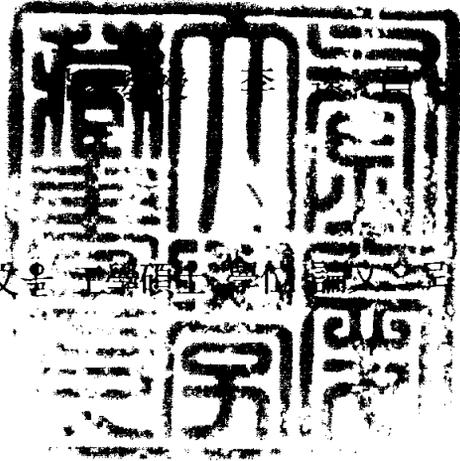


工學碩士 學位論文

H.263 영상압축 방식을 이용한  
인터넷 기반 탐사робот의  
원격 운용 시스템



이 論文은 工學碩士學位論文으로 提出함

2004年 2月

釜慶大學校 大學院

電子工學科

崔 峻 榮

# 崔峻榮의 工學碩士 學位論文을 認准함

2003年 12月 26日

主 審 工學博士 姜 根 澤



委 員 工學博士 崔 赫 煥



委 員 工學博士 林 昌 憲



# 목 차

Abstract .....	2
I. 서 론 .....	3
II. 원격 운용 시스템 .....	5
2.1. 시스템 전체 구조 .....	5
2.2. 로봇 시스템 .....	6
2.3. 클라이언트 시스템 .....	8
III. 영상 압축 .....	10
3.1. H.263 개요 .....	10
3.2. H.263 압축 방식 .....	11
IV. 하드웨어 구성과 제어기 .....	15
4.1. 로봇로봇의 구성 .....	15
4.2. DC모터 제어기 .....	18
V. 실험 및 검토 .....	21
5.1. H.263으로 영상 압축과 전송 .....	21
5.2. Time delay .....	23
5.2. 검토.....	24
VI. 결 론 .....	26
참고문헌 .....	27

**Teleoperation System of a Networked Mobile Robot  
Based on H.263 Video Codec**

Jun Yeong Choi

*Department of Electronic Engineering, Graduate school,  
Pukyong National University*

**Abstract**

In this paper we present a teleoperation system of an internet-based mobile robot based on H.263 image compression algorithm. We control remotely the networked mobile robot with vision over the internet under unknown environments in the real time. The main feature of this system is that local operators need to download and install a client program and so they can command the robot in a remote location. The image information of remote location is compressed by H.263 video codec algorithm and then transmitted to a local operator. The additional feature of the system is to handle the network latency occurring when the mobile robot transmits the compressed image data and receives commands from operators. We propose a new algorithm based on time stamp, which is turned out to be effective. We built and tested a networked autonomous mobile robot in the real field. The experimental results show the usefulness of the robot system including all the algorithm implemented on it.

# I. 서론

과학이 발달함에 따라 로봇의 이용이 보편화되었으나 로봇은 단순 작업밖에 할 수 없다고 인식이 되어왔고 사실 대부분의 응용이 그러한 범주를 크게 벗어나지 않았다. 그러나 최근 수 년 간의 눈부신 연구 업적들에 의해 로봇이 단순한 작업만을 반복할 수 있다는 고정 관념들이 깨지고 있다. 미약하지만 스스로 감지하고, 배우며 판단할 수 있는 알고리즘들이 제안되었고, 이를 바탕으로 인간의 조작 없이 스스로 간단한 경기를 할 수 있는 로봇이나 인간과 친근하게 함께 생활할 수 있는 애완 로봇, 나아가서는 인간과 비슷한 겉모습과 구조를 가진 인간형 로봇이 등장하고 있다. 이러한 비약적인 발전에도 불구하고 로봇이 인간을 대신할 수 없는 부분은 여전히 남아 있다. 작업 절차가 표준화되어 있지 않거나 돌발적인 상황이 발생할 가능성이 있는 작업의 경우에는 단순한 자동화나 완전하지 않은 로봇의 대처 능력으로는 한계가 있다. 우주 공간에서의 작업, 원자력 발전소의 유지와 보수, 심해저에서의 작업, 광산에서의 작업 등이 그러한 예이다. 이러한 작업들에 로봇을 투입하고 작업자는 원거리에서 위기 상황에 대비한다면 작업의 위험을 최소화함과 동시에, 뜻하지 않은 돌발 상황에도 능동적으로 대처할 수 있다는 장점이 있다. 현대에 이르러 인터넷의 비약적인 발전으로 인해 인터넷을 이용한 원격운용 시스템은 이러한 작업이 가능하다[1][2].

1994년 웨스턴 오스트리아 대학에서 시작한 초기의 원격 제어시스템인 머큐리 프로젝트[3]는 제한된 환경에서 단순한 동작만을 수행하였지만, 현재의 원격 제어 시스템은 의료진단[4], NASA의 인터넷을 이용한 우주탐사 차량제어[5], 자바 인터페이스를 이용한 인터넷상에서의 페인팅 로봇 제어[6]등 보다 정밀하고, 높은 효율성을 요구하는 분야로 범위를 넓혀가고 있으며, 그 실용성과 필요성이 증명되고 있다.

본 논문에서는 탐사로봇을 원격지에서 운용함으로써 기존의 기능에서 탈피한 능동적인 운용을 하고자 하며, 인터넷을 이용함으로써 운용 공간의 한계를 극복하고, 원격지의 탐사 로봇과 양방향 통신을 통하여 로봇을 효율적으로 운용하고자 한다. 로봇을 원격지에서 운용하려면 원격지의 정보가 필요하다. 일반적으로 원격지의 정보를 얻는 방법으로는 여러 센서를 이용해서 얻을 수 있다.. 여기서는 그중 카메라를 통해 원격지의 영상정보를 얻어 탐사로봇을 운용한다. 영상데이터는 원격지의 영상정보를 실시간으로 획득하기 위해 영상 데이터를 압축해서 보내야 한다. 이에 보다 효과적이고 높은 압축률을 가지는 압축 알고리즘 개발이 쟁점이 되고 있으며 이러한 연구에 있어 1996년에 표준화 작업이 마무리된 ITU-T/LBC 그룹의 H.263은 64Kbps 이하의 전송 속도를 갖는 동영상 압축 표준안으로서 공중전화망이나 이동통신망과 같은 초 저속망에서의 영상 전화, 영상 회의 등의 실현을 그 목적으로 하고 있으며 인터넷을 이용하여 실시간적인 영상획득방법의 일환으로 그 실용성이 증가하고 있다[7].

