

유비쿼터스 컴퓨팅 환경에서 상황인식 기반 TV 응용 서비스

정희원 문애경*, 이강우*, 김형선*, 김현*, 이수원**

Context-aware based TV Application Services in Ubiquitous Computing Environments

Aekyung Moon*, Kangwoo Lee*, Hyoungsun Kim*,
Hyun Kim, Soowon Lee** *Regular Members*

요 약

유비쿼터스 컴퓨팅 환경이 도래함에 따라 사용자의 명시적 요구에 따라 제공되는 서비스 보다는 상황정보를 활용하여 능동적인 서비스를 지원할 수 있는 기술이 필요하다. 따라서 본 논문에서는 콘텐츠 추천 서비스 에이전트와 상황인식 기반 태스크를 포함하는 CAMUS(Context-Aware Middleware for URC Systems) 시스템을 이용한 상황인식 기반 능동형 서비스를 제안한다. CAMUS는 사용자의 요청이 없더라도 로봇 또는 컴퓨터가 현재의 상황을 인식하여 그 상황에 맞는 정보와 서비스를 제공할 수 있도록 지원하는 소프트웨어 프레임워크이다. 제안된 서비스를 평가하기 위하여 TV 응용 도메인에 적용한다. 이를 위해, TV 프로그램 추천 및 TV 제어 서비스 에이전트, 그리고 TV 도우미 태스크를 구현한다. TV 도우미 태스크는 사용자 위치, 음성 등의 상황 정보에 따라 TV 프로그램 추천 및 제어 서비스를 실행할 수 있도록 한다.

Key Words : Ubiquitous Computing, Context-aware, Sensor, Recommendation System

ABSTRACT

With the advent of ubiquitous computing environments, it has become increasingly important for applications to take full advantage of context information, such as the user's location, to offer greater services to the user without any explicit request. In this paper, we propose context-aware active services on the basis of CAMUS (Context-Aware Middleware for URC Systems). CAMUS is a middleware for providing context-aware applications with development and execution methodology. Accordingly, the applications developed by CAMUS respond in a timely fashion to contexts. To evaluate, we apply proposed active services to TV application domain. Therefore, we implement and experiment the TV contents recommendation service agent, control service agent and TV task based on CAMUS. The context-aware TV task is to recommend programs and control of TV according to user preference, location and voice commands.

I. 서론

다채널을 통한 TV 방송 프로그램의 폭발적 증가

는 시청자에게 다양한 방송 프로그램의 제공과 맞춤형 방송 서비스의 기회가 제공될 수 있기 때문에 시청자 개인 중심의 방송 서비스 시대가 열릴 것

※ 본 논문은 정통부 및 정보통신연구진흥원의 정보통신선도기반기술개발사업의 연구결과로 수행되었습니다.

* 한국전자통신연구원 ({akmoon, kwlee, kimhs, hyunkim}@etri.re.kr) ** 숭실대학교 컴퓨터학부 (swlee@computing.ssu.ac.kr)

논문번호 : KICS2006-03-101, 접수일자 : 2006년 3월 2일, 최종논문접수일자 : 2006년 6월 20일

로 기대한다¹¹. 그러나 TV 채널 수 증가에 비례하여 시청할 수 있는 프로그램이 다양해졌으나, 사용자는 수많은 프로그램을 탐색하고 선별해야 하는 부담을 갖는다¹². 이러한 문제를 해결하기 위하여 사용자 선호도 정보와 방송 프로그램 정보를 이용하여 사용자 취향에 보다 적합한 프로그램을 추천하는 지능형 TV 추천 서비스에 관한 연구가 진행되어왔다^{13, 38}. 기존 연구에서는 사용자의 명시적인 요구에 따라 사용자 프로파일 정보를 활용하여 방송 프로그램을 추천하는 데 중점을 두고 개발되었다. 하지만, 유비쿼터스 환경이 도래함에 따라서 사용자의 요구에 따라 제공되는 서비스 보다는 상황을 인지하여 능동적인 서비스를 지원할 수 있는 기술이 필요하다⁹.

이러한 관점에서 본 논문은 CAMUS(Context-Aware Middleware for URC Systems) 상황인식 미들웨어¹⁰를 이용하여 개인의 위치 상황 정보에 따라 능동적으로 제공될 수 있는 상황인식 TV 프로그램 추천 및 제어 서비스를 제안한다. CAMUS 상황인식 미들웨어는 사용자가 로봇 또는 컴퓨터에게 정보나 서비스를 요청할 때 혹은 사용자의 요청이 없더라도 로봇 또는 컴퓨터가 현재의 상황을 인식하여 그 상황에 맞는 정보와 서비스를 제공할 수 있도록 지원하는 소프트웨어 프레임워크이다¹⁰. 이러한 목적을 위해 CAMUS는 다음과 같은 기능을 제공한다. 첫째, 환경 내 센서로부터의 정보를 획득하고 가공할 수 있는 기능을 제공하며, 환경 내 장치를 동적으로 탐색하고 탐색된 장치를 제어할 수 있는 수단을 제공한다. 둘째, 사용자에 대한 성향 및 선호도에 대한 정보를 학습할 수 있는 소프트웨어 엔진을 제공한다. 셋째, 상황정보 관리 및 액세스를 지원한다. 마지막으로, 상황정보 기반의 응용 프로그램을 개발할 수 있도록 지원하며, 개발된 상황 기반 응용이 구동될 수 있는 소프트웨어 엔진 기능을 제공한다.

CAMUS를 이용하여 상황정보 기반 응용을 개발하기 위해서는 먼저 필요한 서비스 에이전트와 태스크를 정의해야 한다. 서비스 에이전트는 유비쿼터스 환경 내에 존재하는 센서, 장치 그리고 응용프로그램들을 제어하는 소프트웨어 모듈이고 태스크는 사용자의 요청이나 상황정보에 따라 수행되는 일련의 작업을 의미한다. 즉, 태스크는 상황정보에 따라 특정 서비스 에이전트를 구동하게 되는 것이다. 이러한 관점에서 본 논문에서는 상황정보 기반 TV 프로그램 추천 및 제어 서비스를 제공하기 위하여 TV 프로그램 추천 서비스 에이전트와 TV 제어 서

비스 에이전트 및 태스크를 구현한다. 구현된 TV 도우미 태스크는 사용자 위치 정보와 음성 이벤트를 받아서 TV 프로그램 추천 서비스 에이전트와 TV 제어 서비스 에이전트를 구동함으로써 능동적인 서비스를 제공한다. TV 추천 서비스 에이전트는 내용 기반의 추천 기법을 바탕으로 수백 개의 채널에서 방송되는 프로그램 및 사용자의 선호도 정보를 분석하여 사용자가 원하는 프로그램을 추천한다. 그리고 공통 모델과 TV관련 세부 모델로 나누어 사용자 모델을 정의함으로써 TV 추천이외의 다른 응용 분야에 적용이 용이하다. 뿐만 아니라, TV 도우미 태스크는 사용자 위치 기반으로 설계함으로써 다양한 환경에 적용이 가능하도록 설계한다.

본 논문의 구성은 다음과 같다. 2절의 관련 연구에서는 기존의 상황인식 기반 응용 서비스와 TV 추천 시스템에 대하여 언급하고 3절에서는 CAMUS 상황인식 미들웨어에 대하여 설명한다. 4절에서는 본 논문에서 제안하는 상황인식 TV 프로그램 추천 및 제어 서비스를 설명하고 5절에서는 구체적인 구현에 대하여 설명한다. 마지막으로 6절에서 결론을 맺는다.

