
협업 환경 디스플레이와 지능형 사용자 인터랙션을 위한 그래픽 유저 인터페이스 설계

Design of Graphic User Interface for Collaborative Display and Smart User Interaction

고수진, Sujin Ko, 김종원, JongWon Kim
광주과학기술원 정보기전공학부, 네트워크미디어 연구실

요약 빠른 인터넷의 보급과 빈번하게 발생하는 원격지 구성원간의 협업으로 인해서 쉽고 편리하게 이용 가능한 다자간 협업 환경의 필요성이 대두되었으며 현재 이와 관련한 연구가 활발히 진행되고 있다. 커뮤니티 컴퓨팅 환경에서의 원격 협업을 지원하기 위한 구성 기술로는 먼저 지능형 원격협업을 지원할 수 있는 시스템 및 네트워킹 구조를 설계하는 기술, 다양한 종류의 입력 채널의 내용을 동시에 보여줄 수 있는 디스플레이를 제공하는 기술, 협업 구성원 간의 자연스러운 데이터 및 태스크 마이그레이션 기반 기술, 그리고 다양한 사용자 인터랙션을 제공하는 기술 등이 있다. 그러나 무엇보다도 사용자 편의성 면에서는 위치적인 이질감을 극복하고 현장감 있는 공동 작업을 지원할 수 있는 사용자 인터랙션 기술이 매우 중요하다고 할 수 있다. 본 논문에서는 이러한 인터랙션을 지원하기 위한 다양한 인터랙션 기술을 소개하며 사용자의 인터랙션이 발생했을 경우 사용자의 의도에 따라 변경된 협업 환경, 특히 공통 디스플레이 화면의 변경 정보를 사용자 컨트롤 머신에도 Graphic User Interface(GUI) 기술을 통해 제공하는 방안이 논의되어 진다.

핵심어: Collaboration system, User Interaction, Interactive Display, Interaction GUI

1. 서론

컴퓨팅 환경에서의 원격 협업은 사람들의 대내외적인 활동 영역이 확대됨에 따라 필수적인 시스템으로 인식되게 되었다. 다자간 원격협업을 지원하는 지능형 협업 시스템을 구축하기 위해서는 유연한 서비스 요소들을 주어진 커뮤니티 환경에 부합하도록 하는 프레임워크의 정립이 요구된다. 이를 위해서 기본적으로 요구되어지는 기술은 원격협업을 지원할 수 있는 시스템 및 네트워킹 구조를 설계하는 기술, 다양한 종류의 입력 내용이 동시에 보여 질 수 있는 지능적인 디스플레이를 제공하는 기술, 다중 사용자가 동시에 사용할 수 있는 사용자 중심의 인터랙션 기술, 원격 협업 환경을 중심으로 한 데이터 및 태스크 마이그레이션 기술 등이라고 할 수 있다[1]. 그러나 무엇보다도 원활한 협업을 위한 사용자 측면의 편의성을 고려해보면 협업 환경의 다양하게 공유된 자원을 쉽게 제어할 수 있도록 사용자 인터랙션을 제공하는 것이 중요하다고 할 수 있겠다. University of Toronto 의 DGP (Dynamic Graphics Project) Lab에서 이러한 HCI(Human Computer Interaction) 연구가 선행되고 있으며[2] vision wand[3] 나 Free hand 제스처 인식 [4] 과 같은 다양한 연구가 수행되고 있다.

지능형 협업 환경 시스템에서는 이런 식으로 제공된 사용자 인터랙션을 통해서 사용자의 의도를 파악하고 이를 적절히 응용시키는 기술이 요구되어진다. 본 논문에서는 이러한 협업시스템에서 사용할 수 있는 인터랙션 기술을 소개하고 특히 사용자의 인터랙션이 발생했을 경우 인터랙션 장치로부터 전달되어지는 해당 정보를 분석하여 협업 환경, 특히 디스플레이 부분에 응용되어질 수 있는 방법을 논의한다. 이러한 경우 이 정보는 구성원 전체에게 공유되어지는 디스플레이 화면 이외에도 개별 사용자의 컨트롤 머신에서 인터랙션 결과가 반영될 수 있는 Graphic User Interface 기능을 통해 공유되어질 수 있다.

