

집적 영상 시스템을 위한 향상된 이미지 매핑 방법

이 빛 나 라[†] · 조 용 주^{**} · 박 경 신^{***} · 민 성 욱^{****}

요 약

집적 영상(Integral Imaging) 시스템은 사용자가 특수 안경을 착용하지 않고 3차원 입체영상을 볼 수 있도록 해주는 무안경식 양안시차 디스플레이 방식 중 한가지이다. 집적 영상 시스템은 3차원 정보를 조금씩 다른 방향에서 제한된 크기로 촬영된 이미지만인 기초영상(Elemental Image)의 형태로 저장한다. 그리고 이런 이미지들을 다시 렌즈 어레이를 통해서 보여주어 사용자들이 하여금 3차원 정보를 얻을 수 있도록 한다. 이런 기초영상은 컴퓨터 그래픽을 이용해서 만들어낼 수도 있는데 이런 집적 영상 방식을 CG(Computer Generated) 집적 영상 시스템이라고 한다. 그리고 컴퓨터를 이용해 기초영상을 만드는 과정을 이미지 매핑(Image Mapping)이라고 한다.

이제까지 제안된 이미지 매핑 방식에는 PRR(Point Retracing Rendering), MVR(Multi-Viewpoint Rendering), PGR(Parallel Group Rendering) 등이 있다. 그러나 이런 방식들은 계산량이 많거나 렌즈 어레이 개수의 증가에 의해 생성 속도에 영향을 받는 단점이 있어, 아직 가상현실 같은 실시간 CG(Computer Graphics) 응용 분야에서 사용하기 어려운 문제가 있다. 따라서 본 논문에서는 기존의 방법에 비해 속도 향상을 이룬 VVR(Viewpoint Vector Rendering)이라는 향상된 이미지 매핑 방법을 제안한다. 먼저 VVR 개념을 설명한 후 VVR을 사용한 집적 영상 시스템을 구현하여 MVR 방법과 비교 분석한 실험결과와 추후 개선되어야 할 방향을 제시한다.

키워드 : 집적 영상 시스템, 무안경식 양안시차 디스플레이, 이미지 매핑, 실시간 그래픽스 렌더링 알고리즘

Enhanced Image Mapping Method for Computer-Generated Integral Imaging System

Bin-Na-Ra Lee[†] · Yongjoo Cho^{**} · Kyoung Shin Park^{***} · Sung-Wook Min^{****}

ABSTRACT

The integral imaging system is an auto-stereoscopic display that allows users to see 3D images without wearing special glasses. In the integral imaging system, the 3D object information is taken from several view points and stored as elemental images. Then, users can see a 3D reconstructed image by the elemental images displayed through a lens array. The elemental images can be created by computer graphics, which is referred to the computer-generated integral imaging. The process of creating the elemental images is called image mapping.

There are some image mapping methods proposed in the past, such as PRR(Point Retracing Rendering), MVR(Multi-Viewpoint Rendering) and PGR(Parallel Group Rendering). However, they have problems with heavy rendering computations or performance barrier as the number of elemental lenses in the lens array increases. Thus, it is difficult to use them in real-time graphics applications, such as virtual reality or real-time, interactive games. In this paper, we propose a new image mapping method named VVR(Viewpoint Vector Rendering) that improves real-time rendering performance. This paper describes the concept of VVR first and the performance comparison of image mapping process with previous methods. Then, it discusses possible directions for the future improvements.

Key Words : Integral Imaging System, Auto-stereoscopic Display, Image Mapping, Real-time Graphics Rendering Algorithm

1. 서 론

가상현실(Virtual Reality)은 사실감 있는 정보를 실시간 컴퓨터 그래픽을 이용해 보여주며 관찰자로 하여금 가상현

경에 몰입되어 상호작용을 할 수 있도록 지원해주는 시스템 전반을 일컫는다[1]. 가상현실 시스템은 디스플레이 시스템, 렌더링 엔진(rendering engine), 트래킹(tracking) 시스템 등으로 구성되어 있다. 이 세 가지 시스템 중 인간의 정보 습득 의존도가 시각에 가장 큰 영향을 받는다는 점 때문에, 사용자에게 현실감 있는 시각 정보를 줄 수 있는 3차원 입체영상 디스플레이에 대한 연구가 활발히 이루어지고 있다. 지금까지 개발된 3차원 입체영상디스플레이 방식으로는

