

임베디드 시스템 기반의 무인 감시 시스템

김승문^o, 이성준, 안광선
경북대학교 컴퓨터 공학과
{zeroth79, imggaibi, ahngs}@knu.ac.kr

Invader Detection System Using Embedded System

SeungWoon Kim^o seongJoon Lee KwangSun Ahn
Dept. of Computer Engineering, Kyungpook university

요 약

본 논문에서는 감시 장치를 위한 간편하고 실용적인 감시 장치 구현에 관하여 논한다. 본 장치는 실시간 영상 분석을 이용하여 움직이는 물체의 영역(변환영상)을 차영상기법, 이진화, 침식, 평창 알고리즘을 이용하여 검출한다. 이런 간단한 알고리즘을 이용하므로 제한적인 임베디드 시스템의 데이터 처리에 부담을 주지 않는다. 배경영상과 변환영상을 구분하여 저장하는데 첫 프레임을 배경영상으로 저장하고 그 다음 프레임부터는 화면의 변화가 있을 경우, 그 변환영상만을 구하여 메모리에 저장하고 클라이언트로 전송한다. 변환영상만을 저장, 전송하기 때문에 메모리 활용과 전송 효율을 높이고 패킷의 양이 적어 전송부하를 줄인다.

1. 서론

최근에 멀티미디어를 기반으로 한 감시 시스템의 중요성이 점차적으로 부각되어가고 있는 추세이며, 다양한 시스템이 출시되고 있다. 특히 산업현장에서의 자동화 감시시스템과 일반 사회에서 널리 사용되어지고 있는 무인 경비 시스템은 인건비 절약과 작업자가 현장에서 감시하기 어려운 작업등에 매우 필요한 장비로 부각되고 있다. 이런 시스템은 산업현장의 공장 자동화 시스템과 빌딩의 감시 모니터링 시스템 등에 있어서 이미 보편화되어오고 있으며, 그 외에 일반 사회생활의 여러 분야인 아파트 관리 시스템이나 은행, 백화점, 대형 할인마켓, 편의점 등에도 필수적인 시설물로 설치되어지고 있다.

그러나 현재의 생산되고 있는 대부분의 시스템은 자체적인 검출기능이 없이 단순히 동영상 또는 이미지를 저장만 가능하다. 그로인해 침입 탐지가 발생한 경우 해당 정보를 찾기 위해 저장된 데이터를 순차적으로 검토해야만 하기 때문에 비효율적이고, 침입의 발생을 관리자가 인지하기까지 많은 시간이 걸린다. 또한, 데이터의 영상을 저장하기 위해 많은 양의 추가적인 데이터 저장 공간을 필요로 한다.

본 논문에서는 이러한 문제점을 해결하기 위해 침입 탐지가 이루어진 경우 이를 관리자에게 즉각적으로 알리고 침입탐지가 발생한 시간 정보와 침입이라고 고려되는 객체의 정보만을 저장함으로써 필요 저장 공간의 크기와 패킷의 양을 줄일 수 있는 방법을 제안한다. 특히 제안된 방법은 부재중, 또는 특정 지역의 무인 감시 시에 효율적으로 사용되어질 수 있다.

본 논문은 2장에서 제안된 방법이 갖추어야할 구조를 설명하고 3장에서 제안된 구조를 구현하기 위해 필요한 하드웨어 구조를 설명한다. 4장에서 구현된 시스템의 성능을 평가하고 5장에서 결론 및 향후 연구 과제를 기술한다.

2. 무인 감시 시스템 구조

무인 감시 시스템은 인터넷 기반의 시스템으로 다음 그림 1과 같은 전체적인 구조를 갖는다.

2.1 침입탐지 서버

침입탐지 서버의 역할은 카메라를 통해 받은 영상을 저장하고 침입 탐지를 수행하는 장치로 임베디드 시스템에 구현되어져 있다. 임베디드 시스템이 갖는 작은

크기의 메모리를 효율적으로 사용하고 네트워크의 부하를 줄이기 위해 침입 발생 후보 지역만 추출하여 클라이언트에게 전달한다.

