Journal of The Korea Society of Computer and Information
Vol. 22 No. 6, pp. 131-140, June 2017
www.ksci.re.kr
https://doi.org/10.9708/jksci.2017.22.06.131

Implementation of Effective Automatic Foreground Motion Detection Using Color Information

Hyung-Hoon Kim*, Jeong-Ran Cho**

Abstract

As video equipments such as CCTV are used for various purposes in fields of society, digital video data processing technology such as automatic motion detection is essential. In this paper, we proposed and implemented a more stable and accurate motion detection system based on background subtraction technique. We could improve the accuracy and stability of motion detection over existing methods by efficiently processing color information of digital image data. We divided the procedure of color information processing into each components of color information : brightness component, color component of color information and merge them. We can process each component's characteristics with maximum consideration. Our color information processing provides more efficient color information in motion detection than the existing methods. We improved the success rate of motion detection by our background update process that analyzed the characteristics of the moving background in the natural environment and reflected it to the background image.

▸Keyword: Motion Detection, Digital Video Data, Background Subtraction, CCTV

I. Introduction

사회 전반에 안전 강화 및 범죄 예방의 필요성이 증가됨에 따라 CCTV 등의 영상장비를 이용한 지능형 영상보안 시스템의 구축이 활발하게 진행되고 있다. 보안 감시 카메라의 여러 가지 활용 형태 가운데 대표적인 형태가 대량의 촬영 영상물로부터 자동적으로 침입 상황을 탐지하여 알려주는 기능일 것이다. 카메라의 저장 용량 및 통신 환경 등의 이유로 초창기에 주로 흑백 영상이 사용되었으나 카메라 및 통신 환경이 급속히 발전됨에 따라 점차 칼라 영상을 기본으로 사용하고 있는 추세이다. 흑백영상에 비하여 칼라영상은 보다 정확하게 감시 대상에 대한 영상정보를 나타내고 있다.

그러나 칼라영상을 이용한 움직임 탐지의 경우 보다 정확한 정보를 제공하는 장점도 있지만 흑백영상에 비하여 큰 데이타 용량으로 인하여 많은 처리 시간과 그 만큼의 고가의 처리 장비를 요구하는 문제, 영상에 대한 상세한 정보로 인하여 움직임 탐지에 있어서 오히려 오탐지를 발생하는 등 여러 가지 문제점과 어려움도 있다. 움직임 탐지의 경우 칼라 영상이 담고 있는 정밀한 정보는 오히려 움직임 탐지의 대상이 아닌 움직이는 배경이나, 조명의 변화 등을 움직임으로 잘못 탐지하게 되는 경우가 많았다. 다른 연구 논문에서도 이와 같은 상황을 고려하여 칼라 영상의 모든 정보를 사용하지 않고, RGB 24비트 칼라 영상의 경우 각 R, G, B 채널의 하위 4비트만을 사용한 움직임 탐지를 하는 방법을 사용하였다.

그러나 칼라영상 정보로부터 하위 4비트, 총 12비트의 특정 부분의 정보를 일괄적으로 제외시키고 사용하는 방법은 여전히 오탐지의 상황을 발생시키거나, 탐지 대상의 움직임을 탐지하지 못하는 문제점을 나타내었음을 본 논문의 실험 과정에서 확인할 수 있었다. 따라서 CCTV로부터 촬영된 동영상의 칼라영상 정보를 보다 효율적으로 처리하는 새로운 방법을 제안하고 이에 대한 시스템을 구현하였다.

본 논문에서 자동 전경 움직임 탐지 방법으로 구현한 시스템

• First Author: Hyung-Hoon Kim, Corresponding Author: Jeong-Ran Cho
*Hyung-Hoon Kim (hhkim@kwu.ac.kr), Dept. of Biomedical Systems, Kwangju Womens University
**Jeong-Ran Cho (jrcho@kwu.ac.kr), Dept. of Biomedical Systems, Kwangju Womens University
• Received: 2017. 03. 13, Revised: 2017. 04. 08, Accepted: 2017. 05. 28.

은 실시간 환경에서도 사용될 수 있도록 하기 위하여 비교적 연산이 간단하고 이를 처리함에 있어 자원 사용량은 적은 배경 차영상 방법을 사용하였다. 이 배경 차영상 방법은 간단하면서도 효과적인 자동 전경 움직임 탐지 방법으로 많이 사용된다. 배경 차영상 방법은 배경영상과 현재 프레임 영상 사이의 차이를 계산하고 임계값 이상의 화소 위치를 전경의 움직임이 있는 화소로 판단하는 방법이다.

본 논문에서는 HSI(HSI: Hue Saturation Intensity) 색상모델에 대한 배경 차영상 방법을 새롭게 제안하였다. 촬영된 동영상 정보로부터 프레임별로 처리하는 과정을 통하여 움직이는 전경 물체를 탐지하는 방식이다. 이때 배경이미지와 현재이미지 사이의 차영상을 활용한다.

움직이는 전경 물체의 판단을 위하여 명도 차영상(diffVal)에 대해 명도 값에 적정한 임계값(ThVal)을 적용하였고, 색상 차영상(diffHue)에 대해 색상 값에 적정한 임계값(ThHue)을 적용하여 각 임계값 이상의 영역을 얻고, 각 명도와 색상에 대한 영역을 결합한 결과로 움직이는 전경 물체의 움직임의 여부를 판단하였다. 이와 같이 명도와 색상에 대해 서로 다른 임계값을 사용함으로써 기존의 방법에서처럼 하나의 임계값으로 처리하는 것 보다 안정적이고 올바른 움직임 판단의 결과를 얻을 수 있었다.

본 논문은 자연 환경에 보안 감시 목적으로 설치되어 있는 IP카메라로 촬영되는 칼라 디지털 영상으로부터 자동적으로 움직이는 전경을 탐지하고 움직이는 전경 물체를 추출하는 알고리즘을 제안하고 시스템으로 구현하였다. 특히 칼라영상정보를 효율적으로 처리하도록 하는 새로운 방법을 고안하여 구현함으로서 기존의 칼라영상처리 방법 보다 오탐지율을 줄이며, 보다 정확하게 탐지할 수 있는 향상된 성능을 얻을 수 있었다.

II. Related works

2.1 Motion Detection System

본 논문에서 제안한 자동 전경 움직임 탐지 시스템은 그림 1의 일반적인 영상탐지시스템을 기본으로 하고 있다.

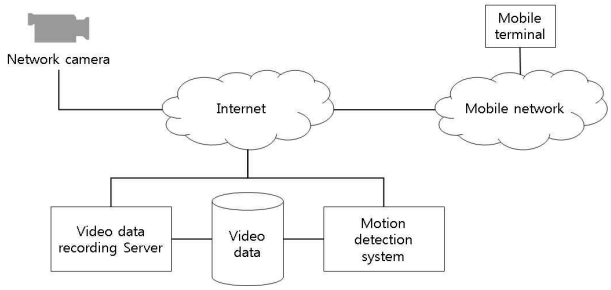


Fig. 1. General Overview of Motion Detection System

네트워크 카메라에서 촬영된 디지털 영상데이터는 인터넷으로 연결된 영상데이터 녹화 서버를 통하여 지정된 대용량 저장매체에 저장되며, 최근의 사용 환경에 적합하도록 모바일 단말기에 적정한 어플리케이션을 설치하여 이동 중에도 실시간으로 촬영되는 디지털영상을 조회할 수 있도록 운영되고 있다. 이때 움직임 탐지 시스템의 작동 형태는 이의 처리 기능의 위치에 따라 두 가지 형태가 존재한다. 첫 번째는 네트워크 카메라에 탑재 되어 영상데이터 녹화서버 이전 단계에서 움직임 탐지를 실시간으로 처리하여 움직임 탐지 결과를 영상데이터 녹화서버에 전달하는 방법이다. 두 번째는 그림 1과 같이 영상데이터 녹화 서버에 의해 저장된 영상데이터를 실시간으로 처리하여 움직임 탐지 결과를 녹화서버에 전달하는 방법이다.