UIC(University of Illinois at Chicago) 의 Electronic Visualization Lab에서 개발한 SAGE(scalable adaptive graphics environment) [5]는 다중 입력 스트림을 지원하는 방식의 Tiled Display를 개발하고 이의 제어를 위한 다중 입력과 다수의 사용자 인터페이스를 지원한다. 본 논문은 위의 SAGE GUI를 이용하여 사용자 인터랙션이 반영되도록 기능을 확장하는 내용을 기술한다.

본 논문은 다음과 같이 구성된다. 2절에서는 인터랙션 GUI 기능을 설계하고 3절에서는 설계를 바탕으로 일부 기

능의 구현 및 검증을 보이며 4절에서는 향후 연구 방향을 언급한 후 결론을 내린다.

2. 인터랙션 GUI 기능의 설계

2.1 사용자의 인터랙션이 반영된 GUI 필요성

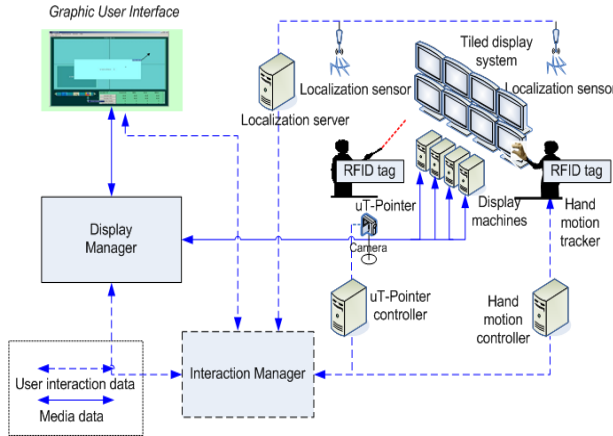


그림 1. 지능형 다자간 협업 구성 예.

그림 1은 본 GUI 설계가 목표하는 지능형 다자간 협업 시스템 구성 예를 보여준다. 해당 시스템은 지능형 Tiled 디스플레이, 디스플레이 관리자 머신, 사용자 인터랙션을 위한 포인터 장치, 포인팅 위치를 판단하는 카메라 및 컨트롤 머신, 핸드 트래킹과 관련된 장갑 및 컨트롤 머신, 사용자 위치 파악을 위한 Localization 센서들, Localization Server, RFID Tag, 그리고 모든 인터랙션 장치 등을 관리하는 인터랙션 관리자들이 있으며 이를 반영하는 GUI로 구성되어 있다.

위에서 보여지는 시스템에서 사용자 인터랙션을 위해서 제공되는 기술은 포인터 장치를 이용하여 디스플레이를 제어하는 기술과 장갑 등 손에 부착 할 수 있는 장치를 이용한 핸드 트래킹 기술, 마지막으로 Localization 센서기를 이용하여 RFID tag를 지니고 있는 사용자의 위치를 파악한 후 해당 정보를 적절히 응용시키는 기술 등이 있다.

사용자는 포인터를 이용하여 지능형 디스플레이에 특정 오브젝트를 지시하는 일반 기능 외에도 특정 동작이나 포인터 자체의 기능을 이용하여 디스플레이를 제어할 수 있는 기능을 가질 수 있다. 예를 들어 사용자가 포인터의 특정 버튼을 누르게 되는 경우 그 이후의 사용자의 동작은 일반 지시 기능으로 인식되지 않고 제어 하는 기능 모드로 인식되어 인터랙션 관리자에 의해서 특정 오퍼레이션을 수행할 수 있도록 한다. 이와 같이 해당 패턴 정보에 따라서 인터랙션 관리자는 지정된 object 의 이동, 사이즈 변경, 여러 object

간의 디스플레이 순서 변경, 또는 오브젝트의 display 중단 등의 기능을 수행시킬 수 있다.

그림 1에서 점선으로 표시된 데이터의 흐름이 사용자 인터랙션과 관련된 것으로 이렇게 추가된 사용자 인터랙션에 의한 디스플레이의 변경도 사용자 GUI 에 반영되는 것이 필요하다. 그러나 기존의 SAGE GUI 는 해당 인터랙션에 의한 제어를 고려하지 않았으므로 본 논문에서는 그 부분의 기능을 추가로 설계, 그리고 일부 구현함을 소개한다.

