

클라우드 기반의 모바일 지능형 관제시스템에서의 움직임 감지 알고리즘에 관한 연구

A Study on the Moving Detection Algorithm for Mobile Intelligent Management System Based on the Cloud

박 성 기*, 김 옥 환**

Sung-Ki Park*, Ok-Hwan Kim**

Abstract

This study suggested the mobile intelligent management system based on the cloud service. The mobile intelligent management system are composed of cloud server, middleware and sensor networks. Each modules are controlled on mobile environment and observed operating status of each apparatus for environment. In this pater, the image-based moving detection algorithm applied in order to detect an intruder and average 12.3% are measured in moving detection experiments. it was confirmed the validity of the security device.

요 약

본 논문에서는 클라우드 기반의 모바일 지능형 관제시스템 개발을 제안하였다. 모바일 지능형 관제시스템은 클라우드 서버, 미들웨어, 센서로 구성되어 있다. 각 모듈들은 모바일 환경에서 제어되고 주변 환경에 대한 각 기기의 동작 상태를 모니터링 할 수 있다. 본 논문에서는 침입자를 감지하기 위해 영상 기반 움직임 감지 알고리즘을 적용하였고, 움직임 감지 실험에서 움직임 검출율이 평균 12.3% 높게 측정되어 보안장치로서의 타당성을 확인하였다.

Key words : Wireless Remote Control, Intelligent Management system, Mobile Web, Image Processing, Moving Detection

1. 서론

* Department of Digital Management Information, Nambu University.

★ Corresponding author

bbkang6615@naver.com +82-62-970-0225

※ Acknowledgment

Manuscript received Feb.5, 2015; revised Mar.8, 2015;
accepted Mar.9.2015

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

모바일 네트워크 환경이 빠른 속도로 일상생활에 침투하면서 스마트홈 및 가정용 보안 솔루션에 대한 수요 증가에 의해서 세계 스마트홈 시장은 빠르게 성장하고 있다. 학계, 연구소 등에서는 지능형 솔루션을 위한 카메라 제어기술, 영역탐지, 물체추적, 생체인식, 원거리 물체인식 등에 관한 연구를 진행하고 있다 [1][2]. 보안 분야는 IT 기술을 이용한 무인경비서비스와 노동력을 이용한 유인경비서비스 등을 중심으로 발전되어 왔으나 최근 들어 이동단말을 통해 고객이 요구하는 위치에 설치된 CCTV 영상을 전송해주는 관제시스템에 대한 요구가 증가하고 있다[3].

대부분의 관제시스템은 유선으로 구축되기 때문에 설치의 어려움, 고비용의 유지·관리비용(설치환경, 저

장매체 등)이 소요되는 문제점이 있다. 특히, 사건 발생시 호출(문자, 알람)에 의해 현장에 출동하고 사고를 처리하기 때문에 정확하고 세부적인 대응이 어려운 문제점을 가지고 있다.

본 논문에서는 실시간 영상/음성 전송을 통해 보안을 강화하고, 쌍방향 무선통신을 통해 온도, 전기기기 등을 제어하여 에너지를 절감할 수 있도록 클라우드 기반의 모바일 관제시스템을 제작하였으며, 영상 알고리즘을 통해 움직임을 감지하도록 하여 보안장치로의 타당성을 검토하였다.

II. 본론

1. 시스템 개요

본 논문에서는 그림 2.1과 같이 IP 카메라, 팬틸트 구동부, 인체 감지 센서, RF 통신 모듈, MCU로 지능형 관제 시스템을 구성하였으며 그림 2.2와 같이 시제품을 제작하였다. 움직임을 감지 알고리즘을 적용하여 보안 기능을 강화하고, 사건 발생시 영상 전송을 이용한 신고 시스템을 도입하여 신속히 사건에 대처할 수 있도록 하였다.

메인제어기는 카메라 영상과 감지 센서를 통해 움직임을 분석하고, 분석된 데이터는 클라우드 서버에 전송되어 클라우드 DB에 저장된다.

사용자는 단말기(PC, 스마트 폰 등)를 통해 영상, 음성, 동작상태 등을 모니터링하고 카메라를 제어할 수 있다.

