

---

# 개선된 모바일 P2P 네트워크 구현에 관한 연구

김용훈\* · 박종민\*\* · 조범준\*

## A Study on Networks Applications improved Mobile P2P

Yong-Hun Kim\* · Jong-Min Park\*\* · Beom-Joon Cho\*

### 요 약

P2P(Peer-to-Peer)는 중앙서버 없이 개인 간에 직접적인 연결을 통하여 디지털 자원(mp3, software, game, video 등)을 효과적으로 공유하는 방식으로 기존의 클라이언트/서버 방식과 달리 관련 프로그램만으로 디지털 자원에 대한 손쉬운 공유 및 교환이 가능하다. 본 논문에서는 이 제안기법을 이용한 P2P 응용 프로그램은 최초 실행할 때 하나 이상의 IP 주소를 입력받아야 하지만, 실행 이후에는 사용자로부터 어떠한 정보도 받지 않으면서 네트워크 그룹에 지속적으로 연결될 수 있다. 제안기법의 평가를 위해 시뮬레이션을 통해 연결되어진 피어들이 종료할 경우 네트워크 그룹에 재연결할 때까지 걸리는 시간을 측정하였다.

### ABSTRACT

A P2P is a method that can share and exchange on digital resources through a direct connection on personnel without a central server. In this paper, Although the P2P application that uses our proposed framework should obtain one or more IP addresses of the neighbor peers manually, after instantiate, the application can do its job while maintaining connection to the network continuously and automatically. To evaluate our proposed scheme, we measured and analyzed the time for a peer to reconnect to the network when the mediating peer fails and the network isolation occurs.

### 키워드

Mobile, P2P, SFA, Network

## I. 서 론

인터넷은 한 컴퓨터에서 다른 컴퓨터 혹은 컴퓨터들의 네트워크를 통해서 정보에 접근하고자 하는 필요에 의해 발전 되었다. 초기의 인터넷은 단말기를 통해 데이터를 접근할 수 있도록 해 주는 소수의 컴퓨터들로 구성된, 규모가 별로 크지 않은 네트워크였다. 점차적으로 이 개념은 무한한 가능성을 열었고, 인터넷은

전 세계를 연결하고 사용자들이 정보를 공유, 접근 그리고 자유로운 통신을 할 수 있는 수백만의 컴퓨터들을 포함하는 글로벌 네트워크로 발전했다. 그러나 최근 들어 인터넷은 새로운 도약기를 맞이하고 있다. 움직이는 네트워크의 필요성이 대두 되고 있다. m-Commerce란 모바일 장비들의 모바일 애플리케이션을 사용하여, 무선으로 여러 가지 행위를 할 수 있는 능력을 말한다[1]. m-Commerce는 개인 주소록 동기화와 같

---

\* 조선대학교  
\*\* 동신대학교

은 매우 단순한 것에서부터 신용 카드 업무와 같은 복잡한 업무에 이르기까지 광범위의 개념으로서, 행위의 주체에 따라 일반 개인 소비자 기반의 m-Commerce와 사업체 기반의 m-Commerce로 구분된다.

특히 오늘날 사업체 기반의 m-Commerce를 주도하고 있는 모바일 SFA(sales force automation)로 보험회사와 제조 및 유통업체를 중심으로 빠른 속도로 보급되고 있다. 그러나 현재의 모바일 SFA는 상업적으로는 매우 다양하게 적용되고 있으나, 개인 사용자들 사이의 모바일 환경에서는 아직까지 별다르게 적용되고 있지 않다는 문제점이 있다. 특히, 모바일 기기 간 정보 공유를 위해 항상 서버와 통신해야 하고, 그로 발생하는 통신 선로상의 패킷 단위의 금액 결제가 이루어지고 있다. 따라서 본 연구에서는, 모바일 사용자 간의 효율적인 데이터 공유를 위해서 모바일 P2P 서비스의 도입을 실험하였다. 모바일 P2P 서비스란 SFA 서버의 중개 없이, 서버에 접속해 있는 한 모바일 클라이언트에서 바로 다른 모바일 클라이언트로 다양한 정보 및 데이터를 전송하는 방식으로, 종래의 P2P 개념을 모바일 기반으로 확장한 것이다.

