

에이전트 시스템의 연구 및 개발 동향

서울시립대학교 이재호*

1. 개 요

에이전트 및 다중에이전트 시스템 분야의 연구는 초기의 분산인공지능 분야의 한정된 연구에서 고도의 분산 복합 시스템을 분석하고 설계하여 구현하는 효과적이고 실용적인 소프트웨어 개발 방법론으로까지 다양하게 적용되고 있다. 현재 객체 지향적 소프트웨어 시스템 개발 방법이 분산 시스템 개발에 중요한 개념과 기법을 제공하여 체계적인 개발 도구를 제공하고 있는 것과 마찬가지로 에이전트 중심 소프트웨어 시스템 개발 방법은 점차 단순 사용자 프로그램으로부터 전자상거래 시스템과 같은 복합적 분산 시스템 개발에 이르기까지 다양한 분야에서 분석, 설계에서 구현 및 유지 보수에 이르는 소프트웨어 개발 전과정에 걸쳐 새로운 개념을 제공하고 있다. 이러한 시스템에서 에이전트라는 개념은 자연스러우면서도 적합한 추상적 단위를 가능하게 한다. 80년대 인공지능 분야에 대한 도를 넘은 관심과 상대적으로 지나친 기대로 인하여 오히려 어려움을 겪었던 것에 비추어 에이전트 분야에 대한 지나친 관심을 경계해야 할 정도로 현재 에이전트의 잠재적 효용에 대한 기대가 큰 실정이다. 이 글에서는 에이전트 연구 및 개발 활동 전반에 대한 소개를 통하여 에이전트의 중심 개념과 논점을 제시하고자 한다.

2. 역사적 배경

에이전트에 관한 연구는 지난 반세기 동안 인

공지능 분야의 사실상의 주된 연구 과제였다고 해도 과언이 아니다. 인공지능 분야의 초창기인 50년대와 60대에 단순한 모델과 이론을 토대로 지능시스템을 구현하려던 낙관적 태도는 70년대와 80년대에 들어서 퇴색되면서 특정 기술 개발을 위한 구체적인 연구로 발전되었고 그 결과로 지식표현, 탐색기법, 게임방법론, 전문가시스템, 기계학습, 자연어처리 등 인공지능 연구는 세분 또는 전문화되었으며 그 과정 중에서 70년대 후반에 분산인공지능(Distributed Artificial Intelligence)이라는 새로운 연구 분야가 형성되어 다양한 종류의 분산인공지능 문제들을 에이전트라는 추상적 단위와 에이전트간의 상호작용을 토대로 해결하려는 시도가 시작되었다. 에이전트라는 용어는 80년대와 90년대에 걸쳐 데이터베이스나 운영체제 또는 전산망연구 등의 기존 컴퓨터 연구분야에서 사용되기 시작하였으며, 기본적으로는 컴퓨터 환경과 전산망 환경에 구애받지 않고 동일한 기능을 제공하는 소프트웨어 프로세스를 의미하였다. 90년대에 들어서 인터넷(internet)과 웹(web)의 급속한 확장과 더불어 웹 정보환경하의 소프트웨어 에이전트의 출현을 맞게되었다. 이러한 에이전트는 컴퓨터 사용자를 대신하여 주어진 일을 수행하는 대행자(proxy)나 중개자(middleware) 역할을 하며 특히 개인비서(personal assistant) 역할의 에이전트를 개발하려는 최근의 연구도 이와 직접적으로 연관된다.

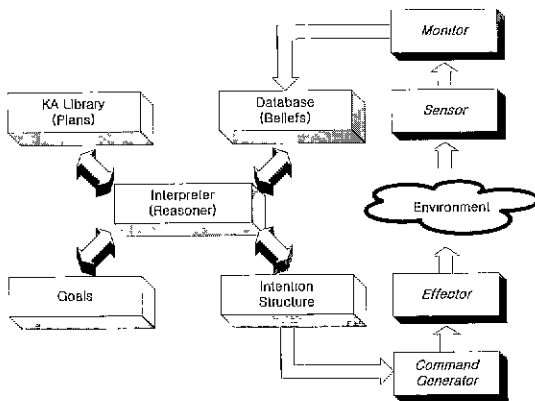
3. 에이전트의 정의

에이전트를 소개하면서 당면하는 첫 번째 문

* 정회원

제는 에이전트가 무엇인가를 정의하는 문제이다. 에이전트라는 용어 외에도 에이전트기반 시스템(agent-based system)과 다중에이전트 시스템(multi-agent system) 용어 또한 정의할 필요가 있다.