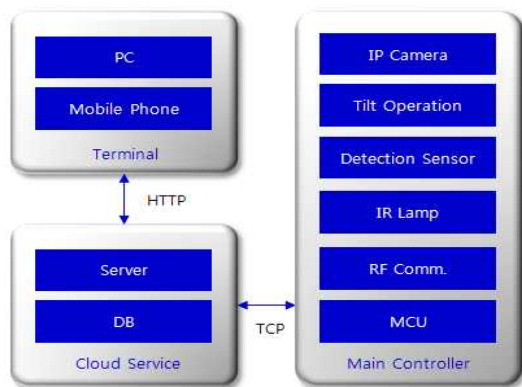


Fig. 2.1 Intelligent management system diagram for moving detection

그림 2.1 움직임 감지를 위한 지능형 관제시스템 구성도

IP 카메라를 통해서 영상, 음성, 사진을 저장할 수 있고, LED 적외선 램프를 통해 야간 촬영이 가능하

며, 좌우 160도 팬틸트 기능을 통해 원격으로 카메라를 제어 할 수 있고 아래와 같이 구성되어 있다.

- 영상 해상도 : VGA급 (640X480)
- 사진 해상도 : 410,000 화소
- 프로토콜 : HTTP, TCP/IP



Fig. 2.2 Prototype main controller

그림 2.2 프로토타입 메인제어기

팬틸트 구동부는 스테핑 모터를 이용하여 좌우 160°로 몸체를 회전시켜 영상의 모니터링 범위를 확대시킬 수 있다.

- 회전각 계산 : 엔코더를 통해 모터의 회전수에 따른 회전각을 산출
- 모터 회전수 제어 : PWM 제어

메인 처리 장치는 각 구성요소들을 제어하기 때문에 다음과 같은 주요 성능을 갖도록 하였다.

- CPU : 32bit Embedded CPU
- OS : Embedded Linux
- Flash Memory : 128MByte
- SDRAM : 128MByte
- 인터페이스 : RS-232C, RS-485, SPI, UART/USRT

관제시스템으로부터 수신되는 각종 데이터(영상 및 센서 계측 값)을 저장하기 위해서 오라클을 기반으로 클라우드 서버와 DB를 구축하였으며, 어플 개발언어는 java, jsp / html5, CSS3, jquery-mobile-1.3.2 / phonegap이고 서버사이드는 java, jsp로 제작하였다.

그림 2.3은 스마트폰 어플 화면을 나타내고 있다.

소프트웨어는 자바스크립트, HTML5, CSS3를 이용하여 모바일 기기를 위한 응용 프로그램들을 제작하였다. 응용 프로그램 결과물들은 하이브리드 형태로써 모든 레이아웃 렌더링이 플랫폼의 네이티브 UI 프레임워크가 아닌 웹 뷰를 통해 수행되며, 단지 웹앱일 뿐만 아니라 배포를 위한 패키지를 통해 네이티브 기기 API에 접근할 수 있도록 하여 안드로이드나

아이폰에 모두 호환되도록 하였다.
JSP 프로그램을 이용해 각종 프로토콜 처리 및 화

드에서는 html5, CSS3, jquery-mobile-1.3.2를 이용해 화면을 제작하였다.

2. 영상 기반 움직임 감지 알고리즘

기존 움직임 감지 알고리즘들은 각 픽셀의 색상정보를 이용하여 배경을 구축하기 때문에 복잡한 배경을 가진 환경에서 수행할 경우 정확도가 확연히 낮아지는 문제점을 가지고 있다.

본 논문에서는 그림 2.4와 같이 영상 기반 움직임 감지 알고리즘을 이용해 입력 영상에서 움직임이 발생하면, 배경제거(Background Subtraction) 알고리즘과 전경추출 알고리즘을 이용하여 움직이는 대상을 감지하도록 하였다.

제안된 방법은 먼저 배경을 구축하고 전경추출 알고리즘을 이용하여 고정되어 있는 배경으로부터 움직임이 있는 영역(전경)을 추출하여 추출된 영역이 미리 정한 임계값(τ)보다 클 때 알람신호를 발생하고 영상과 문자를 전송하여 침입 상황을 알린다.

