

---

# 확장형 초고해상도 타일드 디스플레이 시스템에서 동작하는 협업용 뷰어 시스템 기술 개발

김민영\* · 이빛나라\*\* · 조용주\*\*\*

Development of Cooperative Viewer System Technology Running on the Scalable  
High-Resolution Tiled Display System

Minyoung Kim\* · Binnara Lee\*\* · Yongjoo Cho\*\*\*

---

이 논문은 2009년 교육과학기술부의 재원으로 한국연구재단의 지원을 받아 수행된  
기초연구사업 연구임 (NRF-2009-0076832)

---

## 요 약

타일드 디스플레이는 고해상도 이미지나 실물 크기의 3차원 개체를 표현할 수 있고, 다수의 사용자가 개인화된 화면을 사용하거나 남들과 공유할 수 있도록 지원하므로 효율적으로 협업에 활용될 수 있다. 본 연구에서는 분산 플랫폼 위에 3차원 모델, 이미지와 동영상을 조작할 수 있는 협업 뷰어 시스템을 개발했다. 본 시스템에서는 사용자가 다양한 입력장치를 활용하여 따로 콘텐츠를 조작할 수 있다. 본 논문에서는 뷰어 시스템의 구조와 이를 이용해 구축한 프로토타입 시스템에 대해 설명한다.

## ABSTRACT

A tiled display can present a high resolution image or real-sized 3D objects. It can be used for effective collaboration by allowing multiple users to allocate their own spaces, which are used privately or shared with others. In this research, a collaborative viewer system, which allows manipulating 3D models, images and movie clips, was constructed on a distributed platform. This allows multiple users to manipulate different contents separately with various input devices. This paper describes an architecture and a prototype system built with the viewer.

## 키워드

타일드 디스플레이, 확장형 디스플레이, 협업용 뷰어 시스템, 분산 렌더링 시스템

## Key Word

Tiled Display, Scalable Display, Collaborative Viewer System, Distribute Rendering System

---

\* 상명대학교 대학원 컴퓨터학과

\*\* (주) 리얼타임비주얼

\*\*\* 상명대학교 디지털미디어학부 (교신저자)

접수일자 : 2010. 01. 19

심사완료일자 : 2010. 03. 23

## I. 서 론

1990년부터 진행되어 온 대형 디스플레이를 활용하는 협업 환경 연구는 주로 기상 정보와 해양자료 같은 매우 복잡한 대용량의 과학적 자료를 대형 화면에 공용 디스플레이의 형태로 제공하며, 거기에 작은 디스플레이를 개인용 작업 공간으로 추가하여 활용하는 연구들이 많았다[1-3]. 하지만, 이러한 형태의 공용 디스플레이 시스템들은 주로 개인 화면을 공용화면 전체에 보이도록 하는 형태로 구성되어, 한번에 보여줄 수 있는 데이터의 양이 단일 디스플레이 장치의 해상도에 국한되는 한계를 가졌다[2].

이런 문제를 해결하기 위해 프로젝터 등을 여러 개 붙여서 고해상도 가시화 시스템을 구축하는 경우가 있으나 구축비용이 많이 들고 관리가 어렵다는 단점이 존재했다[4]. 그래서 최근에는 저렴한 가격의 LCD 디스플레이 여러 개를 붙여서 마치 한 개의 디스플레이인 것처럼 활용할 수 있도록 지원하는 타일드 디스플레이 시스템(Tiled Display System)을 기반으로, 더 많은 데이터를 보여주고 분석을 용이하게 하려는 시도들이 진행되고 있다[5,6]. 특히 타일드 디스플레이 시스템은 더 높은 해상도를 지원해야 하는 경우에 디스플레이와 컴퓨터를 추가로 설치하여 붙이기만 하면 되는 높은 확장성을 지닌 경우가 많다. 그러나 단일 디스플레이와 다르게 여러 개의 컴퓨터가 함께 운용되는 분산 환경이기 때문에 소프트웨어 개발의 복잡도가 높아지는 문제가 있다.