† 준 회원 : 상명대학교 컴퓨터학과 석사과정
 ** 종신회원 : 상명대학교 미디어학부 조교수
 *** 종신회원 : 한국정보통신대학교 디지털미디어연구소 연구교수
 **** 정 회원 : 미국 Virginia Tech 박사 후 연구원
 논문접수 : 2006년 3월 20일, 심사완료 : 2006년 4월 21일

양안시차(stereoscopic) 방식, 부피표현(volumetric) 방식, 홀로그래픽(holo-graphic) 방식으로 분류할 수 있는데, 현재 가상현실 시스템에서 많이 사용되고 있는 방식은 양안시차 방식이다. 양안시차 방식은 사실적 입체 표현이 가능하지만, 연속적인 시차 정보를 제공하지 못하기 때문에 관찰자에게 영상의 깊이감만을 전달할 뿐 공간 정보는 전달하지 못한다. 또한 서터 글라스나 편광 안경 같은 특수한 장비를 관찰자가 착용을 해야 하므로 사용자로 하여금 즉각적으로 입체 영상을 볼 수 있도록 해주는 즉시성이 떨어진다.

이런 문제를 해결하기 위해 렌티큘라 렌즈(lenticular lens) 등을 디스플레이 장치에 설치하여, 관찰자가 보조 장치 없이 육안으로 3차원 입체 영상을 관찰할 수 있는 무안경식 양안시차(auto-stereoscopic) 기술이 개발되고 있지만, 주로 고정되거나 한정된 시점만을 제공하므로 제한적이라 할 수 있다[2]. 하지만 최근에 주목을 받는 집적 영상 시스템(integral imaging system)은 무안경식 양안시차 방식임에도 불구하고, 마치 부피표현(volumetric) 디스플레이 방식처럼 연속된 시점을 제공해 준다는 장점이 있다[3].

집적 영상 시스템의 기본적인 원리는 1908년 Lippman이 처음 제안하였다[4]. 집적 영상 시스템은 (그림 1)에서와 같이 픽업(pick-up)하는 부분과 디스플레이(display)하는 부분으로 구성된다. 픽업 부분에서는 렌즈 어레이(lens array)와 CCD 카메라 같은 픽업 장치를 이용해서 3차원 정보를 2차원 이미지인 기초영상(elemental image)들로 저장한다[5]. 기초영상들은 미리 결정된 조금씩 다른 방향들로부터 렌즈 어레이에 있는 기초렌즈의 방향과 크기에 영향을 받는 일정한 크기로 촬영된 사진들이다.

집적 영상 시스템은 이렇게 반복되는 영상을 가진 사진들의 주기와 렌즈 어레이의 주기의 차이를 이용하여 깊이 정보를 저장하고, 연속된 시차를 제공할 수 있게 된다. 이렇게 만들어진 기초영상들을 디스플레이 부분에서 다시 렌즈 어레이를 통하여 관찰자가 보게 되면, 연속된 시점을 가지는 입체 영상을 볼 수 있게 된다.

집적 영상 시스템을 구성할 때에는 (그림 1)에서 보인 바와 같이 직접 렌즈 어레이와 픽업 장치를 사용해서 기초영상을 얻는 방법 외에도, 표현하고자 하는 3차원 영상 정보에 대해 일련의 계산을 통해 컴퓨터 그래픽을 통해서 기초영상을 만들어낼 수도 있다. 이와 같이 컴퓨터 계산에 의해

생성된 기초영상을 이용하는 집적 영상 방식을 CG(computer-generated) 집적 영상이라 부른다. 그리고 기초영상을 얻는 과정을 이미지 매핑(image mapping)이라고 한다.

본 논문에서는 기존의 이미지 매핑 방법에 비해 VVR(Viewpoint Vector Rendering)이라는 향상된 방식을 제안한다. 우선 기존의 이미지 매핑 방법을 설명하고 VVR의 개념과 수식을 보여주고 기존 방법과 비교 분석한 실험결과를 토론한다. 또한 향후 개선해야 할 연구 방향에 대해 논한다.