2.2 GUI 기능의 상세 설계

GUI 는 사용자 인터랙션의 결과로 이루어진 정보를 받아서 해석할 수 있어야 한다. 다양한 인터랙션의 결과물에 대한 규격을 맞추고 해당 메시지를 분석하여 적절하게 표현할 수 있어야 한다. 예를 들어 사용자가 포인터를 이용해서 오브젝트를 이동시켰다면 오브젝트의 이동된 위치 정보를, 사용자가 포인터를 이용해서 오브젝트의 사이즈를 변경시켰다면 해당 오브젝트의 변경된 사이즈 정보를 GUI에게 알려왔을 때 GUI 자체적으로 표현될 수 있도록 변경하고 (전달 받은 좌표 정보등을 GUI의 상대좌표로 변경할 수 있어야 한다) 사용자의 GUI 에 표현해 줄 수 있다. 그림 2는 위의 내용을 나타낸다.

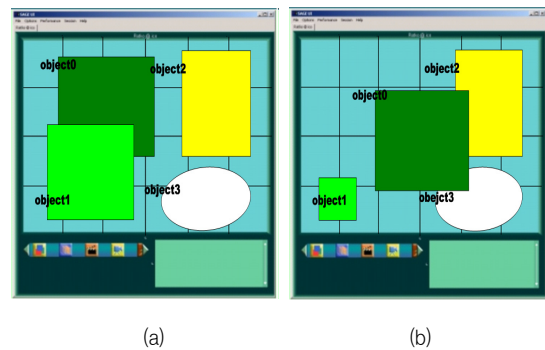


그림 2. 사용자 인터랙션이 반영된 GUI: (a) 인터랙션 발생 전; (b) 인터랙션 발생 후.

그림 2에서 (a) 는 인터랙션 발생전의 GUI 화면을 나타내고 (b) 는 발생후의 GUI 변경 사항을 보여주는 것으로 object0 는 포인터 등에 의해서 위치가 변경되고, object1 은 사이즈가 변경되었음을 보여주고 있다.

각각의 인터랙션 장치와 GUI의 인터페이스 및 기능 설계를 좀 더 자세히 언급하면 다음과 같다.

1) 포인터 장치

포인터 장치를 사용하는 사용자에게 의해 디스플레이 상의 특정 위치를 가리키고 있는 경우 포인터 컨트롤 머신에서는 해당 위치의 좌표를 전달해준다. 해당 위치의 좌표는 2차원으로 표기되어 (x,y) 형식으로 전달된다. 전달된 좌표는 디

스플레이 상에 마우스 포인터의 역할과 유사한 기능을 하는 오브젝트에 의해 표현되며, 동시에 사용자의 GUI 에도 포인터의 위치가 나타나게 할 수 있다. 이 때 표시되는 모양은 임의로 가능하고 그림 3과 같은 형태로 나타낼 수 있다. 그림 3의 object 2에 위치하는 화살표 모양의 포인터 좌표를 발견할 수 있다. 포인터 장치로부터 전달되는 위치정보는 인터랙션 관리자에 의해 적절히 보정되어진다. 예를 들어 포인터를 사용하는 사용자의 손 떨림 등으로 발생하는 노이즈 등이 지연이 발생하지 않는 수준에서 부드럽게 표현될 수 있도록 인터랙션 관리자가 보정할 수 있다. 그렇기 때문에 GUI 에서는 인터랙션 관리자로부터 전달되는 데이터를 특별한 작업 없이 바로 표현해 줄 수 있다.

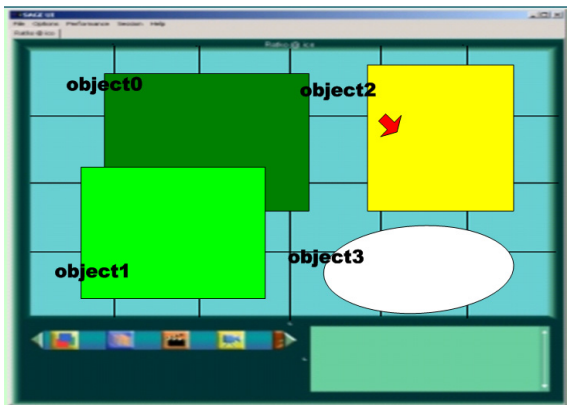


그림 3. GUI 상에 표기되는 포인터 좌표.

