

<http://dx.doi.org/10.7236/JIIBC.2013.13.1.181>

JIIBC 2013-1-25

움직임 감지용 센서와 정지 영상을 이용한 감시 시스템

A Surveillance System Using Images and Movement Detection Sensors

차충용*, 김상철**

Zhong-Yong Che, Sangchul Kim

요약 일반적인 영상감시시스템은 비디오 녹화 및 전송 방식을 채택하고 있기에 특성상 다량의 데이터를 전송하고 저장하여 시스템의 부담이 크다. 하지만, 쓰레기 불법투기나 불법 주차현장의 포착 및 기록을 같은 용도에서는 비디오가 아닌 정지영상을 이용하는 감시가 충분할 것이다. 본 논문에서는 움직임 센서와 정지 영상을 이용한 영상감시시스템을 제안한다. 우리의 시스템에서는 움직임 이벤트의 발생을 센서값 변화로 인식한 후, 현장을 기록하는 정지영상을 캡처하고 원격지의 데이터베이스에 저장하는 하는 기능을 제공한다. 따라서 제안된 시스템은 기존 비디오 기반의 감시 시스템과 비교해서 감시 상황의 발생을 보다 신속하면서도 정확하게 인식할 뿐만 아니라 처리하는 데이터의 양을 현저히 줄이는 장점이 있다. 또한, 우리의 시스템은 에이전트 기반으로 동작하기에, 이후 새로운 모듈을 추가하거나 기존 모듈을 수정하는 작업을 쉽게 할 수 있도록 한다.

Abstract Since conventional image surveillance systems employ methods for video recording and transmission, a huge amount of data is transferred and stored so that those systems are overloaded. However, for capturing and recording the scenes of illegal trash throwing and unpermitted parking, it is sufficient to use a surveillance system using images. In this paper, we propose a surveillance system using images and motion detection sensors. Our system recognizes the occurrence of movement events through changes of sensors, captures still images of the region under surveillance, and stores them into the database at a remote site. The system proposed herein provides a functionality to detect the occurrence of those events more accurately and faster than previous video-based systems, and has an advantage of reducing the amount of data significantly. Also, our system is agent-based, it enables us to add new modules or modify existing modules easily later.

Key Words : System, Image-Based, Motion Detection Sensor

1. 서론

영상감시시스템은 보안 및 안전사고 예방을 위하여 최근 많은 기관에서 도입해 운영하고 있다. 감시 현장의 상황을 관리자가 직접 모니터링 하면서 나중에 증거 또

는 검증 자료로 활용하기 위해서, 이들 시스템은 현장에서 캡처한 비디오를 관리자의 PC로 전송하면서 저장하는 기능을 제공하고 있다. 감시 현장과 관리자 또는 데이터베이스와 거리가 먼 경우에는 유무선 네트워크를 사용하게 된다.

*준회원, 한국외국어대학교 컴퓨터공학과

**정회원, 한국외국어대학교 컴퓨터공학과

접수일자 2012년 12월 21일, 수정완료 2013년 1월 21일

게재확정일자 2013년 2월 8일

Received: 21 December 2012 / Revised: 21 January 2013 /

Accepted: 8 February 2013

**Corresponding Author: kimsa@hufs.ac.kr

Dept. of Comp. Sci. & Engr., Hankuk Univ. of Foreign Studies.

일반적인 영상 감시시스템에서는 CCTV나 IP 카메라 등을 이용하여 현장상황을 비디오로 녹화하고 경우에 따라 사용자가 직접 모니터링하고 있다. 이러한 방식의 시스템은 비디오의 특성상 방대한 양의 데이터를 전송하고 저장하게 되므로, 통신네트워크와 데이터베이스 서버에 부담을 주게 된다. 하지만 확보된 비디오데이터에서 감시의 대상이 되는 이벤트가 발생하는 장면은 전체 데이터양에 비해서는 일부에 불과하다. 따라서 장기간 녹음된 비디오데이터로부터 이러한 이벤트를 수작업으로 추출하기 위하여 많은 시간과 노력이 필요하다. 또한 비디오의 화질에 따라 이러한 이벤트를 자동으로 인식하는 영상처리 알고리즘의 정확도는 높지 않을 수 있다.

영상 감시시스템을 도입하는 대표적인 분야로서는 군용 경계, 공항 감시, 공장 생산 라인 감시, 교통 감시 및 일상생활에서의 안전이나 보안유지 위한 감시를 들 수 있다. 다른 분야와 달리 생활주변의 현장 감시에서는 관심 있는 이벤트가 자주 발생하는 경우가 많기 때문에 비디오가 아닌 정지영상만으로 현장사진 확보 및 관리자 확인이 가능할 것이다. 이러한 현장의 예로서는 쓰레기 무단 투기가 발생하는 지점, 불법주차 단속 구역, 동물 무단 침입을 억제 하고자 하는 농장구역 등을 들 수 있다. 앞에서 언급한 정지 영상을 이용한 감시시스템에서는 관심이 되는 이벤트가 발생하는 시점을 자동으로 인식하는 기능이 필수적이다. 이런 자동인식은 다양한 센서를 이용함으로써 가능할 것이다.

