

모바일 로봇 및 네트워크 카메라 기반 지능형 감시 시스템 설계

A Design of Intelligent Surveillance System Based on Mobile Robot and Network Camera

박정현¹, 이민영², 심귀보^{1*}

Jung-Hyun Park¹, Min-Young Lee², and Kwee-Bo Sim^{1*}

¹ 중앙대학교 전자전기공학부

² (주) 바이오 인식

요 약

보안이 중요시 되는 공간에서 지능형 감시 시스템의 필요성이 점차 중요시 되고 있다. 본 논문에서는 embedded Linux 기반의 Mobile Robot에 Network Camera를 탑재 하여 침입자를 추적할 수 있는 시스템 구현에 목적을 두고 있다. Network Camera부터 Wireless Lan을 이용하여 서버로 영상을 전송하고, 서버에서 블록매칭 알고리즘을 이용하여 침입자의 이동경로를 파악하며 침입자에 대한 방향 정보를 전송하여 침입자를 추적한다. 로봇이 침입자를 추적함에 따라 침입자의 유효 영상을 얻는다. 본 논문에 의해서 구현된 시스템은 다른 감시 시스템과 연동하여 지능형 감시 시스템으로서 신뢰성을 더할 수 있다.

Abstract

The necessity of intelligent surveillance system is gradually considered seriously from the space where the security is important. From this paper will load Network Camera in Mobile Robot based on embedded Linux and Goal is in the system embodiment will be able to track the intruder. From Network Camera uses Wireless Lan transmits an image with server, grasps direction of the intruder used Block Matching algorithms from server, transmits direction information and tracks an intruder. The robot tracks the intruder according to gets the effective image of an intruder. In compliance with this paper the system which is embodied is linked with a different surveillance system and as intelligent surveillance system there is a possibility of becoming worse a reliability.

Key Words : Intelligent surveillance system, Tracking, Network Camera, Mobile Robot, Embedded Linux

1. 서 론

인터넷 망의 발달로 영상의 전송이 가능해 짐에 따라 대부분의 지능형 감시 시스템은 카메라를 이용하여 원격지에서 모니터링을 할 수 있는 기능을 가지고 있다. 고정 카메라를 이용하여 영상을 저장하고 전송하는 것은 유효 영상을 얻기 힘들다. 이에 여러 개의 고정 카메라와 PT(Pan-Tilt) 카메라를 조합하여 네트워크를 구성하는 방법이 제안되었다[1]. 이 방법은 유효 영상을 얻기 위해서는 효과적이지만 감시하기 위한 모든 임의의 공간에 카메라를 설치해야 되는 단점이 있다. 이에 고정 카메라와 PTZ(Pan-Tilt-Zoom) 카메라를 조합하여, 고정카메라로 움직임을 감지하고, PTZ카메라를 이용하여 움직임을 추적함으로써 영상을 획득하는 방법이 제안되었다[2]. 이 방법 또한 카메라가 설치되지 않

거나 사각지대에서는 유효 영상을 얻을 수 없는 방법이다. 카메라가 설치되지 않은 장소에서도 유효 영상을 얻기 위해 효과적인 지능형 감시 시스템은 자율 이동로봇을 이용하는 것이다. 최근 빌딩 경비 시스템으로 자율 이동 경비 로봇을 이용하는 것으로 보고되고 있으며 wheel-type으로써 객체 추적이나 모터 진단, 화재·누수 검출, 장애물 감지의 센서 시스템을 가지고 있다[3].

대부분의 시큐리티 로봇은 카메라를 탑재하여 임의의 장소를 순찰 하면서 모니터링 스테이션으로 이미지를 전송한다[3]. 또한, 이동로봇이 순찰을 마친 후 로봇의 도킹 위치로 돌아와 자동 충전시스템을 가진 시큐리티 로봇도 있다[4]. 이 로봇은 클라이언트 사용자가 컴퓨터나 PDA를 이용하여 인터넷을 통하여 모니터링 할 수 있는 기능도 가지고 있다. 이러한 시스템[3-5]은 영상 전송을 확장성이 있는 인터페이스로 전송하지 않고, 무선으로 전송하는 모듈을 이용하였기 때문에 영상을 전송할 수 있는 유효 범위에 서버컴퓨터를 필요로 하거나, 다른 카메라나 다수 이동 로봇에 탑재된 카메라와 연동하여 사용하기에는 어려움이 있다.

