

# 홈 환경에서 사용자상태를 고려한 유비쿼터스 음악방송 시스템의 설계와 구현

권영우<sup>1</sup>, 이석희<sup>2</sup>, 김종원<sup>1\*</sup>, 우운택<sup>2</sup>  
광주과학기술원 정보통신공학과  
네트워크 미디어 연구실<sup>1</sup> 및 U-VR 연구실<sup>2</sup>  
{ywkwon, sheelee, jongwon, wwoo}@kjist.ac.kr

## Design of Ubiquitous Music System Incorporating Listener State in Home Environment

Young-Woo Kwon<sup>1</sup>, Soekhee Lee<sup>2</sup>, JongWon Kim<sup>1</sup>, Woontack Woo<sup>2</sup>  
Networked Media Lab.<sup>1</sup> and U-VR Lab.<sup>2</sup>  
Dept. of Info & Comm, Kwang-Ju Institute of Science and Technology

### 요약

스트리밍 서비스의 형태로 이루어지는 음악방송 서비스는 최근 인터넷에서 가장 널리 제공되는 서비스 중 하나이다. 그러나 기존의 음악방송 서비스의 형태로는 다가오는 유비쿼터스 컴퓨팅 환경에 적절한 서비스를 제공하기 어렵다. 본 논문에서는 유비쿼터스 컴퓨팅 환경에서 다수 사용자의 상태를 각각 고려한 음악방송 시스템을 설계하고 구현한다. 제안된 시스템은 5W1H 형태의 컨텍스트를 사용하여 사용자의 상태를 추출하고, 추출된 상태에 따라 음악 서비스를 제공한다. 그리고 카메라를 사용하여 사용자를 인식하고 여러 공간에 존재하는 컴퓨터를 사용하여 지속적인 음악방송을 제공하는 시스템을 설계하고 구현한다.

Keywords : Ubiquitous computing, context-aware service, and personalized streaming service.

## 1. 서론

최근 유비쿼터스 컴퓨팅에 대한 관심이 증가하면서 이를 충족시키는 다양한 서비스의 개발에 초점이 모아지고 있다. 유비쿼터스 컴퓨팅 환경에서 사용자는 편재된 디바이스를 사용하고 필요한 서비스를 제공받을 수 있기를 원한다. 또한 자신의 변화에 따라 서비스들의 능동적인 변화를 기대한다. 이런 요구 사항들을 충족시키기 위하여 사용자 및 환경의 변화에 적응적인 서비스 개발에 대한 노력이 증가되어야 한다[1]. University of Colorado의 Adaptive house[2]는 이러한 적응형 서비스를 구현하기 위한 프로젝트이다. 하지만 Adaptive house는 홈 자동화만을 고려하고 다양한 컨텍스트들을 고려하지 않는다. 이에 따라 사용자

중심의 개인화 된 서비스는 제공해 줄 수 없다는 단점이 있다. 반면 사용자 중심의 개인화 된 서비스의 종류로 Media Unbound의 AudioInsightTM[7]와 넷스터의 Gigabeat[8] 를 들 수 있다. 이들 서비스는 사용자의 음악적 취향에 대한 사전질의를 거친 후 그에 맞는 서비스를 제공한다. AudioInsightTM은 Collaborative filtering[3,4] 기법과 전문가들에 의해 상세하게 분류된 음악 데이터베이스를 사용함으로써 조금 더 체계적으로 개인화 된 음악 서비스를 제공한다. 그러나 이런 서비스들은 사용자가 환경에 따른 자신의 기분과 취향을 직접 입력해야 한다는 단점이 있다. 이에 따라 본 논문에서는 인터넷 음악 서비스를 사용자의 입력을 최대한 줄이고 개인의 특성에 맞추어 제공하는 시스템(UMS: Ubiquitous Music System)을 설계하고 구현한다. 개인화 된 음악 서비스 시스템은 5W1H (WHO, HOW,

† Corresponding author.