원격 운용 시스템은 영상 처리, 컴퓨터 네트워크 프로그램, 인공 지능, 각종 센서, 제어 알고리즘등 여러기술이 효율적으로 연계되어야만 성공적으로 수행되질 수 있다. 본 논문에서는 이미지 전송을 위한 영상 압축프로그램, 소켓을 이용한 네트워크 프로그램, 시리얼통신 프로그램을 실제 구현하여 로봇 시스템과 클라이언트 시스템으로 구성되는 원격제어 시스템을 구축하였고 로봇을 설계하여 DC모터 제어알고리즘을 사용하여 원격지에서 로봇의 실시간 제어 가능성을 검증하고자 한다.

## II. 원격 운용 시스템

### 2.1 시스템 전체 구조

원격 운용 시스템은 그림 1과 같이 로봇 시스템, 클라이언트 시스템으로 나누어진다. 로봇 시스템과 클라이언트 시스템은 윈도우즈 환경에서 GUI(Graphic User Interface), 객체지향, 멀티스레드기능이 뛰어난 언어인 MFC(Microsoft Foundation Class)로 구현된다.

사용자가 탐사로봇을 이용하기 위해서는 인터넷이 연결된 컴퓨터에서 로봇에 접속함으로써 원격지에서 탐사로봇을 운용할 수 있다. 이를 위해서는 인터넷이 사용가능한 컴퓨터가 있어야한다. 또한 로봇은 클라이언트와 통신을 하기 위해 고유한 주소(Ethernet Address)를 가져야한다. 로봇 시스템과 클라이언트 시스템은 무선랜(Wireless Lan)을 사용하여 영상 정보와 명령을 교환한다.

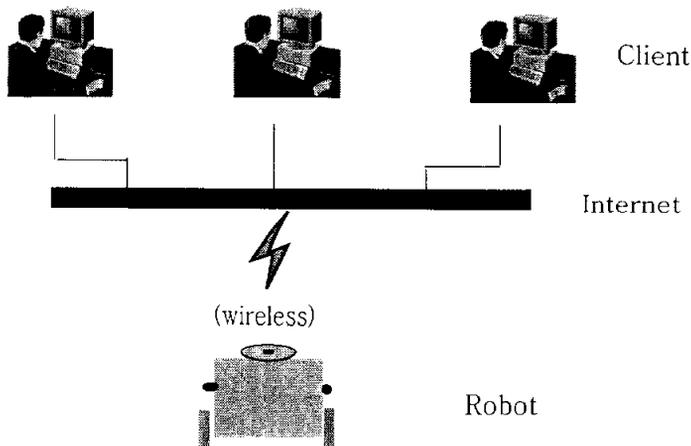


그림 1. 시스템의 전체 구조

## 2.2 로봇 시스템

로봇 시스템에는 영상 압축 프로그램, 서버 프로그램, 로봇제어 명령 프로그램으로 구성되어 있다.

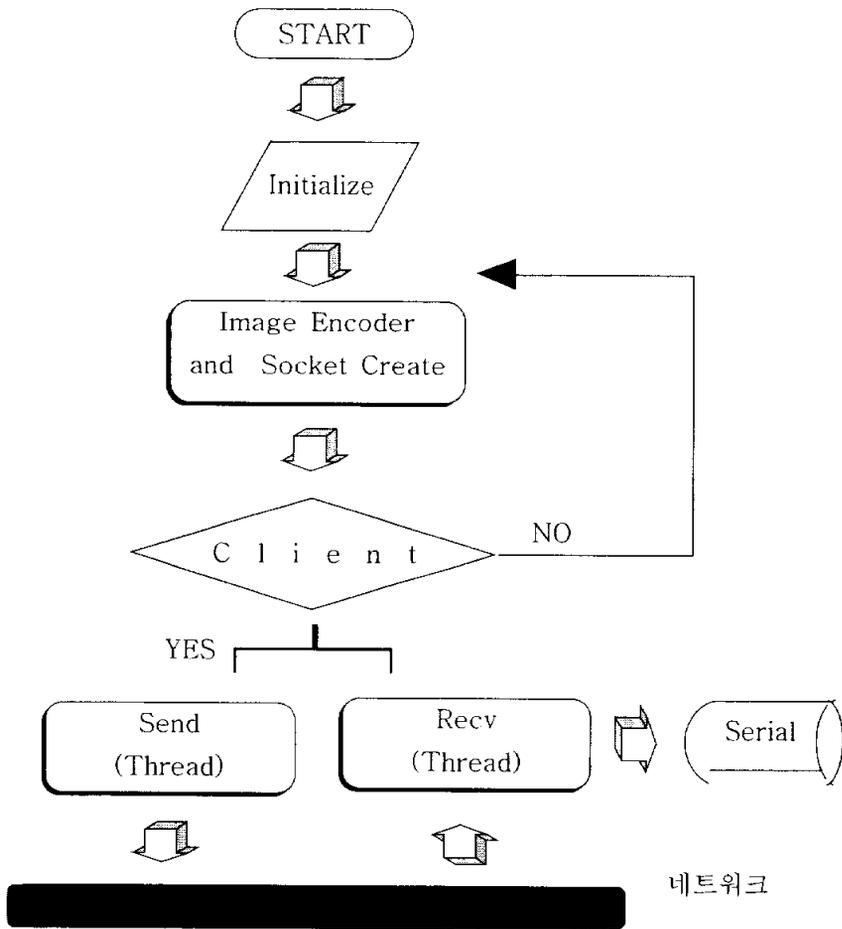


그림 2. 로봇 프로그램 개략도

영상 압축 프로그램은 초 저속망에서 유용하고 압축률이 높은 h.263 압축방식을 이용한 알고리즘을 사용하였다. 서버프로그램은 클라이언트 시스템이 로봇에 접속할 때 로봇 시스템에 필요한 네트워크 프로그램으로 MFC 소켓을 사용하였고, 다중 접

속을 위하여 스레드를 이용하여 소켓을 생성한다. 로봇 시스템은 USB 카메라로부터 영상을 받아서 계속 압축하고 있고 서버소켓을 생성하여 클라이언트와의 접속을 위해 포트를 열어두고 클라이언트가 접속할 때 까지 기다린다. 만일 클라이언트가 접속을 하면 접속한 클라이언트에 대한 스레드를 생성하여 압축한 영상 데이터를 클라이언트로 전송한다. 로봇 제어 명령 프로그램은 클라이언트에서 로봇에게 제어 명령을 보내면 소켓을 통하여 로봇 시스템으로 명령이 전달되고 이 명령은 다시 시리얼 포트를 통하여 로봇에 전달되어 모터를 제어한다.

