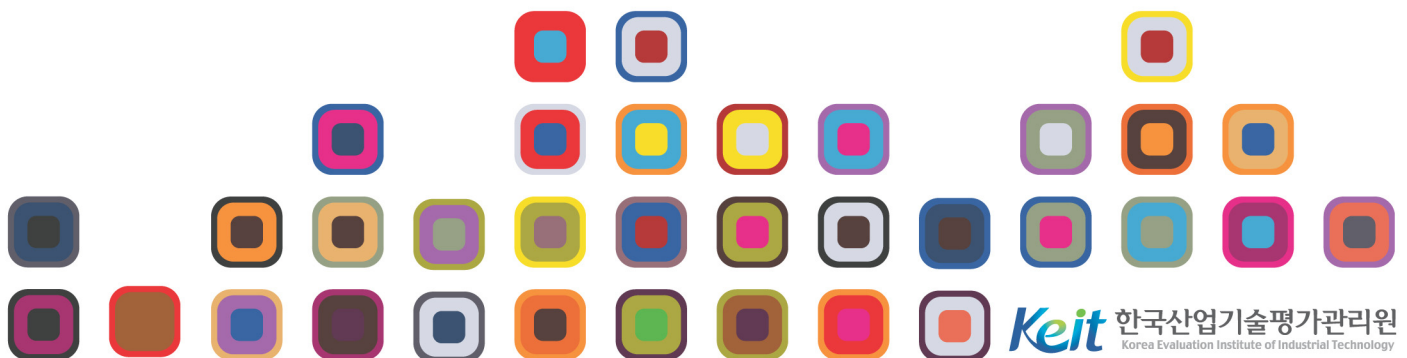


이슈 3

클라우드 기반 로봇틱스의 현황과 발전방향

이상무 로봇 PD



Issue Report

로봇
(2011.11)

클라우드 기반 로봇틱스의 현황과 발전방향

주저자 : 이상무 KEIT 로봇PD

부저자 : 김홍석 KITECH 수석연구원

요 약

□ 개요

- 로봇은 컴퓨팅 파워를 작게하여 간단하고 쉽게 제작하고, 복잡한 알고리즘 수행, 대용량의 데이터 처리 등은 인터넷 상의 서버를 활용하여 서비스를 제공하는 방식

□ 필요성

- 제한적인 로봇 컴퓨팅 파워를 극복하고, 로봇이 가지고 있는 HW 자원을 기반으로 동작 환경에 맞는 앱과 컴포넌트를 지원하여 동작환경과 로봇의 자원을 최적으로 활용할 수 있는 서비스 제공

□ 국내외 기술현황

- 국내에서는 정부 과제를 통하여 제한적이거나 부분적인 개념만을 도입한 사업이 진행되었으나, 본격적인 클라우드 기반 로봇틱스(이하, 클라우드 로봇)에 대한 연구는 시작되지 않은 상황
- 해외에서는 로봇의 운용에 클라우드 서비스를 적용한 몇 가지 예가 있으나, 제한적인 기능만을 수행하고 본격적인 클라우드 기반 로봇서비스에 대해서는 연구 중

□ 기대효과

- 클라우드 로봇의 개념 정립 및 인프라 구축으로, 이동·조작·HRI 등 로봇원천 기술 전반을 포괄하는 로봇 앱 확산 및 로봇산업 생태계 조성을 통해, 로봇서비스 창출의 활성화 및 세계시장 선도

1. 개념

□ 정의

- 도쿄대 이나바 교수가 “Remote-Brained Robots” 개념을 발표한 것이 최초의 클라우드 로봇 아이디어로 볼 수 있음
 - ※ 센서·모터 등 간단한 하드웨어로 구성된 로봇이 원격에서 제공되는 소프트웨어를 통해 구동됨
- (기본정의) 로봇에 사용되는 각종 데이터, 정보, 응용 소프트웨어(앱 혹은 콘텐츠) 등을 클라우드를 통하여 빠르게 처리 혹은 수행하여 결과를 제공해주는 서비스 기술
 - 로봇이 갖고 있지 않은 운용환경에 대한 지식체계와 정보를 다수의 서버를 이용한 분산처리기술을 통해 빠르게 검색·처리할 수 있고, 지식데이터 베이스로 공유할 수 있는 서비스
 - 대용량 컴퓨팅 파워를 요구하는 로봇 작업을 다수의 서버를 이용한 분산처리기술을 통해 빠르게 처리하여 결과를 얻을 수 있는 서비스