움직임탐지시스템은 실시간으로 전달되는 디지털 영상데이터로부터 움직임이 있는 전경을 탐지하고 그 결과를 다양한 형태로 확인할 수 있도록 운영되고 있다. 영상탐지시스템에서 가장 핵심 요소는 움직임 탐지시스템으로써 이를 다루는 많은 연구가 진행되고 있다.

2.2 Previous Subtraction in RGB Color

차영상 움직임 탐지 기법은 이전 영상과 현재 영상의 차이가 주어진 임계값 이상인 경우를 움직임으로 탐지하는 방법이다. 기존의 칼라색상을 이용한 움직임 탐지방법에서는 영상보안 목적의 움직임 탐지를 위해 차영상을 기반으로 한 방법을 사용하였다. 차영상 움직임 탐지 기법은 이전 영상과 현재 영상의 차이가 주어진 임계값 이상인 경우를 움직임으로 탐지하는 방법이다. 감시카메라로부터 실시간으로 전송되는 영상데이터를 저장함과 동시에 움직임탐지시스템으로 전송한다. 움직임탐지시스템에서는 움직임이 탐지되면 움직임 부분에 대한 이동 물체의 윤곽부분영상을 캡쳐하여 정지영상을 저장하고, 움직임 탐지 시점의 전후 일정 시간에 대한 동영상을 별도 저장하는 시스템을 구현하였다. 연구에 사용된 영상데이터의 크기는 640x480 프레임 크기를 기본 값으로 칼라영상데이터를 기준으로 구현되었다.[1]

이전영상과 현재영상의 각 픽셀을 구성하는 RGB의 각 바이트마다 하위 4비트가 모두 다른 값을 갖게 되면 서로 다른 픽셀로 판정하고, 전체 이미지의 10% 이상의 서로 다른 픽셀이 존재하는 것으로 판정이 되면 움직임이 있는 것으로 탐지를 하는 방법이다. 움직임 탐지가 판정되면 움직임 영역 추출 작업을 위하여 이미지 차 연산을 반복 적용하는 방법을 사용하였다.

본 논문에서 이 방법을 구현하여 실험한 결과에 의하면 움직이는 배경이 없고 조명이 일정한 경우에는 어느 정도의 성공률을 나타내었다. 그러나 일반 자연 환경에서 촬영된 영상에서는 움직이는 배경, 자연 환경의 변화되는 햇빛의 변화되는 조명 등의 상황으로 인하여 오탐지의 오류가 높게 나오는 문제점이 나타났다. 그리고 RGB 색상모델에서의 각 RGB의 하위 4비트만을 사용함으로써 색상정보에 대한 의미를 제대로 살리지 못하

는 제한점이 존재했다. 다시 말하면 색상의 차이가 있는 전경 물체의 움직임에 대해서도 정확히 움직임으로 탐지하지 못하는 문제점이 존재한다. 본 논문에서는 이러한 문제점을 해결하기 위한 효율적 칼라영상정보 처리 방법을 제안하여 구현하였고, 이를 비교 분석하였다.

2.3 Background Subtraction Using YUV Color

배경 차영상 기반의 움직임 탐지 방법은 움직이는 전경 물체 이외의 배경영상 정보를 활용하는 방법이다. 즉, 배경영상과 현재 영상의 차이가 주어진 임계값 이상인 경우에 움직임으로 탐지하는 방법이다. 이때 배경영상은 촬영된 동영상 정보로부터 시간 변화에 따른 영상정보의 평균값, 중간값 등을 계산하여 배경영상으로 사용하고 있다. 움직임 탐지방법을 통한 홈 모니터링 서비스 시스템은 YUV 칼라색상모델을 사용하고 배경 차영상 기반의 움직임 탐지 방법을 사용하였다.[2]

배경 차영상 기반의 움직임 탐지 방법은 햇빛 등 조명의 변화, 디지털화시 발생되는 노이즈 형태의 색상변화 등에 있어서 이전 영상과의 차영상 기반의 방법 보다 안정적인 장점이 있다. 다만 바람에 움직이는 나무와 같은 움직이는 배경에 대해 여전히 움직이는 물체로 잘못 탐지하는 문제점이 있다. 이와 같은 오탐지 문제를 개선하기 위하여 배경영상 정보에 대한 통계적 특성을 이용함으로써 움직이는 배경에 대해서도 보다 안정적으로 탐지할 수 있는 방법을 적용할 수 있다.[3]

홈모니터링서비스시스템에서는 YUV 칼라모델을 사용하여 움직임 탐지 시스템을 구현하였다. 이때 YUV 칼라모델의 Y성분만을 사용하여 배경 모델링 및 움직임 판별에 사용함으로써 실질적으로 온전한 칼라 영상 정보를 사용하지 않는 방법으로 구현되었다. 따라서 명도에 의한 차이를 갖는 경우에 대해서는 정상적인 움직임 탐지가 가능하지만 동일한 명도를 가지면서 다른 색상을 갖는 경우에는 올바른 움직임 탐지가 어려운 문제점이 있다.

III. Proposed Scheme

자동 전경 움직임 탐지 시스템을 구축하는 방법으로는 크게 이전 영상과 현재 영상의 차이 값을 이용하는 차영상 방법과 움직이는 전경 물체를 제외한 배경영상을 구하고 배경영상과 현재 영상의 차이값을 이용하는 배경 차영상 방법이 일반적이다. 본 논문에서는 보안시스템 등 자연 환경에 많이 설치 운영되는 상황을 고려하여 이러한 환경에 더 적응력이 있는 배경 차영상 방법을 중심으로 전경 움직임 탐지 시스템을 구현하였다.

본 논문에서 제안한 움직임 탐지 방법은 전통적인 배경 차영상 방법에 근간을 두면서, 디지털 영상의 칼라 색상정보를 움직임 탐지에 적합한 형태로 처리하는 새로운 방법을 적용하였으며, 감시용 카메라가 설치되어 사용되는 자연 환경에서 일반적으로 존재하는 조명의 변화, 나무와 같은 움직이는 배경이 존재하는 상황에서도 보다 안정적인 움직임 탐지가 가능한 배경영상 처리에 대한 새로운 방법을 제시하였다.

3.1 Implementation Environments

본 논문에서 처리하는 디지털 영상데이터는 실제로 보안 감시용으로 사용되고 있는 CCTV의 영상데이터와 기존의 방법과의 성능 차이를 비교하기 위한 목적으로 편집된 영상데이터를 병행하여 입력 자료로 사용하였다. 보안 감시용으로 설치된 CCTV의 영상데이터는 다양한 환경적인 노이즈가 포함되어 있으며, 특히 움직이는 나무 등으로 잘못된 탐지가 발생될 가능성이 많이 존재하는 영상데이터이다.

