

## CCTV 개인영상 정보보호를 위한 오프라인 객체추적

이석호\*

### Offline Object Tracking for Private Information Masking in CCTV Data

Suk-Ho Lee\*

Division of Computer & Information Engineering, Dongseo University, Busan 617-716, Korea

#### 요 약

최근 CCTV의 영상정보 공개시 타인의 영상에 대해서는 모자이크 처리하여 식별이 불가능하도록 하는 개인정보 보호법이 발표되었다. 반면에 CCTV를 통하여 획득된 영상 데이터를 활용한 범죄 수사는 점차 더 많이 활용되고 있는 추세이다. 이렇게 상반된 2가지의 요구사항에 의해 촬영된 CCTV 동영상을 수사기관 등의 기관에서 요청할 경우 개인정보(개인 얼굴 이미지 등)를 마스킹(Masking)해 배포함으로써 개인정보 유출을 차단하는 마스킹/언마스킹 솔루션에 대한 시장의 요구가 크게 성장할 것으로 예측된다. 정보주체가 정보열람 의뢰기관에 정보를 요구하기 전에 객체에 대한 선별 마스킹을 하기 위한 정보보호 솔루션에 있어서 가장 핵심되는 기술은 객체에 대한 추적기술이다. 본 논문에서는 CCTV영상에 대한 후처리로서의 정보보호 솔루션에 적합한 객체추적 알고리즘을 제안한다. 안정적인 추적이 가능하도록 하기 위해 움직임정보와 색상정보를 함께 사용하였다. 그러면서도 시간이 많이 걸리지 않는 컬러 중심점 이동(Color Centroid Shifting) 기반의 방법을 사용하여 추적의 속도 성능을 높였다.

#### ABSTRACT

Nowadays, a private protection act has come into effect which demands for the protection of personal image information obtained by the CCTV. According to this act, the object out of interest has to be mosaicked such that it can not be identified before the image is sent to the investigation office. Meanwhile, the demand for digital videos obtained by CCTV is also increasing for digital forensic. Therefore, due to the two conflicting demands, the demand for a solution which can automatically mask an object in the CCTV video is increasing and related IT industry is expected to grow. The core technology in developing a target masking solution is the object tracking technique. In this paper, we propose an object tracking technique which suits for the application of CCTV video object masking as a postprocess. The proposed method simultaneously uses the motion and the color information to produce a stable tracking result. Furthermore, the proposed method is based on the centroid shifting method, which is a fast color based tracking method, and thus the overall tracking becomes fast.

**키워드** : CCTV, 정보보호, 객체추적, 정보 마스킹 솔루션

**Key word** : CCTV, Information Security, Object Tracking, Information Masking Solution

접수일자 : 2014. 09. 11 심사완료일자 : 2014. 10. 20 게재확정일자 : 2014. 11. 10

\* **Corresponding Author** Suk-Ho Lee(E-mail:petrasuk@gmail.com, Tel:+82-51-320-1744)

Division of Computer & Information Engineering, Dongseo University, Busan 617-716, Korea

**Open Access** <http://dx.doi.org/10.6109/jkiice.2014.18.12.2961>

print ISSN: 2234-4772 online ISSN: 2288-4165

©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.  
Copyright © The Korea Institute of Information and Communication Engineering.

## I. 서 론

최근에 CCTV의 영상정보를 정보기관에 제공할 시 관심대상이 아닌 타인에 대해서는 모자이크 처리하여 식별이 불가능한 상태로 최소한의 정보만 제공할 수 있도록 개인정보보호법이 발표되었다. 현재까지는 권고 사항이지만 내년에는 개인정보를 유출할 경우 법적 책임을 지게 된다. 반면에 CCTV를 통해 획득된 영상 데이터들을 범죄 수사에 보다 많이 활용하고 있는 추세이다. 예를 들어 군포 여대생 실종사건, 일산 어린이 납치 미수 사건, 연쇄살인범 강호순 사건 등에서 CCTV데이터가 수사에 활용되었다. 그러므로 각종 CCTV영상 저장매체 또는 인터넷 상에 남아 있는 각종 디지털 정보를 분석해 범죄 단서를 찾는 수사기법의 일종인 디지털포렌식(digital forensic)을 지원하는 솔루션이 계속 개발되고 있는 추세이다.