포인터 장치가 미리 약속된 제스처를 이용하여 단순 포인터 모드가 아닌 오퍼레이션 모드로 동작되는 경우 위의 GUI 에 표기되는 포인터는 동작을 멈추고 오퍼레이션에 의한 결과만을 반영하도록 한다.

기술된 바와 같이 GUI에 포인터 장치에 의한 인터랙션 정보가 반영되는 경우 공통 디스플레이 화면의 오브젝트들은 기존 GUI 사용자와 포인터 사용자에 의해 모두 조정 가능하게 된다. 이렇게 되면 동일한 오브젝트에 대해 두 사용자가 동시에 오퍼레이션을 수행하려고 하는 경우 충돌이 발생하며 각 사용자의 원래 의도대로 조작되지 않게 된다. 이를 방지하기 위한 방법이 필요하며 이 논문에서는 각 인터랙션 장치마다 우선 순위를 주도도록 하여 해결한다. 포인터에 의한 오브젝트 작업이 GUI 사용자에 의한 오브젝트 작업보다 우선 순위가 높은 경우 GUI 사용자는 오브젝트를 변경하려는 시도를 했을 때 동일한 오브젝트가 현재 포인터 사용자에 의해 점유되었다는 정보를 제공 받으며 해당 작업이 끝날때까지 오퍼레이션 작업은 제한을 받게 된다.

2) 핸드 글로브 장치

핸드 글로브 장치는 “핸드 모션 트래킹 서비스” 를 수행하기 위한 장치이며 이 서비스는 협업 환경의 디스플레이 화면상의 오브젝트들과의 상호작용을 위해 사용자가 자신의

자연스러운 손 동작을 이용할 수 있도록 하는 것이다[8]. 현재 제안된 시스템 내에서는 하나의 오브젝트(윈도우) 내의 3D 데이터와의 상호작용만을 지원하며 GUI는 해당 윈도우를 하나의 오브젝트로 인식, 조작하므로 다른 오브젝트들을 처리하는 절차와 차별점이 없으나 차후 핸드 글로브 장치가 전체 디스플레이상의 여러 오브젝트를 조작하도록 기능이 확대되는 경우 핸드 글로브 사용자의 오퍼레이션에 따라 변경되는 디스플레이의 결과를 GUI 에 반영해야 한다.

3) Localization 센서 장치

Localization 센서와 RFID tag를 이용하여 하나의 협업 환경 시스템 내에 함께 참여하는 사용자의 위치를 파악할 수 있다. Localization server 로부터 수신된 사용자의 위치는 인터랙션 관리자에 의해 수집되며, 적절히 보정 작업을 거친 가공 데이터가 GUI에게 전송된다. 기 설계된 인터랙션 GUI 의 경우 협업 환경 시스템의 디스플레이 화면에 맞춰서 설계된 것이므로 협업 공간내의 사용자의 위치 정보는 동일한 GUI에 표시되기 힘들며 이 때문에 별도의 Location GUI를 구성하도록 제안된다. Location GUI를 위한 초기 요구사항은 Localization server 로부터 전송된 사용자의 위치를 적절히 표시해주는 것으로 Location GUI 사용자의 요구사항에는 Location GUI 화면에 표시되는 각 사용자에 대한 상세한 정보를 나타내 줄 수 있도록 하는 것이다. 초기 버전의 Location GUI 는 그림 4와 같으며 Location GUI 에서 X로 표시 되는 그림이 현재 RFID tag를 가지고 있는 사용자의 위치로 (x,y,z) 3차원으로 표시되도록 지원한다. 그림 4에서 보여지는 상황은 한 명의 RFID tag를 가지고 있는 사용자가 공통 타일드 디스플레이 머신이 있는 협업 시스템에 있는 경우를 도시한 모습이며 해당 Location GUI 는 기계안된 인터랙션 GUI 와 같은 머신에서 함께 표시되는 것이 바람직하다. 이 논문의 나머지 부분에서 특별히 Location GUI 라는 명백한 명시가 없을 시에는 GUI라는 표기는 기계안된 인터랙션 GUI를 의미한다.