본 연구에 사용된 CCTV는 네트워크 카메라 EasyCAM-200V Pro라는 제품을 사용하였다. 이 네트워크 카메라는 고화질급인 200만 화소의 1920x1080 크기까지 녹화가 가능하며 인터넷이 연결되는 곳이면 어느 곳에서나 설치하여 사용할 수 있는 제품이다. 또한 특별한 녹화 장비가 필요 없고 일반 PC에 업체에서 제공하는 관제 프로그램만 설치하면 쉽게 설치하여 사용할 수 있도록 되어있다. 또한 스마트폰 어플을 설치하면 스마트폰을 이용하여서도 실시간으로 설치된 네트워크 카메라를 통하여 감시할 수 있도록 되어있다.

본 논문에서는 네트워크 카메라의 관제 프로그램에 의해 녹화된 디지털 영상데이터를 읽어 들여 움직임을 탐지하는 방식으로 처리하였다. 구현된 시스템의 실험에 사용된 디지털영상데이터의 해상도는 640x352 크기의 영상데이터를 사용하였다. 주간 시간에 칼라색상정보를 포함하고 있는 영상데이터를 실험에 사용하였다.

본 논문에서 제안된 알고리즘을 구현하는 프로그래밍 언어는 Visual Studio 2012 C/C++과 OpenCV 라이브러리를 사용하였다. OpenCV는 오픈소스로 운영되는 컴퓨터 비전(Computer Vision) 관련 라이브러리이다. OpenCV는 소스가 공개되어 있어 사용자가 자신의 환경에 맞게 재 컴파일하여 사용할 수 있으며, 윈도우스, 리눅스, iOS, 안드로이드 등 다양한 플랫폼에서 C, C++, Java 등 다양한 언어와 함께 사용할 수 있도록 되어 있다. OpenCV의 라이브러리는 영상 및 비디오의 입출력, 영상처리 및 컴퓨터 비전 관련 알고리즘이 구현되어 있고 지속적으로 버전이 업그레이드되고 있으며, 본 논문에서는 최신 버전 3.0을 사용하였다.[4]

개발에 사용된 컴퓨터는 Intel Core2 Duo 2.0Ghz CPU와 3GB RAM이 탑재된 Windows 7환경이다.

3.2 Overview of Color Model

디지털 영상을 표현하기 위한 색상의 표현은 이를 표시하는 장치에 따라 서로 다른 색상모델(color model)을 사용한다. 일반적으로 컴퓨터에서는 칼라 영상을 표현하기 위해 빨강(Red), 파랑(Blue), 녹색(Green)의 3가지 색상의 가산 조합으로 만들어지는 RGB 색상모델을 사용한다. 이 외에도 사용하는 장치에 따라 YIQ, YCbCr, CMY 등의 색상모델이 존재하며, 이들 색상

모델간에는 상호간에 변환이 가능하다.

본 논문에서는 사람이 색상을 인식하는 정보 체계와 유사한 색상 모델을 만들기 위해 고안된 색상 모델인 HSI 색상 모델을 사용하였다. H는 색상(Hue)을 의미하고, S는 채도(Saturation), I는 명도(Intensity)를 의미한다. 본 논문에서 HSI 색상 모델을 사용한 이유는 사람이 인지하는 색상에 대한 인식 체계 및 움직이는 물체의 인식에 있어서 사람의 색상 정보의 활용 개념을를 본 논문의 구현 시스템에 적용시키기 위한 것이다.

RGB 색상 모델과 HSI 색상 모델의 관계식은 식 1과 같으며, 이를 통하여 본 논문에서 사용한 색상 정보의 체계를 설명하고자 한다. 식 1에서 설명된 것과 같이 HSI 색상 모델에서 I 요소는 영상의 명도를 나타내는데 R,G,B 색상 요소의 평균 값을 사용함으로써 색상의 차이를 반영한 값의 역할도 하고 있는 요소이다. 본 논문에서는 움직이는 영상의 탐지에 I요소의 값을 일차적으로 사용하였다. 대부분의 경우에 I요소의 값에 의해 움직이는 물체의 판단 근거로 이용될 수 있다.

뿐만 아니라 H, S, I의 모든 요소를 사용한 움직임 탐지를 하는 것은 조명의 변화 등에 따른 영향을 움직임으로 잘못 판단하게 만드는 문제점을 발생시킨다.

$$H=\begin{cases}\theta & \text{if } B\le G\\ 360-\theta & \text{if } B>G\end{cases}$$

$$\theta=\cos^{-1}\left(\frac{\frac{1}{2}((R-G)+(R-B))}{((R-G)^2+(R-B)(G-B))^{\frac{1}{2}}}\right) \quad (1)$$

$$S=1-\frac{3*\min(R,G,B)}{(R+G+B)}$$

$$I=\frac{(R+G+B)}{3}$$

그러나 I요소는 칼라정보가 배제된 명도에 대한 정보만을 나타내고 있기 때문에 명도의 차이만으로 구별되지 않는 경우에는 정상적인 움직임 탐지가 불가능하게 된다. 예를 들면 그림 2와 같이 배경은 빨간색(RGB=(255,0,0))이고, 초록색 객체(RGB=(0,255,0)), 파란색 객체(RGB=(0,0,255))가 존재 한다고 가정해 보았다. 움직임 탐지를 위하여 I 요소인 명도 정보만을 비교해 보면 빨강색 배경의 명도는 85(I=(255+0+0)/3=85)이고, 초록색 객체의 명도는 85(I=(0+255+0)/3=85), 파란색 객체의 명도는 85(I=(0+0+255)/3=85)로서 배경영상으로부터 전경물체를 전혀 탐지하지 못하는 상황이 된다.

(1) Background Image (2) Foreground Image

Fig. 2. Example of image that can not detect motion by brightness

따라서 본 논문에서는 H, S, I 요소에 대한 정보처리 단계를 구분하여 움직임 탐지에 가장 적합한 처리 절차를 고안하였다. 첫 번째 처리과정은 명도 요소인 I 요소에 대해 현재 영상과 배경 영상에 대한 명도 차영상을 구하고 이에 적절한 임계값을 적용하여 결과를 사용하였다. 두 번째 처리과정은 명도 차영상을 기준으로 하여 H 요소와 S 요소의 색상 정보의 비교가 필요한 영역을 결정하고 이 영역에 대해서 H요소와 S요소를 고려한 색상 차영상을 구하고 이에 적절한 임계값을 적용하여 결과를 사용하였다. 그리고 명도 차영상과 색상 차영상을 병합한 차영상 결과를 획득하고 이를 근거로 움직임 탐지를 수행함으로써 명도와 색상의 차이를 모두 활용한 움직임 탐지가 가능하도록 하였다.

3.3 Overview of Using Color Information

본 논문에서 제안한 자동 움직임 탐지 알고리즘은 칼라 정보를 효율적으로 사용하는 새롭게 고안된 알고리즘을 사용함으로써 기존 그레이 스케일 영상을 기반으로 한 방법의 효율성을 최대한 유지하면서 칼라 정보가 빠진 상태에서 움직임 탐지의 한계점을 극복할 수 있도록 하였다.

네트워크 카메라로부터 촬영된 디지털 영상은 다양한 색상 모델로 표현될 수 있으며, 본 논문에서는 HSI 칼라 색상 모델을 사용하였다. 저장된 디지털 영상은 OpenCV의 라이브러리인 inputVideo()함수를 사용하여 프레임 단위로 처리하였으며, 색상 모형은 cvtColor()함수를 사용하여 HSI 색상 모델로 변환하여 사용하였다.