본 논문에서 우리는 움직임 센서값 변화를 이용한 정지영상 기반의 감시시스템을 제안한다. 센서 값의 변화를 감시 이벤트의 발생신호로 판단하고, 이벤트 발생순간의 현장상황을 정지영상으로 캡처하여 저장하고 관리자에게 알린다. 본 시스템에서는 적외선 센서, 거리 센서 등의 다양한 센서를 움직임 감지용으로 사용하고 있다. 본 시스템의 또 다른 주요 특징은 블랙보드(BLACKBOARD) 및 에이전트(AGENT)기반의 시스템 구조이다. 에이전트란 자신의 업무를 자율적으로 수행하는 프로세스이고 블랙 보드는 에이전트들 간의 자료 공유를 위한 버퍼 객체이다. 이 구조는 특성상 시스템 사용 중에 영상객체 인식기와 같은 새로운 소프트웨어 모듈을 추가하거나 기존 모듈을 수정하는 것을 편리하게 한다.

본 시스템을 구성하는 센서 노드와 서버 시스템간의 자료 전송은 LAN 또는 ZigBee 무선네트워크를 통하여 이루어진다. ZigBee 무선네트워크를 지원하는 이유는 본

시스템을 야외와 같이 배선 작업이 어려운 지역에 설치하거나 또는 센서 노드 위치를 쉽게 변경하는 환경에서도 편리하게 사용하기 위함이다.

기존 연구에서 우리의 시스템과 관련되는 정지영상기반의 현장 감시 시스템에 대한 연구는 다소 발표되어 왔다^[1,2,3,4]. [1][2]에서 제안하는 시스템은 현장을 지속적으로 캡처하면서 얻은 두 개 프레임의 서로 비교하는 카메라 보드와 영상저장 및 확인기능을 제공하는 관리서버로 구성된다. 카메라 보드에서는 두 개 프레임이 다른 경우를 움직임 발생으로 감지하고, 현장 사진과 함께 그 사실을 관리서버에 알려준다. 이러한 시스템에서는 지속적인 정지영상 캡처와 프레임비교를 위한 영상처리가 필요하기에 고성능의 카메라 보드가 필요하다. 또한 프레임간의 차이만을 이용하기에 조명의 변화와 같은 일반적인 감시 이벤트와 무관한 상황을 서버에 보고하는 오동작이 자주 발생한다.

[3]의 감시시스템에서는 정지영상 캡처용 고성능 카메라 한 대 와 저성능 카메라 두 대를 함께 장착한 카메라 보드를 이용한다. 저성능 카메라들은 일정한 간격을 두어 설치되어 있다. 카메라 보드에서는 지속적으로 두 카메라의 영상이 같은 지를 비교하는데, 서로 다른 경우에 고성능 카메라로 현장 사진을 확보한다. 이 시스템은 [1][2]보다는 높은 움직임 감지율을 제공하지만 여전히 고성능의 카메라 보드가 필요한 단점이 있다. [4]의 감시 시스템은 무선 네트워크를 이용하여 주기적으로 센서보드의 영상을 관리 서버로 전송한다. 감시 이벤트의 발생 유무와 관계없이 영상을 전송하기 때문에 네트워크에 부하가 많이 걸리는 단점이 있다.

유비쿼터스 기술이 발전하면서, ZigBee와 같은 무선 센서네트워크 기술을 이용하여 정지영상을 전송하는 연구가 현장감시를 포함한 여러 분야에서 활발히 진행되고 있다^[2,5]. 이러한 기존연구에 따르면, ZigBee 무선네트워크의 성능은 비디오 스트리밍은 어려우나 일정한 시간 간격을 가진 정지영상 전송에는 충분하다. 시스템 확장성을 위해 에이전트 기반의 구조를 갖는 시스템에 대한 연구는 헬스케어^[6], 환경감시^[7] 등의 다양한 분야에서 진행되어 왔다. 하지만, 우리의 조사에 의하면 무선 센서 네트워크와 정지영상을 지원하는 현장 감시 시스템은 거의 발표되고 있지 않다.

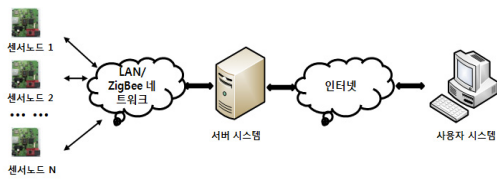
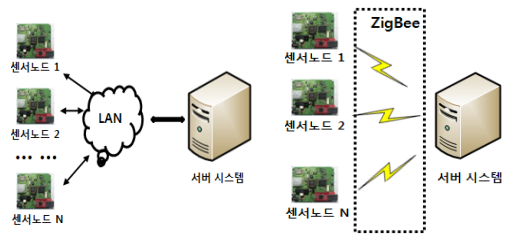
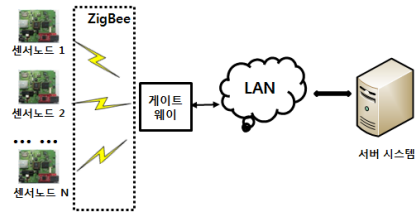


그림 1. 전체 시스템 구조
Fig. 1. The Overall System Structure



(a) LAN 타입 (b) ZigBee 타입



(c) 복합 타입

그림 2. 센서 노드와 서버 시스템간의 연결 유형
Fig. 2. Types of Connections between Sensor Nodes and the Server System
(a) LAN-type (b) ZigBee-type (c) Combined type

II. 시스템 전반

우리가 제안하는 시스템은 그림 1과 같이 3개 부분으로 구성 된다. 이들은 현장에 설치된 데이터 수집 및 감시를 수행하는 센서 노드 부분, 데이터를 저장하고 다양한 업무를 처리하는 서버 시스템, 현장을 모니터링하고 각종 관리업무 수행을 지원하는 사용자 시스템이다.