에이전트에 대한 정의는 많이 있지만 공통적으로 에이전트란 누군가를 위하여 무엇인가를 대신 해주는 컴퓨터 프로그램을 통칭한다. 좀더 구체적으로 정의하면 에이전트는 지속적으로 환경에서 지각된 것과 내부 지식을 바탕으로 추론하며 그 결과 행동하여 환경에 영향을 미치고 또한 사용자를 포함한 다른 에이전트와 의사 소통하는 지속적으로 존재하는 소프트웨어 요소라고 할 수 있다[3]. 따라서 에이전트는 일반적으로 다음 그림에 보여진 바와 같이 주변환경(environment)으로부터 감지기(sensor)를 통하여 지식을 얻어 내부에서 추론하여 환경에 작용하는 구조를 가지고 있다. 아래 그림은 에이전트 구조 중에서 잘 알려진 Procedural Reasoning System (PRS)의 구조를 보여주고 있다[8,9].



Jenning과 Wooldridge의 좀 더 학술적인 정의에 따르면, 에이전트란 특정 환경(environment) 내에 위치(situated)하여 에이전트의 설계된 목적(objectives)을 만족시키기 위하여 자율적(autonomous)으로 유연하게(flexible) 행동할 능력이 있는 컴퓨터 시스템이다[2]. 환경 내에 위치하였다는 것은 환경을 지각하고 에이전트의 행동을 통하여 환경에 영향을 미친다는 앞의 정의와 서로 일치한다. 이러한 환경이란 물리적 환경뿐만 아니라 인터넷과 같은 인위적 환경도 모

두 포함한다. 자율적이란 단순히 사용자나 다른 에이전트의 직접적인 간섭 없이 행동을 결정할 수 있는 능력을 일컫는다. 적극적인 의미로서는 경험을 통하여 학습할 수 있는 능력을 말하기도 한다[3]. 유연성은 [4]에 논의된 바에 의하면 (1) 환경에서 감지된 변화에 적시에 반응할 수 있는 능력과, (2) 단순히 환경의 변화에 반응하는 것 외에 필요에 의하여 기회를 이용하여 목표를 달성하는 방향으로 행동하는 능력, 또한 (3) 사회성이 있어 다른 에이전트나 사용자와 상호 작용하면서 자신의 문제를 해결할 수 있을 뿐만 아니라 다른 에이전트도 도울 수 있는 능력을 일컫는다.

이러한 에이전트의 정의를 기초로 다른 용어들도 추가로 정의할 수 있다. 에이전트 기반 시스템(agent-based system)은 시스템 설계 및 구현에 에이전트가 중심 개념으로 사용된 시스템이다. 이는 객체지향 방법론에서 객체가 중심 개념이 되는 것과 같은 이치이다. 다중에이전트 시스템(multi-agent system)은 상호 작용하는 다수의 에이전트를 이용하여 설계하고 구현된 시스템으로서 문제에 대한 서로 다른 시각과 해결방식을 갖고 있는 다수의 에이전트로 구성된 문제를 표현하기에 적합하다. 이에 대한 연구는 현재 소프트웨어 시스템 개발 관련 분야 중에서 가장 활발한 활동이 이루어지고 있는 분야라 해도 과언이 아니다. 이러한 배경에는 여러 가지 이유가 있지만 그 중에서도 가장 중요한 것은 에이전트 시스템의 특징인 자율 시스템으로서 다른 에이전트와 상호작용(interaction)을 통하여 각 에이전트의 목적을 달성할 수 있는 기능이 바로 소프트웨어 설계자의 요구와 자연스럽게 일치하기 때문인 것으로 여겨진다. 공통적인 상호작용 중으로는 (1) 공동의 목표 실현을 위하여 같이 일하는 협력(cooperation) 관계, (2) 에이전트간의 유해한 상호작용은 피하고 상호 유익한 상호 작용을 도모하는 공조(coordination) 관계, (3) 관련된 모든 에이전트들이 수용할 만한 에이전트간의 일치된 합의점에 도달하고자 하는 협상(negotiation) 관계 등을 생각할 수 있다.