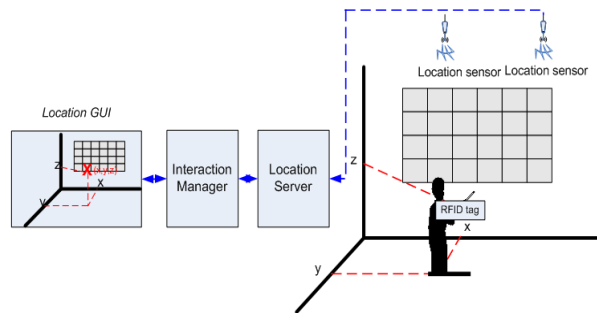


그림 4. Location GUI 개념도.

3. 인터랙션 GUI 기능의 부분 구현 및 검증

현재 구현된 GUI 는 SAGE GUI를 그림 1에서 제안된 시스템에 맞게 변경 구현한 것으로서 실제적으로 GUI와 인

터페이스를 가지고 있는 모듈은 디스플레이 관리자이다. 그림 5에서 나타내는 바와 같이 현재는 디스플레이 관리자가 인터랙션 관리자의 역할도 함께 수행하고 있다. 예를 들어 포인터를 이용한 사용자가 오퍼레이션 모드에서 오브젝트를 이동시키는 경우 해당 정보가 디스플레이 관리자에게 전송되며 해당 정보는 GUI 에 전달되어 적절히 표시되어 질 수 있다.

위에서 언급된 기능을 수행하기 위한 GUI 와 디스플레이 관리자 간의 인터페이스 및 절차는 다음과 같다. 해당 인터페이스 및 절차는 SAGE GUI 구현을 따르며 인터랙션과 관련된 인터페이스를 추가 또는 기존의 인터페이스 기능을 확장시키도록 한다[6].

1) 인터페이스

서로 전송되는 메시지 형태는 GUI 와 디스플레이 관리자를 표기하는 dst, 메시지를 구분하는 code, 오브젝트 마다 관련된 application code, 메시지마다 다른 실제 data와 해당 데이터의 size 로 구분된다.

dst	code	appCode	size	data
-----	------	---------	------	------

사용되는 code 는 다음과 같다.

- 1000 : Register UI
- 1001 : App Start
- 1003 : Move
- 1004 : Resize
- 1005 : Request Performance
- 40000 : SAGE Status Message
- 40001 : App Info Return
- 40004 : Display Info
- 40006 : App Exec Config
- 40007 : Block Object
- 40008 : Release Object
- 40009 : Pointer Info

2) 절차

그림 5의 절차는 처음 GUI 가 기동하는 시점에서부터 초기 환경 정보를 전달 받는 부분, 그리고 기타 GUI 사용자에 의한 디스플레이를 조정하는 부분을 보여준다. Register UI, Display Info, Status Message, App Exec Config 등이 초기 환경 정보를 주고 받는 부분이며, Move, Resize, Z change 등은 GUI 사용자에 의해 디스플레이 부분이 변경될 수 있도록 하는 절차를 나타낸다.