본 논문에서 구현된 HSI 색상 모델 기반의 영상의 표현은 색상(Hue), 채도(Saturation), 명도(Intensity, Value)의 3개의 요소로 표현되며, 디지털 영상의 현재 프레임인 색상 모형의 각 요소는 식 2와 같은 내용으로 표현된다.

$$curFrameHue=\{h(i,j)|0\le h(i,j)<180, i<height, j<width\}$$

$$curFrameSat=\{s(i,j)|0\le s(i,j)<256, i<height, j<width\} \quad (2)$$

$$curFrameVal=\{v(i,j)|0\le v(i,j)<256, i<height, j<width\}$$

지금까지 움직임 탐지에 칼라영상을 사용하는 경우에 일반적으로 RGB 색상모델을 사용하는 경우가 있었으며, 각 R, G, B 요소의 상위 4비트만을 사용하여 움직임 여부를 판단하였다. 본 논문에서도 RGB색상모델에 대한 이와 같은 방법에 대한 검증을 위해 구현하여 실험한 결과, 각 R, G, B 요소의 전체 8비트를 사용하는 것보다, 상위 4비트를 사용하는 방법이 조명 변화 등에 덜 민감하게 되어 오탐지의 발생을 줄이는 효과가 있었다. 그러나 상위 4비트만을 취하는 것에 대한 합리적 근거가 부족하였고, 그레이스케일 영상을 이용한 움직임 탐지보다 실제적으로 오탐지의 발생율이 높은 문제점을 보였다.

그레이스케일 영상을 이용한 움직임 탐지 방법보다 칼라영상을 이용한 움직임 탐지 방법이 훨씬 안정되지 못한 탐지 결과를 보이는 것으로 실험결과 확인되었다. 칼라영상이 그레이스케일 영상 보다 3배 이상의 영상 정보 양을 가지고 있음에도 불구하고 움직임 탐지에 있어서는 오히려 더 높은 오탐지를 하게 만드는 원인을 찾고자 노력하였다.

이 문제를 해결하기 위해 사람이 움직이는 물체를 인지하는 과정에서 색상정보를 이용하고 인지하는 경험적 과정을 기반으로 본 논문에서의 색상정보 처리 절차 및 체계를 설계하였다. 사람의 색상 인지 체계를 그대로 반영한 색상 모델이 HSI 색상 모델이며, 따라서 본 논문에서는 이와 같은 사람의 색상 인지 체계를 사용하기 위해 이 HSI 색상모델을 채택하였다.

또한 본 논문에서 HSI 색상 모델을 기반 하여 움직임 탐지를 위한 실험을 수행한 결과 색상(H), 채도(S), 명도(I) 요소에 대한 가장 적절한 처리 절차를 새롭게 찾을 수 있었다. 본 논문에서 사용하는 차영상 기반의 움직이는 물체를 탐지함에 있어서 일반적으로 사용하는 방법은 색상모델의 각 요소의 값의 차이 값을 계산하여 만들어진 차영상을 사용하는 것이다. 예를 들면, RGB색상 모델에서는 각 요소인 R, G, B 요소의 차이 값을 사용하는 것이고, HSI 색상 모델에서는 각 요소인 색상(H), 채도(S), 명도(I) 요소의 차이 값으로 만들어진 차영상을 이용하는 것이다.

그러나 이 차영상 방법에 있어서 문제가 되는 상황은 정상적인 움직이는 물체를 탐지함에 있어서 불필요하게 색상 정보의 값이 매우 민감하게 표현되어 있다는 것이다. 이러한 문제점을 해결하기 위해 다른 연구에서는 RGB 요소의 각 상위 4비트만을 사용하는 등의 방법을 사용하고 있다. 그러나 이와 같이 일괄적인 특정 비트 이하는 버리는 방법은 칼라 영상을 사용하는 의미를 살리지 못하는 또 다른 문제점이 발생되어 탐지되어야 할 움직임을 탐지하지 못하고, 또 불필요한 색상의 민감한 반응에 대한 적절한 효과도 많이 줄어들지 않는 문제점이 실험 결과 발견되었다.

그래서 본 논문에서는 이 문제점을 해결하기 위해 HCI색상모델을 채택하고 움직임 탐지에 적절하게 색상정보를 활용하는 방법을 고안하였다. 첫 번째 절차로 명도요소를 이용하여 차영상을 도출하였다. 이 명도 요소는 그레이스케일 영상과 동일하게 색상에 대한 정보가 제외되어 있는 채로 밝기의 차이만을 이용한 차이로 움직이는 물체의 부분을 나타내는 역할을 한다. 그러나 이 결과는 당연히 그림 2와 같이 동일한 밝기를 가지고 있지만 색상이 차이가 나는 경우에 대해서는 구분을 하지 못하는 상태이다.

이제 색상 요소에 해당하는 색상(H) 요소에 대한 차영상을 도출 하였다. 다만 색상 요소는 앞에 기술한 것과 같이 매우 민감하게 변화하는 특성을 갖고 있으므로 이 문제를 줄이기 위해 다음과 같은 방법으로 차영상을 도출하였다. 즉, 명도 요소의 차영상 결과에서 차이가 없는 부분에 대해서만 색상 요소에 대한 차영상의 계산을 수행하였다. 또한 채도 요소의 특정 값 이하의 영역에 대해서도 제외시킴으로서 의미 있는 색상을 가지고 있는 영역에 대해서만 차영상의 결과를 만들 수 있도록 하였다.

이렇게 색상정보를 크게 명도에 대한 차영상인 diffVal과 색상에 대한 차영상인 diffHue의 두 차영상으로 분리하여 도출하였다. 그리고 이 두 차영상 diffVal과 diffHue를 병합하여 움직이는 물체를 탐지하기 위한 차영상 diffMask로 사용하였다.

3.4 Automatic Foreground Motion Detection Using Color Information Algorithm

칼라색상 정보를 이용한 전경 움직임 탐지 및 추출 알고리즘은 크게 차영상계산, 움직임판단, 전경추출, 배경영상갱신의 4단계로 구성되어 있다. 첫 번째 단계인 차영상계산 단계는 디지털 영상데이터의 현재 영상과 배경영상 사이의 변화된 부분을 찾아내기 위한 차이 값을 명도요소와 색상요소를 구분하여 계산하는 단계이다.

두 번째 단계인 움직임판단 단계는 앞 단계에서 계산된 두 종류의 차영상, 즉 명도 차영상과 색상 차영상의 결과에 대해 먼저 명도 차영상에 대하여 움직임 판단을 위한 영상처리를 시행한다. 명도 차영상의 처리 결과를 통하여 색상 정보를 이용해야할 영역, 즉 명도 차이에 의해서 움직이는 전경 물체에 대한 판별이 불가한 영역을 선택한다. 본 논문에서는 이 영역을 색상 영역(maskColor)으로 선정하여 이 색상 영역에 대해서 색상 차영상을 계산하였다. 이렇게 구해진 명도 차영상과 색상 차영상의 결과를 움직임 판단에 사용하기 위한 하나의 움직임 판단 영상(mask)을 구하였다. 이 움직임 판단 영상은 판단 근거가 되는 부분을 흰색(255) 값으로, 그렇지 않은 부분은 검정색(0) 값으로 표시하고, 이를 이용하여 움직임 판단 영상의 흰색이 전체의 10%이상 존재할 때 움직임이 있는 것으로 판단하였다.

세 번째 단계인 전경추출 단계는 전 단계의 전경 움직임판단 단계의 결과가 움직임이 있는 것으로 판단된 경우에 탐지된 전경 물체를 보다 정밀하게 추출하기 위한 과정이다.