상기의 상반된 2가지의 요구사항에 따라 촬영된 CCTV 동영상을 수사기관 등의 기관에서 요청할 경우 개인정보(개인 얼굴 이미지)를 마스킹(Masking)해 배포함으로써 개인정보 유출을 차단하는 마스킹/ 언마스킹의 솔루션에 대한 시장이 크게 성장할 것으로 예측이 된다. 이에 따라 정보주체의 영상정보 열람권을 보장하고, 정보주체 이외의 얼굴영상에 대해 선별적인 영상보호를 하는 솔루션의 개발이 필요하다. 정보주체가 정보 열람 의뢰기관에 정보를 요구하기 전에 객체에 대한 선별 마스킹을 하기 위한 정보보호 솔루션에서 가장 핵심되는 기술은 객체에 대한 추적기술이다. 각 프레임마다 수동적으로 마스킹 작업을 할 수 없을 뿐더러 추적기술의 추적성능이 떨어지면 수작업이 많아지기 때문이다.

객체추적에 있어서 조명의 변화, 겹쳐짐(occlusion), 객체의 모양의 변화 등의 많은 문제가 있기 때문에 이를 해결하기 위해 다양한 이론에 근거한 많은 추적알고리즘들이 제안되었다[1-8]. 이중 추적대상의 색상만을 기반으로 한 추적 알고리즘들은 비교적 계산량이 적은 반면 추적의 성능이 불안정할 때가 많다[1-5]. 반면에, 파티클 필터 프레임워크(particle filter framework)에 넣어서 추적알고리즘을 돌리게 되면 안정성이 올라가는 대신에 계산량이 증폭된다[6-8]. 그러나 CCTV로 취득된 영상에서의 마스킹을 위한 객체추적은 몇 가지 상황이 제약적이기 때문에 굳이 불필요하게 복잡한 알고리즘을 사용하지 않아도 되는 특징이 있다.

가장 큰 제약사항은 오프라인 추적이라는 것이다. 이 때문에 사용자의 인터랙션이 가능하다. 또한 CCTV 영상의 특성상 추적대상에 대한 조명의 갑작스러운 변화도 없고, 매우 짧은 시간구간의 추적이 많다는 것이다. 본 논문에서는 이러한 제약사항을 염두에 두고 충분히 안정적이면서도 계산량이 적은 CCTV영상정보 마스킹에 적합한 추적 알고리즘을 제안한다.

## II. 본 론

대부분의 개인정보 마스킹은 CCTV를 통해 취득된 영상이 통합관제센터의 저장매체에 저장된 후에 처리되기 때문에 마스킹을 위한 추적 알고리즘은 실시간으로 동작해야 한다는 제약조건을 만족하지 않아도 된다는 점에서 대부분의 추적 알고리즘들과 차별된다. 그러나 실시간까지는 아니라도 빠른 마스킹처리를 위해 여전히 빠른 추적이 가능해야 한다. 또한 오프라인에서 이루어지기 때문에 사용자의 인터랙션이 가능하다. 예를 들어 초기의 추적영역의 범위를 사용자가 마우스로 지정하는 것이 가능하다. 그러나 CCTV영상자료가 많기 때문에 수작업이 많이 들어가지 않고 가능한 사용자의 개입을 최소화할 수 있도록 개발하는 것이 필요하다. 또다른 제약사항은 추적대상을 현재의 CCTV화면에서 마스킹할 수 있을 정도로만 추적하면 된다는 점이다. 이 때문에 추적하는 시간 등이 매우 짧고, 화면에서 사라졌다가 나타나는 상황 등은 고려하지 않아도 된다. 이 장에서는 이 같은 조건들을 염두에 두고 CCTV영상 마스킹을 위한 추적 알고리즘을 제안한다. 객체의 위치 정보를 결정하기 위해서는 움직임 정보를 이용한 컬러 중심이동 기반 추적 알고리즘을 사용하고, 크기정보를 결정하기 위해서는 움직임영역 보정 배경모델링을 통해 구한 전경 영역을 사용한다.