본 논문에서는 다른 네트워크 카메라나 네트워크를 이용하는 다른 감시 시스템과 연동이 가능한 인터페이스를 갖춘 지능형 감시 시스템을 구현하고자 한다. 무선 네트워크 카메라를 Embedded Linux 기반의 Mobile Robot에 탑재하

접수일자 : 2008년 4월 4일

완료일자 : 2008년 7월 30일

* 교신 저자

본 연구는 2008년도 지식경제부 성장동력산업과제의 연구비지원에 의해 수행되었습니다. 연구비지원에 감사드립니다.

여, 침입자의 이동 경로에 따라 로봇이 추적하여 감지 영상을 로컬 또는 원격지 모니터링 시스템으로 실시간 전송하는 시스템을 구현하고자 한다. 본 논문의 2장에서는 시스템 구성과 네트워크 카메라, Mobile Robot에 대해 기술하고, 3장에서는 로봇에서 획득한 영상을 이용하여 침입자를 추적하는 알고리즘과 로봇이 객체를 추적하기 위한 로봇의 이동 좌표에 대해 기술한다. 4장에서는 제안한 시스템의 실험 결과에 대해 기술하고, 마지막으로 5장에서는 결론 및 향후 과제에 대해 언급한다.

2. 지능형 감시 시스템의 구성

본 논문에서 구현된 지능형 감시 시스템은 네트워크 카메라를 탑재한 모바일 로봇과 로컬네트워크나 원격지의 컴퓨터로 구성되어 있다. 모바일 로봇은 임의의 공간에서 획득한 영상을 모니터링 컴퓨터로 스트리밍 한다. 영상을 받은 컴퓨터는 영상에서 블록매칭 알고리즘[6, 7]을 이용하여 객체의 움직임을 감지한다. 감지된 움직임의 정보를 인터넷 망을 통하여 모바일 로봇으로 전송하여 로봇이 침입자를 추적하게 된다. 이때, 추적자의 이동 경로를 탐색하면서 유효 영상을 얻는다. 또한 모바일 로봇을 모니터링 시스템에서 수동 컨트롤이 가능함에 따라 원격지에서 로봇이 위치한 임의의 공간의 영상을 획득 할 수 있다.

2.1 시스템의 구성

본 논문에서 구현된 지능형 감시 시스템의 구성은 그림 1과 같이 구성되어 있다. 다른 네트워크 보안 장비와 연동이 가능한 모바일 로봇 구현이 목적이기 때문에 모든 통신 인터페이스를 유·무선 랜으로 구성되어 있다

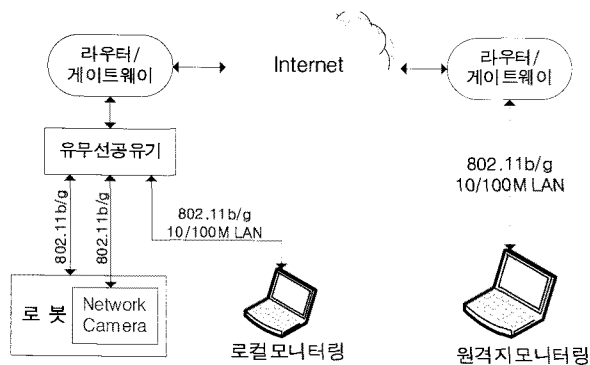


그림 1. 지능형 감시 시스템의 구성

Fig. 1. Consist of Intelligent surveillance system

로봇과 Network Camera는 IP공유기와 연결되어 있으며 Private(사설) IP를 할당받아 사용된다. Public(공인)

IP도 가능하지만 보안상의 이유로 공유기의 포트포워딩 기능을 이용하여 각각 4000번과 3001번의 포트를 이용한다. 모니터링 컴퓨터는 로컬 네트워크나 원격지에서 가능하다. 유무선 공유기에 접속하기 위해서는 디바이스의 MAC(Media Access Control) Address를 등록해야 하기 때문에 로봇, 네트워크 카메라, 로컬 모니터링 PC를 제외한 다른 디바이스는 접속을 할 수 없고, 네트워크 카메라에 접속하기 위해서는 아이디와 패스워드 인증을 하기 때문에 보안이

유지 된다.