4. 에이전트 기반 소프트웨어 시스템 개발

에이전트 및 다중에이전트 시스템에 관한 연구는 에이전트 기반 소프트웨어 시스템 개발에 도움이 되는 직접적인 연구 성과를 거두는 단계에 도달한 상태이다. 특히 소프트웨어 시스템 모델링, 시스템 구조, 협력 작업 지정 언어, 분산 시스템 표준화 등의 분야에서 가시적인 효용성을 보이고 있다. 이에 따라 에이전트 기법은 대규모 분산 소프트웨어 시스템 개발을 위한 차세대 개발 방법으로서 주목받고 있다. 반면에 에이전트 기법의 성공적 적용을 위해서는 효율적인 소프트웨어 공학적 방법으로 발전시켜 초기의 소프트웨어 시스템 설계에서부터, 구현, 검증, 유지 보수에 이르는 전 소프트웨어 생명주기에 효과적으로 적용되어야 하는 과제를 해결해야 한다.

소프트웨어 공학 연구를 통하여 시스템 구성 요소간의 상호작용(interaction)이 소프트웨어 개발의 난이도를 결정짓는 가장 중요한 요소임이 파악되었다. 소프트웨어 구조 중에서 단순한 실행경로와 명확한 입출력 구조를 가진 시스템에 비하여 독자적인 실행경로(thread)를 가지고 다른 구성요소와 복잡하게 상호작용을 하는 시스템은 정확하고 효율적으로 분석하고 구현하기에 훨씬 힘든 것이다. 그렇지만 이러한 특성은 실생활에 필요한 많은 소프트웨어 시스템이 갖고 있는 공통적 특징이기 때문에 소프트웨어 공학 분야에서는 구성요소간의 상호작용을 올바르게 이해하고 손쉽게 모델링하여 효율적으로 구현하는 것이 주요 연구 대상이 되어왔다. Unified Modeling Language(UML)로 대변되는 현재의 객체지향 설계방법론도 이러한 추세를 반영하여 에이전트 기법을 표준화하고 도입하려는 노력이 진행되고 있다.

5. 응용분야

에이전트 기법은 실험실 연구 단계에서 실제계의 산업 및 상업용 응용 단계로 급속히 영역이 확대되고 있다. 앞 절에서 살펴본 에이전트 기반 소프트웨어 공학도 대표적 응용사례라 할 수 있다. 이 절에서는 지금까지 알려진 산업용, 상업용, 오락용, 의료용으로 구분하여 응용 분야와 대표적 사례를 살펴보고 이에 따라 앞으로의 응용 분야와 전망을 가늠해 본다.

◆ **산업체 응용** : 산업체 응용은 제조, 공정 제어, 통신, 항공관제, 물류 분야 등을 꼽을 수 있다. 제조 분야에서는 세부적으로는 생산물형상관리, 협력 설계, 공정 스케줄링 등의 분야에서 다중에이전트를 이용한 예 들을 볼 수 있다. YAMS[5] 시스템은 단조, 주조, 연마 등의 생산 과정을 작업세포(workcell) 단위로 분리하여 이들을 다시 계층적으로 조직하여 조립, 페인팅 등의 상위 작업세포를 구성하고 분산인공지능에서 잘 알려진 Contract Net Protocol[6]을 이용하여 생산 공정을 효율적으로 관리하는 방안을 제시한다. 생산 공정은 생산 품목, 가용 자원, 시간 제약 등의 끊임없이 변하는 매개변수들로 정의되며 작업세포 간의 복잡한 상호 관계를 다중에이전트를 이용하여 나타낸다. 즉 각각의 작업세포들을 에이전트로 표현하고 각각의 에이전트들로 하여금 독자적인 작업계획과 능력을 갖추도록 한 후에 Contract Net Protocol을 이용하여 계층적으로 구성된 작업세포들에게 작업을 부여하도록 하였다.

공정제어 분야 또한 공정 제어기들을 자율 분산적 에이전트로 구성하기에 적합하기 때문에 에이전트 기법이 활발히 응용되는 분야이다. 통신업 분야에서는 가입자에게 보다 빠르고 안정적인 서비스를 제공하기 위한 경쟁에서 차별화되기 위해서는 대규모 분산 통신망에 접속된 요소들을 실시간에 효율적으로 감시하고 관리하는 것이 필요하다. 대규모 분산성은 다중에이전트를 응용하기 적합한 분야이며 실제로 다중에이전트 간의 협상을 통하여 가입자가 원하는 부가서비스를 효과적으로 제공하는 기술이 개발된 바 있다.