배경 구축을 위한 기존의 방법(가우시안 분포나 비모수적 접근법을 이용한 방법)은 각 픽셀의 색상정보를 이용해서 배경을 구축하기 때문에 복잡한 배경을 가진 환경에서 수행할 경우 정확도가 확연히 낮아지는 문제점을 가지고 있다[4][5].

본 논문에서는 픽셀의 색상정보가 아닌 블록 내의 픽셀들로부터 텍스처 정보를 계산하는 Local Binary Pattern(LBP) 알고리즘을 이용하여 배경모델을 수행하도록 하였다.

픽셀단위에서 배경의 변화 정도에 따라 히스토그램



Fig. 2.3 App. windows of mobile phone
그림 2.3 스마트폰 어플 화면 구성

면에서 요구하는 정보를 처리하고, 폰이나 PC 사이

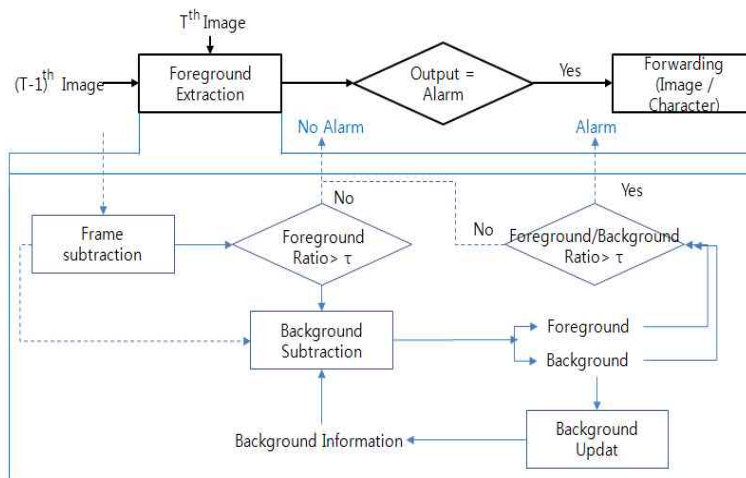


Fig. 2.4 Moving detection algorithm based on the image

그림 2.4 영상 기반 움직임 감지 알고리즘

의 수를 자동으로 조절하기 때문에 히스토그램의 수가 고정되면 모델링할 수 있는 변화 정도가 고정되어 전경추출에 성능이 저하되는 단점이 있다.

그러나 히스토그램의 수를 자동으로 조절하면 배경의 변화를 모두 포함할 수 있기 때문에 움직임을 검출하기 위해서는 효과적이다.

그림 2.5는 배경모델링에 적용된 LBP 알고리즘의 흐름을 나타낸 것으로써, NxN 윈도우 내의 모든 픽셀에 대해 수행하면 텍스처 특징을 표현하는 256개의 빈(bin)으로 구성된 히스토그램이 된다.

