

특정컬러정보 검출기반의 이동객체 탐색 알고리즘 구현

김영빈* · 류광렬* · 로버트 스크라바시**

*목원대학교 · **UPMC

Moving object Tracking Algorithm Based on Specific Color Detection

Young Bin Kim* · Kwang Ryol Ryu* · Robert J. Sclabassi**

*Mokwon University · **University of Pittsburgh Medical Center

E-mail : conan9977@yahoo.com

요 약

특정컬러정보 검출기반의 이동객체 탐색 알고리즘을 구현한다. 입력 이미지에 대해 조도변화 및 노이즈 제거 등을 위해 전처리 과정을 거치고, 이동객체 탐색은 R,G,B 각 채널의 영상차를 이용하여 객체를 검색한다. 실험 결과 검색 속도는 윤곽선 탐색 및 정합법에 비해 15% 향상되었고 안정적이다. 또한 컬러 정보 기반의 객체 탐색이 가능함을 제시하였다.

ABSTRACT

A moving object tracking algorithm for image searching based on specific color detection is proposed in this paper. That is preprocessed for a luminance variation and noise cancellation to be robust system. The motion tracking is used the difference between input image and reference image in R, G, B each channels for a moving image. The proposed method is enhanced to 15% fast in comparison with the contour tracking method and the matching method, and stable.

KEYWORDS

Object tracking algorithm, color detection, moving image, RGB difference images

1. 서론

보안 감시 시스템은 이동 물체를 검출하여 접근 및 이동경로 등을 인식 하고 관측자에게 긴급 상황을 알려야 한다. 이러한 비전 시스템에서 카메라를 이용한 이동 물체의 검출은 중요한 기술 요소이다. 그러나 기존의 비전 시스템 구현에는 많은 어려움이 있다. 움직임을 감지하기 위한 센서를 장착한 카메라나 객체를 추출하기 위해 많은 연산량을 필요로 하기도 한다. 따라서 실시간 객체 추적이 어렵다.[1,6,7] 객체를 추적하는 방법은 보안, 의료, 군사, 교통, 제어분야 등과 같이 다양한 응용분야로 인해 많은 연구가 이루어지고 있다. 기존의 추적 기법으로는 3차원 모델 기반을 이용한 방법, 영역 기반의 방법, 윤곽선 기반의 방법, 특징기반의 방법, 시공간 경사법, 무게 중심법, 정합법 등이 있다.[3] 윤곽선을 이용한 기법은 물체의 경계인 윤곽선을 검출하고, 윤

곽선의 이동 여부를 추적하는 방법이다. 영역 기반의 기법에 비해 연산량은 줄어드나 부분적인 가려짐이 발생할 경우 객체의 추적이 어렵다.[4] 정합법은 화소 자체의 정보나 객체의 특징을 추출하여 이미지에서 유사성이 최대인 정합점을 찾는 방법이다. 물체의 확장 및 축소, 또는 회전하는 경우 추적이 어려운 단점을 갖는다.[5]

본 연구에서는 감시 시스템 분야에서 적용할 수 있는 저가의 CCD 카메라를 이용하고, 카메라를 고정하고, 외부 환경이 아닌 실내 환경이며, 배경영상의 변화가 거의 없는 환경으로 한정하는 조건에서 입력영상에 대한 움직이는 특정 컬러의 객체를 검출하는 알고리즘을 제안한다.

II. 이동객체 탐색 알고리즘

컬러 정보를 갖는 객체 검색은 전처리 과정과

배경 이미지 생성, 입력 이미지에서 대한 컬러 탐색 과정을 거치고, 객체 검출 과정으로 나눌 수 있다. 구현 알고리즘의 처리과정은 그림 1과 같다.