## II. 관련연구

### 2.1 모바일 SFA

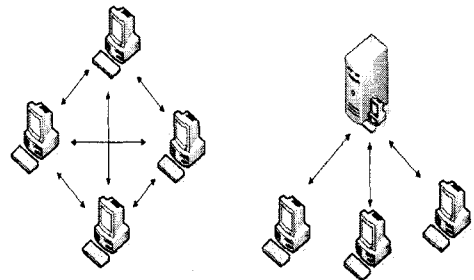
모바일 SFA는 종래의 SFA를 모바일 기술기반으로 이전한 것을 말하며, 그 개념은 영업 및 마케팅 활동과 관련된 각종 하드웨어와 소프트웨어 애플리케이션들을 다양하게 결합하여 사용함으로써 수작업으로 진행되어온 영업활동을 전자적인 과정으로 전환시키는 것이다[2]. 모바일 SFA는 기존 데스크탑 애플리케이션이 충족시키지 못한 ‘장소와 시간의 초월’이라는 한계를 극복하고자 하는 영업 분야의 요구와 모바일 단말기의 급속한 보급, 이동통신과 인터넷의 융합, 3세대 이동통신으로의 융합이라는 모바일 비즈니스 환경의 변화가 맞물려, 최근 들어 급속히 발전하고 있다[3].

### 2.2 모바일 P2P 구조

그림 1은 기존 네트워크 응용 프로그램의 대표적 구조인 클라이언트/서버 구조와 P2P 응용 프로그램의 구조의 차이를 나타낸 것이다.

클라이언트/서버 구조에서는 서버와 클라이언트의 명확한 구분이 있고 서버는 서비스 제공의 역할만 수행하고, 클라이언트는 서비스를 제공받는 역할을 수행하며 클라이언트와 클라이언트 사이에는 직접적인 데이터 교환은 이루어지지 않으며 서버를 통해서만 이루어진다. 데이터의 전달은 클라이언트의 요구와 서버의 응답 방식으로 이루어진다.

그에 비해 P2P 시스템 구조는 peer들 각각이 서로에게 서비스 제공자이면서 또한 서비스를 이용하는 사용자의 기능을 동시에 수행하는 응용프로그램 구조이다. 즉 peer가 서버의 역할을 수행함과 동시에 클라이언트의 역할을 수행하는 것으로 데이터의 전달은 양방향이다. 데이터의 흐름을 비교해 볼 때 클라이언트/서버 구조는 주로 서버에서 클라이언트로의 데이터 흐름이 대부분을 차지하였으나 P2P 구조는 서로 간에 데이터를 교환하기 때문에 어느 한 peer로 트래픽이 집중되거나 방향성을 띄지 않고 peer들 사이에 자유로운 데이터 교환이 이루어지는 것이다.



(a)P2P 구조 (b) client/server 구조  
그림 1. client/server 구조 및 P2P 기본적 구조  
Figure 1. Structure of client/server and basic structure of P2P

### 2.3 P2P 응용프로그램의 분류 및 특징

현재 P2P 응용 프로그램으로 인식되는 네트워크 응용프로그램의 형태는 크게 두 가지로 분류할 수 있다. 하나는 메신저 응용 프로그램이고[7, 8] 다른 하나는 파일 공유 응용 프로그램이다[9~12]. P2P 응용프로그램들에서 사용되는 연결구조 또한 다양한 형태를 띄우고 있는데 이들의 대표적인 시스템 구조는 Central Arbiter Type과 Pure Distributed Type의 두 가지로 볼 수 있다. Central Arbiter Type의 가장 대표적인 것으로 메신저 프로그램들로 이는 중앙의 서버가 peer 들의