2. 기존의 이미지 매핑 방법들

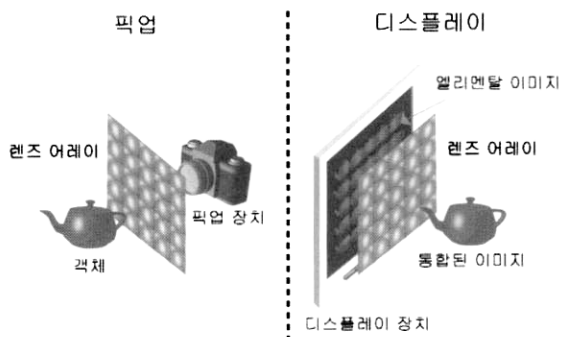
기존에 제안된 컴퓨터 그래픽 집적 영상 시스템에서의 이미지 매핑 방법으로 PRR(Point Retracing Rendering)[5], MVR(Multiple Viewpoint Rendering)[6], PGR(Parallel Group Rendering)[7] 등이 있다.

PRR 방식은 이미지 매핑 방법 중 가장 간단한 알고리즘이다. 이 방법은 화면에 표시할 개체의 모든 점들을 기초영상들의 픽셀에 대응시켜 만들어 낸다. 대상물의 한 점이 일정한 간격을 갖는 점들의 형태로 기초영상이 표현되며, 이 과정을 대상물의 모든 점에 대해 적용시키는 것이다. PRR 방식은 구현이 간단하지만 대상물의 모든 점들에 대해 기초영상들을 만드는 계산이 수행 되므로 대상물의 크기가 커짐에 따라 계산량이 늘어나 실시간 처리가 불가능하다는 단점이 있다.

MVR은 집적 영상 시스템이 렌즈 어레이를 이용하여 기초영상을 픽업하는 과정을 컴퓨터 그래픽을 이용해서 유사하게 수행하는 방식이다. MVR에서는 OpenGL 같은 3차원 컴퓨터 그래픽 라이브러리로 가상의 3차원 공간에 존재하는 개체를 가상의 카메라를 이용해 촬영하여 기초영상들을 만들어 낸다. 이때 카메라는 기초렌즈 피치(pitch) 만큼씩 이동을 하고, 뷰포인트와 배율이 기초영상 생성에 영향을 미치게 된다.

MVR은 렌더링 과정이 단순하면서도 정확한 기초영상을 얻을 수 있다. 또한 다양한 3차원 객체를 별다른 변환 과정을 거치지 않고, 집적 영상 시스템에 적용시킬 수 있으며, PRR 방식과는 달리 대상물의 크기가 기초영상 생성 속도에 영향을 주지 않는다. 하지만 MVR은 PRR 방식과는 달리 렌즈 어레이에 존재하는 기초렌즈의 개수만큼 뷰포인트를 생성해 가상의 카메라를 각각의 기초렌즈 위치에 놓고 찍게 된다. 그렇기 때문에 기초렌즈의 수가 증가함에 따라 기초영상을 만들어내는 속도가 느려지게 되므로, 해상도가 높아지거나 기초렌즈의 개수가 많아질 경우에는 실시간 처리가 어렵다는 단점이 있다.

PGR은 집적 영상 시스템에서 포커스드 모드(focused mode)일 경우, 각각의 기초렌즈가 픽셀처럼 보이는 특성을 적용한다. 포커스드 모드는 기초렌즈의 초점거리와 디스플레이 장치와 렌즈 어레이와의 간격이 같은 값을 갖는 경우로, 이 경우 집적 영상 시스템은 렌즈 어레이 앞뒤로 동시에 집적 영상을 표시할 수 있다. PGR은 시스템 요소들에 의해 정



(그림 1) 집적 영상 시스템 개념도

해진 방향에서 촬영된 디렉셔널 신(directional scene)들을 픽셀 단위로 처리하여 기초영상을 만드는 방법이다.

MVR에서 촬영되는 장면의 개수가 렌즈 어레이의 기초렌즈의 개수에 의해 결정되는 반면에 PGR은 기초렌즈 영역안의 픽셀의 수로 결정된다. 그러므로 PGR에서는 렌즈 어레이의 기초렌즈의 개수는 생성 속도에 영향을 미치지 않는다. 그러나 PGR은 포커스드 모드의 특성을 이용한 방법이므로, 포커스드 모드의 특성상 집적 영상의 두께는 충분히 표현할 수 있지만 해상도가 기초렌즈 크기의 역수로 떨어진다.