2.2 Mobile Robot

본 감시 시스템에서 사용된 Mobile robot은 그림 2와 같다. Intel사의 32bit RISC CPU (PXA270)를 탑재한 휴먼스사의 EMBOT270을 사용하였다. 모바일 로봇에 탑재된 임베디드 보드는 리눅스 커널 2.6.18버전에 스테핑 모터와 무선랜, 초음파 센서를 포팅하였고, 어플리케이션은 gcc 컴파일러를 이용하여 크로스 컴파일 하였다.

원격지 모니터링 시스템과 통신을 위하여 USB 타입의 무선랜을 장착하여 무선 IP 공유기를 통하여 원격지 모니터링 시스템과 TCP/IP 연결이 가능하다. 또한 객체 추적 시 장애물 회피를 위해 전면, 좌우45°, 좌우90°에 장착 되어 있는 5개의 초음파 센서를 이용한다.

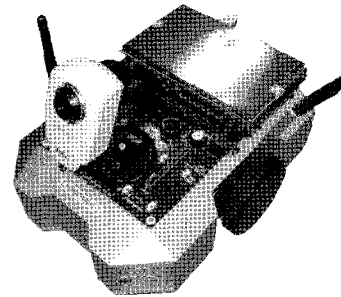


그림 2. 네트워크 카메라를 탑재한 모바일 로봇
Fig. 2. Mobile robot and Network Camera

2.3 Network Camera

Network Camera는 Motion JPEG와 MPEG-4를 동시에 지원하는 영상 압축 칩 ARTPEC-A를 내장하고 있으며, 유·무선 인터페이스를 가지고 있는 AXIS사의 AXIS207W를 사용하였다. AXIS사에서 제공하는 HTTP API를 이용하여 QVGA(320×240 pixels), BMP 또는 JPEG/MJPEG영상을 25fps이상으로 모니터링 시스템으로 전송받는다. 영상을 받기 위해 모니터링 시스템에서는 표 1과 같이 네트워크 카메라로 Request를 하게 되고, 요청을 받은 네트워크 카메라는 Return 값으로 영상 이미지 데이터가 모니터링 시스템으로 전송된다. 또한 파라미터나 값을 설정, 제거, 리스트 확인을 하기 위해 표 1과 같은 Parameter 패킷을 전송할 수 있으며 시큐리티 레벨에 따라 view/operator/admin으로 나뉜다.

3. 객체 추적 알고리즘

영상에서 객체의 움직임 정보를 추출하는 방법 중 가장 널리 사용하는 방법은 차 영상을 이용하는 방법[8], 옵티컬 플로우(Optical flow)[9]와 블록 매칭(Block Matching)[6, 7]방법 등이 있다. 차 영상을 이용하는 방법은 기존 영상과 참조 영상의 밝기 차이를 구한 차 영상(difference image)을 기반으로 움직임 정보를 추출하여 임계화 방법을 사용하기 때문에 임계치에 따라 잡음이 이동물체로 검출 될 수 있고, 이동 물체를 검출하지 못할 수 있는 문제점이 있다. Optical flow는 과도한 계산 량을 요구하기 때문에 실시간 처리에는 적당하지 않다.

표 1. 네트워크 카메라의 요청 및 응답
Table 1. Request and Return of Network Camera

Image Request	http://<servername>/axis-cgi/bitmap/image.bmp?resolution=320x240&camera=1
Image Return	HTTP/1.0 200 OK\r\n Content-Type: image/bitmap\r\n Content-Length: 1216566\r\n \r\n <bitmap image data>\r\n
Parameter	http://<servername>/axis-cgi/<type>/param.cgi?<parameter>=<value>[&<parameter>=<value>...] <type> : view, operator, admin

따라서 본 논문에서는 영상의 압축에 사용하는 움직임 예측 방법 중 하나인 블록 매칭 움직임 추정 방법을 이용하여 실시간으로 영상 내 객체의 움직임 정보를 추출하여 침입자를 추적하고 유효 영상을 얻는다.

