
웹 카메라 기반 홈 서버용 모니터링 기술에 관한 연구

정종근* · 천승환**

A Study on Monitoring Technique for Home Server based on Web Camera

Jong-Geun Jeong , Sung-Hwan Cheon

이 논문은 2004년도 한국학술진흥재단의 지원에 의하여 연구되었음(KRF-2004-042-D00185)

요 약

사생활 보호에 대한 요구가 증가되고 있는 때 보안을 위한 모니터링 시스템의 요구는 날로 증가되고 있다. 본 논문에서는 인터넷을 통해서 PC나 PDA로 모니터링 할 수 있는 기술을 연구하였다. 마이크가 내장된 USB 카메라로 촬영한 영상과 음성 데이터를 H.263의 Multiplex를 통해서 인터넷을 통해 PC나 PDA로 전송한다. 그리고 웹 카메라 기반 홈서버용 모니터링 시스템 각 장치들의 동작 처리부를 일관된 구동 모듈로 통일하고, 관리자에게 일관된 구동 모듈을 제공하여 시스템의 운용과 관리에 있어서 편리함을 제공하였다.

ABSTRACT

A monitoring system for the security is increasingly demanded in line with the issue of the infringement of privacy. This paper has conducted the study on the monitoring technique via the Internet using PC or PDA. This is a monitoring system for web-camera based home server, which transmits the images and sound data captured by mike-embedded USB camera, through multiplexer of H.263 and internet. It integrates the operating processors of each device into a unified drive module and provides it the operator to facilitate the system operation and maintenance.

키워드

PDA, USB 카메라, H263 codec, 뷰어

I. 서 론

최근 들어 사회적 범죄의 증가로 인해 CCTV와 같은 영상 감시 장치들이 많이 출시되고 있으며 이와 관련된 보안 분야의 사업도 크게 증가하고 있는 추세이다. 하지만 CCTV 시스템은 근거리에서는 사용이

가능하지만 원거리에서는 이용하기가 불편하다. 따라서 본 연구에서는 마이크가 내장된 USB 카메라로 촬영한 영상과 음성 데이터를 인터넷을 통해 PC나 PDA로 전송하는 웹 카메라 기반 홈 서버용 모니터링 시스템을 구현하였다. 본 연구에서는 모니터링 시스템을 서버와 클라이언트로 구분하였다. 카메라 장치 구

* 호남대학교 컴퓨터 공학과
** (주)도올정보기술

동 모듈은 각 가정이나 사무실에 설치되어 서로 다른 카메라의 장비마다 서로 다른 장치 드라이버가 존재하고 있다. 따라서 서로 다른 웹 카메라의 동일한 관리 및 개발을 위해서는 일관된 장치 구동 모듈이 필요하다. 이를 위하여 웹 카메라 장치제어 처리부, 메시지 처리부, 레지스트리 처리부, 화면출력 처리부 등의 다양한 구동 모듈들이 필요하다. 본 웹 카메라 장치 구동 모듈은 서로 상이한 웹 카메라 장치들 간의 동작 처리부를 일관된 장치 구동 모듈로 통일하고, 관리자에게 일관된 구동 모듈을 제공함으로써 시스템 운용 및 관리의 편리성에 목적이 있다. 모니터링 시스템은 자신의 Home GateWay에 설치된 카메라를 사용하여 인터넷을 통해 외부에서 집안을 모니터링 할 수 있는 시스템으로 HG와 PC 상에서 수행되어야 하며, 각각 서로 다른 OS(WinCE, Windows)에서 동작할 수 있도록 개별적으로 프로그램이 작성되어야 하며 개발되는 홈 뷰어 프로그램은 타 하드웨어(HG)에 쉽게 포팅 될 수 있도록 하드웨어 의존적인 부분을 모듈화 하여, 포팅 시 변화를 최소화해야 해야한다. 따라서 본 논문에서는 웹카메라를 기반으로 하여 전송되는 영상과 음성을 PC나 모바일 폰에서 모니터링 할 수 있는 기술에 대해 연구한다.