다양한 설치 환경을 지원하고자 3가지 유형의 센서 노드와 서버 시스템간 연결 방식을 지원한다. 그림 2는 이들 3가지 유형을 도식적으로 보여주고 있다. 그림 2(a)에서 센서 노드들은 LAN을 통하여 서버 시스템에 접속되고, 서버는 각 센서 노드들을 별도로 관리하게 된다. 이는 LAN 구성이 가능한 지역에서 많은 수의 감시현장으로부터 정지영상을 고속으로 수집하여야 하는 경우에 적합하다. 예를 들면 LAN이 갖추어진 건물 내 주차장의 차량 입출입 및 정차여부 감시에 활용할 수 있다.

그림 2 (b)는 센서 노드와 서버 시스템간을 ZigBee 무선네트워크로 연결하는 것이다. 이 구조는 감시시점까지 LAN이 설치되어 있지 않고 감시기간이 상대적으로 짧아 LAN 포설이 번거로운 환경에서 적합하다. 예를 들면 건물이나 집 주변 지역의 동물 출현 또는 골목에서의 쓰레기 투기를 감시하는 경우에 적합할 것이다.

그림 2 (c)는 센서 노드들은 게이트웨이라고 불리는 중계기와 ZigBee 무선네트워크로 통신하고, 중계기는 서버 시스템과 LAN으로 연결되는 구조를 갖는다. 이 구조는 감시 현장 주변 지역까지는 LAN이 포설되어있지만 세부 감시시점까지는 여건상 LAN 배선 작업이 어려운 환경에 적합할 것이다.

III. 센서 노드 및 사용자 시스템

1. 센서 노드

그림 3에서와 같이 본 논문에서 제안하는 시스템의 센서 노드는 하드웨어 플랫폼과 운영 소프트웨어로 구성된다. 하드웨어 플랫폼으로서 우리는 본 시스템의 기능에 가장 적합하다고 판단한 셀로코(주)의 SSN-1000 보드를 사용한다.

센서 노드의 하드웨어는 중앙처리장치인 8051 계열의 MCU외에 LM2455 기반의 ZigBee 모듈, W3150 TCP/IP 칩 및 센서부로 구성 된다. MCU에는 JPEG 압축 모듈이 내장되어 고속으로 해상도 640x480 (VGA급) 및 384x288 (CIF급) 화질의 압축된 이미지를 출력할 수 있다. ZigBee 모듈과 TCP/IP 칩은 외부 장치와의 통신을 지원한다. 센서부는 움직임 감지용 센서, 카메라 모듈 및 보조용으로 사용할 수 있는 온도도 센서로 구성된다. 움직임 감지용 센서로는 적외선 센서, 거리 센서 및 광센서가 지원된다. 적외선 센서는 사람이나 동물이 방출하는 적외선 양을 감지하는 하는 센서이고, 거리센서는 센서 앞에서 일정 거리내에 위치한 물체와의 거리를 측정하는 센서이다.

광센서는 스위치를 끄고 켜는 경우와 같이 빛의 양 변화를 감지할 수 있다.

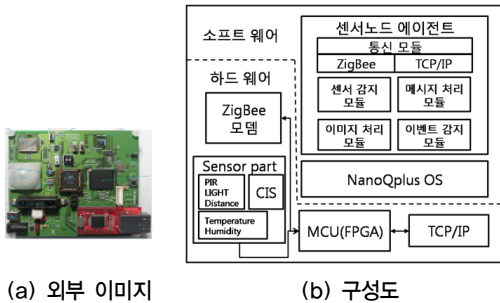


그림 3. 센서 노드
Fig. 3 The Sensor Node (a) The Appearance (b) The Block diagram of the Structure

운영 소프트웨어는 NanoQplus OS와 센서 노드 에이전트로 구성된다. 센서 노드 에이전트는 센서 노드의 동작초기에 서버 시스템에 자신의 존재를 알린다. 그 후 서버 시스템과 접속을 유지하면서 센서 노드에 운영과 관리를 담당하는 역할을 수행한다. 이 에이전트는 센서 감지 모듈, 메시지 처리 모듈, 이미지 처리 모듈, 이벤트 감지 모듈과 통신 모듈로 나누어진다.

센서감지 모듈은 센서값을 읽어 들여 이벤트 감지 모듈에 전달하는 역할을 담당한다. 메시지 처리 모듈은 서버로부터 받은 명령어를 해석하여 적절한 조치를 취하거나, 캡처한 이미지 또는 센서값 등의 데이터를 메시지 형태로 서버 시스템에게 전달하는 역할을 수행한다. 서버로부터 전달 받는 명령어에는 센서 노드 상태확인, 이미지 캡처, 이벤트 감지조건 설정 등 있다.

이미지 처리 모듈의 주요 업무는 카메라 모듈을 이용하여 현장사진을 이미지 형태로 캡처 하는 역할이다. 일반적으로 이미지 캡처는 이벤트 모듈의 캡처 신호를 수신하거나 관리자가 임의의 시점에 서버 시스템을 통해 보낸 캡처 명령어를 받으면 수행된다.

