

## 스마트폰 제어 및 영상처리를 수행하는 바퀴와 4족을 결합한 약병 전송 로봇

이상영\* · 김현수\* · 김영룡\* · 홍석호\* · 김동환\*†

\* 서울과학기술대학교 기계시스템디자인공학과

### Drug Bottle Delivery Robot Capable of Smartphone-Based Control and Image Process and Combining Wheel and Quadruped

Sang Young Lee\*, Hyun Su Kim\*, Young Long Kim\*, Seok Ho Hong\*  
and Dong Hwan Kim\*†

\*Dept. of Mechanical System Design Engineering, Seoul Nat'l Univ. of Science and Technology, Seoul, Korea

(Received July 10, 2012 ; Revised December 29, 2012 ; Accepted January 7, 2013)

**Key Words:** Smart Phone Image Processing(스마트 폰 영상처리), Quadruped and Wheel Combined Robot(4족 및 바퀴 통합로봇), MAC-OS Based Programming (매킨토시 기반 프로그래밍)

**초록:** 이 연구에서는 스마트폰과 Wi-Fi통신을 이용한 로봇의 조종과 장착된 카메라를 통한 영상처리 기술에 대하여 서술하였다. 제안된 로봇은 바퀴와 4족을 환경에 따라서 선택적으로 사용할 수 있도록 메커니즘을 구성하였다. 카메라의 스트림 데이터 중 이미지 데이터만을 이용하도록 네트워크를 형성하였으며 영상처리 기법을 응용하여 약병을 구분하고, 로봇 팔을 이용하여 약병을 사용자에게 전달해주는 로봇 메커니즘에 대해서도 서술한다. 본 논문에서 개발된 영상처리 알고리즘과 처리는 별도의 컴퓨터 없이 스마트폰만을 이용하여 구현이 가능하도록 설계하였으며 스마트폰의 강력한 기능과 연산능력을 최대한 활용하여 약병 로봇의 지능화 및 소형화 방안을 제시하였다.

**Abstract:** Robot control and image processing using a smartphone and Wi-Fi communication is introduced. The robot has a wheel and quadruped mechanism that is transformed according to the environment and is mainly used for drug bottle delivery. The captured image on the camera is transmitted to the smartphone in the form of stream data, and the image data is processed in the smartphone to enable the robot to identify an object and to control the robot itself. A network was constructed so that only image data from the stream data was used, and an image processing scheme to identify the drug bottle and deliver it to a person using a robot arm is also presented. In this study, image processing techniques and algorithms were purely implemented on a smartphone with considerable computational power and multiple functions rather than a computer, which contributes to the intelligence and miniaturization of the robot system.

### 1. 서 론

본 연구에서는 집안에 거주하는 독거노인을 대상으로 하였다. 핵가족화의 증가와 비례하여 독거노인의 비율은 날이 갈수록 증가하고 있으며 동물로봇의 개발은 반려동물을 대체하여 외로움을 느끼는 노인들을 보조하는데 그 목적을 두고 있다. 기존 감성로봇의 연구로는 일본의 노인용 로봇

바다표범 파로, 미국의 자폐치료용 테디베어 로봇, 벨기에의 외로운 어린이용 로봇 프로보 등이 있다.<sup>(1)</sup> 본문에서 다룰 로봇은 기존의 동물로봇과 같이 노인들의 외로움을 달래는 목적을 두고 있을 뿐만 아니라 약병을 노인에게 시간에 맞추어 전달하는 기능을 부여함으로써 기존의 동물로봇 개발과 차별을 두고 있다.

본 연구에서 중점을 두고 있는 부분은 스마트폰에 의한 로봇 구동이다. 기존의 로봇시스템에서는 컴퓨터를 주제어기로 사용하여 로봇에 내장하거나 외부에 컴퓨터를 두고 신호를

† Corresponding Author, [dhkim@seoultech.ac.kr](mailto:dhkim@seoultech.ac.kr)  
© 2013 The Korean Society of Mechanical Engineers

무선으로 주고받는 원격제어 방식의 로봇이 일반적인 로봇시스템 개발의 핵심이었다.

이 경우 컴퓨터를 활용해야 하기에 로봇이 비대해지거나 원격제어를 할 경우 많은 데이터를 무선으로 보내기 때문에 로봇의 실시간 제어에 한계가 발생하였다. 이러한 컴퓨터를 주제어기로 사용하여 발생한 한계는 스마트폰을 통해 해결이 가능해졌다.

임베디드 (Embedded) 컴퓨터가 일정부분 대안이 되지만 컴퓨터만큼의 연산능력과 빠른 처리시간에 제한이 따르게 된다. 스마트폰은 날로 그 기능이 확장되고 속도도 빨라지므로 기존 로봇의 제어기를 대체할 가능성이 더 높아지고 있다<sup>(2)</sup>.

스마트폰의 또 다른 장점은 사용하기에 기존의 컴퓨터보다 훨씬 편하고 간단하다는 점이다. 로봇을 스마트폰을 이용하여 제어할 경우 편리성과 다양성에서 컴퓨터 제어의 로봇보다는 그 유용성이 더 확대 될 것이다. 특히 노령 자를 대상으로 하는 로봇의 경우 사용의 편리성은 컴퓨터제어 방식의 로봇과는 비교가 될 수 없을 것이다. 기존의 스마트폰을 로봇에 적용한 사례가 발표되었으니 대부분 어플리케이션 프로그램에 의한 간단한 동작이나 영상 전송을 한 것이 대부분이다.<sup>(3-5)</sup> 본 연구에서는 로봇에 스마트폰을 주제어기로 사용하고 컴퓨터를 배제하기에 영상처리나 제어기 설계 등이 스마트폰내에서 이루어 지도록 하여 보다 효율적이고 소형의 로봇 시스템 제작이 가능하게 하였다. 다양한 어플리케이션 프로그램을 스마트폰에 올려놓아 자유롭게 노약자나 고령자가 손쉽게 사용하도록 하는 것은 로봇의 저변확대에 새로운 장을 열게 된다.

우리 개발 로봇의 주 기능은 환자나 노약자에게 약병을 자율적으로 갖다 주는 것을 목표로 하였으며 이 로봇을 스마트폰으로 제어하거나 영상을 처리하여 정확한 약병을 갖다 주는 임무를 부여 받게 제작하였다. 방 안의 공간을 자유롭게 이동하기 위하여 바퀴형태의 구동뿐 아니라 4족으로 구동할 수 있게 구동 부를 설계하여 장애물 극복이나 로봇 구조의 변형을 가능하게 하여 장애물을 쉽게 극복하게 하였다.