### 2.1. 움직임정보를 이용한 컬러중심이동 기반 추적알고리즘

객체의 위치정보를 결정하기 위해서 본 논문에서는 본인이 [3]에서 제안한 컬러중심이동 기반 추적 알고리즘에 움직임영역 보정 배경모델링의 결과로 나오는 전경 정보( $F^m$ )의 이용을 접목한 다음과 같은 거리함수를 제안한다:

$$\sum_{u=1}^m q_u F^m \|\hat{y} - C_u^m\|^2 \quad (1)$$

여기서  $q_u$ 는 초기 프레임에서 컬러빈  $u$ 의 히스토그램이며,  $\hat{y}$ 는 추적대상의 현재위치,  $m$ 은 컬러빈(color bin)의 갯수를 나타낸다.  $F^m$ 은  $n$ 번째 프레임의 전경함수를 나타내며 특정위치에서 전경이면 1, 배경이면 0이라는 값을 가진다. 전경함수  $F^m$ 을 계산하는 방법은 다음 장에서 설명한다.  $C_u^m$ 은 컬러중심점을 나타내며 다음과 같이 계산된다:

$$C_u^m = \frac{\sum_{i=1}^{n_s} x_i \delta[b(x_i) - u]}{\sum_{i=1}^{n_s} \delta[b(x_i) - u]} \quad (2)$$

여기서  $b(\cdot)$ 는 현재 프레임의 화소  $x_i$ 의 컬러 값을 양자화된 컬러빈(bin)으로 매핑 시켜주는 함수이고,  $n_s$ 는 영역의 범위를 각각 나타낸다. 그리고  $\delta[\cdot]$ 은  $[\cdot]$ 의 내용이 0이면 1을, 0이 아니면 0을 반환하는 함수이다. 즉, 위의 식은 일정 영역에서 컬러가  $u$ 인 화소들의 위치의 평균을 계산한다.

식 (1)의 거리함수를 벡터  $\hat{y}$ 에 대해 미분하면,

$$\hat{y} = \frac{\sum_{u=1}^m q_u \delta[p_u^n] F_u^n C_u^m}{\sum_{u=1}^m q_u \delta[p_u^n] F_u^n} \quad (3)$$

으로 현재의 위치정보  $\hat{y}$ 을 구할 수 있다. CCTV영상 데이터 중 모자이크 처리하고자 하는 관심대상에 대해 초기 프레임에서 마우스로 관심범위를 설정하고 나면 이 초기 프레임에서  $q_u$ 를 계산하게 된다. 이후부터는 추적 알고리즘이 이전 중심위치를 중심으로 추적대상보다 약간 크게 검색 윈도우를 설정하여 검색영역 안에서 현재 프레임의 컬러들에 대하여 식 (2)를 이용하여 각 컬러들의 중심을 구한다.

초기 프레임에서 계산한 가중치를 이용하여 현재 프레임의 컬러 중심들의 가중평균인  $\hat{y}_{current}$ 와 다음 프레임의 컬러중심들의 가중평균인  $\hat{y}_{next}$ 을 구하고 두 위치벡터의 차이  $\hat{y}_{shift} = \hat{y}_{next} - \hat{y}_{current}$ 에 의해 객체가 움직인 벡터를 계산한다.

컬러 중심 이동 기반 추적 알고리즘을 개선한 알고리즘을 사용하는 이유는 컬러 중심 이동 기반 추적 알고리즘이 컬러기반의 추적 알고리즘들 중에서도 계산량이 매우 적은 알고리즘이면서도 컬러기반의 추적 알고리즘 중에서 유명한 평균이동(meanshift) 기반의 알고리즘들보다 안정적이기 때문이다. 무엇보다 객체의 부분 겹쳐짐(partial occlusion)문제도 어느 정도 해결할 수 있다. 그러나 단순 컬러중심 이동 기반 추적 알고리즘을 사용할 경우, 추적대상의 주변 컬러에 의해 추적대상의 위치 결정에 오류가 많이 나기 때문에 움직임 정보를 이용한 개선된 모델을 사용하게 된다. 추적대상에 대한 크기정보는 2.3절에서 제안한 방법으로 추출하기 때문에 움직임 정보를 이용한 컬러 중심 이동 기반 추적 알고리즘에서는 위치정보를 정확하게 추출하는 것만 집중하면 된다.