이벤트 감지 모듈은 센서값들이 감시 조건을 만족하는 지를 지속적으로 검사하고, 만족하는 경우에는 이미지 캡처 신호를 발생시키는 역할을 담당한다. 전체 감시 조건은 센서별로 설정된 개별 감시조건들을 AND 또는 OR과 같은 관계연산자로 연결한 관계식으로 정의된다. 물론 상황에 따라서는 센서 노드의 센서들 중에서 감시 조건을 설정하지 않은 것도 있을 수 있다. 센서 s 에 대한 감시 조건은 다음같이 정의된다:

$$CHANGE_RATIO (V_s) > T_s \tag{1}$$

V_s 는 센서값이고 $CHANGE_RATIO(\cdot)$ 는 기준값에 대한 센서값의 변동률을 계산하는 연산이다. 기준값이란 센서를 설치한 후 정상시의 센서값을 나타낸다. T_s 는 실험적으로 결정되는 임계치로서, 위의 조건식은 센서값인 V_s 의 변동률이 T_s 보다 크면 만족하게 된다. 물체의 움직임이 발생하는 경우 관련 센서값의 변화가 있게 되어 감시조건이 만족된다.

2. 게이트웨이

게이트웨이의 하드웨어는 센서 노드의 하드웨어에서 센서부분을 제거한 것과 동일하다. 게이트웨이의 주된 소프트웨어는 통신모듈, 메시지 처리 모듈, 센서 노드 관리 모듈 등으로 구성된다. 통신 모듈은 ZigBee망이나 인터넷을 통해 센서 노드와 서버 시스템과의 통신을 제공한다. 메시지 처리 모듈은 서버 시스템과 센서 노드들간의 메시지를 중계하는 역할을 담당한다.

그림 2 (b)에서 보듯이 서버 시스템은 센서 노드와 직접 연결되어 있지 않기 때문에 서버 시스템 측에서는 센서 노드의 ZigBee 관련 정보 (ZigBee 주소)를 알기 어렵다. 따라서 센서관리 모듈의 주된 기능은 서버에서 부여한 센서 노드의 ID와 해당 ZigBee주소를 매핑하는 것이다. 또한 센서관리 모듈은 감시업무를 담당하는 센서 노드들의 목록을 관리하면서, 새로운 노드들의 가입 또는 기존노드의 이탈을 서버에게 보고한다.

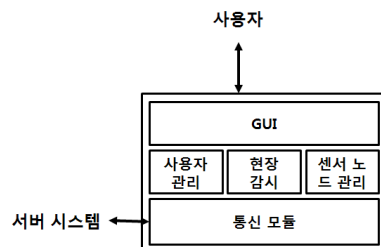


그림 4. 사용자 시스템의 구조
Fig. 4. The Structure of the User System

3. 사용자 시스템

사용자 시스템은 현장 감시의 책임자를 위한 것으로서, 현장을 모니터링하고 시스템 전체의 운영을 관리하는 기능을 제공한다. 본 시스템은 그림 3과 같이 GUI 모

들과 서버 시스템과의 통신 모듈 외에 3개의 주요 모듈로 구성되어 있다. 이들 모듈은 기본적으로 사용자의 요구를 서버 시스템으로 보내고 서버 시스템의 처리 결과를 사용자에게 제시하는 방식으로 동작한다.

사용자 관리 모듈은 신규 사용자의 등록, 비밀번호 변경, 사용자의 상태 조회 등과 같은 업무를 지원한다. 센서 노드 관리 모듈은 현장에 배치되어 운영 중인 센서 노드들의 상태 조회 및 동작 제어, 신규 센서 노드 등록 등의 업무를 지원한다. 동작 제어에는 센서 노드의 동작 여부 확인, 감시 시작 및 중지, 현장 이미지의 즉석 캡처 후 전송, 센서 노드의 환경 변수 설정 및 조회 등이 포함된다.

현장 감시 모듈의 주요 기능은 센서 노드로부터 전달되어오는 현장 이미지를 보여주거나 저장된 이미지를 조회하는 것으로, 사용자는 이 모듈이 제공하는 화면을 통해서 감시 현장을 모니터링하게 된다. 특정 센서로부터 움직임 이벤트 감시 사실과 함께 현장 이미지가 전송되면, 주의를 의미하는 팝업창을 띄우고 경고음을 발생시키는 기능도 제공한다. 이 기능은 사용자가 여러 현장을 동시에 감지하는 경우와 잠시 감시 화면에서 눈길을 떼 경우 사용자에게 움직임 발생된 현장에 즉각 주목하도록 하는 효과가 있을 것이다.

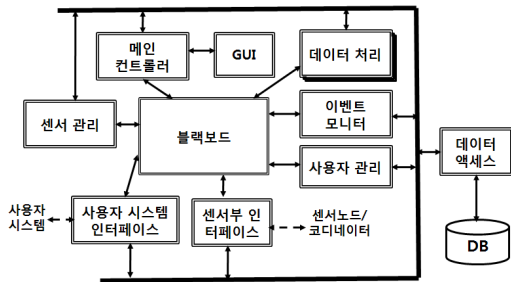


그림 5. 서버 시스템의 구조
Fig. 5. The Structure of the Server System

IV. 서버 시스템 구조 및 동작원리

1. 에이전트 기반의 구조

서버 시스템은 그림 5와 같이 여러 에이전트들로 구성되는데 이들의 역할은 다음과 같다.