II. 관련 연구

상황정보를 이용하여 TV 추천 서비스를 제공하는 상황인식 서비스를 제안한다. 따라서 본 절에서는 기존에 개발된 상황인식 기반 응용과 TV 추천 서비스에 대하여 설명한다.

2.1 상황인식 기반 응용 서비스

상황인식 미들웨어에서는 상황정보와 상황정보를 센싱하는 기술이 가장 중요하다. 상황정보에 대한 정의는 여러 가지 있지만 Schilit는 상황정보를 컴퓨팅관련 상황정보(Computing Context), 사용자관련 상황정보(User Context), 그리고 물리적인 상황정보(Physical Context)의 3가지로 나누었다¹⁴. 컴퓨팅관련 상황정보는 네트워크 연결 상태, 통신 대역폭, 그리고 프린터, 디스플레이, 워크스테이션과 같은 주변의 컴퓨팅 자원들을 의미한다. 사용자관련 상황정보는 사용자의 프로파일, 위치, 주변의 사람들 등을 갖는다. 마지막으로 물리적인 상황정보에는 조명, 소음 레벨, 교통 상태, 온도 등이 있다. Chen은 시간을 많은 응용에서 중요하게 여기는 상황정보라고 언급하고 시간, 주(week), 달(month), 계절 등을 갖는 시간관련 상황정보를 제안했다⁹. 컴퓨팅, 사용

자, 물리적 상황 등을 시간 축에 따라 기록한 상황 정보 이력은 현재 상황분석을 더 용이하게 한다. 예를 들면, 현재의 장소와 시간을 알고 사용자의 일정이 적힌 목록을 갖고 있으면, 응용은 사용자가 회의 중인지, 수업을 듣고 있는지, 공항에서 대기 중인지 등과 같은 사용자 현재 상태를 더욱 잘 파악할 수 있을 것이다. 하지만, 현재까지 개발된 대부분의 상황인식 응용은 센싱이 가장 용이한 사용자 관련 상황정보를 주로 이용한다.

사용자관련 상황정보를 이용한 대표적인 상황인식 기반응용은 다음과 같다. Call Forwarding^[12]는 Olivetti 연구소에서 개발한 것으로 사용자 위치를 상황정보로 사용한다. 이것은 사용자의 위치를 인식해서 전화가 왔을 때 사용자로부터 가장 가까운 곳에 있는 전화기로 연결해 주는 것으로 액티브 배지(Active Badge)를 이용하여 사용자 위치를 파악한다. 액티브 배지는 적외선으로 개인의 위치를 검출하는 시스템이다. 적외선을 발생하는 배지를 옷에 부착하여, 환경 내 설치된 적외선 센서를 통해 어느 방에 누가 있는 지등의 정보를 제공한다. Cyberguide^[13]는 Georgia 기술 연구소에서 개발한 것으로 관광객의 위치와 시간을 상황정보로 이용한다. 다양한 종류의 시스템이 사용자에게 현재 위치에서 이용 가능한 정보 서비스를 제공한다. 외부의 위치정보는 GPS를 통해 획득하고 내부에서는 IR positioning 시스템을 사용한다. Shopping Assistant^[14]는 AT&T Bell 연구소에서 개발하였다. 이 장치는 사용자에게 물품의 위치와 상세정보를 알려주고 세일 중인 물품 목록을 제공하며 가격을 비교 분석하는 등 매장에서의 쇼핑을 안내한다. Georgia 기술 연구소에서 개발한 Conference Assistant^[15]는 회의 참석자를 돕기 위해 다양한 상황정보를 이용한다. 회의 일정, 사용자의 위치, 사용자의 관심 분야 등을 조사하고 사용자가 발표장에 들어가면 자동으로 발표자의 이름과 발표 주제 및 다른 관련 정보를 보여준다. People and Object Pager^[16]는 Kent대학에서 개발한 것으로 상황정보로 사용자의 위치, 근처의 사람과 사물을 이용한다. Pager는 방문객으로부터 가장 가까이에 위치한 사람에게 메시지를 라우팅함으로써 페이지링 장치는 갖고 있지 않지만 액티브 배지를 가진 방문객에게 메시지를 보낼 수 있다. 마지막으로, [17]는 smart dust^[18]라는 초소형 센서를 이용하여 사용자 위치를 추적하기 위한 연구이다. 크기가 눈에 보이지 않을 정도로 작은 smart dust는 센싱 연산 기능과 무선통신장비가 결합된 센서로 Römer는 이것을

이용하여 사물의 위치 추적 시스템을 구현하였다^[17].

2.2 TV 추천 서비스

현재 많이 사용되고 있는 대표적인 추천 기법에는 내용기반 추천(Content-based Recommendation)과 협업 추천(Collaborative Recommendation)이 있다. 내용기반 추천은 정보 검색(Information Retrieval) 분야에 바탕을 두고 있으며 사용자의 프로파일과 추천의 대상이 되는 아이템의 구성 요소를 비교하여 유사도가 높은 것들을 추천하는 기법이다. 협업 추천은 추천의 대상이 되는 목표 사용자와 비슷한 프로파일을 가진 다른 사용자를 찾아 그 사용자가 높은 평가를 매긴 아이템을 목표 사용자에게 추천한다^[9]. 이 외에 사용자의 성별, 나이, 직업 등의 인구통계학적 정보를 활용하여 추천을 하는 인구통계학적 추천 기법과 내용 기반 추천과 협업 추천을 결합한 하이브리드 추천 기법 등이 있다.

협업 기반 TV 프로그램 추천의 경우 다른 사람이 추천한 프로그램을 내가 시청해야 하는데 어떤 프로그램을 이미 시청한 후에 추천을 해 준다면 추천의 효과를 상실하게 된다. 또한 많은 시청자들의 시청기록과 프로그램을 관리해 줄 중앙 서버가 필요하기 때문에 대부분의 TV 추천 서비스는 내용기반 방식을 사용한다. 내용기반 추천 기법을 이용한 대표적인 연구로는 [1], [5] 그리고 [8]이 있다. [8]에서는 키워드 기반의 내용기반 추천을 제안하여 장르, 세부장르, 채널, 인물, 키워드 등을 추천에 반영한다. 그러나 사용자의 생활방식에 따라 시간대 및 요일별로 장르에 대한 선호도가 차이는 반영하지 않고 있다. [1]은 시간대별 장르, 채널 선호도와 키워드 선호도를 활용한 내용기반 추천을 제안하고 있다. 그러나 여기서의 키워드 선호도는 선호도 편집이나 프로그램 검색 시에 사용자가 직접 입력한 키워드들만을 대상으로 하고, 사용자 모델 갱신에서 암시적으로 발생하는 키워드에 대한 선호도가 포함되지 않는다. [5]에서는 에이전트를 기반으로 베이지안(Baysian) 알고리즘을 활용한 내용기반의 추천 기법을 사용하며 사용자의 명시적, 묵시적 피드백을 추천에 반영한다.