## 2.2. 움직임정보를 이용한 전경영역 추출

전경영역  $F^m$ 을 계산하기 위해 배경을 모델링하여 전경과 배경영역을 구분하여야 한다. 배경모델링에 MOG (Mixture of Gaussian)등을 이용하는 방법들 [9-12]도 있으나 속도가 느리고 많은 프레임을 저장하여야 하기 때문에 본 논문에서는 이동평균(running average) 배경모델링을 수정한 움직임 영역 이동평균 방법을 제안한다. 일반적인 이동평균 방법은 다음과 같은 식을 통해 현재 프레임의 배경을 모델링한다:

$$B^n = (1 - \alpha)B^{n-1} + \alpha I^n \quad (4)$$

여기서  $B^n$ 과  $B^{n-1}$ 은 각각 현재와 이전의 배경영상을 나타내며,  $I^n$ 은 현재 프레임이며,  $\alpha$ 는  $0 < \alpha < 1$ 의 범위내에 있는 상수로 현재의 배경영상  $B^n$ 을 만들 때 이전의 배경영상  $B^{n-1}$ 과 현재 프레임  $I^n$ 을 어느 정도의 비율로 가중 평균할 것인지를 결정해준다.



**그림 1.** 움직임정보를 이용한 배경모델링 (a) 입력 프레임 (b) 단순 이동평균에 의한 배경모델링에 의한 배경영상 (c) 차영상 (d) 차영상에 대한 팽창연산 (e) 움직임정보를 이용한 배경모델링에 의한 배경영상 (f) (e)를 사용한 전경영역 추출

**Fig. 1** Running average using motion information (a) input frame (b) background image obtained by original running average (c) difference frame (d) dilated difference frame (e) background image obtained by motion information using running average (f) foreground region using (e) as the background image

보통  $\alpha$ 의 값을 작게 설정하여 현재 프레임의 전경영역이 배경영상에 많이 반영되지 않게 한다.

그러나  $\alpha$ 의 값을 작게 잡아도 현재 프레임의 전경영역과 배경영상의 밝기값의 차가 크면 전경영역의 잔상이 배경영상에 많이 남는다(그림 1(b) 참조). 이 잔상으로 인해 앞장의  $F^n$ 의 계산에 어려가 발생하고, 이로 인해 추적의 안정성이 영향을 받으므로 본 논문에서는 식 (4)의 이동평균방법을 수정한 다음과 같은 움직임영역 보정 이동평균 모델링 방법을 제안한다:

$$B^n = (1 - \alpha(x))B^{n-1} + \alpha(x)I^n \quad (5)$$

이때  $\alpha(x)$ 은 공간 적응적인 함수로 픽셀의 위치  $x$

에 따라 다음과 같은 값을 갖는다.

$$\alpha(x) = \begin{cases} 0 & x \in D \circ M(x) \\ \alpha & x \notin D \circ M(x) \end{cases} \quad (6)$$

이다. 여기서  $M(\cdot)$ 은 움직임 맵 (motion map)으로 전후프레임에 움직임차가 있는 부분에 대해서 1이라는 값을 반환한다. 여기서 움직임 차는 전후 프레임간의 밝기값의 변화가 일정 한계치를 넘을 때인 것으로 정의한다. 즉,

$$M(x) = \begin{cases} 0 & \text{if } |I^k(x) - I^{k+1}(x)| < Th \\ 1 & \text{if } |I^k(x) - I^{k+1}(x)| \geq Th \end{cases} \quad (7)$$