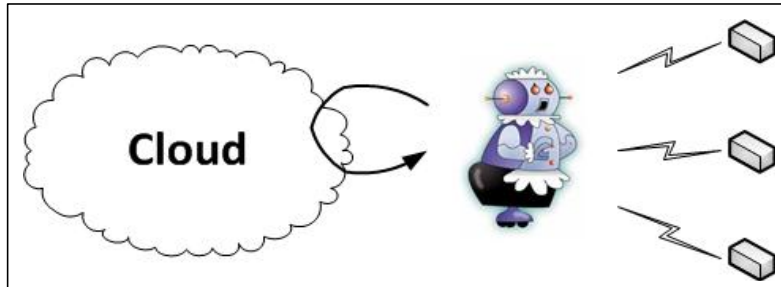


[그림 1] 클라우드 기반 로봇틱스 개념도

- (확장정의) 클라우드 서비스를 통해, 로봇을 중심으로 IT 기기, 센서, 인간이 같이 연동하여 변화하는 환경에 대응하여 목표로하는 서비스를 원활히 수행하도록 하는 기술
 - ※ 구동 환경에서 달라진 연동 구조(Configuration 등)에 대한 데이터와 그에 따른 어플리케이션을 인터넷 서버로부터 제공받아 사용
 - ※ 로봇이 동작하는 세계에 대한 지식체계와 정보를 지식 데이터베이스로 공유, 복잡한 상황에서 작업을 수행하는데 필요한 로봇의 인식 및 행동 라이브러리를 재사용 가능한 인터페이스로 활용

□ 유형

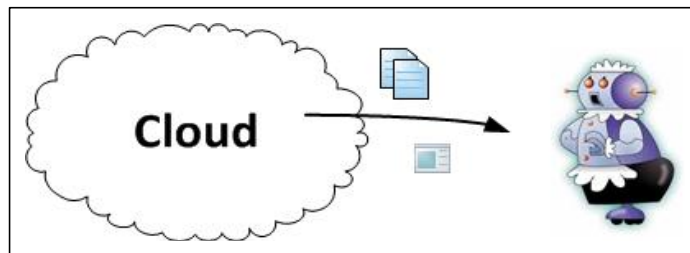
- **(유형#1)** 클라우드 컴퓨팅 파워를 활용하는 유형으로, 로봇의 컴퓨팅 능력에 한계가 있을 경우, 클라우드 서비스의 컴퓨팅 파워를 이용하여 문제 해결(예: 물체인식, 음성인식 등)



[그림 2] 유형#1 데이터 흐름

- **(유형#2)** 클라우드로부터 로봇의 제어프로그램, 컴포넌트, 콘텐츠를 다운로드하여 서비스 제공

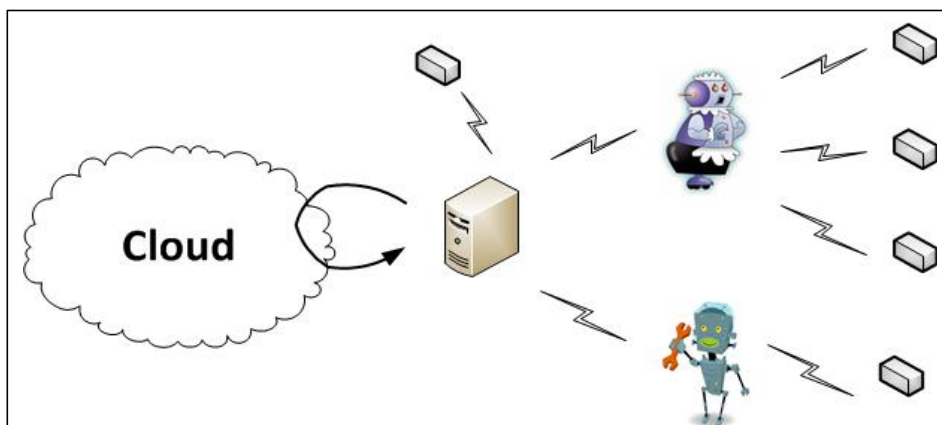
※ 관련기술 : Reusable Components and Applications



[그림 3] 유형#2 데이터 흐름

- **(유형#3: Semi-Cloud/Server)** 동작 환경 내의 로봇, IT기기 등의 리소스를 자동으로 파악하여 해당되는 서비스 제공, 해당 리소스의 스펙이 바뀌면 자동으로 서비스 재구성