느린 이동속도 문제를 바퀴와 4족을 동시에 사용할 수 있게 함으로써 빠른 속도와 장애물 극복의 두 가지 목적을 동시에 달성할 수 있게 설계하였다. 다른 기능으로는 독거노인 또는 부양가족의 부재나 따로 떨어져 사는 경우에

요양기관이나 가족이 상황을 지켜볼 수 있게 카메라를 탑재하여 어느 곳에서든 집안 상황을 지켜볼 수 있게 했다. 평소에는 자유롭게 로봇이 돌아다니다가 약을 먹거나 원하는 물체를 집을 때 조작능력이 떨어지는 사람이라도 간단한 작동법만 숙지한다면 조작을 하여 로봇을 이동시키고 물건을 집는 모션을 취하여 원하는 물건을 가져다 주는 역할을 한다.

약병 전송 로봇은 환자의 특수성을 고려하여 설계되어야 하며 조작이 쉽고 로봇 시스템의 크기가 상당히 작은 것이 필요하다. 따라서 스마트폰을 활용한 로봇의 제어나 영상처리 등은 매우 효과적인 로봇제어 시스템이라 할 수 있다. 또한 비교적 빠른 기동성과 장애물 회피를 위해서 로봇을 크기를 작게 하면서 바퀴와 4족 동작을 동시에 하는 변형형 메카니즘의 설계 역시 로봇의 소규모화에 유익하다.

## 2. 본 론

### 2.1 Hardware 구성

#### 2.1.1 기구 구성

이 로봇은 4족 보행과 바퀴 이동을 겸하고 동시에 물건을 잡을 수 있는 로봇 손, 물건을 수납할 수 있는 수납 부, 그리고 카메라를 장착하고 있다. Fig.1은 제작된 로봇의 외관이다.

우선 4족 보행 시 로봇은 총 4개의 다리를 가지고 있으며 다리 하나당 3 자유도를 가지고 있다. 두 개의 자유도는 로봇의 다리를 들어 올리는데 사용되고 몸 쪽에 위치한 또 하나의 자유도는 다리의 전후 운동을 담당한다. 로봇 앞에는 팔이 설치되어 있으며 팔 또한 다리와 마찬가지로 3자유도를 가지고 있으며 물체를 잡기에 적합한 구조로 설계되어 있다. 전체적인 형상은 로봇 손에 카메

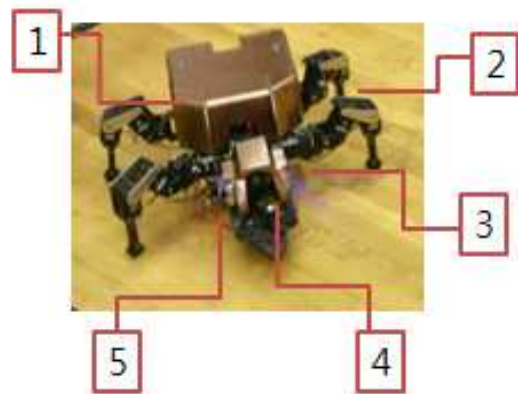


Fig. 1 Image of the drug delivery robot and major part

라와 기타 센서들이 Fig. 1에서 1번으로 표기된 것은 서보형 DC 모터로서 HR-350se란 제품으로 40g의 가벼운 무게를 가지고 있음에도 불구하고 저렴한 가격에 120rpm, 6kgf·cm의 높은 성능을 가지고 있다. 이 모터는 물건을 잡는 로봇 손을 오므리고 펴는 운동을 하는 부분과 물건 수납 부를 여닫는데 사용되고 있다. 2번은 RC 서보 모터로서 (AX-12A&AX-18A) 4발을 구동하는데 사용된다. 'ROBOTIS' 사의 바이오로이드 AX시리즈를 사용하고 있으며 15kgf·cm와 18kgf·cm의 토크 값을 가지는 총 두 가지 모델이 사용되고 있는데 이 모터들은 로봇의 다리와 로봇의 팔을 움직이는데 사용되고 있다. 3번은 엔코더형 DC 모터로서 '모터플러스사'에서 제작한 RE-11이라는 모델로 엔코더가 장착되어 있으며 2kgf·cm의 토크와 100rpm의 회전 속도를 가지고 있고 로봇의 바퀴를 구동시키는데 사용되고 있다. 4번은 로봇 전방에 부착된 IP 카메라이며 영상을 WiFi를 통하여 전송할 있게 한다. 5번은 초음파 센서로서 로봇이 이동 시 전방의 장애물 여부를 감지하는데 사용된다.

로봇 팔 끝 단에 있는 손은 내부에 랙과 피니언을 사용해서 선형 운동을 하며 물건을 잡는 모션을 취한다. 이는 구조가 복잡해 보이지 않으며 다양한 크기의 물건을 잡기 위함이다. 물건을 붙잡았을 경우 손에 달려 있는 압력센서로부터 신호를 받아 그 물체 파지여부를 판단한다. 프레임은 아크릴 밀도의 1/2을 가지고 있는 폼팩스(Fomax)라는 재질을 사용하였으며 이는 로봇 손의 무게를 최소화 하기 위함이다.

로봇의 등에는 물건을 수납하기 위한 수납공간이 있는데 구조는 물건을 담는 케이스와 내부에서 물건을 고정 시키는 내부 그립으로 구성되는데 이는 모터 하나를 작동시키면 연동 되어 움직이도록 설계하였다.