네 번째 단계인 배경영상갱신 단계는 현재 영상의 내용 가운데 움직이는 전경 물체가 아닌 부분을 배경영상에 누적 평균값으로 갱신하는 단계이다. 이 과정을 통하여 움직이는 배경이 불필요하게 움직이는 전경 물체로 잘못 탐지 되지 않도록 하는 역할을 하게 된다. 또한 본 논문에서는 단순한 배경영상에 대한 누적 평균만을 적용하지 않고, 움직이는 배경의 특성과 낮과 밤의 교차, 배경의 새로운 변화의 내용도 배경 영상에 잘 반영되어 오탐지의 원인이 되지 않도록 개선하였다.

1) Background Subtraction Phase

차영상 계산단계는 디지털영상 데이터로부터 프레임 단위로 현재 영상데이터를 추출하고, 현재 영상데이터와 배경영상데이터 사이의 차이값을 계산하는 단계로써 칼라 색상 정보를 효율적으로 사용하기 위해 고안된 본 논문에서 구현된 알고리즘의 개요는 그림 3과 같다.

명도 차영상 계산(calcDiffVal) 과정은 HSI 칼라 색상모델의 명도(I) 요소를 사용하여 현재 영상의 명도 요소(curVal)와 배경 영상의 명도 요소(avgVal) 사이의 차이값을 계산하는 과정이다. 즉 현재 영상과 배경 영상의 명도의 차이값에 의해 구별

되는 부분을 찾아내기 위한 과정이다. 명도 요소의 차이값은 아래 식 3과 같이 계산되어 명도 차영상(diffVal)에 저장된다.

$$diffVal(i,j) = abs(curVal(i,j) - avgVal(i,j)) \quad (3)$$

칼라 색상 영역 선택(makeColorMask) 과정은 명도 차영상 계산 과정에서 계산된 명도 차영상(diffVal)의 결과를 사용하여 칼라 색상을 사용할 영역을 찾아내는 과정이다. 즉, 현재영상과 배경영상에 구별되는 전경 물체가 존재함에도 불구하고 현재 영상의 명도와 배경 영상의 명도의 차이가 크지 않아서 명도 차영상만으로는 구별되지 않는 영역을 칼라 색상 영역(maskColor)으로 선택하는 것이다.

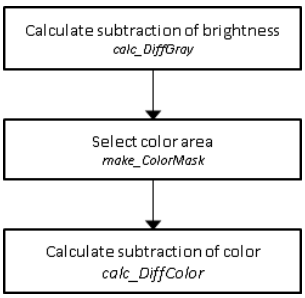


Fig. 3. Overview of Background Subtraction Algorithm

색상 차영상 계산(calcDiffColor) 과정은 HSI 칼라 색상 모델의 색상요소(H)와 채도요소(S)를 사용하여 칼라 색상 영역에 대해서 현재 영상의 색상요소와 배경 영상의 색상요소 사이의 차이값을 계산하는 과정이다. 즉, 현재영상과 배경영상의 명도 요소에 의해서 구별이 안되는 영역에 대해 색상 요소의 차이값을 구하여 그 결과를 색상 차영상(diffHue)에 식 4와 같이 저장한다.

$$diffHue(i,j) = abs(curHue(i,j) - avgHue(i,j)), \quad \text{if } maskColor(i,j) \text{ is } true \quad (4)$$

또한 의미있는 색상값에 대한 색상 차영상을 구하기 위하여 채도요소에 대해 미리 정해진 임계값 이상의 색상영역에 대해 색상 차영상을 구하였다.

2) Foreground Motion Detection and Extraction Phase

움직임 판단 단계는 전 단계인 차영상 결과를 이용하여 현재 영상에 움직이는 전경 물체가 존재하는지 여부를 판단하는 단계로써 이에 대한 알고리즘은 그림 4와 같다. 명도 움직임 탐지(detectMotionVal) 과정은 전 단계에서 구해진 명도 차영상 결과를 이용하여 움직이는 전경 물체의 차이값에 해당하는 영역을 미리 정해진 임계값을 사용하여 판단하고, 그 움직이는 전경 물체가 존재하는 영역을 나타내는 부분을 명도 움직임 탐지 마스크(maskVal)에 저장한다.

색상 움직임 탐지(detectMotionHue) 과정은 전 단계에서 구해진 색상 차영상 결과를 이용하여 움직이는 전경 물체의 차이값에 해당하는 영역을 미리 정해진 임계값을 사용하여 판단하고, 그 움직이는 전경 물체가 존재하는 영역을 나타내는 부분을 색상 움직임 탐지 마스크(maskHue)에 저장한다. 이때 명도 움직임 탐지와 다른 임계값을 사용함으로써 디지털 영상데이터에서의 색상의 특성을 고려하고 움직임 탐지에 적정한 임계값을 사용할 수 있다는 것이 기존의 방법 보다 안정적인 움직임 탐지 결과를 나타내는 요인이 되고 있다.

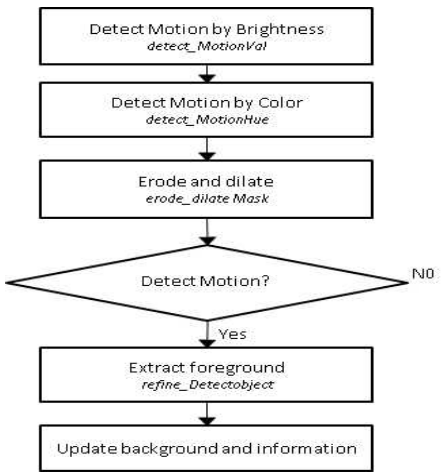


Fig. 4. Overview of foreground motion detection

노이즈제거및팽창(ErodeAndDilate) 과정에서는 먼저 명도 움직임 탐지 마스크(maskVal)와 색상 움직임 탐지 마스크(maskHue)를 논리 합 연산자로 병합하여 움직임 탐지 마스크(mask)를 식 5와 같이 생성하였다.

$$mask(i,j) = maskVal(i,j) \| maskHue(i,j) \quad (5)$$

이렇게 명도에 의한 차영상 결과와 색상에 의한 차영상 결과를 결합한 결과에 대해 노이즈 제거와 팽창과정을 반복 수행하여 칼라 색상을 이용한 움직이는 전경 물체 탐지를 위한 움직임 탐지 마스크를 준비하였다. 전경 움직임 판단은 움직임 탐지 마스크의 움직임을 나타내는 픽셀의 비율이 10%이상인 경우에 움직임 탐지로 판단한다. 전경추출 단계는 움직임 탐지 판단 결과가 움직임으로 결정된 경우에 현재 영상으로부터 전경 물체 부분을 추출하는 과정이다.

3) Background Updating Phase

배경영상 갱신 단계는 디지털 영상데이터로부터 움직이는 전경 물체를 제외한 배경영상을 생성 및 유지하는 단계이다. 배경 차영상을 이용한 방법에서 움직이는 전경물체를 정확하게 탐지하기 위해서는 배경영상의 정확성이 많은 영향을 준다고 볼 수 있다. 본 논문에서는 감시 카메라와 같은 CCTV가 설치되어 운영되는 자연환경에 존재하는 움직이는 배경영상을 반영하기 위하여 개선된 배경영상 생성 및 유지 방법을 고안하였으며 이에 대한 알고리즘은 그림 5와 같다.

움직임특성분석 과정은 움직이는 배경에 대한 특성을 이용

하여 전경 물체로 오탐지 되지 않고 배경으로 반영될 수 있도록 하기 위한 과정으로서 크게 두 가지 특성을 사용한다. 첫 번째 특성은 현재 영상과 이전 영상의 차이값을 이용하는 것으로서 이 값이 규칙적인 차이값을 갖는 경우에 이를 움직이는 전경 물체가 아닌 움직이는 배경으로 반영하기 위한 처리 과정이다.