※ 관련기술 : Re-Configuration/Zero-Configuration, 운용환경의 변동 파악, 보안 이슈



[그림 4] 유형#3 데이터 흐름

□ 기능

o 클라우드 컴퓨팅의 SaaS와 IaaS에 대응하는 클라우드 로봇의 기능은 다음과 같음

※ SaaS : Software as a Service, IaaS : Infrastructure as a Service

- 로봇은 클라우드로부터 서비스를 받을 수 있어야 한다
- 로봇은 클라우드에 정보를 저장할 수 있어야 한다

[표 1] 클라우드 로봇틱스 vs 클라우드 컴퓨팅

구분	클라우드 기반 로봇틱스	클라우드 컴퓨팅
클라우드의 제공서비스 (클라우드 → 로봇/PC)	<ul style="list-style-type: none"> ▪ 로봇 응용 컴포넌트 ▪ 응용 컴포넌트의 수행 결과 ▪ 데이터(실내외 지도) ▪ 주행 경로(계획) ▪ 환경(영상, 객체) 인식 결과 ▪ 음성인식 및 TTS 서비스 	<p>SaaS</p> <ul style="list-style-type: none"> ▪ 응용 소프트웨어(문서도구 등) ▪ 문서 제공/동기화 ▪ 지도 서비스 ▪ 영상(객체)인식 ▪ 음성인식서비스
클라우드로의 전송 정보 (로봇/PC → 클라우드)	<ul style="list-style-type: none"> ▪ 경로 데이터 ▪ 로봇위치 및 주변 객체 데이터 ▪ 환경 정보 ▪ 응용 컴포넌트 입력 데이터 ▪ 응용 컴포넌트 성능 지표 	<p>IaaS</p> <ul style="list-style-type: none"> ▪ 문서 저장 ▪ 음악/영상 정보

o 클라우드 컴퓨팅이 언제 어디서든 원하는 서비스를 이용할 수 있다는 장점이 있지만, 항상 인터넷 접속을 전제로 하는 점은 제약

o 이러한 한계의 보완을 위해 다중접속 서비스(인터넷 접속경로 다양화), 로컬시스템 연동(오프라인 상태에서의 구동) 등의 대안 마련 중

※ 로봇의 경우에도, 접속 불가능 지역에서의 동작을 위한 기본 기능을 내장하고, 연결 재개 후 동기화되는 로컬 시스템 연동기능 필요

o 클라우드 컴퓨팅의 PaaS에 해당하는 'R-PaaS' 구축도 가능



[그림 5] R-PaaS 개념도

※ R-PaaS(Robot Platform as a Service) : 로봇의 설계에서 알고리즘·서비스 개발 및 검증(시뮬레이션 및 참조 로봇 기반)에 이르는 전 과정을 클라우드에서 제공하는 서비스

2. 출현배경 및 필요성

□ 출현배경

- 로봇서비스 창출을 위한 플랫폼으로서 클라우드 컴퓨팅 활용
 - 한 기관이 모듈, 시스템, 운영 소프트웨어, 어플리케이션, 콘텐츠, 서비스까지 모든 것을 기획·구축하기 때문에 기술적 전문성과 소비자 욕구에 대응하는 다양성이 부족하고, 로봇산업의 에코시스템이 형성되지 못하였음
 - 다양한 로봇서비스의 창출을 위해서는, 클라우드 서비스를 개발자와 사용자가 모두 참여하는 플랫폼으로 활용하는 것이 바람직함
- 개방된 시스템 도입 및 다수의 개발자를 동시에 활용, 사용자로부터의 피드백이 가능한 전략 필요
 - 클라우드 로봇은 새로운 시스템을 구축하는 것이 아니라, 현재의 거대한 클라우드 환경에 로봇을 하나의 I/O로 추가하는 것임
 - 개발자 입장에서는 로봇은 최종 서비스를 구현하기 위한 하나의 도구일 뿐, 실제로 서비스를 구현하기 위해서는 다양한 정보 및 데이터가 필요함
 - 온라인을 통한 다수 개발자 참여 유도, 불특정 다수의 개발자에 의한 정보 및 알고리즘 구축, 커뮤니티 생성 등을 통해 더욱 완전한 서비스 개발이 가능함

□ 필요성

- 로봇에 탑재 가능한 컴퓨팅 파워의 한계 및 운용환경과 로봇의 자원을 최적으로 활용할 수 있는 서비스 필요성 증대
 - 로봇의 운용환경이 달라지므로 추가적인 컴포넌트와 컴퓨팅 파워가 필요함
 - ※ 현재 가지고 있는 컴포넌트보다 더 다양한 응용 서비스에서 사용할 수 있는 완성도 높은 컴포넌트 라이브러리가 필요(HRI, 자율주행 등)
 - ※ 로봇 HW자원, 운용환경, 주변의 다른 로봇·IT기기 및 센서 등을 최적으로 활용할 수 있는 적합한 앱과 콘텐츠를 지원하여 서비스 제공
- 로봇 소프트웨어 마켓 활성화를 위해서는 로봇 SW 프레임워크와 응용 프로그램 인터페이스의 표준화가 선행되어야 함
 - ※ 예로서, 로봇 SW 프레임워크 표준에는 OProS(Open Platform for Robotics Services, 한국), ROS(Robot Operation System, 미국) 및 OpenRTM-aist(Open Robot Technology Middleware, implemented by AIST, 일본) 등이 있고, 응용프로그램 인터페이스의 표준에는 RoIS(Robot Interaction Service)와 RLS(Robot Localization Service) 등이 있음

