Journal of Institute of Control, Robotics and Systems (2012) 18(1):35-41
http://dx.doi.org/10.5302/J.ICROS.2012.18.1.035 ISSN:1976-5622 eISSN:2233-4335

SURF 와 Label Cluster 를 이용한 이동형 카메라에서 동적물체 추출

Moving Object Detection Using SURF and Label Cluster Update in Active Camera

정 용 한, 박 은 수, 이 형 호, 왕 덕 창, 허 욱 열, 김 학 일*
(Yong-Han Jung[1], Eun-Soo Park[1], Hyung-Ho Lee[1], De-Chang Wang[1], Uk-Youl Huh[1], and Hak-Il Kim[1])
[1]Inha University

Abstract: This paper proposes a moving object detection algorithm for active camera system that can be applied to mobile robot and intelligent surveillance system. Most of moving object detection algorithms based on a stationary camera system. These algorithms used fixed surveillance system that does not consider the motion of the background or robot tracking system that track pre-learned object. Unlike the stationary camera system, the active camera system has a problem that is difficult to extract the moving object due to the error occurred by the movement of camera. In order to overcome this problem, the motion of the camera was compensated by using SURF and Pseudo Perspective model, and then the moving object is extracted efficiently using stochastic Label Cluster transport model. This method is possible to detect moving object because that minimizes effect of the background movement. Our approach proves robust and effective in terms of moving object detection in active camera system.

Keywords: moving object detection, active camera, SURF, pseudo perspective transforms

I. 서론

동적 물체 검출은 카메라에서 얻은 영상으로부터 배경을 제외하고 움직이는 물체를 추출하는 분야로 지능형 이동로봇, 감시 시스템, 지능형 자동차 시스템등과 같은 분야에서 다양하게 사용되고 있다.

연속된 두개의 영상을 비교하고 차이에 의해 움직이는 물체를 찾는 동적 물체 검출 연구는 대부분 카메라가 고정된 환경에서 이루어졌다. Wren [1]은 배경의 차를 이용하여 물체를 검출하였고, Stauffer [2]는 배경 변화를 여러 그룹으로 나누고 학습하여 사용하였다. Haritaoglu [3]는 사람의 행동학적 특성을 이용하였고, Cucchiara [4]는 HSV 색공간을 이용하여 배경과 물체를 구분하였으며, Leibe [5]는 물체를 미리 학습하여 유사한 물체를 추출하는 알고리즘을 사용하였다. 이러한 알고리즘들은 주로 도로 환경 모니터링[6], 고정형 감시 카메라 시스템[7] 등에서 사용되어 왔다. 그러나 이동형 로봇과 같이 카메라가 움직이는 환경에서는 움직이는 물체가 없는 상황에서도 간단한 카메라의 움직임에 의해 큰 차이가 발생하게 된다[8]. 따라서 단순히 프레임의 차이를 이용한 방법은 많은 에러를 발생시킨다.

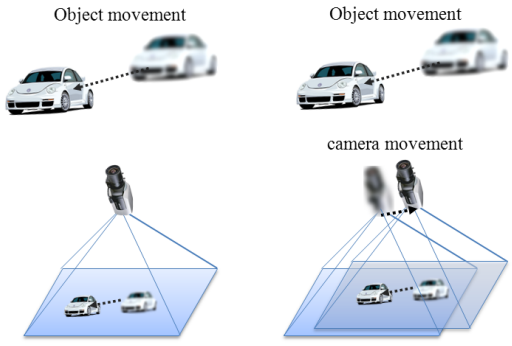
따라서 본 논문에서는 특징점의 이동으로 카메라의 이동을 추정하고 이를 보정하여 동적 물체를 추출하는 알고리즘을 제안한다. 여기서 특징점 추출은 SURF [10] 알고리즘을 사용하였고, 특징점 이동 모델은 Pseudo Perspective [11] 모델을 사용하였다. 또한 추출된 물체의 움직임은 Labeling과 Clustering을 이용하여 군집화 하였으며, 이러한 Label Cluster를 지속적으로 추출하여 다음 프레임에 사용함으로서 동적 물체의 움직임에 대한 신뢰도를 높였다.

* 책임저자(Corresponding Author)
논문접수 : 2010. 10. 19., 수정 : 2011. 4. 15., 채택확정 : 2011. 11. 15.
정용한, 박은수, 이형호, 왕덕창, 김학일: 인하대학교 정보통신공학부
(papuya@naver.com/espark@vision.inha.ac.kr/hylee@vision.inha.ac.kr/dcwang@vision.inha.ac.kr/hikim@inha.ac.kr)
허욱열: 인하대학교 전기공학과(uyhuh@inha.ac.kr)
※ 본 연구는 과학재단 특정기초 연구 지원 사업의 연구비 지원으로 연구 되었음.
Copyright© ICROS 2012

II. 본론

1. 동적 물체 추출 알고리즘의 전반적 구성

이동형 카메라 환경은 그림 1(b)에서 보이는 것처럼 기본적으로 카메라가 움직이는 환경으로 크게 두 개의 독립적인 움직임이 포함되어 있다. 하나는 카메라가 장착된 이동형 로봇의 움직임 또는 카메라 자체의 모터로 인한 카메라의 이동이고, 다른 하나는 카메라의 움직임에 독립적이며 물체가 가지는 성격에 따라 움직임이 변하는 물체의 이동이다. 이러한 독립적인 두 움직임이 하나의 영상에 포함되어있기 때문에 물체의 움직임을 추출하기 위해서는 카메라의 움직임을 모델링하고 이를 제거해야 한다.