II. 웹 카메라 기반 홈 서버용 모니터링 시스템 설계

2.1 클라이언트

모니터링 시스템의 클라이언트에서는 서버에 연결되어 있는 카메라와 인터넷이나 무선으로 메시지와 데이터를 교환한다. Socket 초기화 부분에서는 메시지와 비디오, 사운드 등의 Socket을 제일 먼저 초기화하여 서버와 연결이 되었을 때 서버로부터 데이터를 전송받아서 클라이언트의 창에 비디오와 사운드를 출력한다 [3]. 클라이언트 부분의 Socket들이 초기화되고 카메라에 연결이 확립되었을 때 서버와도 데이터를 교환한다. 클라이언트에서 처리 가능한 수준의 비디오와 사운드를 받을 수 있도록 Buffer size에 따라 서버에 보내는 요청메시지의 빈도를 결정하여 처리한다.

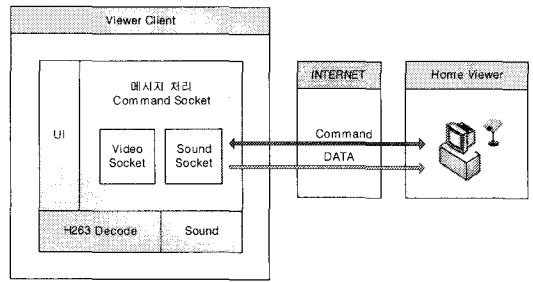


그림 1.클라이언트 시스템 구성도
Fig 1. Structure of Client system

2.2 서버

서버 부분은 클라이언트와 연결되어 명령어와 데이터를 주고받으며, 서버의 Socket들을 통해 클라이언트에 사운드와 비디오를 전송한다. UI에서는 환경설정이나 클라이언트 접속기록을 담당하고, Shell 연동 부분에서는 메인 메시지 처리를 위해 Shell 연동 기능을 수행한다. 그리고 DVR(Digital Video Recorder)에 연결된 카메라의 정보를 가져와 캠(CAM)을 초기화하고 사용자는 연결된 카메라 중에서 원하는 카메라를 선택하여 구동한다. 움직임 감지 기능을 이용하여 움직임이 발생하면 자동으로 캠이 추적한다. 그리고 TCP/IP 통신으로 접속되는 다수의 클라이언트와 연동하며 각각의 클라이언트는 각각의 소켓으로 독립적으로 관리된다.

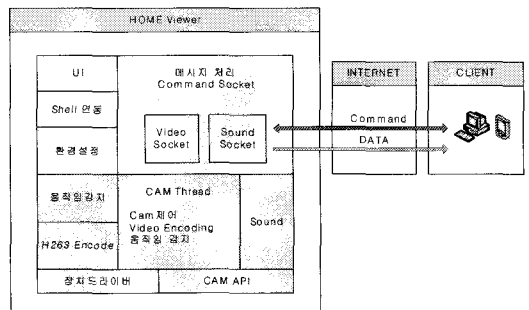


그림 2. 서버 시스템 구성도
Fig 2. Structure of Server System

다음은 홈 뷰어 서버를 구성하는 중요 모듈에 대한 설명이다.

2.2.1 H.263 Codec

H263 Codec은 MPEG-4와 견줄 수 있는 품질과 압축

를을 보장하는 고 품질 Lossy Interframe Codec으로 비디오 컨퍼런스 또는 큰 파일들의 압축시에 사용될 수 있으며, 카메라로부터 캡처된 비디오 및 스크린으로부터 캡처된 비디오들에 적용될 수 있다. 또한 카메라로부터 이미지를 가져와서 네트워크 전송을 위한 인코딩과 클라이언트에서의 디코딩을 위해서도 사용된다. H.263의 비디오 Codec의 구성은 다음과 같다. a) Video Signal이 입력되었을 시 Source Coder에서 입력을 받아 Multiplex에서 Encoding을 수행하고 Transmission Buffer에서 Streaming을 하기 위해 대기시킨다. Coding Control은 Video Source에 대해서 Encoding에 필요한 제어를 하는 부분이다. b) 는 전송된 Encoding된 Video Streaming이 Client에 전송되었을 시 multiplex에서 Decoding이 수행하고 원 영상으로 비디오를 출력하는 역할을 한다.