2.1.2 제어부

Fig. 2는 로봇시스템의 구성도를 보여준다. 스마트폰과 로봇이 통신을 하기 위하여 로봇에 장착된 IP카메라와 Wi-Fi 모듈을 공유기(Router)를 통하여 연결되어 있어 상호간의 통신이 가능하다. 스마트폰에서 로봇의 구동 방식(Auto/Manual Mode)이나 여러 명령을 Router를 거쳐서 로봇에 장착된 Wi-Fi모듈을 통해서 받으면 DSP의 SCI통신을 이용하여 명령어가 처리되게 설계되었다.<sup>(6)</sup>

명령어는 Char형태의 프로토콜(Protocol)을 정하여 그에 따라 처리되며 각각의 명령어에 대한 모



Fig.2 Robot system structure

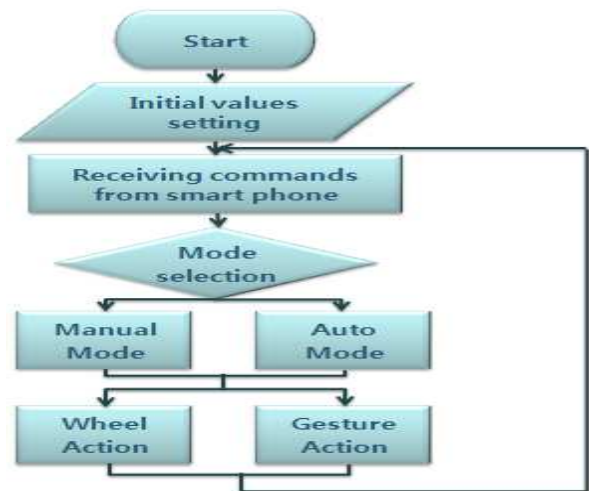


Fig.3 Flow chart of robot operation

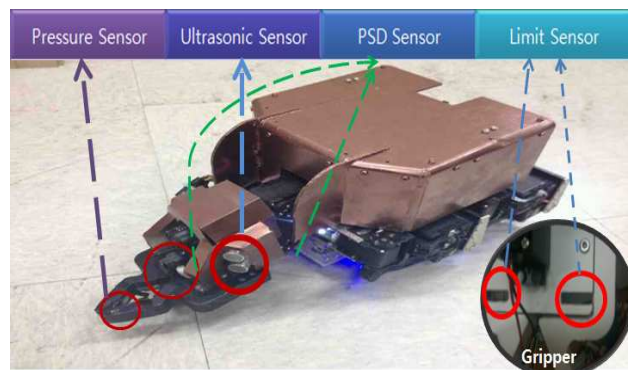


Fig.4 Series of sensors equipped to the robot

터제어, 각종 센서를 DSP2812를 이용하여 구현하였다.<sup>(7,8)</sup> Fig. 3은 로봇 작동이 이루어지는 프로세스를 보여주고 있다.

로봇의 전원을 키게 되면 초기값을 세팅한다. 그 후 Wi-Fi를 통해 스마트폰의 명령 값을 수신 고 스마트폰의 기본 모드가 선택된다. 모드는 로봇이 스마트폰에 의해 조작되어 사용자가 수동으로 로봇을 움직이는 모드와 로봇이 자율적으로 다닐

수 있는 자율주행 모드로 나뉜다. 수동 조작 모드에서는 스마트폰의 조작에 의하여 로봇으로 수신되는 명령어에 따라 기본적으로 바퀴주행하며 직진, 좌회전, 우회전, 후진동작, 물건을 수동으로 잡아줄 수 있는 그리퍼(Gripper) 조작, 바닥의 장애물을 극복할 수 있는 4족 주행모드가 있다. 자율주행 모드는 기본적으로 바퀴 주행을 하며 직진 모드로 작동된다. 자율 주행을 하며 구동되는 모드는 로봇에 달린 센서를 이용하여 거리나 장애물 여부를 판단하여 구동된다.

Fig. 4는 로봇에 사용된 센서로서 압력센서, 초음파센서, PSD센서, 리미트센서가 부착된 로봇의 모습을 보여주고 있다.

압력센서는 그리퍼 앞부분에 달려있어 약병이 잡혔는지 유무를 판단하기 위하여 사용되었다. PSD센서는 로봇의 바닥부분과 그리퍼 윗부분에 각각 달려있다. 바닥에 달린 PSD센서는 방문 턱을 만났을 때 로봇의 형태를 4족 구동 방식으로 변신하기 위한 감지 센서이다.

초음파센서는 자동 주행 시 장애물을 회피하기 위하여 사용되었다. 또한 리미트센서는 그리퍼와 뚜껑이 열고 닫힐 때를 판단해주는 역할을 하였다.

물체인식에서 영상에만 의존을 하게 되면 영상 신호의 불확실성으로 물체를 지나가거나 통신상의 오류로 인하여 물체인식을 실패할 경우가 발생하기 때문에 부가적으로 그리퍼 앞부분의 PSD센서를 이용하였다. 화면의 특정부분에 물체가 걸리면

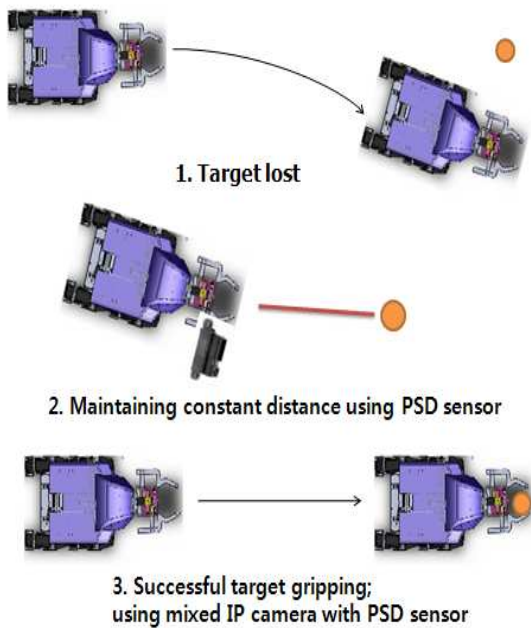


Fig. 5 Object identification with camera and PSD sensor

스마트폰에서 명령을 보내주며 PSD의 값을 받아 물체와의 거리를 측정한다. DSP에서 스마트폰으로 물체가 인식되었다는 확인을 통신으로 보내주면 영상처리를 하여 잡으려는 물체가 맞는지 확인 후 물체를 잡는다. 물체인식을 할 때 카메라만 의존하는 것이 아니라 센서와 함께 사용하여 사용해서 물체의 인식 성공확률을 높였다.<sup>(9)</sup> Fig. 5에서 각 경우에 대한 로봇이 수행하는 행동을 보여주고 있다.

3.1 영상처리 시스템의 구성

3.1.1 소프트웨어 구성

로봇의 구동과 약통을 인식하기 위한 시스템의 소프트웨어 환경은 Table 1과 같다.

본 연구에서는 MAC OS 기반의 Objective-C 언어를 이용하는 X-code 프로그램으로 제작하였으며 스마트폰은 Apple A5(dual-core) 프로세서가 내장된 I-Phone 4S를 사용하였다. 윈도우 프로그램과 마찬가지로 라이브러리와 UI (user interface)를 사용자 편의에 맞도록 꾸밀 수 있는 View가 있으며 코드와 인터페이스간 링크를 할 수 있다. 링크를 하는 방법은 윈도우 프로그램과 다르게 명시적으로 타겟을 지정할 수 있다.

