

## 객체 추적을 위한 고유 배경의 동적 모델링

김 성 영\*

### Dynamic Modeling of Eigenbackground for Object Tracking

Sungyoung Kim\*

#### 요 약

본 논문에서는 비디오 스트림으로부터 움직이는 객체를 추출하기 위해 고유 배경(eigenbackground)을 사용하여 효율적으로 배경을 모델링하는 방법을 제안한다. 배경은 모델링하더라도 시간이 지남에 따라 날씨나 조명의 변화에 따라 변화가 발생하므로 변화 요소를 반영할 수 있도록 배경 모델을 갱신해야 한다. 이를 위해 본 논문에서는 R-SVD 방법을 기반을 두고 배경 모델을 갱신하도록 한다. 이 때 영상 변화도를 정의하여 이 값에 따라 동적으로 배경을 모델링하여 처리시간을 단축할 수 있도록 한다. 또한 고유 배경을 사용하는 경우 충분한 훈련 데이터를 사용해야만 정확한 모델을 생성할 수 있지만 본 논문에서는 적은 수의 데이터만을 사용하여 정확한 모델을 생성할 수 있도록 입력 프레임을 재구성하여 사용한다. 제안한 방법은 초기 고유 배경 모델 및 기존의 주기적으로 배경을 갱신하는 방법과의 비교를 통해 그 우수성을 확인한다.

▶ Keyword : 고유 배경, 동적 배경 모델링, R-SVD, 객체 추적

#### Abstract

In this paper, we propose an efficient dynamic background modelling method by using eigenbackground to extract moving objects from video stream. Even if a background model has been created, the model has to be updated to adapt to change due to several reasons such as weather or lighting. In this paper, we update a background model based on R-SVD method. At this time we define a change ratio of images and update the model dynamically according this value. Also eigenbackground need to be modelled by using sufficient training images for accurate models

• 제1저자 : 김성영 • 교신저자 : 김성영

• 투고일 : 2012. 02. 28, 심사일 : 2012. 03. 07, 게재확정일 : 2012. 03. 23.

\* 금오공과대학교 컴퓨터공학과(Dept. of Computer Engineering, Kumoh National Institute of Technology)

※ 본 논문은 금오공과대학교 학술연구비에 의하여 연구된 논문임.

but we reorganize input images to reduce the number of images for training models. Through simulation, we show that the proposed method improves the performance against traditional eigenbackground method without background updating and a previous method.

▶ Keyword : eigenbackground, dynamic background modelling, R-SVD, object tracking

## 1. 서 론

객체 추적(object tracking)은 컴퓨터 비전에서 매우 중요한 분야이다[4]. 객체 추적을 위해서는 추적 대상 객체의 검출, 검출한 객체의 추적 및 추적 궤적의 분석 등이 필요하다. 추적 대상 객체를 검출하기 위해서는 포인트 검출자(point detector), 영상 분할(image segmentation), 배경 모델링(background modelling), 지도 학습(supervised learning) 등을 사용할 수 있다[4]. 이 중에서 배경 모델링은 정적인 카메라를 사용하여 움직이는 객체를 추출하는데 널리 사용된다. 이 방법은 배경을 모델링하여 참조 프레임(reference frame)을 생성하고 이 참조 프레임과 현재 프레임의 차를 계산하여 추적 대상 객체를 검출한다. 이 과정에서 배경을 제거하여 추적 대상 객체만을 분리하므로 배경 제거(background subtraction)라고 한다.

배경 제거는 가우시안 혼합(mixture of Gaussians), 커널 밀도 추정(kernel density estimation), 고유 배경(eigenbackground) 등의 접근 방법을 사용한다[5]. 고유 배경은 충분한 훈련 데이터를 사용하는 경우 좋은 성능을 나타내는 것으로 알려져 있고 가우시안 혼합보다 적은 연산량을 요구한다[1]. 또한 실시간 응용에도 사용 가능하다. 하지만 훈련 집합 영상에서 추적 대상 객체들이 유사한 위치에 존재하는 경우에는 배경 모델에 이들 객체의 정보가 함께 저장되므로 입력 프레임 영상으로부터 정확한 객체 검출이 어려운 문제가 있다.

배경 제거에서 배경이란 움직이는 객체를 제외한 나머지 영역으로 정의할 수 있다. 하지만 배경도 완전히 정적인 것은 아니다. 시간이 지남에 따라 날씨가 주변 조명 밝기의 변화 등에 의해 배경도 변한다. 따라서 배경 제거에서는 적응적으로 배경을 갱신하는 것이 매우 중요하다[10]. 고유 배경을 사용하여 배경을 제거하는 경우에는 일반적으로 주기적으로 고유 배경을 다시 계산하여 배경 모델을 갱신한다.