등록정보를 관리하고 있으면서, 어떤 peer 가 P2P 네트워크에 접속하게 되면 이 peer 와 관련 있는 다른 peer 들에게 그 peer의 접속을 알려줌으로써 peer들 간에 통신을 가능케 하는 구조이다. Pure Distributed Type 은 Gnutella II와 같은 몇몇 파일 공유 프로그램들에서 사용하는 방식으로는 중앙의 관리 서버 없이 모든 peer 들이 서로 대등한 관계로 연결되는 구조이다. 이러한 분산 구조에서는 중앙의 관리 서버가 없기 때문에 트래픽이 한 곳에 집중하지 않고 P2P 네트워크가 견고하게 형성되는 장점은 있으나 서로 간에 데이터 전송이나 peer를 검색하는 기능에 있어서는 효율성이 떨어진다. 대부분의 P2P 응용프로그램들은 이 형태를 그대로 따르고 있거나 두 형태를 혼합한 hybrid 형태를 취하고 있다.

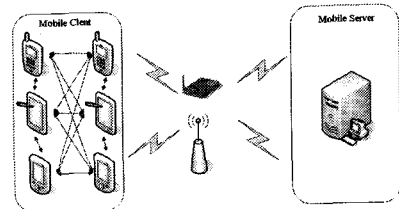
#### 2.4 모바일 P2P 서비스

P2P란 인터넷 기반 하에 연결된 양측의 클라이언트가 중앙의 서버를 경유하여 연결되거나 직접적으로 연결되어 양측 클라이언트의 자원을 공유하여 상호간의 이익을 도모하는 것을 말 한다[5]. 최근에 이루어진 모바일 단말기의 보편화, LBS 서비스의 확산[4], 유무선 연동 기술의 발전 등과 같은 변화는, 이러한 유선 P2P 서비스가 모바일로 이전될 수 있는 충분한 여건을 조성하고 있으며 이에 따라 마이크로소프트사의 메신저 서비스와 같은 모바일 P2P 서비스가 서서히 그 모습을 드러내고 있다. 그러나 현재까지 P2P를 위한 표준화된 기반구조와 아키텍처가 존재하지 않기 때문에, 현실화된 모바일 P2P 서비스는 아직까지 많지 않다[6]. 현재 모바일 P2P 서비스는 IM(instant messaging) 중심으로 발전하고 있으며, 이것이 파일 공유 및 다른 분야로 확산되어 가는 추세이다. 마이크로소프트사에서는 메신저 서비스 프로토콜을 중심으로 P2P 서비스를 제공하고 있으며, SUN에서는 모바일 IM서비스를 위해 J2ME WMA API를 제공하고 있다. 이외 WLAN과 Wi-Fi 네트워크에서 사용 가능한 JXTA ME(JXME) API를 제공하고 있다. 이러한 기술들을 토대로 한 모바일 P2P 서비스가 다양한 모바일 환경에 도입될 경우, 사용자 간의 직접적인 데이터 공유를 가능하게 함으로써 서버 부하를 줄일 수 있으며, 더 나아가 보다 안정적인 시스템 운영이 가능하게 된다.

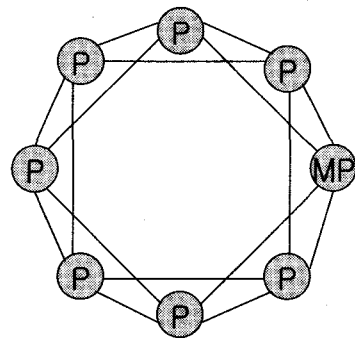
### III. 시스템 설계

#### 3.1 시스템 구성

프로토타이핑한 모바일 P2P 네트워크 시스템은 그림 2(a)와 같이 유무선 통합 기능을 담당하는 모바일 SFA 미들웨어와 데이터 정보, 사용자 정보, 기타 등록 정보의 기능을 수행하는 모바일 클라이언트로 구성된다. gCRM 엔진과 Agent 및 DB Manager로 구성되는데 gCRM 엔진은 서버에 저장된 사용자 위치 정보의 분석을 통해 클라이언트에 디스플레이 한다. Sync 엔진은 유무선 통합을 위한 Sync 메커니즘을 제공하며, Messaging 엔진은 이메일과 SMS등 메시지 전송 메커니즘을 제공한다. gCRM 엔진은 위치 기반 데이터의 기본 통계 분석과 고객 정보 분석의 역할을 담당하며, Agent Manager와 DB Manager는 각각 P2P 서버 기능과 DB 제어 기능을 제공한다. 그림 2(b)에서는 네트워크 환경을 보인다.