3. Viewpoint Vector Rendering

본 논문에서 제안하는 VVR은 표현하는 대상의 크기나 시스템에서 사용하는 렌즈 어레이의 기초렌즈의 수에 영향을 받지 않아서 실시간 처리에 보다 유리하다. VVR은 촬영해야 하는 장면의 개수가 MVR과 같이 렌즈 어레이의 기초렌즈 개수에 영향을 받지 않고 배율에 따라 디렉셔널 신을 사용하는 점에서 PGR과 흡사하다. 하지만 PGR과는 달리 디렉셔널 신을 픽셀 단위가 아니라 미리 계산된 크기로 잘라 처리하기 때문에 집적 영상 시스템의 디스플레이 모드 중 포커스드 모드뿐만 아니라, 렌즈의 초점거리가 디스플레이 장치와 렌즈 어레이와의 간격이 작아서 집적 영상이 어레이 앞에 맺히는 리얼 모드(real mode)와 렌즈의 초점거리가 디스플레이 장치와 렌즈 어레이와의 간격이 커서 집적 영상이 어레이 뒤에 맺히는 가상 모드(virtual mode)에서 다 사용할 수 있다. 따라서 VVR 알고리즘은 실시간 기초영상 생성 처리가 가능하고, OpenGL 같은 그래픽 라이브러리를 사용하여 가상 3차원 물체에 대한 기초영상을 형성하는 CG 집적 영상 시스템에 적용이 쉽다.

(그림 2)는 렌즈 어레이의 기초렌즈 개수(N)가 13이고, 배율(M)이 5인 집적영상 시스템에서 VVR 이미지 매핑 과정을 나타낸 것이다. (그림 2) (가)는 디렉셔널 신을 촬영하는 과정이고, (그림 2) (나)는 영상처리를 통해 기초영상들을 생성하는 과정이다.

식 (1)은 집적 영상 시스템의 배율(Magnification)을 나타낸다.

$$M = \frac{|L|}{g} = \frac{f}{|g-f|} \tag{1}$$

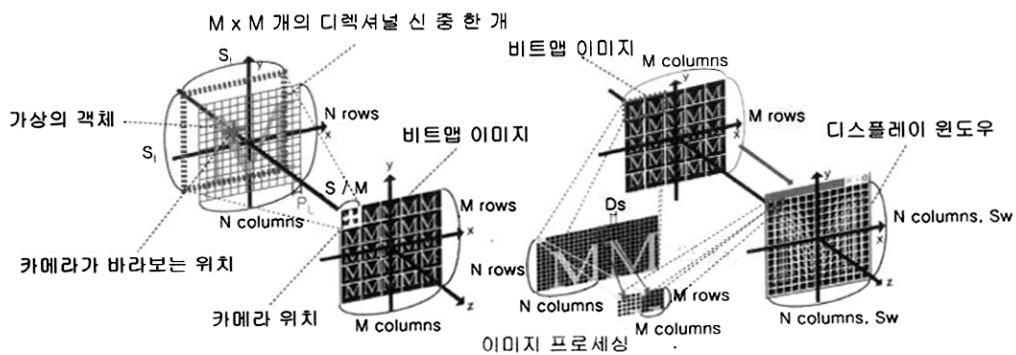
식 (1)에서 f 는 렌즈의 초점 거리, g 는 디스플레이 장치와 렌즈 어레이와의 거리, L 은 렌즈 어레이와 CDP(Central Depth Plane)와의 거리이다. CDP란 g 에 의해서 정해지는 초점이 맞는 위치이다. 식 (1)에 의해 구해진 M 은 (그림 2) (가)의 비트맵 이미지 부분에서 보이듯이 VVR 에서 촬영할 디렉셔널 신의 개수를 결정한다.