고유 배경을 계산하기 위해서는 입력 영상의 집합을 행렬로 변환한 영상 행렬로부터 고유 벡터(eigenvector)와 고유 값(eigenvalue)을 계산해야 한다. 행렬의 고유 벡터와 고유 값은 행렬의 특이값 분해(singular value decomposition)[6]

를 통해 계산할 수 있다. 일반적으로 영상 행렬은 크기가 매우 크므로 고유 벡터와 고유 값을 계산하기 위해서는 상당한 계산량이 요구된다. 특히 입력 프레임 영상이 추가될수록 영상 행렬도 그 크기가 계속 증가하므로 계산량은 더욱 증가한다. 이로 인해 계산량을 줄일 수 있는 방법들이 발표되었다[7,3]. *Lirui Xu*[7]는 계산량을 줄이기 위해 기존 배경 모델 정보를 무시하고 새로운 입력 프레임 영상만을 사용하여 새로운 배경 모델을 생성하였다. 또한 누적 에러(accumulated error)를 정의하고 이 에러값이 일정치 이상인 경우에만 고유 배경을 다시 계산하였다. 하지만 이 방법은 새로운 입력 프레임 영상을 사용하여 새로운 고유 배경을 계산하므로 기존에 만들어진 고유 배경에 대한 정보를 사용하지 못하며 여전히 많은 계산이 필요하다. *Wang et. al*[3]은 고전적인 R-SVD 방법[2]에 기반을 두고 고유기저(eigenbasis)를 갱신하기 위해 효과적이고 순차적인 변형 Karhunen-Loeve 알고리즘[8]을 사용하여 일정한 프레임 단위마다 배경 모델을 주기적으로 갱신하였다. 하지만 훈련 집합의 영상들이 유사한 위치에 객체를 포함하는 경우 배경 모델에 이들 객체의 정보를 함께 포함하는 문제는 여전히 존재하며 속도 개선의 여지가 있다.

이에 본 논문에서는 고유 배경을 사용하여 배경을 모델링 하되 이 모델을 효과적이고 빠르게 갱신할 수 있는 방법을 제안한다. 이를 위해 본 논문에서는 R-SVD를 사용하되 버퍼에 입력 프레임 영상이 가득차면 주기적으로 고유 배경을 갱신하는 것이 아니라 배경 제거 단계에서 영상 변화도(change ratio)를 계산하여 이 값에 따라 고유 배경의 갱신이 이루어지도록 하여 갱신의 횟수를 현저하게 줄일 수 있도록 한다. 또한 유사 지점에 위치한 객체들의 정보가 배경 모델에 저장되는 것을 최소화하기 위해 입력 프레임 영상을 재구성하여 고유 배경 생성에 사용할 수 있도록 한다. 실험 결과에서는 초기 고유 배경 모델[1] 및 *Wang et. al*[3] 방법과의 처리 시간 및 객체 검출 결과를 비교하여 제안한 방법의 우수성을 검증한다.

본 논문의 구성은 다음과 같다. 우선 II장에서는 고유 배경의 개념에 대해 설명하고 III장에서는 R-SVD 방법 및 본 논문에서 제안하는 고유 배경의 동적 갱신 방법에 대해 설명한다. IV장에서는 실험 결과를 제시하고 마지막으로 V장에서 결론을 맺는다.

## II. 고유 배경

고유 배경(eigenbackground)[1]이란 배경을 모델링하는 고유 공간(eigenspace)을 의미한다. 고유 배경은 날씨나 구름에 의한 태양 빛의 변화에 의한 배경의 변화 정보를 함께 포함하여 배경을 효과적으로 모델링한다.

고유 배경의 모델을 생성하기 위해 우선  $L$ 개의 입력 프레임 영상을 사용하여 수식 (1)과 같이 입력 벡터의 집합( $F$ )을 구성한다. 본 논문에서는 변수는 이텔릭체의 소문자 혹은 대문자로 표기하고 집합(set)은 이텔릭체의 대문자로 표기한다. 그리고 벡터와 행렬은 각각 굵은 글씨체의 소문자와 대문자로 표기한다.

$$F = \{ \mathbf{f}_i \}_{i=1 \dots L} \quad (1)$$

이 때  $\mathbf{f}_i$ 는  $i$ 번째 입력 프레임 영상에 대응하는 열벡터이며 크기는  $MN \times 1$ 이다.  $M$ 과  $N$ 은 각각 입력 프레임 영상의 가로 및 세로 길이이다.

입력 벡터 집합  $F$ 를 사용하여 평균 벡터  $\mu_b$ 를 계산하고 수식 (2)에 의해 공분산 행렬(covariance matrix)  $\mathbf{C}_b$ 을 구한다. 각 입력 벡터와 평균 벡터의 차이를 사용함으로써 공분산 행렬의 평균은  $\mathbf{0}$ (영) 벡터가 된다.

$$\mathbf{C}_b = E\{ \mathbf{x}_i \mathbf{x}_i^T \} \approx \frac{1}{N} \mathbf{X} \mathbf{X}^T \quad (2)$$

이 때  $E(\cdot)$ 는 인자의 기대치(expected value)를 나타낸다. 그리고  $\mathbf{x}_i = \mathbf{f}_i - \mu_b$ 이고  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$ 이다.