(a) Stationary camera system (b) Active camera system

그림 1. 고정형 카메라 시스템과 이동형 카메라 시스템.
Fig. 1. Stationary camera system and Active camera system.

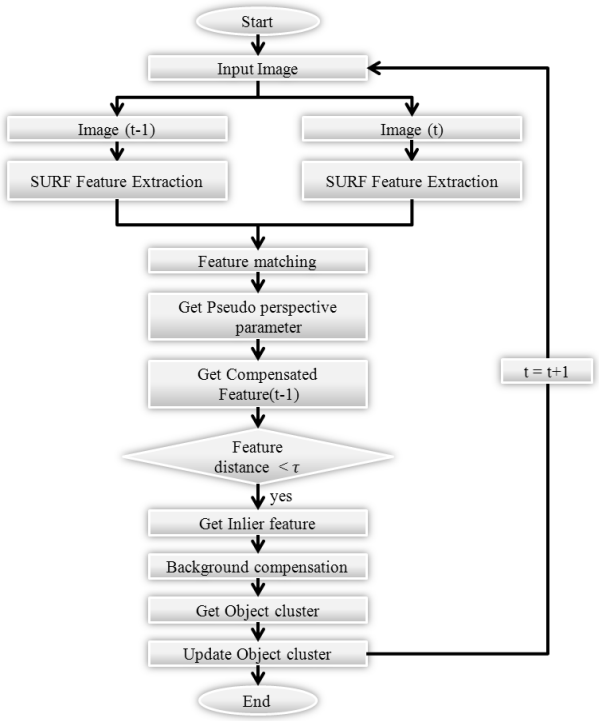


그림 2. 동적 물체 검출 알고리즘의 순서도.
Fig. 2. Moving object detection flow chart.

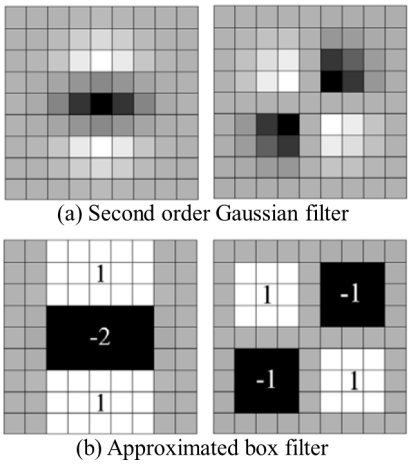
이동형 카메라 환경에서의 동적 물체 검출 과정은 연속된 영상에서 이전 프레임 영상을 현재 프레임 영상으로 변환시켜 배경 부분을 제거하고 남은 부분으로 동적 물체 영역을 추출하게 된다.

이전 프레임 영상을 현재 프레임 영상으로 변환하기 위해 각각의 영상 프레임에서 SURF [10] 특징점을 추출하고 대응점을 찾는다. 이렇게 찾아진 대응점을 Pseudo Perspective 모델 [11]을 사용하여 영상 변환 관계를 찾는다. 이렇게 찾아진 Pseudo Perspective Parameter를 이용하여 이전 프레임 영상을 현재 프레임 수준으로 변환하고 이 두 프레임의 차이에서 노이즈를 제거한다. 이후 Labeling과 Clustering을 이용하여 물체의 움직임 영역을 추출한다. 이렇게 생성된 물체의 움직임 영역을 이전 프레임에서의 결과로 업데이트 하여 그 중 연속성을 갖는 것을 동적 물체로 추정한다. 순서도는 위 그림 2와 같다.

2. SURF Feature Extraction

이동형 카메라 환경에서의 동적 물체 검출을 위해서 입력된 영상에서 동일한 특징을 갖는 대응점들을 찾아야한다. 이러한 대응점을 찾기 위해 특징점들은 영상의 크기 변화 및 회전에 상관없이 유사한 값을 가져야한다.

특징점 및 대응점을 찾는 대표적인 알고리즘으로 SIFT (Scale Invariant Feature Transform) [9]와 SURF (Speeded Up Robust Features) [10]가 있다. SIFT 알고리즘은 영상의 크기 및 회전 변화에 강인한 특징점을 추출하는 것으로 알려져 있지만 처리 시간이 많이 소요된다. 따라서 이러한 단점을 개선한 것이 SURF 알고리즘이다. SURF 알고리즘은 특징점을 찾기 위해 Hessian detector를 기반으로 적분영상과 Fast-Hessian detector를 사용하여 연산 속도를 높였다.