(a) 모바일 SFA 시스템 구성도



(b) 네트워크 환경

그림 2. 모바일 SFA 시스템 구성도 및 네트워크 환경  
Figure 2. System diagram of mobile SFA and network environment

### 3.2 모바일 P2P 에이전트

모바일 P2P 에이전트는 모바일 기기 간의 직접적인 네트워킹을 기반으로 IM이나 파일공유 등의 작업을 수행하는 소프트웨어 에이전트를 말한다. 이 에이전트는 사용자의 개입 없이 한 모바일 Peer에서 필요로 하는 데이터 정보를 보유한 다른 Peer를 검색하고, 그 Peer에 직접 접속하여 CDMA나 WLAN을 통해 필요한 데이터를 다운로드 받으며, 이 데이터를 Peer의 데이터와 동기화시킨다. P2P 에이전트는 일반적 P2P 서비스의 중앙집중식 모델과 유사하게, P2P 서비스를 위해 하나의 중앙 검색 서버를 필요로 접속한 사용자 정보를 관리하고, Peer들에서 요청하는 데이터를 보유한 다른 Peer의 ID를 Index DB에서 검색하며, 이를 위해 필요한 콘텐츠 정보를 관리한다.

이러한 기능의 수행을 위해 서버의 Agent Manager는 모바일 클라이언트에 내장된 SmartAgent를 이용하고, SOAP, UDDI, XML 등과 같은 표준 프로토콜을 이용하여 통신한다.

## IV. 시스템 구현

### 4.1 개발 환경

본 논문에서 구축한 모바일 P2P 시스템은 휴대폰, PDA, 테블릿 PC 세 가지 기기에서 함께 사용 가능하며, 미들웨어는 다양한 종류의 모바일 기기를 지원한다. 먼저 미들웨어와 PDA용 클라이언트 애플리케이션이 스마트 폰에 적용되었다. PDA용 P2P 시스템의 H/W 플랫폼으로는 서버의 경우 Intel IBM-PC 호환 기종을, PDA는 싸이버 뱅크의 스마트폰 POZ X301 기종을 채택하였다. OS는 Windows2000 Server와 PocketPC 2002 Phone Edition(PDA)을, DB는 마이크로 소프트의 MS-SQL2000와 MS-SQLCE 3.0을 사용하였다.

### 4.2 모바일 P2P 네트워크 구성

본 논문에서는 Gnutella 방식과 같은 순수 P2P 네트워크 환경에서, 다른 peer와의 연결이 끊어졌을 경우, 특정 peer에게 네트워크 소켓 연결이 집중되지 않도록 여러 peer에게 소켓 연결을 분산시킴으로써, 각 peer들이 네트워크 그룹에 항상 참여할 수 있게 하는 기법을 적용하여 모바일 P2P 네트워크를 구성하였다. Gnutella

방식은 순수 P2P 방식으로 특정 서버를 필요로 하지 않으며, 각 peer들로 하여금 서버와 클라이언트의 기능을 모두 가지게 하는 방식이다. 그림 3에서는 Gnutella 방식의 네트워크 구조를 보인다.