WHAT, WHEN, WHERE, WHY) 형태의 사용자 컨텍스트를 사용하여 사용자의 감성에 맞는 서비스를 제공하는 Context-aware 모듈[5], 음악을 선곡하고 방송하는 서버, 그리고 음악을 재생하는 클라이언트로 구성되어 있다.

본 논문의 구성은 다음과 같다. 2 절에서 제안한 시스템의 전체 프레임워크에 대해서 설명하고 3 절에서는 Context-aware 모듈에 대해서 구체적으로 설명한다. 이어서 4 절에서는 서버와 클라이언트의 세부적인 구조에 대해서 기술한다. 5 절과 6 절에서는 실험과 그에 대한 결론 및 추후과제를 언급하며 논문을 마감한다.

## 2. 시스템 구성

UMS (Ubiquitous Music System)의 구조는 그림 1 과 같이 Context-aware 모듈, 스트리밍 서버, 스트리밍 클라이언트, 그리고 이들간의 메시지전달을 담당하는 에이전트로 구성되어 있다.

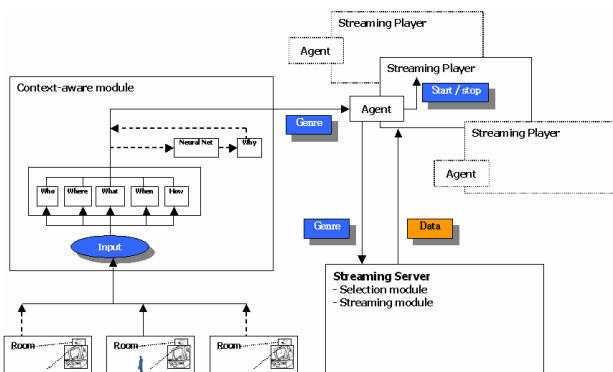


그림 1. 시스템 프레임워크.

시스템의 동작 과정은 다음과 같다. Context-aware 모듈에서 사용자를 파악하여 어떠한 서비스를 제공할 것인지 결정하고 결정된 서비스를 에이전트를 통하여 스트리밍 서버로 요청한다. 서버는 에이전트가 보내온 음악 장르에 따라 음악을 선곡하여 클라이언트에게 전송한다.

스트리밍 서버는 클라이언트의 커넥션 요청을 처리하고 요청된 서비스에 따라 음악을 선곡하는 모듈과 음악을 스트리밍하는 모듈로 구성된다. 스트리밍 클라이언트는 음악을 재생하는 재생기이며 Context-aware 모듈로부터 전달 받은 서비스의 종류에 따라 에이전트를 통하여 음악 서비스를 요청

하고 서버로부터 전송 받은 음악을 재생한다. 5WH 형태의 사용자 컨텍스트를 사용하여 서비스를 제공해주는 Context-aware 모듈은 입력센서로 USB 카메라와 컴퓨터에 내장된 시계, 키보드 등을 사용한다. 컨텍스트에 기반하여 사용자 행동을 파악하고 학습한다음 제공할 서비스의 종류, 즉 음악 장르를 결정하고 이것을 클라이언트에게 넘겨준다.

## 3. Context-aware 모듈

### 3.1 Context-aware 모듈

Context-aware 모듈은 유비쿼터스 컴퓨팅 환경이 갖춰진 가정에 적용될 경우 다양한 서비스를 제공할 수 있다. 본 논문의 시스템에서는 음악 서비스를 제공하는 시스템에 적용되어 구현되었다.

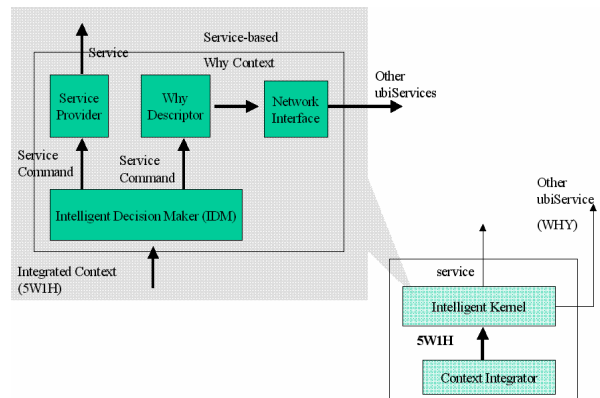


그림 2. Context-aware 모듈의 구조.