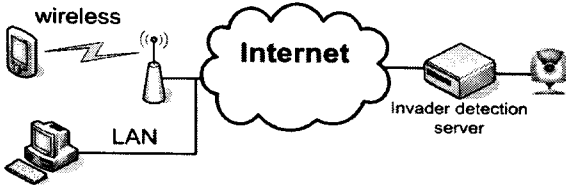


그림 1 무인감시 시스템의 구조

침입탐지 서버가 전달하는 패킷은 두 가지 형태로 나뉜다. 먼저 배경영상은 침입 탐지가 발생하지 않은 상태의 영상으로 매 5분 단위로 저장한다. 변환 영상은 침입 탐지가 발생한 경우 침입이 발생한 곳으로 여겨지는 지역의 위치 정보와 추출된 객체의 이미지 데이터이다. 그림 1은 침입탐지 서버 프로그램의 흐름도를 나타낸 것이다.

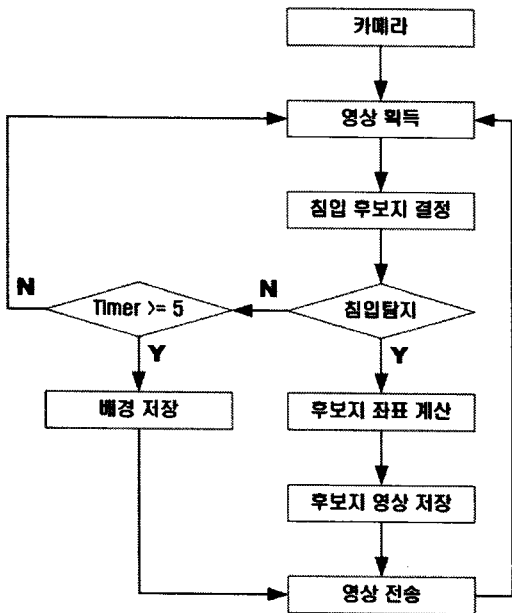


그림 2 침입탐지 서버 흐름도

시스템이 작동되면 첫 번째 프레임의 이미지를 획득하여 배경영상으로 파일 시스템에 저장한다. 버퍼에 저장되어 있는 배경영상과 다시 획득되어 들어오는 영상을 비교하여 영상의 변환이 있다면 변환된 부분의 좌표를 계산하여 변환영상을 저장하고 접속되어진 클라이언트에게 변환영상의 좌표와 이미지 데이터를 UDP/IP로 전송한다. 만일 침입 탐지 영상이 아니고, 5분 동안 침입탐지가 이루어 지지 않았다면 새로운 배경영상으로 저장한다. 새로운 배경영상의 필요성은 시간의 흐름에 따라 빛의 양이 달라지기 때문이다.

2.2 침입 후보지 결정

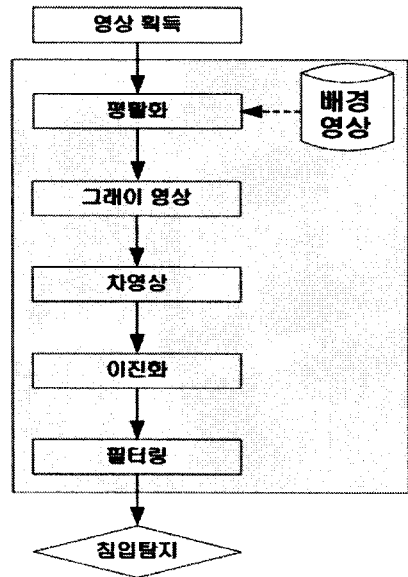


그림 3 침입 후보지 결정을 위한 흐름도

침입 후보지의 결정을 위해 파일시스템에 저장되어 있는 배경 영상과 새로운 입력된 영상의 차를 이용한다. 이 경우 가장 문제시 되는 것은 노이즈이다. 본 논문에서는 노이즈의 제거를 위해 입력된 영상의 평활화 과정과 그레이 영상 변환을 거쳐 비교 영상을 만들고 새롭게 생성된 영상과의 차를 이용하여 영상의 변화를 감지한다. 그림 3은 침입 후보지의 결정을 위한 흐름도를 나타낸다. 그림 4는 침입 후보지 결정을 위한 각 단계별 중간 결과를 표현하고 있다.