- 블랙보드: 에이전트들 간의 정보교환을 담당하는 버퍼 역할을 하는 모듈이다. 서버 시스템 내에 있는 여러

종류의 에이전트들이 서로 협조하여 어떠한 업무를 처리하기 위해 중계자 역할을 한다. 서버 시스템의 시작 초기에, 모든 다른 에이전트들은 자신의 존재와 자신이 담당할 수 있는 태스크 타입을 본 모듈에 보고하게 된다.

- 메인 컨트롤러: 서버 시스템의 전반적 동작을 제어하고 수행하는 역할을 한다. 시스템을 초기화하고 에이전트들에 대하여 수행 시작, 종료 및 상태 감시 등을 하는 에이전트이다.
- 이벤트 모니터: 움직임 감시와 관련되는 다양한 업무를 처리한다. 이런 업무에는 감시 조건 저장 및 조회, 센서 노드로 감시조건 전달, 움직임 이벤트 발생 시 사용자 측에 그 사실을 전달하고, 센서 노드로부터 수신된 감시 현장의 이미지저장 등이 포함된다.
- 사용자 관리: 신규 사용자의 등록, 비밀번호 변경 등의 사용자 정보를 관리하는 역할을 한다.
- 센서 관리: 서버 시스템에 연결되어 운영 중인 센서 노드를 관리하는 역할을 담당한다. 이들 업무에는 센서 노드 등록 및 삭제, 센서 노드 위치 수정, 센서 노드의 동작 등이 포함된다.
- 사용자 시스템 인터페이스: 사용자 시스템과의 모든 메시지교환은 본 에이전트를 통하여 이루어진다. 서버 시스템과 사용자측 시스템간의 통신 연결을 담당하고, 사용자측으로부터 전달되는 메시지를 태스크 형태로 전환하여 블랙보드에 포스팅하거나 또는 타 에이전트의 요구에 따라 관련 메시지를 사용자측에 전달하는 역할을 한다.
- 센서부 인터페이스: 본 에이전트는 서버 시스템과 센서 노드와의 메시지 교환을 담당한다. 따라서 타 에이전트들이 센서 노드에 보낼 명령어는 본 에이전트를 통하여 이루어진다. 또한 센서 노드로부터 전달되는 메시지를 태스크 형태로 변경하여 블랙보드를 통하여 관련 에이전트에게 처리를 요청하게 된다.
- 데이터 처리: 우리의 시스템이 실제로 운영되는 경우에, 운영 목적에 따른 추가적인 데이터 처리에 대한 요구가 있을 수 있다. 예를 들어 불법 주차감시의 경우에 현장사진의 분석을 통하여 차 번호판 인식이 필요할 수 있다. 이와 같은 요구가 있을 때에는 해당 업무를 수행하는 새로운 에이전트를 구현하여 추가함으로서 본 시스템을 쉽게 확장할 수 있다.

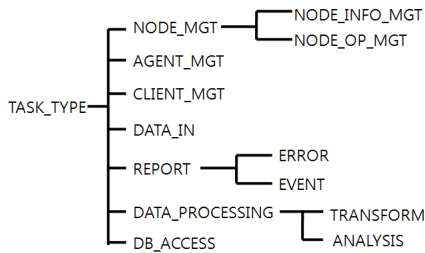


그림 6. 태스크 타입 계층도
Fig. 6. The Hierarchy of Task Types

2. 에이전트 동작제어

그림 6은 우리 시스템에서 사용되는 태스크 타입들과 그들 간의 계층구조이다. 에이전트들은 이 태스크타입들 중 하나 또는 여러 개를 처리하는 능력을 가진다. NODE_MGT는 NODE_INFO_MGT와 NODE_OP_MGT라는 두 개의 서브 태스크로 나누어진다. 전자는 노드 등록 및 위치 정보 변경 등과 같은 관용 정보 수정을 위한 것이고, 후자는 센서 노드의 시작, 중단 및 각종 환경 변수의 변경 또는 조회 등과 같은 센서 노드에 전달되는 각종 명령을 처리하기 위한 것이다.

AGENT_MGT는 시스템 내에 있는 에이전트들의 상태 확인, 동작 변수 설정 등의 에이전트 제어에 사용된다. CLIENT_MGT는 신규 사용자의 등록 등 같은 제반 사용자 관리를 위한 것이다. REPORT는 움직임 이벤트 발생 시 사용자 시스템에게 센서 값 및 현장 이미지를 전달하거나 또는 센서 노드의 동작 예라 등을 보고하는 용도로 사용된다.

DATA_IN의 용도는 움직임 감지 센서 노드로 부터 전달되는 데이터를 처리하기 위한 것이다. DATA_PROCESSING은 추가적인 각종 데이터 처리 또는 분석을 위해 사용된다. 예를 들면, 센서 노드로부터 받은 센서 값이나 이미지에 대하여 화질 개선, 확대 및 기타 분석 업무를 위해서 사용될 수 있다.

ID	TASK_TYPE	SRC	DES	IN_PAR AM	OUT_PAR AM	TIME_STAMP	STATUS
----	-----------	-----	-----	-----------	------------	------------	--------

그림 7. 태스크양식
Fig. 7. The Task Format

서버 내의 에이전트들이 서로 협조하여 문제를 적절히 해결하기 위하여 태스크는 그림 7과 같은 양식을 갖는다. ID는 해당 태스크의 고유번호이고, TASK_TYPE은

해당 태스크의 타입을 명시한다. SRC는 해당 태스크를 생성하는 에이전트를 표시하고, DES는 해당 태스크를 처리할 에이전트를 지칭한다. DES는 해당 태스크를 처리한 에이전트가 미리 정해져 있는 경우에 사용하기 위함인데, 만약 본 필드가 명시되어 있지 않는 경우에는 블랙보드가 적절한 에이전트를 선택하게 된다.