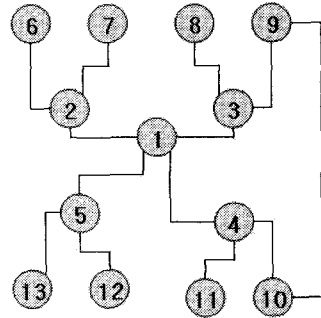


그림 3. Gnutella 방식 네트워크 구조  
Figure 3. Network structure of Gnutella system

한 peer에서 Gnutella 방식을 이용하는 프로그램이 처음 실행될 시에는 Gnutella 네트워크에 참여하고 있는 다른 하나 이상의 peer IP 주소를 입력/연결해 주어야 하며, 이러한 방식으로 모든 peer들은 피라미드식으로 연결되어 무제한의 자료를 공유하게 된다. 그러나 Gnutella 방식은 각 peer들이 자원을 검색하기 위해 다른 peer에게 메시지를 전송하는 경우, Gnutella 네트워크에 참여하고 있는 수많은 peer들에게 메시지들이 전송되기 때문에 자원 검색 속도의 저하를 일으키는 단점이 있다. 또한 프로그램에 입력된 IP 주소와의 연결이 모두 끊어지는 경우, 더 이상의 서비스를 제공받을 수 없다는 단점을 갖는다.

따라서 기존의 Gnutella 방식과 같은 P2P 네트워크 환경에서, 다른 peer와의 연결이 끊어졌을 경우, 특정 peer에게 네트워크 소켓 연결이 집중되지 않도록 여러 peer에게 소켓 연결을 분산시킴으로써 각 peer들이 네트워크 그룹에 항상 참여할 수 있게 하는 기법을 적용함으로써 모바일 시스템 환경하의 발생 문제점을 해결하도록 하였다.

### 4.3 모바일 P2P 네트워크 구현

#### 1) 연결/요청 단계

모바일 P2P 네트워크 환경에서 각 peer들은 다른 peer에게 소켓 연결을 한 후 IP 주소 목록을 요청하게

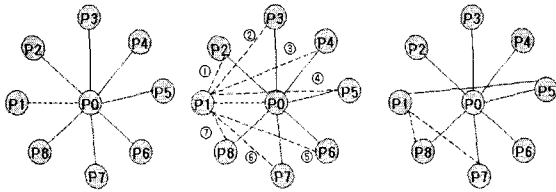
되며, 이 때, 다른 peer의 소켓 연결 시도로 인해 연결된 peer는 소켓 수가 권장 소켓 수(N\_Sockets)보다 많아질 경우 오랫동안 연결되어 있는 소켓 순으로 연결 포기 패킷을 전송한다. 또한 네트워크 소켓 수가 N\_Sockets보다 작은 peer는 소켓 수가 N\_Sockets에 도달하도록 하기 위해 IP 주소 테이블 내에서 아직 연결을 시도하지 않은 IP 주소로 소켓 연결을 시도한다. 연결/요청 단계에서는 IP 주소 테이블 내의 모든 IP 주소에 이미 연결 시도를 하였을 경우, 지금까지 연결을 시도했던 모든 IP 주소들에 대해 연결 시도가 없었던 것으로 설정한다. 그러나 IP 주소 테이블 내에 연결 시도를 하지 않은 IP 주소가 존재한다면 연결/요청 단계 중인 peer는 연결 시도를 하지 않은 것으로 표시된 엔트리를 획득한 후, 엔트리 내의 IP 주소로 연결을 시도하게 되며, 해당 엔트리의 Try 플래그 값을 TRUE로 설정한다. 또한 각 peer들은 다른 peer로의 접속이 실패할 경우 해당 엔트리의 NumFail 플래그 값을 1만큼 증가시키지만, 접속이 성공할 경우에는 접속된 peer에게 REQ\_AD 메시지를 전송하고, 연결된 소켓 핸들을 해당 엔트리의 S\_Handle에 저장한다. 각 peer들은 연결된 소켓의 수가 N\_Sockets보다 많아질 경우 KeepSock 플래그 값이 TRUE로 설정되어 있는 엔트리를 얻어 온 후, FALSE로 세팅하며 상대방 peer에게 소켓 연결을 끊어도 좋다는 RESET 메시지를 전송한다. 이 때 상대방 peer가 소켓 연결을 종료할 경우 연결이 끊어지게 된 peer는 해당 엔트리의 KeepSock 플래그 값을 확인하여 FALSE로 설정되어 있다면 연결/요청 단계를 재실행하지 않는다. 예를 들어 MAX\_FAIL은 연결이 실패할 경우의 최대 연결 시도 횟수를 의미하며, 시스템에 값을 미리 주어지는 것으로 한다.