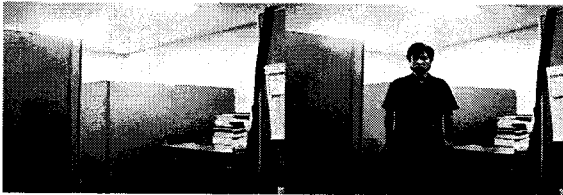


그림 4 단계별 중간 결과

차영상에서 넘겨진 값은 처리가 용이한 이진영상으로 변화시키고, salt-and-pepper 노이즈로 인해 침입으로 간주되는 영역을 제거하기 위해 필터링 작업을 수행한다.

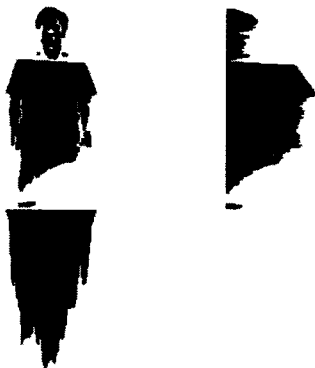


그림 5 차영상 히스토그램

작업이 완료되어진 후 각 축에 대해 나타나는 마킹 값을 카운트 한다. 각 축의 검정색 픽셀의 카운트가 0

이 아닌 값의 폭이 10 이하이면 무시하고, 그렇지 않은 경우가 하나 이상 존재 하면 침입이 있는 경우로 간주한다.

이후 폭이 10 초과인 지역의 좌표를 넘겨줌으로써 예상 후보 지역의 좌표를 산출한다. 후보지역 영상 저장에서는 산출된 좌표를 이용하여 원 영상에서 필요한 부분만 추출하고, 추출된 부분의 영상을 클라이언트로 전달하기 위해 영상 전송 함수를 호출하게 된다.

2.3 클라이언트

클라이언트는 관리자가 원격지에서 침입탐지 서버에 접속하기 위한 도구로 분리되어 전달된 변환 영상을 이전에 받은 배경영상과 합성하여 나타내게 된다. 예를 들어 침입탐지 서버가 영상을 전달하면 클라이언트는 전달된 영상이 배경 영상인지 아니면 침입탐지 영상인지 결정한다. 배경 영상인 경우 클라이언트는 파일시스템에 이를 저장하고 새로운 영상이 도달할 때 까지 대기한다. 이후 침입 탐지 영상이 전달되면 저장된 배경 영상과 합성하여 화면에 표시함으로써 기존의 영상과 유사한 품질을 유지할 수 있다. 그림 6는 클라이언트의 흐름도를 나타낸 것이다.

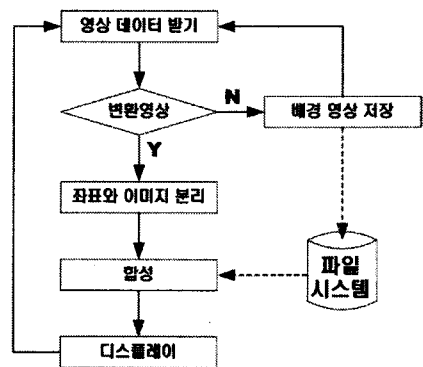


그림 6 클라이언트 흐름도

그림 7은 클라이언트 프로그램을 실행한 화면이다. 그림의 왼쪽 부분은 침입 탐지가 이루어진 경우 이전에 받은 배경 영상과 합성하여 화면에 나타낸 것을 캡처 받은 것이다. 우측의 리스트는 침입탐지가 일어난 시간을 리스트 형태로 작성하여, 언제든지 그 침입 탐지 영상을 재검토 할 수 있도록 하였다. 이러한 이미지 데이터는 서버로부터 다시 받을 수 있다.