VVR의 기초영상 생성 과정에서 디렉셔널 신 촬영을 위해 먼저 몇몇 요건들의 계산이 필요하다.

$$\begin{aligned} S_l &= P_l \times N \\ S'_l &= P_l \times M \\ Val &= \frac{S_l}{S'_l} \\ S_w &= \frac{S_l}{P_x} = \frac{P_l}{P_x} N \\ S'_w &= \frac{S'_l}{P_x} = \frac{P_l}{P_x} M \end{aligned} \tag{2}$$

식 (2)에서 S_l 는 VVR에서 가상의 객체를 촬영한 장면에 대한 한 변의 크기로 렌즈의 지름(P_l)과 렌즈 어레이의 기초렌즈 개수(N)의 곱으로 결정된다. S'_l 는 MVR에서 촬영할 장면의 한 변의 크기로 렌즈의 지름(P_l)과 배율(M)의 곱으로 결정되며, 이 값을 이용하여 VVR에서 가상 카메라가 촬영할 FOV(Field of View)를 구한다. Val 은 MVR 시스템 기준으로 계산된 요건들을 VVR 시스템에 알맞게 변환시킬 때 사용되는 값이다. 그리고 구해진 S_l , S'_l 값을 디스플레이의 도트 크기(P_x)로 나누면, 각각 VVR과 MVR 시스템이 생성하는 디스플레이 윈도우 크기 픽셀 개수 S_w 와 S'_w 이 나온다.

(그림 2) (가)에서 보이는 디렉셔널 신 촬영을 위하여 3차원 공간상의 정확한 카메라 위치 V_{ij} 을 구해야 한다. 이를 위해 먼저 M 의 중간 값인 C_M 을 계산한다. 그리고 카메라 위치 V_{ij} 값은 식 (3)으로 계산한다. 현재 구현된 VVR



(그림 2) VVR 이미지 매핑 : (가) 가상 카메라의 픽업 과정 (왼쪽) (나) 기초영상들 생성 과정 (오른쪽)

알고리즘은 C_M 을 이용하여 0부터 M^2 까지 V_{ij} 의 x, y축 값을 계산하는 방식으로 M 이 홀수일 경우에만 적용가능하다.

$$V_{ij} \left(\frac{i}{M} - C_M + 1, C_M - \frac{j}{M} + 1, L \times Val \right) \quad \text{식 (3)}$$

where $i, j = 0, 1, 2, \dots, M - 1$

(그림 2)의 (나)에서 보이듯이 VVR 집적 영상 시스템은 이 디렉셔널 신을 일정한 크기로 조각내서 정해진 위치에 재배치하는 간단한 영상처리를 거쳐 기초영상들을 생성한다. 이 때 조각내는 크기(D_s)는 식 (4)로 계산하는데, 조각낸 크기는 디스플레이 도트 크기(P_x)보다 작을 수 없다.

$$D_s = \frac{P_L}{P_x M} \quad \text{where } D_s \geq P_x \quad \text{식 (4)}$$

그리고, 조각들이 디스플레이 윈도우에 위치할 좌표(V_{xy})는 식 (5)로 계산한다.

$$V_{xy} \left(D_s \left(\frac{S_w}{M} \times \frac{n_x}{N} + \frac{m_x}{M} \right), D_s \left(\frac{S_h}{M} \times \frac{n_y}{N} + \frac{m_y}{M} \right) \right)$$

where $n_x, n_y = 0, 1, 2, \dots, N - 1$ $m_x, m_y = 0, 1, 2, \dots, M - 1$ 식 (5)

이 과정을 마치게 되면 하나의 디렉셔널 신에 대한 처리가 완료된 것이다. 그리고 계속하여 x, y축으로 M^2 개수만큼 다른 디렉셔널 신을 촬영하고, 그 디렉셔널 신을 조각내어 배치하는 영상처리 과정을 반복하여 최종 기초영상들을 만들어낸다.

4. 구현

VVR 방식을 이용한 집적 영상 시스템은 C++와 OpenGL을 이용하여 구현하였다. VVR이 만들어 내는 기초영상들이 얼마나 정확한지 비교하기 위해 기준으로 사용할 기초영상들을 만들어내는 MVR도 함께 구현하였다. 비교 기준이 되는 시스템 요건들의 값은 <표 1>과 같다.

VVR 시스템의 대략적인 동작 순서는 다음과 같다.

1. <표 1>에 명시된 시스템 요건들을 지정된 속성 파일로부터 읽고 식 (1)에서부터 식(5)까지 차례대로 적용하여 시스템에서 사용할 모든 요건들의 값을 미리 계산한다.
2. 계산이 완료된 후 (그림 2) (가)와 같이 하나의 디렉셔널 신을 촬영하게 되는데, 촬영할 장면의 크기는 식 (2)의 S_I 이고, 생성된 이미지의 크기는 S_I / M 이 된다.
3. (그림 2) (나)와 같이 촬영된 디렉셔널 신을 식 (4)의 D_s 크기의 세그먼트로 잘라 식 (5)의 V_{xy} 에 재배치하는 영상처리 과정을 수행한다.
4. 2와 3 과정을 x, y축으로 M 만큼 반복한다.