항공관제 분야에서도 항공기와 다양한 관제용 제어기들을 자율적 에이전트로 표현하여 이들 간의 상호작용으로 문제를 해결하도록 하는 시험적 운용의 예가 있다. 물류 분야 역시 지역적 분산성으로 인하여 에이전트 기반 시스템이 적합한 것으로 평가되고 있다.

◆ **상업적 응용** : 산업체 응용이 특정 목적을 위한 고난도 맞춤형 시스템인 것과는 대조적으로 상업적 응용은 대부분 대규모 시장을 바탕으로 하고 있다. 세부적으로는 정보관리, 전자상거래, 업무절차관리 등을 들 수 있다. 정보관리 분

야는 인터넷의 대중화로 대두된 정보의 홍수로 야기된 정보과부하(Information Overload) 문제의 해결로 집약된다. 정보과부하 문제는 정보여과(information filtering)와 정보수집(information gathering)의 양면적 문제로 나누어진다. 정보여과는 쏟아지는 방대한 양의 정보 중에서 실제로 관련된 정보만을 걸러내는 문제이며 정보수집은 특정 문제 해결에 필요한 정보를 여러 곳에 산재한 정보들로부터 찾아 모으는 문제이다. 이러한 문제 해결을 위한 에이전트 이용 방법은 이미 주위에서 검색엔진이나 인터넷 관련 정보가공 기술에서 찾아볼 수 있다. 정보관리는 앞서 정의한 에이전트의 특징 중에서 사용자의 요구를 대신하여 자율적으로 해결하는 에이전트의 특징과 밀접한 관련이 있다. 정보 여과의 예로서는 사용자의 특성을 관찰하여 전자메일을 우선 순위를 결정짓거나 삭제하고 전달하고 분류하고 보존하는 에이전트 시스템이 있으며, 정보수집의 예로서는 사용자의 재무관리를 위하여 상시 주식시장의 지수들을 감시하고 재무 관련 뉴스 및 분석자료, 회사들의 수익보고서 등의 정보를 수집하여 사용자의 질의에 대답하고 나아가서 사용자가 관심을 가질만한 정보를 미리 제공하는 에이전트 시스템을 예로 들 수 있다.

전자상거래 분야에서는 현재 대부분 사람과 사람간의 거래가 주종을 이루고 있다. 즉 전자상거래 시스템의 사용자가 직접 상품을 선정하고 가격을 비교하여 구매한다. 이러한 과정이 사용자를 대신하는 에이전트에 의해 자동화될 수 있으며 점차 응용 분야가 확대되고 있다. 인터넷 쇼핑몰에 있는 다양한 상품 가격과 서비스 정도를 비교하여 상품을 선정하거나 사용자가 지정한 조건 내에서 자동으로 경매에 응하는 에이전트들을 예로 들 수 있다.

업무절차관리의 관심은 여러 부서에서 제공한 정보와 판단을 근거로 관리자가 적합한 업무 결정을 할 수 있도록 하는 데 있다. 올바른 판단을 내리기 위해서는 모든 관련 최신 정보를 종합적으로 이용하는 것이 바람직하지만 현실적으로 대규모 사업체의 모든 관련 부서의 최신 정보를 획득하여 일관성 있게 이용하는 것은 쉬운 일이 아니다. 따라서 많은 기관에서 정보기술(Infor-

mation Technology)을 이용한 업무절차관리에 관심을 기울이고 있다. 에이전트 시스템에서는 업무절차를 에이전트간의 협상과정으로 파악하여 각각의 에이전트에게 직책(role)이나 부서를 대표하게 하고 다른 에이전트로부터 협상을 통하여 필요한 서비스를 제공하거나 제공받는 형태로 표현하여 업무절차를 관리하는 시스템을 개발한 예와, 이동에이전트(mobile agent)를 이용하여 부서간의 원활한 업무 협조가 이루어지도록 구성된 시스템들을 예로 들 수 있다.