### 2) 메시지 응답 단계

REQ\_AD, REP\_AD, 그리고 RESET 메시지를 수신한 peer의 작업 절차는 다음과 같다. REQ\_AD 메시지를 받은 peer는 메시지를 송신한 peer에게 연결된 peer들의 IP 주소들이 포함된 REP\_AD 메시지를 전송하며, REP\_AD 메시지를 받은 peer는 메시지 내의 IP 주소들을 IP 주소 테이블에 추가한다. RESET 메시지를 받은 peer는 연결되어 있는 네트워크 소켓의 수가 N\_Sockets보다 많을 경우, RESET 메시지를 보낸 peer와의 소켓 연결을 끊는다.

### 3) 시나리오

본 시나리오에서 각 peer들은 연결 시도 주체에 따라 연결을 시도하는 peer와 다른 peer로부터 연결되는 peer로 분류할 수 있으며, 특정 peer가 다른 peer로 접속할 경우의 모습을 그림 4에서 보인다. 그림 4의 (a)에서, 각 peer들을 연결하는 선은 peer들 간 네트워크 소켓 연결을 의미하며, peer P1이 peer P0에게 접속하기 전에 peer P0은 7개의 소켓, peer P2, P4, P6, P8은 5개의 소켓, peer P3, P5, P7은 4개의 소켓을 생성하고 있는 모습을 보인다. peer P1은 실행할 때 프로그램 내에 미리 지정된 peer P0으로 소켓 연결을 시도한다. peer P1과 연결된 peer P0은 이미 연결되어 있는 다른 peer들의 IP 주소들을 peer P1에게 전송하게 되고, peer P1은 peer P0이 송신한 peer P2, P3, P4, P5, P6, P7, P8의 IP 주소들을 수신하게 된다. 그림 4의 (b)에서, peer P0으로부터 주소 정보를 수신한 peer P1은 peer P2부터 peer P8까지 네트워크 소켓 연결을 시도하게 되며, peer P1이 ①~⑦ 순서로 각 peer들에게 접속함을 보인다. 이후 네트워크 소켓 연결 수가 N\_Sockets보다 많아질 경우 각 peer들은 가장 오래된 네트워크 소켓 연결 순으로 RESET 메시지를 보내게 된다. 그림 4의 (c)에서는 RESET 메시지를 받은 peer들의 소켓 수가 N\_Sockets보다 많을 경우, 연결을 끊은 상태를 보이며, 각 peer들은 소켓의 수가 N\_Sockets보다 작거나 또는 RESET 메시지가 오지 않은 peer와의 소켓 연결만을 유지하게 된다. 또한 RESET 메시지를 수신한 peer P2, P4는 연결 소켓의 수가 N\_Sockets 수보다 많기 때문에 peer P0과의 네트워크 소켓 연결을 끊게 된다. 그림 4의 (c)에서 실선은 peer P3, P5, P7의 소켓 수가 N\_Sockets이기 때문에 더 이상의 소켓 수를 줄일 수 없다는 의미를 나타내며, 점선은 peer P6, P8의 소켓 수가 N\_Sockets보다 많기 때문에 peer P6, P8이 peer P1에게 RESET 메시지를 보낸 상태를 나타낸다. 이후 peer P1의 네트워크 소켓 수가 N\_Sockets보다 많아지게 된다면 peer P1은 peer P6과 P8과의 연결을 우선적으로 제거한다.



