
URC 로봇 원격 모니터링기술 개발

The Development of Remote Monitoring Technology for URC Robot

김주만

부산대학교 바이오정보전자

Joo-Man Kim(joomkim@pusan.ac.kr)

요약

유비쿼터스 환경에서 지능형 로봇의 실시간 원격 제어 및 모니터링 기술은 로봇의 센서나 카메라 영상으로 획득한 다양한 콘텐츠를 시간과 지역에 비 의존적으로 공유할 수 있는 서비스를 창출한다. 본 논문에서는 지능형 로봇의 원격 실시간 감시/제어 방법을 제안한다. URC는 기존의 로봇에 네트워크 및 정보 기술을 접목하여 유선 및 무선 통신이 가능한 외부 디지털 장치와 상호 연동하도록 설계된 지능형 로봇이다. ISSAC4라는 국민 로봇에 본 연구결과를 적용하고, 실용적 가치를 검증하였다. 연속성을 보장하는 Client-Pull 방식의 실시간 영상 전송 방식으로 위치에 상관없이 웹 브라우저를 통한 원격 제어가 가능하도록 설계하였다.

■ 중심어 : | 로봇 | 실시간 | 임베디드시스템 | 웹서버 | 모니터링 | 유비쿼터스 |

Abstract

In the ubiquitous environment, the real-time remote control and monitoring technology for intelligent robot creates service as a sharable and independent of time-location for various contents to get from a sensor or camera of the robot. In this paper, We propose the real-time monitor and control mechanism for intelligent robot called URC(Ubiquitous Robotic Companion). URC are intelligent robots designed as to interact with external digital device that can communicate through wire or wireless by integration the network and information technology into traditional robot. It has been carried out by implementing this technology into the target robot called ISSAC4 and proving its practical worth. We designed feasibly to control on remote site by web-browser. It guarantees a continuity of real-time image transferring by Client-Pull method.

■ keyword : | Robot | Real-Time | Embedded System | Web Server | Monitoring | Ubiquitous |

1. 서론

최근 초고속 인터넷 기술의 확산으로 인터넷을 기반

으로 하는 새로운 시장 창출이 다각화 되고 있다. 특히 인터넷이 개인 생활 속으로 파고들면서 유비쿼터스 환경 구축을 향한 서비스개발이 가전기구나 개인정보 단

말기 등 서비스 접점의 위치와 시간에 비 의존적인 서비스 개발이 활발하게 진행되고 있다[1-4].

URC(Ubiquitous Robotic Companion)는 기존의 로봇에 네트워크 및 정보 기술을 접목하여 유선 및 무선 통신이 가능한 외부 디지털 장치와 상호 연동하거나, 각종 센서 인식으로 처리하여야 할 기능을 외부 장치에 분담시킴으로써 로봇 구성을 단순화하고, 가능한 외부의 고성능 서버를 활용하여 협업하는 구조를 지향하는 새로운 개념의 지능형 로봇으로 국민 보급형 로봇으로 개발 중에 있다[9][10].

일반적으로 로봇은 센싱하고, 판단하며 이를 바탕으로 결정된 행동을 수행하는 기본적인 기능을 가진다. URC 로봇은 이들 세 가지 기능을 네트워크와 연동하여 주변 의 고성능 서버나 제어기에 기능을 분산하자는 것이며, 네트워크를 통하여 원격지에서 인지할 수 있도록 하자는 개념이 내포되어있다. 본 연구는 이러한 URC의 기본 개념 하에 실시간 원격 모니터링 기법을 구현하는데 있으며, 사용자는 로봇을 떠나 언제 어디서나 다양한 로봇의 서비스는 물론 로봇에게 명령전달이나 로봇의 상태를 인지할 수 있을 것이다.

로봇의 원격 모니터링 기술은 사용자가 기기가 위치한 현장에 직접 가지 않고 원격에서 제어 및 모니터링을 가능하게 하는 매우 유용한 기술로서, 로봇 내부 이벤트 정보의 모니터링 및 인간과 상호 작용이 원격 제어가 가능하도록 하는 웹 서버 기반의 제어 기술이 요구된다 [6][7].

본 논문에서는 임베디드 시스템의 제한된 자원을 수용하면서, 원격지 웹 브라우저를 통한 동영상 및 상태 및 제어 신호가 실시간 교환이 가능하도록 실시간 운영 체제인 Q+ 기반의 웹기반 로봇 모니터링 기법을 설계 및 구현하며, 국민 생활 로봇에 적용하기 위하여 기존의 원격 제어 기법들을 통합하여 새로운 서비스 프레임워크 및 콘텐츠를 제공하고 있다. 따라서 본 연구의 구현 결과는 향후 가사용 로봇의 원격 사용자 인터페이스를 위한 URC/GUI를 제공할 것이다.

본 논문의 구성은 2장에서 시스템 구성도, 하드웨어 및 소프트웨어 구조도를 보이고, 3장에서 모니터링 구성 모듈과 UI 콘텐츠 설계 방식을 설명하고, 4장에서 구현

방법과 결과에 대하여 기술하고 5장에서 결론을 맺는다.

II. 시스템 구조

1. 시스템 운용 환경

URC 로봇은 최소한의 센싱 기능과 프로세싱 기능을 갖는 보급형 국민 로봇으로 최소한의 메모리, 저성능의 프로세서 등 제한된 자원을 갖는 임베디드 시스템으로 구성된다. 그럼에도 불구하고 로봇의 상태는 물론, 명령 전달, 영상 정보 등의 실시간 제어가 원격지에서도 가능한 성능을 갖추어야 할 것이다[5].