(a) Second order Gaussian filter
(b) Approximated box filter

그림 3. 2차 Hessian detector 필터 (a) 가우시안 미분 필터 (b) 근사 필터.
Fig. 3. Hessian detector filter (a) Gaussian second order partial derivative filter (b) Approximation box filter.

적분 영상 $II(x,y)$ 은 원점으로부터 각 픽셀의 위치까지의 사각형 영역의 모든 픽셀 값을 더한 것으로 해당 영역에 해당하는 픽셀의 합을 빠르게 구하게 해준다. 이는 식 (1)로 표현된다.

$$II(x,y)=\sum_{i=0}^{x}\sum_{j=0}^{y}I(i,j) \tag{1}$$

여기서 I 는 입력 영상이고 i,j 는 픽셀 좌표를 나타낸다.

Hessian detector는 다음 식 (2)와 같이 정의 된 행렬식에 기반을 둔 특징점 추출 알고리즘으로 속도와 정확도 면에서 안정적인 성능을 보인다.

$$H(x,y,\sigma)=\begin{bmatrix}L_{xx}(x,y,\sigma) & L_{xy}(x,y,\sigma)\\ L_{xy}(x,y,\sigma) & L_{yy}(x,y,\sigma)\end{bmatrix} \tag{2}$$

여기서 $L_{xx}(x,y,\sigma)$ 는 분산 σ 를 갖는 가우시안의 x 방향 2차 미분값을 말하고 $L_{xy}(x,y,\sigma)$ 과 $L_{xy}(x,y,\sigma)$ 는 각각 y 방향과 xy 방향의 2차 미분값을 의미한다.

Fast Hessian detector는 앞서 언급한 행렬식을 그림 3(b)와 같이 간략화 하여 특징점을 추출한다.

Fast Hessian detector를 이용하여 추출된 Interest Point들을 중심으로 이웃 픽셀과의 x 와 y 방향 Haar Wavelet Response d_x, d_y를 계산한다. 이렇게 구한 Response를 이용하여 특징벡터 $V(\sum d_x,\sum d_y,\sum|d_x|,\sum|d_y|)$ 를 만들어낸다. 이 중 가장 큰 값을 특징점의 방향으로 할당한다.

3. Pseudo Perspective Transform

SURF 특징점을 이용하여 각각의 연속된 영상 각각에서 특징점을 구하고 대응점을 찾으면 이를 이용하여 카메라의 움직임을 추정하게 된다. 카메라가 시간 $t-1$ 에서 시간 t 로 움직이게 될 때 이 때 얻어진 영상 I^{t-1} 와 I^{t} 은 서로 다른 좌표계에 존재 하게 된다. 따라서 이를 보정해주기 위해 Pseudo Perspective 모델을 사용하였다. 이는 식 (3)과 같이 표현된다.

$$\begin{bmatrix} wx' \\ wy' \\ w \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{3}$$

여기서 (x, y) 와 (x', y') 는 각각 영상 I^{t-1} 와 I^{t} 에서의 상응하는 특징점 좌표이다. 위의 식 (3)은 다음 식 (4)로 쓰여 질 수 있고 이는 다시 식 (5)로 변환된다.

$$x' = \frac{ax + by + c}{gx + hy + 1}, \quad y' = \frac{dx + ey + f}{gx + hy + 1} \tag{4}$$

$$\begin{aligned} x' &= ax + by + c - gx'x - hx'y \\ y' &= dx + ey + f - gy'x - hy'y \end{aligned} \tag{5}$$

식 (5)를 이용하여 Pseudo Perspective Parameter $\mathbf{P}$ 를 구하기 위해선 4쌍의 대응점만 있으면 된다. 그러나 정확한 카메라의 움직임을 추정하기 위해 더 많은 대응점들이 사용되기 때문에 이렇게 Over Constraint된 상태에서 에러가 적은 Parameter를 구하기 위해 Pseudo Inverse를 이용하여 fitting을 진행하게 된다. 이를 위해 식 (5)를 이용하여 n 개의 특징점의 대응 관계를 행렬로 만들면 다음 식 (6)과 같이 된다.

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x'_1 x_1 & -x'_1 y_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -y'_1 x_1 & -y'_1 y_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x'_2 x_2 & -x'_2 y_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -y'_2 x_2 & -y'_2 y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & 1 & 0 & 0 & 0 & -x'_n x_n & -x'_n y_n \\ 0 & 0 & 0 & x_n & y_n & 1 & -y'_n x_n & -y'_n y_n \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ \vdots \\ x'_n \\ y'_n \end{bmatrix} \tag{6}$$

식 (6)은 간단히 $\mathbf{AP} = \mathbf{x}$ 와 같은 형태로 나타낼 수 있다. 이를 식 (7)과 같이 변환하여 $\mathbf{P}$ 를 구한다.