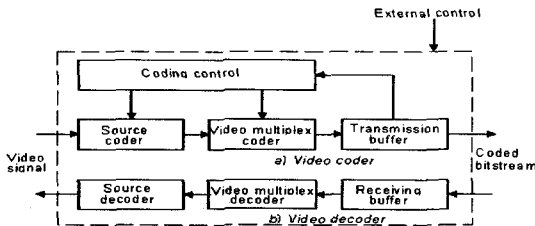


그림 3. H.263 비디오 코덱
Fig 3. H.263 Video Codec

2.2.2 CAM 제어

이 부분에서는 카메라를 제어하는 부분으로서 카메라의 방향을 조정하고 움직임 감지하여 전송하는 역할을 담당한다.

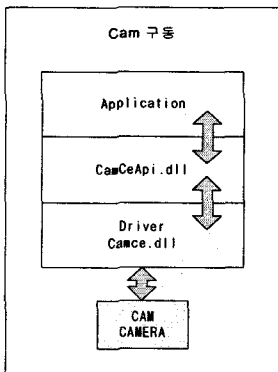


그림 4. CAM 제어
Fig 4. CAM Control

CAM을 제어하기 위하여 아래와 같은 API Method를 호출한다.

- GetActiveCam() : 현재 활성화된 Cam를 찾아낸다
- LoadLib : 캡제어에 필요한 Api Method를 로드한다.
- ControlOpen : Serial Port를 개방한다.
- GetProperty : 캠 환경변수를 얻는다
- SetProperty : 캠 환경변수를 설정한다
- ControlClose : Serial Port를 닫는다.
- Close : Cam작동을 종료한다
- Free : Method와 리소스를해제한다

2.2.3 영상 및 음성 전송 기능

캠으로부터 영상을 RGB24로 가져와서 보여준다 [5,6,7]. 클라이언트 접속시 캠으로부터의 영상을 H263으로 encoding하여 전송하고 음성을 초기화 한 후 클라이언트로 전송한다. 영상On/Off, 음성On/Off를 이용하여 영상, 음성 전송유무를 선택할 수 있다.

다음은 홈뷰어 서버에서의 Cam 영상 제어기능이다.

- VideoStart : 캠의 영상전송을 시작한다
- VideoStop : 캠의 영상전송을 종료한다
- GetRgb24 : 캠의 영상을 RGB24로 가져온다
- GetJpeg : 캠의 영상을 JPEG로 가져온다

III. 시스템 구현

웹 카메라 기반 모니터링 시스템은 자신의 집 HomeGateway에 설치된 카메라를 가지고 인터넷을 통하여 외부에서 집안을 모니터링 할수 있는 서비스로 HG와 PC 상에서 수행되어야 하며, 각각 서로 다른 OS(WinCE, Windows)에서 동작할 수 있도록 개별적으로 프로그램이 작성되어야 한다. 모니터링 서비스는 서버 프로그램과 클라이언트 프로그램으로 나누어서 작성하며 서버 프로그램은 다수의 클라이언트 접속을 관리하며 데이터를 전송하도록 구성되어진다.

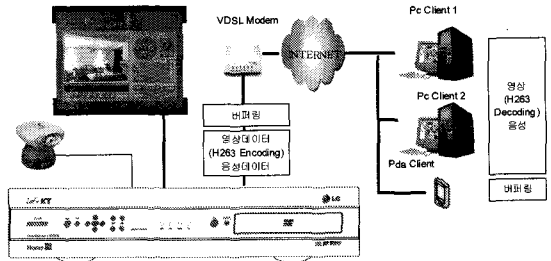
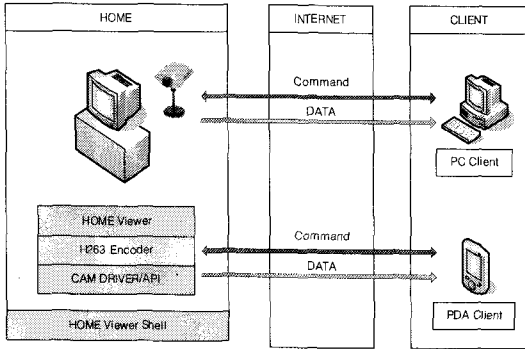


