

USB Camera를 이용한 이미지 트래킹을 위한 Pan/Tilt 제어용 Embedded System 개발

Development of an Image Tracking System Using an USB Camera on an Embedded System

김희식*, 남철**, 오드게렐***, 하관용***
 Hiesik Kim, Chul Nam, Odgera Ayurzana, Kwanyong Ha

Abstract - An embedded system has been applied to many fields including households and industrial sites. The embedded system is implemented for image tracking in security area. This system supports a fixed IP for the reliable server operation on TCP/IP networks. A real time video image on the is analyzed to detect a certain invader who jumped into the observed area. The digital camera is connected at the USB host port of the target board. The video images from the video camera is continuously analyzed and displayed at the Linux web-server. The moving vector of the invaders on the continuous image frames is calculated and then it sends the calculated pan/tilt movement. That used Block matching algorithm and edge detection algorithm for past speed. And the displacement vector is used at pan/tilt motor control through RS232 serial cable. The experiment result showed tracking performance by the moving part speed of 10 to 150 pixels /sec.

Key Words : Embedded System, Camnera, Image Tracking, Pan/Tilt, Linux, Block Matching Algorithm

1. 장 서론

전자정보통신 기술 분야에서 많은 제품이 임베디드 시스템으로 구현되고 임베디드 장치는 유선과 무선을 통해 네트워크로 연결되고 있다. 임베디드 리눅스를 운영체제로 하여 저렴한 임베디드 시스템을 구축하고 Camera를 이용하여 움직이는 물체를 지속적으로 추적하는 보안 용도의 임베디드 시스템에 적용하는 기술 개발이 필요하다.

저렴한 영상 트래킹 기술을 구현하기 위해서는 고정 IP를 임베디드 시스템에 할당하고 네트워크 환경을 구축하여 시스템의 USB port를 통해 입력된 Web Camera의 영상 이미지를 이용해야 한다. 실시간 빠른 카메라 2축 위치 제어를 통한 영상 트래킹을 위해서 영상 추적 계산과정에서 각 프레임별로 8*8 또는 5*5 개의 Block으로 나누어 비교하는 Block matching Algorithm(BMA)을 사용하면 실시간 추적이 가능하다. 물체의 이동 변위를 계산하여 pan/tilt 모터를 제어함으로써 지속적으로 움직이는 물체를 관찰이 가능하다. 원격 모니터링을 위하여 입력된 이미지는 CGI 프로그램을 이용하여 인터넷을 통해 Real-Time으로 방송한다.

시간적인 모니터링이 필요한 많은 장소에서 real time 이미지 전송 시스템은 산업현장이나 학교 등에서 저렴한 비용으로 설치가 필요하고 적용분야 요구가 많다.

2. 시스템 설계 및 구성

2.1 시스템 개발 환경

그림1은 임베디드 리눅스 상에서 USB 카메라를 이용한 실시간 방송 시스템을 구현하기 위한 개발환경 구성도이다. 개발 환경은 Host PC(Linux server), SoB(System on Board), 개발용PC(windows xp), HUB(Ethernet LAN), JTAG Dongle, RS232 Serial Cable로 구성된다.

Host PC는 Red Hat Linux 7.1을 기본 운영체제로 사용하고, 커널은 SoB에 사용된 버전과 같이 V2.4.18을 하드웨어 환경에 맞도록 커널 컴파일하여 사용했다. Host PC는 SoB의 Flash Memory에 설치될 Kernel, Boot loader, Root file System, 각종 응용 프로그램을 컴파일하고 수정하는 작업하는 공간으로 활용된다.

SoB는 주)GES에서 제작된 Target Board로 삼성의 S3C2410 SoC를 사용하였다. S3C2410은 ARM920T core를 내장하고 UART, USB, Memory controller를 포함한다. 외부 Memory로는NAND Flash Memory(64M)와 SDRAM(32M) 2개가 사용되었다.