과 같이 정의한다. 여기서  $I^k(x)$ 은 이전 프레임의 밝기값을,  $I^{k+1}(x)$ 은 현재 프레임의 밝기값을, 그리고  $Th$ 은 미리 설정한 한계치를 나타낸다. 또  $D \circ$ 는 모폴로지 연산자 중 팽창(dilation)연산자를 의미한다. 즉,  $D \circ M(\cdot)$ 함수는 현재 프레임중에서 움직임이 있는 영역을 팽창시킨 후 이 영역에 대응하는 현재의 배경영상에 현재 프레임의 밝기값이 반영되지 않게 한다. 이로써 현재 프레임의 전경에 의해 생기는 잔상을 제거하는 역할을 한다. 그림 1(c)는 움직임 맵 영상  $M$ 을 보여주고 있고, 그림 1(d)는  $\alpha = D \circ M$ 함수를 보여주고 있다. 그림 1(e)는  $\alpha$  함수를 식 (5)에 적용하여 계산한 배경영상을 보여주고 있다. 그림 1(f)는 그림 1(e)의 배경영상을 사용하여 계산한 전경영상  $F^n$  영상을 보여주고 있다. 전경영상  $F^n$ 을 계산할 때는 배경영상과의 절대차 영상에 대해 이진화(thresholding)를 수행한 결과로 나온 영역에 대해 팽창(Dilation)연산을 취하여 연결된 전경영역을 얻는다.

### 2.3. 크기정보 추출

제안한 추적방법은 추적대상의 위치는 2.1절에서 제안한 움직임정보를 이용한 컬러중심이동 기반의 추적방법에 의해 구한 위치로 결정하고, 추적대상의 영역은 2.2절에서 제안한 방법으로 계산한  $F^n$ 의 영역에 의해 결정한다. 즉, 추적대상의 영역을 박스로 표현하기 때문에 전경영역  $F^n$  중 연결이 되어 있는 영역을 포함하는 최소 크기의 박스를 추적대상의 영역으로 결정한다.

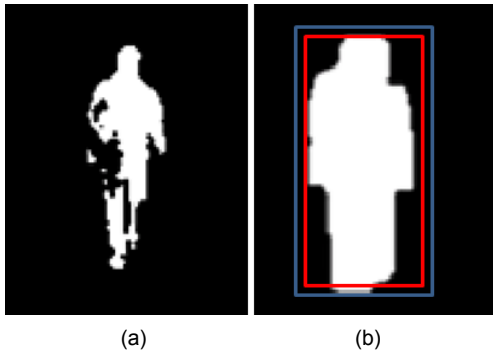


그림 2. 추적대상의 크기 결정 (a) 모폴로지 연산전의 전경영역 (b) 모폴로지 연산후의 전경영역

Fig. 2 Deciding the size of the object (a) foreground region before morphological filtering (b) foreground region after morphological filtering

먼저 하나의 추적대상에 대해서 내부영역들이 되도록 가능한 연결이 되도록 전경영역에 대해 팽창(dilation)연산을 수행한다. 그리고 연결 요소 분석(connected component analysis)에 의해 연결된 영역을 추정한다.

그림2(b)는 그림 2(a)의 전경영역에 대해 팽창연산을 수행한 결과를 보여주고 있다. 그러면 본래 대상의 크기보다 크기가 커지게 되는데 (그림2(b)의 바깥쪽박스), 추적대상의 크기를 구할 때는 팽창(dilation)에 의해 추적대상이 더 커지게 된 정도를 보정하여 추적대상의 크기를 결정한다(그림 2(b)의 안쪽박스).

그런데  $F^n$ 의 연결된 영역이 하나의 추적대상과 반드시 일치하지 않는다는 단점이 있다. 첫 프레임에서는 추적대상의 범위를 수동으로 지정하기 때문에  $F^n$ 이 연결되어 있는 여부와 상관없이 추적대상의 영역과 일치시킬 수 있다. 그러나 이후의 프레임부터는 하나의 추적대상의 영역이 서로 다른 영역들로 분리되거나 서로 다른 추적대상의 영역들이 합쳐지는 문제가 생긴다. 이때 추적대상의 영역을 보정하는 것이 필요하다.