◆ **오락산업 분야** : 새로운 세기와 더불어 여가산업의 중요성이 세삼 인식되고 있으며 소프트웨어 분야에서도 오락산업의 역할이 증대하고 있다. 개인 및 인터넷 게임, 영화 및 양방향 텔레비전의 사업 등을 예로 들 수 있다. 게임 특히 인터넷 게임의 등장 인물들을 에이전트로 표현하면 가상으로 주어진 환경의 변화를 스스로 인식하여 학습하고 자율적 추론으로 행동을 결정하는 현실감 있는 게임을 구성하기에 적합하다. 영화 산업 및 사용자 인터페이스 분야에서 각광받는 캐릭터 에이전트 등도 에이전트의 기법을 응용하기에 적합한 분야라 할 수 있다.

◆ **의료 분야** : 의료 분야에 적용된 에이전트 대표적 시스템은 환자상태 감시 분야와 진료 시스템을 들 수 있다. 환자상태 감시를 목적으로 만들어진 GUARDIAN[7]의 경우 외과수술후의 중환자 상태를 대상으로 한다. 이 시스템은 환자의 상태를 돌보기 위해서 각 분야의 전문가로 구성된 팀을 시스템 내부에 구현한 형태를 가지고 있으며 팀원간의 효과적인 정보 교환을 통하여 협동 감시 작업이 가능하도록 구성하였다. 흥미롭게도 감시업무를 담당하는 에이전트들을 기능별로 분리하였는데 이 기능들이 에이전트를 구성하는 기본 요소들과 거의 일치한다. 즉 지각과 행동을 담당하는 에이전트들이 감지기(sensor)와 실행자(effector)를 통하여 외부 세계 작용하며, 추론 담당 에이전트들이 판단작업을 하고, 제어 에이전트가 있어 시스템 전체를 제어한다.

진료 시스템은 환자 관리 업무를 포함한 진료 체계 전체를 대상으로 하고 있으며 분야 별 전문가들간의 의사소통 및 정보 교환을 원활하게 하는 통합시스템이다. 이러한 시스템 역시 관련

개개인을 에이전트로 표현하고 이들 간의 조직 및 상호작용을 에이전트 시스템으로 표현하기에 적합한 응용 분야라 할 수 있다.

6. 전망과 결론

에이전트 기반 응용 시스템의 효용성은 앞 절에서 나열한 응용 분야들과 이를 위해 개발된 시스템들을 통하여 입증되었다고 할 수 있다. 앞 절에서 언급된 분야들은 모두 실생활에서 요구되는 시스템들이며 다른 실생활에 필요한 많은 소프트웨어 시스템들도 에이전트 시스템으로 표현하기에 적합한 비슷한 성격을 가지고 있을 것을 볼 수 있다. 소프트웨어 공학적 측면에서 보면 에이전트 시스템이 제공하는 개념, 즉 자율적 문제 해결 능력이 있는 다수의 개체들의 상호작용(interaction)을 통하여 독자적인 목표와 시스템 전체 목표를 달성한다는 개념은 객체지향 시스템을 구성하는 객체 개념보다 진일보한 매우 유용한 개념이라고 할 수 있다. 그럼에도 불구하고 아직까지는 에이전트 개념을 이용한 시스템 구축이 용이한 것은 아니다. 전통적 연구 분야인 분산시스템이나 병렬시스템에서 발생하는 문제를 해결해야 할 뿐만 아니라 에이전트의 유연성 및 자율성, 에이전트간의 상호작용에서 비롯되는 추가적인 문제까지 해결해야 하기 때문이다. 이런 이유로 아직까지 개발된 에이전트 시스템들은 에이전트 전문가에 의해서 개발되어 온 현실이다. 에이전트 시스템의 잠재력이 반영된 시스템의 대중적 개발을 위해서는 따라서 에이전트 시스템을 구성하는 명확한 방법론의 개발과 개발에 직접 응용될 상업제품 수준의 에이전트 개발 도구들이 필요하다. 이를 위한 노력들이 현재 다방면에서 진행되고 있으며 예를 들어 객체지향 시스템 설계에 이용되고 있는 UML의 기능을 확장 보완하여 에이전트 시스템 설계에 적합한 새로운 표준을 만들려는 노력을 들 수 있다.

기술적인 면 외에도 사회적 수용성을 고려할 필요가 있다. 에이전트란 기본적으로 사용자를 대신하는 역할을 한다. 사용자를 대신하는 에이전트를 사용자가 신뢰할 수 없다면 에이전트로 구성된 시스템 자체를 신뢰할 수 없게 될 것이다. 이를 해결하기 위한 노력도 기술적 문제와

함께 논의되어야 할 사항이다.