IN_PARAM은 해당 태스크 수행을 위한 입력 파라미터를 저장하는 용도로 사용된다. 입력된 파라미터들은 '파라미터 = 값'의 형태로 이루어진다. 두 개 이상의 파라미터를 가질 경우 ';' 로 구분한다. 예를 들면 '파라미터1=10;파라미터2=10'. OUT_PARAM은 태스크 처리후 출력이 발생할 경우 그것을 저장하는 용도로 사용된다. 그 외 태스크 생성시간을 기록한 TIME_STAMP 및 태스크 처리 상태를 알리는 STATUS가 있다.

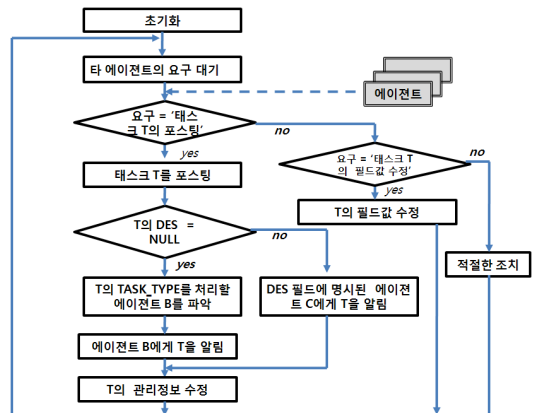


그림 8. 블랙보드 에이전트의 동작
Fig. 8. The Operation of the Blackboard Agent

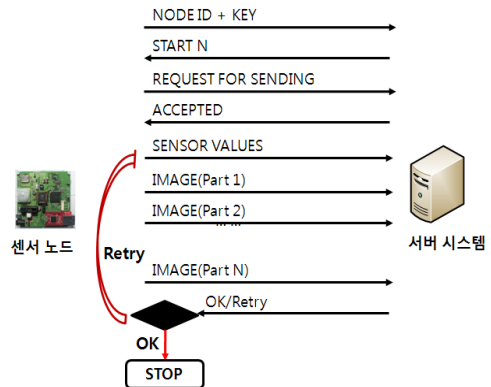


그림 9. 센서 노드와 서버 시스템간의 자료 전송(ZigBee 기반)
Fig. 9. Data Transmission Between Sensor Nodes and the Server System (ZigBee-based)

본 서버 시스템의 전반적인 동작은 그림 8에 나타난 블랙보드의 동작을 보면 쉽게 이해가 된다^[9]. 블랙보드는 기본적으로 타 에이전트들의 요구를 받아 처리하는 작업을 반복한다. 수신한 요구가 새로운 태스크 객체의 포스팅이라면, 해당 객체를 공통 버퍼에 포스팅한다. 그런 후, DES 필드 값에 맞추어 적절한 에이전트를 선택하여 그 태스크 객체의 존재를 알린다.

3. 서버 시스템과 센서 노드간의 통신

서버 시스템과 센서 노드간의 통신은 센서부 인터페이스 에이전트와 센서 노드 에이전트가 담당한다. 센서 노드가 동작을 시작하면 먼저 서버 시스템에 접속하여 자신의 ID와 암호를 전달함으로써 자신의 존재를 알린다. 서버 시스템은 등록된 센서 노드에게는 START 메시지를 전달하여 감시 시작을 명령한다.

센서 노드가 서버 시스템에게 센서 값 및 현장 이미지를 전송하는 경우는 3 가지이다. 그 중 하나는 앞에서 언급한 대로 동물 또는 물체의 움직임을 감지한 경우이고, 다른 둘은 사용자의 전송 요구에 따른 것과 주기적인 전송 모드가 설정된 경우이다. 주기적 전송 모드에서는 이미 정해진 시간 간격으로 규칙적으로 현장 이미지와 센서값을 보내게 된다.

센서 노드가 서버 시스템에 현장 이미지를 전송시에는 여러 패킷들로 나눈 후 이들을 차례로 전송한다. 따라서, 서버 시스템에서는 이미지 데이터에 대한 재조합 과정이 필요하다. 패킷의 크기는 센서 노드와 서버 시스템 간의 연결 방식에 따라 다르다. 이들이 LAN으로 연결된 경우에는 사용하는 TCP/IP 칩의 특성상 2K 바이트이고, ZigBee 네트워크를 이용하는 경우에는 ZigBee 네트워크의 최대 데이터 크기인 78 바이트이다.

Zigbee 네트워크를 이용해서 이미지를 전송하는 경우, 우리의 실험 결과에 따르면 센서 노드 여러 대가 동시에 이미지를 전송하는 경우에는 거의 전송 에러가 발생하여 상당한 수의 재전송이 필요하였다. 따라서, 그림 9와 같이 센서 노드와 서버 시스템간의 연결에서 ZigBee 네트워크를 이용한 하는 경우에는, 먼저 센서 노드가 서버 시스템으로부터 전송 요구(REQUEST FOR SENDING)를 보낸 후 전송 허락(ACCEPTED)을 받은 후에 전송을 시작한다. 서버 시스템은 한 번에 센서 노드 한 대만이 이미지 전송을 하도록 전송허락을 하고 있다.