그림 2 는 Context-aware 모듈의 구조를 나타내고 있다. Context-aware 모듈은 센서들로부터 사용자의 컨텍스트 정보를 수집하는 컨텍스트 통합자 (Context Integrator)와 수집된 컨텍스트 정보를 분석하는 지능형 커널 (Intelligent Kernel) 부분으로 이루어져있다. 지능형 커널부분은 다시 5WH 형태의 정보를 받아서 실행할 서비스의 종류를 결정하는 지능형 의사결정 생성자 (Intelligent Decision Maker), 지능형 의사결정자에 의해 생성된 서비스 명령을 실행하는 서비스 제공자 (Service Provider), 사용자의 의도 및 감성정보를 추출하는 WHY 표현자 (Why Descriptor), 그리고 추출된 감성정보를 다른 Context-aware 모듈로 전달하는 네트워크

인터페이스 (Network Interface)로 나뉜다.

지능형 의사결정자는 Context-aware 모듈의 지능 부분에 해당하는 컴포넌트로서 신경망을 사용하여 만들어진다. 사용자의 의도를 컨텍스트에 기반하여 파악/학습하고 사용자 중심의 서비스를 제공한다. 지능형 의사결정자는 상황에 맞는 적절한 학습속도, 사용자의 눈에 보이지 않는 학습과정, 그리고 multi-user 와 다양한 컨텍스트 정보에 대한 수용성을 요구한다. 첫번째 요구조건을 위해서 가변적인 learning factor 를 적용하고 두 번째 요구조건을 만족시키기 위하여 실시간으로 입력패턴 및 학습패턴을 생성하고 학습하는 알고리즘을 적용한다. 그리고 C-N 테이블(Context-neural network matrix file table)을 구현하여 마지막 요구 조건을 만족시킨다.

사용된 신경망은 Feedforward 방식이고 Back propagation 을 사용하여 학습한다[6]. 처음에는 5WHH 형태의 단순한 입력 패턴이 사용되어지므로 Hebbian learning 을 사용하지만 특정 컨텍스트 정보에서는 더 이상 학습이 진행되지 않는 local minima 가 생김을 확인할 수 있다.



그림 3. 시스템에 적용된 IDM 의 구조.

입력층으로 40 개의 neural 을 사용하고 WHO, HOW, WHAT, WHEN, WHERE, WHY 각각 6 개의 neural 이 할당되는데, 그림 3 은 시스템에 적용된 지능형 의사결정자의 구조를 나타내고 있다. 표 1 은 입력층에서 사용된 센서, 컨텍스트 그리고 그에 따른 값들을 보여준다. 본 시스템의 경우에 WHAT 컨텍스트가 음악서비스로 제한되어 있고 WHY 컨텍스트는 추출해야 할 컨텍스트이므로 아무런 값이 할당되지 않는다. 은닉층은 입력층의 개수와 같이 40 개를 사용하고 출력층은 6 개를

사용한다. 처음 neural 은 서비스 제공 여부를 결정하고 나머지 5 개의 neural 은 서비스가 제공될 때의 인자를 정의한다.

표 1. 사용된 컨텍스트와 센서 그리고 그에 따른 사용 가능한 값.

