

무선센서노드를 이용한 다중객체 추적 감시 시스템

최락현^o 홍원기

대구대학교, 정보통신대학

choimosi@hanmail.net, wonkihong@daegu.ac.kr

Multiple object tracking surveillance system using Wireless sensor nodes

Rockhyun Choi^o, Wonkee Hong

Dept of Information and Communication Engineering, Daegu University

요 약

무선센서네트워크 기술은 유비쿼터스 컴퓨팅에서 정보전달을 위한 중요한 매개체가 되는 등 그 중요성이 점점 더 커지고 있다. 그런 무선센서네트워크에 이미지처리능력을 추가하면 사용자가 이용할 수 있는 좀 더 다양한 서비스를 제공할 수 있다. 그 중 재산 보호를 위한 감시시스템의 관심이 커지고 있고 이에 관련된 몇몇 연구들이 진행되어오고 있다. 몇몇 연구에서 카메라 네트워크 망에서 발견된 객체를 연속적으로 추적을 하지만 카메라와 서버간 직접적인 통신을 통해 다량의 정보를 주고 받기에 Ad-hoc 무선망을 사용하는 방식에서는 적합하지 않다. 또 다른 방안으로는 타겟 추적에 time critical한 방법들을 쓰고 있기에 시간 동기화를 위한 추가적인 프로토콜의 사용이 필요해진다. 본 연구에서는 다중 카메라에서 관계자와 비관계자를 명확히 구분해서 관계자를 제대로 추적함으로써 비관계자 인원이 감시구역에 접근하는 행동을 방지하고 노드간의 시간 동기화 의존도를 줄이기 위한 방법을 제시한다.

1. 서 론

유비쿼터스 컴퓨팅은 사용자의 상황에 따라 언제 어디서나 어느 장치로도 편리하게 적합한 서비스를 제공하는 새로운 정보기술이다[1]. 이 혁신적인 기술은 산업 전반에 걸쳐 새로운 바람을 일으키고 있다[2]. 특히 그 정보 전달 매개체로 무선센서네트워크 기술이 중요한 역할을 담당하고 있고 다양한 기능으로 많은 분야의 정보를 제공해서 사용자가 이용할 수 있다.

현대 생활에서 시설물의 안전과 보호에 대한 관심이 커지고 있기에 U-computing 서비스도 사용자에게 적절한 감시 서비스를 제공 할 필요가 있다. 기존의 무선센서네트워크에서는 경량의 정보만 처리해서 이미지 처리 같은 큰 대역폭을 처리하는 이종(heterogeneous) 센서노드와의 융합이 필요하다[3]. 무선 센서 노드와 카메라 센서 보드를 융합하면 많은 데이터처리를 요구하는 알고리즘을 처리할 수 있어서 추적, 감시 시스템을 구성할 수 있다.

컴퓨터가 카메라로 움직이는 한 명의 사람을 인식할 때 다른 카메라에 같은 사람이 나타나도 그 사람이 같은 사람이라는 것을 인식 시키는 것은 어려운 문제이다.

가장 큰 이유로는 카메라가 바라보는 각도나 거리에 따라 달라 보이는 외양에 있다. 그래서 기존 연구에서 카메라N개:서버의 통신이라 카메라와 서버간의 직접적인 연결이 필요해서 네트워크 망이 커지면 장거리 케이블을 사용해야만 할 것이다. 그리고 객체 구분을 위한 다량의 자료를 수집 한 뒤 구분이 가능했다. CITRIC[4]은 WSN과 카메라 모듈의 결합에 대한 좋은 롤모델이 되었지만, 이동 중인 객체가 감시카메라를 이동 할 때 연속된 관찰을 할 수 없다. 다른 문제점으로는 Cyclops[5] 같은 구조를 이용하면 고성능의 비전 알고리즘을 연산하기 위한 성능이 부족했고, SensEye와 CITRIC의 경우에는 타겟을 추적해서 위치인식을 하는데 두 카메라간 겹침화면 내에 발견 된 객체를 파악해 시간에 따라 기록된 값을 이용한다. 하지만, 시간동기화의 문제는 이동 중인 네트워크가 커지고 다른 성능의 x-tal 이용 같은 요인으로 시간이 지나면 시간동기화 문제가 필연적으로 발생하기 때문에 추가적인 시간동기화 프로토콜 사용이 요구가 된다.