그림 1. 구현 알고리즘 블럭도

2.1 전처리 과정

색상을 이용하여 일련의 작업을 하려고 할 때 큰 문제로 나타나는 것이 조명의 영향이다. 조명의 변화에 색상은 영향을 받지 않으나 휘도는 크게 영향을 받으므로, 이러한 조명에 의한 영향을 제거시킬 수 있는 정규화 과정을 기칠 필요가 있다. 전처리 과정에서는 조명 상태에 무관한 컬러 정보를 획득하기 위하여 입력영상에 대해 R, G, B 값의 정규화 과정을 수행한다. 모든 색상은 R, G, B의 서로 다른 파장의 길이를 가지는 세 가지 성분의 조합과 빛의 휘도성분으로 표현된다. 일반적으로 비디오 카메라는 RGB 모델을 많이 사용하고 있으며 이 모델은 다른 컬러 모델로 쉽게 변환이 가능하다. 각 화소는 각각 [0~255] 사이의 값을 가지는 R, G, B의 조합으로 표현되며, 이 모델은 하나의 색상뿐만 아니라 빛의 휘도까지 포함된 값이다. 따라서 이런 빛의 색상에 있어 식 (1)과 같은 관계를 가지고 있다면, 이 두 점은 같은 색상을 가지고 있으나 다른 휘도 값을 가지고 있다는 것을 의미한다.

$$\frac{r1}{r2} = \frac{g1}{g2} = \frac{b1}{b2} \quad \text{-----(1)}$$

식 (2)~(4)의 정규화 과정을 거쳐 순수한 컬러 정보만을 지닌 이미지로 변환 한다. 식 (5)에서 알 수 있듯이 r, g, b의 값은 비례적인 값을 나타낸다.

$$r = \frac{R}{R+G+B} \quad \text{-----(2)}$$

$$g = \frac{G}{R+G+B} \quad \text{-----(3)}$$

$$b = \frac{B}{R+G+B} \quad \text{-----(4)}$$

$$r + g + b = 1 \quad \text{-----(5)}$$

2.2 배경 이미지 생성

전체적인 배경 이미지에 대하여 탐색을 할 경우 많은 연산량이 요구 된다. 이는 실시간 객체 탐색의 성능을 저하시킨다. 따라서 본 논문에서는 NxM 마스크를 이용하여 배경영상을 갱신하고 객체 탐색을 하도록 한다. 입력 이미지와 배경 이미지의 NxM 마스크 내에서 각 픽셀의 RGB 채널에 대하여 차이 값을 구한다. 배경 이미지를 $R_{BI(i,j)}, G_{BI(i,j)}, B_{BI(i,j)}$ 라고 하고, 입력 이미지를 $R_{CI(i,j)}, G_{CI(i,j)}, B_{CI(i,j)}$ 라고 하였을 때, R, G, B 각 채널의 NxM 마스크 이미지를 $R_{mask}(i,j), G_{mask}(i,j), B_{mask}(i,j)$ 라고 하자. 배경 이미지와 카메라 입력 이미지와 차이값을 나타낸 것이 식 (6) 이다.

$$\begin{aligned} R_{mask}(i,j) &= |R_{BI(i,j)} - R_{CI(i,j)}| \\ G_{mask}(i,j) &= |G_{BI(i,j)} - G_{CI(i,j)}| \\ B_{mask}(i,j) &= |B_{BI(i,j)} - B_{CI(i,j)}| \end{aligned} \quad \text{---(6)}$$

각 채널의 차이값이 임계값인 α 보다 작다면 배경 이미지로 판단하여 입력 이미지의 마스크 영역을 배경 이미지로 갱신한다. 반대로 α 보다 크다면 새로운 객체 영역으로 판단하여 마스크 영역에 해당하는 배경 이미지를 변경하지 않는다. α 는 임의의 값인 20으로 적용하였다. 식 (7)~(9)은 배경 이미지 생성 시 입력 이미지와 배경 이미지의 차에 대한 임계값의 조건에 대한 수식을 채널별로 보이고 있다.