스마트폰 이벤트는 마우스와 키보드가 아닌, 스크린을 터치하는 것으로 구성되어 있다. Objective-

Table 1 Software development environment of MAC and Smartphone

Operation system	MAC
Programming Language	Objective-c
Smart-phone	I-Phone 4S

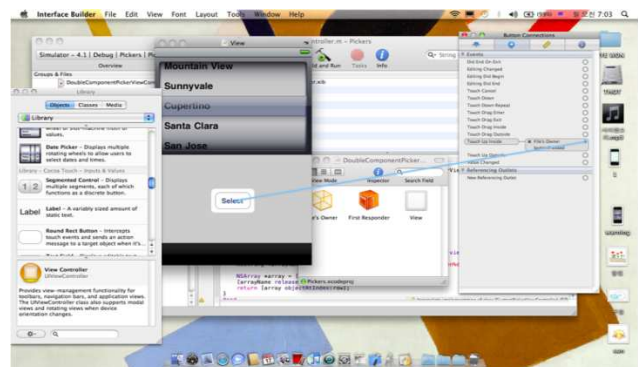


Fig. 6 Programming process for I-phone from X-code



C는 C언어를 기반으로 하고 있으며, 최초의 객체 지향 프로그래밍 언어중의 하나인 Smalltalk<sup>(10)</sup>의 장점을 가진 언어이다. 또한 Fig. 6 에서 보듯이 실제적으로 사용하는 UI를 직관적으로 프로그래밍 할 수 있으며 X-code에서 지원하는 Simulator를 통하여 컴퓨터 상에서 실제적으로 UI 구현상황을 테스트해 볼 수 있다.

구성은 주로 Model, View 그리고 Controller 세 개의 객체로 구성되는데 여기서 Model은 어플리케이션에서 사용되는 데이터들을 저장하는 클래스이며 View에서는 흔히 볼 수 있는 윈도우나 각종 버튼, 텍스트 박스 등등 실제로 보이는 부분을 총칭한다.

Controller는 Model과 View 사이에서 서로의 입출력을 조정하고 제어하는데 이 3가지 개념을 이용해서 최대한 독립적으로 프로그래밍하여 객체를 구성하는 것을 목표로 했다.<sup>(11)</sup>

본 시스템에서는 로봇과의 통신 및 카메라의 스트림 데이터(Stream data)를 연결해주는 URL Connection을 이용한 두 개의 네트워크가 구성된다.

로봇과의 통신은 BSD 소켓 통신을 이용하였으며 Sonde 알고리즘<sup>(12)</sup>을 이용하여 스트림 데이터를 이용한 영상처리 기법을 이용하였다. 즉 BSD 소켓을 이용하여 로봇과 통신하면서 간단하게 로봇을 조작할 수 있으며 실시간으로 약통을 찾아가기 위하여 URLConnection을 이용하여 카메라 영상을 확인한다.

IOS에서 지원하는 URLConnection을 이용하면 카메라의 스트림 데이터를 didReceiveData Method를 이용하여 일반적인 소켓통신처럼 스트림 데이터를 열어(open) 실시간으로 데이터를 받아서 영상처리를 할 수 있으며 간단한 Cancel Method를 이용하여 통신을 마칠(close) 수 있다. IOS에서 URLConnection은 데이터를 실시간으로 받는 경우, 연결 에러가 발생하는 경우, 데이터 로딩(Data Loading)이 끝나는 여러 가지 경우에 대하여 기준을 제공한다.

### 3.1.2 BSD 소켓

스마트폰의 Wi-Fi통신으로 로봇에 장착된 DSP와 명령어를 주고받기 위해 Wi-Fi 모듈을 이용한다. Wi-Fi 모듈은 Wi-Fi 통신과 직렬통신을 상호 변환 시켜주는 역할을 한다. 스마트폰 Wi-Fi는 개발 환경 X-code에 I-Phone SDK를 설치해 직접 Device나 시뮬레이터를 구동시켜 디버깅을 할 수 있도록 지원해 주기 때문에 쉽게 사용이 가능하다.

**Table 2** Wi-Fi communication system between robot and smart phone

Smart phone →DSP	Smart-phone Wi-Fi → robot WI-FI module → Serial communication → Robot DSP
DSP →Smart phone	Robot DSP → Serial communication → Robot Wi-Fi module → Smart-phone Wi-Fi

스마트폰과 로봇간의 Wi-Fi 통신의 과정을 간략하게 나타내면 Table 2와 같다.

네트워킹은 여러 단계의 계층(Layer)를 가지고 있는데 가장 하위에는 BSD 소켓이 자리 잡고 있으며 이 위로 계층들이 있는데 물론 계층이 올라갈수록 추상화의 정도도 높아진다. BSD 소켓의 바로 위에는 CFNetwork 계층이 있다.

우리는 무선 통신 중에 블루투스, 지그비(Zigbee) 등이 있으나 가장 빠른 Wi-Fi 통신을 이용하였다. 참고로 IEEE \*802.11 Wi-Fi 프로토콜 중 802.11n은 이론적으로 최대 600Mbps까지 가능한 것으로 나와있다. Wi-Fi를 사용하기 위하여 IP주소와 포트(Port)가 필요하고 영상을 획득하기 위한 IP 카메라, 로봇을 제어하기 위한 MCU인 DSP 2812, 그리고 로봇을 조정하고 영상처리를 하기 위한 스마트폰 이 세 가지의 Process들이 서로 통신하기 위한 방법이 필요했다. 이 세 가지의 프로세스들을 앞에서 언급한 Fig. 2와 같이 클라이언트 이용하고 클라이언트들이 서로 통신하기 위하여 서버로 라우터를 사용하였다. 로봇 접속에 필요한 IP주소는 라우터에서 DHCP로 자동적으로 할당이 된다. 스마트폰에서 보내 주는 명령어를 확인하기 위하여 Device\_Terminal 소프트웨어를 이용하여 통신 값을 확인해 볼 수 있었다.