V. 구현 및 실험

1. 구현

우리는 본 논문에서 제안한 시스템의 주요 기능과 동작원리를 중심으로 영상 감시 시스템을 구현하였다. 센서 노드 에이전트는 keil4 컴파일러를 사용하여 구현하였다. 서버 시스템은 Visual Studio2008을 이용하여 구현하였고 DBMS는 Firebird^[8]을 사용했다. 사용자 시스템은 ActionScript 3.0을 이용하여 FLASH로 구현하였다.

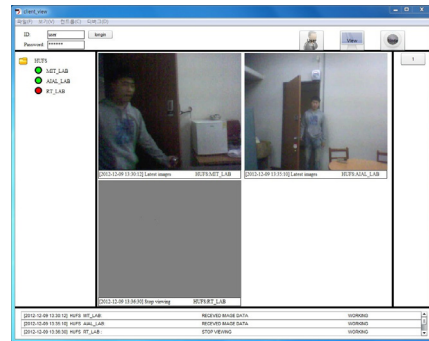


그림 10. 사용자 시스템의 화면 예

Fig. 10. A Screen Example of the User System

그림 10은 사용자 시스템의 화면 예제이다. 본안상의 이유로 사용자는 먼저 ID와 PASSWORD를 제시하여 로그인한 후에 시스템을 사용할 수 있다. 화면의 좌측에는 운영 중인 센서 노드들의 목록을 보여주고 중앙에는 가장 최근에 보내온 현장 이미지를 보여줌으로써, 사용자의 현장 모니터링 업무를 지원한다. 그림 9에서는 현장 3곳을 감시 중으로서, 마지막 현장과 같이 상대적으로 주의가 필요하지 않는 현장의 이미지는 잠정적으로 가림으로써 사용자가 나머지 현장들에 집중할 수 있도록 하였다.

2. 실험

우리는 실험을 통해서 본 논문에 기술한 시스템의 성능과 정상동작 여부를 파악해 보았다. 서버 시스템은 Intel i7 Duo CPU 3.0GHz, 메모리 4GB의 사양을 가진 컴퓨터에서 동작시켰다. 사용자 시스템을 사용자가 서버와 유사한 사양의 컴퓨터에서 웹 브라우저 상에서 실행시켰다. 실험에 사용한 현장 이미지의 해상도는 640x480이었다.

(주) 셀로코가 아직 센서 노드의 하드웨어인 SSN-1000을 양산하는 단계가 아니기에 실험은 크게 2가지 형

태로 진행되었다. 하나는 실제 센서 노드 몇 대를 현장에 설치한 후 실시한 것이고, 다른 하나는 센서 노드의 현장 이미지 전송을 흉내내는 가상 센서 노드 소프트웨어로 구현하여 사용하는 것이다.

첫 번째 형태의 실험에서는 대학 실험실과 건물주변의 외부주차장을 감시 현장으로 선택하였다. 대학 실험실을 선택한 이유는 외부인에 의한 도난 사건이 가끔 발생하기 때문이다. 따라서, 우리는 대학 실험실 3곳에 각자의 출입문으로부터 4 ~ 5m 정도 떨어진 지점에 센서 노드를 설치하여, 사람들이 실험실에 출입하는 이벤트를 감지하게 하였다. 센서 노드와 서버 시스템간의 연결 방식은 그림 2 (a)를 사용하였다. 적외선 센서를 움직임 감지용으로 사용하였고, 이를 위해 조건식의 임계치 $T=20\%$ 로 설정하였다. 우리의 측정에 의하면, 타 센서와 같이 적외선 센서의 출력값은 특성상 평상시에도 기준값을 중심으로 5~10% 만큼 변동하였다. 따라서 이들 변동을 감안해서 이보다 큰 값을 임계치로 설정하였고, 거의 모든 사람의 출입을 감지하여 사용자 시스템에게 보고하였다. 그리고 움직임 이벤트 감지 후 센서 노드가 서버 시스템에게 VGA 급의 현장 이미지를 전송하는데 걸리는 시간은 대략 1.5초 정도였다. 센서 노드 3대가 동시에 현장 이미지를 전송하도록 하였을 때도 거의 같은 시간이 걸렸다.

외부주차장에서 실시한 실험의 목적은 특정인 전용주차구역에 허가받지 않은 차량이 주차하는 지를 감시하는 목적에서이다. 각 주차 구역 뒤편에 센서 노드를 설치한 후 어떤 차량이 주차를 시도하면 이를 거리 센서가 감지하도록 설정하였다. 이때 거리 센서의 감시 조건식에서 임계치는 15%로 설정하였을 때, 거의 모든 주차 시도를 감지하여 차량의 후면을 촬영한 이미지를 서버 시스템에 전송하였다. 센서 노드 한 개가 VGA 급의 현장 이미지를 서버 시스템까지 전송하는데 걸리는 시간은 평균 50초 정도였다. 앞의 실험보다 전송기간이 긴 이유는 ZigBee 네트워크의 느린 특성으로 인한 것이었다. 여러 개의 센서 노드가 이미지를 전송을 할 때의 전체 전송 시간은 한 개의 전송시간과 노드 개수를 곱한 값과 같았다.