Context	Sensor	Value
Who	Keyboard	user1, user2, ....
How	USB Camera	is, sit....
Where	USB Camera	lab1, lab2, lab3...
When	Computer Clock	1~24 hours

WHY 표현자는 사용자의 의도 및 감성을 나타내는 WHY 컨텍스트를 생성한다. 제안된 시스템에서는 각각의 Context-aware 모듈마다 제공되어지는 서비스 명령으로부터 WHY 정보를 생성하여 이를 필요로 하는 Context-aware 모듈과 공유하는 서비스 기반 WHY 컨텍스트로 구현된다. 예를 들어 본 논문의 시스템의 경우에는 “댄스음악을 틀어라”라는 명령을 생성하고 이를 기반으로 사용자가 음악을 듣기를 원한다는 의도정보와 ‘댄스음악을 듣고 싶은 기분’이라는 감성정보를 얻을 수 있다. 감성정보는 전등을 제어하는 Context-aware 모듈의 경우에는 유용한 정보가 될 수 있지만 본 시스템에 적용된 Context-aware 모듈에서는 아직 활용되지 않는 정보이다. 그림 4 는 시스템에서 사용된 서비스 기반 WHY 컨텍스트를 보여준다.

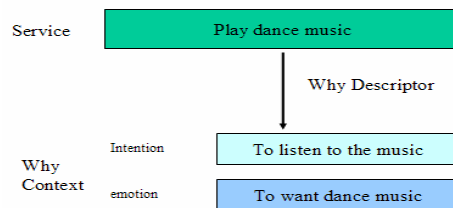


그림 4. 음악서비스를 위한 WHY 컨텍스트.

본 논문의 시스템에 적용된 Context-aware 모듈은 새로운 컨텍스트가 들어왔을 때 C-N 테이블에 등록하고 신경망을 초기화 시키기 때문에 기존의 컨텍스트를 기반으로 새로운 컨텍스트에 부합하는 서비스를 추론하는 능력이 부족하다. C-N 테이블을 U-N 테이블(User-Neural network matrix file)로 바꾸고 같은 사용자의 다른 컨텍스트 일 때는 입력패턴의 개수를 증가시켜 학습시키는 방법은 이 문

제에 대한 해결책이 될 수 있을 것이다.

### 3.2 사용자 식별 및 감지

Context-aware 모듈은 5W1H 형태의 사용자 정보를 입력 받기 위해 여러 센서들을 사용한다. 본 논문의 시스템에서는 사용자를 감지하기 위하여 USB 카메라를 사용한다. 사용자가 감지되면 Context-aware 모듈에게 사용자, 현재시간, 위치 등과 같은 정보를 전달해 사용자가 원하는 음악 장르의 음악들을 서버로부터 제공받는다. 시스템은 움직임 감지방법을 사용하여 사용자의 존재를 인식하고 WHERE 정보를 Context-aware 모듈에게 전달한다. 본 시스템에서는 ‘is’, ‘sit’, ‘sleep’, ‘shower’ 등과 같은 HOW 정보가 WHERE 정보에 종속되어 추출된다. 이러한 방식은 시스템의 실제환경에 적용에 상당히 제한되어있다. 그리고 본 시스템에서는 카메라를 사용한 감지로는 각각의 사용자를 식별할 수 없기 때문에 키보드를 추가로 사용한다. 그러나 키보드를 사용하는 것은 사용자로부터 직접적인 입력을 요구하는 것이기 때문에 얼굴인식, 홍채인식, color code[9]등을 사용하여 이에 대한 문제를 해결 할 수 있다.

사용자 감지 모듈은 USB 카메라를 이용하여 움직임을 감지하는 방식으로 동작한다. 먼저 카메라를 이용하여 기본이 되는 base 영상을 촬영하고 초당 15 프레임으로 영상을 촬영하여 base 영상과 현재의 영상을 비교한다. 이때 차영상(difference image)을 구하여 사용자를 감지한다. 움직임이 감지되면 에이전트를 통하여 Context-aware 모듈에게 사용자의 정보를 전달한다.

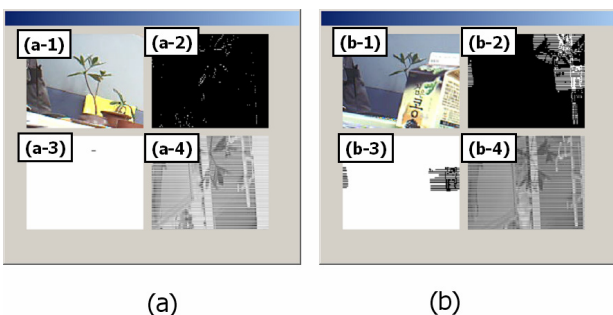


그림 5. 영상처리를 통한 움직임 감지 예.