이러한 문제를 해결하기 위해, 타일드 디스플레이 시스템에서 실시간의 병렬 그래픽스 렌더링을 가능하도록 지원하는 WireGL/Chromium[7]과 Equalizer[8], 저사양의 컴퓨터와 메가비트 네트워크 기반의 확장 가능한 타일드 디스플레이 렌더링 기능을 제공하는 Garuda[9] 등 여러 개의 타일드 디스플레이 프레임워크가 개발되었다. WireGL/Chromium과 Equalizer는 가시화 작업을 위한 그래픽 렌더링 작업을 분산 컴퓨팅 시스템을 이용해서 나누어 처리한 후에 보여주는 것을 주목적으로 하였다. 또 Garuda는 Open Scene Graph 라이브러리를 이용하는 응용프로그램들을 수정하지 않고도 타일드 디스플레이 시스템에서 동작하도록 만드는 데 중점을 두고 있다. 고성능 네트워크를 이용한 고해상도 타일 디스플레이 렌더링 미들웨어인 SAGE[10]는 다른 컴퓨터에서

수행시킨 프로그램들을 화면에 스트리밍 해서 보여주거나 아니면 직접 미들웨어를 이용해서 구현해서 보여주는 것이 가능하다. 또한 iTDF[11] 프레임워크 역시 타일드 디스플레이에서 동작하는 3차원 그래픽 프로그램을 쉽게 작성할 수 있도록 만들어졌다.

본 연구에서는 타일드 디스플레이에 표출되는 고해상도 화면을 다수의 사용자들이 자유롭게 사용하거나 공유하면서 협업할 수 있는 증강 협업 공간을 진제한다. 그리고 이러한 형태의 증강 협업 공간을 타일드 디스플레이 시스템에 쉽게 구축할 수 있도록 지원하는 협업용 뷰어 시스템을 개발하였다. 본 시스템은 다수의 사용자들이 전체 화면을 공용 디스플레이의 형태로 사용하거나, 화면을 일부분 할해서 개인 작업 공간으로 활용할 수 있도록 다중 사용자의 인터랙션을 지원하는 증강 협업 공간을 제공한다.

본 연구에서 개발된 뷰어 시스템은 그래픽 렌더링 작업의 부하를 덜기 위해 분산 시스템을 활용한 기존의 WireGL/Chromium과, Equalizer 등과는 달리 다중 사용자가 함께 작업할 수 있는 협업 공간을 제공하는데 목적을 두고 있다. 또 윈도우 크기 조절이나 이동 등과 같은 간단한 형태의 인터랙션만을 지원하는 SAGE나 단일 사용자의 입력만을 염두에 두고 개발된 iTDF 등과 같은 연구와는 다르게 다중 사용자의 다양한 인터랙션을 지원할 수 있도록 개발되었고 이러한 작업을 응용프로그램 개발자에게 쉽게 지원할 수 있도록 프레임워크의 형태로 제공한다는 점은 본 연구 결과물의 차별성이 다.

본문에서는 이러한 협업용 뷰어 시스템 기술을 구성하는 요소들의 구조와 기능에 대하여 설명한다. 그리고 이를 활용해서 제작된 타일드 디스플레이 시스템을 소개하고, 끝으로 결론과 함께 차후 연구 방향에 대해 논한다.

## II. 확장형 초고해상도 타일드 디스플레이 구축을 위한 협업 뷰어 시스템 기술

### 2.1 협업 뷰어 시스템 구조

본 연구에서 전제된 확장형 초고해상도 타일드 디스플레이 시스템은 복수 사용자의 협업을 지원하기 위하여 다양한 3차원 콘텐츠의 렌더링과 조작성을 가능하게 하

는 증강 협업 환경이다. 이를 지원하기 위한 협업용 뷰어 시스템 기술은 그림 1에서 보인 것처럼 하나의 마스터 노드와 다수의 슬레이브 및 컨트롤러 노드로 구성되는 분산 시스템으로 구성된다. 마스터 시스템은 슬레이브와 컨트롤러 사이의 메시지 교환과 동기화를 총괄하여 관리하고, 슬레이브 시스템의 뷰어(Viewer) 모듈은 한 개 이상의 디스플레이 윈도우를 열고 렌더링하는 것을 담당한다. 컨트롤러는 뷰어와 트리거(Trigger) 모듈을 조합하여 구현되는데, 뷰어는 타일드 디스플레이의 전체 화면을 축소하여 작은 크기로 보여주기 위해 사용된다. 그리고 트리거는 사용자의 입력을 추상화하여 처리하면서 사용자의 인터랙션을 담당한다. 새로운 입력 장치의 API(Application Programming Interface)를 활용하여 컨트롤러에 포함된 타일드 디스플레이 프레임워크의 트리거 클래스를 상속 받은 새로운 트리거 모듈을 구현하면 바로 기존의 컨트롤러와 연결하여 쉽고 다양한 상호작용 방법을 지원하는 것이 가능하다. 이러한 3가지 서브시스템이 적용된 타일드 디스플레이는 UDP와 멀티캐스트 프로토콜을 활용해서 자체적으로 커맨드 메시지와 데이터를 주고받으면서 하나의 플랫폼으로써 유기적으로 동작하게 된다.