(a) 최초 연결 단계 (b) peer 연결 단계 (c) 소켓 정리 단계

그림 4. 접속 후 연결 3단계(N\_Sockets:5)  
Figure 4. 3 steps after connecting

4) 비교분석

각 peer들이 다른 peer에게 접속함으로써 네트워크 그룹에 참여하는데 걸리는 평균시간을 측정하였다. peer의 수는 50개로 하였고 결과는 표 1에서 보인다.

표 1. N\_Sockets와 소요되는 시간  
Table 1. N\_Sockets working time

N_Sockets	기존	제안
3	6~17초	4~7초
5	2~3초	1~2초
10	2~3초	1~2초

V. 결론

본 논문에서는 모바일 P2P 서비스를 이용하여 보다 개선된 모바일 P2P 구현 기법을 제안하였다. 본 제안 기법을 이용한 peer들은 모바일 P2P 네트워크 환경을 기반으로 하여, 연결된 peer로부터 얻은 또 다른 peer들의 IP 주소들을 목록으로 관리함으로써, 연결되어 있는 peer가 종료하여도 목록으로 유지된 다른 IP 주소로 연결을 유지할 수 있게 된다. 본 논문을 통해 제안된 모바일 P2P 서비스 기술은, 모바일 사용자 간의 정보 공유가 신속하게 이루어지게 함으로써, 시간과 비용을 절약하며 모바일 시스템의 효율성은 기존의 방법보다 향상 시킨다.

참고문헌

[1] Shin, G. and Shim S.S.Y.; A Service Management Framework for M-Commerce Applications, Mobile Networks and Applications, Vol.7, pp. 199-212, 2002.  
 [2] Rivers & Dart; The Acquisition and Use of Sales Force Automation by Mid-Sized Manufacturers, The Journal of Personal Selling & Sales Management, 1999.  
 [3] 한국소프트웨어진흥원; 모바일 비즈니스를 위한 애플리케이션 사업자 전략, 한국소프트웨어진흥원, 2002.  
 [4] 김서균, 이준석; LBS 시장 및 관련업계 동향 분석, 정보화기술 동향분석, 제 9권 제 4호, 한국전자통신연구원 정보화기술연구소, 2003.  
 [5] 컨설팅베이; P2P-차세대 인터넷의 대안, (주)컨설팅베이 발표자료, 2000.  
 [6] 전현성, 조용중, 박천구; 세상을 바꾸는 힘의 중심 P2P, 프로그램세계, 2002.  
 [7] MSNMessage, <http://messenger.msn.co.kr>  
 [8] YahooMessage, <http://www.yahoo.co.kr>  
 [9] Napster, <http://www.napster.com>  
 [10] Gnutella, <http://gnutella.wego.com>  
 [11] 소리바다, <http://www.soribada.com/>  
 [12] eDonky2000, <http://www.edonkey2000.com>

저자소개



김용훈(Yong-Hun Kim)

2005년 조선대학교 컴퓨터공학과 졸업(박사과정수료)

※관심분야 : 패턴인식, 정보보호 및 보안

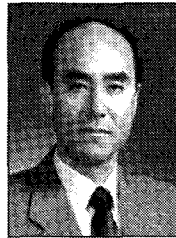


**박종민(Jong-Min Park)**

1988년 조선대학교 산업공학과  
졸업(공학석사)  
2005년 조선대학교 컴퓨터공학과  
졸업(공학박사)  
2004년~ 현재 동신대학교 컴퓨터  
학과 겸임교수

2001년~ 현재 매크로테이터(주) 이사

※관심분야 : 생체인식, 패턴인식, 인공지능, 정보보  
호 및 보안



**조범준(Beom-Joon Cho)**

1980년 조선대학교 전기공학과(공  
학사)  
1988년 한양대학교 전기공학 과  
(공학박사)  
2004년 한국과학기술원 전자전산  
학과(공학박사)

1980년~ 현재 조선대학교 전자정보공과대학 컴퓨터  
공학부교수

2002년~ 현재 한국멀티미디어학회 부회장

※관심분야 : 인공지능, 패턴인식, 뉴로컴퓨터