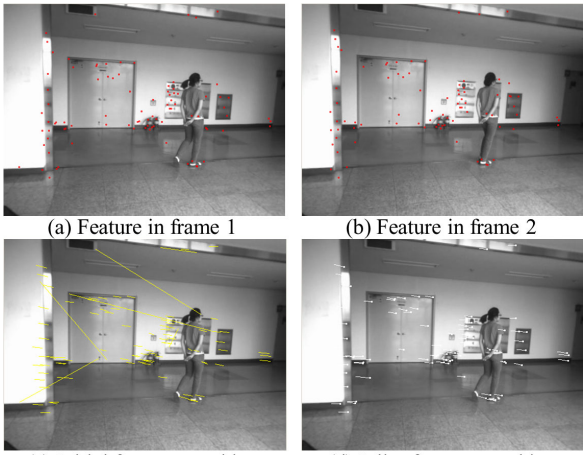
$$\begin{aligned} \mathbf{A}^T\mathbf{AP} &= \mathbf{A}^T\mathbf{x} \\ \mathbf{P} &= (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{x} \end{aligned} \tag{7}$$

4. Inlier Feature Extraction

앞의 과정을 통해 SURF 특징점을 구하고 이를 이용한 Pseudo Perspective Parameter $\mathbf{P}$ 를 구하였다. 그러나 앞의 과정에서 구한 $\mathbf{P}$ 는 카메라의 움직임뿐만 아니라 물체의 움직임도 포함하고 있기 때문에 $\mathbf{P}$ 를 보정해줄 필요가 있다. 이는 다음과 같은 절차에 따라 수행된다.

1) 이전 프레임에서의 SURF 특징점 $\mathbf{F}^{t-1}$ 과 현재 프레임에서의 SURF 특징점 $\mathbf{F}^{t}$ 의 대응점으로부터 최초 변환 $\mathbf{P}_0$ 와 대응점의 평균 거리 τ 계산
2) 변환 파라메터 $\mathbf{P}_0$ 를 이용하여 특징점 $\mathbf{F}^{t-1}$ 를 $\mathbf{F}'^{(t-1)}$ 으로 변환
3) 변환된 특징점 $\mathbf{F}'^{(t-1)}$ 과 $\mathbf{F}^{t}$ 의 거리를 비교하여 Inlier 대응점 $\mathbf{F}_{in}$ 추출

$$\begin{cases} \mathbf{F}_{in} & if \quad \left|\mathbf{P}\cdot\mathbf{F}(t-1) - \mathbf{F}(t)\right| < \tau \\ \mathbf{F}_{out} & otherwise \end{cases} \tag{8}$$



(a) Feature in frame 1 (b) Feature in frame 2

(c) Initial feature matching (d) Inlier feature matching

그림 4. (a),(b)영상에서 특징점, (c)초기 대응점, (d) Inlier 대응점.

Fig. 4. (a),(b) Feature in frame, (c) Initial feature matching, (d) Inlier feature matching.

4) 추출된 Inlier 대응점으로 $\mathbf{P}_{in}$ 를 계산
5) $\mathbf{P}_{in}$ 를 이용하여 2~4의 과정을 반복하여 최적화 된 Pseudo Perspective Parameter $\mathbf{P}_{opt}$ 를 계산

그림 4는 초기 대응점과 $\mathbf{P}_{opt}$ 에 의해 필터링된 Inlier 대응점의 결과의 비교를 보인다.

5. Background Compensation & Initial Object Motion Data Estimation

Inlier 대응점으로 계산한 Pseudo Perspective Parameter $\mathbf{P}_{opt}$ 는 물체의 이동으로 인하여 생긴 대응점 및 잘못 매칭된 대응점들을 제거하였기 때문에 카메라의 움직임을 반영한다고 할 수 있다. 따라서 $\mathbf{P}_{opt}$ 로 영상을 변환하여 카메라의 이동에 따른 변화를 최소화 하고 노이즈를 제거하여 동적 물체의 움직임 영역을 추출한다. 이 과정은 다음과 같은 절차로 수행된다.

1) $\mathbf{P}_{opt}$ 를 이용하여 이전 프레임 영상 변환

$$I_{comp} = \mathbf{P}_{opt} \cdot I^{t-1} \tag{9}$$

2) 변환된 영상 과 현재 프레임 영상의 차이를 계산

$$I_{diff} = \left|I_{comp} - I^{t}\right| \tag{10}$$

3) 차이값이 α 이하인 값 제거하여 I_{diff_cutoff} 생성
4) 중심 픽셀로부터 거리가 $\frac{w}{2}$ 이하인 주변 픽셀의 분포가 절반 이하 인값 제거

$$p_{sum} = \sum_{i,j=w} I_{diff_cutoff}(i,j)$$

$$I_{motion}(i,j)\begin{cases} 1 & if \quad p_{sum} > \dfrac{w^2}{2} \\ 0 & otherwise \end{cases} \tag{11}$$

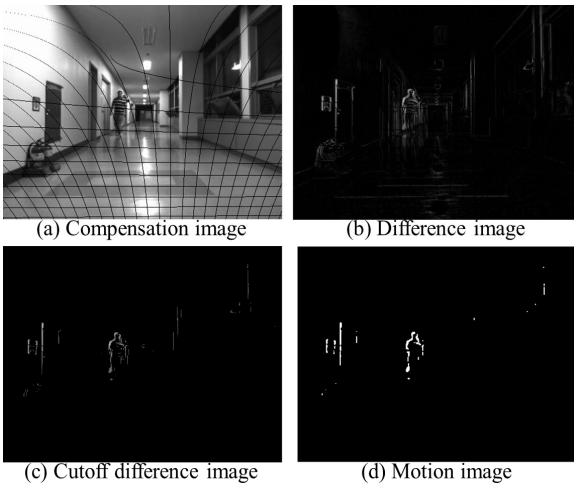

(a) Compensation image (b) Difference image
(c) Cutoff difference image (d) Motion image

그림 5. 초기 동적 물체 움직임 데이터 영상.
Fig. 5. Initial moving object motion data image.