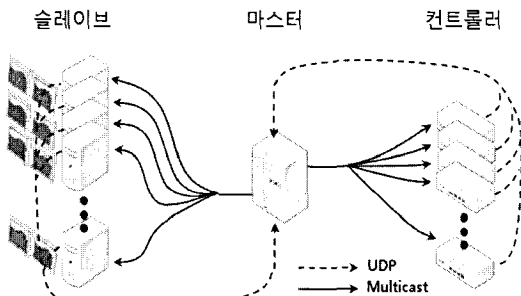


그림 1. 협업용 뷰어 시스템의 네트워크 구성  
Fig. 1 The network organization of cooperative viewer system

본 협업용 뷰어 시스템은 마스터/슬레이브 구조의 분산 렌더링을 지원하는데, 이에 필요한 렌더링과 네트워크 통신 기술은 그림 2에 나타난 것과 같이 Open Scene Graph(이하 OSG)[12]와 ACE[13] 라이브러리를 통해서 구현 되었다. 뷰어의 마스터, 컨트롤러, 슬레이브 노드의 근간을 이루는 타일드 디스플레이 프레임워크는 타일

드 디스플레이 시스템의 구현을 위해 마스터, 컨트롤러, 슬레이브 노드들 간의 네트워크 연결을 자동으로 구성할 수 있는 기능을 제공한다. 본 협업용 뷰어 시스템에서는 마스터 노드에서 슬레이브와 컨트롤러들에게 각각의 커맨드 메시지를 전달하기 위해 각기 다른 멀티캐스트 그룹으로 구성된 네트워크 연결을 자동으로 생성한다. 그리고 슬레이브나 컨트롤러 노드에서 마스터 노드로 메시지를 보낼 때에는 독립적인 UDP 소켓을 사용한다. 타일드 디스플레이 시스템에서 각 서브시스템들은 메시지 관리자(Message Manager)를 통해서 저마다 필요한 커맨드 메시지를 생성할 수 있으며, 모든 커맨드는 우선적으로 뷰어 시스템의 마스터 노드로 전달된 후에 전달 목표 지점에 따른 선별 작업을 통해 각 종류의 노드로 보내진다.

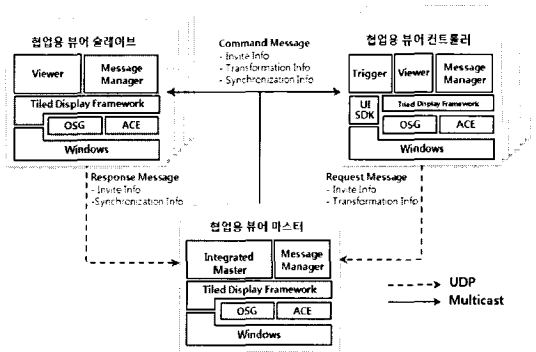


그림 2. 협업용 뷰어 시스템의 구조도  
Fig. 2 The architecture of cooperative viewer system

## 2.2 협업용 뷰어 마스터 노드

협업용 뷰어 시스템의 마스터 노드는 LAN환경에서 슬레이브와 컨트롤러 노드가 실행되면 그 노드들을 찾아내고 연결시키면서 관리하는 핵심적인 역할을 한다. 또한 마스터 노드는 각각의 슬레이브나 컨트롤러들이 메시지를 주고받는 데 있어 브로커 역할을 하는 중간자 역할을 하며 두 서브시스템 그룹을 연결시켜주는 구심점이 된다. 본 협업용 뷰어 시스템을 활용하는 타일드 디스플레이 시스템 안에는 통상적으로 한 개의 마스터 노드가 존재하며, 각 슬레이브 노드들 간의 디스플레이 동기화나 커맨드 메시지의 수신 및 전송은 마스터 노드를 통해서 처리하게 된다. 마스터 시스템은 윈도우즈 플랫

폼 기반의 GUI 인터페이스로 구성되었고, 그림 3에서 보인 것처럼 현재 타일드 디스플레이 그룹에 참여한 모든 슬레이브와 컨트롤러 노드의 개별 정보 및 상태 등을 보여준다.