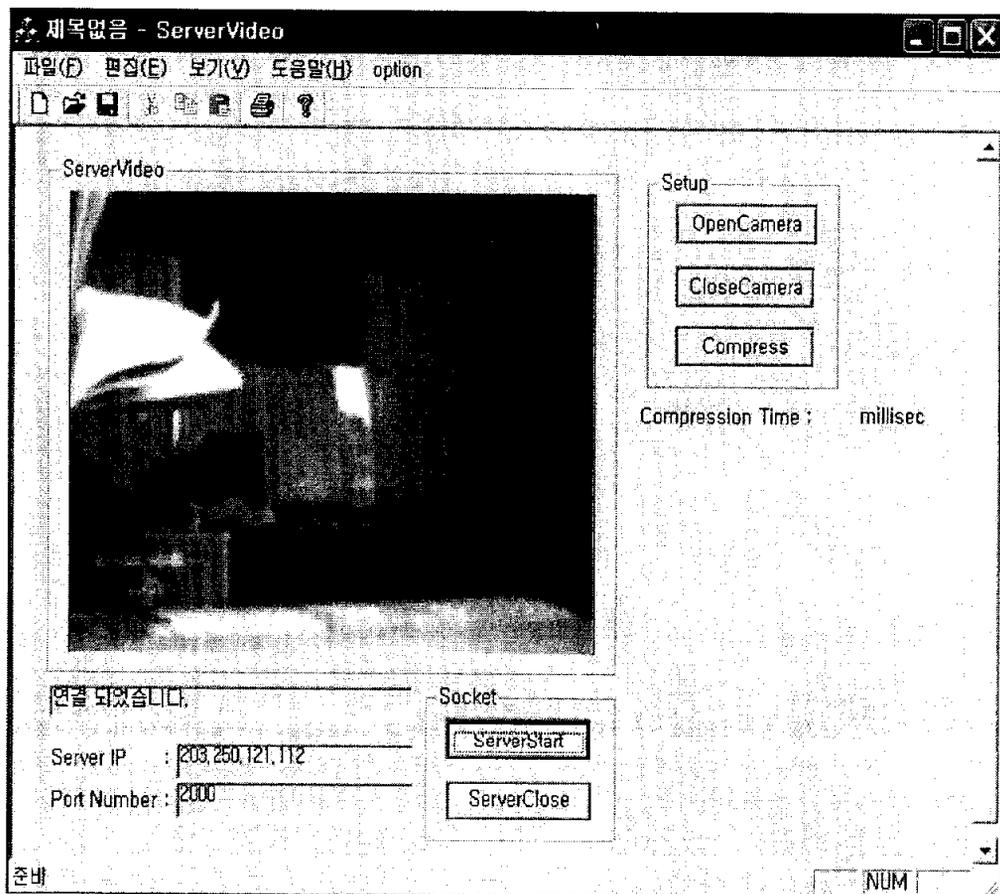


그림 3. 로봇 시스템

## 2.3 클라이언트 시스템

클라이언트는 사용자와의 직접적인 대화가 일어나는 곳으로 서버로부터 영상정보를 받아 화면에 보여주고, 사용자의 로봇 제어 입력을 받아 들여서 서버로 보내 준다. 클라이언트 시스템은 클라이언트 소켓 생성, 명령어 처리 부, 영상 데이터의 디코더 및 디스플레이로 구분되어지며 원격지의 로봇으로부터 전송된 영상정보와 데이터를 운영자가 획득함으로써 원격지의 환경정보를 얻을 수 있다. 생성된 클라이언트 소켓을 통하여 서버로 명령을 전송 최종적으로 원격 지의 로봇에 명령을 내릴 수 있게 구현하였다. 다수의 클라이언트들이 서버로 접속할 경우 로봇 시스템의 제어권한은 접속자중 한 사람만이 가질수 있도록 하였다. 아래 그림 4.와 그림 5.는 클라이언트 시스템과 프로그램 개략도이다.