성별, 나이, 직업 등의 인구통계학적 정보를 활용하여 추천을 하는 인구통계학적 추천 기법을 사용한 대표적인 연구에는 [6]과 [20]이 있다. 이들 연구에서는 성별 나이 등의 Stereotype에 의한 추천 기법을 사용한 TV 추천을 제시하였지만, 사용자 한 사람을 위한 것이 아닌 인구 통계학적으로 유사한

집단에 대한 모델이기 때문에 사용자의 선호 정보를 활용한 개인화된 정보를 제공하기에는 부족하다. 현재 상용화되고 있는 TV 추천 서비스로는 Tivo^[21]가 있다. Tivo는 PVR(Personal Video Recorder)이라는 저장 기능을 가진 TV를 활용하여 양방향 서비스를 제공하고 있으며 사용자가 선호하는 키워드를 입력하면 해당 키워드가 나타나는 프로그램을 자동으로 녹화해주는 기능을 제공한다. 하지만, 사용자의 명시적인 선호도 정보와 프로그램 정보의 단순 비교를 통한 추천 결과를 제시하는 단점을 갖는다.

III. 상황인식 미들웨어

본 논문에서 제안한 상황 인식 TV 응용 서비스를 이해하기 위하여 먼저 유비쿼터스 컴퓨팅 환경 하에서 각 가정에서 발생할 수 있는 시나리오를 이해하여야 한다. 특히 기존의 공중파 중심으로 이루어진 TV에서는 시청자가 원하는 프로그램을 시청하기 위해 특별히 고려해야 할 사항이 없었지만, 디지털 방송 시대가 열리고 각 가정마다 디지털 방송이 일반화되면 채널의 수가 수백 개로 늘어난 상황에서 사용자는 원하는 채널을 선택하는 데 많은 시간적 오버헤드를 갖는다. 또한 1가구 1대 이상의 TV를 보유하는 가구 수가 증가함에 따라 그에 맞는 서비스를 제공할 필요가 있다. 이를 위해 아래 [예 1]의 시나리오를 살펴보자.

[예 1] 가정에는 거실과 안방에 TV가 존재하며 음성 명령을 통해 TV의 제어가 가능하다고 가정하자. 또한 사용자는 TV를 시청하던 상태에서 TV가 있는 곳으로 이동하면 동일한 서비스를 계속 제공받기를 원한다고 가정한다. 구체적인 시나리오는 다음과 같다.

시나리오:

1. 사용자가 거실에 진입하여 소파에 앉는다.
2. 자동적으로 TV가 켜지고 선호도가 가장 높은 채널로 이동한다.
3. 사용자의 명령을 기다린다.
4. 음성 명령으로 채널 및 볼륨을 조절한다.
5. 사용자가 안방으로 이동한다.
6. 거실의 TV가 자동으로 꺼진다.
7. TV가 켜진 상태에서 이동하였으면 안방에 TV를 켜준다.

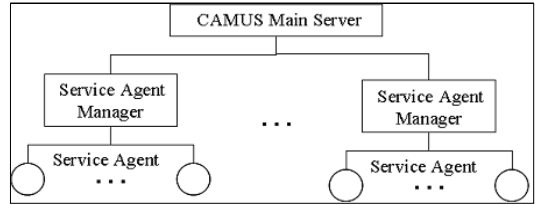


그림 1. CAMUS 시스템 구성

8. 공부방으로 이동한다.
9. 공부방에는 TV가 없으므로 동작하지 않는다.

사용자의 위치 정보나 음성 명령 등과 같은 상황 정보를 이용한 서비스를 제공하기 위해서는 상황인식 미들웨어가 필요하다. 상황인식 미들웨어는 사용자가 정보나 서비스를 요청할 때 현재의 상황을 이해하여 그 상황에 맞는 서비스를 제공하거나, 사용자의 요청이 없는 경우에도 필요한 시점에 필요한 정보와 서비스를 능동적으로 제공할 수 있어야 한다. CAMUS는 그림 1과 같이 CAMUS 메인서버(Main Server)와 서비스 에이전트 관리자(Service Agent Manager)로 구성되는 상황인식 미들웨어이다.

3.1 서비스 에이전트 관리자

서비스 에이전트는 CAMUS와 통신을 통해 유비쿼터스 환경 내 존재하는 센서나 장치 및 legacy 응용의 기능을 수행할 수 있도록 만들어진 소프트웨어 모듈로써 서비스 에이전트 관리자에 의해 수행된다. 즉, 서비스 에이전트는 CAMUS와 연동되는 외부 장치 및 시스템에 대한 인터페이스(Interface)를 구현(Implementation)한 소프트웨어 모듈이라고 할 수 있다. 따라서 서비스 에이전트는 서비스 에이전트 인터페이스를 통해 해당 서비스 에이전트가 제공하는 속성 및 동작을 외부 시스템에 노출한다. 예를 들면, 사용자 위치를 인식하기 위한 서비스 에이전트는 사용자 id와 location 정보를 넘겨주는 인터페이스를 갖고 환경에 RFID 센서가 있다고 가정하자. 그러면, RFID 센서에서 제공되는 디바이스 드라이버를 이용해 해당 인터페이스를 프로그램으로 구현해야 한다. 이와 같이 구현된 프로그램이 결국 사용자의 위치 정보를 제공하는 서비스 에이전트가 된다.

서비스 에이전트 관리자는 구현한 서비스 에이전트들을 관리 및 제어하는 프로그램으로 여러 다양한 유비쿼터스 환경에 설치되어 환경 내 위치한 센서로부터 정보를 얻고 그 정보를 CAMUS 메인서버에 전달하는 한편 CAMUS 메인서버로부터의 실

행명령을 받아 환경 내 장치를 제어하는 역할을 한다. 이러한 서비스 에이전트 관리자는 임의의 공간 내에 설치될 수 있으며, 로봇 단말(Robot Platform)이나 개인 휴대 단말(Sobot Platform) 등에도 설치될 수 있다.

3.2 CAMUS 메인서버

CAMUS 메인서버는 여러 서비스 에이전트 관리자로부터 전달되는 상황정보를 수집하여 상황정보를 운용하고, 액세스를 지원하며, 상황인식 응용 개발에 필요한 제반 기능을 지원하는 프레임워크이다. CAMUS 메인서버에서는 환경관련 상황정보와 사용자 관련 상황정보를 관리하고 있으며, 상황정보의 변화에 따른 이벤트를 응용시스템으로 전달하여 상황정보에 적합한 행위를 취할 수 있도록 한다. 또한 음성인식, 영상인식, 행위제어 등 로봇 또는 소프트웨어 로봇의 기본적인 서비스와 연동할 수 있는 서비스 컨테이너를 제공한다. 그림 2는 CAMUS의 메인 서버 구조를 나타낸다.

환경 내 설치된 서비스 에이전트 관리자로부터의 전달된 센서 이벤트들은 CAMUS 메인서버의 이벤트 시스템(Event System)을 통해 전달된다. 이벤트 시스템은 센서나 기타외부 서비스를 통해 상황 관리자에 의해 전달되는 이벤트를 이를 원하는 작업에게 전달하는 역할을 한다. 전달된 상황정보는 상황정보 해석기(Context Interpreter)를 통해 보다 상위의 상황정보로 해석되어 CAMUS 메인서버 내에서 관리된다. 상황처리 엔진(Context Engine)은 이벤트 시스템으로 전달된 상황정보로부터 상위 수준의 상황정보를 얻고 이를 별도의 저장소에 관리하고, 태스크 엔진(Task Engine)에서 작업을 수행하는데 필요한 상황정보를 제공하는 역할을 한다. 이러한 상황정보는 각각 서비스 에이전트 디렉토리 서비스(Service Agent Directory Service), 태스크 관

(User Preference Engine) 등을 통해 관리된다. 태스크 관리자는 개별 작업을 기동시키고, 수행중인 작업 프로세스를 관리 또는 제어하는 역할을 한다. 태스크 엔진은 상황에 따른 실제 작업을 수행시킨다. 작업 엔진은 해당 작업에 대한 상황 정보(Fact)와 작업 규칙(Rule)에 따라 특정 서비스 에이전트를 호출한다.