두 번째 형태의 실험에서는 가상 센서 노드를 이용하여 실시한 부하 테스트이다. 그 이유는 센서 노드와 서버 시스템 사이에 LAN으로 연결된 경우에는 여러개의 센서 노드가 동시에 이미지를 전송할 수 있기 때문이다. 수십개의 가상 센서 노드를 별도 컴퓨터에서 생성한 후 이

들이 동시에 가상 현장 이미지를 서버 시스템에 전송하도록 하였다. 실험 결과, 현장 이미지가 서버 시스템에 성공적으로 저장되고 사용자 시스템에 전송되는 전 과정에서 에러 없이 동작하였다. 실제 응용에서 수십대의 센서 노드가 동시에 현장 이미지를 전송할 경우가 거의 없는 점을 감안하면, 본 시스템의 유용성은 높다고 할 수 있다.

VI. 결 론

일반적인 영상감시시스템은 감시 현장을 비디오로 녹화하고 서버에 저장 하는 방식을 채택하고 있기에 특성상 다량의 데이터를 처리한다^[10]. 하지만, 출입문에서의 외부인 출입 파악 등과 같은 생활 주변에서 필요한 많은 현장 감시를 위해서는 비디오가 아닌 정지영상을 이용하는 감시시스템이면 충분할 것이다. 본 논문에서는 움직임 감지용 센서를 이용한 에이전트 기반의 정지영상 감시시스템을 제안하였다. 본 논문에서 제안한 시스템은 감지 센서값의 변화가 감시조건을 만족할 시 현장 상황을 정지영상으로 캡처 및 저장하고 원격지에 있는 관리자에게 알리게 된다. 이렇게 함으로서 감시 상황의 발생을 보다 정확하게 인식할 뿐만 아니라, 처리하는 데이터의 양을 현저히 줄일 수 있는 장점이 있다. 또한 에이전트 기반의 시스템 구조를 채택하였기에 향후 기술변화에 따라 새로이 도입되는 모듈을 추가하거나 기존 모듈을 수정은 작업을 쉽게 수행 할 수 있다. 이외에도 사용자 시스템은 웹 브라우저상에서 동작하기 때문에 사용자의 위치에 제한되지 않고 시스템에 접근할 수 있는 편리함을 제공한다.

우리 실험결과 본 시스템은 사용자 편의성과 사용자 반응 속도 면에서 사용자가 만족할 만한 성능을 가지고 있었다.

참 고 문 헌

- [1] Arvind Kandhalu, Anthony Rowe, Ragunathan (Raj) Rajkumar "DSPcam: A Camera Sensor System for Surveillance Networks", Distributed Smart Cameras, 2009, ICDCS 2009, Third ACM/IEEE International Conference, pp.1-7, 2009.

- [2] Anthony Rowe, Dhiraj Goel, Raj Rajkumar "FireFly Mosaic: A Vision-Enabled Wireless Sensor Networking System", Real-Time Systems Symposium, 2007. RTSS 2007. 28th IEEE International, pp.459 - 468, 2007.
- [3] S Hengstler, H Aghajan "A Smart Camera Mote Architecture for Distributed Intelligent Surveillance", ACM SenSys Workshop on Distributed Smart Cameras, Oct. 2006.
- [4] Ian Downes, Leili Baghaei Rad, Hamid Aghajan "Development of a Mote for Wireless Image Sensor Networks", Proc. of COGnitive systems with Interactive Sensors (COGIS), Paris, France, 2006.
- [5] G. Pekhteryev, Z. Sahinoglu, P. Orlik, G. Bhatti, "Image transmission over IEEE 802.15.4 and ZigBee networks", IEEE International Symposium on Circuits and Systems(ISCAS 2005), May 2005, vol. 4, pp. 3539 - 3542, 2005.
- [6] A. Moreno, A. Valls, D. Isern, and D. Sánchez, "Applying Agent Technology to Healthcare: The GruSMA Experience", IEEE Intelligent Systems, vol. 21, no. 6, pp.63-67, Nov./Dec. 2006.
- [7] M. Oprea, E. Dragonmir, M. Carbureanu, "On the use of collaborative intelligence in an agent-based environmental monitoring and analysis system", System Theory, Control, and Computing (ICSTCC), 2011 15th International Conference, pp.1-6, 2011.
- [8] Firebird DBMS, <http://www.firebirdsql.org>
- [9] Hyun Joo Cho, Sangchul Kim, "EasyCare: An Agent-based u-Healthcare System for Managing Patients with Heart Diseases," Journal of Korea Society of IT Services, Vol. 10, No. 1, pp.177-190, 2011.
- [10] Won-Ho, Kim, "Implementation of an Intelligent Video Surveillance System based on Digital Media Processor," Journal of the Korea Academia-Industrial cooperation Society, vol. 11, No. 3, pp.841-846, 2010.

※ 본 논문은 2011년 중소기업청 지원 사업의 결과를 기반으로, 2012년 한국외국어대학교 학술연구비 지원에 의하여 연구되었음

저자 소개

차 충 용(준회원)



- 2010년 8월 : 연변대학 전산학과 학사
 - 2010년 9월 ~ 현재 : 한국외국어대학교 컴퓨터공학과 석사과정
- <주관심분야 : 기능성게임, WSN>

김 상 철(정회원)



- 1994년 5월 : 미시간주립대학교 컴퓨터공학과 박사
 - 1983년 3월 ~ 1994년 8월 : ETRI 연구원
 - 1994년 9월 ~ 현재 : 한국외국어대학교 컴퓨터공학과 교수
- <주관심분야 : 기능성게임, WSN>