- (a) 움직임이 없을 때의 영상.
- (b) 움직임이 감지되었을 때의 영상.

그림 5 의 (a-1) 은 현재 촬영되고 있는 영상이며 (a-4)는 기본 영상(base Image)을 2 진 화상처리 기법을 적용하여 흑백영상으로 변환한 기본 영상이다. (a-2)는 흑백으로 된 기본영상과 현재의 영상을 비교하여 구한 차영상이며 (a-3)은 이를 잡음 및 빛에 대한 민감도를 제거한 후의 차영상이다. 그림 5-a 는 사용자가 없는 상태의 영상들이며 그림 5-b 는 사용자가 감지되었을 때의 영상들을 보여 주고 있다.

## 4. 스트리밍 클라이언트와 서버

### 4.1 스트리밍 클라이언트

에이전트는 스트리밍 클라이언트의 시작과 종료를 관리하며 Context-aware 모듈로부터 넘겨받은 음악 장르를 서버로 전송하여 스트리밍 서비스를 요청하는 작업을 수행한다. 스트리밍 클라이언트는 음악을 재생하는 재생기로써 NULL SOFT사에서 제공하는 WINAMP Player v3 을 사용한다. WINAMP Player 는 HTTP 통신을 사용하여 웹 서버에 있는 음악을 재생하는 기능을 가지고있다. WINAMP Player 의 실행인자로 음악파일이 존재하는 특정 URL 을 넘겨주면 WINAMP Player 는 해당 URL 에서 HTTP 통신을 사용하여 음악파일을 읽어 음악을 재생한다.

### 4.2 스트리밍 서버

WINAMP Player 가 HTTP 통신을 사용하여 음악을 재생하기 때문에 스트리밍 서버는 웹 서버와 같은 역할을 가진다. 스트리밍 서버는 음악을 선곡하는 모듈과 스트리밍 모듈로 구성되어 있으며 각각의 모듈은 여러 개의 프로세스들로 동작된다. 에이전트에서 클라이언트의 접속 요청을 스트리밍 서버로 전달하면 선곡 모듈은 클라이언트의 접속 요청을 처리하고 개인화된 스트리밍을 위해 사용자별로 음악을 선곡하여 리스트를 만든다. 그리고 스트리밍 모듈을 동작시켜 클라이언트로 음악 파일을 전송한다. 스트리밍 모듈은 음악 리스트에서 읽은 음악 파일을 클라이언트로 전송하는 기능을

수행한다.

선곡 모듈은 5 개의 장르로 미리 분류(발라드, 댄스, 재즈, 클래식, 락)된 음악 폴더로부터 현재 재생하고자 하는 음악의 리스트를 동적으로 생성하여 파일에 기록한다. 각각의 장르는 0~4 까지의 정수로 나타내어지며 사용자가 요청할 수 있는 음악장르는 최대 4 개이다. Context-aware 모듈은 0 부터 4 까지의 조합된 네 개의 숫자를 사용하여 사용자에게 제공할 음악 장르를 표현하고 에이전트는 이를 스트리밍 서버로 전송한다. 스트리밍 서버의 선곡 모듈은 조합된 네 개의 숫자를 가지고 분류되어진 음악 폴더로부터 파일을 가져와 음악 리스트를 생성한다. 예를 들어 발라드와 댄스를 반반씩 섞어 듣고 싶을 때에는 ‘0011’ 이라는 문자가 들어간다. 만약 ‘0114’라는 문자가 들어간다면 전체 음악 리스트 중 발라드는 25%, 댄스는 50%, 락은 25%가 포함되어 구성된다. 그림 6 은 Context-aware 모듈이 에이전트로 전달하는 서비스 요청의 형식이다.