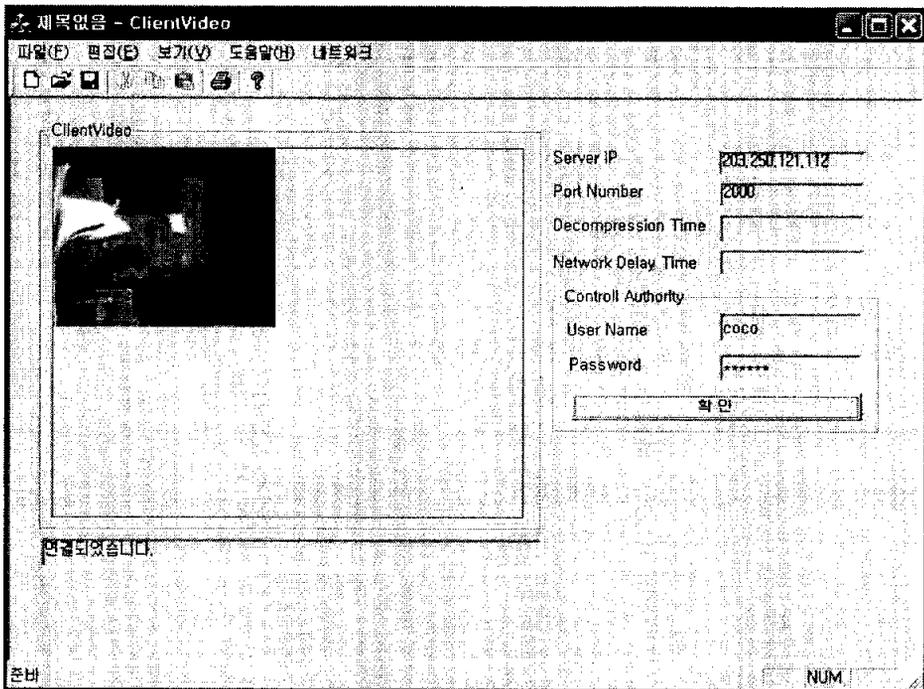


그림 4. 클라이언트 시스템

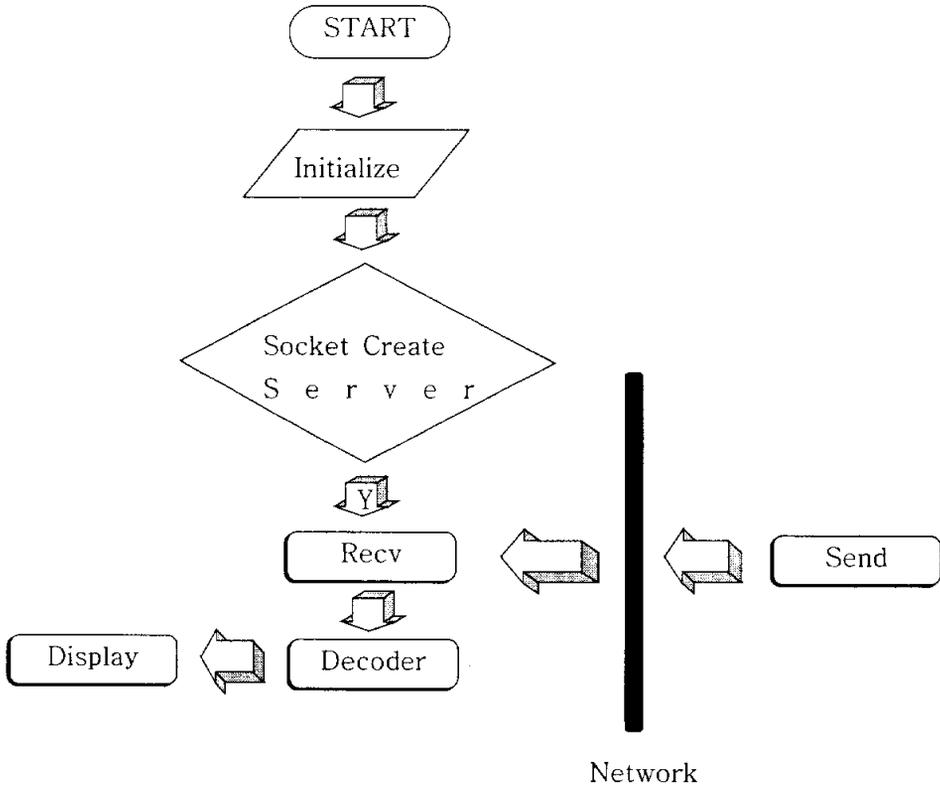


그림 5. 클라이언트 프로그램 개략도

### III. 영상 압축

영상이 320 x 240인 영상데이터의 한 프레임의 용량은 1 pixel이 3 byte 일 경우 3 x 320 x 240 = 225Kbyte이다. 동영상은 이런 프레임들의 연속이므로 상당히 많은 데이터를 필요로 한다. 이런 동영상들은 용량이 너무 커서 현재의 인터넷 속도로는 실시간으로 영상을 전송을 하기가 힘들다. 그래서 일반적으로 영상을 보낼때는 영상을 압축해서 보내게 된다.

영상을 압축하는 방법에는 원래의 데이터로 복원이 불가능한 손실 압축과 원래의 데이터로 복원이 가능한 무손실 압축이 있다. 영상의 무손실 압축은 데이터의 손실을 허용하지 않는 경우에 쓰이고 보통 손실 압축을 이용한다.

원격 운용 시스템을 실시간으로 제어하기 위해서는 USB카메라로부터 받은 영상 데이터를 빠른 시간안에 압축하는 것이 중요하다. 또한 압축한 영상데이터를 인터넷을 통하여 클라이언트로 전송되어야하므로 영상데이터의 용량이 작을 수록 유리하다.

영상 압축 알고리즘에는 대표적인 것이 MPEG 알고리즘과 H.261 알고리즘이 있는데 H.261은 초 저속망에서의 영상회의나 영상 회의를 목적으로 만들어졌기 때문에 압축률이 높고 영상의 실시간 전송이 가능하다. 그래서 본 논문에서는 초 저속망에서 사용되어지는 H.261에서 성능이 더 향상된 H.263 압축 알고리즘을 사용하였다.

#### 3.1 H.263의 개요

H.263은 저 전송률을 가지는 통신 선로(64K)이하에서 영상 회의나 비디오 전화등을 위한 멀티미디어 통신 서비스의 동영상 부분에 대한 압축에 쓰이는 국제 표준 알고리즘이며 H.261에 기반을 두고 있다[7][8].

동영상을 압축하는 기본적인 방법은 다음과 같다

- 1) 매크로 블록(16 x 16 pixel) 휘도 신호에 대해 움직임보상에 대한 프레임간 예측을 행한다.
- 2) 매크로 블록을 8 x 8 pixel의 블록으로 세분화하고 DCT에 의한 공간적 정보압축을 행한다. DCT연산의 대상은 휘도신호 블록 4개와 색차신호 블록 2개이다. 단 색 정보에 대해서는 인간의 시감도가 휘도에 비해 민감하지 않기 때문에 수평,수직 방향모두 반으로 줄인다.
- 3) 움직임 보상 프레임간 예측과 DCT에 의한 부호화정보의 발생확률의 편중을 이용하여 허프만 부호에 기초한 가변장부호화(엔트로피 부호화)를 행한다. DCT계수의 양자화제어에 의해 전체부호발생량을 제어한다[9].

### 3.2 H.263 압축 방식

표 1. H.263 picture format

Picture Format	lum. pixel (dx)	lum. pixel (dy)	chrom. pixel (dx/2)	chrom. pixel (dy/2)
sub-QCIF	128	96	64	48
QCIF	176	144	88	72
CIF	352	288	176	144
4CIF	707	576	352	288
16CIF	1408	1152	704	576

H.263은 화면의 크기에 따라 표 1과 같이 5개의 표준 picture format을 지원하고 있다. 이 format에 맞추어서 영상 신호를 YCbCr로 변환 할 경우 Y값은 밝기(luminance)를 나타내는 값으로 사람의 눈이 그 변화를 잘 인식한다. 하지만,CbCr값은 색차정보를 담고 있는데 사람의 눈은 그 변화를 잘 인식하지 못한다. 그래서 그림 6와 같이 4개의 휘도신호당 1개의 색차신호를 저장하고 나머지 색차신호는 버려

도 사람의 눈은 거의 차이를 느끼지 못한다[10].