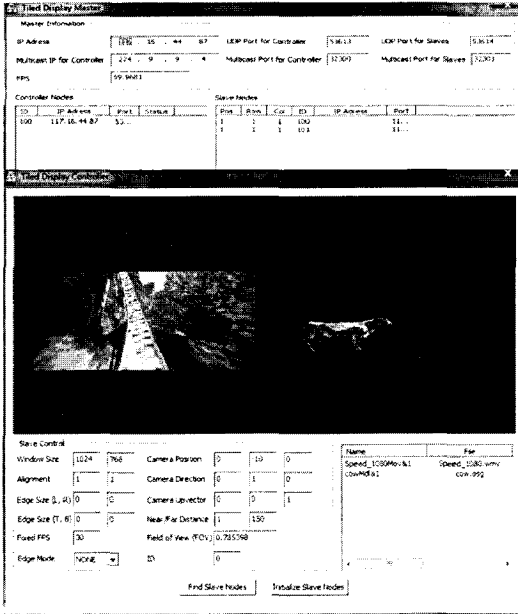


그림 3. 마스터와 컨트롤러 노드의 화면  
Fig. 3 The screen-shots of the master and controller nodes

그림 1과 2에서 보인 것처럼 하나의 타일드 디스플레이 그룹에서 동작하는 뷰어 시스템은 총 2개의 멀티캐스트 그룹을 가지고 동작한다. 마스터 노드가 실행된 후에는 각각의 멀티캐스트 그룹에 지속적으로 자신의 네트워크 주소와 포트 번호 등을 포함하는 노드 탐색 메시지를 구성해서 보낸다. 이러한 노드 탐색 메시지를 받은 슬레이브와 컨트롤러들은 각각 마스터 노드로 UDP 연결을 열고, 자신의 접속 주소와 포트 번호를 포함한 정보들을 마스터에게 전송한다. 협업용 뷰어의 마스터 노드는 각각의 구성요소들에게 고유의 식별자를 부여하고 이것을 다시 각 구성요소들에게 통보한 후에 통신을 시작한다.

네트워크 연결을 구성한 후에 협업 뷰어 마스터의 가장 중요한 역할은 렌더링 클러스터에 해당하는 각 뷰어 슬레이브들의 디스플레이 화면, 즉 렌더링 화면을 동기

화시키는 것이다. 협업 뷰어 마스터는 렌더링 클러스터들이 수행할 커맨드를 멀티캐스트 프로토콜을 통해 전송하고 해당 커맨드에 대한 완료 메시지를 기다린다. 그리고 수행 완료 커맨드를 수신 한 경우에만 다음 명령 커맨드를 전달한다. 이러한 과정을 통해서 뷰어 마스터는 네트워크로 연결된 모든 슬레이브 노드에서 동시에 같은 내용을 가진 화면을 그릴 수 있도록 명령어를 전달한다. 그리고 슬레이브 노드들과 동기화 커맨드를 지속적으로 주고받음으로써 클러스터 기반의 타일드 디스플레이 시스템이 마치 한 대의 컴퓨터에서 동작하고 재생되는 것과 같은 효과를 지원한다.

이렇게 마스터 노드는 자신이 생성한 커맨드 메시지를 슬레이브에 전달하고, 그에 대한 응답 메시지를 받으면서 동기화를 수행시키기도 한다. 그 밖에, 마스터 노드는 슬레이브와 컨트롤러 노드 사이에서 서로 주고받는 메시지를 중계해주는 역할을 담당하기도 한다. 이러한 역할을 맡음으로써 컨트롤러와 슬레이브에서는 각기 다른 그룹 노드에 대해서 알고 있어야 하는 M:N 네트워크 연결이 아니라 한 개의 노드에 대한 정보만을 가지고 데이터를 전송하는 구조를 갖게 되므로, 멀티캐스트 그룹과의 연결을 최소화하고 단순화시킬 수 있다. 그리고 컨트롤러가 여러 개 사용될 때 모든 슬레이브에게 따로 연결하고 데이터를 전송하지 않고 한 개의 마스터를 통해서 전달할 수 있으므로, 네트워크의 대역폭 사용량을 줄일 수도 있다. 또 다중 사용자가 동시에 같은 뷰나 개체를 제어하려고 할 때 나타나는 충돌 문제를 한 곳에서 관리할 수 있으므로, 협업을 지원하는 데 있어서도 편리하다.

### 2.3 협업용 뷰어 슬레이브 노드

협업용 뷰어 시스템의 슬레이브 노드는 3차원 모델, 고해상도 이미지, HD급 동영상 등의 다양한 콘텐츠를 3차원 공간에 렌더링하여 사용자에게 보여준다. 이렇게 디스플레이에 나타나는 디지털 정보들은 사용자와 상호작용 할 때에 각기 하나하나의 독립적인 뷰(View)의 개념으로 다뤄진다. 원칙적으로 슬레이브 시스템은 확장형 타일드 디스플레이 시스템의 렌더링 컴퓨터에서 독립적으로 동작해야 하며, 마스터로부터 스크린의 위치나 크기를 포함하는 부가 정보를 전달 받아서 스크린을 생성하고 삭제하기도 하며, 담당 스크린 영역을 렌더링하는 커맨드들을 수행하게 된다.