여기서 α 값은 조명변화에 따른 에러 수치 보정값으로 실험적 결과에 의해 결정되었다.

이러한 과정의 중간 결과는 그림 5와 같다. 그림 5(a)는 I'^{-1} 를 $\mathbf{P}_{opt}$ 를 이용하여 변환한 결과 이며, 영상에서 보이는 선은 I'^{-1} 의 픽셀이 Forward Warping 되어 I_{comp} 로 변환 될 때, 기존 영상에 비해 부분적으로 늘어났기 때문에 영상에 Hole 이 생긴 것이다. 그림 5(b)는 변환된 I_{comp} 와 I^t 의 차 영상이고, 그림 5(c)는 I_{diff} 중 차이값이 α 이하인 I_{diff_cutoff} 영상이며, 그림 5(d)는 I_{diff_cutoff} 에서 주변 픽셀의 확률을 이용하여 필터링한 I_{motion} 영상이다.

6. Label Clustering and Update

동적 물체의 움직임 영역에서 밝은 부분은 동적 물체의 동작일 가능성이 높은 곳이다. 그림 5(d)에서 보이는 것처럼 이러한 부분은 물체의 외곽부분에서 두드러지게 된다. 따라서 움직임 영역을 Labeling하여 각각의 Label에서 가까운 거리에 있는 Label들을 Clustering하여 표현 할 필요가 있다.

이전의 과정에서 카메라의 이동에 의한 배경의 변화가 대부분 제거 되었으나 Pseudo Perspective 모델이 거리 값을 반영하지 못하여 물체와의 거리에 따른 변화에 완벽히 대응하지 못한다. 즉, 복도와 같이 배경이 다양한 거리에 존재하게 되는 환경에서는 카메라의 이동에 따른 배경의 이동량을 정확히 추정하지 못하기 때문에 에러가 발생하게 된다. 따라서 앞의 과정에서 계산된 동적물체 움직임 영역의 결과만을 가지고 물체의 영역을 판별하기에는 무리가 있기 때문에 연속되는 영상에서 물체의 이동을 업데이트 하여 가장 확률이 큰 값을 동적 물체라고 판별하게 된다.

1) 동적 물체 움직임 영역을 Labeling하여 n개의 Label $L(n)$ 생성
2) 각각의 Label의 크기에 따른 거리값 δ 를 계산한 후 Label의 중심 값 주변의 Label과의 거리가 δ 값 이하인 Label들을 Clustering하여 m개의 Label Cluster $C(m)$ 생성

$$\begin{cases} L(j) \in C(i) & if \quad \left| L_{distance}(i) - L_{distance}(j) \right| < \delta \\ L(j) \notin C(i) & otherwise \end{cases} \tag{12}$$

3) 가장 큰 Label Cluster를 동적 물체 영역이라 가정

$$C_{obj} = \max\left\{ C_{size}(1), C_{size}(2), \cdots, C_{size}(m) \right\} \tag{13}$$

4) 다음 프레임에서의 Label Cluster에 이전 프레임에서의 물체 영역과의 거리와 크기에 따른 가중치 ω 를 곱하여 가장 유사한 Label Cluster를 동적 물체인 것으로 가정하고 업데이트

$$\begin{aligned} C^t_{obj} &= \min\left\{ \omega_1 C^t(1), \omega_2 C^t(2), \cdots, \omega_m C^t(m) \right\} \\ C^{t-1}_{obj} &\approx C^t_{obj} \end{aligned} \tag{14}$$

여기서 ω 는 이전에 구한 물체의 영역 C^{t-1}_{obj} 과 다음 프레임에서의 Label Cluster 의 거리차와 크기 차이의 절대값을 곱한 값으로 다음 식 (15)와 같이 표현된다.

$$\omega_m = \left| C^{t-1}_{obj,\ distance} - C^t_{distance}(m) \right| \times \left| C^{t-1}_{obj,\ size} - C^t_{size}(m) \right| + 1 \tag{15}$$

그림 6은 연속된 두 프레임에서의 움직임 영역 그림 6(a)과 가장 큰 Label Cluster 영역 그림 6(b) 그리고 이전 프레임에서 추출된 물체 영역을 이용하여 업데이트한 결과 그림 6(c)를 보인다. 두 번째 프레임 결과를 보면 가장 큰 영역은 물체가 아닌 배경에 맞추어져 있다. 그러나 이전 프레임에서 얻은 물체의 정보를 이용하기 때문에 이러한 에러는 제거 되게 된다.