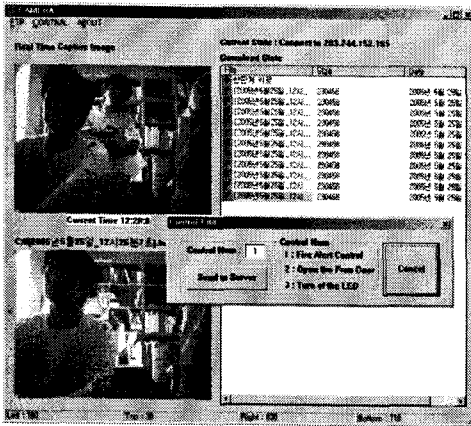


그림 7. 클라이언트 실행 화면

4. 성능 평가

제안된 시스템의 성능을 평가하기 위해 본 논문에서는 Inter Xscale 기반의 CPU를 탑재한 시스템에 임베디드 리눅스 v2.4를 포팅하여 실험을 하였다.

표 1에서 보여주는 바와 같이 침입 탐지가 없는 경우 데이터의 저장과 전달이 발생하지 않기 때문에 저장 공간 활용능력이 기존 시스템에 비해 탁월한 성능을 가지고 있음을 알 수 있다. 또한 침입 탐지가 발생한 경우 기존의 방식에서 보다 약 103배 정도의 작은 저장 공간과 패킷의 수를 가지고 있으며, 복원, 재생은 원본과 동일한 화질을 가지고 있다.

표 1. 영상 변환이 없을 경우

	기존 시스템	제안 시스템
영상 개수	9프레임×10초	0프레임×10초
전체 데이터 양 (byte)	20736000	0
패킷 (1패킷=1024byte)	20250개	0개

표 2. 영상 변환이 있을 경우

	기존 시스템	제안 시스템
영상 개수	9프레임×10초	9프레임×10초
전체 데이터 양 (byte)	20736000	200256
패킷 (1패킷=1024byte)	20250개	196개

5. 결론 및 향후 과제

본 논문에서는 효율적인 무인 감시 시스템을 구현하였다. 기존 시스템은 전체의 영상을 압축하여 저장한다. 하지만 제안된 시스템은 침입 감지 영상과 배경 영상을 분리하여 저장함으로써 저장 공간을 효율적으로 사용하고, 더불어 전송 패킷의 수를 줄임으로써 최적화된 무인 감시 시스템을 구현하였다. 따라서 본 논문에서는 감시 영상에서 동시에 둘 이상의 물체가 움직이는 경우는 없다고 가정하여 물체 검출을 시도하였으므로, 향후 다수의 물체를 검출, 추적할 수 있는 방법에 대한 연구가 필요하다. 그리고 변환 영상에 압축 알고리즘을 적용하는 연구가 필요하다.

참고 문헌

- [1] 박재호, 임베디드 리눅스, 한빛미디어, 2002.
- [2] 나중화 외5, "Embedded System Programming - 이론 및 실습 -, 한국전자통신연구원, 2003.
- [3] Alessandro Rubini (역자 : 김인성 외), "리눅스 디바이스 드라이버 (Linux Device Drivers)", 한빛미디어, 2000.
- [4] 전준철, "디지털 영상처리", 정익사, 2005.
- [5] Majid Rabbani, Paul W. Jones: Digital Image Compression Techniques, 1990.
- [6] 강동중 외, "Visual C++을 이용한 디지털 영상처리", 사이텍미디어, 2003.
- [7] <http://vision.incheon.ac.kr/>