작업 규칙은 PLUE(Programming Language for Ubiquitous Environments)라고 하는 자바를 확장한 프로그래밍 언어를 통해 기술된다. PLUE는 사용자가 보다 쉽게 작업 규칙을 기술하기 위해 만들어진 언어이다. 다음은 PLUE로 기술된 간단한 작업 규칙의 예를 보여준다.

```
on ( $place.temperature::ValueChanged e )
if ( e.value > $place.resident.preferred_temp.high )
{
    $place.air_conditioner.turnOn();
}
```

“on” part에는 작업 규칙의 수행을 유발하는 이벤트 연산식(Event Description)이 기술되고, “if”는 해당 작업 규칙이 수행되기 위한 조건을 기술하고, if 내부는 “action”에 해당하는 부분으로써, 명시된 조건이 만족하는 경우 수행될 서비스를 기술한다. 이 예에서는 온도값이 변화되고, 사용자가 원하는 온도보다 높으면, 에어컨을 작동시키는 간단한 규칙을 보여준다. 변수에 대한 값들은 상황처리 엔진으로부터 얻어진다.

상황에 맞는 작업 수행과정을 정리하면 다음과 같다. 상황정보의 변경은 시스템 내에서 이벤트를 발생시키며, 이 이벤트는 모든 CAMUS 태스크로 전달된다. CAMUS 태스크에서는 해당 이벤트가 관련된 이벤트일 때 관련 작업규칙을 체크한다. 작업 규칙 추론 과정에서 현재의 상황정보를 검색할 수 있으며 상황정보가 규칙에 부합될 때 실제 행위가 발생된다. 이 행위는 서비스 에이전트 관리자에게 환경 내 서비스 에이전트의 동작을 요청하는 형태가 된다.

사용자와 CAMUS 간의 상호작용은 그림 2의 소프트웨어 로봇(Software Robot)인 소봇(Sobot)에 의해 이루어진다. 소봇은 사용자의 명령을 받아 시스템에 전달하거나 시스템으로부터의 메시지를 사용자에게 전달하는 역할을 한다. 소봇은 다양한 정보단말기 상에서 구동하며, 아바타 형태의 소프트웨어 캐릭터 및 음성 인터페이스를 갖는다.

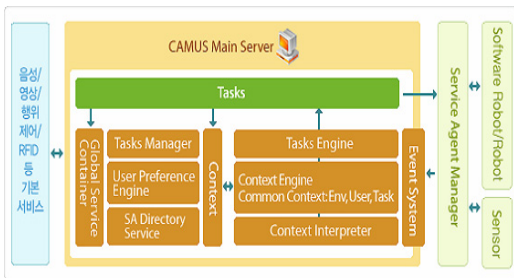


그림 2. CAMUS 메인서버의 구조
리자(Task Manager) 및 사용자 선호정보 처리 엔진

IV. 상황인식 기반 능동형 서비스

CAMUS를 이용하여 [예 1]의 상황인식 TV 응용 서비스를 제공하기 위해서는 TV 추천 서비스 에이전트, TV 제어 서비스 에이전트 그리고 TV 도우미 태스크가 필요하다. TV 추천 서비스 에이전트는 TV 프로그램 콘텐츠와 사용자 선호 정보를 바탕으로 사용자가 선호하는 TV 프로그램을 추천하는 역할을 한다. TV 제어 서비스 에이전트는 TV 볼륨을 조절하거나 채널을 변경하는 등의 TV 제어와 관련된 역할을 하는 것으로 실제 물리공간 상의 TV와 연동한다. TV 도우미 태스크는 CAMUS에서 제공되는 상황 정보를 바탕으로 작업을 수행하기 위해 구현되는 모듈로써 다양한 환경에 적용이 가능해야 한다. 즉, TV의 위치가 변경되더라도 서비스 태스크를 변경하지 않고 서비스 수행이 가능하도록 설계해야 한다.

4.1 TV 도우미 태스크

상황인식 서비스를 제공하기 위해서는 상황에 맞게 해당 서비스 에이전트의 동작을 호출하는 CAMUS 태스크가 필요하다. 앞서 설명했듯이 CAMUS 태스크는 이벤트와 조건을 통해 필요한 상황정보를 기술하고 해당 상황에서 수행할 서비스 에이전트를 action부분에 기술한 다수의 작업 규칙으로 구성된다.

[예 1]의 시나리오를 바탕으로 설계한 CAMUS 태스크의 상태 전이도는 그림 3과 같다. 상태 전이도는 특정 이벤트가 입력되었을 때 상태의 변화를 나타낸 것으로, PLUE를 통해 구현하기 위한 설계 부분에 해당한다. 본 구현에서는 UserEntered와 Speech-

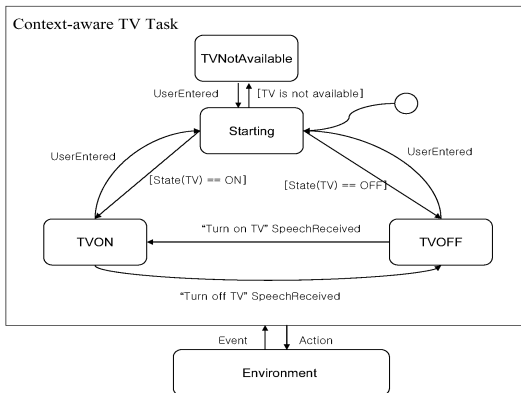


그림 3. TV 도우미 태스크의 상태 전이도
Received의 CAMUS 시스템 이벤트를 사용한다.

UserEntered 사용자가 특정 위치에 진입한 것을 알리는 이벤트이고 SpeechReceived는 사용자의 음성 명령을 알리는 이벤트이다. 그리고 Starting, TVNotAvailable, TVON, 그리고 TVOFF의 4가지 상태로 구성된다. 본 태스크는 사용자 위치가 변경되면 Starting 상태로 전이하여 해당 위치에서 TV의 유무 및 TV 상태를 검사하게 함으로써 환경이 변화하더라도 적용이 가능하다.

4.1.1 Starting 상태

TV도우미 태스크를 시작한 경우 진입하는 초기 상태이다. 현재 사용자 위치를 알아내고 해당 위치의 TV의 상태를 확인하여 TVON, TVOFF TVNotAvailable 상태 중 하나로 분기한다.

4.1.2 TVNotAvailable 상태

현재 위치에서 TV가 가용하지 않은 상태로 사용자의 위치가 변경되면 Starting 상태로 전이한다.

4.1.3 TVON 상태

현재 위치에서 TV 켜져 있는 상태이다. 상태 진입시 TV 추천 서비스 에이전트의 동작을 실행함으로써 선호도 정보를 구하여 추천 채널로 이동한다. 음성 이벤트에 따라 TV를 제어한다. 사용자의 위치가 변경되면 TV를 끄고 Starting 상태로 전이한다. 사용자가 “TV꺼”라는 명령에 TV를 끄고 TVOFF 상태로 전이한다.

4.1.4 TVOFF 상태

현재 위치에서 TV 꺼져 있는 상태로 사용자의 위치가 변경되면 Starting 상태로 전이한다.