URC 로봇은 홈네트워크 게이트웨이에 무선 AP에 결합되고, 인터넷상의 URC 서버나 다양한 통신 단말기의 웹 브라우저에서 접근할 수 있는 경로를 제공한다. 가정의 정보 가전기기는 홈 네트워크에 연결되어 URC 로봇을 통하여 모니터링이 가능하도록 조직되었다.

사용자 인터페이스는 기본적으로 웹을 통해서 모든 기능을 사용할 수 있도록 하며, 시스템 확장을 위한 유연성을 제공하도록 한다. 사용자 인터페이스가 사용자에게 보여 지는 것이라면, 실제적으로 사용자의 입력을 처리하거나 사용자에게 보여 지는 데이터를 생성하기 위하여 상태 모니터링 모듈, 로봇 제어 모듈, 그리고 홈 네트워크에 접속된 정보 가전기기를 제어하기 위해 연체가 필요한 홈 서버 연계모듈로 크게 나눌 수 있다.

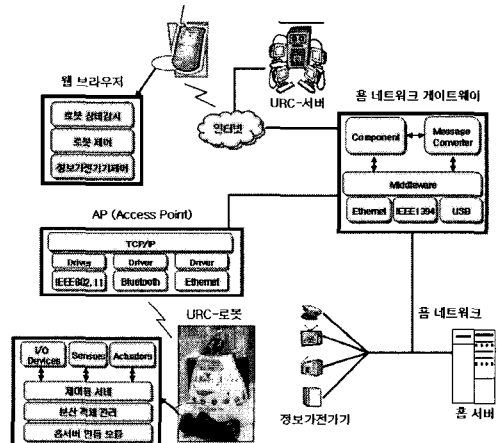


그림 1. URC 로봇 운용 구조

URC 로봇에 탑재된 웹 서버의 기능은 로봇의 기능을 관리하고 제어하는 데 그 목적이 있다. 이러한 기능은 [그림 1]과 같이 URC 로봇이 운용되는 전체적인 구조를 통해 더욱 명확해질 수 있다.

URC 로봇은 단순하고 반복적인 일을 대신 해주는 가정의 도우미나, 사용자가 외출 시에 집안을 대신 관리해주는 역할, 또는 사용자에게 인터넷을 통해 제공할 수 있는 다양한 서비스를 제공해주는 도우미 역할을 한다. 사용자는 음성이나 터치스크린을 이용하여 가정에서 URC 로봇을 사용할 수도 있으며, 웹을 통하여 원격에서 집안 내부의 영상을 보거나, 안정상의 문제점을 이동 통신 디바이스를 통해 전달 받을 수 있다.

URC 로봇을 홈 네트워크에 연동하면 외부에서 홈 서버를 통하여 로봇의 감시나 제어가 가능할 뿐만 아니라, 반대로 로봇을 통해 홈 네트워크에 접속된 모든 정보가 전기기의 사양이나 네트워크를 알지 못하더라도 홈 서버를 통하여 이들 가전기기의 제어가 가능하게 된다. 홈 네트워크는 다양한 업체에서 개발된 제품들의 제어를 위해, 표준적인 방법을 통해 제품 독립적인 미들웨어를 포함하여, 집안의 기기들을 배타적으로 제어할 수 있도록 한다.

URC 로봇은 홈 네트워크와 연동되어, 홈 서버에서 제공하는 기능을 사용자에게 보여주거나 사용할 수 있도록 할 수 있는데, 사용자는 URC 로봇을 통해 집안 전체를 파악할 수 있는 시각적인 정보 이외에 가전 기기를 제어할 수 있다.

2. URC 로봇 하드웨어 구조

URC 로봇은 다양한 센서와 액츄에이터, 모터 등으로 구성되어 있다. 원격 모니터링이란 이러한 제어기나 자원들의 상태 및 제어를 원격에서 가능하도록 하는 사용자 인터페이스의 제공이다. 이들 자원들은 리눅스 실시간 운영체제인 Q+ 하에서 구동되는데, 각 자원들에 대한 API를 라이브러리 형태로 제공한다. [표 1]은 URC 로봇인 ISSAC 에서 제공하는 모니터링 가능한 자원과 그 속성을 보여주고 있다. 각 자원의 항목들은 제어 및 모니터링 대상이며, 로봇 웹 서버의 후면 서버는 이들 자원의 API를 사용하여 자원들에 접근하게 된다.

표 1. 로봇 모니터링 자원 정보

Category	Item	Contents
Sensor Data	Wheel Motor Encoder	Left Wheel Motor Status Right Wheel Motor Status
	Sonar	Sonar Status(On/Off)
	PSD	전방 3set의 상태(On/Off) 후방 2set의 상태(On/Off)
	Bumper	전방 2set의 각 상태 후방 2set의 각 상태
	Dock	on(충전스테이션 접촉 여부)
	IR(docking)	on(충전스테이션 신호 탐비 여부)
Dead-Reckoning	Battery Level	100%(Battery 잔량)
		Position & Orientation (x, y, θ) Linear & Angular Velocity (v, w)
Camera		Pan & Tilt Angle (ϕ, γ)
Map Information		Obstacle Map (Grid) 200x200
Operation Mode & System Monigoring		Robot's Internal Mode & Status

3. 소프트웨어 구조

[그림 2]는 소프트웨어 구조도를 도시하고 있다. URC 로봇에는 BOA 웹 서버가 탑재되며, 후면(Background) CGI 서버에 의해 로봇 기능을 제어하거나 모니터링 한다. 이들 후면 CGI 서버는 기능적으로 제어 모듈, 모니터링 모듈 그리고 홈 서버 연동 모듈로 구성된다. 공통 모듈은 DBMS나 장치 정합을 위한 공통 라이브러리 모듈이다. DBMS는 로봇의 상태를 주기적으로 갱신하면서 일정 기간의 이벤트를 기록하며, 원격지 요청에 따라 시간 트레이스를 제공할 수 있는 소형 임베디드 실시간 모바일 데이터베이스이다.