그림6. 시스템 전체 구성도
Fig.6. Structure of Total System

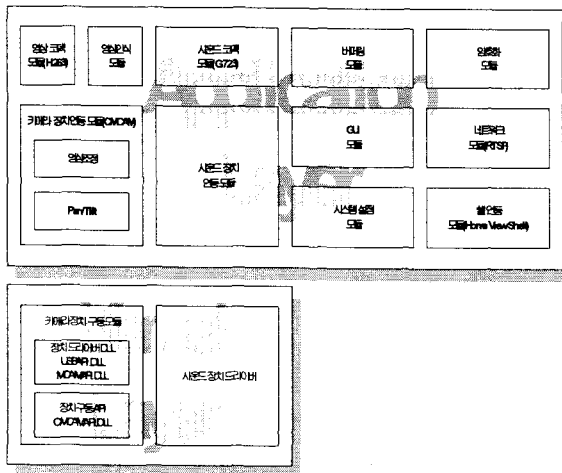


그림 5. 시스템 모듈구성도
Fig 5. System Module Structure

클라이언트는 Window Plattform과 WinCe Plattform으로 나누어져서 작성하며 서버에 접속하여 데이터를 전송 받아 재생한다[6,7].

서버와 클라이언트는 TCP/IP 기반으로 영상과 음성 전송하며, Home Gate Way에 접속된 Cam의 동영상을 H263코덱으로 인코딩한 후 쓰레드로 실행하여 클라이언트의 접속을 기다린다. 그리고, Home Gate Way의 음성 장치를 초기화하여 클라이언트 접속을 기다린다. 그리고 클라이언트 접속 시 영상을 클라이언트 영상소켓을 통하여 전송하고 클라이언트 접속 시 음성전송 쓰레드를 실행시켜 음성소켓을 통하여 전송한다. 클라이언트 Disconnect 시 영상소켓을 닫고 전송을 중지하고 클라이언트 Disconnect 시 음성소켓을 닫고 음성전송 쓰레드를 중지한다.

서버프로그램은 크게 아래와 같이 구성되어 있다.

Client 접속 관리

영상/음성 전송

Client의 메시지처리(카메라제어, 환경설정)

다음은 서버측의 역할로서 움직임이 감지된 결과를 e-mail로 발송하는 코드로서 등록된 사용자에게 지정된 SMTP서버에 접속하여 업로드한다. SMTP프로토콜을 구현한 CEMail 클래스에 서버와 메일내용을 설정하여 전송한다. 이 기능과 관련된 코드로는 클래스 CEMail의 CameraThread.cpp가 있다.

```
// CameraThread.cpp
UINT CCameraThread::DetectSendThread(LPVOID lpvParam)
{
    ...
    // Mail Set
    SYSTEMTIME st;
    GetLocalTime(&st);
    CEMailMessge Emsg;
    Emsg.Head.SetFrom("HomeViewer", "watcher@unitedpower.ne.jp");
    Emsg.Head.AddTo("evilblue", "evilblue@vitamincpp.net");
    Emsg.Head.AddTo("evilblue", "evilblue@gmail.com");
    Emsg.Head.SetSubject("HomeView Detect Notice");
    strncpy(Emsg.Head.XMailer, "View Watch", 256);
    Emsg.Head.Notification = FALSE;
    Emsg.Head.SetDate(st);
    strncpy(Emsg.Head.ContentType.Content, "text/plain", 256);
    strncpy(Emsg.Head.ContentType.charset, "utf-8", 256);
    CEMailBody* pBody = (CEMailBody*)malloc(sizeof(CEMailBody));
    pBody->Init();
    char* ptxt = "HomeView Detect Notice";
    pBody->SetData(ptxt, strlen(ptxt));
    Emsg.AddBody(pBody);
    // Mail Send
    m_email.SetHeloCmd("HELO");
    m_email.SetOption("211.111.136.197", 25);
    BOOL bSend = m_email.Send(Emsg);
    ...
}
```