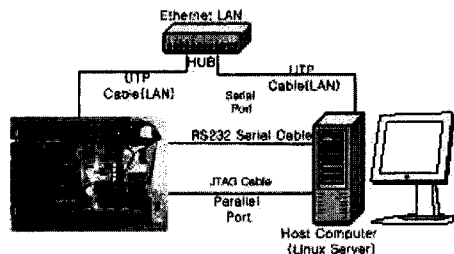


Fig 1. Embedded System Scheme

저자 소개

- * 正會員 : 서울市立大學 電子電氣工學部 教授 工博
- ** 準會員 : 서울市立大學 電子電氣工學部 碩士課程
- *** 準會員 : 서울市立大學 電子電氣工學部 博士課程

2.2 호스트 PC와 Target system 간의 통신 환경

임베디드 시스템을 개발 하는데 있어 네트워크 환경을 구축 하는 것은 매우 중요하다. 네트워크 환경은 타겟보드의 열약한 작업 환경을 HOST PC에서 작업함으로 효율을 극대화하기 위함이다. 하지만 네트워크 환경을 구축하기 위해서는 먼저 HOST PC와 타겟 간의 물리적인 연결이 우선 되어야한다. 본 시스템에서는 JTAG Dongle을 이용한 Parallel 통신과 RS232를 이용한 serial 통신을 동시에 이용했고 통신 emulator로 minicom을 이용하였다.

HOST PC는 기본적으로 NFS SERVER, FTP SERVER, APACHE SERVER로 동작 하도록 구축하였다. 이러한 네트워크 환경을 통해 임베디드 시스템 보다 상대적으로 개발환경이 유리한 HOST PC에서 모든 작업을 하고 Cross Compiler를 통해 만들어진 실행 파일을 임베디드 보드에 인식하는 방법으로 실험하여 SoB (System on Board)의 개발 환경 이용하였다.

2.3 Image tracking Algorithm

실시간 이동물체 추적을 위해 ED(Edge Detection) and BMA(block matching Algorithm)이 많이 사용된다. ED 방법은 많은 시간이 소요되는 방법이다. 그이유는 모든 픽셀에 대한 복잡한 계산을 해야 하기 때문이다. 따라서 움직이는 물체에 대한 이미지 처리 기법으로는 적절치 않다. BMA는 ED보다 속도면에서 훨씬 빠르다. 하지만 BAM에는 예기치 않은 에러발생 가능성이 있다. 또한 연산의 편의성을 위해 Block size 를 고정하기도 한다. 본논문에서는 두 알고리즘 ED BMA의 장점을 이용하여 물체의 인식능력과 Image tracking 성능을 개선했다. BMA의 경우는 BLOCK byBLOCK MATCHING algorithm으로 이미지를 처리한다. 이것은 Previous (n)frame 과 Current (n-1)frame의 block data의 Maximum correlation 분석이 이용된다. BMA (Block matching algorithm)에서는 MAD (Mean Absolute Difference) 와 MSB (Mean Squared Difference)가 주로 사용된다. MAD 및 MSD 수식은 다음과 같다.[7]

$$MSD(u, v) = \frac{1}{N_1 N_2} \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} [L^{N+1}(i+u, j+v) - L^n(i, j)]^2$$

$$MAD(u, v) = \frac{1}{N_1 N_2} \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} [L^{N+1}(i+u, j+v) - L^n(i, j)]$$

L: pixel in n th and (n+1)th frame ... (1),... (2)
 u,v: search point in the search area
 N*N: block size in MAD, MSD

간 겹쳐진 부분의 영상 밝기 값이 서로 비슷하면 MSD나 MAD 값은 0에 가까운 값이 나온다. 프레임의 크기(R*C)이고 블럭 사이즈가 (M*N) 일때 (R-M)*(C-N)의 블럭간 비교가 필요하다. 320*240의 프레임에 8*8의 블럭을 나누었을때 Block size는 40*30 이고 블럭간 비교를 위한 280*210번의 겹침 횟수가 발생하기 때문에 시간 복잡도(time complexity)가 문제가 발생한다. MAD의 경우 각각의 겹침 상태에서 샘플을 M*N번을 해야 하므로 총 샘플 횟수는 다음과 같다.