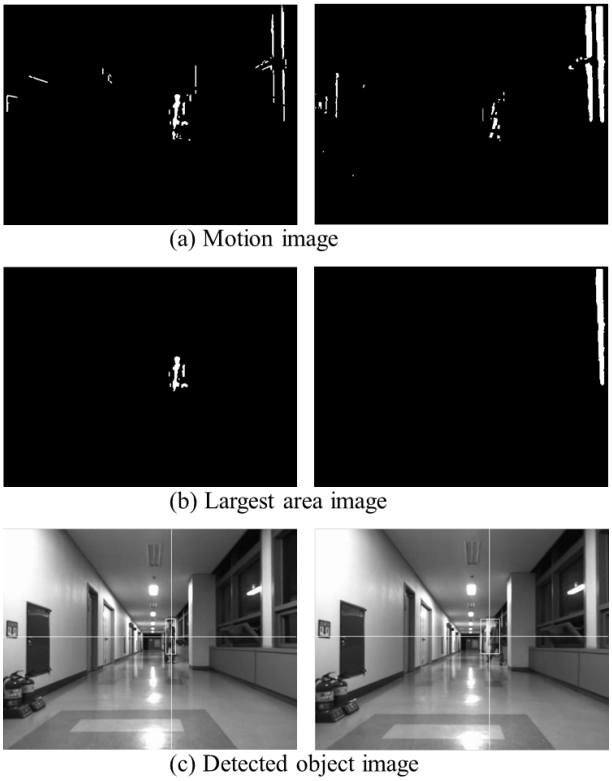

(a) Motion image
(b) Largest area image
(c) Detected object image

그림 6. Label Cluster의 가중치를 이용하여 동적물체 추출.
Fig. 6. Moving object detection using weight of Label Cluster.

III. 실험 결과

본 절에서는 앞서 언급한 이동형 카메라시스템에서의 동적 물체 추출결과를 다양한 데이터에서 실험하여 보고 그 결과를 비교하여 본다. 실험에 사용된 환경은 다음과 같다.

실험은 512x384 크기의 영상을 가지고 진행하였으며 제안한 알고리즘을 고정형 카메라 시스템에서 많이 사용되는 Background Subtraction [12]의 프레임 차분영상과 CAM-Shift [13]의 결과 와 각각 비교 하였다.

Background Subtraction은 동적 물체 검출에 많이 사용되는 알고리즘으로서 카메라가 연속으로 획득한 영상을 분석하여 배경을 분리해내고 물체의 이동만을 추출하는 알고리즘이다. 따라서 제안한 알고리즘과 직접적인 비교는 힘드나 Background Subtraction 또한 프레임의 차이를 이용하므로 제안한 알고리즘이 Pseudo Perspective 모델을 이용하여 보정한 프레임 차이값과 비교하였다. 그림 7은 이러한 두 모델의 비교를 보인다. 고정형 카메라 시스템을 바탕으로 한 Background Subtraction에서 사용되는 Frame Difference 그림 7(a)에 비해 제안한 알고리즘에서 사용하는 Compensated Frame Difference의 결과 그림 7(b)에서 에러가 적음을 알 수 있다.

CAM-Shift는 대표적인 물체 추적 알고리즘으로서 많은 연구에서 이를 활용하고 있다. CAM-Shift는 초기에 추적해야 할 대상을 선정해주어야 하므로 초기 값 없이 이동 물체를 추출하는 제안 알고리즘과 비교를 위해 초기 값으로 이동물체의 위치를 선정해주었다. 그림 8은 CAM-Shift와 제안 알고리즘의 결과를 보여준다.

CAM-Shift는 대상 영역의 색상을 분석하여 비슷한 영역을

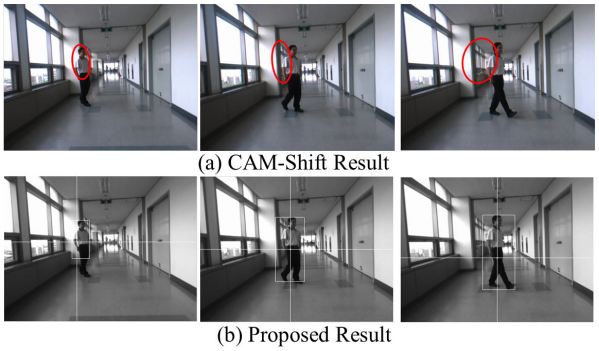
표 1. 실험 환경.
Table 1. Experimental system.

CPU	Intel Core 2 Quad CPU 6600@ 2.40GHz
Memory	4 GB
OS	Window XP
Tools	Visual Studio 2008
Camera	Bumblebee XB3

(a) Frame difference

(b) Compensated frame difference

그림 7. 이동형 카메라 시스템에서의 보정 전후의 프레임 차이 비교.
Fig. 7. Comparison of frame difference in active camera system.