### 3.1.3 카메라 데이터의 처리

스마트폰을 이용한 영상 처리의 연산 속도에 제한이 있으므로 데이터를 최소화하기 위하여 자체적인 통신을 구축하였다. URLConnection을 이용하여 스트림 데이터를 획득하였으며, Marker를 이용하여 이미지의 시작과 끝을 찾아내 이미지 데이터만을 스마트폰에서 확인할 수 있도록 하였다. 이는 스트림 데이터 중 음성 데이터, 카메라를 구동하는 PTZ (Pan, tilt, zoom) 모터 제어 데이터 등 많은 데이터가 들어오기 때문에 스마트폰에서 구현하기 위한 데이터를 최소화 하기 위하여 반드시 필요하다. Fig. 7은 영상데이터의 시작인 ffd8을 찾아서 시작하는 부분을 나타내며 영상데이터의 마지막인 ffd9를 찾아내 이미지 데이터의 추출을 하는 부분을 보여주고 있다.

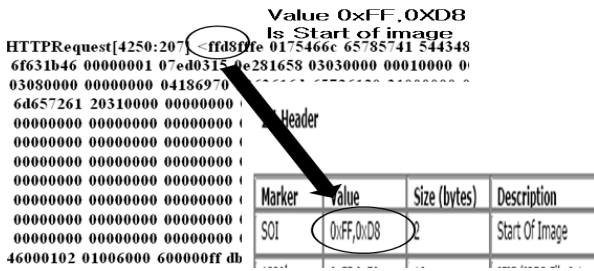


Fig. 7 Start Bit and End of Image Data

이미지가 시작되는 부분에 NSURLConnection을 이용하여 들어오는 데이터를 데이터 변형이 가능한 NSData의 형태로 저장하였고 데이터가 끝나는 EOF에서 완성된 이미지를 실시간으로 이미지 창에 뿌려줌으로 실시간 영상이 완성된다.

3.1.4 영상 처리

로봇이 약통을 가져다 줄 때 영상 처리를 하여 약통을 인식하고 로봇이 스스로 약통을 잡을 수 있도록 영상처리를 통하여 약통의 형태나 색깔을 구분하게 하였다. 실험에 사용한 약통의 사진은 Fig. 8에 나타나 있다. 각기 다른 색깔로 구별된 약통을 대상으로 영상처리 실험을 스마트폰에서 수행하였다. 기본적인 RGB 색 공간(Color Space)는 조명과 같은 주변 환경에 영향을 많이 받기 때문에 RGB 색 공간의 단점을 보완할 수 있는 HSV 색 공간<sup>(13)</sup>을 사용하였다. Hue 채널과 Saturation 채널을 함께 사용하여 원하는 색 정보에 대한 임계값을 실험적으로 구하여 범위를 지정하였다.<sup>(14)</sup> 영상처리는 스마트폰 내에서 데이터 처리를 통하여 이루어지므로 기존의 영상처리를 컴퓨터에서 처리하여 무선으로 결과를 알려주는 방식과는 차별성이 있다.

여기서 스마트폰만을 이용한 연산 속도에 한계가 있고 Hue 값과 Saturation 값만으로도 원하는 색 정보의 추출이 가능하기 때문에 Value 값 연산은 하지 않았다. 결과는 Fig. 9와 같다.

Fig. 9는 원 영상을 HSV 색 공간을 이용하여 약통의 이미지만을 추출한 영상이다.



Fig. 8 Drug bottles used in experiments



Fig. 9 Drug bottle represented by HSV Color Space

여기서 보는 것처럼 같은 RGB상에서 같은 빨간색 공간이 라도 약통의 이미지만 추출되는 것을 확인할 수 있다.

로봇이 약통을 인식하고 잡을 때는 카메라와 PSD센서를 함께 사용하였다. 주변 환경에 민감한 카메라의 단점을 인식률이 높은 PSD센서를 이용하여 물건 인식의 정확도를 높였으나 PSD센서만을 이용하여서는 로봇이 계속 움직이므로 약통과의 정확한 거리의 측정이 힘들어 카메라를 함께 이용하여 약통과 로봇과의 거리를 측정할 수 있는 방법을 고안하였다.

그러나 하나의 카메라로는 스테레오비전과 같은 기술을 사용할 수 없기 때문에 생각해낸 방법은 원근법에 의한 방법이다. 물체가 가까워지면 가까워질수록 화면에서 보이는 크기는 점점 커지기 때문에 다음과 같은 방법을 이용하였다.

약통의 뚜껑에 대해 Fig. 10에서 보이는 붉은 라인 부분을 이진화를 시켜서 검은색 배경의 흰색의 픽셀 값을 계산하였다. 원근법에 의하며 물체가 가까워지면 가까워질수록 Fig. 10에서 보이는 이진화된 흰색의 픽셀 값이 증가하기 때문이다.

실시간으로 화면의 중간 부분에 흰색의 가로 길이를 측정된 값과 우리 로봇에서 그리퍼의 안쪽에 위치하였을 때의 픽셀 정보를 가지고 있어 현재

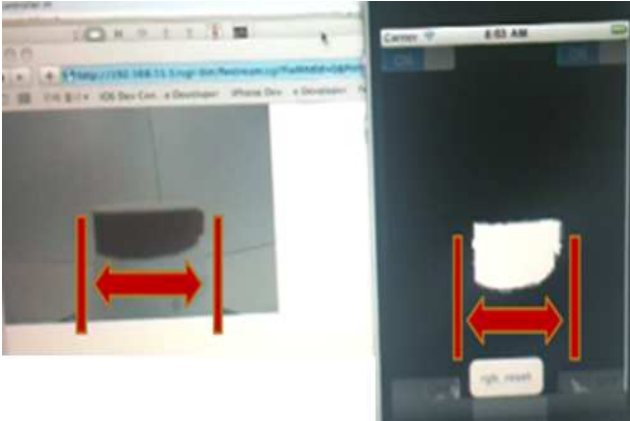


Fig. 10 Drug bottle image (left) and binary image (right)

계산된 픽셀 값과 저장되어 있는 픽셀 값을 매칭시켜 약통이 로봇의 그리퍼 안쪽에 위치한 것이 확인되면 그 순간에 로봇을 멈추고 약통을 집을 수 있도록 DSP 에 명령을 보내 주었다. DSP에서는 이미 Test를 통하여 PSD가 인식할 수 있는 30cm내의 범위에 대한 정보를 가지고 있으며 PSD에 물건이 감지되는 경우 다가가서 물건을 잡고 물건의 감지되지 않을 경우 스마트폰으로 물건의 없음을 알려준다.