클라이언트PC에서는 서버와 같이 TCP/IP 기반으로 영상과 음성을 재생한다. 클라이언트는 서비스 중인 Home Gate Way에 접속하여 영상과 음성데이터를 요청하고 전송받은 영상을 H263으로 인코딩하여 출력하며 전송받은 음성을 버퍼링하여 오디오장치를 통하여 출력한다. 클라이언트의 소켓은 커맨드소켓, 비디오소켓, 사운드소켓의 3가지로 이루어져 있다[8,9]. 접속 순서를 살펴보면 다음과 같다.

1. 서버의 IP를 입력하면 커맨드소켓으로 접속한다.
2. 커맨드소켓이 서버로부터 클라이언트의 인덱스를 받으면 비디오소켓과 사운드소켓을 접속한다.
3. 비디오소켓과 사운드소켓으로 받은 인덱스를 전송해서 소켓셋 306 속 절차를 완료한다.
4. 서버가 3개 소켓의 접속이 완료됨을 확인하면 로그인성공 메시지를 보내준다.
5. 클라이언트의 종류가 PC인지 PDA인지 서버로 알려준다.
6. 영상 데이터를 받는 경우 바로 H263코덱에 연결하여 화면에 재생한다.
7. 음성 데이터를 받는 경우 버퍼에 넣는다.
8. 스피커 장치에서는 버퍼에 데이터가 있는 경우 음성을 재생한다.
9. 소켓과 관련 라이브러리의 종료

다음은 클라이언트에서 서버로 접속하는 코드이다.

```

// TestTCPClientDlg.cpp

void CTestTCPClientDlg::OnButtonConnect()
{
    m_bFirstExit = TRUE;
    if (m_bConnected) {
        CloseProgram();
        return;
    }

    CConnectDlg dlg;
    dlg.m_strServerIP = m_strServerIP;
    if (dlg.DoModal() == IDOK) {
        SocketManager.InitManager(this);
        RecvInit();
        m_strServerIP = dlg.m_strServerIP;
        SocketManager.Connect(m_strServerIP);
        if (SocketManager.cs) SendMsgToMain("Init Cs", MSG_STRING);
        if (SocketManager.vs) SendMsgToMain("Init Vs", MSG_STRING);
        if (SocketManager.ss) SendMsgToMain("Init Ss", MSG_STRING);
    }
}
    
```

3.1 서버 기능별상세 작업 흐름도

3.1.1 Connect 통신 흐름도

클라이언트 접속 시 서버는 임의의 Index를 생성하여 클라이언트로 전송한다. Index를 받은 클라이언트는 Video, Audio 접속을 시작한다. 서버는 접속되는 클라이언트의 Index를 관별하여 Command, Video, Sound 소켓이 모두 접속되었는지를 검사하여 클라이언트에 "Login" 되었다고 알려주고 새로운 클라이언트 ID로 관리를 한다.

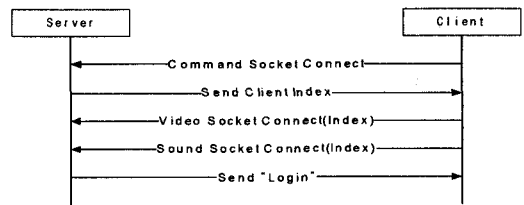


그림 7. 커넥트 통신 흐름도
Fig 7. Flow of Connect communication

3.1.2 Client Connect Process

클라이언트 관리자는 접속되는 모든 클라이언트의 정보를 리스트형태로 관리하며 각각의Command Socket, Video Socket, Sound Socket들도 접속된 클라이언트의 소켓을 리스트 형태로 관리한다.