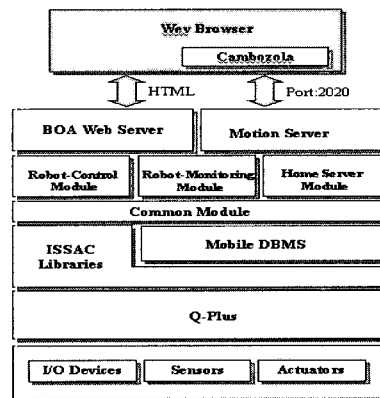


그림 2. 소프트웨어 구조도

III. 로봇 모니터링 모듈의 설계

1. 세부 구성 모듈

내장형 웹 서버는 웹 서버의 기능에 충실하면서 URC 로봇 모니터링 및 가전기기의 감시와 제어를 안정적으로 수행할 수 있도록 불필요한 기능은 배제하고, 각각의 모듈을 최적화하여 필요로 하는 기능을 수행 가능하도록 설계된다.

웹 서버의 모든 기능은 웹을 통해서 접근이 가능하며, 사용자는 웹을 통해 홈 서버와 URC 로봇과 연계된 기능을 사용하여 원격 제어 및 모니터링 할 수 있다. URC 로봇과의 연동은 URC 로봇의 ISSAC 라이브러리를 통해서, 로봇의 입출력 디바이스와 센서, 그리고 Actuator 모듈과 통신을 할 수 있다. 또한 URC 로봇과 별개로 외부에 존재하는 홈 서버와의 연계는 소켓 프로토콜에 의한 무선 통신으로 이루어지며, ISSAC 라이브러리에 포함된다.

1.1 로봇 제어 모듈

로봇 제어 모듈은 카메라 틸트, 휠 제어 및 토키 스테이션 제어등을 관리하며, 센서나, 모터, 액츄에이터 등의 제어 장치들에 대한 사용자 명령의 해석과 실행에 관한 소프트웨어 모듈이다. 실시간성은 Q+ 실시간 운영체제에서 보장해 주며, 로봇 서버 내부에 분산된 자원들에 대한 분산 객체 관리 기법에 의한 내부 통신 망을 관리한다.

1.2 모니터링 모듈

로봇 모니터링 모듈은 로봇의 내부 각종 자원 및 상태를 모니터링 하기 위한 것으로, 사용자의 요청이나 또는 능동적으로 상태 변화에 대하여 응답한다. 로봇에서 제공하는 영상 스트림은 DB에 큐 로깅(Queue Logging)하여 후진 검색이 가능하도록 하며, 로봇 자율적인 오류나, 이상상태등도 DB에 로깅되어 모니터링 명령을 통해 검색이 가능하도록 한다.

1.3 홈서버 연동 모듈

홈 서버 연동 모듈은 홈 네트워크에 연결되어 있는 가

전기기를 제어하기 위해, 홈 서버와의 연동을 담당하는 모듈이다. 홈 서버는 URC 로봇과 무선 랜에 의해 물리적으로 구별되어 있으며, 사용자의 요청은 제어와 모니터링으로 분류하여 소켓 프로토콜에 의해 구현된다.

1.4 공통 모듈

공통모듈은 로봇 제어 모듈, 모니터링 모듈, 홈서버 연동모듈들이 공통적으로 사용하는 기능으로 구현된 라이브러리이다. 모바일 DB와 ISSAC 라이브러리로 구성되며, 다양한 프리미티브를 제공하여 각 구현 모듈의 구현을 용이하게 한다.

1.5 사용자 인터페이스

사용자 인터페이스는 웹 서버의 모든 기능을 사용할 수 있도록 HTML을 통한 웹 인터페이스를 제공한다. BOA 임베디드 환경에서 제한된 자원을 지원하는데 적합한 웹 서버로 사용자 인터페이스의 미려함은 떨어지나, 가사로봇인 URC 로봇을 제어하는데 적합한 다양한 콘텐츠로 구성한다.

웹 서버는 클라이언트의 웹 브라우저를 통하여 사용자 인터페이스 창을 생성하는데, 여러 개의 페이지로 구성되어 목록에 따라 용이하게 접근할 수 있도록 설계된다.

2. 구성 모듈의 설계

2.1 로봇 제어 모듈

로봇 제어 모듈은 웹 인터페이스를 통해서 들어온 사용자 입력에 따라 로봇 제어를 수행하는 모듈로서 [그림 3]과 같이 3개의 서브 모듈로 구성된다.

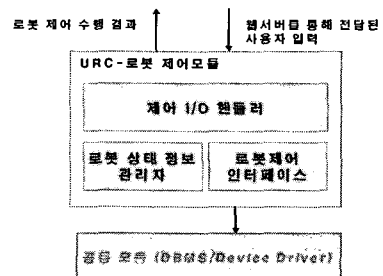


그림 3. URC 로봇 제어 모듈 구성도

제어 I/O 핸들러는 사용자로부터의 웹 프로토콜 접속을 처리하며 하부 로봇의 모션 제어를 위한 상태 및 제어 명령을 생성 분기하도록 한다. 즉, 웹 요청을 해석하여 로봇 명령을 생성하고, 그 결과를 메시지화하여 웹 서비스 서버에게 전달하도록 설계한다.