지난 역사적 배경과 현실적 소프트웨어 수요를 감안할 때 다가오는 시대에서 에이전트 기술의 중요성을 새삼 강조할 필요가 없다. 에이전트 관련 연구는 분산시스템, 객체지향 시스템, 소프트웨어 공학, 인공지능, 경제학, 사회학, 인지과학 등의 분야를 망라하는 아이디어가 모두 적용될 수 있는 다분야 통합적 연구 분야라 할 수 있다. 상업화된 일부 에이전트 관련 제품들이 이미 시장에 등장하고 있지만 소프트웨어 시스템 설계 및 구현의 새로운 방향을 대변하는 에이전트 기법에 대한 종합적이고 체계적인 연구가 필요하다라고 할 수 있다.

참고문헌

- [1] Michael N. Huhns and Munindar P. Singh, editors. Readings in Agents, chapter 1, pages 1-23. Morgan Kaufmann, 1997.
- [2] Nicholas Jennings, Katia Sycara and Michael Wooldridge. Roadmap of Agent Research and Development. *Autonomous Agents and Multi-Agent Systems*, vol. 1 pp. 7-38, 1998.
- [3] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, 1995.
- [4] Michael Wooldridge and Nicholas R. Jennings. Intelligent Agents: Theory and Practice. *The Knowledge Engineering Review*, vol. 10(2) pp. 115-152, 1995.
- [5] H. Van Dyke Parunak. Manufacturing Experience with the Contract Net. In M. Huhns, editor, *Distributed Artificial Intelligence*. Pitman Publishing, pp.285-310, 1987.
- [6] Randall Davis and Reid G. Smith. Negotiation as a Metaphor for Distributed Problem Solving. *Artificial Intelligence*, vol. 20 pp. 63-100, 1983.
- [7] Barbara Hayes-Roth, Micheal Hewett, Richard Washington, Rattikorn Hewett, Adam Seiver. *Distributing Intelligence*

within an Individual. In Les Gasser and Michael N. Huhns, editors, *Distributed Artificial Intelligence Volume II*. Pitman Publishing, pp. 385-412, 1989.

1. Anand S. Rao and Michael P. Georgeff. BDI agents: From theory to practice. In *Proceedings of the First International Conference on Multiagent Systems*, pages 312-319, San Francisco, California, June 1995.
2. Jaeho Lee, Marcus J. Huber, Edmund H. Durfee, and Patrick G. Kenny. UM-PRS: an implementation of the procedural reasoning systems for multirobot applications. In *Conference on Intelligent Robotics in Field, Factory, Service, and Space (CIRFFSS'94)*, pages 842-849, Houston, Texas, March 1994.
3. Patrick G. Kenny, Clint R. Bidlack, Kari C. Kluge, Jaeho Lee, Marcus J. Huber, Edmund H. Durfee, and Terry Weymouth. Implementation of a reactive autonomous navigation system on an outdoor mobile robot. In *Association for Unmanned Vehicle Systems Annual National Symposium (AUVS-94)*, pages

233-239, Detroit, MI, May 1994. Association for Unmanned Vehicle Systems.

4. Edmund H. Durfee, Marcus Huber, Michael Kurnow, and Jaeho Lee. TAIPE: Tactical assistants for interaction planning and execution. In *Proceedings of the First International Conference on Autonomous Agents (Agents '97)*, pages 443-450, Marina del Rey, California, February 1997.

이 재 호



1985 이학 학사 서울대학교 계산통계학과
 1987 이학 석사 서울대학교 계산통계학과
 1997 공학 박사 미시간대학교 전산공학(미국 앤아버 소재)
 1996. 8 ~ 1998. 2 수석엔지니어, ORINCON(미국 샌디에고 소재)
 1998. 3 ~ 2000. 3 전임강사, 서울시립대학교 전자전기공학부
 2000. 4 ~ 현재 조교수, 서울시립대학교 전자전기공학부

E-mail jaeho@ee.uos.ac.kr

• 2000년도 컴퓨터그래픽스 춘계 워크숍 •

- 일 자 : 2000년 5월 20일
- 장 소 : 중앙대학교
- 주 최 : 컴퓨터그래픽스연구회
- 문 의 처 : 중앙대학교 컴퓨터공학과 윤경현 교수
 Tel. 02-820-5308