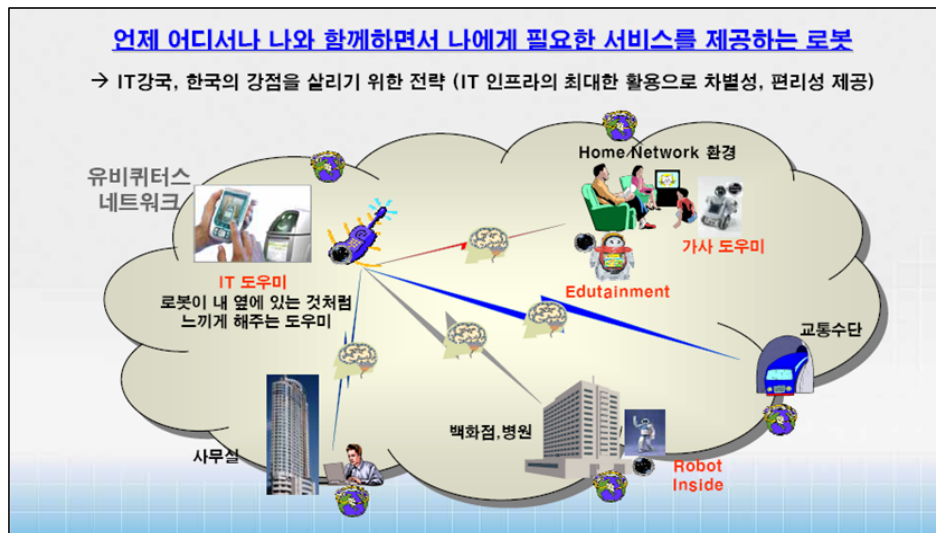
3. 국내외 기술개발 현황

□ 국내 개발 현황

o URC(Ubiquitous Robotic Companion*)

※ 언제 어디서나 나와 함께 하며 나에게 필요한 서비스를 제공하는 로봇을 의미

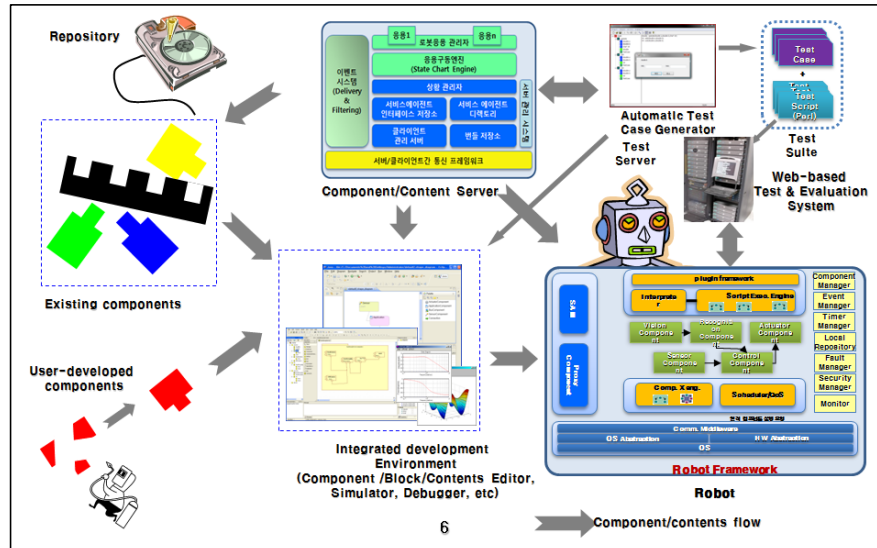
- 기존의 로봇 개념에 네트워크를 활용하는 URC 과제를 추진하면서 클라우드 로봇의 개념을 일부 사용
 - 유사점 : 서버를 통하여 센서와 IT 기기와 로봇들이 연동되도록 제어되고 서버의 컴퓨팅 능력을 빌려서 사용
 - 차이점 : 컴포넌트 혹은 앱을 다운로드 받아서 수행하거나 로봇의 데이터를 저장하는 기능 등은 없음. 즉, 제한적인 클라우드 로봇 개념을 구현함