2.2 모니터링 모듈

모니터링 모듈은 사용자 입력에 반응하여 로봇의 상태 및 영상 정보를 로봇의 관점에서 사용자 뷰에 제공하도록 처리하는 모듈로서 [그림 4]와 같이 2개의 서브 모듈로 구성된다.

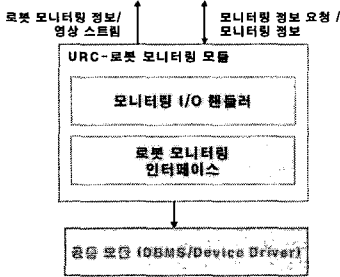


그림 4. URC 로봇 모니터링 모듈

웹 서버를 통해 사용자로부터 들어온 모니터링 요청은 로봇을 구성하고 있는 센서와 Actuator, 파워에 대한 상태에 대한 정보 요청을 처리한다. 모니터링 I/O 핸들러는 주기적으로 요청되는 모니터링 명령을 인지하여 영상 모니터링 및 로봇의 상태 모니터링을 데이터베이스와 연동하여 수집되어 축적된 정보를 가공하여 사용자에게 제공하도록 한다.

2.3 홈서버 연계 모듈

홈 서버 연계 모듈은 로봇의 사용자의 입력에 반응하여 홈 서버와 인터페이스 하도록 연동되며, 소켓에 의한 연결을 제공한다. 홈 서버 연계 모듈은 [그림 5]와 같이 3개의 서브 모듈로 구성된다.

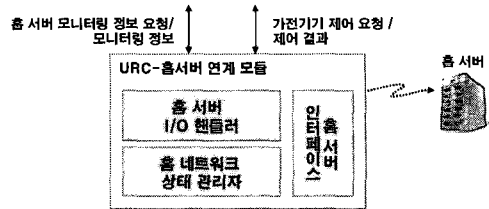


그림 5. 홈서버 연계 모듈 구성도

웹 서버를 통해 사용자로부터 들어온 모니터링 및 가전기기에 대한 제어 요청은 홈 서버와의 연계를 통해 수행하며, 홈 서버가 인식할 수 있는 형식으로 명령 패킷을 구성하며, 웹으로부터의 요청을 정의된 형식에 맞게 변환하는 역할을 한다. 홈 네트워크 상태 관리자는 홈 서버 I/O 핸들러로부터 오는 요청을 받아 홈 서버 인터페이스의 API를 호출하여 그 결과를 구성하는 모듈로, 내부적으로 홈 네트워크에 연결된 가전기기의 ON/OFF상태와 ON 상태의 상태정보를 유지 관리하도록 설계한다.

2.4 공통 모듈

공통 모듈은 실시간 운영체제인 커널인 Q+와 DB에 접속하기 위해 라이브러리 형태로 구현된다. 이 공통모듈은 이기종간 표준 인터페이스로서 모듈간의 호환성과 적응성을 고려한 디바이스 드라이버 형식을 가지며, [그림 6]과 같이 구성된다.

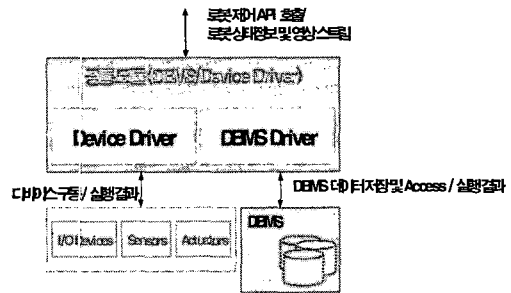


그림 6. 공통 모듈 구성도

Device Driver 모듈은 로봇의 다양한 장치 제어가 가능하도록 설계되며, 상위 모듈에 장치 상태나 제어가

가능하도록 라이브러리 형태로 제공한다. 영상 로깅 및 탐색을 위하여 영상 정보는 일시적으로 DB에 저장 관리된다. 또한 로봇의 보안 기능을 위하여 인터넷으로 접속한 로깅 정보를 트레일스 할 수 있도록 DB에서 관리한다. DB Driver는 DBMS를 구성하고, 검색 및 조회 등을 위한 인터페이스를 제공하도록 설계한다.

3. 사용자 인터페이스(UI) 설계

3.1 UI Map

사용자 인터페이스 맵은 기능에 따라 [그림 7]과 같이 크게 로봇 제어와 로봇 모니터링, 그리고 영상정보보기로 나눌 수 있다. 사용자 인터페이스는 웹 서버 상에서 HTML과 CGI로 작성한다.

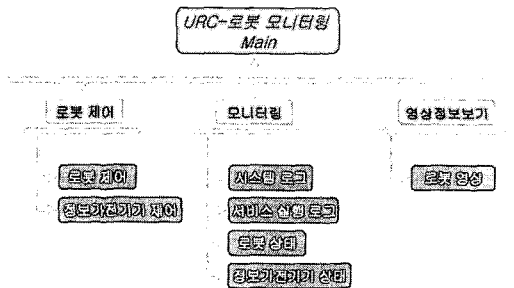


그림 7. 사용자 인터페이스 맵

로봇에서 제공하는 웹 서버의 경우, 로봇을 제어하거나 상태를 모니터링 한다는 점에서 임의의 사용자가 사용하는 것이 아니기 때문에 웹 인터페이스의 초기화면에는 사용자 인증을 통하여 로봇 세션 관리 권한을 갖도록 한다. 웹 서버의 후면 서버는 CGI로 프로그래밍되며, 로봇 운영체제인 Q+ 시스템 호출 프로그램에 의해 로봇 센서나 장치 제어 프로그래밍 또는 DB API로 프로그래밍 한다.