<표 1> 구현된 VVR System의 시스템 요건

디스플레이와 렌즈 어레이와의 거리 (g)	17.6 mm
렌즈 어레이와 CDP와의 거리 (L)	88 mm
렌즈의 지름 (P_L)	10 mm
디스플레이의 도트 크기 (P_x)	0.25 mm
배율 (M)	5
렌즈 어레이 한 번의 렌즈 개수 (N)	13
렌즈의 초점거리 (f)	22 mm

5. 최종적으로 (그림 2) (나)의 디스플레이 윈도우에 표현될 완전한 기초영상들 집합을 얻게 된다.

<표 2> VVR 방식과 MVR방식으로 나타난 기초영상 PSNR 비교

MVR vs. VVR	PSNR_r	PSNR_g	PSNR_b
2차원 M 글씨 그림	143.84	143.84	143.84
2차원 주전자 그림	121.45	120.51	120.89
3D 소 모델	59.73	63.75	62.31
3D 주전자 모델	49.17	53.63	56.17

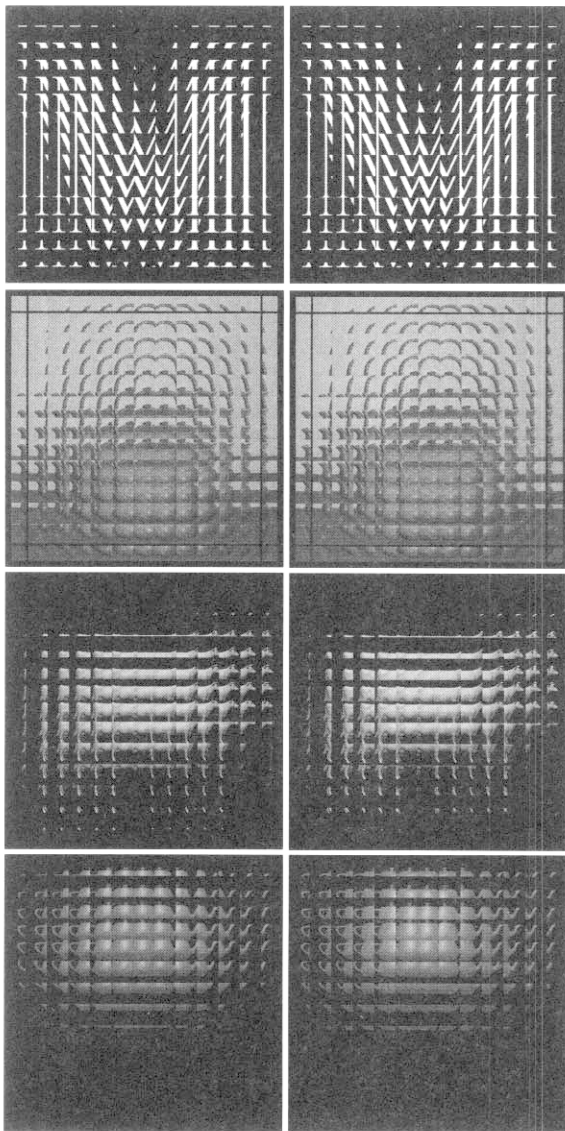
5. 실험 결과 및 토론

본 논문에서는 먼저 VVR 시스템에서 생성한 기초영상의 유효하다는 것을 검증하기 위해 MVR시스템에서 생성한 기초영상과 PSNR(Peak Signal to Noise Ratio) 값을 비교하였다. PSNR은 그림 간의 유사성을 찾기 위하여 사용하는 것으로 일반적으로 40dB 이상일 경우 유사한 그림으로 생각한다.

(그림 3) (가), (나)의 좌측 그림과 우측 그림은 각각 2차원 이미지에 대한 VVR과 MVR 방식을 사용하여 만들어진 기초영상들이다. (그림 3) (다), (라)의 좌측 그림과 우측 그림은 3차원 객체에 대한 VVR과 MVR 방식으로 만든 기초영상들이다. VVR과 MVR로 생성된 두 그림은 육안으로 볼 때에 큰 차이를 느낄 수 없다. 그리고 <표 2>의 PSNR 값을 보았을 때도 50dB에서 150dB 정도로 이미지 차이가 거의 없음을 알 수 있다.