본 연구에서는 특정 구역 감시를 위해 WSN과 카메라 모듈 노드와 무선센서와의 통신으로 객체구분을 할 수 있는 감시시스템을 구성하였다. 명확한 피아의 구별을 위해 센서노드 보유자와 비보유자를 관계자와 비관계자로 각각 나누어 구분을 하였다. 노드를 보유한 관계자는 비겹침 영역에서 객체의 일치점을 찾지 못하던 점을 극복하여 어떤 카메라에 나타나더라도 관계자 파악은 되어 비관계자를 명확히 구분하여 추적할 수 있다. 특정구역을 화면상에 영역설정을 통해서 구분하여 무선노드

+ 본 지식재산권은 지식경제부 및 정보통신산업진흥원의 지원을 받아 수행된 연구결과임. (IT특화연구소 설립)

비보유자 즉 비관계자의 접근을 파악할 수 있게 했다. 카메라 노드 두개를 이용해서 객체 위치추적을 하던 방법은 시간동기화의 문제가 있었지만, 이런 구성으로 비관계자의 중요지점 접근을 알아낼 수 있기 때문에 감시 추적시스템으로써 역할을 해낼 수 있다.

이런 무선 센서네트워크에서의 추적감시 시스템을 연구하기 위해 2장에서는 관련연구를 조사하였고, 3장에서는 추적감시 비주얼 센서 시스템 구성, 그리고 4장에서는 결론 및 향후 연구과제의 순서로 본 페이퍼를 구성했다.

2. 관련 연구

2.1 비주얼 센서 네트워크

현재까지 관련된 연구들 중에 낮은 대역폭으로 동작해서 통신을 담당하는 모듈 즉 무선센서보드와 카메라 센서보드가 분리되어 통신과 영상처리 각각의 역할에 충실한 연구를 중심으로 조사를 했다.

먼저, Cyclops[5]는 Atmega128L과 Xilinx XC2C256을 사용해서 각각 통신과 영상처리를 담당하게 했다. CPLD를 사용해 빠른 영상 데이터 전송을 위해 구성하였다. 저전력 시스템으로 에너지 소비에서 봤을 때는 뛰어난 점을 보인다. 그러나 부족한 메모리공간과 약 8Mhz로 동작하는 CPU의 성능을 가지고 있다. 더군다나 많은 처리량을 요구하는 비전알고리즘을 저장하기에 메모리도 부족하고 처리하기 위한 CPU의 성능도 부족했다.

SensEye[6]의 경우에는 3가지 보드로 구성해서 무선 통신 노드는 Atmega128을 사용하고 웹캠 같은 저해상도의 카메라는 OKI ArmThumb, 그리고 고해상도의 카메라를 처리하기 위해서는 PXA255를 사용했다. 그리고 2단계로 구성해서 영상처리에 좀 더 성능 좋은 CPU를 사용한 CITRIC[4]이 있다. CITRIC는 전력효율이 좋은 MSP430을 통신보드로 쓰고 영상처리 보드로는 PXA270을 사용했다. SensEye와 CITRIC의 경우에 다양한 객체를 추적할 때 필연적으로 시간동기화의 문제가 발생해서 추가적인 시간동기화의 프로토콜을 써야 할 필요성이 있다. 시간동기화의 문제의 원인으로서는 세 가지를 들 수 있다[7]. 첫째로 신규 노드의 진입이고 둘째로 이종 노드의 다른 x-tal 사용, 마지막으로 인터럽트 분주비의 차이를 들 수 있다. 먼저, SensEye의 경우에는 두 번째 원인이 문제가 될 수 있다. 2개 층에 각각 다른 카메라와 다른 속도의 CPU사용으로 2가지 카메라 화면으로 부터 오는 움직이는 객체의 정보(같은 시간에 발견된 위치, 각도)로 위치인식을 하는데 장시간 사용 시 다른 클럭의 사용으로 시간동기화의 문제발생가능성은 점점 커지게 된다. 그리고 CITRIC의 경우에는 같이 연동된 카메라가 아니라 서로 떨어진 카메라로 부터 겹침 공간이 있는 곳에서 발견된 객체를 보고 시간마다 만들어진 트랙을 통해 객체 위치 파악을 해내는데 카메라

라 노드들 사이와 서버간의 시간동기화가 어긋나기 시작하면 위치파악을 제대로 수행하기가 힘들어진다.