[그림 6] URC 개념

o OPRoS(Open Platform for Robotic Services)

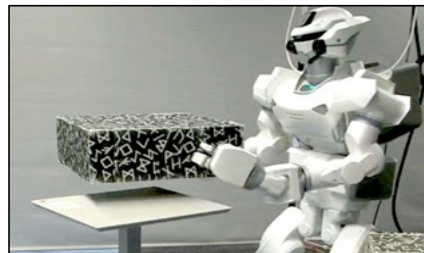
- 컴포넌트를 구성, 편집, 다운로드하여 실시간으로 실행시키는 일련의 로봇 소프트웨어 개발 스위트(Suite) 개발을 목표로 함
 - 서버의 컴퓨팅 인프라를 활용하고, 데이터와 정보의 저장 및 재사용을 고려하지 않았다는 점에서, 클라우드 로봇의 개념을 부분적으로 활용
 - 온라인 컴포넌트 스토어에서 컴포넌트를 다운로드하거나 컴퓨팅 파워를 사용하여, 컴포넌트는 웹을 통하여 검증 받도록 되어 있음



[그림 7] OPRoS 개념

□ 해외 개발 현황

- (일본) 2001년부터 AIST의 휴머노이드 연구그룹에서, 환경 내의 물체인식 및 행동 선택 등에 대한 인공지능 구현을 위해 클라우드 서비스 개념을 도입



[그림 8] AIST의 HRP-3

- (미국) 2010년 구글은 무인 차량의 주행 제어를 위해 구글의 클라우드 서비스인 구글맵을 활용함



[그림 9] 구글의 무인차량 'Pribo'

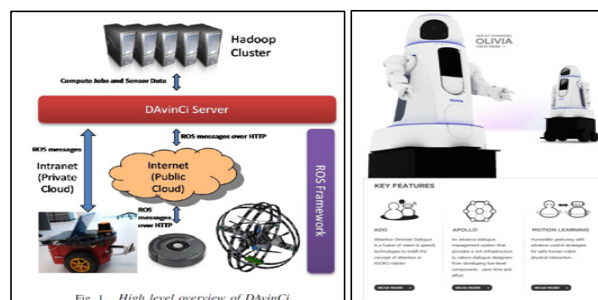
※ '2007 Urban Challenge'를 우승한 CMU의 Chris Urmson, '2005 Grand Challenge'를 우승한 스탠포드의 Mike Montemerlo, 역시 'DARPA Grand Challenge'에서 세계 최초로 자율주행 모터사이클을 만들어 참가했던 Anthony Levandowski의 협업으로, 프리우스를 개조하여 무인차량(Pribo)을 제작

- o 2011년 발표된 미국정부의 NRI는 개방형 로봇 구조 및 공통 플랫폼 구축, 연구자 및 사용자 참여에 의한 R&D, 클라우드 로봇틱스를 위한 사이버 인프라스트럭처 구축 등, 교육, 상용화 전반의 지원을 명시함
 - NRI 발표로 인하여, 기업들의 로봇 비즈니스 전략이 정비되고 새로운 에코시스템이 정착될 것으로 전망됨
- o 2011년 Willow Garage사는 자사의 로봇(PR2, 터틀봇)과 ROS에 구글서비스(사진을 통한 물체인식, 음성인식 등)를 적용한 로봇서비스를 개발 중이며, 현재 ROS-안드로이드 연동 구현중임



[그림 10] 구글의 ROS-Android 연동

- o (EU) FP7의 일환으로 로봇협업과제 ‘RoboEarth’ 수행 중
 - 로봇들이 정보를 공유하고 행동과 환경에 대하여 서로 배우는 거대한 네트워크이며 DB 리포지토리(Repository), 즉, 로봇용 클라우드를 구축하여 관련 정보를 찾아 배우게 하는 시스템
 - ROS를 기반으로 개발하지만, HW 추상화계층을 도입하여 컴포넌트개발, 이 컴포넌트가 관련 DB에 접속하여 원하는 정보를 얻는 구조
- o (싱가포르) 2010년 싱가포르의 Data Storage Institute A*STAR에서 로봇의 클라우드 컴퓨팅을 위한 프레임워크인 DAVinCi를 개발하고 로봇의 Fast SLAM 기능을 클라우드에서 구현함