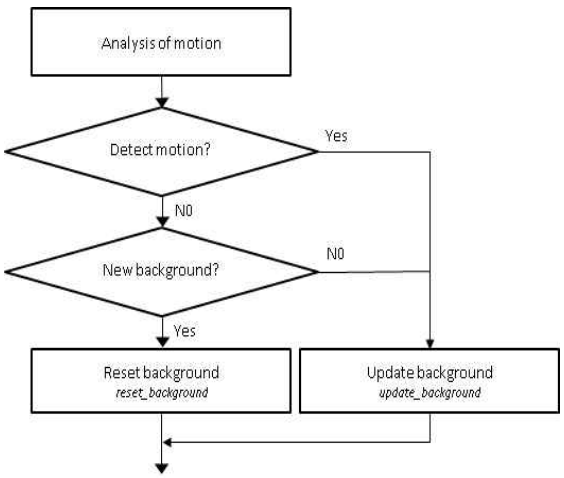


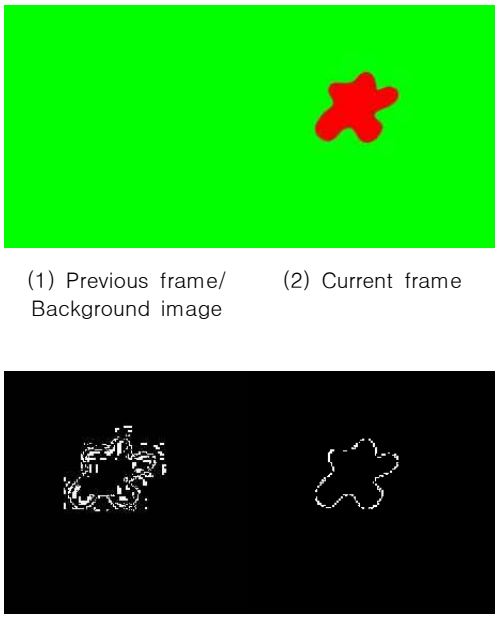
Fig. 5. Overview of updating background Algorithm

두 번째 특성은 기존 배경에 새로운 배경에 해당하는 물체가 추가되는 경우로서 특별한 처리를 두지 않는 경우에는 움직이는 전경 물체로 잘못 탐지되는 상황이 만들어진다. 이를 처리하기 위해 탐지결과로 판단되면서 이전영상과 현재영상의 차이값이 미리 정해진 임계값 이하인 경우에는 배경영상에 현재영상의 반영 비율을 높여 결과적으로 배경영상을 리셋하는 효과를 주도록 하였다.

3.5 Analysis of Experimental Result

그림 6은 본 논문에서 구현한 시스템과 기존 방법의 성능을 비교하기 위하여 편집된 영상의 이전영상과 현재영상의 한 부분을 보여주고 있다. 먼저 기존의 움직임 탐지의 한 방법인 이전영상과 현재영상의 차영상을 이용하는 방법의 구현 결과를 실험하였다. 이 방법은 RGB 색상의 하위 4비트를 사용하여 RGB 각 요소가 모두 다른 픽셀의 값이 전체 영상의 10% 이상인 경우에 움직임으로 탐지하는 방법이다. 실험결과 그림 6의 (3)에서 보이는 결과는 흰색으로 나타낸 부분이 움직이는 전경 물체 부분을 나타내는 것으로 보이는 결과와 같이 정상적으로 움직이는 전경 물체를 인식하지 못하는 결과를 보이고 있다. 즉, 이전영상과 현재영상의 차영상결과가 전체 영상의 10%를 넘지 못하여 움직임으로 탐지하지 못한 결과이다.

그림 6의 (3)의 영상은 10%이하의 차영상 결과를 보여주고 있다. 칼라 색상정보를 사용하였지만 하위 4비트만을 일괄적으로 선택하여 사용함으로써 정상적인 색상정보의 활용이 되지 못한 것이 이 방법의 문제점으로 분석되었다. 또다른 기존의 움직임 탐지의 한 방법은 배경 차영상을 이용한 방법이며, YUV 색상모델을 사용하였으며, 움직임 탐지에는 Y요소만을 이용하였다. 즉, 디지털영상의 명도요소만을 이용하여 움직임 탐지를 하는 방법이다. 본 논문에서 이 방법에 대한 비교 실험을 위하여 이 방법대로 구현하고 실험한 결과의 한 부분을 그림 6의 (4)에 나타난 배경 차영상 결과로 보였다.



(1) Previous frame/ Background image (2) Current frame

(3) Difference between previous frame and current frame (4) Difference between background image and current frame

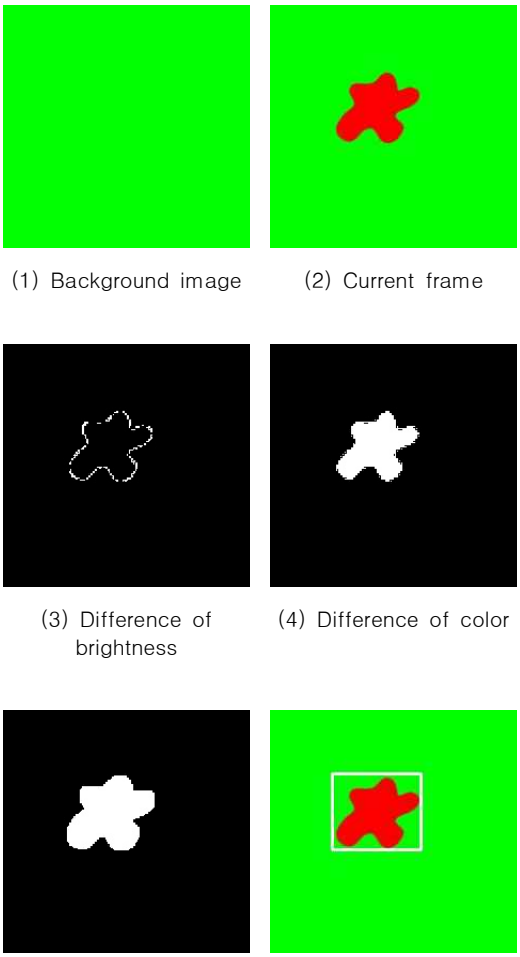
Fig. 6. Sample video

색상 정보를 사용하지 않고 명도요소인 Y요소만을 사용함으로써 움직이는 전경 물체를 정상적으로 인식하지 못하여 움직임으로 판단하지 못하는 배경 차영상 결과를 보이고 있다. 물론 이 방법은 명도요소의 차이만으로 움직이는 전경 물체를 구분하는 경우, 즉 색상에 대한 정보 없이 명도 요소만으로 구별되는 경우에는 정상적인 움직임 탐지의 역할을 할 수 있다. 하지만 위 그림 6의 샘플 디지털영상과 같이 서로 다른 색상을 가지면서 동일한 명도값을 갖는 경우에는 정상적인 움직임 탐지를 하지 못하는 문제점을 갖고 있다.

그림 7에서는 디지털영상의 색상정보 활용에 있어서 기존 방법과의 처리 결과 및 성능을 비교하기 위하여 앞에서 사용되었던 편집된 샘플 디지털영상을 사용하여 본 논문에서 구현한 시스템의 처리 과정을 보여주고 있다. 그림 7의 (1) 배경영상은 디지털영상에서 움직이는 전경물체를 제외한 배경영역을 누적평균값을 구하여 생성되고 관리되는 배경영상이다. 그림 7은 디지털영상의 색상정보 활용에 대한 설명을 위해 움직이는 배경이 없는 편집된 영상에 대한 누적평균 배경영상을 보이고 있다. 그림 7의 (2) 현재영상은 움직이는 전경 물체가 나타나있는

영상이다.