2.2 비겹침 카메라

다수의 카메라 센서를 사용해서 감시시스템을 구축할 때 감시되고 있는 객체가 어떤 화면에서 보이더라도 그 객체의 일치점을 찾아야 지속적인 추적이 가능하다. 이전 연구를 보면 겹침 화면을 이용해 다른 카메라에서 객체의 일치점을 찾을 수 있었다. 하지만, 비겹침 화면에서는 다음과 같은 문제점이 있다. 첫째, 현재 관찰중인 객체가 시간과 공간이 분리되어 다른 카메라에 나타나게 되면 이미 사용된 시공간 값을 다른 카메라에서는 사용할 수가 없다. 둘째로, 특정 카메라에서 바라본 외양이 다른 카메라에서 관찰되면 그 외양이 매우 달라 보여서 일치점을 찾는 것이 매우 어렵다[8].

Knight[9]의 경우에는 카메라의 Field of view line(FOV line)을 이용해서 카메라간 떨어진 거리를 이용해 이전 카메라에서 관찰된 객체가 도착할 위치를 예측했다. 하지만, 중간에 객체가 떨어진 화면에서 멈추는 등 예측할 수 없는 행동이 발생하면 위치 예측이 어려워진다. [10]의 연구를 보면 사람의 반복적인 움직임을 관측하여 특정 인물이 특정 위치에서 들어오고 나가는 것을 이용해 다른 카메라에 나타나는 그 객체가 같다는 것을 계산하여 미리 예측하여 추적했다. 이 연구에서는 어느 정도 시간동안 객체의 움직임에 관한 데이터가 축적되어야 정확한 구분이 가능하다. [11]의 연구를 보면 계층적인 사람모델을 만들어 객체의 정보를 서버가 기록하고 있다가 여러 대의 카메라 간에 실시간으로 정보를 전달하여 인식할 수 있도록 네트워크를 형성했다. 그리고 [12]의 연구도 비겹침 카메라를 사용하고 연관된 픽셀의 위치, 색상, 그리고 각 발견된 객체의 경사도를 이용해 구분한다. 그 자료들은 비디오 저장소에 기록되고 Query-By-Example 방식을 이용해서 링크에 접근해서 다중객체 예시를 카메라에서 분산처리를 하는데 이용한다. [11], [12]의 연구는 객체를 처리할 때 마다 서버에 기록된 데이터를 조사해서 정보를 다시 받아야 보내 주어야만 한다. 더해서 네트워크 구조를 구성할 때 서버와 카메라가 직접적으로 연결을 했다. 그렇기 때문에 CC2410을 이용하고 Ad-hoc 네트워킹을 이용하는 본 논문의 무선센서네트워크 실험에서는 대량의 자료를 빠른 시간대로 서버와 주고받게 하는 것은 어렵다.

3. Surveillance Video Sensor System 구성

3.1 하드웨어 & 소프트웨어 구조

카메라 보드는 실험을 위해서 X-Hyper270TKU를 이용해서 구성했다. 프로세서인 PXA270을 사용하기 위해서 인데 520MHz의 속도로 640X480의 이미지를 background-subtraction으로 처리하는데 0.3-0.8sec의

성능[4]을 보여주었기 때문이다. 그러므로 이미지 사이즈가 작으면 더 빠른 작업을 할 수 있기에 속도발전 가능성을 보여준다. 이 것 보다 낮은 사양의 CPU는 다른 비전 알고리즘을 처리하는 능력이 부족하기 때문에

면 되기 때문에 많은 개발 시간을 줄여 줄 수 있다. 그래서 OS인 SenWeaver가 제공하는 표준 MAC프로토콜과 Ad hoc 통신이 가능한 라우팅 프로토콜을 이용해 카메라 센서 어플리케이션을 구성할 수 있다.