$$R_{BI}(i,j) = \begin{cases} R_{CI}(i,j), & \text{if } R_{mask}(i,j) < \alpha \\ R_{BI}(i,j), & \text{else} \end{cases} \quad \text{-----(7)}$$

$$G_{BI}(i,j) = \begin{cases} G_{CI}(i,j), & \text{if } G_{mask}(i,j) < \alpha \\ G_{BI}(i,j), & \text{else} \end{cases} \quad \text{-----(8)}$$

$$B_{BI}(i,j) = \begin{cases} B_{CI}(i,j), & \text{if } B_{mask}(i,j) < \alpha \\ B_{BI}(i,j), & \text{else} \end{cases} \quad \text{-----(9)}$$

2.3 특정 컬러의 객체 탐색

고정된 PC 카메라의 배경 이미지에서 특정 컬러

러를 갖는 물체를 검출하기 위한 방법은 배경 이미지와 현재의 카메라 이미지의 차연산을 이용한다. 배경 이미지와 현재 입력 이미지에 대하여 각 셀의 차이값이 임계값 이상이면 해당 셀은 동작 셀이고 임계값 이하이면 비동작 셀로 다룬다. 입력 이미지와 임계값 β 의 관계는 식 (10)에 나타내고 있다.

$$\begin{aligned} R_{\Delta}(i,j) &= \begin{cases} 1, & \text{if } R_{obj}(i,j) \geq \beta \\ 0, & \text{else} \end{cases} \\ G_{\Delta}(i,j) &= \begin{cases} 1, & \text{if } G_{obj}(i,j) \geq \beta \\ 0, & \text{else} \end{cases} \\ B_{\Delta}(i,j) &= \begin{cases} 1, & \text{if } B_{obj}(i,j) \geq \beta \\ 0, & \text{else} \end{cases} \end{aligned} \quad \text{-----}(10)$$

동작셀의 픽셀이 카메라 이미지의 노이즈와 구분하기 위해 15 이상의 연속된 픽셀이 동작 셀로 검출된 경우 취급하고, 이하라면 잡음으로 판단하도록 한다. 입력 이미지는 RGB 컬러 이미지이므로 각 채널에 대하여 컬러 조합비율에 의하여 일치하는 경우 검색하고자 하는 컬러의 물체로 판단하게 된다.

2.4 객체 영역 추출

R,G,B 컬러 비율을 이용하여 탐색한 객체의 픽셀 영역은 객체의 중심점을 찾아 최종 객체로 추출한다. 객체의 중심점을 $T(x,y)$ 라고 한다면, 중심점을 기준으로 하여 최소 사각 영역(MBR: Minimum Bounding Rectangle)을 설정한다. 최소 사각 영역은 탐색 객체의 사이즈로 인식한다. 객체 영역의 설정에 대한 수식은 식 (11)과 같다.

$$\begin{aligned} T_l &= \min(T_x) - \delta \\ T_r &= \min(T_x) + \delta \\ T_t &= \min(T_y) - \delta \\ T_b &= \min(T_y) + \delta \end{aligned} \quad \text{-----}(11)$$

III. 실험 및 고찰

실험은 CCD 카메라를 이용하여 320(H)x240(V)의 픽셀수를 갖는 이미지를 사용하였다. 배경 이미지와 입력 이미지의 영상차는 객체와 배경 이미지의 차이가 크도록 $\alpha=20$ 으로 설정하였고, 카메라 입력 영상이 노이즈 또는 밝기에 영향을 적게 받도록 $\beta=45$ 값을 설정 하였다. 검색하고자 하는 객체의 컬러는 Red, Blue, Green 세 가지 컬러에 대하여 15 픽셀 이상의 연속된 이동 객체를 검색 하도록 설정 하였다. 배경 이미지의 업데이트를 위한 마스크 사이즈는 8x8을 사용하였다.