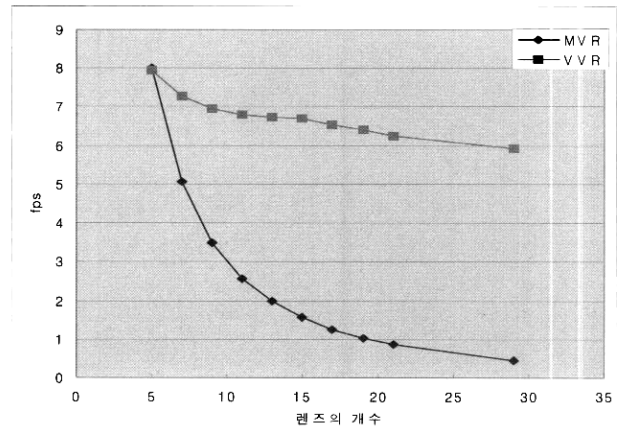
2차원 이미지(2차원 글씨와 주전자 그림)의 경우에는 PSNR 값이 매우 높게 나타났다. 그러나 3차원 객체들은 객체의 두께, 조명, 그리고 식 (3)에서 계산된 카메라 위치의 차이 등의 요소로 인하여 각각 카메라 위치에서 촬영된 장면이 조금 더 차이가 생긴다. 그래서 <표 2>에서 보는 바와 같이 3차원 객체로 만든 기초영상은 2차원 이미지로 만든 기초영상에 비해 PSNR값이 상대적으로 낮다. 이는 VVR의 뷰포인트 위치 차이로 인해 깊이를 표현할 때에 어려움이 생기기 때문이다.

예를 들어 3차원 주전자 모델의 경우, 깊이 정보가 다른 모델에 비해 상대적으로 두꺼웠던 점 때문에 MVR과 비교



(그림 3) 위에서부터 (가) VVR(좌측)과 MVR(우측)로 만든 2차원 M 글씨에 대한 기초영상이미지 (나) VVR(좌측)과 MVR(우측)로 만든 2차원 주전자 그림에 대한 기초영상이미지 (다) VVR(좌측)과 MVR(우측)로 만든 3차원 소 모델에 대한 기초영상이미지 (라) VVR(좌측)과 MVR(우측)로 만든 3차원 주전자 모델에 대한 기초영상이미지

했을 때 VVR의 기초영상의 품질이 좀 더 떨어진 것을 볼 수 있다. 하지만 이것 역시 40dB이상이므로 VVR 방식으로 3차원 객체를 이용해 생성한 기초영상도 유효하다고 볼 수 있다. 이렇게 비슷한 기초영상을 만들어낼 수 있으면서도, MVR과 VVR 두 방식의 렌더링 속도를 비교하면 VVR이 약 1.5~5배 정도 빠르다. 이는 <표 1>의 요건을 갖는 집적 영상 시스템의 경우, MVR이 13x13만km 카메라 촬영을 해야 하는 반면에 VVR은 5x5만km 촬영하기 때문에 이론상 6배 이상 속도가 빨리 나와야 하지만, VVR 방식이 3차원 렌더링의 부담을 줄이기 위해 2차원 영상처리를 하는 과정을 수행하기 때문에 전체적인 렌더링 속도를 떨어뜨린 결과이다.



(그림 4) 렌즈 개수에 따른 MVR과 VVR의 FPS

(그림 4)는 3차원 소 객체 모델을 이용하여 렌즈 어레이의 기초렌즈 개수에 따른 VVR과 MVR의 FPS(frames per second)를 측정된 그래프이다. 현재 VVR 구현은 Windows PC 기반의 오프스크린 렌더링(offscreen rendering) 방식으로 구현된 것으로 그래픽 하드웨어 가속이 되지 않아 FPS가 아주 낮다.