3.1 블록 매칭 움직임 추적 방법

블록 기반의 움직임 벡터를 계산 하는 방법은 블록 매칭 방법을 사용하여 매칭 에러 값을 최소로 갖는 블록을 측정하여 구할 수 있다.

블록매칭은 평균 절대값 차이 (SAD : Sum of Absolute Difference)[7]를 이용하여 두 블록의 유사도를 평가하는 것이다. 즉 현재 이미지 t 번째 프레임 I_t 에서 좌표 (x, y) 에 $M \times N$ 크기의 블록이 있다고 가정했을 이 블록과 레퍼런스 이미지 $t-1$ 번째 프레임 I_{t-1} 에서 좌표 $(x+u, y+v)$ 에 위치한 블록과의 절대 값 차이의 합은 식 (1)과 같이 정의한다.

$$SAD_{(x,y)}(u,v) \equiv \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} |I_t(x+i, y+j) - I_{t-1}(x+u+i, y+v+j)| \quad (1)$$

식(1)에서 SAD 값을 가장 작게 만드는 변위 (u, v) 의 값 이 해당 블록의 움직임 벡터이다. $SAD_{(x,y)}(u, v)$ 의 최소 값을 갖는 움직임 벡터 (u, v) 를 $v(\hat{u}, \hat{v})$ 로 나타내면 식(2)와 같다.

$$v(\hat{u}, \hat{v}) \equiv \arg \min_{(u,v)} SAD_{(x,y)}(u, v) \quad (2)$$

검색 윈도우 안에서 해당 블록과 SAD값이 가장 작은 위치의 블록을 찾고, 그 위치 변화를 움직임 벡터로 인지하게 된다.

3.2 추적 객체 이동에 따른 로봇 이동 좌표

객체의 움직임이 검출되고 최소 SAD를 찾아 움직임 벡터로 인지를 하게 되면 로봇은 이 좌표로 이동을 해야 한다. 이를 위해 그림 3과 같이 320 x 240의 영상을 5 x 3의 블록으로 나누어 움직임 벡터로 인지를 한 벡터의 수를 구하게 된다. 그림 3은 블록의 좌표와 움직임벡터의 수를 나타내고 있다.

y, x 좌표에서의 움직임 벡터의 수($P_{SAD}[y][x]$)를 나타내

는 그림 3의 값을 이용하여 식 (3)과 같이 3 x 3 윈도우를 통과하여 주변 벡터의 값을 참조하여 가중치(B_{SAD})를 계산한다.

$$B_{SAD}[y][x] \equiv \sum_{j=-1}^1 \sum_{i=-1}^1 (2 * P_{SAD}[y][x] + P_{SAD}[y+j][x+i]) \quad (3)$$

B_{SAD} 의 값 중에서 최대 값을 가지는 블록이 모바일 로봇의 이동 좌표($B_{coordinates}$)가 되고 식 (4)와 같이 정의 된다.

$$B_{coordinates}[y][x] \equiv \arg \max_{(y,x)} (B_{SAD}[y][x]) \quad (4)$$

그림 4는 그림 3의 영상에 식 (3)을 적용하여 로봇의 이동 좌표의 가중치 값을 구하였다. 최대값을 가지는 블록 $B_{coordinates}[1][3]$ 이 로봇의 이동좌표가 된다.

B[0][0]	B[0][1]	B[0][2]	B[0][3]	B[0][4]
0	0	0	10	25
B[1][0]	B[1][1]	B[1][2]	B[1][3]	B[1][4]
0	0	0	35	27
B[2][0]	B[2][1]	B[2][2]	B[2][3]	B[2][4]
0	0	18	10	0

그림 3. 이미지 분할과 벡터의 수
Fig. 3. Image Division and sum of vector

B[0][0]	B[0][1]	B[0][2]	B[0][3]	B[0][4]
0	0	45	107	122
B[1][0]	B[1][1]	B[1][2]	B[1][3]	B[1][4]
0	18	73	160	134
B[2][0]	B[2][1]	B[2][2]	B[2][3]	B[2][4]
0	18	81	100	72