입력 이미지에 대하여 선처리 한 후 그림 2와 같이 배경 이미지를 생성하였다. 그림 3은 배경 이미지에 녹색을 띄는 물체가 있는 이미지이다. 그림 3을 입력 이미지로 하고, 컬러 정보는 (R, G, B) = (120, 240, 220)으로 설정하였을 때 객체 탐색을 시도하였다. 그 결과 그림 4와 같이 24x22의 픽셀 크기를 갖으며, 객체 영역이 추출된 이미지를 볼 수 있었다. 평가를 위하여 기존의 윤곽선 기반법과 정합법을 이용하여 객체 탐색에 대한 탐색 속도와 검출율을 비교하였다. 그 결과 제안한 기법이 윤곽선 기반의 기법에 비하여 검출율이 떨어지기는 하지만 처리 속도는 15% 향상 되었다. 표1은 기존의 방법과 제안 기법의 성능 비교 결과 이다.

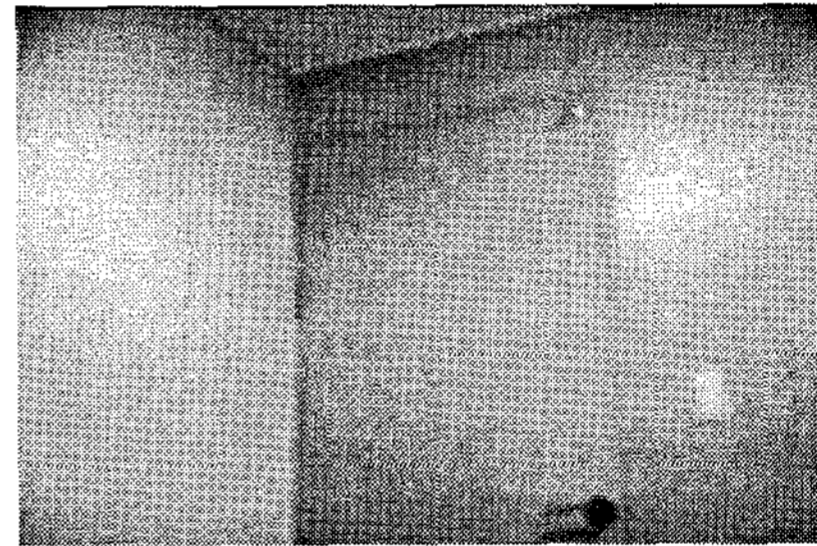


그림 2 배경 이미지

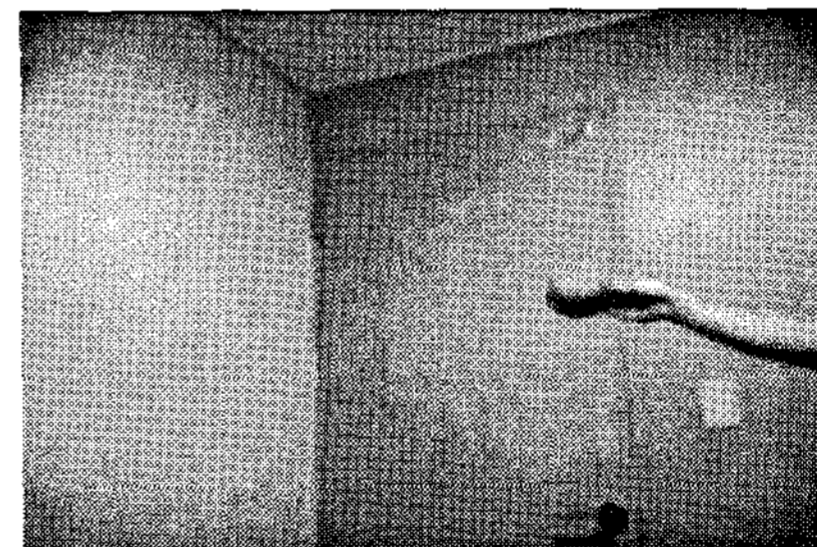


그림 3 객체 포함 이미지

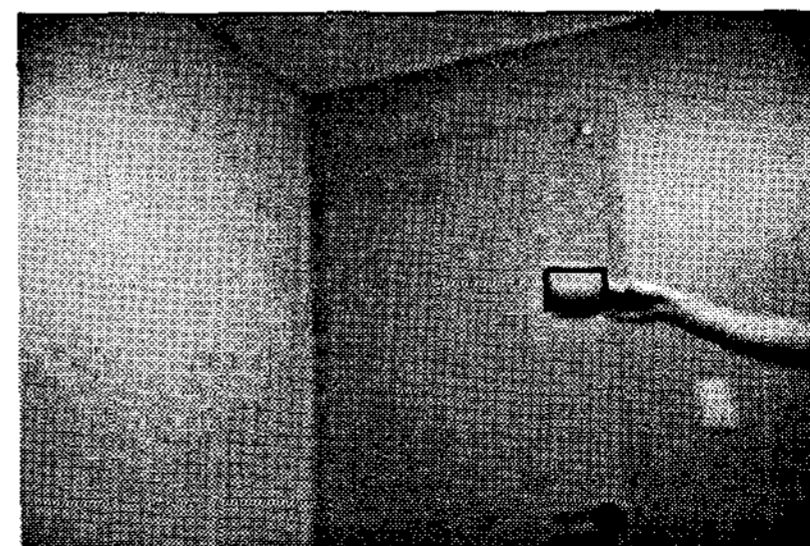


그림 4 객체 추출 이미지

표 3. 성능 비교

기 법	처리속도	검출율
윤곽선 기반 법	14~19	98%
정합법	11~14	95%
제안 기법	15~22	97%

IV. 결 론

본 논문에서는 CCD 카메라를 이용하여 입력된 이미지에서 특정 컬러를 갖는 객체를 검색, 추출하는 알고리즘을 제안하였다. 카메라의 설치 환경은 배경 영상에 변화가 거의 없는 실내이고, 카메라를 고정하여 이미지를 추출 하였다. 입력되는 이미지에 대하여 주위 조명 및 노이즈를 제거하기 위하여 전처리 과정을 거쳐 정규화 하였고, 배경 이미지를 설정하였다. 이후 입력 이미지와 배경 이미지에 대하여 블록 단위로 R,G,B 각 채널에 대하여 이미지 차이를 비교하여 객체를 검출한다. 미리 검출 하려고 하는 객체의 컬러 정보를 이용하여 객체를 검출 하도록 하였다. 실험 결과 기존의 윤곽선을 이용한 방법, 정합법에 비하여 처리 속도가 15% 향상 되었고 정합법 보다 안정적인 검출을 확인 할 수 있었다. 이는 보안 및 인식 시스템에서 불특정 물체의 접근 및 출현에 대하여 컬러 정보를 이용한 검출이 가능함을 보였다.

REFERENCES

[1] Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, Prentice-Hall, 2002.