추적대상의 영역이 전후 프레임에서 크기에서 큰 차이를 보일 수 없다는 사실에 근거하여 제안한 방법에서는 현재 프레임에서 추적대상의 영역의 블롭(blob)크기가 전프레임에 비해 80%이상 차이가 날 경우 이를 다른 객체와의 겹쳐짐에 의한 것이거나, 객체영역이 분리된 경우로 판단하고 있다. 이 경우 전 프레임의 추적대상의 영역의 크기를 그대로 유지한다.

그림 3(a), (b)는 한 객체가 2개의 블롭 영역으로 분리

된 경우의 전후 프레임을 보이고 있고, 그림 3(c), (d)는 타객체와 영역이 겹쳐져서 블롭영역이 갑자기 커진 경우를 보이고 있다. 두 경우 모두 전후 프레임의 블롭의 크기가 80%이상의 차이를 보이고 있기 때문에 이 경우 전 프레임의 추적객체의 박스의 크기를 유지한다.

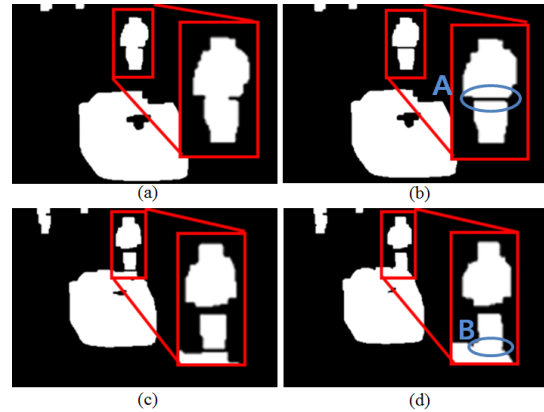


그림 3. 블롭영역의 분리와 겹침 (a)(b) 분리전후의 블롭영역 (c)(d) 겹침전의 블롭영역

Fig. 3 Blob regions before and after splitting and merging (a)(b) blob regions before and after splitting (c)(d) blob regions before and after merging

### III. 실험결과

본 논문에서는 비교 알고리즘으로 Camshift알고리즘과 단순 중심기반(Centroid Shifting)추적알고리즘, 스케일을 조절할 수 있는 평균이동기반의 알고리즘의 대표 알고리즘으로 평균이동 블롭(Meanshift Blob)알고리즘, 그리고 확률적 프레임워크(stochastic framework)를 사용한 알고리즘으로 [13]에서 제안한 적응적 몬테카를로법(Adaptive Basin-Hopping Monte Carlo)을 이용한 추적 알고리즘을 사용하였다. 추적의 정확성을 계산하는 척도로서 [14]에서 제안한 윤곽박스 중첩 비율(bounding box overlap ratio)를 사용하였다:

$$S = \frac{|r_t \cap r_a|}{|r_t \cup r_a|} \quad (8)$$

여기서  $r_a$ 는 추적대상의 참여영역,  $r_t$ 는 추적 알고리즘에 의해 추정된 추적대상의 영역을 나타내며,  $\cap$

표 1. 추적 알고리즘들간의 평균 윤곽박스 중첩 비율 및 처리속도 비교

Table. 1 Comparison of bounding box overlap ratio and processing time between tracking algorithms

추적 방법	알고리즘									
	Camshift		Centroid Shifting		알고리즘 [13]		제안한 방법			
	평균 윤곽박스 중첩비율	평균 처리 시간 (sec/frame)	평균 윤곽박스 중첩비율	평균 처리 시간 (sec/frame)	평균 윤곽박스 중첩비율	평균 처리 시간 (sec/frame)	평균 윤곽박스 중첩비율	평균 처리 시간 (sec/frame)	평균 윤곽박스 중첩비율	평균 처리 시간 (sec/frame)
시퀀스 1	34.5	0.034	72.4	0.029	89.6	0.032	95.2	0.132	93.2	0.047
시퀀스 2	28.8	0.031	68.3	0.027	83.2	0.029	96.4	0.112	95.3	0.043
시퀀스 3	32.5	0.028	71.9	0.026	85.6	0.027	93.0	0.103	90.7	0.042

는 교집합 연산자,  $\cup$  는 합집합 연산자를 나타내고,  $|\cdot|$  는 픽셀의 갯수를 계산하는 연산자를 나타낸다. 윤곽박스 중첩 비율  $S$ 는 0에서 1까지의 범위를 가지며, 값이 클수록 추적대상에 대한 위치 및 영역에 대한 추정이 잘 되었다는 것을 의미한다.