그림 4. 로봇의 이동좌표 가중치
Fig. 4. Weight of robot coordinate

4. 실험 결과 및 고찰

4.1 침입자 추적 시스템의 구성

침입자 추적 시스템의 모니터링 화면은 그림 5와 같이 로봇에서 전송한 영상을 디스플레이, 로봇이 이동시 장애물을 회피하기 위한 초음파 센서 값을 나타내고 있다. Network Config은 로봇과 카메라에 접속하기 위한 IP, Port 설정을 위한부분이다. 또한 모니터링 시스템에서 로봇을 수동으로 컨트롤 할 수 있는 Robot Control 메뉴가 있다. System Display 버튼을 클릭 시 모달리스 대화 상자가 생성되며 입력영상, 검출 영상, 움직임 벡터, 트래킹 영상을 라디오 버튼을 선택하여 디스플레이 할 수 있다. 시스템 로그, 동작 메시지, 움직임 벡터 위치 등은 Message box에 표현이 되고 있다.

4.2 실험방법

지능형 감시 시스템의 성능 검토를 위해 임의의 침입자가 출연하여 그림 6과 같은 경로로 이동할 경우 로봇이 모니터링 시스템에 영상을 전송하고, 모니터링 시스템에서 침

입자 이동 정보를 전송하여 로봇이 침입자를 추적하는 과정에 대해 실험 하였다.

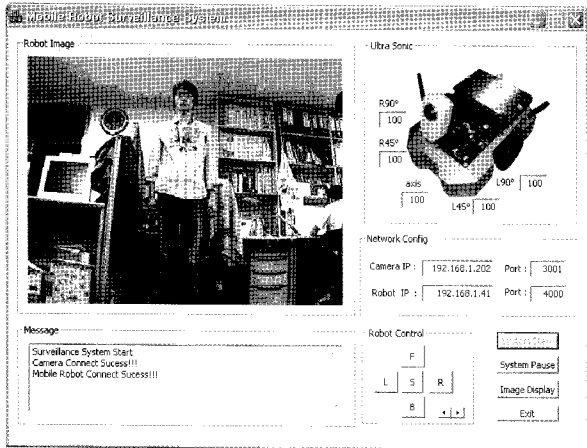


그림 5. 침입자 추적 시스템 화면
Fig. 5. Display Monitoring system



그림 6. 침입자 이동 경로
Fig. 6. Mobile course of Intruder

4.3 실험결과

그림 7은 침입자가 그림 6과 같이 ①~④경로로 이동할 경우 모바일 로봇이 추적하는 화면을 나타내고 있다. 그림 8은 로봇이 침입자를 추적함에 따라 로봇에 탑재된 네트워크 카메라가 원격지의 모니터링 시스템에 영상을 전송한 이미지이다. 그림 7과 그림 8의 이미지는 로봇이 추적하는 이미지와 로봇이 모니터링 시스템으로 전송하는 이미지로서 같은 상황에서의 이미지이다.

로봇에 탑재된 네트워크 카메라에서 획득한 이미지는 모니터링 시스템으로 전송된다. 모니터링 시스템에서는 실시

간으로 스트리밍 받은 영상을 이용하여 연속된 두 영상을 그레이 스케일 이미지로 변환한 후 두 이미지의 차 영상을 얻는다. 차 영상을 이진 영상을 변환 후 모폴로지 기법인 침식, 팽창 연산의 전처리 과정을 거쳐 잡음을 제거한 후 그림 10(좌하)과 같은 검출 이미지를 생성한다.

블록 매칭 알고리즘을 이용하여 움직임 벡터를 구하기 위해서는 연속된 두 영상을 그레이 스케일 이미지로 변환한 후 식 (1)에 의해 움직임 벡터를 추출 한다.