[그림 11] DAVinCi의 구조

4. 활용방안 및 기대효과

□ 활용방안

- 음성인식 기능이 없는 로봇 플랫폼이 음성인식 클라우드 서비스를 통해 사용자 명령을 인식하고, 카메라를 통한 영상을 통해 동작 환경에 대한 인식 및 작업 판단을 할 수 있음
 - 음성인식, 영상인식 등 고성능의 컴퓨팅 파워가 요구되는 처리를 클라우드 서비스를 통해 해결 가능
- 광범위한 환경에서 각기 다른 로봇플랫폼이 환경에 대한 정보를 수집하고, 수집된 정보를 클라우드 서버에 저장, 다른 로봇 플랫폼과 공유 가능
 - 환경 정보 및 로봇 플랫폼 등의 리소스가 바뀌면 자동으로 재구성하고 실시간으로 변동 파악 가능
- 로봇 앱 스토어를 통한 제어프로그램, 컴포넌트 및 콘텐츠를 다운로드하여 해당 환경에 맞는 서비스 제공
 - 해당 환경 내에서 구동되는 로봇 플랫폼의 모니터링, 제어 가능
 - 로봇이 수행해야 하는 서비스에 대한 최소의 적합성을 보유하면, 로봇 기능 확장 가능

□ 기대효과

- 기술적 효과
 - 사용자 적합 에코시스템 전략 및 글로벌 표준화 추진 가능
 - 국내 및 국제 개발자 커뮤니티 조성을 통한 로봇 개발자 역량 증대
 - 글로벌 표준화를 통한 이종 클라우드 간의 연동 및 활용, 이를 이용한 다양하고 풍부한 서비스 창출 가능
- 경제·산업적 효과
 - 클라우드 컴퓨팅 기술 도입으로, IT업계 참여 유도 및 기타 연관 서비스 창출 가능
 - 클라우드 컴퓨팅 기술 보안 관련 업체, 클라우드 서비스 제공 업체 등 IT관련 업체 참여 유도
 - 로봇에 한정된 기술뿐만이 아닌, 일반적인 SW알고리즘, 기술개발 결과를 로봇에 쉽게 적용 가능

5. 정책 제언

□ 현 상황에서의 문제점 및 극복방안

	현 상황에서의 문제점	극복방안
표준화된 체계 측면	<ul style="list-style-type: none"> • 로봇 앱의 개념 불명확 • 공통 인터페이스 부재 • 로봇과의 통신표준 부재 	<ul style="list-style-type: none"> • 이종 로봇을 비롯한 다양한 기기를 포함하는 클라우드 로봇틱스 환경에 대한 표준화 * 앱, 컴포넌트, 공통 인터페이스, 통신 등에 대한 표준화를 우선 추진(일부 추진 중)
초기성과 확산 측면	<ul style="list-style-type: none"> • 다양한 운영체제 사용 • 모듈화된 SW개념이 초기 단계 • 이동·조작 등 복잡성을 유발하는 요소를 포함 	<ul style="list-style-type: none"> • 초기에는 로봇에서 많이 사용되는 운영체제를 중심으로 모듈화가 쉬운 인간-로봇 상호작용(HRI) 기술을 활용한 앱 개발에 주력 * 클라우드 로봇의 앱 개념은 서비스 로봇에 가장 적합. 현재 상용화 가능성이 높은 서비스 로봇은 이동·조작보다는 HRI 기능이 중심

□ 추진 방안

o 표준화된 체계 및 초기 성과 확산 측면에서 2단계로 사업 추진 필요

- 1단계에서는 클라우드 로봇틱스 체계 확립 및 인프라 구축, HRI 기술을 활용한 앱 개발 추진을 통한 초기 성과 달성을 추진
- 2단계에서는 클라우드 로봇틱스 체계의 확산·보급 및 이동·조작·HRI 전반을 포괄하는 앱 확산전략으로 세계 로봇 분야 표준 및 시장 선도

	1단계	2단계
표준화된 체계 측면	<ul style="list-style-type: none"> • 클라우드 로봇 체계 개발 및 인프라 구축(OPRoS연계) * OPRoS는 ROS와 RT-middleware의 Gateway 기능을 우선적으로 구현 • 클라우드 로봇 체계 표준화 * KS 및 국제단체표준 추진 	<ul style="list-style-type: none"> • 클라우드 로봇 체계의 글로벌화 • 클라우드 로봇 산업의 표준화 * ISO 등 파급력이 큰 표준 추진
초기성과 확산 측면	<ul style="list-style-type: none"> • 기 개발된 HRI 기술의 컴포넌트화 • 클라우드 소싱을 통한 정보·지식 공유 • 다양한 IT 기기를 포함하는 클라우드 환경을 최적으로 활용하는 HRI 핵심원천 기술 및 앱 개발 	<ul style="list-style-type: none"> • 이동·조작·HRI 등이 유기적으로 연계된 앱 개발 및 확산 • 이종 로봇, 이종 플랫폼 지원