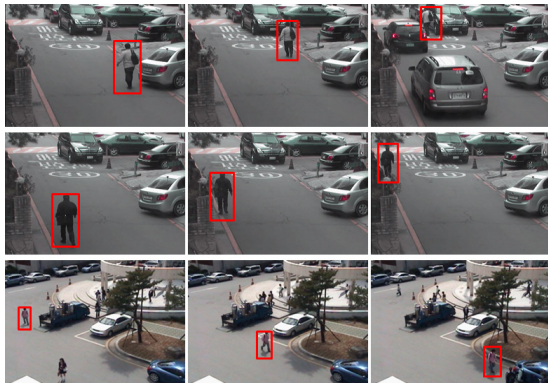


그림 4. 제안한 방법을 사용하여 CCTV영상의 객체추적에 적용한 추적결과

Fig. 4 Tracking results on objects in a CCTV video using the proposed method

그림 4는 하나의 추적대상이 CCTV의 감시영역을 지나가는 짧은 3개의 시퀀스에 대해 제안한 추적 알고리즘의 결과를 보여주고 있다. 주변에 비슷한 색상의 배경영역들이 나타나고 있으나 움직임영역 정보를 이용하기 때문에 주변영역의 색상에 큰 영향을 받지 않고 안정적으로 추적이 되는 것을 볼 수 있다. 표 1은 그림4의 3개의 비디오 시퀀스에 대해 서로 다른 추적 방법들의 추적 결과를 윤곽박스 중첩 비율 및 속도로 비교한 것이다. 제안한 방법은 알고리즘이 간단하여 평균처리 시간이 빠르면서도 [13]의 알고리즘과 비슷한 안정적인

추적결과를 보여주고 있다. 평균처리 시간이 빠른 이유는 복잡한 확률적 프레임워크 대신에 컬러기반의 빠른 추적 알고리즘을 이용했기 때문이다. 실험은 2.33GHz의 CPU로 RAM 메모리가 16GB인 PC에서 윈도우7 운영체제하에서 수행되었다. 그림 5는 시퀀스1에 대해서 프레임별로 계산한 윤곽박스 중첩 비율의 값을 보여주고 있다. CamShift의 경우 불안정하여 추적대상을 놓치고 있으며, 제안한 방법과 [13]의 알고리즘은 비슷한 안정도를 보이고 있음을 볼 수 있다.

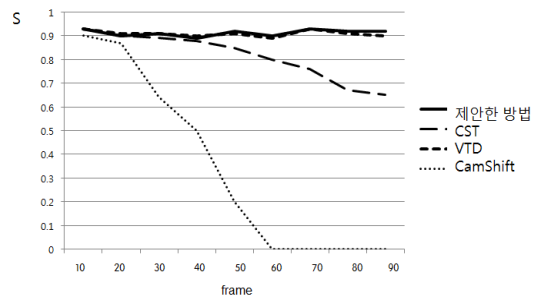


그림 5. 시퀀스1에 대해 프레임별로 계산한 윤곽박스 중첩 비율  
Fig. 5 Computed bounding box overlap ratio of frames in sequence 1

#### IV. 결론

본 논문에서는 CCTV영상의 사후 마스크 처리에 특화된 추적 알고리즘을 제안하였다. 제안한 추적 알고리즘은 오프라인 상에서 추적대상의 영역을 수동으로 결정 한 후에 대상영역의 크기에 적응적으로 자동 추적 및 크기 추정을 수행하여 대상영역을 마스크하는 솔루션에 적용할 수 있다. CCTV영상의 마스크 처리에 특화된



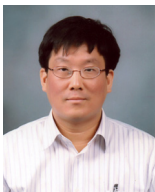
어 계산량이 복잡하지 않으면서도 CCTV의 대상들을 마스킹 처리하는데 충분한 안정성을 갖도록 설계되었다. 추적대상의 크기를 추정할 때 색상정보도 함께 사용하는 방안에 대해 추가적인 연구가 이루어지면 더욱 안정적인 결과가 도출될 것으로 보인다.