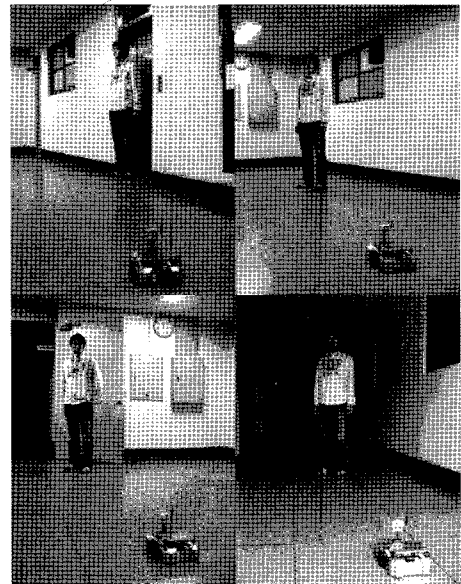


그림 7. 로봇의 침입자 추적
Fig. 7. Intruder Tracking of Robot



그림 8. 로봇의 침입자 영상 전송
Fig. 8. Image transmission of Robot

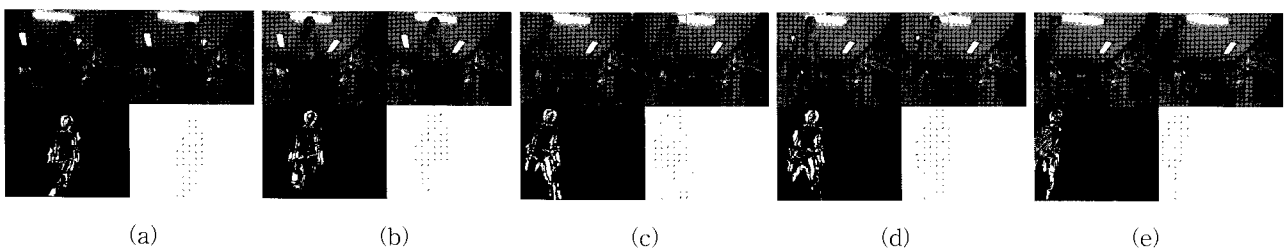


그림 9. 연속 추적 이미지
Fig. 9. Tracking Continuous Image

추출된 벡터를 표현하기 위해 이전 영상에서 다음 영상으로 움직인 방향을 그림 10(우하)과 같이 표시 하였다.

객체가 그림의 오른쪽에서 왼쪽으로 이동하였기 때문에 벡터 표현이 가로 방향으로 나타난 것을 볼 수 있다. 움직임 벡터를 표시하기 위해 입력 받은 컬러 이미지에

움직임 벡터가 나타난 부분을 그림 11과 같이 빨간 사각형으로 표시 하였다.

그림 9는 움직임 객체가 이동함에 따라 입력영상(상좌), 검출 영상(하좌), 움직임 벡터(하우), 트래킹 영상(상우)을 연속적으로 표현한 그림이다. 움직임 객체의 진행방향 따라 (a)~(e)가 표현 되고 있다.

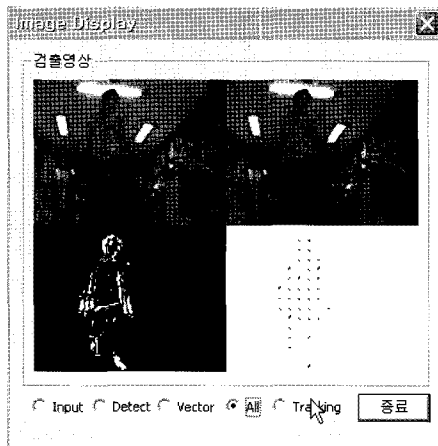


그림 10. 침입자 추적 이미지
Fig. 10. Tracking Image of Intruder

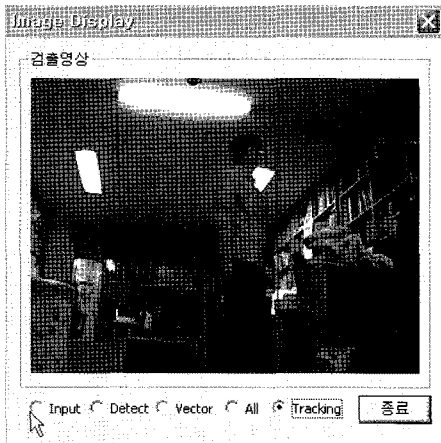


그림 11. 움직임 벡터 영역
Fig. 11. Vector area

5. 결론 및 향후 과제

본 논문에서는 더욱 효과적인 감시 시스템 구현을 위해 모바일 로봇과 네트워크 카메라를 이용한 지능형 감시 시스템을 구현 및 적용 실험을 하였다. 실험 결과 침입자가 이동함에 따라 로봇이 모니터링 시스템에 영상을 스트리밍하고, 모니터링 시스템은 영상을 이용하여 움직임 벡터를 추출하고 이동경로를 파악하였으며, 로봇은 모니터링 시스템으로부터 받은 좌표 데이터를 이용하여 이동함으로써 객체

추적을 수행 하였다.