$$T=(R*M)*(C-N)*(M*N) \dots\dots\dots (3)$$

이는 임베디드 시스템에서 많은 영상을 처리하기에 복잡하므로,

영상 위의 이동물체의 이동 변위를 측정하기 위해 RB (대표 블럭)의 대표값을 설정하여 연산 과정을 단순화 시켰다. [7]

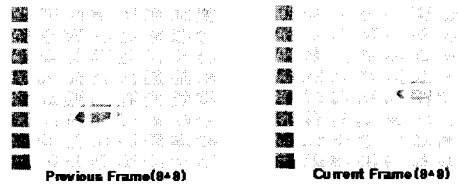


Fig 2. 영상에서 이동물체 포함하는 전과 후 프레임

Fig.2, 움직이는 물체는 배경 화면이 같은 n frame and n+1 frame에서 이미지의 변화를 인식하고 특정 물체의 이동 유무를 판단 하면 된다. BMA에 의해 M*N 개의 Block을 차례로 비교하면서 블럭의 대 블럭간의 대표값을 검사하여 motion vector를 검출한다.

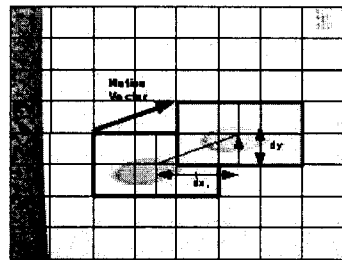


Fig.3. BMA 이용 motion 벡터 계산

2.3 pan/tilt moter control

이동벡터 (moving vector)값을 이용하여 Target 보드의 RS232 serial통신으로 pan/tilt moter를 제어한다. 기본적인 pan/tilt 통신정보는 다음과 같다.

Baud Rate	19200bps	Parity bit	None
Data bit	8bits	Stop bit	1bit
Tilt Offset	0~60	Pan Offset	0~180
Pan/tilt motion: 2Byte			
Pan: (360/412.3)degree per offset			
Tilt: (360/362.15)degree per offset			

표 1. Pan/tilt 모터제어 통신정보 구성

3. 시스템 설계 및 구성

3.1 시스템 구성

움직이는 물체의 변위를 계속하기 위해서는 웹 카메라로 부터 획득한 영상 데이터의 각 프레임들 비교해야 한다. 다음은 이미지 트래킹 시스템을 위한 시스템 구성도이다.

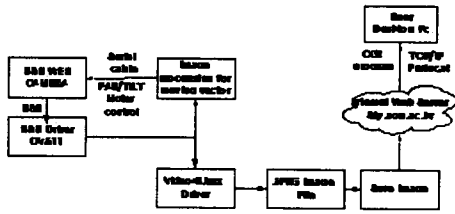


Fig 4 System Flow Chart

임베디드 보드와 웹 카메라는 USB와 Serial port로 연결되어 있다. USB 포트를 통해 입력되는 이미지 프레임의 전송 로고, RS232 직렬통신선은 pan/tilt 모터를 제어하기 위해 사용했다. 웹 카메라를 임베디드 시스템에 설치하기 위해서는 Kernel compile 환경 설정중에 Device Driver를 Linux kernel에 직접 포함시키는 방법과 Module로 제작하여 Kernel에 포함시키는 방법이 있다. 본 실험에서는 디바이스 드라이버 모듈을 커널에 올리는 방법을 사용했다. 임베디드 보드는 고정 IP를 갖는 서버로 동작 하며 임베디드 보드의 부족한 메모리 자원은 HOST PC에 NFS 서버를 설치한후 메모리를 공유하는 방법을 이용하여 data를 저장 하였다.

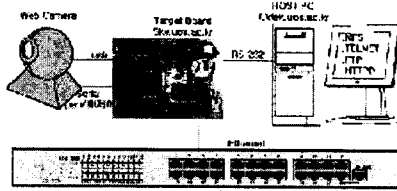


Fig 5 Image Tracking System

3.2 임베디드 환경에서 WEB SERVER 구현

web server는 SoB의 작은 메모리 공간에서 무리 없이 구동시키기 위하여 Boa web server를 이용하여 구현 하였다. 연구에서 사용된 임베디드 시스템은 CGI (Common Gateway Interface) 프로그램을 사용했다. Web Server 구성은 HTML을 기본으로 제작했다.