### 감사의 글

본 연구는 이 논문은 2014년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업 (No. NRF-2013R1A1A4A01007868) 및 중소기업청의 창업성장-기술개발사업 (과제번호 : S2112375)의 지원에 의해 이루어진 연구로서, 관계부처에 감사드립니다.

### REFERENCES

- [ 1 ] R. Collins, "Mean-shift blob tracking through scale space," *IEEE Conf. on Computer Vision and Pattern Recognition*, Vancouver, pp. 234-240, 2003.
- [ 2 ] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.25, pp. 564-575, 2003.
- [ 3 ] S. H. Lee, E. Choi, and M. G. Kang, "Object tracking based on area weighted centroids shifting with spatiality constraints," *IEEE Int. Conf. Image Processing*, San Diego, USA, pp. 2632 - 2635, 2008.
- [ 4 ] G. R. Bradski, "Computer Vision Face Tracking For Use in a Perceptual User Interface," *Intel Technology Journal*, pp. 1-12, 1998.
- [ 5 ] A. Yilmaz, "Object tracking by asymmetric kernel mean shift with automatic scale and orientation selection," *IEEE Conf. on Computer Vision and Pattern Recognition*, Minneapolis, USA, pp. 1-6, 2007.
- [ 6 ] P. Brasnett, L. Mihaylova, D. Bull, and N. Canagarajah, "Sequential monte carlo tracking by fusing multiple cues in video sequences," *Image and Vision Computing*, vol. 25, no. 8, pp. 1217 - 1227, 2007.
- [ 7 ] C. Yang, R. Duraiswami, and L. Davis, "Fast multiple object tracking via a hierarchical particle filter," *International Conference on Computer Vision*, Beijing, China, vol. 1, pp. 212 - 219, 2005.
- [ 8 ] Q. Wang and J. Liu, "Visual tracking using the kernel based particle filter and color distribution," *International Conference on Neural Networks and Brain*, Beijing, China, vol. 3, pp. 1730 - 1733, 2005.
- [ 9 ] C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real time tracking," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 747-767, 2000.
- [ 10 ] X. Gao, T. E. Boult, F. Coetzee, and V. Ramesh, "Error Analysis of Background Adaption," *In IEEE Int. Conf. on Computer Vision*, Hilton Head, SC, USA, vol. 1, pp. 503 - 510, 2000.
- [ 11 ] L. Li, W. Huang, I. Y. H. Gu, and Q. Tian "Foreground object detection from videos containing complex background", *ACM Multimedia*, 2003.
- [ 12 ] P. K. T. Pong and R. Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection," *Video-Based Surveillance Systems*, pp. 135-144, 2001.
- [ 13 ] J. Kwon and K. M. Lee, "Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive Basin Hopping Monte Carlo sampling," *IEEE Conf. on Computer Vision and Pattern Recognition*, Miami, USA, pp. 1208-1215, 2009.
- [ 14 ] Y. Wu, J. W. Lim, and M. H. Yang, "Online Object Tracking: A Benchmark," *IEEE Conf. on Computer Vision and Pattern Recognition*, Portland, USA, pp. 2411 - 2418, 2013.



이석호 (Suk-Ho Lee)

2013년 연세대학교 전기전자공학과 공학박사  
 2013년~2006년 연세대학교 수학과 박사후과정 연구원  
 2006년~2008년 연세대학교 전기전자공학과 연구교수  
 2008년~현재 동서대학교 컴퓨터정보공학부 부교수  
 ※관심분야 : 영상처리, 컴퓨터비전