 [2] Chang-Hyun Sun, Young-Tak Kim "Real-time tracking algorithm of a moving object using a CCD camera", Journal of The Frontier Technology Innovation Center, Vol.11 No.2, pp.33-44, 2002.
 [3] Nae-won Choi, Jeong-gyu Jee, "Real-time Object Tracking using Adaptive Background Image in Video", Journal of Korea Multimedia Society, Vol.6 No.3, pp.409-418, 2003.
 [4] Sang-Wook Lee, "A Moving Objects Tracking Using Contour Model and Feature", Journal of Institute Marine Industry, Vol.14 No.0 pp.42-51, 2001
 [5] Yong-Ho Lim, Joong-Hwan Baek, Soo-Chan Hwang, "Tracking of Moving Object using

Fuzzy Prediction", The journal of Korea Navigation Institute, Vol.5 No.1 pp.26-36, 2001

[6] Young-Bin Kim, Kwang-Ryul Ryu, Robert J. Sclabassi, "Changing Scene Detection using Histogram and Header Information of H.264 Video Stream", Proceedings of the Korean Institute of Maritime Information and Communication Sciences Conference, pp.197-200, 2006. 06
 [7] Akio Yoneyama, Yasuyuki Nakajima, Hiromasa Yanagihara, Masaru Sugano, "Moving Object Detection from MPEG Video Stream", System and Computers in Japan, Vol.30, No.13, 1999