3.2 메인 UI

사용자 인터페이스 맵에 따라 로봇 서버의 URL로 접속할 때 처음 보이는 화면이다. 로봇의 기능에 따라 각 모듈에 접속할 수 있도록 Top-Menu와 Left-Menu로 구성한다.

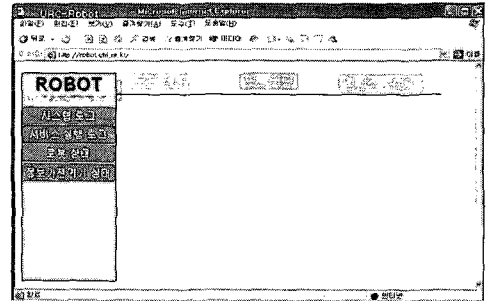


그림 8. 메인 화면

3.3 접속 인증 UI

로봇 서버에 접속하기 위하여 인증을 하도록 한다. 로봇 관리자 및 등록된 사용자는 id와 비밀번호를 통하여 접속할 수 있다. 세션 번호를 부여 받은 사용자는 로봇 관리 권한을 부여받는다. 로봇 통제 권한을 부여 받은 사용자는 로봇 통합 관리자로부터 id와 비밀번호를 부여 받는다. 따라서 사용자 등록과 비밀번호 등록을 위한 인터페이스를 제공한다.

3.4 로봇 제어 UI

로깅 절차를 통하여 인증 받은 사용자는 로봇 세션 권한을 부여받는다. 로봇 제어는 카메라 방향 제어와 구동부인 모터에 의한 움직임 제어이다. 구동부를 클릭하면 제어 페이지로 이동하며, 마우스 위치 시에는 간단한 상태 정보를 보여준다.

3.5 로봇 상태 정보 UI

로봇의 센서 및 컨트롤러 이미지를 클릭하면 로봇 상태 보기 페이지로 이동한다, 이미지의 각 센서 및 컨트롤러 부분에 마우스 포인터를 접근 시키면 각 부분의 간략한 상태 정보가 표시된다. 로봇 각 자원들에 대한 세부적인 상태는 별도의 화면으로 보여준다.

3.6 가전기기 제어 UI

로봇과 무선으로 연결된 홈 서버의 정보가전기기 리스트를 보여주고 사용자는 실행 및 정지 버튼에 의해 가전기기 제어가 가능하도록 설계 한다. Update 버튼을 두어 제어가 반영되면 현재 상태가 갱신된다.

3.7 가전기기 상태 표시 UI

로봇에 무선으로 연결된 홈서버의 정보가전기에 대한 현재 상태를 표시한다. Refresh 버튼을 두어 현재 실행 상태를 표시하도록 한다.

3.8 서비스 실행 로그 UI

사용자는 로봇 모니터링 시스템의 서비스 요청에 대한 실행 히스토리를 조회할 수 있다. 실행 날짜를 기준으로 검색할 수 있도록 하였다.

3.9 실행 로그 UI

사용자는 로그의 갱신 주기와 표시 라인수를 설정하여 실행로그를 모니터링 할 수 있다.

4. URC 로봇의 영상 정보 전송

4.1 영상 전송 방식 선정 기준

영상 전송 방식의 선정은 다음의 기준에 따라 선정하였다.

- **목표 지향성** - 영상정보를 사용자에게 제공하는 방식을 선정함에 있어, 영상정보가 사용되는 목적에 충분히 적합하여야 한다. 예를 들어 영상정보가 로봇의 움직임을 제어하기 위한 것이라면, 영상정보는 실시간 영상이어야 하며, 시간적인 버퍼가 최소이어야 한다.
- **편의성** - 사용자는 웹 브라우저를 사용하여 URC 로봇이 제공하는 영상정보를 보는데 있어, 사용상의 편의성을 제공하여야 한다. 사용자는 추가적인 프로그램 설치와 같은 영상을 보기위한 추가적인 작업이 최소화되어야 한다.
- **호환성** - 영상정보는 Microsoft와 Netscape로 크게 구분되는 웹 브라우저에서 정상적으로 실행되어야 한다. 특정 웹 브라우저에서만 실행되는 방식은 지양되어야 한다.
- **성능** - 영상정보를 사용자의 웹 브라우저에 제공할 때에는 가장 효율적인 방식을 제공하여야 한다.