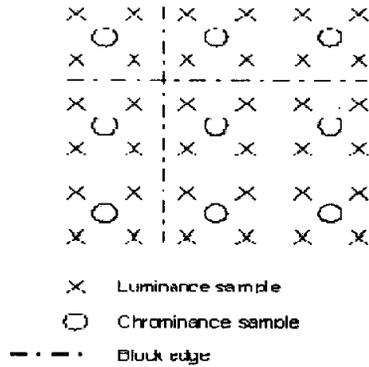


그림 6. 휘도와 색차의 위치

H.263의 비디오 스트림 구조는 그림 7과 같이 Picture, GOB(Group of Blocks), MB(Macroblock), Block 4개의 계층 구조로 되어 있다[7][9].

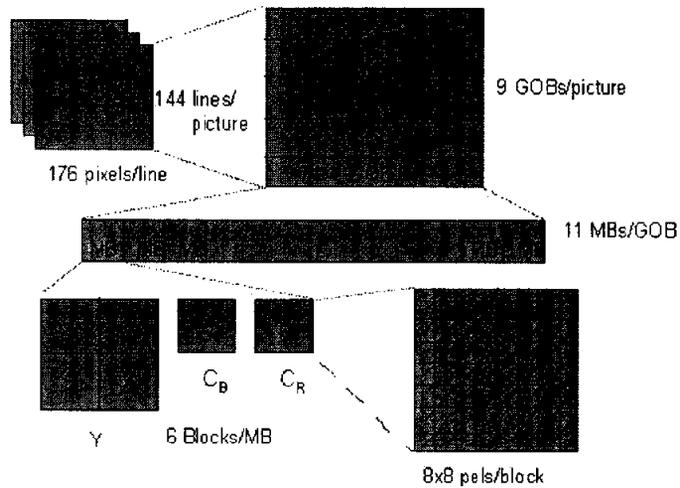


그림 7. H.263 비디오 스트림 구조

**\*픽처층**

픽처층은 한 화면분의 화면 데이터로 각 픽처의 데이터는 픽처 헤더와 그 뒤를 잇는 GOB 데이터로 구성되어 있다.

**\*GOB층**

GOB는 한 픽처를 CIF일 때는 12분할, QCIF일때는 9분할한 화면 데이터로 각 GOB의 데이터는 GOB헤더와 그뒤에 이어지는 MB데이터로 이루어진다.

**\*MB층**

1개의 GOB에는 11개의 MB가 있다. 1개의 MB에는 4개의 Y값과 1개의 Cb값, 1개의 Cr값으로 구성된다. 각각의 YCbCr은 8 x 8 pixel의 Block으로 되어 있다.

**\*Block층**

각 블록에는 변환 계수와 종료부호(EOB)로 구성되어 있는데 양자화된 변환계수는 저주파성분으로부터 이차원적으로 고주파쪽으로 지그재그 주사되어 일차원 데이터로 나열된다.

일반적으로 고주파성분은 거의 0으로 되기 때문에 EOB를 사용함으로써 부호화효율을 높힐수 있다.

영상 압축 과정은 현재 이미지를 MB(Macro Block : 16 x 16 pixel)단위로 나누어서 이전 프레임의 MB와 가장 일치되는 MB를 찾고 난 다음 두 MB의 차를 block(8 x 8 pixel) 단위로 나누고 DCT(Discrete Cosine Transform)을 한다. DCT를 적용하게 되면 저주파성분에는 많은 정보를 가지게 되고 고주파 성분에는 적은 양의 정보를 가지게 된다. 그후 양자화 과정을 거치면 저주파 성분의 DCT계수는 조밀하게 양자화하고, 고주파 성분의 DCT계수는 넓게 양자화 하면서 전체적으로 영상의 손실을 최소화하면서 데이터량을 감소시킨다. 그리고 VLC(Variable Length Coding)을 사용

하여 부호값의 발생확률에 따라 Huffman coding을 수행하여 VLC테이블을 만들어 그 테이블에 따라 coding하여 압축한다[7]. 그림 8은 H.263의 Encoder에 대한 블록도를 나타낸 것이다[7].

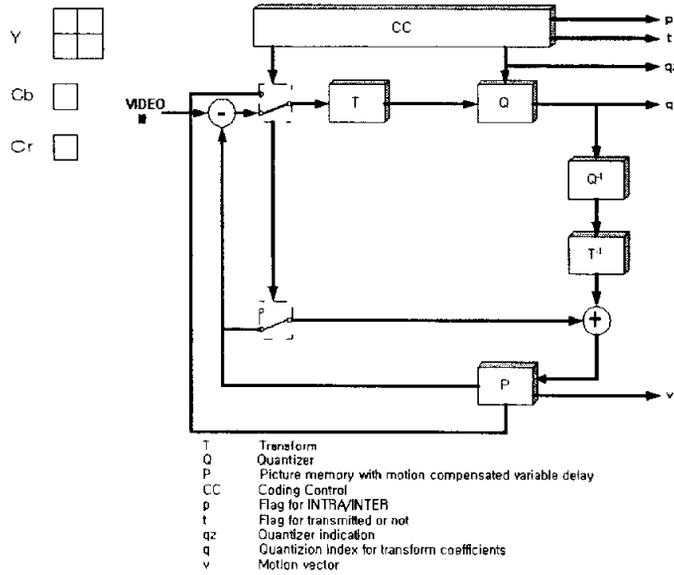


그림 8. H.263 Encoder 블록도

## IV. 하드웨어 구성과 제어기

### 4.1 로봇의 구성

로봇 시스템은 영상데이터를 압축하고 네트워크를 통해서 전송하는 프로그램부분하고 로봇을 구성하는 하드웨어 부분으로 나누어 진다. 로봇의 하드웨어 부분은 메인 보드, 무선 랜(Wireless Lan), USB 카메라, DC모터, DC모터 제어 보드[11], 배터리, DC-DC컨버터등으로 구성된다.