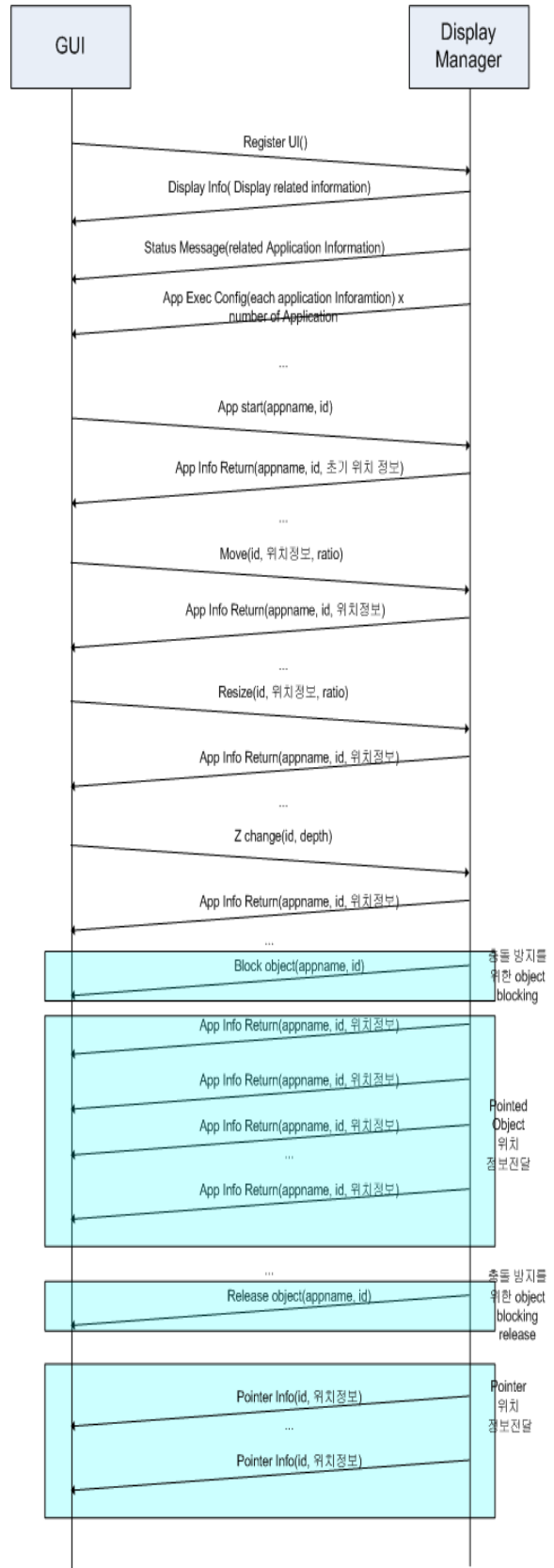


그림 5. GUI 와 Display Manager 의 Flow.

그림상의 첫 번째 박스로 표시되는 부분이 인터랙션에 의해 변경되는 오브젝트의 정보를 GUI 사용자에게 알려주어

해당 오브젝트에 대한 GUI 사용자에게 대한 오퍼레이션이 제한되도록 하는 부분이며 두 번째 박스로 표시되는 부분이 포인터나 핸드 글로브를 사용하는 사용자의 인터랙션에 의해 변경되는 정보가 GUI에게 전달되는 부분인데 해당되는 정보를 받아서 요구되는 오브젝트를 움직이도록 나타낼 수 있다. 사용자 인터랙션에 의한 오퍼레이션이 끝남과 동시에 디스플레이 관리자는 제한된 오브젝트에 관한 해제 메시지를 주어 GUI 사용자가 조정을 가능하도록 한다. 마지막에 보여지는 Pointer Info 메시지는 사용자가 포인터를 단순 지시 기능으로 이용하는 경우 해당 위치를 GUI에 전달하기 위한 인터페이스를 나타낸다.

그림 1에서 제안된 시스템의 프로토타입은 그림 6과 같으며 해당 상황에서의 GUI가 현재의 공통 디스플레이 화면과 관련하여 나타내어진다. 그림의 프로토타입은 핸드글로브를 이용한 사용자가 공통 디스플레이의 오브젝트내의 3D data를 조정하는 상황을 보여주고 있다. 현재까지의 구현 상황은 하나의 GUI만을 지원하고, 인터랙션 장비를 이용한 사용자의 오퍼레이션도 동시에 여러 개가 이루어지지 않고 한 번에 한 명의 사용자에게 의해서 한 종류의 interaction만 이루어지는 것으로 가정한다. 인터랙션 GUI는 현재 디스플레이 관리자와 인터페이스를 갖도록 구현되어 있으며 디스플레이 관리자가 인터랙션 관리자의 역할을 수행하고 있다. 현재까지는 인터랙션 정보가 반영되도록 하는 기능이 GUI에 구현되어 있지 않으며 이를 위한 기능 설계 및 인터페이스만이 구현되어 있는 상황이다.

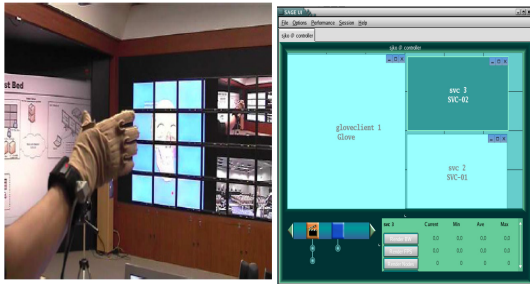


그림 6. 협업환경 시스템의 프로토타입과 GUI.