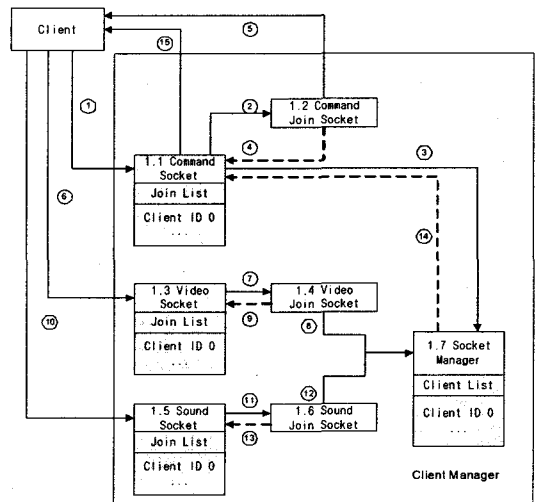


그림 8. 클라이언트 접속처리 프로세스
Fig 8. Client Connect Process

Command Socket 모듈에서는 새로운 클라이언트의 접속을 Listen하고 Client 접속 시 새로운 Command 조인 소켓을 생성하여 연결시키고 관리한다. Client 관리자에게 Client List Create 요구하고 Check Client의 Client 로그인을 정보를 Client(Index)에 전달한다. Video Socket 모듈에서는 Client Video Socket의 Index를 Listen하고 Client 접속 시 새로운 Video 조인소켓을 생성하여 연결시키고 관리한다. Sound Socket 모듈에서는 Client Sound Socket의 Index를 Listen하고 Client 접속 시 새로운 Sound 조인소켓을 생성하여 연결시키고 관리한다.

3.1.3 캠 제어 및 영상전송 Process

Cam Thread 모듈에서는 Cam을 연동과 Cam을 초기화하고 영상을 요청하는 기능과 함께 Cam으로부터의 영상을 인코딩하는 역할을 담당한다. Cam 모듈은 Cam을 제어하고 영상을 전송하는 역할을 담당하는데 Cam의 영상을 클라이언트로 Video Join Socket을 통하여 전송한다. Socket manager는 클라이언트의 메시지를 전송하고 클라이언트로 H263인코딩 영상을 전송한다.

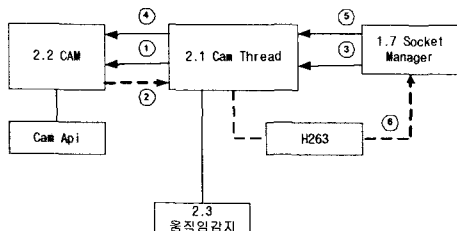


그림 9. 캠 제어 및 영상전송 처리
Fig 9. CAM Control and Video Transfer Process

3.1.4 음성전송 Process

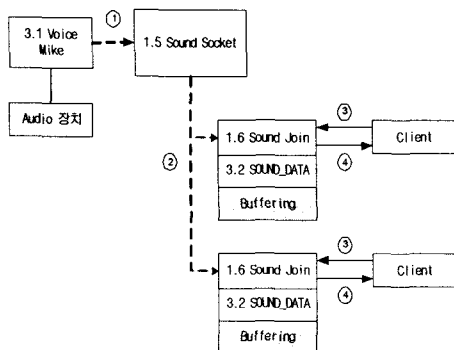


그림 10. 음성전송 처리
Fig 10. Audio Transfer Process

Sound Socket 모듈은 Voice Mike로부터 전송되어 온 음성 데이터를 Sound Join Socket으로 전송하여 클라이언트에 전송한다. 이때 음성데이터를 버퍼링 시켜서 음성데이터 요구 시 버퍼링된 음성 데이터를 전송한다.

3.2 Client 기능별 상세 작업 흐름도

3.2.1 Connect Process

클라이언트에서는 서버와 connect하기 위해 Socket Manager를 이용하는데 여기에서는 서버와의 데이터나 메시지 교환을 위해 Command Socket과 Video Socket, Sound Socket가 필요하다.