표 1 무선센서노드/카메라노드 소프트웨어 구조

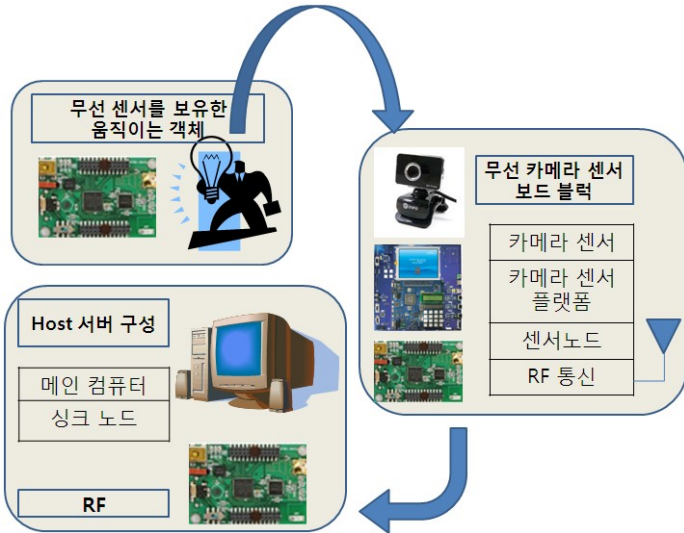
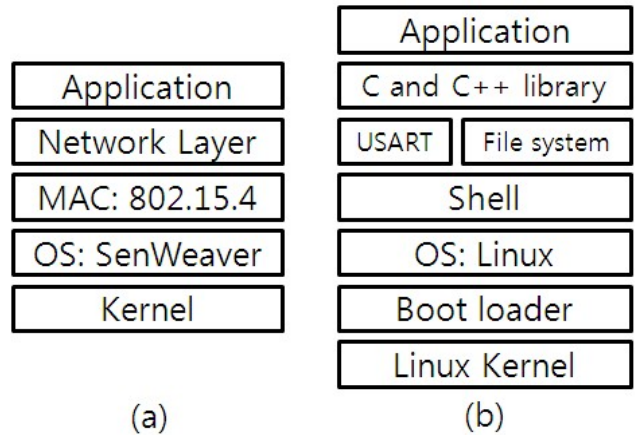


그림 1 하드웨어 구조 도식화

PXA270 보드를 실험에 사용하게 된 가장 큰 이유가 되었다. 무선센서노드에 ATmega128L을 사용하는 보드를 선정했다. 이것을 선택한 주된 원인은 802.14.5 표준 MAC, OS, 그리고 routing protocol, 등을 설치하면 이미 용량이 90KB가까이 된다. 그렇기 때문에 MSP430를 사용한 Tmote같은 구조는 전력 효율이 더 높지만 자체 플래쉬 메모리가 64Kb로 부족하기 때문에 외부 메모리를 사용해야 한다. 그런 이유로 자체 메모리가 풍부한 ATmega128L을 사용하는 노드를 사용하게 되었다.

Host 서버인 메인 컴퓨터와 일반적으로 게이트웨이를 사용하지만 무선센서노드중의 하나를 싱크노드가 USB 선을 통해 서버 컴퓨터와 통신이 되기 때문에 무선센서노드중에 하나를 이용했다. 그 이유는 FDTI칩을 통해 usb-to-serial 통신으로 싱크노드로 들어오는 정보가 메인서버에서 바로 확인이 가능하기 때문이다. 받아들인 데이터의 처리는 서버를 중심으로 이루어진다. 노드에서 보내오는 정보는 노드ID, 색상, 노드좌표값만을 받아오게 된다. 그 값들이 들어오는 순서대로 서버화면에 추적되는 상황이 나타내지기 때문에 시간동기화가 필요 없이 서버컴퓨터의 시간이 중심이 된다. 그림 1을 보면 일단 일반적인 웹캠과 카메라 센서보드 그리고 센서노드가 RF 통신을 통해서 서버인 메인 컴퓨터와 통신을 하는 게 주를 이룬다. 그리고 무선센서노드를 보유한 관계자가 등장하면 무선센서노드+카메라센서보드+카메라 모듈이 포함된 보드와의 통신이 이루어지고 그 결과들이 서버 컴퓨터로 전달되는 순서로 통신이 진행된다.