표 2. 영상전송 방식 평가

분류	구현방법	목표지향	편의성	호환성	성능
Client Pull	Meta Tag	●	●	●	▲
	Java Applet	●	●	●	●
	Java Script	●	●	●	▲
Server Push		●	▲	X	●

● 양호 ▲ 보통 X 미흡

위의 [표 2]에 나타나듯 Client Pull 방식 중 Java Applet 방식이 선정 기준에 가장 부합됨을 알 수 있다. Client Pull 방식은 클라이언트에서 주기적으로 웹 서버에 접속하거나, 웹 서버의 특정파일을 읽어 사용자에게 보여주는 방식이다. Java Applet은 URC 로봇 내의 영상정보를 제공하는 서버와 연결되어 사용자의 웹 브라우저에 영상정보를 계속해서 갱신하는 방식으로 사용자에게 마치 동영상을 보는 듯한 효과를 제공한다. 또한 HTML 메타 태그와 Java Script와는 달리 깜빡임 현상이나 성능에 있어 더 나은 결과를 보여주고 있다

4.2 URC 로봇 영상 서버

영상전송 서버는 제공하는 정보가 스트림(동영상)인가, 이미지인가에 따라 영상 스트림 서버와 영상 이미지 서버로 구분한다. 이때, URC로봇에서 전송되는 포맷의 차이는 클라이언트에서 사용할 기술에 영향을 미치게 되는데, 기술 결정을 위해 영상 스트림과 영상 이미지를 사용한 방법을 각각 사용하여, 그 장단점을 비교 한다. 각각의 방법의 장단점 비교는 웹캠 드라이버(pwc, pwcx)상에서 영상 스트림 서버와 영상 이미지 서버를 설치한 후, 웹 브라우저나 윈도우즈 미디어 플레이어를 이용하여 그 결과를 평가한다. 평가 기준은 4.1절의 4가지 평가 요소를 적용하여 다음과 같은 결과를 보였다.

영상 스트림 전송 방식은 영상 스트림 형식의 인코딩하는 시간으로 인해, 빠른 응답시간을 위해서는 추가적인 하드웨어 업그레이드가 요구된다. 또한 실시간 동영상을 사용자의 웹 브라우저에 보여주기 위해서는 웹 브라우저에 윈도우즈 미디어 플레이어와 같은 외부 프로그램이 임베드되어 사용되어야 하는데, 임의의 웹 브라우저에서나 윈도우즈 미디어 플레이어의 버전에 민감한 특징이 있다.

영상 이미지 전송의 경우는 Java Applet이 영상 이미지 서버로부터 얻은 이미지를 계속해서 교체하여 마치 동영상과 같은 효과를 내므로, 영상 스트림 서버와는 달리 지연 없이 웹 브라우저에게 전달할 수 있다. 최소한의 지연시간은 로봇의 움직임을 제어할 때, 사용자에게 변경된 이미지를 바로 보여주므로 영상 스트림 방식보다 여러 항목에서 적절함이 평가되었다.

지금까지 비교한 내용에 따라, URC 로봇에 영상 이미지 서버인 motion을 구축하고, 클라이언트에 Java applet을 사용하여 구현한다.

IV. 로봇 모니터링 구현

1. URC 로봇 관리 환경 구축

1.1 BOA 웹 서버 설치

BOA는 일반적인 웹 서버가 외부 연결 요청에 대해 프로세스를 생성하여 처리하는 방식과 달리, 성능과 안전을 목적으로 별도의 프로세스 생성(fork) 없이 외부의 요청을 처리하는 single-tasking HTTP 서버이다. BOA는 이러한 독특한 처리방식과 성능으로 인해 데이터 다운로드용이나 이미지와 동영상 전송을 위한 서버를 구현하는데 많이 사용되고 있으며, 대부분 임베디드 시스템의 자원 제약에 적합하여 주로 사용되고 있다.

본 논문에서는 BOA 웹 서버를 로봇 서버의 환경에 적합하도록 환경 구성 파일을 조정하여 설치한다.

1.2 카메라(웹캠) 설치

URC 로봇의 눈에 부착된 카메라는 로봇의 움직임에 대한 영상정보를 제공하며, 사용자는 웹 브라우저를 통하여 영상을 볼 수 있다. 웹 서버를 통하여 웹캠의 영상 정보를 획득하기 위해서는 Q+용 웹캠 디바이스 드라이버를 설치하여야 한다.

본 논문에서 사용한 카메라는 Philips Chipset을 채용한 Logitech QuickCam Pro 4000으로 USB 인터페이스로 컴퓨터와 연결되어 있다.

Q+를 위한 Philips Chipset Driver는 pwc, pwcx로 구성되어 있다. pwc는 linux 커널의 일부로서 사용되며, 기본적인 frame rate와 해상도의 영상정보를 얻을 수 있으나, pwcx의 경우 커널 밖에서 사용되어 캡의 모든 기능을 충분히 활용할 수 있도록 구성되어 있다. 두 모듈은 pwc에서 hook이라는 기능을 이용하여 보다 많은 기능을 제공하는 pwcx를 대신 메인 드라이버로 사용할 수 있도록 한다.

1.3 영상 이미지 서버 설치

영상 이미지 서버는 웹캠을 통해 계속적으로 정지화면 이미지를 제공받아 클라이언트에게 제공한다.

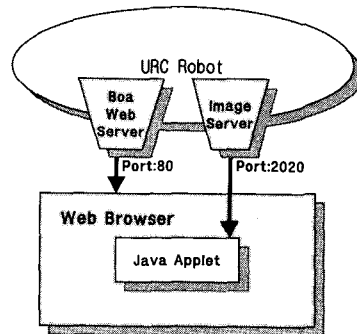


그림 9. 서버 포트 인터페이스

때문으로 실행되는 영상 이미지 생성 서버는 [그림 9]와 같이 웹 서버와 별도로 실행되어 웹 브라우저내의 애플릿과 직접 연결되어 영상이미지를 전달한다. 이와 같은 기능을 하는 도구는 오픈 소스로 배포되고 있는 motion이 있다.

motion은 오픈 소스로 개발되고 있는 움직임 감지 프로그램으로, 웹캠에서 생성된 이미지를 캡처하여 관심있는 움직임이 포함된 영상만을 저장할 수 있는 도구이다. 그러나 본 논문에서는 motion이란 도구의 "live webcam server" 기능을 이용하였다. live webcam server란 motion 내부에 포함된 간단한 http server로 영상정보를 계속해서 출력해주는 영상 이미지 생성 서버 역할을 한다.

