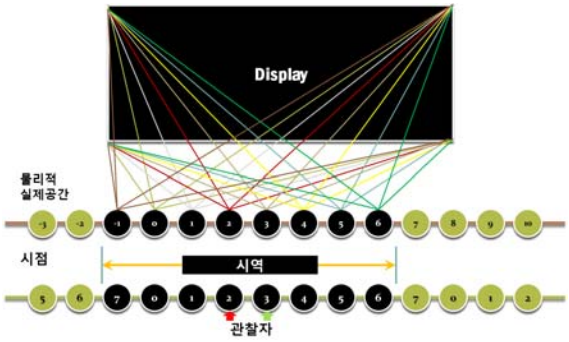

그림 14. 관찰자 위치 (좌측)이동에 따른 시점 재구성

그림 14는 관찰자가 좌측으로 한 시점 이동한 경우이다. 그림 13에서 관찰자의 움직임이 없는 경우 관찰자의 두 눈은 3번과 4번 시점에 위치하였고, 그림 14에서는 관찰자가 한 개의 시점 크기만큼 이동하여 관찰자의 두 눈은 2번과 3번 위치에 있다. 이 경우 관찰자의 위치를 중심으로 좌, 우 동일한 시점을 재생성 하였다. 물리적 공간은 -1이라고 정의한 위치까지 확장되었고, 이 물리적 공간 -1에서 시점영상을 만든다.

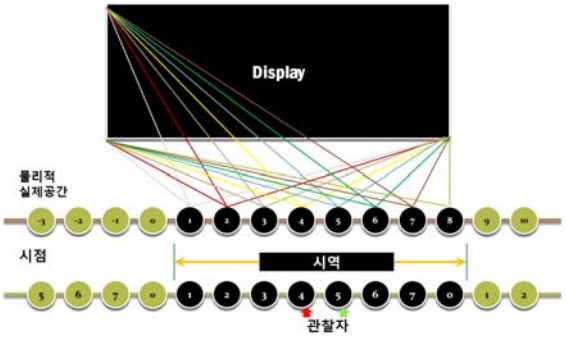

그림 15. 관찰자 위치 (우측)이동에 따른 시점 재구성

관찰자가 우측으로 움직일 경우도 방향만 다를 뿐 동일하다. 관찰자의 위치가 4,5번 시점에 위치하였기 때문에 물리적 공간 8번으로 시점을 확대하고 0번 시점을 없애, 관찰자 중심으로 좌, 우 동일하게 시역을 유지시킨다.

위와 같은 시역의 변화를 실시간으로 적용할 경우, 관찰자는

좌, 우 자유로운 움직임에도 끊김 없이 입체영상을 볼 수 있다.

6. 결론 및 향후연구

본 연구에서는 무안경식 다시점 입체 디스플레이 시스템을 구축하고, 이 시스템에서 관찰자 위치변화에도 부드러운 입체영상을 보여주기 위한 효율적인 GPU기반 렌더링 기법을 소개하였다. 관찰자의 현재 위치를 실시간으로 알아내기 위해 얼굴추적 기법을 사용하였고, 이를 통해 얻어진 관찰자 위치를 중심으로 실시간 새로운 시역을 형성하여 관찰자는 디스플레이 설계 시 정해진 시역범위 밖에서도 입체영상을 끊기지 않고 볼 수 있다. 렌더링 프로세스의 전체적인 흐름은 그림 16과 같다.

하지만, 현재의 본 연구에서는 입체영상을 볼 수 있는 시역의 좌, 우 범위만 고려되었다. 차후 좌, 우 범위뿐만 아니라 디스플레이 장치와 관찰자간의 거리 변화, 관찰자 위치의 눈높이 변화에 대한 연구가 진행되어야 한다.

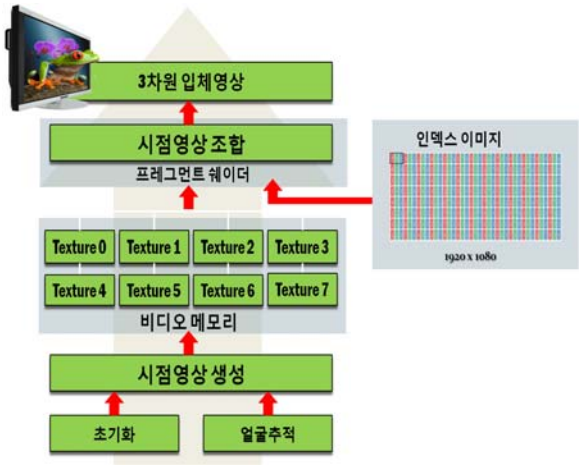

그림 16. 렌더링 프로세스

참고문헌

[1] Kooima, R. Peterka, T. Girado, J. Ge, J. Sandin, D. DeFanti, T. (2007). A GPU Sub-pixel Algorithm for Autostereoscopic Virtual Reality. In Proceedings of IEEE Virtual Reality 2007. Charlotte, pp. 131-138.