4.2 TV 프로그램 추천 서비스 에이전트

그림 4는 TV 프로그램 추천 서비스 에이전트의 시스템 구조를 나타낸다. 제안된 서비스 에이전트는 TV 추천을 위한 방송 정보 관리자, 프로그램 추천 관리자 그리고 사용자 모델 관리자로 구성되고 CAMUS 메인서버의 태스크 관리자를 통해서 액세스된다.

방송 정보 관리자는 추천에 사용하기 위한 TV 프로그램 정보를 방송국 서버로부터 획득하는 작업과 획득하는 일을 담당한다. 추천 관리자는 방송국 관리자에서 획득한 프로그램 정보에 대해 사용자 모델을 바탕으로 각각의 선호도를 계산하여 선호도가 높은 순으로 추천 리스트를 제공한다. 사용자 모델 관리자는 사용자 선호도 정보를 작성하고 시청

히스토리를 바탕으로 사용자 모델의 선호도를 갱신하는 작업을 담당한다.

본 절에서는 TV 프로그램 추천 서비스 에이전트를 설명하고 프로그램 추천 관리자와 사용자 모델 관리자에서 사용하는 추천 알고리즘 및 사용자 모델 갱신 알고리즘을 설명한다.

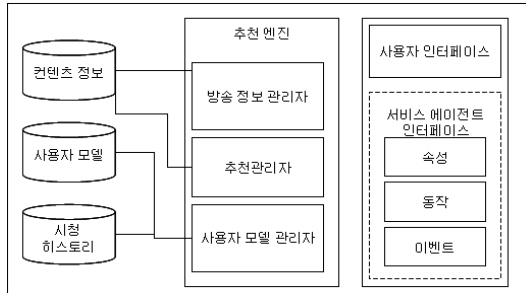


그림 4. TV 추천 서비스 에이전트의 구조

4.2.1 구조

CAMUS를 통해 상황인식 서비스 에이전트를 구현하기 위해서는 그림 4의 속성(Property), 동작(Operation), 이벤트의 3가지 구성요소를 정의해야 한다.

첫째, 속성은 속성명과 그 속성값의 쌍으로 표현된다. CAMUS 시스템에서는 서비스 에이전트의 속성값을 읽어서 서비스 에이전트에 해당되는 센서 또는 장치의 상태를 확인하거나, 그 값을 변경함으로써 물리적 장치 뿐만 아니라, 서비스 에이전트의 서비스를 구동시킬 수 있다. TV 추천 서비스 에이전트는 추천할 프로그램 수 및 추천 대상이 되는 프로그램 시간 정보 등의 속성을 사전에 정의한다. 다음 표 1은 TV 추천 서비스 에이전트의 속성과 의미를 나타낸다.

recommend_no의 속성 값을 “5”로 설정하면 CAMUS의 사용자 인터페이스인 소봇을 통해 PDA

표 1. TV 프로그램 추천 서비스 에이전트의 속성

속성	의미
recommend_no	추천할 프로그램 수
interval	수집될 방송 편성 정보의 기간
tv_start_limit	추천 대상으로 삼을 프로그램의 시작 시간 허용치
tv_end_limit	추천 대상으로 삼을 프로그램의 끝나는 시간 허용치
channel_code	추천 지역코드

혹은 컴퓨터 단말기 화면으로 다섯 개 프로그램을 추천한 결과를 보여준다. interval의 값을 “2”로 변경하면 모레까지의 방송 편성 정보를 방송국 서버로부터 수집한다. tv_start_limit의 속성값을 “10”으로 하면 시작한 지 10분 이상 지난 프로그램들이 추천대상이 된다. tv_end_limit의 속성값을 “10”으로 하면 끝날 시간이 10분 이상 남은 프로그램들이 추천대상에 포함된다. 지역 별로 방송국 채널 번호가 다르기 때문에 channel_code의 속성 값에 추천 지역을 등록한다.

둘째, 서비스 에이전트의 동작을 정의한다. 서비스 에이전트의 동작은 서비스 에이전트에 해당되는 연산 처리 또는 물리적, 논리적 장치의 제어를 위한 함수 호출에 해당된다. 이러한 동작 호출에 의한 결과는 환경에 영향을 미친다. 다음은 추천 서비스 에이전트의 동작을 나타낸다. 상세한 알고리즘은 4.2.2 절에 나타난다.

(1) getRecommendationInfo(String uid)

선호도 정보와 방송 프로그램 정보간의 유사도를 계산하여 사용자가 흥미를 가질만한 방송 프로그램의 목록을 제공한다. 내용 기반 추천 기법을 바탕으로 사용자의 요일별, 시간대 별 시청 장르에 대한 빈도를 분석한 후 사용자의 취향에 맞는 TV 프로그램을 추천한다.

(2) saveHistory(TVProgramInformation info)

프로그램 정보와 사용자의 선호도 정보를 저장하기 위해 방송 정보 저장소와 사용자 모델을 필요로 하며 사용자의 선호도를 동적으로 계산하기 위해 시청 정보를 저장한다. 추천 프로그램에 대한 긍정 피드백과 부정 피드백 정보를 포함한다.

(3) userModelUpdate(Watching History)

사용자의 시청 히스토리 정보를 바탕으로 선호도 정보를 갱신한다. 사용자 모델 갱신은 TV 세부정보 갱신과 공통 모델 갱신 작업으로 구성된다.

셋째, 이벤트를 정의한다. 서비스 에이전트는 특정 상황 발생을 CAMUS 메인서버의 센서 프레임워크로 통보한다. 따라서 서비스 에이전트는 그 특성에 따라 발생하는 다양한 이벤트를 갖는다. 예를 들어 음성 센서의 경우, 사용자의 음성명령이 들어올 때, SpeechRecieved 이벤트를 발생시키며, RFID 센서의 경우, 사용자가 임의의 위치에 들어올 때 UserEntered 이벤트를 발생시키는 경우에 해당된다. 현재 TV 추천 서비스 에이전트는 특정 이벤트를

발생하지는 않지만, 향후에는 아주 높은 선호도 값을 가진 프로그램이 발생하였을 때 이벤트를 발생할 예정이다.

4.2.2 알고리즘

그림 4의 추천 관리자는 선호도 정보와 방송 프로그램 정보간의 유사도를 계산하여 사용자가 흥미를 가질만한 방송 프로그램의 목록을 제공한다. 방송 프로그램 정보와 사용자의 선호도 정보를 저장하기 위해 방송 정보 저장소와 사용자 모델을 필요로 하며 사용자의 선호도를 동적으로 갱신하기 위해 “빈도(Frequency) 테이블(F-TBL)”과 사용자의 시청 정보를 갖는 “히스토리 테이블(H-TBL)”을 갖는다.

F-TBL은 사용자마다 요일별, 시간대별 장르에 따라 서로 다른 시청 패턴을 보이며 TV 프로그램의 경우 평일과 주말의 편성 특징이 크게 다르므로 이를 반영하기 위해 3차원 배열로 구성되며 요일별/시간대별 시청 장르에 대한 빈도 값을 저장되고 TV Specific Model에 추가적으로 저장된다. H-TBL에서 요일별/시간대별로 시청한 요일, 시간, 장르를 분석하여 F-TBL을 구한다. 즉, 사용자의 시청 히스토리에서 요일별/시간대별로 특정 장르를 사용자가 시청한 수만큼 갱신 되고 해당 정보는 다음의 선호도 계산식에 사용된다. 그림 5는 F-TBL의 갱신 과정을 보여준다. F-TBL의 값을 구하기 위하여 3차원 배열로 구성된 “갱신 테이블(U-TBL)”을 갖는다. 먼저 U-TBL을 초기화하고 시청 히스토리의 정보를 읽어 [요일, 시간, 장르]에 해당 되는 U-TBL[요일][시간][장르]의 값을 “1” 증가한다. 시청 히스토리의 정보를 모두 읽었다면 U-TBL의 정보를 F-TBL에 더하여 빈도 매트릭스 값을 구한다.