그러나 (그림 4)에서 보듯이 VVR이 MVR에 비해 상대적으로 시스템 요건에 영향을 덜 받는 것을 보여준다. MVR이 렌즈 어레이의 기초렌즈 개수의 증가에 따라 속도가 급속하게 감소하는 것에 비해 VVR은 렌즈 개수에 영향을 받지 않는다. 또한 PGR은 포커스드 모드만 지원하는 반면 VVR은 리얼 모드, 가상 모드에서도 다 사용할 수 있으며, 포커스드 모드일 때 VVR은 PGR과 일치한다.

6. 결론 및 향후 연구 방향

본 논문에서 제안하는 VVR 방식은 표현하는 대상의 크거나 시스템에서 사용하는 렌즈 어레이의 렌즈의 개수에 영향을 받지 않아서 실시간 처리에 유리한 새로운 이미지 매핑 방식이다. 정확한 기초영상들을 만들어 내는 기존의 이미지 매핑 방식인 MVR과 비교 분석 결과, VVR은 MVR과 비교해서 손색이 없는 기초영상들을 생성함에도 불구하고 수행속도가 약 1.5~5배 정도 빠른 것을 확인했다. 하지만 VVR에서 만들어지는 기초영상들은 가상카메라 뷰포인트의 위치 차이로 인해 3차원 모델에 깊이가 더 있을 때 MVR과 비교해서 에러가 생기고, 따라서 MVR과 VVR 간의 PSNR 차이가 커진다.

그리고 그래픽 하드웨어 가속을 사용하지 못해 아직 실시간 시스템에 적용하기에는 생성 속도가 느리다. 앞으로 VVR의 이런 단점들을 해결할 수 있도록 알고리즘을 개선할 예정이다. 이런 단점들이 해결되면, VVR을 이용한 집적 영상 시스템을 사용자들이 특수한 안경을 착용하지 않고도 게임이나 가상현실 같은 실시간 3차원 입체영상 응용분야에 사용될 수 있을 것이다.

참 고 문 헌

- [1] Frederick P. Brooks, Jr, "What's Real About Virtual Reality?" IEEE Computer Graphics & Applications, Nov./Dec., 16, 1999
- [2] Sung-Wook Min, Myung Hoon Suk, Kyoung-Shin Park "Study about 3D display system suitable to VR", HCI2005, pp.576-581, 2005.
- [3] Takanori Okoshi, "Three-Dimensional Displays," Proceedings of the IEEE, Vol.68, No.5, pp.548-564, 1980.
- [4] G. Lippmann "La photographie integrale," Comptes-Rendus 146, pp.446-451, 1908.
- [5] Yoshihide Igarashi, H. Murata and M. Ueda, "3D display system using a computer generated integral photography," Japan J. Appl. Phys., 17, pp.1683-1684 1978.
- [6] Myunghoon Suk "Enhanced image mapping algorithm for 3D integral imaging display system", Information and Communications University 2005.
- [7] Ruigang Yang, Xinyu Huang, Shunnan Chen, "Efficient Rendering of Integral Images", SIGGRAPH 2005.



이 빛 나라

e-mail : hlstyle79@hotmail.com
 2004년 상명대학교 소프트웨어학부(학사)
 2004년~현재 상명대학교 컴퓨터학과 석사과정
 관심분야 : 가상현실, 게임엔진

조 용 주



e-mail : ycho@smu.ac.kr
 1993년 일리노이대학 컴퓨터학과(학사)
 1997년 일리노이대학 전기전자컴퓨터학과(공학석사)
 2003년 일리노이대학 컴퓨터학과(공학박사)
 2004년~현재 상명대학교 미디어학부 조교수

관심분야 : 가상현실, HCI, 인터랙티브 컴퓨팅

박 경 신



e-mail : park@icu.ac.kr
 1991년 덕성여자대학교 수학과(학사)
 1997년 일리노이대학 전기전자컴퓨터학과(공학석사)
 2003년 일리노이대학 컴퓨터학과(공학박사)

2004년~현재 한국정보통신대학교 디지털미디어 연구소 연구교수
 관심분야 : 가상현실, HCI, Tele-Immersion, Computer Supported Cooperative Work

민 성 욱



e-mail : klein91@icu.ac.kr
 1995년 서울대학교 전기공학과(학사)
 1997년 서울대학교 전기공학부(공학석사)
 2004년 서울대학교 전기컴퓨터공학부(공학박사)
 현재 미국 Virginia Tech 박사 후 연구원

관심분야 : 3차원 디스플레이 시스템