Genre of the music	Genre of the music	Genre of the music	Genre of the music
0000 - only ballad	1113 - provides 75% dance and 25% jazz	1243 - user will like in order of dance, classical, rock, jazz	6666 - stop the music

그림 6. 서비스 요청의 형식과 예.

한편 Collaborative filtering[3,4] 방법은 사용자 선호에 맞춘 추천 시스템에서 널리 사용되는 방법이다. 이는 사용자가 자발적으로 제공한 정보를 사용하여 사용자를 비슷한 선호도를 가진 집단으로 나누어 그 집단 내에서 서로에게 추천하는 방식으로 사용자의 선호에 맞는 서비스를 제공할 수 있다. 이 방법을 본 논문의 시스템에 적용한다면 새로운 컨텍스트에서 Context-aware 모듈의 부족한 추론 능력을 보완할 수 있을 것이다.

스트리밍 모듈은 5 개의 주 프로세스들과 3 개의 보조 프로세스들로 이루어진다. 주 프로세스들은 Main, Play, Next, Prev, Stop 프로세스이고, 보조 프로세스는 Broadcast, List 프로세스이다. 각 프로세스들은 Ring 으로 연결되어 Play, Next, Prev, Stop 과

같은 동작을 하고 Broadcast 를 통하여 음악 서비스를 제공한다. 스트리밍 모듈의 프로세스들은 파이프를 통하여 메시지전송이 이루어지며 메시지를 받은 프로세스만 동작한다. 그림 7 은 스트리밍 모듈에서 프로세스들간의 메시지 흐름을 보여준다.

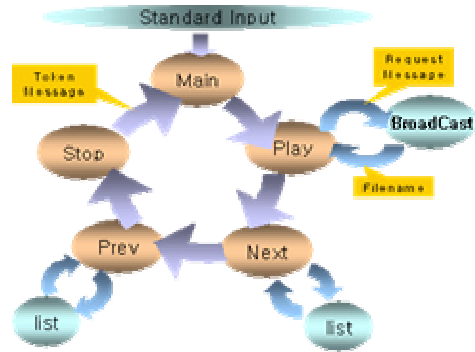


그림 7. 스트리밍 모듈에서 프로세스간 메시지 흐름도.

## 5. 실험

본 논문에서 구현된 시스템은 그림 8 을 바탕으로 실험하였다.

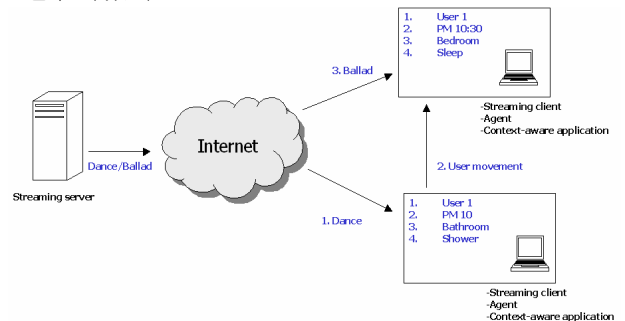


그림 8. 시연 환경.

떨어진 두 장소에서 사용자는 환경에 따라 서로 다른 장르의 음악을 들을 수 있으며 음악을 듣던 중 장소를 옮기더라도 추가적인 설정 없이 지속적으로 음악을 들을 수 있음을 실험하였다. 시스템은 3 번에 걸쳐서 사용자의 상태에 따른 음악 선호도를 학습하고 4 번째부터 학습된 지식에 의해 자동으로 음악을 실행시켰다. 특정 환경에서 발라드를 선호하여 듣던 사용자가 한번 다른 음악을 선택한다면 예외적인 경우로 간주하여 낮은 비율로 학습에 반영되고 서비스가 제공된다. 표 2 는 Context-aware 모듈에게 학습시킨 사용자 정보이고

그림 9 는 이를 바탕으로 실행된 Context-aware 모듈과 시스템의 실행 결과이다.

표 2. 학습시킨 사용자 컨텍스트 정보.