선호 정보에 따라 TV 프로그램을 추천하기 위한

```

initialize U-TBL
while(has more H-TBL)
  program information obtains next history
from H-TBL
  d is index of day
  t is index of time
  g is index of genre
  increase U-TBL[d][t][g]
end of while
for i∈day
  for j∈time
    for k∈genre
      F-TBL[i][j][k] += U-TBL[i][j][k]
    end of for
  end of for
end of for
    
```

그림 5. 가중치 갱신 알고리즘

사용자별 프로그램에 대한 선호도 계산은 식 1과 같이 TV Specific Model과 공통 모델의 선호도를 계산하여 합한 값으로 이루어진다. TV Specific Model에 의한 선호도는 식 2와 같이 장르, 채널, 세부장르에 의한 선호도를 계산한 후 각각의 가중치를 곱한 값의 합으로 계산된다. 공통 모델에 의한 선호도 계산은 식 3과 같이 방송 프로그램의 제목, 내용, 개요, 줄거리로부터 추출한 키워드 벡터(Vector_c)와 사용자 키워드 벡터(Vector_m)와의 코사인 유사도를 구한다.

$$Preference^{u,p} = \alpha \times Preference_{Specific}^{u,p} + \beta \times Preference_{Common}^{u,p} \quad (1)$$

u : 사용자, p : 방송 프로그램
 Preference^{u,p}: 사용자 모델에 의한 사용자 u의 p에 대한 선호도

Preference^{u,p}_{Specific} : TV Specific Model에 의한 선호도

Preference^{u,p}_{Common} : 공통 모델에 의한 선호도

α : TV Specific Model에 대한 가중치

β : 공통 모델에 대한 가중치

$$Preference_{Specific}^{u,p} = w_G \times Preference_{Genre}^{u,p} + w_{sg} \times Preference_{subGenre}^{u,p} + w_C \times Preference_{Channel}^{u,p} \quad (2)$$

$$Preference_{Keyword}^{u,p} = COSINE_SIM(Vector_c, Vector_m) \quad (3)$$

사용자 모델 관리자는 사용자 모델을 작성하고 시청 히스토리를 바탕으로 사용자 모델의 선호도를 갱신하는 작업을 담당한다. 사용자 모델 갱신은 TV Specific Model 갱신과 공통 모델 갱신 작업으로 구성된다.

TV Specific Model 갱신에서 선호 세부 장르, 선호 채널, 선호 장르에 대한 갱신 방식은 동일하며 사용자가 명시적으로 입력한 내용은 갱신하지 않고 유지한다. 사용자 모델 갱신을 위해 가장 먼저 일정 기간 이상 시청하지 않는 세부 장르나 채널을 삭제하기 위하여 각각의 선호도를 0.1씩 감소시킨다. 다음 그림 6은 세부장르에 대한 갱신 식을 나타낸다. 사용자의 시청 히스토리로부터 긍정 피드백과 부정 피드백을 구분하여 세부 장르에 각각의 가중치를 곱한 후 각각의 최대(max) 값을 이용하여 정규화하여 갱신할 값을 계산한다. 마지막으로 모델에 저장되어 있지 않은 내용은 새로이 추가하고 이미 존

$$V_{sg}^u = \sum_{i \in F(+)} P_{subGenre} + \sum_{i \in F(-)} N_{subGenre}, \quad sg \in subGenre$$

$$V_{sg}^u = \frac{V_{sg}^u}{\max |V_{sg}^u, sg \in subGenre|}$$

$$Preference_{sg}^u = Preference_{sg}^u + \frac{V_{sg}^u}{sg \in subGenre}$$

u : 사용자
 F(+): 사용자가 positive로 피드백을 준 프로그램의 집합
 F(-): 사용자가 negative로 피드백을 준 프로그램의 집합
 subGenre : 프로그램의 세부 장르
 P_{subGenre} : positive 프로그램의 세부 장르에 대한 가중치
 N_{subGenre} : negative 프로그램의 세부 장르에 대한 가중치

그림 6. 세부장르에 대한 선호도 갱신 식

재하는 내용에 대해서는 기존의 선호도에 갱신 값을 더한다. 갱신 후의 선호도가 0일 때에는 모델에서 삭제한다. 채널에 대한 갱신 방법도 동일하다. 공통 모델의 선호 키워드를 갱신하기 위해서는 우선 가장 최근의 관심사를 반영하기 위하여 공통 모델에 저장되어 있는 모든 키워드의 선호도를 0.1씩 감소시킨다. 공통 모델의 갱신 방법도 그림 6과 동일하다.

4.3 TV 제어 서비스 에이전트

TV 프로그램 추천 에이전트와 마찬가지로 TV 제어 서비스 에이전트도 속성, 동작을 정의한다. 제공되는 이벤트는 없으므로 정의하지 않는다.

- (1) 디지털 TV 혹은 셋탑 박스와 통신하기 위한 IP와 Port에 대한 속성 및 속성 값을 정의한다.
- (2) 필요한 동작은 다음 표 2와 같다. TV를 켜고 끄는 작업을 위한 setPower, 채널을 제어하기 위한 동작, 그리고 소리 크기에 관련된 동작들로 구성된다.

표 2. TV 제어 서비스 에이전트의 동작

boolean getPower()	TV의 현재 상태 알림
void setPower(boolean power)	TV를 끄고 켜
void setChannel(int channel)	해당 채널 이동
void channelUp()	채널 올림
void channelDown()	채널 내림
void volumeUp()	소리 올림
void volumeDown()	소리 내림

V. 구현

본 논문에서 제안하는 TV 추천 서비스 에이전트는 다른 응용에도 활용이 용이하도록 사용자 모델

표 3. 사용자 모델 및 초기 가중치 정보

TV Specific 모델	
선호세부장르	선호채널
가정/주방/생활용품(1.0) 모험(1.0) 액션/무협(1.0) 버라이어티(1.0) 의료건강(1.0) 영어(1.0) 멜로(1.0) 축구(1.0) 스포츠(1.0)	KBS2(1.0) KBS1(1.0) SBS(1.0) MBC(1.0)
공통모델	
키워드	정치(1.0)
선호인물	비(1.0) 홍명모(1.0)

no.	프로그램명	방송시간	채널	장르	시청
1	2004 이태년 올림픽	22:00-24:00	MBC	스포츠/종합	<input type="checkbox"/> 취소
2	물하우스	22:05-23:00	KBS2	드라마/멜로	<input checked="" type="checkbox"/> 선택
3	바람난 가족	21:50-23:50	케치온	영화/드라마	<input type="checkbox"/> 취소
4	벤티맨	20:20-22:20	OCN	영화/액션	<input type="checkbox"/> 취소
5	물하우스	22:00-24:00	MBC	스포츠/종합	<input type="checkbox"/> 취소
6	바람난 가족	22:05-23:00	KBS2	드라마/멜로	<input type="checkbox"/> 취소
7	벤티맨	21:50-23:50	케치온	영화/드라마	<input type="checkbox"/> 취소
8	물하우스	20:20-22:20	아리랑	드라마/멜로	<input type="checkbox"/> 취소
9	바람난 가족	21:50-23:50	KMTV	영화/드라마	<input type="checkbox"/> 취소
10	물하우스	21:50-23:50	KBS2	영화/드라마	<input type="checkbox"/> 취소