서버와 접속한 다음, 서버로부터 "index_n" 메시지를 전송받고 "login" 메시지를 전송 받는다. "login" 메시지를 전송받는 순간부터 영상과 음성데이터를 전송 받으며, 서버로 Cam 제어 메시지를 전송한다.

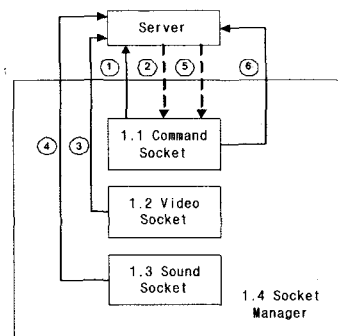


그림 11. 연결 처리
Fig 11. Connect Process

3.2.2 영상/음성재생 Process

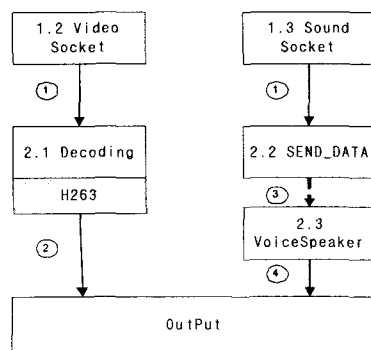


그림 12. 영상/음성 재생 처리
Fig12. Video/Audio Play Process

Video Socket으로 받은 영상을 디코딩하여 출력하고 Sound Socket으로 받은 음성 데이터를 버퍼링하여 출력한다.

IV. 실험 및 결과

모니터링 프로그램은 한 순간에 1개의 클라이언트의 접속을 허용하는 것이 기본이지만, 하드웨어 리소스가 허용하는 한 동시 접속을 가능하여야 하며, 동시 접속 시 동시 컨트롤이 문제가 되는 특정 기능에 대해서는 한 클라이언트에게만 허용할 수 있도록 기능이 구성 된다.



그림 13. 홈뷰어 서버
Fig13. Home Viewer Server

영상은 리얼 타임 전송 및 버퍼링을 통한 전송 방식을 지원하여야 하며, 사용자가 선택할 수 있도록 하였다. 홈뷰어 클라이언트와 서버간의 음성을 교환 할 수 있도록, 음성 연결 설정, 음성 교환 기능이 있으며, 클라이언트는 Jpeg 영상 및 홈뷰어 프로그램에서 설정한 다른 다양한 종류의 코덱 영상을 볼 수 있도록 하였다. PDA 클라이언트는 서버측에서 제공하는 176*144 영상을 수신할 수 있도록 하였으며, 서버에서 설정한 여러 기능과 연동하여 동작할 수 있도록 하였다.

시스템이 동작할 수 있도록 하기 위해 서버측의 DVR에서 전송되기 전에 음성과 영상 데이터에 대한 버퍼링 작업과 H263 Encoding 작업이 필요하고, 클라이언트 PC나 PDA에서도 서버측에서 전송되어 온 음성과 영상데이터에 대한 버퍼링 작업과 Decoding 작업

이 이루어진다.

PC 클라이언트의 경우에는 문제가 없지만 PDA 클라이언트의 경우 하드웨어 사양의 제약으로 전송량을 처리범위 내에서 제한할 수 있도록 프로토콜을 On Demand 형식으로 구성해야 한다.

본 실험 결과 서버에서는 다수의 클라이언트(PC, PDA)의 접속을 수용하였고 음성과 영상데이터를 전송하여 버퍼링과 Decoding 과정을 거쳐 원활하게 클라이언트 쪽에서 재생하였다.