그림 7의 (3) 명도 차영상은 배경영상의 명도요소(intensity)와 현재영상의 명도요소 사이의 차이값을 이용한 차영상이다. 명도 차영상 결과에 보인 것과 같이 움직이는 전경 물체를 인식하지 못하는 차영상 결과를 나타내고 있다. 즉, 배경영상과 움직이는 전경물체의 명도 요소의 차이에 의해서는 움직이는 물체를 인식하지 못하는 상태를 보이고 있다.



(1) Background image (2) Current frame

(3) Difference of brightness (4) Difference of color

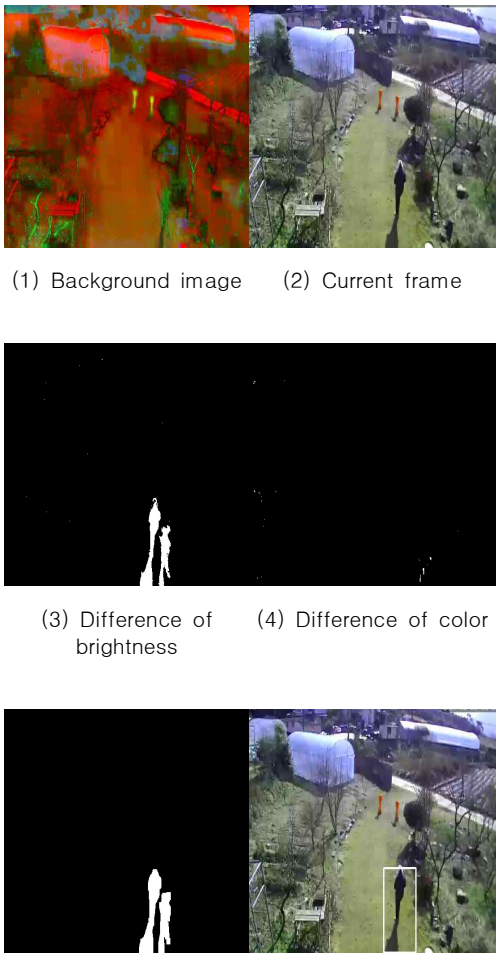
(5) Difference between background image and current frame (6) Extract foreground

Fig. 7. Procedure of Foreground Motion Detection Using Color Information

그림 7의 (4) 색상 차영상은 배경영상의 색상요소(hue)와 현재영상의 색상요소 사이의 차이값을 이용한 차영상이다. 색상 차영상의 생성은 명도 차영상의 결과를 근거로 하여 색상 요소의 사용이 필요하다고 판단되는 영역에 대해서만 색상 차영상을 만들어 사용한다. 또한 채도(saturation) 요소의 값을 사용하여 의미 있는 색상의 값에 대해서만 색상 차영상을 계산하였다. 이와 같은 형태의 색상 차영상 계산 과정을 통하여 디지털영상 장치의 특성에 의해서 만들어지는 잘못된 색상값의 영향을 최소화할 수 있었다.

그림 7의 (5) 배경 차영상은 명도 차영상과 색상 차영상의 결과를 결합하고, 노이즈 제거 및 팽창의 과정을 적용한 차영상 결과이다. 노이즈 제거 및 팽창의 과정은 명도 차영상과 색상 차영상 각각에 대해서 서로 다른 임계값을 사용하여 처리하였고, 결합된 결과에 대해서도 적용함으로써 움직임 탐지의 성공률을 높일 수 있도록 하였다. 기존의 방법에서는 색상정보 전체에 대해서 노이즈 제거 및 팽창을 적용함으로써 노이즈를 정상적으로 제거하지 못하거나 움직이는 전경 물체를 인지하지 못하는 문제점이 있었던 부분을 본 논문에서는 개선할 수 있었다. 그림 7의 (6) 전경물체 추출 영상은 움직이는 전경 물체를 현재영상으로부터 추출한 결과를 보이고 있다.

그림 8에서는 실제 CCTV 카메라를 이용하여 촬영된 디지털 영상에 대한 움직임 탐지 과정 및 결과를 보이고 있다.



(1) Background image (2) Current frame

(3) Difference of brightness (4) Difference of color

(5) Difference between background image and current frame (6) Extract foreground

Fig. 8. Examples of Foreground Motion Detection Using Color Information

제안된 움직임 탐지 방법을 기존의 방법과 표 1과 같이 비교

분석하였다. 조명의 변화 또는 움직이는 배경 등이 포함된 디지털영상에서 정상적으로 움직이는 전경 물체를 탐지하는 과정을 보이고 있다. 배경영상은 앞의 배경영상갱신 과정에서 설명한 것과 같이 배경으로서의 특성을 분석하여 움직이는 배경을 배경영상에 누적 평균값으로 처리함으로써 움직이는 배경으로 인한 오탐지의 발생이 줄어드는 효과를 얻을 수 있었다. 또한 명도 차영상과 색상 차영상에 대해 서로 다른 임계값에 의해 노이즈 제거 및 팽창 처리를 수행함으로써 기존 방법에서 자연환경에서 촬영된 디지털영상의 칼라 색상을 사용할 경우에 빈번하게 발생하는 오탐지 발생의 문제점을 개선할 수 있었다. 본 논문에서는 주간에 촬영된 칼라색상정보를 포함하는 디지털영상데이터를 대상으로 하여 실험하였고 야간에 촬영된 디지털영상데이터는 칼라색상정보를 포함하지 않고 있는 이유로 본 논문의 실험에서는 포함하지 않았다.

Table 1. Comparative analysis of proposed method and existing method

	Method 1 RGB Color	Method 2 YUV Color	Proposed Method HSV Color
Detection operation	subtraction	background subtraction	background subtraction
Color identification	incomplete	impossible	complete
Detection rate	medium	medium	high
Changing illumination and moving background	weak adaptability	weak adaptability	improved adaptability
Processing speed	fast	fast	slow

그러나 본 논문에서 구현한 시스템을 통한 실험 결과를 통하여 향후 추가적으로 개선되어야할 몇 가지 문제점들도 발견되었다. 예를 들면, 명도 요소에 비하여 색상 요소의 값이 상대적으로 불안정한 값으로 표현되는 문제점이 여전이 남아 있다. 이에 대한 원인을 보다 상세히 분석할 필요가 남아 있다. 또한 보다 정교한 움직임 탐지를 위하여 여러 가지 기능을 추가함에 따라 이를 처리하는 소요 시간 및 용량이 커짐에 따라 실시간으로 처리해야 하는 상황인 경우에는 상대적으로 고성능의 컴퓨팅 자원을 요구하게 된다는 것이다.

편집된 디지털영상과 실제 CCTV로 촬영된 디지털영상을 이용하여 기존의 방법과 본 논문에서 제안한 방법을 비교해볼 때 색상정보를 보다 효율적 이용함으로써 기존 방법으로 탐지가 불가능했던 전경 물체를 탐지할 수 있었고, 다양한 움직이는 배경, 조명의 변화 등의 실제 영상에 대해서도 보다 안정적이고 정확한 움직임 탐지가 가능하다는 것을 확인할 수 있었다.