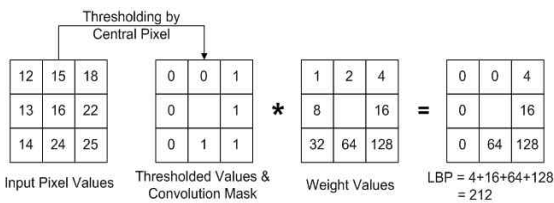


Fig. 2.5 Flowchart of LBP algorithm
그림 2.5 LBP 알고리즘의 흐름도

```
void COPENCY_TEST_VID1G::OnBnClickedButtonModelBackground( )
{
    ///open video stream
    cv::VideoCapture cam;
    cam.open(0);
    if (!cam.isOpened()){
        AfxMessageBox(L"Failed opening camera.");
        return;
    }
    int dimHist = 256, countForBM;
    CvHistogram * m_LBPFeature;
    m_LBPFeature = cvCreateHist(1, &dimHist, CV_HIST_ARRAY);
    cv::namedWindow("BackgroundModeling");
    while (1) {
        cv::Mat im, gray;
        cam >> im;
        cv::cvtColor(im, gray, CV_BGR2GRAY);

        countForBM = 0;
        for (int i = WINDOW_SIZE / 2; i <= WINDOW_SIZE; i++){
            for (int j = WINDOW_SIZE / 2; j <= WINDOW_SIZE / 2; j++){
                cvClearHist(m_LBPFeature);
                LBPOperator(&gray, m_LBPFeature, j, i, WINDOW_SIZE);
                cvNormalizeHist(m_LBPFeature, 1.0);
                UpdateBackgroundModel(m_LBPFeature, countForBM, dimHist);
                countForBM++;
            }
        }

        int c = cv::waitKey(10);
        if (c == 'q') break;
        cv::imshow("BackgroundModeling", im);
    }
    countForBM = 0;
    for (int i = WINDOW_SIZE / 2; i <= WINDOW_SIZE; i++){
        for (int j = WINDOW_SIZE / 2; j <= WINDOW_SIZE / 2; j++){
            ClearNoBackgroundEntries(countForBM);
            countForBM++;
        }
    }
    cv::destroyAllWindows();
    cam.release();
}
```

Fig. 2.6 Background modeling code
그림 2.6 배경 모델링 코드

새로운 배경영상이 입력되면 각 윈도우로부터 LBP 히스토그램 \vec{h}_i 를 계산하고, 현재까지 구축된 배경 히스토그램 집합에 유사한 히스토그램이 있는지를 비교한다.

구축된 히스토그램과 모두 비교하여 유사도가 임계값 T_p 보다 낮으면(일치하는 히스토그램을 찾지 못하면), 입력 히스토그램 \vec{h}_i 는 배경 모델에 추가되고, 일치하는 히스토그램을 찾으면 배경 히스토그램을 업데이트하도록 그림 2.6과 같이 배경 모델링 프로그램을 코딩하였다.

픽셀이 아닌 텍스처 단위 생성된 배경모델로부터 전경을 추출할 때는 외곽선을 제대로 추출하지 못하는 문제점이 있다.

본 논문에서는 정확한 외곽선을 추출하기 위해 전경추출을 텍스처 정보와 이웃한 픽셀정보를 접목한 이진 레이블링(labeling)을 고려하였다.

레이블링을 최소화하기 위하여 그래프 컷 알고리즘을 이용하였고, 오감지를 최소화하기 위하여 차영상 알고리즘을 추가적으로 적용하였다[6][7].

차영상은 이전프레임의 영상((T-1)th 영상)과 현재

```
void COPENCY_TEST_VID1G::OnBnClickedFrameDiff( )
{
    ///open video stream
    cv::VideoCapture cam;
    cam.open(0);
    if (!cam.isOpened()){
        AfxMessageBox(L"Failed opening camera.");
        return;
    }
    int nIsFirst = 1;
    cv::Mat prev;
    cv::namedWindow("Frame Differencing");
    while (1) {
        cv::Mat im, gray;
        cam >> im;
        cv::cvtColor(im, gray, CV_BGR2GRAY);

        if (nIsFirst == 1) {
            nIsFirst = 0;
            prev = im;
        }
        else {
            cv::absdiff(im, prev, im);
            cv::threshold(im, im, 15, 255, CV_THRESH_BINARY);
        }

        int c = cv::waitKey(10);
        if (c == 'q') break;
        cv::imshow("BackgroundModeling", im);
    }
    cv::destroyAllWindows();
    cam.release();
}
```

Fig. 2.7 Differential image algorithm code
그림 2.7 차영상 알고리즘 코드

프레임의 영상(T^{th} 영상) 차이를 의미한다.

현재 프레임의 영상에서 갑작스러운 변화가 감지될 경우 변화 정도가 픽셀단위로 표시되고, 픽셀단위(x, y)에서 차이 값을 계산하여 움직임을 감지하게 된다.

그림 2.7은 차영상 알고리즘 코드로써 현재 프레임과 이전 프레임의 픽셀 값을 비교한 후 0 ~ 255를 갖는 픽셀 값의 절대 값 차이가 임계값인 15 이상이면 움직임이 있는 것으로 판별된다.