표 1(a)에서와 같이 센서 플랫폼을 사용하면 기본적으로 하드웨어 커널, OS, MAC, 라우팅 프로토콜까지 제공하고 프로그래머가 원하는 어플리케이션만 구성하

표1(b)에서와 같이 카메라센서보드의 소프트웨어를 다음과 같이 구성한 가장 핵심이 되는 이유는 일단 리눅스가 Open source이기 때문이다. 그래서 WinCE와 달리 별도의 개발비용 없이 무료로 차후의 개발을 위해 개발 모듈을 변경할 필요가 있을 때도 별도의 추가 비용이 들지 않는다. 그리고 C/C++ library와 특히 비전 알고리즘을 위한 OpenCV library 설치가 용이하다. 더해서 PXA270을 메인 프로세서로 선택했는데 리눅스에서 이 프로세서를 사용하기 위한 정보가 풍부했기 때문이다.

3.2 카메라 배치와 객체 이동 시에 고려해야 하는 사항

그림2의 왼쪽그림의 경우 카메라가 왼쪽 방향으로 향하고 있다. 그 상태에서 전방에 객체가 나타나고 뒤에 노드를 가진 자가 동시에 나타나면 카메라가 나타나 O가 노드를 보유한 객체로 인식되므로 카메라 배치를 할 때 항상 벽을 대고 있거나 구석진 곳에 설치해야 한다.

가운데 그림에서 N2의 경우 시간의 차이를 두고 접근하게 되므로 최후에는 카메라 앞에 잠시 들어났다 N0와 N1이 지나간 다음 나타나므로 문제가 없다. 하지만, N0과 N1이 동시에 카메라에 나타났을 때를 고려해야 한다. 알고리즘으로 일단 노드 ID2개와 색상정보2가지 값을 가지고 녹색 박스로 처리했다가 다시 검사할 기회가 생겼을 때 최종적으로 하나를 선택하는 방법으로 서버에서 자료를 보유한 후 처리하게 된다.

마지막 그림에서도 노드 3가지가 같은 시간에 화면에 나타나서 보이는 문제점을 나타낸다. 무선센서노드를 인증으로 사용하기 때문에 카메라 화면에 나타나서 인증을 시작하게 되고 3가지가 동시에 나타나면 어느 객체가 관계자인지 아닌지를 알 수 없다. 이 경우는 3가지 객체가 모두 노드 ID와 색상정보를 가진 후 황색 박스로 구성된 다음 기회가 있을 때 다시 비교 한다.

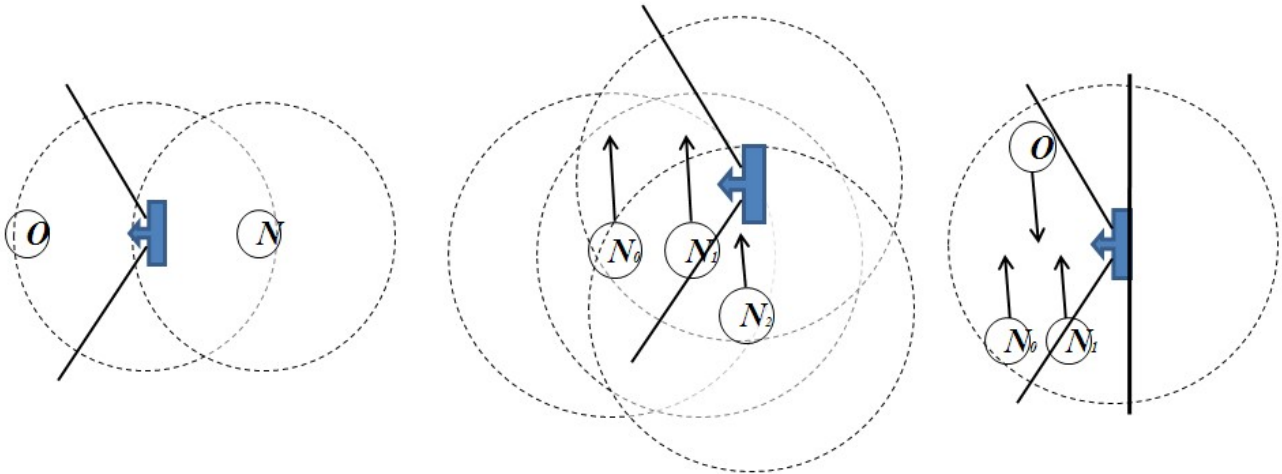


그림 2예상 Topology(O: 움직이는 객체, N:움직이는 객체 + 무선센서노드 보유자)