본 확장형 초고해상도 타일드 디스플레이 시스템에서는 무한의 3차원 가상공간에 큰 용량을 갖는 고품질의 콘텐츠를들 배치하고 실시간으로 화면을 이동하며 상호작용하는 것을 목표로 한다. 따라서 슬레이브에서는 대용량의 3차원 데이터를 효율적으로 렌더링 하기에 적합한 OSG 라이브러리를 활용하여 구현하였다. OSG는 화면에 나타낼 3차원 화면을 계층적 그래프의 자료 구조 형태로 추상화하여 접근할 수 있기 때문에, 개발이 용이하고 그래픽 처리 과정에서 갱신되거나 처리하지 않을 구성 요소를 효율적으로 제거하고 렌더링할 수 있어 보다 빠른 화면 재생에 유리하다.

슬레이브 노드로 구현된 뷰어는 OSG를 이용하여 lwo, obj, flt, 3ds, dxf, iv 등과 같은 다양한 3차원 모델의 재생을 지원한다. 또한 CxImage 클래스를 통해서 bmp, gif, jpg, png, ico, tga, wmf 같은 파일 포맷의 이미지를 3차원 공간에 표현할 수 있다. 그리고 본 연구에서는 타일드 디스플레이 상에서의 HD급 이상의 해상도를 갖는 동영상 재생을 담당하는 기능들을 포함하는 OSG용 신그라프 노드인 비디오 노드를 개발하였다. 이 노드는 DirectShow의 동영상 재생 기능을 활용하여 구현되었으며, 이에 따라 시스템에 설치된 코덱에 맞춰 다양한 포맷의 동영상을 재생할 수 있다. 특히 다양한 코덱을 기본적으로 지원하는 윈도우즈 7 운영체제에서는 wmv, mpg, avi, H264 등의 대부분의 동영상을 볼 수 있다. 그리고 OSG를 통해서 타일드 디스플레이에 보이는 콘텐츠 객체들의 정보와 배치 좌표 등에 관한 정보를 한 개의 모델 파일로 저장하여 관리하는 것도 가능하다.

작은 화면 여러 개가 모여서 하나의 초대형 화면을 구성하는 타일드 디스플레이의 특성을 고려할 때, 슬레이브 시스템은 분산 렌더링 기능을 말단에서 수행하는 가장 기본적인 구성단위가 된다. 슬레이브는 처음 구동될 때, 컨트롤러에 입력된 자신의 담당 렌더링 영역에 관한 정보들(윈도우와 뷰포트의 위치 및 크기, 전체 배열, 3차원 그래픽 카메라에 관한 데이터 등)과 고유 식별 번호를 마스터 노드를 통해서 수신하고 실제 뷰 영역을 생성하는 초기화 작업을 수행한다. 그리고 완전한 네트워크 구성 후에 마스터로부터 전송 받게 되는 동기화 메시지에 반응하면서, 전체 화면 중에 자신에게 분할된 구역만을 재생함으로써 분산 렌더링을 실현한다.

## 2.4 협업용 뷰어 컨트롤러 노드

마이크로소프트사의 윈도우즈 플랫폼에서 구현한 협업용 뷰어 컨트롤러는 타일드 디스플레이 네트워크를 구성할 때, 디스플레이의 개수, 디스플레이의 해상도, 가상 카메라에 관한 위치 및 각도, 슬레이브 시스템의 구성 방식(그림 3 참조) 등에 대한 정보를 제공한다. 즉, 협업용 뷰어에서 활용할 타일드 디스플레이 환경에 대한 요구사항을 사용자로부터 입력 받아 마스터 시스템으로 전송하여 해당 타일드 디스플레이 시스템을 구성하는 모든 슬레이브 노드에 반영시키도록 한다. 이러한 초기 정보를 바탕으로 하나의 협업용 타일드 디스플레이 시스템이 구성되면, 협업용 뷰어 시스템의 컨트롤러 노드는 그 후에 발생하는 모든 종류의 사용자 입력에 대하여 추상 커맨드를 생성하여 UDP 접속을 통해 마스터로 송신한다.