그림 7. TV 프로그램 알림 및 추천 화면

을 공통 모델과 도메인 응용 모델로 나누어 정의한다. 따라서 도메인 응용에는 TV 관련 정보를 저장하고 TV 추천 결과에 따른 피드백 정보를 공통 모델에 반영함으로써 타 응용에 영향을 주도록 설계되었다. 본 절에서는 사용자의 선호도 정보에 따라 TV 추천 서비스 에이전트의 추천 결과와 해당 추천 결과에 대한 사용자의 피드백에 따른 사용자 모델 갱신 결과를 보여주는 실험 및 결과를 설명한다. 또한 상황정보를 이용하여 능동적으로 동작할 수 있도록 TV 도우미 태스크를 구현한다.

전체적인 프로그램의 구현을 위한 언어는 SUN의 자가 JDK 1.5.0_04를 사용했으며 방송 정보는 www.epg.co.kr에서 획득하였다. 사용자의 입력 정보 따라 생성된 사용자 모델과 초기 가중치 정보는 표 3과 같다.

그림 7은 사용자 모델 등록 후 바로 프로그램 추천을 요구했을 때의 결과를 나타낸다. 사용자가 TV 위치에 있는 경우 TV 프로그램 추천 서비스 에이전트의 실행결과 가장 순위가 높은 방송을 소북을 통해 PDA 화면에 알림해주는 모습과 추천 프로그램 리스트 결과를 나타낸다.

이 때 그림 7에서 사용자가 추천 목록 중에서 원하는 프로그램을 선택하면 그 정보는 H-TBL에 저

```

선호 세부 장르 :
오락게입쇼:0.8999  축구:1.0  대중가요:1.0  애니메이션:0.8999  토크쇼:0.8999
시트콤:0.8999  버라이어티:1.0  멜로:1.0  가정/주방/생활용품:1.0
선호 채널 :
KBS2:1.0  코미디TV:0.8999  KBS1:1.0  MBC
MOVIES:0.8999  DTN드라마:0.8999
KBS Sky Drama:0.8999  캐치온:0.8999  SBS:1.0
Home CGV:0.8999
공통 모델의 키워드 백터 :
섹시:0.9889  후배:0.9889  사회:0.9889  이름:0.9889  기운:0.9889
선호 인물 :
서경석:0.8999  파스칼:0.8999  윤여정:0.8999  변정수:0.8999  마이클:0.8999  앤:0.8999  존:0.8999  김수현:0.8999  린치:0.8999  빈:0.8999  신화:0.8999...
    
```

그림 8. 갱신 후 사용자 모델

장되고 주기적으로 H-TBL에 저장된 정보를 이용하여 사용자 모델을 갱신한다. 그림 8은 갱신된 사용자 모델 정보를 나타낸다. 해당 정보는 다음 추천에 이용된다. 이때, 공통 모델의 키워드 백터도 갱신됨으로써 다른 추천 서비스에 이용할 수 있다.

본 구현에서 TV도우미 태스크는 UserEntered와 SpeechReceived의 CAMUS 제공 이벤트를 이용한다. UserEntered는 TV 위치에서 사용자가 인식되었을 때 발생하는 이벤트로써 RFID 태그 또는 영상 인식을 통하여 발생된다. RFID를 이용하는 경우에는 UserEntered 이벤트에는 사용자 id 정보와 위치 정보가 포함된다. SpeechReceived 이벤트는 음성 인식에 의해 발생하는 이벤트이다.

그림 9는 그림 3의 TV 서비스 태스크의 상태 전이도에서 TVON 상태에서의 작업을 PLUE로 기술한 것이다. TVON 상태가 시작되면 TV 프로그램 추천 서비스 에이전트를 통하여 추천 목록을 작성하고 TV 제어 서비스 에이전트에게 가장 상위 채널로 이동하도록 명령한다. \$place는 현재 위치에서 실행 가능한 서비스 에이전트를 의미한다. \$owner는 해당 태스크를 실행하는 사용자의 식별자를 나타낸다. “on” 절에는 작업 규칙의 수행을 유발하는 이벤트 연산식(event SppechReceived(e))이 기술되고, “condition”절에서는 해당 작업 규칙이 수행되기 위한 조건을 기술하고, condition절 내부는 “action”에 해당하는 부분으로써, 명시된 조건이 만족하는 경우 수행될 서비스들을 기술한다. TVON 상태에서 다른 상태로 이동(transition)하는 경우에는 TV를 끄고 이동한다. 그림 10은 TVON 상태에서 “TV 중

료”라는 음성 명령과 사용자의 위치가 변경된 경우 TVOFF 혹은 Starting 상태로 전이하는 구현 예를 보여준다.

```

State TVON {
  entry {
    TVProgram info =
      $place.tv.getRecommendationInfo($owner);
    $place.tv.setChannel(info.channel);
  }
  exit:$place.tv.setPower(false);

  on event SpeechReceived(e)
  condition(e.symbol == 'volume up')
    $place.tv.volumeUp();

  on event SpeechReceived(e)
  condition(e.symbol == 'volume down')
    $place.tv.volumeDown();
}
    
```

그림 9. TV 도우미 태스크의 TVON 상태 구현 예

```

Transition StateTVOn -> StateTVOff {
  on event SpeechReceived(e)
  condition(e.symbol == "tv off")
    $place.tv.setPower(false);
}

Transition StateTVOn -> Starting {
  on event UserEntered(e)
  condition($owner.lastLocation() !=
    $owner.currentLocation())
    $owner.tv = true
    $place.tv.setPower(false);
}
    
```

그림 10. TVON상태에서 특정 이벤트 발생시 다른 상태로 전이하는 경우

그림 11은 [예 1]의 시나리오를 실험한 화면을 나타낸다. 사용자 위치는 RFID 센서와 카메라 센서를 통해서 인식한다. 사용자가 거실에 들어왔을 때 TV가 켜지고 사용자가 안방으로 이동하면 거실의 TV는 꺼지고 안방의 TV가 켜진다. 공부방에는 TV가 없는 상태이므로 동작하지 않는다. 다시 거실로 이동하면 TV가 켜지고 추천 채널로 이동한다.



(a) 사용자가 거실에 있는 경우



(b) 사용자가 거실에서 안방으로 이동한 경우



(c) 사용자가 안방에 있는 경우



(d) 사용자가 안방에서 공부방으로 이동한 경우



(e) 사용자가 공부방에 있는 경우

그림 11. 실험 환경 및 [예 1] 시나리오 실험 결과

VI. 결론

다채널을 통한 TV 방송 프로그램의 증가는 사용자가 시청할 수 있는 프로그램이 다양해졌으나, 사용자는 수많은 프로그램을 탐색하여 선별하는 노력을 요구 받게 되었다. 이러한 문제를 해결하기 위하여 사용자 선호도 정보와 방송 프로그램 정보를 이용하여 사용자 취향에 보다 적합한 프로그램을 추

천하는 지능형 방송 추천 서비스에 관한 연구가 진행되어왔다. 기존 연구에서는 사용자의 프로파일 정보를 활용하여 많은 방송 프로그램으로부터 사용자의 요구에 따라 원하는 방송 프로그램을 추천하는데 중점을 두고 개발되었다. 하지만, 유비쿼터스 컴퓨팅 환경이 도래함에 따라서 사용자의 요구에 따라 반응하는 서비스 보다는 상황을 인지하고 반응하는 능동적인 서비스를 지원할 수 있는 기술이 필요하다.