2. URC 로봇 모니터링 구현 결과

2.1 메인 화면

웹 서버에 접속하여 인증 후 접속하는 최초의 메인 화면은 [그림 10]과 같다. 상부 메뉴와 좌측 메뉴로 구성되며, 그림에서는 시계의 초바늘의 움직임을 관찰하는 영상화면을 보여주고 있다.

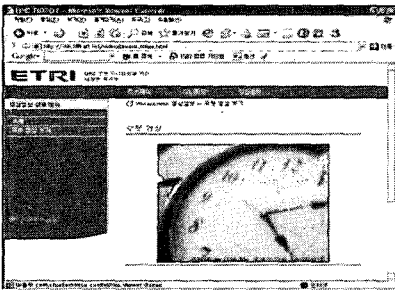


그림 10. 메인 화면

2.2 Java Applet 코드 구현

motion의 HTTP 서버는 2020 port에 영상 이미지를 10 프레임/초를 기록하면, 원격 Java Applet인 Cambozola 는 연속적으로 영상 서버 URL의 2020 Port로부터 영상을 취득하여 Viewer에 표시한다. Viewer의 규격을 설정할 수 있으며 영상 서버 접속 URL과 포트번호를 지정한다.

```

Java Applet
<applet code=com.charliemouse.cambozola.Viewer
archive=/Viewer.jar width="352" height="288">
  <param name=url value=http://168.188.60.161:2020/>
  <param name=accessories value=none>
</applet>
    
```

그림 11. 자바 애플릿 코드

2.3 웹 서버 모니터링 UI 구현

로봇 서버의 URL을 통해 웹 브라우저 화면을 통해 접속하면 인증과정을 거쳐 로봇 세션 매니저에 의해 세션 진입을 한다.

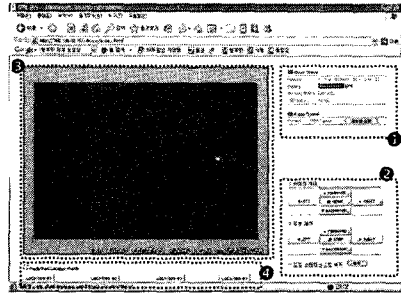


그림 12. 원격 모니터링 통합 화면

제어 및 모니터링을 위한 모듈간의 통합된 화면은 [그림 12]와 같이 모듈 단위의 기능에 대한 전체 메뉴로 구성된다. 모니터링 영역, 제어 영역, 영상 뷰 영역 및 위치 4개의 영역으로 구성된다.

① 모니터링 UI 영역

모니터링 영역에서는 다음 [표 3]의 내용과 같이 로봇의 상태를 실시간으로 [그림 13]처럼 보여준다. 리프레시 간격은 조정 가능하도록 한다.

표 3. 모니터링 영역

모니터링 항목	내 용
Posture	로봇의 좌표 위치
Battery	배터리 잔량(백분율)
Docking Status	Docking 센서 인지 여부
PSD Sensor	PSD 센서 정보
Control	제어권(원격 or 로컬)
Last Comm.	마지막 수행 명령

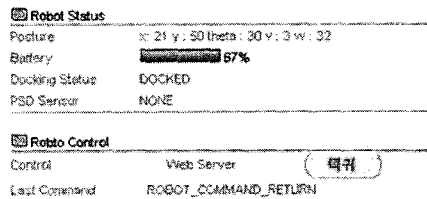


그림 13. 모니터링 UI 프레임

그래핑에 의해 구현되며, 가전기에 제어 및 모니터링에 대한 라이브러리를 통해 구현된다.

가. 가전기기 제어

[그림 18]은 오디오, TV, DVD 및 카메라에 대한 실험적 제어를 보여준다. 본 인터페이스는 기타 전등 및 전원제어, 일광량에 의한 차광제어, 에어컨, 세탁기 및 냉장고등으로 확장 가능하도록 구현되었다. Update 버튼을 통해 현재 제어 반영 결과를 보여준다.

Update				
가전기기	온라인	동작상태		비고
Audio	■	실행	중지	
TV	■	실행	중지	
DVD	■	실행	중지	
Cam	■	실행	중지	

그림 18. 온라인 제어

나. 가전기기 상태 정보

[그림 19]는 오디오, TV, DVD 및 카메라에 대한 온라인 동작 상태를 보여준다.

Refresh 버튼을 통해 현재 상태를 정정하여 보여 줄 수 있도록 한다.

Refresh			
가전기기	온라인	동작상태	비고
Audio	■	정지	
TV	■	실행	
DVD	■	정지	
Cam	■	연결되지 않음	

그림 19. 온라인 동작 상태 정보

2.7 실행 로그

가. 실행 로그

사용자는 로봇 모니터링 시스템의 서비스 요청에 대한 실행 히스토리를 조회할 수 있다. 갱신 주기 및 한 화면에 표시할 수 있는 라인수를 기준으로 검색할 수 있도록 한다.