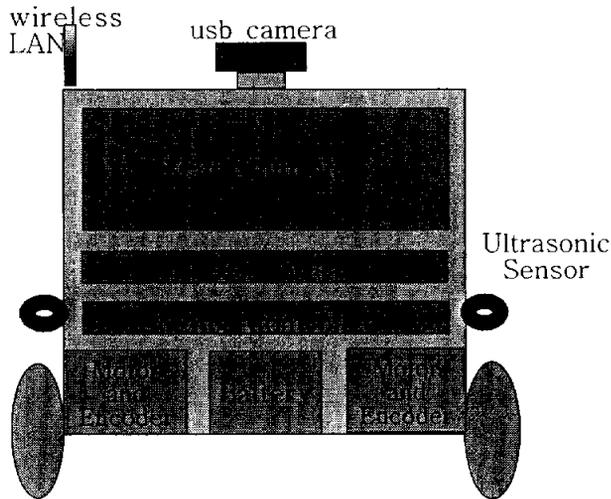


그림 9. 탐사로봇 시스템의 구조

탐사 로봇의 구조는 그림 9과 같이 PC기반으로 설계되었고 전체적인 사양은 다음과 같다.

로봇의 크기.

30cm x 20cm x 18cm

메인 컨트롤러

CPU : 셀러론 850Mhz

Memory : 256Mhz

모터부 ‘

MCU : TMS320LF2407A DSP Controller

7.2V Maxon DC Motor 4EA

L298 DC Motor Driver

16(P/R) Encoder

전원부

1.2V 2000mAh AAA size Battery 20EA

DC-DC Converter 4EA

기타

USB Wireless Lan Card

USB Camera

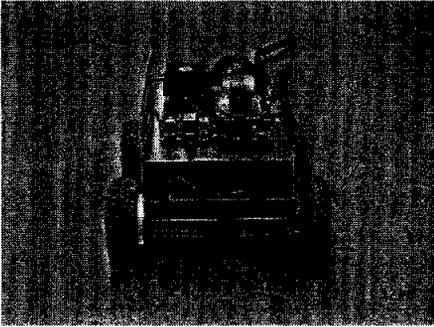


그림 10.1 탐사로봇(전체구조)



그림 10.2 탐사로봇(모터 제어부)

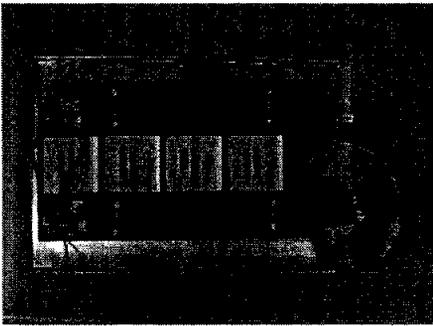


그림 10.3 탐사로봇(전원부)

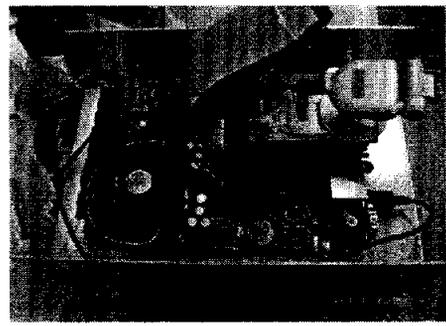


그림 10.4 탐사로봇(메인보드)

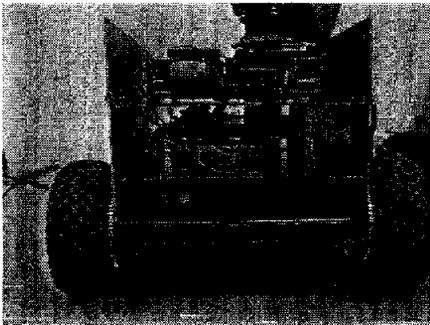


그림 10.5 탐사로봇(앞면)

그림 10.1의 그림이 실제 로봇의 전체 모습이다. 로봇의 하단부에는 모터와 모터를 제어하는 부분으로 구성되어 있다. 모터를 제어하는데는 PID 알고리즘을 사용하였다. 4개의 모터를 각각 실시간으로 제어하기 위해서 고속의 DSP Controller[11]를 사용하였다. 그림 10.3은 전원부를 나타낸 것이다. 이는 메인보드와 모터에 전원을 공급하기 위한 부분이다. 이전에는 Battery에서 DC / AC Inverter를 거쳐서 220V 전원을 가져와서 컴퓨터의 파워서플라이에 연결하여 메인보드에 전원을 공급해주었는데 이런 방법을 사용하면 전원부의 부피가 커지고 무게도 늘어나게 되어 모터도 힘이 좋은 더큰 모터를 사용하게 되어 전체적인 로봇의 크기가 커진다. 로봇의 크기를 줄이기 위해서는 전원부 부분을 소형화 해야한다. 그래서 소형 건전지를 직,병렬로 연결해서 12V의 전압과 충분한 전류를 만들고 거기서 메인보드 전원부분에 필요한 각각의 전압을 DC-DC Converter를 통해서 전원을 공급하게 하여 로봇의 부피와 무게를 줄였다. 맨 윗부분은 메인보드가 장착되고 거기에 USB 카메라와 무선 랜이 붙여지게 된다.

## 4.2 DC 모터 제어기

본 논문에서는 DC 모터제어를 위해 PID제어기를 사용하였다. PID제어를 하기 위해서는 각각의 P,I,D gain을 구해야 한다. 이들 gain을 구하기 위해서는 로봇의 정확한 사양을 알아야하는데 모를 경우에는 실험적인 방법을 통하여 gain을 구하거나 Ziegler-Nichols step Response Method[12]등을 이용하여 제어기의 gain을 구할 수 있다.

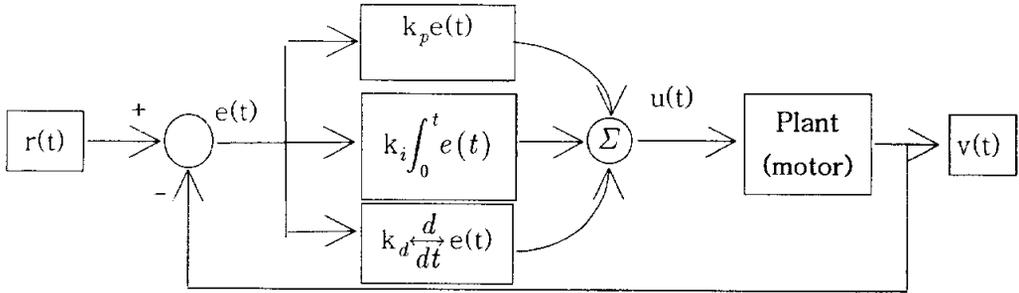


그림 11. PID 제어기

로봇의 속도는 아래와 같이 나타낼 수 있다.