또한 로봇이 스스로 약통을 찾아 가기 위한 모드일 때 실시간으로 영상을 처리하고 스마트폰 자체로 모든 화면을 스캔하고 영상처리를 하기 위해서는 실시간으로 연산을 처리하는 속도를 따라가지 못하기 때문에 Fig. 11와 같은 Sonde 알고리즘<sup>(9)</sup> 적용시켰다. Sonde 알고리즘은 보통 문자를 인식하기 위하여 많이 사용하는 알고리즘으로 예를 들어 숫자 8을 인식하기 위하여 1개의 수직선과 2개의 수평선을 이용한 것이다. 이와 마찬가지로 우리 로봇에서는 전체를 스캔 하는 것이 아니라 Fig 11에서의 이미지 창에서 붉은 부분으로 표시한 곳만큼의 Pixel을 스캔 하여 HSV 색 공간으로 변환한 후 검사한다.

Sonde 알고리즘은 2개의 수평선과 1개의 수직선 부분만의 이미지 데이터정보를 검사한다. 이는 화면상에서 전체 부분을 검사하는 것보다 일부분만을 검사하여 전체 화면의 Pixel을 검사하는 것보다 훨씬 적은 메모리가 사용되므로 연산 처리의 속도가 매우 빨라 연산 량이 많은 실시간 이미지 데이터의 경우 효과적으로 사용할 수 있다. 유사하게 2개의 수직선과 1개의 수평선 부분만을 검사하며 로봇이 자율적으로 다니다가 Fig. 11 우측에서의 이미지 창에서 붉은 부분으로 표시한 곳에 약통이 있음을 인지하면 특정한 명령(좌, 우, 정지 등)을 받으며 약통을 찾아 추적하게 된다. 이는

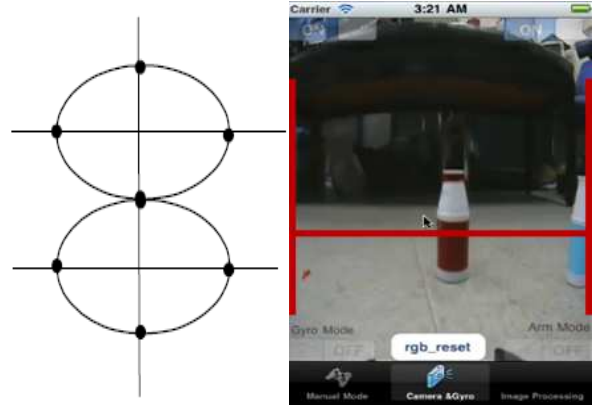


Fig. 11 Sonde algorithm and its application to image scanning

스마트폰 자체에서 실시간으로 영상처리를 하는데 실시간으로 테스트하고 검증하면서 최적화한 영역이다. 만약에 이보다 스캔하는 부분이 많아지게 되면 스마트폰에서 연산처리 속도를 따라가지 못해 카메라 영상이 매우 느려지거나 끊기는 현상이 발생하고 만약에 현재 영역보다 스캔하는 범위가 적어지게 된다면 적은 부분을 검출하기 때문에 약통이 인식되지 않는 경우가 발생할 수 있다.

화면의 하단에 수평선의 역할은 멀리서부터 약통이 인식 되어 점점 가까워지게 되면서 원근법에 의하여 약통이 점점 확대가 되는데 PSD 센서를 사용하기 위해 우리가 원하는 정도의 위치가 되면 정해놓은 수평선의 픽셀 영역에 검출이 가능하게 되며 로봇은 정지하고 PSD 센서를 이용하여 물체를 확인하게 된다.

### 3.1.5 알림 기능

개발 로봇은 거동이 불편한 장애자나 노약자 분들을 대상으로 하였는데 사용자가 약을 먹을 시간을 깜빡 하고 안 먹는 경우가 종종 발생하기 때문에 어플리케이션 프로그램에서 약 먹을 시간을 알려줄 수 있는 기능을 추가하였는데 IOS에서 제공하는 UILocalNotification Class Reference<sup>(16)</sup>를 이용하였다. 이는 어플리케이션이 실행되거나 Background 모드일 때 Push 알림을 구현할 수 있다.

우리 개발 로봇의 어플리케이션이 실행이 되어 있을 때 시간을 맞추어 놓으면 특정한 이벤트가 발생하여 로봇이 스스로 움직이도록 프로그램 되어 있으며 만약 어플리케이션을 백그라운드 모드로 해놓았을 때는 Fig. 12에 보는 것과 같이 알림창을 띄워서 사용자들에게 약을 먹을 시간임을 인식시켜주며 사용자는 스마트폰의 어플리케이션을 실행하여 로봇을 구동하게 된다.



Fig. 12 Alarm operation using Local Notification function



Fig. 13 Graph showing memory leak

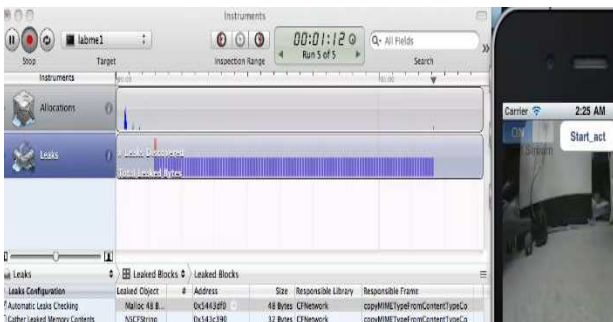


Fig. 14 Graph of memory leak after modifying memory leak confirmation

3.1.6 메모리 관리

스마트폰에서 영상 처리의 연산처리 속도는 일반적인 컴퓨터를 따라가지 못하기 때문에 효율적인 메모리 관리가 필요했고 더욱이 실시간 영상처리를 위해서 메모리 누수가 절대 발생해서는 안 된다.

따라서 실시간 영상처리 과정에 메모리 누수를 최소화하며 또는 없애기 위하여 X-code 시뮬레이션에서 지원되는 누수(Leaks) 기능을 이용하였다.

Table 3 Success rate of identifying drug bottle for their colors and bottle delivery speed

	Red	Blue	Green
성공률	70%	85%	90%
약병전송속도	3분 10초	2분58초	2분 55초

이는 실시간으로 메모리 누수가 발생하면 그래프로 확인할 수 있을 뿐만 아니라 어떠한 부분에서 메모리 누수가 생기는지를 확인할 수 있다. Fig. 13은 처음 프로그램을 실행 했을 때 그래프에서 보이는 것처럼 데이터가 점차 누적이 되면서 쌓이는 것을 확인할 수 있다. 이는 메모리 누수가 발생하는 것을 의미하며 프로그램을 수정했을 때 Fig. 14와 같이 그래프의 형상이 점점 쌓이는 형태가 아니고 처음의 사용되었던 데이터들이 변함없이 일정하게 관리되는 것을 확인할 수 있다. 처음의 경우 우측에 위치한 카메라 영상이 어플리케이션이 시작될 때 잘 작동하나 점점 시간이 지남에 따라 데이터 누수가 발생하여 점점 느려지고 끊기는 것으로 확인 되었으나 프로그램 수정 후에 그런 현상들은 없었다.