갱신 주기(초) : 라인 수 : Update		
시스템 로그		
22078	Get Conection [monitor]	Success !!!
22078	Connection Success For [monitor]	
22594	doPost() end (1), user#[1]	532(ms)
25125	doPost() begin(2), user#[1]	
25172	doPost() end (2), user#[1]	47(ms)
25172	doPost() begin(3), user#[1]	

그림 20. 실행 로그

나. 서비스 실행 로그

사용자는 로봇에 접속한 서비스 단위로 어디서 얼마 동안 접속하였는지, 수행 결과 등에 대한 로그 정보를 제공한다. 검색 기간을 지정하여 접속에 관한 상세한 내역을 검색할 수 있도록 한다.

기간설정 2005 09 01 ~ 2005 09 31 검색					
서비스 수행 로그					
발행 일자	서비스이름	호출자	실행 시간	결과	
Wed Jul 28 14:34:54 +09:00 2005	conn_robot_inf	127.0.0.1	0ms	SUCCESS	
Wed Jul 28 14:52:05 +09:00 2005	get_weatherInfo	127.253.250.271	0ms	FAILURE	
Wed Jul 28 14:58:24 +09:00 2005	play_music	127.0.0.1	1ms	SUCCESS	

그림 21. 서비스 실행 로그

V. 결론

임베디드 웹서버 및 실시간 영상 모니터링 기법을 적용한 원격 로봇 모니터링 기술은 국민 생활 로봇인 URC 로봇을 웹 브라우저를 통하여 언제 어디서나 모니터링 가능하게하며, 본 연구를 통하여 실제 구현하였다.

국민 생활 로봇인 ISSAC4라는 로봇을 목표로 시스템으로 본 연구 결과를 구현하여 실용적 가치를 검증하였다. 로봇의 다양한 제어기, 장치, 액츄에이터, 센서 등의 상태 정보를 획득하고 연속성을 보장하는 Client-Pull 방식의 실시간 영상 전송 방식 및 로봇의 움직임 제어 등 위치에 상관없이 웹 브라우저를 통한 원격 제어가 가능하다.

본 연구에 의한 결과는 기존에 적용한 사례가 없으며, 다양한 기술이 통합되어 국민 생활 로봇에 최적화하도록

록 설계하였다. 국민 생활 로봇에 탑재되어 가정의 도우미 로봇으로서 실용화 될 때, 원격지에서 로봇의 상태 및 로봇 주변 환경 인식과 원격 명령을 통한 가전기기 제어, 보안 및 방재, 간호 도우미, 보육 기능을 통한 U-KOREA를 실현하는데 기여할 것이다.

본 연구는 향후 다양한 편제형 단말기에서도 모니터링 가능하도록 기술을 확장하고, 로봇 서버에서의 외부 인터페이스의 표준화를 통하여 이기종간의 응용 프로그램 호환 가능하도록 하는 미들웨어 기술에 대하여 연구가 이루어져야할 것이다.

참고 문헌

[1] B. McCombie, "Embedded Web Servers now and in the future," Real-Time Magazine, No.1, pp.82-83, Mar., 1998.

[2] A. Wilson, "The Challenge of Embedded Internet," Electronic Product Design, pp.31-34, Jan., 1998.

[3] I. Agranat, "Embedded Web Servers in Network Devices," Communication Systems Design, pp.30-36, Mar., 1998.

[4] 최재우, 노방현, 이창근, 차동현, 황희용, "임베디드 웹 서버를 이용한 원격 감시 및 제어 시스템 구현," 한국산학기술학회 논문지, 제4권, 제3호, pp.301-306, 2003.

[5] I. Agranat, "Engineering Web Technologies for Embedded Application," IEEE Internet Computing, Vol.2, No.3, pp.40-45, May, 1998.

[6] M. F. Zakaria, S. H. M. Amin, and R. Marmat, "Design and Development of Control System for Internet-based Telerobotics," Proc. TENCON 2000, Vol.2, pp.338-342, 2000.

[7] 정종근, 천승환, "웹 카메라 기반 홈 서버용 모니터링 기술에 관한 연구," 한국해양정보통신학회 논문지, 제9권, 제8호, pp.1754-1761, 2005.

[8] P. Saucy and F. Mondada, "KhepOnTheWeb:

Open Access to a Mobile Robot on the Internet," IEEE Robotics & Automation Magazine, Vol.7, Issue1, pp.41-47, Mar., 2000.

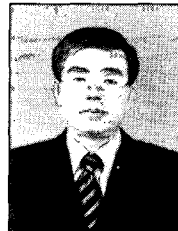
[9] 이승익, 장철수, 정승욱, 김중배, "로봇 소프트웨어 아키텍처의 연구 동향과 현황", 전자통신동향분석, 제20권, 제2호, 2005.

[10] 이태희, 임동선, 김주만, "URC 로봇 원격 모니터링 설계 및 구현", 한국콘텐츠학회 학술대회논문집, 제4권, 제1호, pp.136-139, 2006.

저자 소개

김 주 만(Joo-Man Kim)

정회원



- 1984년 : 숭실대 전자계산학(공학사)
- 1998년 : 충남대 컴퓨터공학(공학석사)
- 2003년 : 충남대 컴퓨터공학(공학박사)

- 1985년 1월~2000년 2월 : ETRI 책임연구원(운영체제연구팀장)
- 1995년 7월~1996년 6월 : Novell Inc. Research Center 방문 연구원
- 2000년 3월~2006년 2월 : 밀양대학교 정보통신공학부 교수
- 2006년 3월~현재 : 부산대학교 바이오정보전자 교수 <관심분야> : 실시간 임베디드 시스템, 저전력 OS, 바이오 센서 제어, 유비쿼터스 컴퓨팅