6. 결론

- o '클라우드 기반 로봇틱스'는 클라우드 컴퓨팅 기술을 로봇에 적용한 것임
- o 컴퓨팅 분야와 로봇 분야는 성숙도와 그 적용 분야가 다르므로, 클라우드 기반 로봇틱스는 적용 시 고려할 사항이 많음
 - 성숙도의 차이
 - 컴퓨팅 분야는 ASP(Application Service Provider)와 컴포넌트 개념이 발달하고, 앱 스토어의 활성화에서 보듯이 모듈화된 SW 개념이 성숙되어 있음. 그러나 로봇은 앱이라는 개념도 명확하지 않고, 앱의 공통 요소도 없으며 모듈화된 SW 개념도 초기단계임
 - 또한 컴퓨팅 분야는 PC 및 스마트 폰에서는 웹이라는 공통적인 인터페이스가 존재함. 또한 PC는 크게 윈도우와 리눅스로, 스마트 폰은 안드로이드와 iOS로 나누어지므로 앱이 단순하나, 로봇 분야는 공통적 인터페이스, 로봇 간 통신 표준이 없고 다양한 운영체제(윈도우, 리눅스, RTOS, 안드로이드 등)를 사용하고 있음
 - 적용분야의 차이
 - 컴퓨팅 분야는 연산과 데이터 조작 등의 CPU 및 메모리 사용 관점이지만, 로봇 분야는 연산과 데이터 조작 뿐만 아니라 이동 및 조작 등 실세계와의 입출력이 포함되어 있어 훨씬 복잡함
 - 클라우드 기반 로봇틱스는 현재 초보 단계로 누가 먼저 사용자에게 호소력 있는 서비스를 제공하느냐에 따라 사실상의 표준을 선도할 수 있음. 특히 현재 안드로이드 진영은 특허 문제로 유료화 가능성이 있으므로 상용화를 포함한 확산이 지연될 가능성이 있음
 - 따라서 OPRoS를 기반으로 한 클라우드 기반 로봇틱스 관련 서비스를 빨리 제공하는 것도 필요할 것으로 판단됨

[참고문헌]

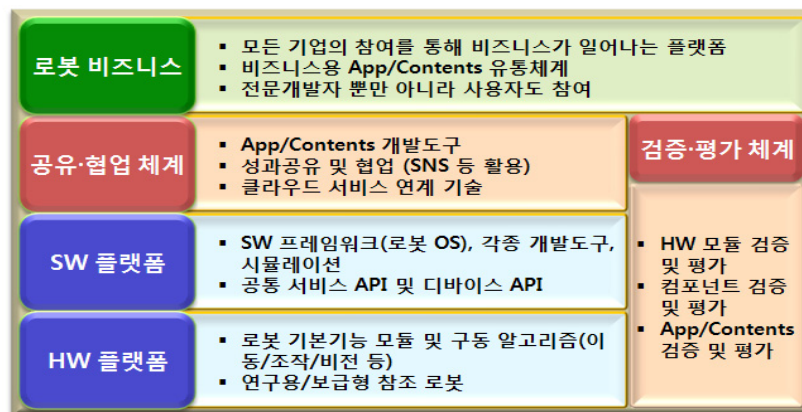
1. Masayuki Inaba, <http://www.jsk.t.u-tokyo.ac.jp/~indaba/>
2. "Remote-Brained Robotics: Interfacing AI with Real World Behaviors", In Robotics Research: The Sixth International Symposium, pp.335-344. International Foundation for Robotics Research, 1993
3. <http://roboticpeple.com/archives/443>
4. "클라우드 기반 로봇원천기술 발전전략위원회" 4차 회의, 2011.8.31
5. <http://www.opros.or.kr>
6. <http://www.ros.org/wiki/>
7. <http://openrtm.org/>
8. http://www.aist.go.jp/aist_e/latest_research2010/20100210/20100210.html
<http://unit.aist.go.jp/is/humanoid/index.html>
9. <http://googleblog.blogspot.com/2010/10/what-were-driving-at.html>
10. <http://www.getrobo.com/getrobo/2010/11/autonomous-car-masterminds-converge-at-google-1.html>
11. <http://www.cmu.edu>
12. <http://www.asoro.a-star.edu.sg/>
13. <http://www.nsf.gov/pubs/2011/nsf11553/nsf11553.htm>
14. <http://www.willowgarage.com>
15. http://cordis.europa.eu/fp7/ict/cognition/home_en.html
16. <http://www.roboearth.org>

[국내외 주요 기술개발 현황]