본 협업용 뷰어 컨트롤러의 그래픽 사용자 인터페이스에는 전체 타일드 디스플레이에 보이는 화면을 작게 축소한 뷰 윈도우가 포함된다. 기본적으로 컨트롤러 시스템에서는 이러한 제어 창을 통해서 모든 종류의 콘텐츠에 대하여 마우스와 키보드 같은 전통적인 입력 장비를 사용한 이동 및 크기 조절, 회전, 플레이, 정지, 추가, 삭제 등의 기본 조작을 수행하는 것이 가능하다. 또한 다양한 형태의 상호작용을 지원하기 위하여 새로운 인터페이스 장치의 SDK를 활용하여 입력에 대한 트리거 모듈을 확장해서 구현할 수 있다. 그리고 이것을 협업용 뷰어 컨트롤러와 연결하는 것만으로 쉽게 새로운 입력 장치를 활용하는 인터랙션을 제공할 수 있다.

타일드 디스플레이를 통한 효율적인 증강 협업 환경의 구성을 위해서 협업용 뷰어 컨트롤러 시스템은 여러 사용자가 서로 다른 자신의 콘텐츠를 동시에 활용하면서 다른 사용자의 콘텐츠에 영향을 주지 않고 서로 다른 입력을 통한 개별적이면서도 통합적인 인터랙션이 수행되는 상황을 가능하도록 지원한다. 본 연구의 타일드 디스플레이 프레임워크에서는 이러한 다중 사용자의 인터랙션 환경을 실현하기 위해 여러 개의 다중 컨트롤러를 함께 구동시킨다. 이 때, 각기 고유의 식별 정보를 부여 받은 개별 사용자의 컨트롤러 노드는 뷰어 시스템의 마스터노드를 통해서 관리 및 통제된다. 마스터는 슬레이브 노드들과 뷰를 연동할 때, 개별 사용자의 입력을 선택적으로 구분하여 해당 컨트롤러와 묶여진 뷰의 객

체에 전달하는 사용자 중심의 제어 서비스를 제공한다. 각 컨트롤러는 마스터로 전송하는 커맨드를 통해서 자신이 조작하는 콘텐츠를 변경할 수 있다.

### Ⅲ. 협업용 뷰어 시스템을 활용한 타일드 디스플레이 프로토타입 개발

본 연구의 프로토타입 구축 과정에서는 빠른 처리 속도와 고성능 그래픽 하드웨어를 요구하는 실시간 3차원 그래픽 기술의 특성을 고려하여, 한 대의 PC가 최대 두 대의 LCD 디스플레이와 연결될 수 있도록 출력 기능을 제한하였다. 그림 4와 5에서 보이는 타일드 디스플레이 구성과 같이, 실제로 구축된 협업용 뷰어의 타일드 디스플레이 프로토타입 시스템에서는 인텔 쿼드코어 2.4GHz CPU를 사용하는 6대의 슬레이브 컴퓨터와 각각 1920x1200의 해상도를 제공하는 24인치 LCD 12대를 붙여서 최대 7680x3600 픽셀의 초고해상도 대형 디스플레이 환경을 제공한다. 마스터와 컨트롤러 노드는 2.66GHz의 i7 CPU를 사용하는 컴퓨터에서 동시에 수행되고, 마스터와 컨트롤러 그리고 슬레이브 노드들은 모두 기가비트 LAN으로 연결되었다.

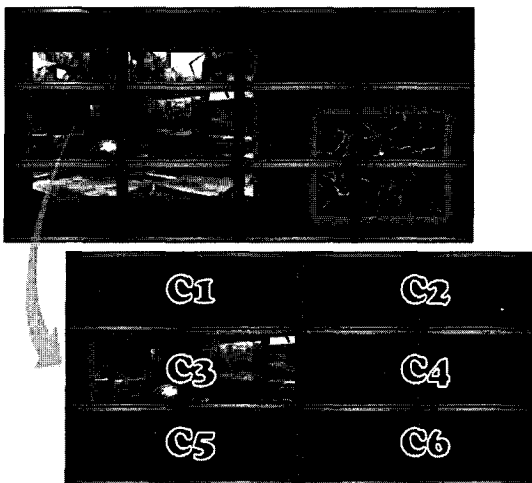


그림 4. 듀얼 그래픽 출력을 사용하는 분산 렌더링의 예시

Fig. 4 A example of the distributed rendering using the dual graphics outputs

그림 4는 3x4 개의 디스플레이 장치로 구성된 타일드 디스플레이를 보여준다. 앞에서 언급했듯이, 본 시스템에서 각 렌더링 클러스터 PC에서 동작하는 한 개의 협업용 뷰어 시스템의 슬레이브는 듀얼 그래픽 출력 기능을 이용하여 동시에 2대의 디스플레이를 재생한다. 따라서 그림 4에 전체 화면은 실제 6개의 뷰어 슬레이브 시스템이 담당하며, C1~C6에 해당하는 6개의 렌더링 영역으로 구분된다. 구체적으로 3번째 슬레이브 컴퓨터는 디스플레이 구성상 2행 1과 2열에 해당하는 C3의 렌더링 영역을 담당하는 것을 그림에서 볼 수 있다.