WHO	HOW	WHAT	WHEN	WHERE	Music
User1	Lie		Morning (at 7)	Bed room	Dance
User1	Be		Morning	Dinning room	Classical
User1	Eat	Table	Morning	Kitchen	Rock
User1	Sit	Sofa	Afternoon	Living room	Ballad
User1	Be		Evening	Bath room	Ballad
User1	Open		dawn	Bed room	Classical
User1	Open		Afternoon	Living room	Dance
User1	Be		Rain Evening	Living room	Jazz
User1	Lie		Night	Bed room	Classical

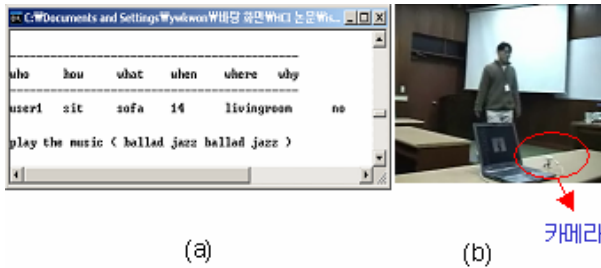


그림 9. Context-aware 모듈과 시스템의 시연.  
 (a) Context-aware 모듈의 실행결과.  
 (b) 시스템 시연.

## 6. 결론 및 추후 과제

본 논문에서는 사용자의 부가적인 입력을 최대한 줄이고 사용자가 듣기를 원하는 음악을 자동으로 들을 수 있게 하는 시스템을 설계하고 구현하였다.

본 논문의 시스템에서는 앞서 지적한 바와 같이 ‘Who’ 컨텍스트를 Keyboard 를 통하여 입력 받고 있다. 이러한 방식은 자동으로 음악서비스를 제공하고자 하는 원래의 목적과는 부합하지 않는 측면이 있다. 이를 해결하기 위하여 이미 언급한 얼굴인식, 홍채인식, color code, 그리고 RF tag 를 사용한 인식기술을 적용할 수 있다. 그리고 본 시스템에 적용된 Context-aware 모듈은 사용자의 만족도와 긴밀한 연관이 있기 때문에 사용자의 사용성 테스트에 기반한 시스템의 보완 작업이 필요하다. 또한 유비쿼터스 컴퓨팅 환경에 존재하는 모든 디바이스나 서비스로의 적용을 가정하고 있기 때문에 더욱 많은 응용에 사용하여 일반성을 증가 시

키는 작업도 필요하다.

## 감사의 글

본 연구는 광주과학기술원 실감방송 연구센터를 통한 정보통신부 대학IT연구센터(ITRC) 사업과 광주과학기술원의 지원에 의한 것입니다.

## 7. 참고문헌

[1] M. C. Mozer, “An intelligent environment must be adaptive”, *IEEE Intelligent Systems and their Applications* 14(2), pp. 11-13, 1999.

[2] M. C. Mozer, R. H. Dodier, M. Anderson, L. Vidmar, R. F. Cruickshank III, and D. Miller, “The neural network house: An overview”, *Current Trends in Connectionism*, pp. 371-380, 1999.

[3] W. Hill, L. Stead, M. Rosenstein, and G. Furnas, “Recommending and evaluating choices in a virtual community of use”, in *Proc. of ACM CHI'95*, pp. 194-201, 1995.

[4] P. Resnick, N. Iacovou, M. Sushak, P. Bergstrom, and J. Riedl. GroupLens: an open architecture for collaborative filtering of Netnews. in *Proc. of Computer Supported Cooperative Work Conference (CSCW)*, pp. 175-186, 1994.

[5] S.Jang, W.Woo, "ubi-UCAM: A Unified Context-Aware Application Model," *Context03*, pp. 178-189, 2003.

[6] P. Werbos, "Backpropagation: Past and future", in *Proc. of the IEEE International Conference on Neural Networks*, pp. 343-353, 1988.

[7] <http://www.mediaunbound.com>.

[8] <http://www.napster.com>.

[9] <http://www.colorzip.co.kr>.