(a) CAM-Shift Result

(b) Proposed Result

그림 8. CAM-Shift와 제안알고리즘의 결과비교.
Fig. 8. CAM-Shift and Proposed algorithm result.

표 2. 샘플 동영상에서의 동적물체 추출결과.
Table 2. Moving object detection result in sample movies.

Name	Total Frame	Detect	Detect Pos.	Detect Neg.	Miss	Detect Rate
Movie 1	25	19	18	1	6	72
Movie 2	36	33	31	2	3	86.11
Movie 3	32	32	27	5	1	84.38
Movie 4	81	56	50	6	25	61.73
Movie 5	33	24	21	3	11	63.64
Movie 6	56	40	32	8	16	57.14
Movie 7	55	47	45	2	6	81.82
Movie 8	29	26	26	0	3	89.65
Average Detection Rate : 74.56%						

추적하는 알고리즘으로 배경과 대상의 색상이 비슷한 경우 그림 8(a)에서 보이는 것처럼 추적에 실패하게 된다. 그러나 제안된 알고리즘은 물체의 움직임을 바탕으로 검출하기 때문에 지속적인 움직임이 있는 상황에서는 그림 8(b)처럼 지속적으로 대상을 검출하게 된다.

위의 표 2는 다양한 샘플 동영상에서의 제안 알고리즘의 동적 물체 추적 결과를 보인다. Total Frame은 영상 전체 프레임수이며, Detect는 검출된 프레임수, Detect Pos.는 동적 물체 검출에 성공한 경우 이며, Detect Neg.는 검출은 하였으나 물체가 아닌 배경을 검출한 경우이다. 또한 Miss 는 물체의 움직임이 있으나 아무것도 검출되지 않은 프레임수이며, Detect Rate는 전체 프레임수에서 동적 물체 검출에 성공한 경우의 % 값이다.

표 2의 결과는 대부분 우수한 성능을 보이나 몇몇 영상의 결과는 좋지 못하다. 4번 영상의 경우 중간에 동적 물체가 멈추어 움직임을 검출하지 못한 부분이 많았고, 5번과 6번의 경우는 느리게 움직여서 낮은 검출율을 보였다. 그림 9는 1, 3, 4, 8번 영상에서 추출한 동적물체 검출 결과를 보인다.

IV. 결론

본 논문에서는 카메라가 움직이는 환경에서 동적 물체 추출 알고리즘을 제안하였다. 카메라의 움직임을 제거하기 위해 연속된 프레임에서 SURF 특징점을 추출하고 Pseudo Perspective 모델을 이용하여 Feature Fitting 하였으며, 중심 픽셀 주변의 픽셀 확률을 이용하여 노이즈를 제거하였고, 이후 움직임 영역에서 Labeling과 Clustering을 이용하여 물체의 움직임 영역을 Clustering하고 이전 프레임에서의 결과로 업데

(a) Input frame (b) Feature fitting (c) Motion image (d) Detection result

그림 9. 동적 물체 검출 결과.

Fig. 9. Moving object detection result.

이트 하였다. 그 결과 그림 7처럼 배경의 움직임으로 인해 생기는 오차를 최소화 했으며, 그림 8의 결과처럼 지속적인 움직임이 있는 물체의 검출 및 추적에도 우수함을 보였다.

일반적으로 선행되어온 이동로봇과 같은 동적 환경에서의 객체 추출 및 추적[14,15]은 물체의 대상을 공과 같은 미리 학습되어 있는 대상에 국한되어 진행되어 왔을 뿐 아니라 배경의 변화가 적은 실험적 환경에서만 수행하였다. 그러나 본 논문에서는 미리 학습된 객체가 아닌 일반적인 동적 객체로 확대하여 검출 · 추적 하였고 배경 변화에 적응적으로 대처하여 수행함을 보였다.

그러나 본 논문의 실험에서는 하나의 대상을 고려하였기 때문에 다중 물체의 움직임 경우 가장 큰 움직임을 나타내는 물체를 추적하거나, 다중의 물체를 커다란 하나의 물체로 검출하는 결과를 보였다. 또한 논문에서 이전 프레임 보상에 사용한 Pseudo Perspective 모델의 경우 물체에 해당하는 거리 값을 고려하지 않는다. 이로 인하여 복잡한 배경에서 에러가 존재하게 된다. 또한, 이동물체의 움직임이 적거나 없는 경우 물체 검출 및 추적에 실패하게 되는 문제가 있다.

향후 이러한 문제점을 개선하여 다중 물체에서의 검출 알고리즘으로 확장하고, 물체와의 거리를 감안한 Perspective 모델을 사용함으로서 복잡한 배경에서 나오는 에러를 줄이고, 물체의 움직임이 적은 상황에서도 추적 및 검출이 가능하기 위해 Mean-shift 와 같은 추적 알고리즘과의 융합을 한다면 좀 더 나은 성능을 가지게 될 것이라 예상한다.

참고문헌

[1] C. Wren, A. Azarbayejani, T. Darrel, and A. Pentland, “PFinder, real time tracking of the human body,” *IEEE PAMI*, vol. 19, no. 7, Jul. 1997.