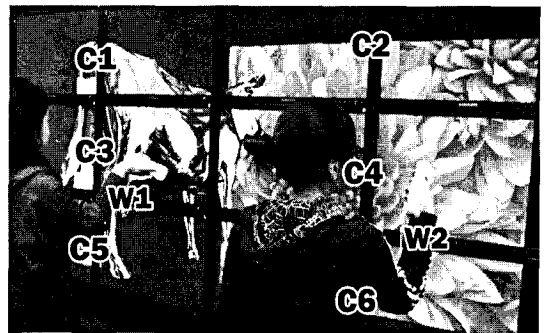


그림 5. 협업용 뷰어 시스템을 이용한 고해상도 확장형 타일드 디스플레이 프로토타입

Fig. 5 The prototype of high-resolution scalable tiled display using cooperative viewer system

위와 같은 타일드 디스플레이 환경을 구성할 때는 컨트롤러의 그래픽 사용자 인터페이스에서 윈도우와 뷰포트의 크기를 3840x1200으로 지정하고, 슬레이브 시스템의 구성 방식(배열)을 3x2로 설정한다. 만약 하나의 슬레이브 시스템이 한 개의 모니터만 재생한다면 즉 12개의 슬레이브 컴퓨터로 구성된 3x4개의 타일드 디스플레이를 구성한다면, 윈도우와 뷰의 크기를 1920x1200으로, 슬레이브 시스템의 구성 방식을 3x4로 지정하면 된다. 협업용 뷰어 시스템에서는 이처럼 컨트롤러에 입력되는 협업 환경의 초기 정보를 변경하는 것으로 1x1과 같은 단일 디스플레이부터 MxN의 확장 타일드 디스플레이 까지 다양한 크기와 해상도를 갖는 협업 환경을 손쉽게 구축할 수 있다.

본 프로토타입에서는 닌텐도의 Wiimote 단말기를 통한 인터랙션을 지원한다. 그림 5는 두 명의 사용자가 협업용 뷰어 시스템으로 구축된 확장형 초고해상도 타일

드 디스플레이에서 서로 다른 미디어 개체를 3차원 작업 공간에 배치하고 동시에 각자 자신이 소유한 Wiimote 장치로 조작하는 모습을 보여준다. 전체 영역에 걸쳐 나타난 3차원 모델을 회전 중인 왼쪽 사용자의 W1 컨트롤러는 마스터 시스템을 통해서 C1~C6의 전체 렌더링 슬레이브 집합과 연동된다. 그러나 오른쪽 영역에 치우친 주홍색 꽃 이미지에 대해 크기를 조절 중인 사용자의 W2 컨트롤러는 마스터 시스템을 통해 C2, C4, C6의 일부 슬레이브 노드의 렌더링 영역에 대해서만 연동됨을 보인다.

본 타일드 디스플레이 시스템에서는 이렇게 개인화된 뷰와 개체를 본인만의 입력 장비를 통해서 사용할 수 있도록 지원한다. 그리고 그 뷰의 위치를 타일드 디스플레이 안에서 사용자가 원하는 대로 자유롭게 옮기고, 크기를 줄이거나 늘릴 수 있다. 따라서 본 시스템에서 사용자는 다른 사용자와 관계없이 개인적인 스크린 공간을 할당해서 인터랙션 하며 사용할 수 있다. 또 필요하다면, 뷰의 화면을 타일드 디스플레이의 전체를 사용하도록 만들고, 다른 사람과 함께 보면서 한 사람이 제어하는 형태로 협업하는 것도 가능하다.

#### IV. 결 론

본 연구에서는 타일드 디스플레이 개발에 적용하여 손쉽게 초고해상도의 대형 화면을 통한 효과적인 증강 협업 환경을 구축할 수 있도록 지원하는 타일드 디스플레이 시스템과 협업용 뷰어 시스템 기술에 대해 설명하였다. 본 뷰어 시스템은 마스터, 슬레이브, 컨트롤러의 주요한 3가지 서브시스템으로 구성된다. 사용자들은 이를 통해서 고품질의 3차원 모델, 초고해상도 이미지, Full HD 급에 해당되는 동영상을 본인의 컨트롤러를 사용해서 동시에 개별 조작하거나 함께 공유하면서 작업할 수 있다.