3.3 알고리즘

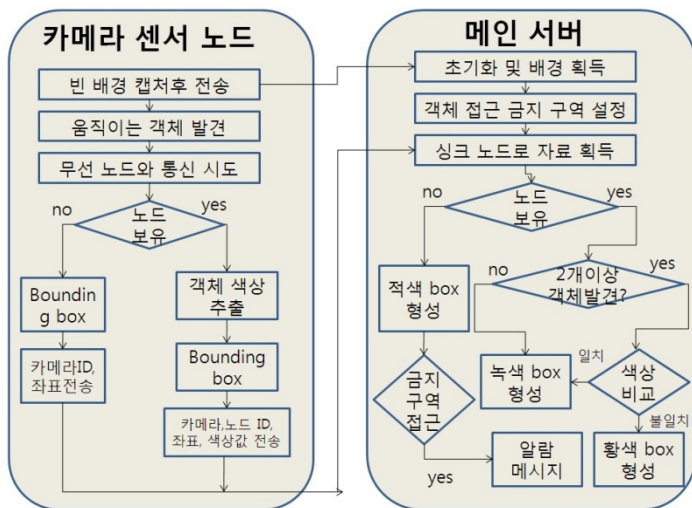


그림 3 flow chart

본 카메라 감시시스템의 알고리즘은 무선센서네트워크의 부족한 전송능력으로 프레임마다 나타나는 모든 캡처된 장면을 보내는 게 아니라 추적에 필요한 정보만을 우선 전달하게 된다. 그래서 최초 카메라 센서 노드에서 서버 컴퓨터로 비관계자 출입금지 표시 구역을 위해 각 카메라 모듈에서 움직임이 없을 때의 배경 캡처 화면을 전송하는 것 초기화 작업이 시작된다.

초기화 작업이 끝나면 움직이는 객체를 background subtraction을 통해 발견해 낸다. 객체가 발견되면 무선센서노드를 소유하고 있는지를 검사해서 bounding box를 이용해 중심 좌표를 구해서 카메라ID와 좌표를 서버로 보내게 된다. 만약 무선센서노드를 보유한 관계자일 경우에는 관계자의 정확한 파악을 위해 카메라ID, 센서노드ID, 좌표, 그리고 추출된 객체의 색상분포 값을 서버로 보낸다. 화면상의 움직이는 객체만 캡처하기 때문

에 calibration같은 과정은 요구되지 않는다.

메인서버에서는 싱크노드를 통해 자료가 획득되면 노드를 보유하지 않은 좌표정보는 화면상에 적색 박스로 나타내어 움직임을 추적한다. 추적하다가 이전에 설정해 둔 금지구역으로 다가오면 오퍼레이터에게 알람 메시지를 전달해주게 된다. 한명의 노드 보유자가 나타나면 그 발견된 객체는 녹색 박스를 형성해서 화면상에 어디에 위치했는지를 나타내준다. 만약 우연히 동시간대에 다양한 객체가 나타난다면, 이럴 경우 어느 객체가 노드를 가지고 있는지 구분이 안 되기 때문에 일단 노드 값과 색상 값을 모두 서버에서 보유해서 저장한다. 그리고 완벽히 구분이 안 되는 상태기 때문에 그런 객체들은 황색 박스로 구분해 두게 된다. 그런 후 추후에 다른 카메라에서 분리가 되었을 때 노드 보유자와 비 보유자로서 다시 구분하게 되는 과정을 거치게 된다.

일단 서버에서 각 화면을 배치될 때 카메라 ID에 따라 화면을 분배한다. 그리고 분배된 화면 내에서 발견된 객체들의 추적중인 박스들이 보이게 되는데 그 때 노드 보유 객체는 노드ID와 좌표값이 서버에 도착한 시간에 따라 객체의 상태에 따라 적색, 녹색, 황색으로 화면에 나타내어 객체들을 계속 추적한다.