3. 실험 결과

본 논문에서는 움직임 감지를 평가하기 위하여 그림 2.8과 같이 비, 흔들리는 나뭇가지, 빠르게 움직이는 차를 포함한 테스트 데이터에 대하여 기존의 GMM기반의 방법과 제안된 방법으로 영상을 추출하였다.

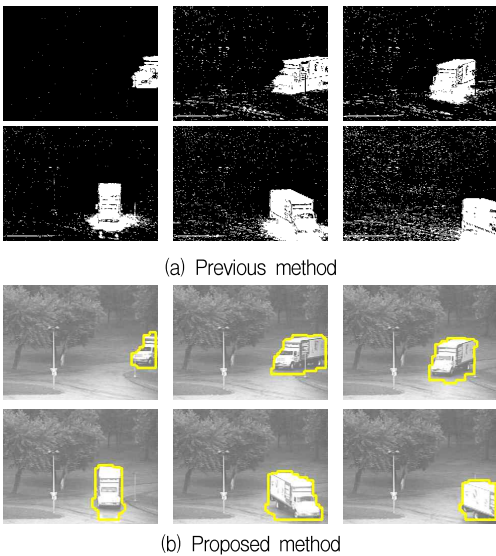


Fig 2.8 Result image data for test data
그림 2.8 테스트 데이터에 대한 결과 영상

기존에 많이 사용되는 GMM기반의 방법은 비와 흔들리는 나뭇가지 때문에 많은 수의 배경 픽셀이 전경으로 추출되었으며, 배경의 많은 부분이 움직임으로 추출되었다.

본 논문에서 제안된 방법은 일정한 간격의 움직임을 가진 비 또는 나뭇가지는 움직임으로 감지하지 않기 때문에 기존 방법보다 상대적으로 정확하게 움직임을 감지할 수 있다.

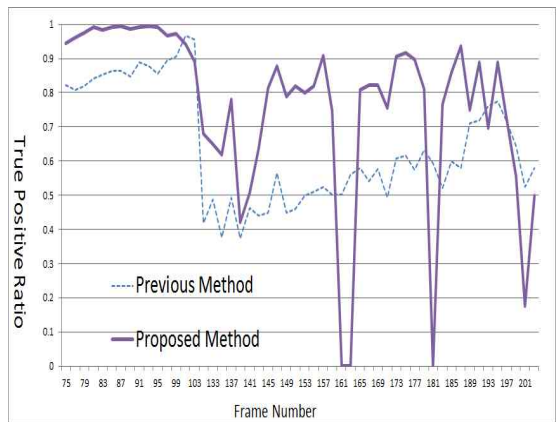
비, 흔들리는 나뭇가지, 빠르게 움직이는 차를 포함한 배경에서 움직임을 평가하기 위하여 True Positive Ratio(TPR)와 False Positive Ratio(FPR) 관

점에서 그림 2.9와 같이 측정하였다.

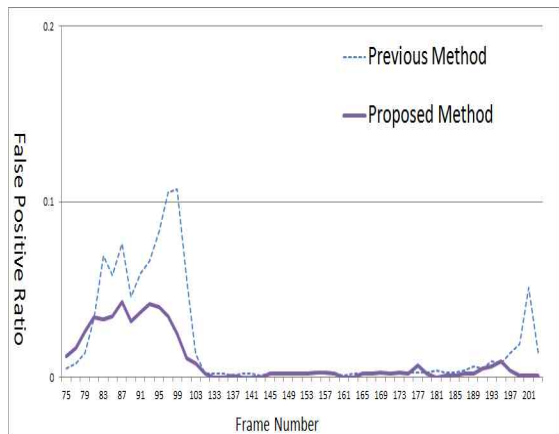
$$TPR = \frac{\text{true positive}}{\text{number of foreground pixels in ground truth}}$$

$$FPR = \frac{\text{false positive}}{\text{number of background pixels in ground truth}}$$

여기에서, True Positive는 정확하게 추출된 전경(움직임) 픽셀의 수, False Positive는 전경인데 배경으로 추출된 픽셀의 수, Ground Truth는 영상과 같이 제공되는 전경 픽셀을 의미한다.