4.1 실험 결과

4.1.1 약병의 종류별 영상 처리 결과

본 연구에서 우리는 여러 가지 색상의 약병들을 테스트 하였는데 HSV 색 공간은 RGB 색 공간의 단점을 보완해 원하는 색상만을 찾아내 추출할 수가 있으나 조명에 따른 반사광 등에 따라서 민감하게 반응하는 것을 확인했다. 그 이유는 영상처리기에 있어 컴퓨터나 스마트폰은 물체의 질과 물체가 빛을 반사하는 강도 등에 따라서 반사 질이 영향을 받기 때문이다. 앞서 실험에서는 붉은색의 유광 재질의 약병을 사용하였으나 반사광이 심한 조명에서는 Hue 채널과 Saturation 채널을 변경하여 다른 색의 약통을 추출하였고 일반적인 방안의 상황을 꾸며놓은 세트장에서의 실험 결과 약병을 집어 오는데 걸리는 평균 시간은 3분 내외였다.

4.1.2 영상처리 시간

실시간으로 전체 이미지뷰(image view)를 영상 처리하게 되면 실제 카메라의 영상보다 3초나 경우에 따라서는 더 많은 시간지연이 생기며 데이터가 실시간으로 계속 들어오지 않아 영상이 끊겨서 보이는 것을 확인하였다.

이는 스마트폰 기능이 날로 확장되고 있으나 전



체 이미지뷰를 영상처리 하기에는 하드웨어 사양이 아직까지는 컴퓨터 보다는 약간 떨어지기 때문이다. 따라서 스마트폰으로 전체 이미지뷰를 영상처리하면 약병을 인식하여도 실제 비추는 이미지뷰와 영상 처리 된 후의 이미지뷰의 속도 차이가 발생하게 되어 로봇이 인식하지 못하고 물체를 지나치는 오차가 발생하게 된다. 따라서 우리는 Sonde 알고리즘을 적용하여 영상처리 하는 픽셀을 줄임으로 실시간 영상처리를 할 때 연산 속도를 높일 수가 있었다.

Sonde 알고리즘을 적용한 이미지뷰와 실시간 보이는 카메라 이미지뷰는 거의 시간지연 없이 동일함을 확인하였다.

#### 4.1.3 영상 데이터 처리 속도

카메라의 스트림 데이터는 사용자가 서버 측에 원하는 데이터의 포맷을 요청하면 서버는 요청 받은 데이터들을 클라이언트로 전달하는 일종의 폴링(Polling)방식으로 이미지 파일을 전송 받을 수 있다.<sup>(15)</sup> 영상처리를 하지 않은 기존의 이미지 데

이터 중지시간(Fig. 15)을 0으로 놓고 실시간으로 데이터를 받아도 스마트폰에서 영상을 확인하는데 문제가 없으나 약통을 추출하는 과정에서는 속도가 다소 지연되는 것을 확인할 수 있었다. 따라서 평상시 카메라 영상을 확인 할 때는 실시간 영상을 확인하고 영상처리 과정이 들어가게 되면 중지시간을 8로 설정하였다. 여기에서의 단위는 10ms이며 중지시간을 8로 설정하였다면  $8 \times 10\text{ms} = 80\text{ms} = 0.08\text{sec}$ 가 된다. 즉 0.08sec 단위로 하나씩의 jpeg의 이미지가 전송되는데 즉 1초당 12.5 프레임이 완성되었다.

Fig. 16는 한 장의 프레임이 완성될 때마다의 Red, Green, Blue 각각 약통에서 HSV Color Space로 변환하는데 걸리는 시간을 정량화하여 나타내었는데 약통의 색에 상관없이 평균적으로 30ms 정도가 걸리는 것을 확인할 수 있었고 앞에서의 언급한 것과 같이 데이터 중지시간은 8로 설정하면 영상이 HSV로 추출되고 이후 실시간 영상화면을 보면서 로봇이 약병은 가지고 오는 경우 영상의 시간 지연 문제가 없었다.

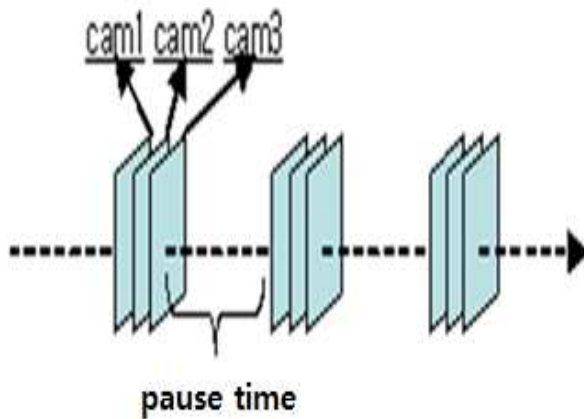


Fig. 15 Concept of pause time

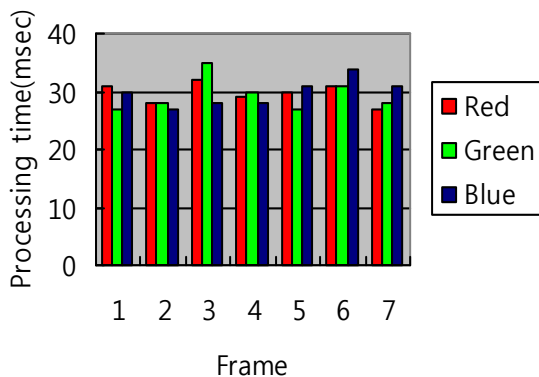


Fig. 16 Processing time for bottle classification

#### 4.2.1 실험결과

여러 가지 실험 결과 센서를 사용하지 않고 카메라 영상에만 의존하여 Sonde 알고리즘만을 적용하였을 때 로봇을 완벽하게 구현하기가 힘들었다. 그러나 PSD 센서와 초음파센서를 함께 융합하여 구현하였을 때는 카메라에만 의존한 것 보다 약병을 잡아내는 오차율을 줄일 수 있었고 80% 이상의 성공확률을 실험 결과로 알 수 있었다.