$$V = \frac{2\pi r}{16t} \times \text{counter}$$

여기서  $V$ 는 모터 속도이고  $r$ 은 바퀴의 반지름  $\text{counter}$ 는 한주기동안 들어온 엔코더의 값이다. 그리고  $t$  샘플링 시간이다. 여기서 사용한 엔코더는 바퀴가 한번 돌때 16개의 펄스가 나온다. 바퀴의 둘레길에서 16을 나누어주면 엔코더의 한펄스당 바퀴가 움직인 거리가 나온다. 거기에서  $\text{counter}$ 값을 곱해주면 움직인 거리가 나오고 샘플링 시간을 나누어주면 모터의 속도를 구할 수가 있게 되고 이를 이용하여 모터의 속도 제어를 할수 있다.

$$e(kT) = V(kT) - V((k-1)T)$$

여기서  $e(kT)$ 는 속도 오차이고  $V(kT)$ 는 현재 속도이고  $V((k-1)T)$ 는 이전 속도이다.

이산 시간에서 비례기와 미분기와 적분기를 수식으로 나타내면 다음과 같다

$$P = K_p \times e(kT)$$

$$I = K_i \times T \sum_{k=0}^n e(kT)$$

$$D = K_d \times \frac{e(kT) - e((k-1)T)}{T_s}$$

여기서  $K_p$ 는 비례이득,  $K_i$ 는 적분이득,  $K_d$ 는 미분이득을 나타낸다. PID 제어기의 최종출력은 위의 세 식을 조합함으로써 나타내어진다.

$$u(kt) = K_p \times e(kT) + K_i \times T \sum_{k=0}^n e(kT) + K_d \times \frac{e(kT) - e((k-1)T)}{T_s}$$

## V. 실험 및 검토

### 5.1 H.263으로 영상압축과 전송

176 x 144 크기의 원영상은 크기가 38016byte이고 압축시 영상의 크기가 200byte에서 2000byte정도의 크기로 압축이 되었다. 1/20에서 1/200정도로 높은 압축률을 볼 수가 있었다. 압축하는데 소요되는 평균적인 시간은 300ms정도 걸렸고 네트워크를 통해서 클라이언트로 전송하는 데는 13ms, 클라이언트에서 압축된 영상을 디코더하여 디스플레이하는데까지는 16ms정도 걸렸다. 그림 11.은 압축하는 시간과 네트워크 지연시간, 그리고 디코더에서 디스플레이까지 걸리는 시간을 비교해놓은 것이다.

h.263 decoder.txt ...	h.263 encoder.txt ...	network delay.txt ...
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)	파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)	파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
16msec	277msec	10msec
14msec	265msec	10msec
16msec	282msec	23msec
16msec	298msec	09msec
15msec	354msec	14msec
27msec	324msec	25msec
16msec	291msec	16msec
16msec	293msec	13msec
16msec	331msec	12msec
15msec	285msec	12msec
16msec	317msec	25msec
42msec	390msec	21msec
43msec	364msec	10msec
16msec	305msec	12msec
24msec	067msec	13msec
21msec	294msec	13msec
16msec	279msec	15msec

그림 11. 인코더 시간, 디코더 시간과 네트워크 지연 시간 비교

위의 자료를 보면 데이터를 압축하는데 시간이 많이 걸림을 알 수 있다. 이는 압축

을 소프트웨어 적으로 하여서 컴퓨터의 성능에 따라 좌우된다. 실험은 연구실 안에서 학교의 빠른 네트워크를 이용하여 네트워크 속도가 빠른 시간에 실험을 하였다. 위의 자료를 보면 압축하는데 시간이 다소 많이 걸리고 압축된 영상을 네트워크를 통해서 전송하는 것과 클라이언트에서 받은 압축영상을 디코더하여 디스플레이하는데는 시간이 작게 걸림을 알 수 있었다..



(A) 원 영상



(B) 압축 영상

그림 12. 원 영상과 압축 영상의 비교

그림 12는 원 영상과 클라이언트에서 압축을 디코딩해서 디스플레이한 영상이다.

## 5.2 Time Delay

인터넷은 다양한 분야에서 이용하는데 많은 이점이 있지만, 인터넷을 사용하여 로봇의 원격운용을 하는데는 몇가지 문제가 있을 수 있다. 데이터 전송시 time delay 문제, 데이터 손실 문제, 전송되는 데이터의 보안 문제등이 해당될수 있다.

본 논문에서는 데이터 손실 문제, 전송되는 데이터의 보안 문제 보다는 time delay 문제가 더 중요하다.

Computer Network상에서 통신은 일정하지 않은 time delay를 가진다. 그래서 현재 보내는 영상과 원격지에서 모니터링하는 영상 사이에 시간 차이가 발생할 수 있다. 특히 대륙간의 아주 먼거리에서 데이터를 전송하는데는 상당한 시간 차이가 발생할 수 있고 이로인해 탐사 로봇의 제어가 힘들어질 수 있다.

time delay로 인한 제어의 어려움을 극복하기 위해 여러 가지 제어 알고리즘 [13][14]들이 있다. 여기서는 time stamp를 찍어서 영상과 같이 보내는 방법을 제안한다.

압축 영상을 전송할 때 현재 시간을 같이 보내게 되면 원격지에서 모니터링하는 영상이 얼마나 시간이 지연되었는지 알 수 있고, 이를 감안하여 제어를 할 수 있다.

매번 데이터를 주고 받으면서 time delay를 계산해서 전송하면 네트워크의 부하가 많이 걸리고 또한 보내는 동안의 time delay는 모른다. 그래서 네트워크의 부하를 줄이는 방법으로 네트워크의 time delay는 단시간 안에 잘 변하지 않는다는 가정하에 처음 클라이언트 접속시 여러번 데이터를 주고 받아 평균 time delay를 계산해서 현재 시간과 평균 time delay 시간을 같이 보낸다. 이렇게 함으로써 원격지에서는 현재 시간과 로봇의 시간을 알 수 있다. 이후로는 로봇의 현재 시간만 영상 정보와 함께 보

내게 되면 원격지에서는 두시간차이를 알 수 가 있게 된다.