V. 결론 및 향후 연구 방향

본 연구에서는 마이크가 내장된 USB 카메라로 촬영한 영상 데이터를 H263 코덱을 통해 영상 데이터를 인터넷을 통해 PC나 PDA로 전송하는 웹 카메라 기반 모니터링 시스템을 구현하였다. 본 연구에서는 모니터링 시스템을 서버와 클라이언트로 구분하였다. 카메라 장치 구동 모듈은 각 가정이나 사무실에 설치되어 서로 다른 카메라의 장비들마다 서로 다른 장치 드라이버가 존재하고 있다. 따라서 서로 다른 웹 카메라의 동일한 관리 및 개발을 위해서는 일관된 장치 구동 모듈이 필요하다. 이를 위하여 웹 카메라 장치제어 처리부, 메시지 처리부, 레지스트리 처리부, 화면출력 처리부 등의 다양한 구동 모듈들이 필요하다. 본 웹 카메라 장치 구동 모듈은 서로 상이한 웹 카메라 장치들 간의 동작 처리부를 일관된 장치 구동 모듈로 통일하고, 관리자에게 일관된 구동 모듈을 제공함으로써 시스템 운용 및 관리의 편리성에 목적이 있다. 특히 DVR를 이용해 모니터링 영상 저장기능과, PC와 모바일 폰을 이용하여 WEB과 WAP 방식으로 모니터링 영상을 전송하였다. 향후 연구 방향으로는 핸드폰의 CALLBACK URL SMS 기능을 보강하여 모바일폰에서도 PC 클라이언트와 동일한 동영상을 전송하는데 있다.

참고문헌

[1] H.C. Andrews and B.R. Hunt, Digiyal Image Restoration, Prentice-Hall, Inc. 2000
[2] John Miano, Compressed Image File Format JPEG,

PNG, GIF, XBM, BMP, Addison Wesley, 2000

[3] Juan Pedro Bandera Rubio, ChangJiu Zhou, Vision-based Walking Parameter Estimation for Biped Locomotion Imitation, Computational Intelligence and Bioinspired System, Lecture Notes in Computer Science, LNCS 3512, p.677-684, IWANN 2005.

[4] Mujtaba Khambatti, Kyung Dong Ryu, Structuring Peer-to-Peer Networks using Interest-Based Communities, Databases, Information Systems, and Peer-to-Peer Computing, Lecture Notes in Computer Science, LNCS 2944, p.48-63, DBISP2P 2003.

[5] Ren-Hoo Cheng, Po-Cheng Huang, A Two-Stage Queuing Approach to Support Real-Time QoS Guarantee for Multimedia Services in TDMA Wireless Networks, Advances in Multimedia Information Processing PCM 2004, Lecture Notes in Computer Science, LNCS 3332, p33-40. 2004.

[6] Ramesh Yerraballi, ByungHo Lee, Distributed Video Streaming Using Multicast, Advances in Multimedia Information Processing PCM 2004, Lecture Notes in Computer Science, LNCS 3332, p122-130. 2004.

[7] Megumu Ueda, Daisaku Arita, Real-Time Free-View Point Video Generation Using Multiple Camera and a PC-Cluster, Advances in Multimedia Information Processing PCM 2004, Lecture Notes in Computer Science, LNCS 3332, p418-425. 2004.

[8] Bing Zheng, Mohammed Atiquzzaman, Network Requirement for Management of Multimedia over Wireless Channel, Management of Multimedia on the Internet, MMNS 2002, Lecture Notes in Computer Science, LNCS 2496, p49-61, 2002.

[9] Ivar Jorstad, Do van Thanh, Toward Service Continuity for Generic Mobile Services, Intelligence in Communication System INTELLCOMM 2004, Lecture Notes in Computer Science, LNCS 3283, p,319-326, 2004

저자소개

정 종 근



1995년 조선대학교 전자계산학과 졸업(이학사)
1997년 조선대학교 대학원 전자계산학과 졸업(이학석사), 2002년 8월 조선대학교 대학원 전자계산학과 졸업 (이학박사)

2004년 3월 - 2005년 현재 호남대학교 컴퓨터공학과 초빙교수

※관심분야 : 인공지능, 검색엔진, 데이터베이스, 정보보안, 전자상거래, 바이러스

천 승 환



1992년 전남대학교 화학공학과 졸업(공학사)

1995년 전남대학교 전산통계학과 석사졸업(이학석사)

1999년 전남대학교 전산학과 박사수료

현 (주)도올정보기술 대표이사 (1996~현재)