USB Digital 카메라로부터 들어온 데이터는 임베디드 리눅스의 Web Camera Program에 의해서 JPEG 파일로 저장된다. JPEG형태의 데이터는 WEB SERVER의 특정 디렉토리에 저장되어, 클라이언트 PC에서 TCP/IP 영상이미지의 변환 시간을 USB카메라의 전송시간에 맞추어 웹브라우저 갱신을 위한 html coding은 다음과 같다.

```

<META HTTP-EQUIV="pragma"
CONTENT="no-cache">
<META HTTP-EQUIV="refresh" CONTENT="1">
  
```

임베디드 시스템 상에서 Web Camera를 동작시키고 멈추게 하는 shell 프로그램을 개발하였다. 임베디드 시스템 보드에 카메라를 연결하여 실시간으로 web 상에서 자동 영상추적을 확인할 수 있었다.

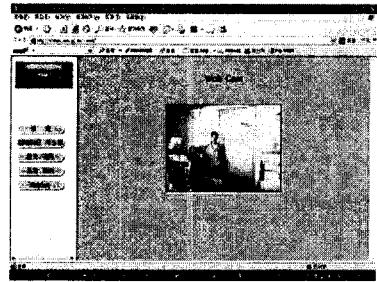


Fig. 6. Internet Screen of Video with Real Time Image Tracking

4. 결론

이동 물체의 image 를 지속적으로 추적하는 임베디드 시스템은 H/W 및 S/W 구현하였다. 이미지 크기가 320*240 해상도 동영상에서 5 frame/sec로 영상을 추적하면서 인터넷으로 실시간 전송하였다. 이미지의 전송 방법에 있어, 웹브라우저를 통해서 JPEG 파일을 갱신하는 형식을 취했기 때문에 연속적인 동영상 속도에는 시간적으로 부족하였다. 인터넷 상에서 임베디드 시스템에 연결된 Camara로부터 입력된 동영상 이미지를 실시간으로 원격지에서 감시가 가능했다. 본 시스템은 산업현장이나 가정에서 적은 비용으로 적용할수 있다.

참 고 문 헌

- [1]. Hwang-Seok Oh, Gunhan Park, Heung-kyu Lee "Block-Matching Algorithm Based on Dynamic Search Window Adjustment" Proceedings of The 4 th International Workshop on Mobile Multimedia Communications September 29-October 2, 1997, pp 407-412
- [2]. Hitoshi KIYA, Jun FURUKAWA, Yoshihiro NOGUCHI "Block Matching Motion Estimation Using Less Grey Level Images for MPEG Video" "Proceedings of ITC-CSSCC 98, July pp 167-170
- [3]. Raphael Canals, Anthony Roussel, Jean-Luc Famechon, and Sylvie Treuillet "A Biprocessor-Oriented Vision - Based Tracking System » IEEE Trans. Industrial Electronics, vol.49.2.APRIL 2002
- [4]. Ahmad,M.B.,Dong Yoon Kim, Kyoung Sig Roh and Tae sun Choi, "Motion Vector Estimation Using Edge Oriented Block Matching Algorithm for Video Sequences ",Image Processing, 2000. International Conference on vol. 1 pp 863
- [5]. Hiesik Kim, Youngil Kim, "Development of a Temperature Monitoring System with Real Time Data Processing by Using an Internet", The International Conference on Mechatronics and Information Technology. December 4-6 2003, ICMIT'2003, pp.400-403
- [7]. Sung-Yug Choi, Haw-Ra-Hur, and Jang -Myung Lee:"Efficient Tracking of a Moving Object Using Representative Blocks Algorithm" ICCAS2004 pp 678-681
- [8]. Hiesik Kim, Youngil Kim, D.Y.Seol, Chul Nam "The Measurement Temperature and Analysis used Embedded System by Internet Explorer" KSPE 2004. 1003-1006