로봇의 전송 영상은 로봇의 낮은 높이로 인해 영상의 기울어짐이 발생 하였다. 이에 영상을 보정할 수 있는 대안이 필요하다. 또한 실험에 사용한 로봇은 wheel-type으로 계단과 같은 단차가 있는 공간에서는 이동이 불가능 하였다. 이에 단차가 존재하는 공간에서도 이동 가능한 대안 필요하다.

본 논문에서 Mobile Robot와 Network Camera를 이용한 지능형 감시 시스템은 위에서 언급한 문제점을 보완, 수정하여 다수의 Network Camera나 감시 시스템과 연동 한다면 유연성과 신뢰성을 갖춘 지능형감시 시스템이 될 것이다.

참 고 문 헌

- [1] C.Micheloni, G.L. Foresti, L. Snidaro, "A network of co-operative cameras for visual surveillance". *IEEE Proc. of Visual Image Signal Processing.* vol. 152, no.2, 2005.
- [2] A.W. Senior, A. Hampapur, M Lu "Acquiring Multi-Scale Images by Pan-Tilt-Zoom Control and Automatic Multi-Camera Calibration", *Proc. of Seventh IEEE Workshop on Applications of Computer Vision*, vol. 1, pp. 433-438, 2005.
- [3] Y.Shimosasa, J Kanemoto, K. Hakamada, H.Horii, T. Ariki, Y.Sugawara, F. Kojio, A. Kimura, and S. Yuta, " Some results of the test operation of a security service system with autonomous guard robot," *Proc. 26th Annu. Conf. IEEE industrial Electronics Society*, vol, 2000, pp.405-409.
- [4] R. C. Luo, S. H. Shen, K.L. Su, and T.C. Chang, "Remote mobile security system using personal digital assistant(PDA) for Security robot," *Proc IEEE/ASME Int. Conf. Advanced Manufacturing Technologies Education 21st Century*, 2002.
- [5] J.N.K. Liu, Meng Wang, Bo Feng, "iBotGuard:An Internet-Based Intelligent Robot Security System Using Invariant Face Recognition Against intruder," *Proc IEEE* vol. 25, pp97-105, 2005.
- [6] Chan, E. and Panchanathan, S. "Review of Block MatchingBased Motion Estimation Algorithm for Video Compression," *CCECEKCGEI, the Proceedings of Canadian Conference on Electrical and Computer Engineering*, Canada, Vol. 1, Sep.
- [7] Yong-Sheng Chen, Yi-Ping Hung, Chiou-Shann Fuh, "Fast Block Matching Algorithm Based on the Winner-Update Strategy," *IEEE transactions on image TRANSACTIONS ON IMAGE PROCESSING*, vol. 10, no. 8, August 2001.
- [8] Paul L. Rosin, "Thresholding for Change Detection," *Brunel University, technical report*

ISTR, 1997.

[9] J. L. Barron, et.all, "Systems and Experiment: Performance of Optical Flow Techniques", *Int'l J. of Computer Vision*, vol. 12, pp. 43-77, 1994.

저 자 소 개



박정현(Jung-Hyun Park)
2003년 : 배재대학교 전자공학과 공학사
2003년~2006 : (주)내일커뮤니티 연구원
2007년~현재 : 중앙대학교 전자전기공학부 석사과정

관심분야 : Intelligent mobile robot, Humanoid robot, Embedded system, Artificial Brain. 등



이민영(Min-Young Lee)
1986년 : 중앙대학교 전자공학과 공학사
1994년 : 경희대학교 정보통신학과 석사
2000년~현재 : (주)바이오인식 대표이사



심귀보(Kwee-Bo Sim)
1990년 : The University of Tokyo
전자공학과 공학박사

[제18권 3호 (2008년 6월호) 참조]

1991년 ~ 현재 : 중앙대학교 전자전기공학부 교수
2006년 ~ 2007 : 한국퍼지 및 지능시스템학회 회장

E-mail : kbsim@cau.ac.kr
Homepage URL : <http://alife.cau.ac.kr>