4. 추가 구현 사항

Multi-user GUI와 여러 명의 인터랙션 사용자를 고려한 GUI를 구현하고, 차후로 인터랙션 관리자가 독립적인 모듈로 분리되는 경우 해당 모듈과의 적절한 인터페이스를 설계한다. 특히 인터랙션 사용자가 여러 명이 되는 경우 각 사용자들 사이에 발생할 수 있는 충돌 등이 방지되도록 하는 절차가 필요하며 인터랙션 관리자와의 인터페이스 설계에 이 부분이 반영되어야 한다. 끝으로 Localization 장치를 이용한 사용자의 위치 파악 정보를 나타내기 위한 Location

GUI를 설계 구현하도록 한다.

- Multi-User GUI 설계: 그림 1에서 제안된 협업 시스템에서는 하나의 GUI만을 고려하고 있으며 이 경우 한 명의 사용자만이 GUI를 이용한 디스플레이 조정이 가능하다. 그러나 여러 명의, 심지어 같은 협업 시스템 공간 내에 있지 않는 사람들도 동일한 디스플레이 화면을 조정하고 상태를 파악하는 방법이 필요하며 이 방법은 이미 SAGE GUI에서 제공되어진다. 이 논문에서는 Multi-User GUI에 사용자의 인터랙션 정보가 반영되는 부분의 기능을 설계하도록 한다. 여러 명의 사람들이 동시에 GUI를 통해서 동일한 디스플레이를 조정하려고 한다면 동일한 오브젝트에 대한 오퍼레이션 수행시에 충돌이 발생할 수 있음을 추측할 수 있다. 이러한 방법은 이미 SAGE GUI 구현시 고려한 사항이므로 이에 대한 언급은 생략하는 대신 사용자의 인터랙션으로 인해 공통 디스플레이 화면에 변경된 사항을 여러 개의 GUI에 동시에 전달할 수 있도록 시스템을 설계한다. 그림 7은 Multi-User GUI를 위해 제안된 시스템 개념도이다.

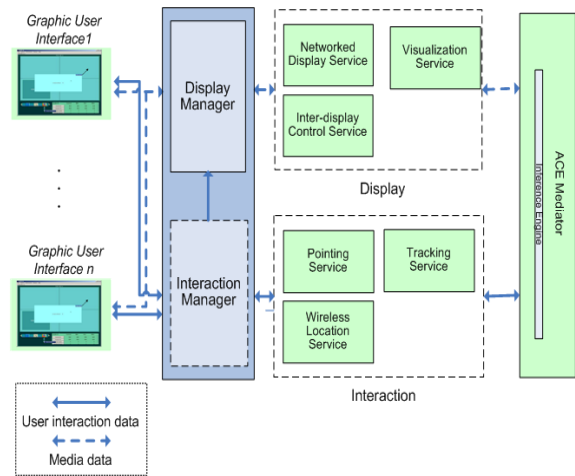


그림 7. Multi-user GUIs를 위한 시스템 개념도.

- Multi User Interaction에 따른 GUI 설계: 현재까지 기술된 사용자의 인터랙션은 암묵적으로 한번에 한 명의 사용자가 한 가지의 인터랙션을 수행하는 것을 전제로 하였으나 차후에는 여러 명, 포인터의 경우는 최대 4명까지 한꺼번에 인터랙션을 수행하는 것을 목표로 하고 있다. 결국 포인터를 가지고 있는 사용자와 핸드 글로브를 이용하는 사용자가 동시에 수행되거나, 여러 명의 포인터를 가진 사용자가 한꺼번에 인터랙션을 수행하는 경우가 발생하게 되고 이 경우 해당 정보를 처리하기 위한 인터페이스 및 충돌 방지 등을 위한 방법이 필요하다. 포인터가 단순 지시 기능이

아닌 경우, 즉 오퍼레이션 모드로 작동되는 경우와 핸드글로브가 여러 개의 오브젝트를 조정하려고 하는 경우 한 개의 오브젝트는 한 번에 한 명의 사용자에게 의해서만 점유되도록 제한하는 것이 필요하다. 또한 포인터 장치를 이용하는 여러 명의 사용자가 단순 지시 하는 기능을 이용하고 있을 경우 인터랙션 GUI는 여러 개의 포인터 좌표를 표시할 수 있어야 한다. 유사한 방식으로 포인터가 여러 명의 사용자에게 의해 오퍼레이션 모드로 작동되는 경우 여러 개의 오브젝트가 한꺼번에 움직이는 거나 사이즈가 변경되는 것 또한 표현이 가능해야 한다.