[2] Electro. http://www.evl.uic.edu/rlk/electro(2010.07.01)

[3] Peterka, T. Koonima, R. Girado, J. Ge, J. Sandin, D. DeFanti, T. (2007) Evolution of the Varrier Autostereoscopic VR Display. In Proceedings of IS&T SPIE Electronic Imaging 2007. San Jose, pp. 649004

[4] FaceAPI. http://www.seeingmachines.com/product/faceapi/ (2010.06.28)

[5] Kessenich, J. The OpenGL2 Shading Language. http://www.opengl.org/documentation/glsl (2010.05.23.)

[6] Perlin, K. Paxia, S. Kollin, J. (2000) An Autostereoscopic Display. In Proceedings of ACM SIGGRAPH 2000. New York, pp. 319-326

[7] Cruz-Neira, C. Sandin, D. DeFanti, T. (1993) Surround-Screen Projection-based Virtual Reality : The Design and Implementation of the CAVE. In Proceedings of ACM SIGGRAPH 1993. New York, pp. 135-142

[8] Matusik, W. Pfister, H. (2004) 3D TV:A Scalable System for Real-Time Acquisition, Transmission, and Autostereoscopic Display of Dynamic Scenes. In Proceedings of SIGGRAPH 2004. Los Angeles, pp. 814-824

[9] Harrold, J. Woodgate, J. (2007) Autostereoscopic display technology for mobile 3DTV applications. In Proceedings of IS&T SPIE Electornic Imaging 2007. San Jose, Vol. 6490A-19

[10] Sorbier, F. Nozick, V. Biri, V. (2008) GPU rendering for autostereoscopic displays. In Proceedings of 3DPVT 2008, Atlanta,

[11] Sandin, D. Margolis, T. Ge, J. Peterka, T. Defanti, T. (2005) The Varrier™ Autostereoscopic Virtual Reality Display. In Proceedings of ACM SIGGRAPH 2005. Los Angeles, pp.894-903

[12] Peterka, T. Sandin, D. Ge, J. Girado, J. Kooima, R. Leigh, J. Johmson, A. Thiebaux, M. DeFanti, T. (2006) Personal Varrier : Autostereoscopic Virtual Reality Display for Distributed Scientific Visualization. Future Generation Computer Systems. pp..976-983

[13] Wii Remote. http://en.wikipedia.org/wiki/Wii_Remote (2010.05.14)

[14] Jinwook Kim. Virtual Physics. http://virtualphysics.imrc.kist.re.kr/ (2010.05.07)

[15] FBO(Framebuffer object). http://oss.sgi.com/projects/ogl-sample/registry/EXT/framebuffer_object.text (2010.04.29)

[16] Cruz-Neira, C. Sandin, J. Defanti, T. Robert V. Kenyon, R. Hart, J. (1992) The CAVE : Audio Visual Experience Automatic Virtual Environment. In Proceedings of ACM SIGGRAPH 1992. pp.64-72

[17] Dodgson, N. (1997) Autostereo Display : 3D without glasses. In Proceedings of the Electornic Information Displays 1997.

[18] Lichtenbelt, P. Brown, P. EXT_gpu_shader4 Extensions Specifications. http://developer.download.nvidia.com/opengl/specs/GL_EXT_gpu_shader4.txt (2010.04.11.)

안 종 길

2001년 2월 ~ 2007년 2월 호서대학교 게임공 학과 졸업. 2008년 3월 ~ 2010년 8월 고려대학교 컴퓨터 · 전파통신학과 졸업(공학석사). 2010년 8월 ~ 현재 고려대학교 컴퓨터 · 전파통신학과 박사과정. 관심분야는 가상현실, 증강현실, HCI임

김 진 욱

2002년 4월 ~ 현재 한국과학기술연구원 영상미디어센터 선임연구원. 2005년 NYU 방문연구원. 2006년 UCLA 방문연구원. 관심분야는 가상현실, HCI, 환경 모델링, 로보틱스, 시뮬레이션임