Table 4 Success rate of drug bottle identification for locations

장소	작동 환경	성공률
연구실	쾌적	90%
학교 운동장	쾌적	85%
공대 건물 로비(주간)	카메라 영상 스트림 약간 끊김	75%
공대 건물 로비(야간)	쾌적	85%
대학기념관 전시회장	카메라 영상 스트림 끊김	60%
포항체육관 대회장	카메라 영상 스트림 매우 끊김, 명령 반응 매우 느림	20%

실패의 주요 원인은 실제 실험에서 오는 센서 데이터 값의 차이와 무선 통신의 트래픽이다. MCU인 DSP2812만을 이용하여 측정한 센서 데이터 값은 일정하게 측정이 되나 실제로 로봇에 장착하여 스스로 약통을 찾아가도록 구현할 때 상황에 따라 들어오는 센서 데이터 값은 주변의 노이즈에 의해 약간의 차이가 있었다. 또한 현재 스마트폰을 비롯한 많은 기기들이 Wi-Fi 통신을 사용하는데 장소 별 인식 성공률은 Table 4와 같고 예를 들어 전시회장 같이 사용자가 아주 많은 환경에서는 주파수 대가 겹쳐 영상 스트림 데이터와 로봇과 스마트폰의 소켓 데이터가 트래픽이 많이 발생하여 데이터가 늦게 들어오게 되어 오차가 생기게 되는 점은 앞으로도 해결해야 되는 점이다.

## 5. 결 론

본 연구는 스마트폰으로 실시간 영상처리를 하며 물체를 찾아가는 약병전송 로봇시스템에 관한 연구이다. 스마트폰을 이용하여 카메라 스트림 데이터에서 이미지 부분만을 분할하는 처리를 하였고 이미지 데이터를 통하여 물체를 추적하는 알고리즘을 구성하였다.

실시간으로 영상처리를 구현하기 위해 Sonde 알고리즘을 통하여 부분적으로 영상처리를 하여 최적화 하였다. 따라서 실시간 영상처리가 가능할 만큼 빠른 처리속도를 가진다.

또한 일반적인 RGB 색 공간은 조명이나 주변 환경에 민감한 반응을 보이기 때문에 특정한 물체를 추적하기 위한 HSV 색 공간을 사용하였다.

영상처리를 통하여 인지된 물체의 추적을 위한 정보를 BSD소켓통신을 이용하여 Wi-Fi 모듈로 전송하였고 이를 DSP에서 시리얼 통신을 이용하여 물체를 찾아가서 잡도록 구현하였다.

컴퓨터를 이용하지 않고 스마트폰만을 이용하여 영상처리 및 로봇제어의 구현을 가능하게 하였으며 핸드폰 카메라가 아닌 IP카메라를 이용하여 실시간으로 영상처리를 가능하도록 하였다.

로봇이 물체를 추적하기 위하여 카메라에만 의존하는 것이 아니고 로봇의 센서와 함께 센서 퓨전을 통하여 높은 인식률을 나타내었으며, 현재처럼 약병만 인식하는 것이 아니고 많은 물건들을 찾아 거동이 불편한 사용자들을 대신해 줄 수 있도록 데이터베이스를 저장하고 인식할 수 있도록 알고리즘의 보완이 이루어지면 거동이 불편한 분들에게나 사회 복지 시설 등에서도 매우 효과적으로 적용될 수 있을 것이다.

## 후 기

이 연구는 서울과학기술대학교 교내 일반 과제 연구비 지원으로 수행되었습니다.

## 참고문헌

- (1) Lim, N., Kang, H., Park, Y., Ahn, D., Oh, J. and Song, J. H., 2009, "Cognitive Function, Mood, Problematic Behavior and Response to Interaction with Robot Pet by Elders with Dementia," *Korean Acad Fundam Nurs*, Vol.16, No.2, pp 223~231.
- (2) Kwak, B. and Lee, J., 2010, " Smartphone Application Interface for Intelligent Human-Robot Interactions," *Korea Computer Congress*, Vol.37, No1(C), pp 399~402.
- (3) Burnette, E., 2009, "Android basis Tutorial," B. Zhang, C. Gao, and Y. Yang, Translation. Beijing: Posts & Telecom Press, pp. 11~189.
- (4) E2ECloud Studio, 2009, "Simple Terms Google Android," Beijing: Posts & Telecom Press, pp. 39~256.
- (5) Hu, P., Symons, N., Indulska, J. and Portmann, M., 2012, "Wireless Multi-hop Video Streaming using Android phones," *8th IEEE PerCom Workshop on Pervasive Wireless Networking*, pp.782~787.
- (6) Texas Instruments, 2009, TMS320x281x Serial Communications Interface (SCI) Reference Guide, TI.
- (7) Texas Instruments, 2011, TMS320F2810, TMS320F2811, TMS320F2812, TMS320C2810, TMS320C2811, TMS320C2812 Digital Signal Processors Data Manual, TI.
- (8) Texas Instruments, 2005, TMS320x281x System Control and Interrupts Reference Guide, TI.
- (9) Texas Instruments, 2005, TMS320F28x DSP Analog-to-Digital Converter(ADC) Reference Guide, TI.
- (10) Choi, D. and Oh, H., 1993, "Efficient Memory Management Technique in an Object-oriented Programming System," *Korea Computer Congress*, Vol.20, No.2, pp. 461~464.
- (11) Mark, D. and Lamarche, J., 2008, "Beginning Iphone Development: Exploring the Iphone SDK(Exploring the iPhone SDK)," APRESS, NewYork, pp. 57~96.
- (12) Lee, P., 2000.2, "Development of an Image Processing Algorithm for the Recognition of Car Types and Number Plate," The University of SEOUL, Master's Thesis, pp. 7~11.
- (13) Bradski, Gary and Kaehler, 2008, "Learning OpenCV: Computer Vision with the OpenCV Library," Oreilly & Associates, California, pp. 31~87.
- (14) RANDY CRANE, 1996, "A simplified approach to

- image processing," Prentice Hall, Texas, pp.18-134
- (15) Jooihl Chon and Sungyoung Lee, 2000, "A Bidirectional Push/Pull Transmission Method Between Client and Server in Multimedia Environments," *Korea Multimedia Society*, 2000, Apr, pp. 36~39.
- (16) IOS Developer Reference [developer.apple.com/iphone](http://developer.apple.com/iphone))