차후에는 본 뷰어 시스템 기술을 확장하여 다중 사용자의 공용 작업 공간과 개인별 작업 공간 개념이 명확하게 분리되거나 혼재될 수 있는 보다 진보된 형태의 타일드 디스플레이 플랫폼으로 개선하는 연구와 함께, 실제 사용자들을 대상으로 사용성을 평가하는 실험을 진행할 계획이다. 그리고 현재 사용되고 있는 마스터 노드를 중심으로 하는 메시지 중계 방법에 대한 부하 검사를 진

행해서 협업에 참여하는 사용자들이 늘어날 때 마스터 노드가 병목현상을 일으키지는 않는 지에 대한 네트워크 성능 테스트를 진행할 것이다.

#### 참고문헌

- [ 1 ] S. Greenberg and M. Rounding, "The Notification Collage," in Proceedings of CHI, pp 514-521, 2001.
- [ 2 ] K. Park, J. Leigh, A. Johnson, Y. Cho, "Design of Awareness Interface for Distributed Teams in Display-Rich Advanced Collaboration Environments," Lecture Notes in Computer Science 3198, pp. 105-120, 2004.
- [ 3 ] E. M. Huang and E. D. Mynatt, "Semi-public displays for small, co-located groups," in Proceedings of CHI, pp. 49-56, 2003.
- [ 4 ] L. Chan, Y. Wei-Shian, S. Liao, Y. Tsai, J. Hsu, Y. Hung, "A Flexible Display by Integrating a Wall-Size Display and Steering Projectors," Lecture Notes in Computer Science 4159, pp. 21-31, 2006.
- [ 5 ] C. Henze, C. Levit, D. Ellsworth, T. Sandstrom, "The hyperwall: a multiple flat-panel display wall for high-dimensional visualization" Technical Report ARC-15037, NASA Ames Research Center.
- [ 6 ] 박경신, "고해상도 가시화를 위한 테이블탑 타일드 디스플레이 시스템 개발 및 적용에 관한 연구", 한국해양정보통신학회논문지, Vol. 13, No. 10, 2009.
- [ 7 ] G. Humphreys, M. Houston, R. Ng, R. Frank, S. Ahern, P. D. Kirchner, J. T. Klosowski, "Chromium: a stream-processing framework for interactive rendering on clusters" Proc. of 29th Annual Conf. on Computer Graphics and Interactive Techniques, pp.693-702, 2002.
- [ 8 ] S. Eilemann, R. Pajarola, "The Equalizer Parallel Rendering Framework" Technical Report IFI-2007.06, Department of Informatics, University of Zürich.
- [ 9 ] N. Harish, P. Narayanan, "Garuda: A Scalable Tiled Display Wall Using Commodity PCs" IEEE Transactions on Visualization and Computer Graphics, Vol. 13, Issue 5, pp. 864-877, 2007.

[10] B. Jeong, R. Jagodic, L. Renambot, R. Singh, A. Johnson, J. Leight, "High-Performance Dynamic Graphics Streaming for Scalable Adaptive Graphics Environment" Proc. of IEEE Information Visualization Workshop, 2005.

[11] 조용주, 김석환, "인터랙티브 타일드 디스플레이 응용프로그램 개발을 위한 프레임워크", 한국해양정보통신학회논문지, 제13권 1호, pp. 37-44, 2009.

[12] D. Burns, R. Osfield. Open Scene Graph, 2002. <http://www.openscenegraph.org/> visited at 28.10.2009.

[13] D.C. Schmidt, "ACE: an Object-Oriented Framework for Developing Distributed Applications" Proc. of 6th USENIX Tech. Conf., 1994.



조용주 (Yongjoo Cho)

1993년 일리노이대학  
컴퓨터과학과 학사  
1997년 일리노이대학 전기전자  
컴퓨터과학과 공학석사

2003년 일리노이대학 컴퓨터과학과 공학 박사  
2004년 ~ 현재 상명대학교 디지털미디어학부 부교수  
※관심분야: 가상현실, 인터랙티브 컴퓨팅,  
인터랙티브 학습 환경

저자소개



김민영 (Minyoung Kim)

2009년 상명대학교  
디지털미디어학부 학사  
2009년 ~ 현재 상명대학교 대학원  
컴퓨터과학과 석사과정

※관심분야: 가상현실, 모바일 인터페이스, 타일드 디스플레이



이빛나라(Binnara Lee)

2005년 상명대학교  
소프트웨어학부 학사  
2007년 상명대학교 컴퓨터과학과  
석사

2007년~2008년 (재)그래픽스연구원 연구원  
2008년~ (주)리얼타임비주얼  
※관심분야: 가상현실, 시물레이션