본 논문은 CAMUS 상황인식 시스템을 이용하여 개인의 위치 상황 정보에 따라 능동적으로 서비스를 제공할 수 있는 TV 응용 서비스를 제안하고 구현하였다. 본 논문에서 제안하는 상황인식 서비스는 사용자 위치 정보 혹은 음성 이벤트를 받아서 능동적인 프로그램 추천 및 TV 제어 서비스를 제공한다. 제안된 TV 프로그램 추천 서비스 에이전트는 내용 기반의 추천 기법을 바탕으로 수백 개의 채널에서 방송되는 프로그램 및 사용자의 선호도 정보를 분석하여 사용자가 원하는 프로그램을 추천한다. 사용자가 추천 목록 중에서 원하는 프로그램을 선택하면 그 정보는 히스토리 정보 테이블인 H-TBL에 저장되고 사용자 정보가 갱신되어 다음 추천에 이용한다. 또한 공통 모델과 TV관련 세부 모델로 나누어 사용자 모델을 정의하였기 때문에, H-TBL을 통해서 공통 모델의 키워드 백터도 갱신됨으로써 다른 추천 에이전트로 확장이 용이하다. 뿐만 아니라, 구현된 태스크는 사용자의 위치에 따라 상태가 전이되므로 다양한 환경에 적용이 가능하다.

참고 문헌

- [1] J. Ryu, M. Kim, J. Nam, K. Kang and J. Kim, "User Preference based Intelligent Program Guide," Journal of The Korean Society of Broadcast Engineers, 7(2), 2002, pp. 153-167.
- [2] W. Lee and T. Yang, "Personalizing Information Appliances: a Multi-agent Framework for TV Programme Recommendations," Expert Systems with Applications, Vol. 25, 2003, pp. 331-341.
- [3] L. Ardissono et al., "Personalized recommendation of TV Programs," LNCS, No. 2829, pp. 474-486, 2003.
- [4] W. Dai and R. Cohen, "Dynamic Personalized TV Recommendation System," Proc.

Workshop on Personalization in Future TV, 2003.

[5] K. Kurapati, "A Multi-Agent TV Recommender," Proc. Workshop on Personalization in Future TV, 2001.

[6] K. Kurapati and S. Gutta, "TV Personalization through Stereotypes," Proc. Workshop on Personalization in Future TV, 2002.

[7] J. Xu, L. Zhang, H. Lu and Y. Li, "The Development and Prospect of Personalized TV Program Recommendation Systems," Proc. IEEE Symp. Multimedia Software Engineering, 2002, pp. 82-89.

[8] S. Yoo, H. Lee, H. Lee and H. Kim, "A Content-based TV Program Recommender," Journal of KISS, 9(6), 2003, pp. 683-692.

[9] A. Dey, D. Abowd and D. Salber, "Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications," HCI Journal, Vol. 16, 2001.

[10] H. Kim et al., "A Software Robot Technology in URC," Journal of The Korean Institute of Communication Sciences, 21(10), 2004, pp. 36-43.

[11] B. Schilit, N. Adams, and R. Want. "Context-aware Computing Applications," Proc. IEEE Workshop on Mobile Computing Systems and Applications, 1994, pp. 85-90.

[12] R. Want, A. Hopper, V. Falcao and J. Gibbons. "The Active Badge Location System," ACM Transactions on Information Systems, Vol. 10, No. 1, 1992, pp. 91-102.

[13] G. Abowd et al., "Cyberguide: A Mobile Context-aware Tour Guide", Wireless Networks, Vol. 3, No. 5, 1997, pp. 421-433.

[14] A. Asthana, M. Cravatts and P. Krzyzanowski. "An indoor wireless system for personalized shopping assistance," Proc. IEEE Workshop on Mobile Computing Systems and Applications, 1994, pp. 69-74.

[15] A. Dey, M. Futakawa, D. Salber, and G. Abowd, "The Conference Assistant: Combining Context-Awareness with Wearable Computing," Proc. Symp. Wearable Computers, 1999, pp.

21-28.

[16] P. Brown. "Triggering Information by Context", Personal Technologies, Vol. 2, No. 1, 1998.

[17] K. Römer, "Tracking Real-World Phenomena with Smart Dust," LNCS, No. 2920, 2004, pp. 28-43.

[18] B. Warneke, M. Last, B. Leibowitz and K. Pister, "Smart Dust: Communication with a Cubic-Millimeter Computer," IEEE Computer Magazine, Vol. 34, No. 1, 2001, pp. 44-51.

[19] M. Balabnovic and Y. Shoham, "Content-Based, Collaborative Recommendation," CACM, Vol. 40, No. 3, 1997, pp. 66-72.

[20] L. Ardissono, "User Modeling and Recommendation Techniques for Personalized Electronic Program Guides," Personalized Digital Television, Human-Computer Interaction Series, Vol. 6, 2004.

[21] <http://www.tivo.com>

문 애 경 (Aekyung Moon)

정회원



1992년 영남대학교 전산공학과 학사
 1996년 영남대학교 컴퓨터공학과 석사
 2000년 영남대학교 컴퓨터공학과 박사

2000년~현재한국전자통신연구원 소프트웨어로봇연구팀 선임연구원
 <관심분야> 분산/병렬 데이터베이스, 상황인식 미들웨어

이 강 우 (Kangwoo Lee)

정회원



1991년 서울대학교 계산통계학과 학사
 1993년 서울대학교 전산과학과 석사
 2000년 서울대학교 전산과학과 박사

2000년~현재한국전자통신연구원 소프트웨어로봇연구팀 선임연구원
 <관심분야> 분산시스템, 상황인식 미들웨어

김 형 선 (Hyongsun Kim)

정회원



1982년 상지대학교 전자계산학과 학사
1990년 광운대학교 컴퓨터공학과 석사
2003년 대전대학교 컴퓨터공학과 박사
1985년~현재 한국전자통신연구원

원 소프트웨어로봇연구팀 책임연구원

<관심분야> 분산컴퓨팅, 정보보호, 상황인식, 전자상거래, 분산데이터베이스

이 수 원 (Soowon Lee)

정회원



1978년~1982년 서울대학교 계산통계학과 학사
1982년~1984년 한국과학기술원 전산학과 석사
1988년~1994년 Univ. of Southern California 전산학과 박사

2003년~2003년 한국지능정보시스템학회 부회장

2003년~2004년 한국정보과학회 인공지능연구회 운영위원장

1995~현재 숭실대학교 컴퓨터학부 부교수

김 현 (Hyun Kim)

정회원



1984년 한양대학교 기계설계학과 학사
1987년 한양대학교 기계설계학과 석사
1997년 한양대학교 기계설계학과 박사
1998년~1999년 한양대학교 산

업공학과 겸임교수

1990년~현재 한국전자통신연구원 소프트웨어로봇연구팀장, 책임연구원

<관심분야> Intelligent System, Distributed Computing, Context-Awareness, Engineering Knowledge Management