IV. Conclusion

사회 각 분야에서 다양한 목적으로 CCTV 등의 영상장비를 이용하고 있으며 이를 보다 효율적으로 이용할 수 있도록 하기 위한 영상정보 처리기술이 요구되어지고 있다. 영상장비에 대한 하드웨어 기술은 비약적으로 발달되어 저렴한 비용으로 고성능의 영상장비를 구축 및 운영하고 있는 상황이다. 그러나 디지털영상 데이터가 대용량화되고 관리 대상의 범위도 대단위화 됨에 따라 단순한 수작업에 의존해서만 사용자가 원하는 영상정보를 찾아내거나 관리하기는 갈수록 어렵고 불가능해져 가고 있는 상황이다. 특히, 시급을 다투는 보안, 치안 목적의 분야이거나 생명의 위급을 가르는 응급의료 분야와 같은 곳에서는 시스템에 의해 자동적으로 필요한 정보를 탐지하여 제공해 줄 필요가 있다. 이러한 관점에서 자동적으로 움직이는 전경 물체 를 탐지하는 방법에 대한 연구는 디지털영상 장비를 활용하는 다양한 분야에서 매우 중요한 부분을 차지하고 있다고 말할 수 있다.

이미 많은 분야에서 칼라 디지털영상 장비가 일반화 되어 있지만 칼라정보를 효율적으로 이용하여 움직임 탐지하는 방법에 대해서는 아직 많이 미진한 상황이다. 따라서 움직임 탐지의 많은 연구가 대부분 부분적으로 칼라 색상정보를 이용하거나 명도요소의 정보만을 사용하는 형태로 이루어지고 있다. 이러한 방법은 본 논문에서 실험한 결과에 의하면 대부분의 경우에 움직임 탐지를 수행할 수는 있지만 특정한 색의 전경 물체와 배경이 겹치는 상황에서는 정상적인 탐지가 불가능하다는 것을 확인할 수 있었다. 또한 색상정보에 대해 일괄적인 임계값을 사용하여 움직임 탐지를 수행하는 기존 방법으로는 오탐지를 발생하여 현실적으로 사용할 수 없는 문제점이 발견되었다.

본 논문에서는 이러한 문제점을 개선한 방식으로 칼라색상정보를 움직임 탐지에 사용하기 위하여 칼라색상정보를 명도요소와 색상요소로 구분하여 처리하였다. 이와 같이 명도와 색상을 구분함으로써 각 요소의 특성을 최대한 고려하여 처리하고 각각에 적정한 임계값을 사용함으로써 보다 안정적이고 정확한 움직임 탐지가 가능하게 되었다. 또한 본 논문에서 기반하고 있는 배경 차영상 기반의 움직임 탐지 방법은 정확한 배경영상을 유지, 관리하는 것이 정확한 움직임 탐지를 하는데 있어서 매우 중요한 역할을 하게 된다. 이를 위하여 본 논문에서는 배경영상 갱신 과정에 움직이는 배경에 대한 특성을 분석하여 자연환경에 존재하는 움직이는 배경을 배경영상에 반영하도록 하였고, 실험결과 보다 정확한 움직임 탐지의 결과를 얻을 수 있었다.

본 논문에서는 디지털영상의 전체 영역을 대상으로 하여 움직임 탐지 방법을 연구하였으나 향후 추가적으로 연구하고자 하는 부분은 영역을 분할하여 각 영역별로 서로 다른 기준을 근거로 움직임을 탐지하는 방법에 대해 연구를 진행하고자 한다.

REFERENCES

[1] Kyu-Woong Lee, “Implementation of Video Surveillance System with Motion Detection based on Network Camera Facilities”, The Journal of IIBC, Vol. 14, No. 1, pp.169-177, Feb. 2014.

[2] O,JiHHye, Ji-Sung Jung, Ahn, SungHo, "Motion Detecting Home Monitering Service System with Background Modeling", Journal of KIIT, Vol. 12, No. 9, pp.83-89, Sep. 2014.

[3] Hyung-Hoon Kim, Jeong-Ran Cho, "Effective Automatic Foreground Motion Detection Using the Statistic Information of Background", Journal of The Korea Society of Computer and Information, Vol. 20, No. 9, pp. 121-128, Sep. 2015.

[4] Jin Keun Seo, Sukho Lee, “Automatic Motion Detection Using False Background Elimination”, The journal of the Korean Society for Industrial and Applied Mathematics, Vol. 17, No. 1, pp.47-54, Jan. 2013.

[5] Yang, Ming-Jiang, et al., “Cost Effective IP Camera For Video Surveillance”, Industrial Electronics and Applications, ICIEA 2009, 4th IEEE Conference, 2009.

[6] D. K. Kim, “C++ API OpenCV Programming”, The Publish Company of KAME, May 2015.

[7] OpenCV Open Source Computer Vision, http://opencv.org

[8] AmmarAnuar, KhairulMuzzammilSaipullah, NurulAtiqah Ismail, and Soo Yew Guan, “OpenCV Based Real-Time Video Processing Using Android Smartphone”, International Journal of Computer Technology and Electronics Engineering, Vol. 1, No. 3, pp.58-63, 2011.

[9] C. Stauffer and W.E.L. Grimson, “Learning Patterns of Activity Using Real Time Tracking”, IEEE Trans. on Pattern Analysis and Machine Intelligence, No. 22, pp747-767, 2000.

[10] Jing Guo, Siong Chng Eng, Rajan Deepu, “Foreground Motion Detection by Difference-Based Spatial Temporal Entropy Image”, TENCON 2004. IEEE Region 10 Conference IEEE 2004, pp379-382, 2004.

[11] S.H. Lee and J.K. Seo, “Level Set-Based Bimodal Segmentation with Stationary Global Minimum”, IEEE Trans. on Image Processing, No. 9, pp2843-2852, 2006.

[12] X. Gao, T.E. Boult, F. Coetzee, and V. Ramesh, “Error Analysis of Background Subtraction”, IEEE Int. Conf. on Computer Vision, 2000.

[13] Hye-Youn Lim, Dae-Seong Kang, “The Moving Object Estimation Using an Efficient Background Extraction in the Outdoor Environment”, Journal of Korean Institute of Information Technology, Vol. 7, No. 3, pp226-231, 2014.

[14] Joon-Goo Lee, Ki-Sun Han, Byoung-Moon You, Doo-Sung Hwang, “Shot Boundary Detection Algorithm by Compensating Pixel Brightness and Object Movement”, Journal of The Korea Society of Computer and Information, Vol. 18, No. 5, pp35-42, 2013.

[15] Shin S. Y., Baik S. E., Pyo S. B., Rhee Y. W., “Scene Change Detection Using Local X2”, Journal of The Korea Society of Computer and Information, Vol. 15, No. 1, pp203-207, 2007.

Authors

Hyung Hoon Kim received the B.S. degrees in Computer Science from Chonnam National University, Korea, in 1986. He received the M.S. degrees in Computer Science from KAIST, Korea, in 1988. He received the Ph.D. degrees in Computer Science from Hanyang University, Korea, in 2007. Dr. Kim joined the faculty of the Department of Biomedical Systems at Kwangju Womens University, Gwangju, Korea, in 1994. He is currently a Professor in the Department of Biomedical Systems at Kwangju Womens University. He is interested in web programming, medical information system, information security, and AI.

Jeong Ran Cho received the B.S., M.S. and Ph.D. degrees in Computer Science from Chonnam National University, Korea, in 1987, 1989 and 1999, respectively Dr. Chojoined the faculty of the Department of Biomedical Systems at Kwangju Womens University, Gwangju, Korea, in 1994. She is currently a Professor in the Department of Biomedical Systems at Kwangju Womens University. She is interested in database, parallel computing, internet and mobile computing, and multimedia contents service.