[2] C. Stauffer and Grimson, “Learning patterns of activity using real-time tracking,” *IEEE PAMI*, vol. 22, no. 8, pp. 747-767, Aug. 2000.

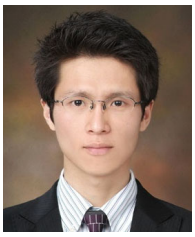
[3] I. Haritaoglu, D. Harwood, and L. Davis, “W4: Real time surveillance of people and their activities,” *IEEE PAMI*, vol. 22, no. 8, pp. 809-830, Aug. 2000.

[4] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, “Improving shadow suppression in moving object detection with HSV color information,” *Intelligent Transportation Systems*, pp. 334-339, Aug. 2001.

[5] B. Leibe, A. Leonardis, and B. Schiele, “Robust object detection with interleaved categorization and segmentation,” *International Journal of Computer vision*, vol. 77, pp. 259-289, May 2008.

[6] J. Zhou, D. Gao, and D. Zhang, “Moving vehicle detection for automatic traffic monitoring,” *IEEE Transaction on Vehicular*

Technology, vol. 56, no. 1, pp. 51-59, Jan. 2007.
[7] O. Javed and M. Shah, "Tracking and object classification for automated surveillance," *Lecture Notes in Computer Science,* vol. 2353, pp. 439-443, 2006.
[8] B. Jung and G. Sukhatme, "Detecting moving object using a single camera on a mobile robot in an outdoor environment," *Conference on Intelligent Autonomous Systems*, pp. 980-987, Mar. 2004.
[9] D. G. Lowe, "Distinctive image features from scale invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.
[10] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," Lecture Notes in Computer Science, vol. 3951, pp. 404-417, 2006.
[11] E. Trucco and A. Verri, *Introductory Techniques for 3-D Computer Vision, Prentice Hall*, 1998.
[12] T. Horprasert, D. Harwood, and L. S. Davis, "A robust background subtraction and shadow detection," *Proc. of Asian Conference on Computer Vision*, pp. 983-988, Jan. 2000.
[13] G. R. Bradski, "Computer vision face tracking for use in a perceptual user interface," *Proc. of Intel Technology Journal*, vol. 2, no.2, pp. 12-21, 1998.
[14] S. J. Kim, J. W. Park, and J. M. Lee, "Implementation of tracking and capturing a moving object using a mobile robot," *International Journal of Control, Automation, and Systems*, vol. 3, no. 3, pp. 444-452, 2005.
[15] I. S. Kim, H. S. Choi, K. M. Yi, J. Y. Choi, and S. G. Kong, "Intelligent visual surveillance - A survey," *International Journal of Control, Automation, and Systems*, vol. 8, no.5, pp.926-939, 2010.



정 용 한

2008년 인하대학교 정보통신 공학부(공학사). 2011년 인하대학교 정보 공학과(공학석사). 2011년~현재 LG전자 재직중. 관심분야는 네비게이션, 스테레오 비젼, 위치 인식.

이 형 호

2010년 인하대학교 정보통신 공학부(공학사). 2010년~현재 인하대학교 정보 공학과 석사 과정. 관심분야는 패턴인식, 물체 추적, 로봇 비전, 병렬 영상 처리.

허 욱 열

1974년 서울대학교 전기공학과(공학사). 1978년 서울대학교 전기공학과(공학석사). 1982년 서울대학교 전기공학과(공학박사). 1980년~현재 인하대학교 전기공학과 교수. 2004년~2006년 한국 과학재단 전기정보 전문위원. 1997년~현재 대한전기학회 부회장. 2007년~2009년 제어로봇시스템학회 부회장. 관심분야는 지능 제어 시스템, 인공지능, 모션제어, 자율 주행로봇.

박 은 수

2007년 인하대학교 정보통신 공학부(공학사). 2010년 인하대학교 정보 공학과(공학석사). 2010년~현재 인하대학교 정보 공학과 박사 과정. 관심분야는 병렬 영상처리, 패턴인식, 로봇 비전.

왕 덕 창

2006년 YanCheng IT.(공학사). 2010년~현재 인하대학교 로봇 공학과 석사 과정. 관심분야는 패턴인식, 기계 학습, 로봇 비전.

김 학 일

1983년 서울대학교 제어계측공학과(공학사). 1985년 Purdue Univ. 전기/컴퓨터 공학과(공학석사). 1990년 Purdue Univ. 전기/컴퓨터 공학과(공학박사). 1990년~2001년 인하대학교 자동화 공학과 조교수. 2001년~현재 인하대학교 정보통신 공학과 교수. 2002년~현재 한국정보보호학회 생체인증연구회 위원장. 2003년~현재 ISO/IEC JTC1/SC37 WG5 Rapporteur. 2005년~현재 ITU-T SG17 Q.8 (Telebiometrics) Rapporteur. 관심분야는 패턴인식, 컴퓨터비전, 바이오 인식, 자율주행로봇, 센서 및 계측 공학.