연구기관명	프로젝트명	개요	연구기간
Willow Garage & Google (미국)	o Cloud Robotics	<ul style="list-style-type: none"> o ROS on Android (rojava library) <ul style="list-style-type: none"> - ROS 프레임워크 내에서 Java 실행 가능하도록 지원 o Open Accessory and Android @ Home APIs <ul style="list-style-type: none"> - 안드로이드 기기와 센서 및 액추에이터 등과의 연동을 위한 API - 자사의 로봇(PR2, 터틀봇)과 ROS에 구글서비스(사진을 통한 물체인식, 음성인식 등)를 적용한 로봇서비스를 개발 중이며, 현재 ROS-안드로이드 연동 구현 중 	Google I/O 2011에서 발표 (2011.5.11)
Data Storage Institute, A*STAR (싱가포르)	o DAVinci project (Cloud computing framework for service robots)	<ul style="list-style-type: none"> o Distributed ROS 아키텍처, open-source Hadoop Distributed File System (HDFS), Hadoop Map/Reduce Framework의 조합 <ul style="list-style-type: none"> - 로봇의 클라우드 컴퓨팅을 위한 프레임워크인 DAVinci를 개발하고 로봇의 Fast SLAM 기능을 클라우드로 구현 	2010.5.3-8 ICRA 2010 학회논문 발표 (연구기간 불명)
Carnegie Mellon University & Google (미국)	o Developing the driverless car project	<ul style="list-style-type: none"> o Self-driving cars that came with the DARPA Grand Challenge <ul style="list-style-type: none"> - 운행 중 발생/수집되는 대용량의 데이터를 구글의 데이터센터에 이용하여 분산처리 	2009-2010
Eindhoven University of Technology 외 EU 내 6개 기관	o RoboEarth	<ul style="list-style-type: none"> o 오픈소스 네트워크 데이터베이스 플랫폼 <ul style="list-style-type: none"> - 로봇들이 정보를 공유하고 행동과 환경에 대하여 서로 배우는 거대한 네트워크이며 DB 리포지토리 (Repository). 즉, 로봇용 클라우드를 구축하여 관련 정보를 찾아 배우게 하는 시스템 - ROS를 기반으로 개발하지만, HW 추상화계층을 도입하여 컴포넌트 개발, 이 컴포넌트가 관련 DB에 접속하여 원하는 정보를 얻는 구조 	2009-2013

개방형 로봇플랫폼 전략(Open Robot Platform Strategy) (案)

- (정의) 로봇플랫폼은 HW·SW 뿐만 아니라, R&D와 서비스창출의 상호작용을 지원하기 위한 체계 등을 포함
 - (HW) 로봇기능의 구현·검증을 위한 참조로봇과 기능모듈
 - * 고급형(연구용), 보급형 참조로봇(Reference Robot)
 - * 로봇을 구성하는 기본기능(이동/조작/비전) 모듈 및 구동 알고리즘
 - (SW) 로봇을 구동하고 운용하기 위한 제반 소프트웨어 요소
 - * 소프트웨어 프레임워크, 개발환경, 계층간/로봇간 접속을 위한 표준 API
 - * 장치기능모듈을 구동하기 위한 드라이버, 로봇의 기본기능을 구현하기 위한 컴포넌트라이브러리
 - (체계) R&D협업, 성과공유, 서비스창출을 활성화하기 위한 공유·협업체계 및 플랫폼 평가·검증 시스템
 - * 클라우드 컴퓨팅, SNS 등 최근의 기술과 트렌트를 활용
 - * HW모듈, SW컴포넌트, App/Contents 검증평가 시스템



- (중요성) 플랫폼은 산업의 에코시스템이 형성되기 위한 기반
 - 플랫폼이란, 응용프로그램이 구동되는 기반(HW구조·SW틀·OS) 등을 의미
 - * 플랫폼은 원래, 승강장(교통수단), 무대(공연), 시합대(스포츠), 프레임(자동차), 기초소재 풀(부품소재) 등 가시적 기반시설/구조를 의미
 - 초대형 글로벌 IT기업들은, 플랫폼 구축을 통해 거대 에코시스템을 형성하는 '플랫폼 전략'을 구사
 - * 구글(클라우드 서비스), 애플(디지털기기), 페이스북(SNS) 등이 현재 대표적인 플랫폼전략 기업
 - 로봇산업도 HW, SW, 콘텐츠, 개발 및 유통 네트워크 등을 아우르는 에코시스템(Ecosystem)을 위한 플랫폼에 주목할 시점
 - * HW(2000년대 초)와 SW('06년 이후) 관점의 플랫폼의 필요성을 인식했으나, 단편적인 기술개발에 그침