- 인터랙션 관리자와 GUI 의 인터페이스: 인터랙션 관리자는 포인터 장치나 핸드 글로브 장치, 그리고 Localization 장치등과 연계하여 공통 디스플레이 화면을 조절하거나 GUI 에게 인터랙션 정보를 전송하는 일련의 절차를 수행한다. 그림 7에서 확인되는 바와 같이 디스플레이 관리자와 인터랙션 관리자의 기능은 통합되어 구현되어질 수 있으나 점차 복잡하고 다양한 인터랙션 오퍼레이션으로 인하여 인터랙션 관리자가 분리되어 구현되어야 하는 필요성이 대두된다. 이 경우 GUI 는 인터랙션 관리자, 디스플레이 관리자와 동시에 인터페이스를 갖도록 설계되어야 한다.
- Location GUI 의 구현: Localization 장치와의 인터페이스가 확립되면 해당 정보를 적절히 표시하기 위해 2절에서 설계한 Location GUI를 구현하도록 한다.

5. 결론

본 논문에서는 지능형 협업 시스템에서 사용되어 질 수 있는 사용자 인터랙션 기능을 소개하고 해당 인터랙션 기술을 이용한 응용 결과를 사용자의 GUI를 통해서 확인할 수 있는 구현을 소개하였다. 각 인터랙션 장치 별로 GUI에서 제공되어야 하는 기능을 설계하고 고려사항을 언급한다. 즉, 포인터 장치의 경우 단순 지시 모드 인 경우 GUI 에 표현되는 방법을 제안하고, 오퍼레이션 모드인 경우 해당 정보가 GUI 에 어떻게 표현되어야 하는지에 대한 사항을 기술한다. 핸드 글로브 장치의 경우 차후 공통 디스플레이 화면의 여러 오브젝트들을 조정할 수 있도록 기능이 확장되는 경우 사용자의 의도하는 오퍼레이션이 확정되는 대로 새롭게 GUI 와의 인터페이스를 설계해야 한다. 또한 Localization 장치로부터의 정보를 표현하기 위한 별도의 Location GUI 도

제안한다.

끝으로 향후 Multi-user GUI 나 여러 명의 사용자 인터랙션을 처리하기 위해 GUI에서 고려되어야 하는 점을 언급하고 실 구현시에 인터랙션 관리자가 분리되는 경우 해당 모듈과의 인터페이스가 설계되어야 함도 명시하였다.

감사의 글

본 연구는 21세기 프론티어 연구개발사업의 일환으로 추진되고 있는 정보통신부의 유비쿼터스컴퓨팅및네트워크원천기반기술개발사업의 지원에 의한 것이다. 최 기호 이하 ACE 팀, 홍익대 박 준 교수님과 한 승용군, 송 제훈군, 마지막으로 기존의 SAGE GUI를 개발한 UIC 의 EVL lab 담당자 분들께 감사 드린다.

참고문헌

- [1] S. Han, J. Kim, K. Choi, and J. Kim, "Intergrating multiple HD video services over tiled display for advanced multi-party collaboration," in Proc. SPIE ITCOM Multimedia System and Applications IX, Jan. 2007.
- [2] DGP: <http://www.dgp.toronto.edu/>.
- [3] X. Cao, R. Balakrishnan, "VisionWand: Interaction techniques for large displays using a passive wand tracked in 3D," in Proc. ACM UIST pp. 173-182, 2003.
- [4] D. Vogel, R. Balakrishnan, "Distant freehand pointing and clicking on very large high resolution displays," in Proc. ACM UIST pp. 33-42, 2005.
- [5] L. Renambot, A. Rao, et al., "SAGE: the Scalable Adaptive Graphics Environment," in Proc. of WACE, Sep. 2004.
- [6] SAGE: <http://www.evl.uic.edu/cavern/sage>.
- [7] 최기호, 김종원, "다자간 협업환경에서 다중 고해상도 비디오와 그래픽스 가시화를 위한 Tiled-display 시스템 구현," 한국 통신 학회(KICS), 2006 하계 종합 학술 발표회, vol. 33, pp. 282-285, 2006년 7월.
- [8] H. Wang, M. Leu, C. Oz, "American Sign Language recognition using multi-dimensional Hidden Markov Models," Journal of Information Science and Engineering, vol.22, no.5, pp. 1109-1123, 2006.