(a) True Positive Ratio(TPR)



(b) False Positive Ratio(FPR)

Fig 2.9 Moving detection for test data
그림 2.9 테스트 데이터에 대한 움직임 추출

기존 방법과 비교해서 높은 TPR과 낮은 FPR을 나타냈으며, TPR과 FPR의 가중치 조화평균인 F-measure를 이용하여 표 2.1과 같이 정량적으로 평가하였다.

$$F\text{-measure} = \frac{(TPR \times (1 - FPR))}{(\zeta \times TPR + (1 - \zeta) \times (1 - FPR))}$$

여기에서, ζ 는 TPR과 FPR의 가중치를 나타내며, 0.5로 계산하였다.

제안된 방법과 기존 방법의 움직임 검출율에 대한 실험을 3회 진행한 결과, 제안된 방법의 움직임 검출율이 평균 12.3% 높게 측정되었다.

Table 2.1 Average TPR and FPR, F-measure

표 2.1 평균 TPR과 FPR 그리고 F-measure

		1st	2nd	3rd
Previous Method	TPR	0.973	0.991	0.644
	FPR	0.177	0.535	0.019
	F-measure	0.892	0.633	0.801
Proposed Method	TPR	0.877	0.942	0.768
	FPR	0.041	0.114	0.009
	F-measure	0.916	0.914	0.865

III. 결론

본 논문에서는 클라우드 서버, 미들웨어, 센서로 구성된 지능형관제시스템을 구현하였다. 각 모듈들은 모바일 환경에서 제어되고 주변 환경에 대한 각 기기의 동작 상태를 모니터링 할 수 있도록 하였다.

보안 기능을 강화하기 위하여 영상 기반 움직임 감지 알고리즘을 적용하였으며, TPR과 FPR의 가중치 조화평균인 F-measure을 이용해 기존 방법과 제안된 방법에서의 움직임 검출율을 측정하였다. 기존 방법에 비하여 제안된 방법의 움직임 검출율이 평균 12.3% 높게 측정되었으며, 이를 통하여 영상 보안 장치로서의 타당성을 확인하였다.

References

[1] Tae-Woong Lee, Cheol-Su Son, Won-Jung Kim "The Implement of Intelligent Home Network System on Smart Phone", The Journal of the Korea Institute of Electronic Communication Sciences, Vol. 6, no. 4, pp.505-509, 2011.
[2] K.H. Park, J.J. Choi, W.S. Yang. "Present State and Prospect of Sensor Technologies for Smart Building Present State and Prospect of Sensor

Technologies for Smart Building", Electronics and Telecommunications Trends, Vol. 24, no. 6, ETRI, 2009

[3] W. J. Kim, "CCTV market flow and estimation," EIC, Jul. 2011.
[4] M. Pietiäkäinen, A. Hadid, G. Zhao, and T. Ahonen, Computer Vision using Local Binary Patterns, Spinger, 2011.
[5] C. Y. Jung and J. W. Han, "Intelligent image analysis event detection technology trend," Electron. Telecommun. Trends, vol. 27, no. 4, pp. 114-122, Aug. 2012.
[6] Y. Boykov, O. Veksler, and R. Zabih, "Fast Approximate Energy Minimization via Graph Cuts," IEEE Trans. Pattern Anal. Mach. Intel, 2001.
[7] Y. Boykov, O. Veksler, and R. Zabih, "Fast Approximate Energy Minimization via Graph Cuts," IEEE Trans. Pattern Anal. Mach. Intel, 2001.

BIOGRAPHY

Park Sung-Ki (Member)



1988 : BS degree in Business Administration, Chonnam National University.
2007 : MS degree in Venture Start-Up Management, Nambu University.

Kim Ok-Hwan (Member)



1981 : BS degree in Communication Engineering, Mokpo National Maritime University.
2002 : MS degree in Electronic Engineering, Chonnam National University.
2013 : PhD degree in Electronic Communication Engineering, Mokpo National Maritime University.
2012~2015 : Professor, Nambu university.