### 5.3 검토

위의 실험에서 영상의 압축률이 20배에서 200배까지 차이가 나는 이유는 영상의 움직임이 없을 경우에는 압축률이 높아지고 영상의 움직임이 많을 경우에는 압축률이 낮아지기 때문이다. 영상을 압축하는데는 다소 시간이 많이 걸리는데 이는 소프트웨어적으로 영상압축을 하였기 때문에 이는 컴퓨터의 성능에 따라서 달라질 수 있다. 또한 프로그램이 최적화가 되면 시간을 줄일 수 있다. 영상압축을 소프트웨어적이지 아니라 하드웨어적으로 하면 압축 시간을 상당히 줄일 수 있을것으로 생각된다. 영상을 압축하는데 시간이 다소 거리지만 압축률이 좋기 때

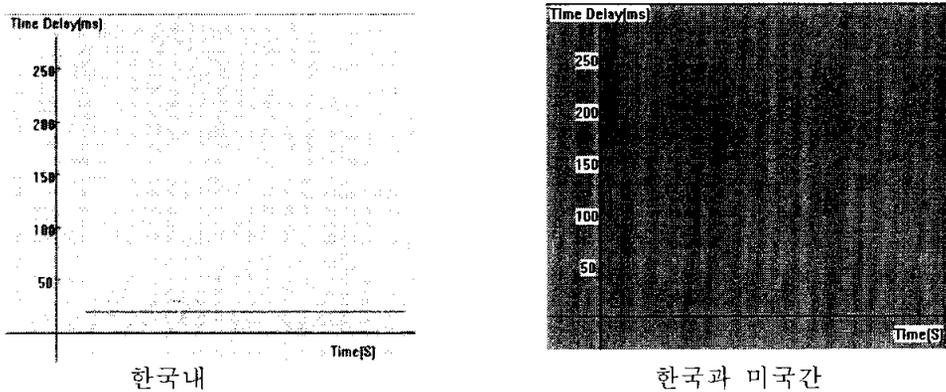


그림 13. Time Delay 측정

문에 압축된 파일의 크기는 상당히 작다. 실험에서도 알 수있듯이 압축된 파일이 작아서 네트워크의 지연시간이 상당히 적게 생겼고 영상을 디코더하고 디스플레이하는데도 만족할만한 성능을 나타내었다.

그림 13은 네트워크상에서의 time delay를 측정한 것이다. 왼쪽그림은 국내에서의 데이터전송시 delay이고, 오른쪽 그림은 한국과 미국간에 데이터 전송시 delay이다. 위의 두그림을 보면 짧은 시간안에는 time delay가 거의 일정하다는 것을 알 수 있고, 앞에서 말한 가정이 맞다는 것을 알 수 있다.

## VI. 결론

본 논문에서는 인터넷의 급속한 발전과 로봇에 대한 관심의 증가로 많은 관심을 가지고 있는 원격 운용 시스템을 H.263 비디오 코덱 알고리즘을 이용하여 실제로 구현하였으며 인터넷을 이용하여 원격지의 로봇을 운용하였다. 인터넷을 이용하여 로봇을 보다 능동적으로 운용할 수 있었고, 무선 랜을 이용하여 공간적인 한계를 극복할 수 있었다. 또한 로봇을 소형화하여 탐사 로봇의 목적에 맞게 인간이 갈 수 없는 곳이나 미지의 환경에서 보다 효율적으로 사용할 수 있게 하였다. H.263을 이용한 영상 압축 알고리즘을 사용하여 사용자에게 원격지의 정보를 실시간으로 공유할 수 있게 하였다.

앞으로 개선해야할점은 영상을 압축 시간을 줄여야한다. 이는 프로그램의 최적화를 하거나 하드웨어로 구현함으로써 가능하다. 그리고 이러한 시스템은 네트워크상에서 예측할 수 없는 지연이 발생할 수 있음을 항상 인지해야한다. 이 시스템을 더욱 발전시키기 위해서는 장애물 회피, 탐사 로봇의 위치 인식에 대한 연구가 더 필요하다고 생각된다.

## 참 고 문 헌

- [1] T. Takahashi and S. Tadokoro, "Working with Robots in Disasters,"  
IEEE Robotics and Automation Magazine, September 2002
- [2] Dirk Schulz, Wolfram Burgard, Dieter Fox, Sebastian Thrun, Armin B. Cremers," Web Interfaces for Mobile Robots in Public Places ", IEEE Robotics & Automation Magazine, March 2000
- [3] K.A. Golberg, "Telerobot garden on the World Wide Web", Rob. Mach. Perception, vol. 5, no. 1, no. 4, 1996
- [4] A. Bejczy, G. Bekey, R. Taylor, and S. Rovetta, "A research methodology for tele-surgery with time delay", in Proc. First Int. Sym. Medical Robotics and Computer Assisted Surgery, Sept. 1994
- [5] K.S. Tso, P.G. Backes, and G.K. Tharp, "Mars pathfinder mission internet-based operations using wits", in Proc. IEEE Int. Conf. Robotics and Automation(ICRA), Leuven, Belgium, 1998
- [6] M.R. Stein, "Painting on the world wide web: The pumapaint project", Proc. of IEEE Int. Conf. on Intelligent Robots and Systems(IROS): Workshop on Web Robots, Victoria, Canada, 1998
- [7] DRAFT ITU-T Recommendation H.263, "Video Codec for Low Bit Rate Communication," 2 May, 1996
- [8] ITU-T SG 15, Working Party 15/1 Expert's Group on Very Low Bitrate Visual Telephony, "Video Codec Test Model TMN5," Jun 1995

- [9] 정제창 번역, “ 최신 MPEG”, (주) 교보문고, 1995
- [10] Rafael C. Gonzalez, Richard E. Wood, “Digital Image Processing”, Prentice Hall Publishing Company, September 1993.
- [11] Texas Instruments, TMS320LF/LC240xA DSP Controller System and Peripherals Reference Guide.
- [12] Karl Johan Astrom, Tore Hagglund, “Automatic Tunning of PID Controllers”, Instrument Society of America.
- [13] ROBERT J. ANDERSON, MARK W. SPONG “Bilateral Control of Teleoperators with Time Delay” IEEE Transactions on Automatic Control. vol. 34, no. 5, May, 1989.
- [14] REN C. LUO, FELLOW, IEEE, KUO L. SU, SHEN H. SHEN, AND KUO H. TSAI “Networked Intelligent Robots Through the Internet” Proceedings of the IEEE, vol. 91, no 3, March, 2003.