4. 결론 및 향후 연구과제

이렇게 구성된 시스템을 이용해서 카메라 네트워크에서 보이는 모든 객체에 대해서 지속적인 추적은 이루어지지 않지만, 무선센서노드를 부여한 노드를 추적함으로써 관계자와 비관계자를 명확히 구분할 수 있다. 두 카메라로 시간에 따라 트랙을 형성해 위치를 인식하는 방법은 시간동기화가 필수라서, 이 방법 대신 특정영역을 두어 수상한 인물의 접근을 알 수 있게 구성되어 졌다. 이 구성은 공항이나, 대학교 같은 곳에서 사람이 자주

돌아다니나 관계자만 들어 갈 수 있는 장소에서 감시카메라 오퍼레이터가 관계자 외 출입금지 구역 근처의 수상한 인원을 빨리 인지하는데 도움을 줄 수 있을 것이다.

여전히 연구해야 할 과제가 남아 있다. 많은 카메라 추적시스템과 같이 이 시스템도 같이 맞물려 활동하는 객체를 구분하는 것은 피하기 어려운 문제이다[13]. 그래서 이것을 해결하기 위해 황색 박스로 남겨두어 구분시켰다. 그리고 이 구성은 외부 전원을 사용하는 보드를 사용하였기 때문에 향후에는 카메라 센서 보드를 제작해서 배터리 파워로 실험을 할 계획이다. 그리고 카메라 정보 전달을 위해 딜레이를 짧게 주었는데 배터리 효율을 위해 에너지 효율적인 프로토콜 설계가 필요하다. 또한 일단 센서 네트워크 프로토콜 보다는 카메라 센서 네트워크를 이용하는 네트워크에서는 움직이는 객체를 발견했을 때 서버로 정보 전송을 위한 보다 빠른 구조의 Point-to-Point 프로토콜 개발이 필요하다. 이 프로토콜의 이용은 차후 좌표만 보낸 지금과 같은 경우를 적어도 한 두장의 객체 사진을 전송할 수 있게 환경을 개선할 수 있을 것으로 보인다.

참고문헌

- [1] Weiser. M, "Hot Topics: Ubiquitous Computing", IEEE Computer, 1993.
- [2] Stankovic J. A., "When sensor and Actuator networks cover the world", ETRI Journal, Vol. 30, Num. 5, October 2008.
- [3] Yarvis. M, etc, "Exploiting Heterogeneity in Sensor Networks", IEEE, 1999.
- [4] Chen. P, Oh. S.H., etc, "CITRIC: A Low-Bandwidth Wireless Camera Network Platform", ICDSC, 2008.
- [5] M Rahimi, R Baer, OI Iroezi, JC Garcia, etc, "Cyclops: in situ image sensing and interpretation in wireless sensor networks", In Embedded Networked Sensor Systems, 2005.
- [6] Kulkarni P, Ganesan D, etc, "SensEye: A Multi-tier Camera Sensor Network", ACM, 2005
- [7] Kim. K.H., Hong. W. K., etc, "Low Cost Time synchronization Protocol for Wireless Sensor Network", IEICE Transactions on Communications, Vol. E92-B4, pp. 1137-1143, Apr. 2009.
- [8] O. Javed, M.Shah, *Automated Multi-Camera Surveillance: Algorithms and Practice*, Springer Science+Business Media, 59, 2008.
- [9] O. Javed, Z Rasheed, O. Alatas, M Shah, "KNIGHTM: A REAL TIME SURVEILLANCE SYSTEM FOR MULTIPLE OVERLAPPING AND NON-OVERLAPPING CAMERAS", in proc. Of the fourth ICME, 2003.
- [10] O. Javed, Z. Rasheed, K. Shafique, "Tracking Across Multiple Cameras With Disjoint Views", in proc. Of the IEEE International Conference. Computer vision, pp. 952-957, 2003.
- [11] 이윤미, 이경미, "모델기반 다중 사람추적과 다수의 비겹침 카메라를 결합한 감시시스템", 정보과학회, Vol. 12, No. 4, 2006
- [12] I. Cohen, Ma. Yunqian, B. Miller, "Associating Moving Objects Across Non-overlapping Cameras: A Query-by-Example Approach", 2008 ICDSC, pp. 566 - 571, May. 2008.
- [13] S. Solo, W. Heinzelman, *A Survey of Visual Sensor Networks*, Advances in Multimedia, Vol. 2009, pp. 21, 2009.