공분산 행렬  $\mathbf{C}_b$ 는 고유값 분해(eigenvalue decomposition)에 의해 수식 (3)과 같이 대각화할 수 있다.

$$\mathbf{L}_b = \mathbf{A} \mathbf{C}_b \mathbf{A}^T \quad (3)$$

행렬  $\mathbf{A}$ 는 공분산 행렬  $\mathbf{C}_b$ 의 고유 벡터 행렬이다. 즉 행렬  $\mathbf{A}$ 의 각 행(row)은 공분산 행렬  $\mathbf{C}_b$ 의 고유 벡터이고 고유 벡터는 고유값의 크기에 따라 내림차순으로 위치한다. 따라서 이 행렬의 첫 번째 행에는 가장 큰 고유값에 대응하는 고유 벡터가 위치하고 두 번째 행에는 두 번째 큰 고유값에 대응하는 고유 벡터가 위치하며 순차적으로 마지막 행에는 가장 작은 고유값에 대응하는 고유 벡터가 위치한다. 결과적으로  $\mathbf{L}_b$ 는 고유값에 대응하는 대각행렬이 된다.

공분산 행렬  $\mathbf{C}_b$ 의 고유 벡터 및 고유값은 수식 (4)의 특이값 분해(SVD)에 의해 구할 수 있다.

$$\mathbf{X} = \mathbf{U} \Sigma \mathbf{V}^T \quad (4)$$

행렬  $\mathbf{U}$ 의 각 열은 공분산 행렬  $\mathbf{C}_b$ 의 고유 벡터이고 대각 행렬인  $\Sigma$ 의 각 대각값은 고유 벡터에 대응하는 고유값의 제곱근에 해당하는 값이다. 수식 (3)의  $\mathbf{A}$ 와 수식 (4)의  $\mathbf{U}$ 는  $\mathbf{A} = \mathbf{U}^T$ 의 관계가 있다. 행렬  $\mathbf{A}$ 의 크기는  $L \times MN$ 이다.

수식 (3)에서 주어진 변환 행렬  $\mathbf{A}$ 를 사용하여 수식 (5)와 같이 입력 벡터  $\mathbf{x}$ 를 새로운 출력 벡터  $\mathbf{y}$ 에 매핑할 수 있다. 이 변환을 주성분 분석(principal component analysis, PCA)이라고 한다. PCA는 주어진 데이터에서 가장 높은 분산을 갖는 방향을 계산하고 이 방향을 새로운 축으로 하여 기존 데이터를 다시 표현한다. PCA에서는 새로운 공간의 차원을 줄이기 위해 고유 벡터 행렬  $\mathbf{A}$ 의 모든 고유 벡터를 사용하는 대신 고유 값이 큰 상위  $K$ 개의 고유 벡터만을 사용한다. 따라서 수식 (5)는 수식 (6)으로 변경된다. 행렬  $\mathbf{A}_M$ 은  $K$ 개의 고유 벡터만을 사용하므로 크기는  $K \times MN$ 이다. 이때 행렬  $\mathbf{A}_M$ 을 고유 배경 혹은 고유 공간이라고 한다.

$$\mathbf{y} = \mathbf{A}(\mathbf{x} - \mu_b) \quad (5)$$

$$\mathbf{y} = \mathbf{A}_M(\mathbf{x} - \mu_b) \quad (6)$$

새로운 입력 프레임 영상( $I$ )이 주어지면 열벡터( $\mathbf{x}$ )로 변환하여 수식 (6)을 사용하여 고유 공간에 투영한 후 다시 수식 (7)을 사용하여 복원하면 그림 1과 같이 입력 프레임 영상의 배경 영상을 생성할 수 있다.

$$\hat{\mathbf{x}} = \mathbf{A}_M \mathbf{y} + \mu_b \quad (7)$$

그림 1(d)는 그림 1(c)의 입력 영상을 고유 배경에 투영하여 복원한 결과영상이다. 이 영상에서는 입력 영상의 오른쪽 상단에 위치한 작은 트럭이 나타나지 않는 것을 확인할 수 있다. 이 트럭은 추적 대상 객체이며 그림 1(c)와 그림 1(d)의 차영상으로부터 분리할 수 있다.

고유 배경을 생성할 때 일반적으로 움직이는 객체는 동일한 위치에 나타나지 않으므로 고유 배경에 대한 기여는 작지만 변하지 않는 배경은 기여 정도가 매우 크다. 따라서 입력 프레임 영상을 고유 배경에 투영하면 움직이는 객체를 제외한 나머지 변하지 않는 영역만을 얻을 수 있다. 하지만 움직이는 객체의 속도가 느리거나 초당 프레임 획득 속도가 빠르면 비슷한 지점에 움직이는 객체가 위치하게 되어 고유 배경에 움직이는 객체의 정보가 포함된다. 이와 같은 경우에는 입력 프레임 영상을 재구성하여 사용해야 한다. 이러한 경우에 대해서는 III장에서 다시 논의한다.

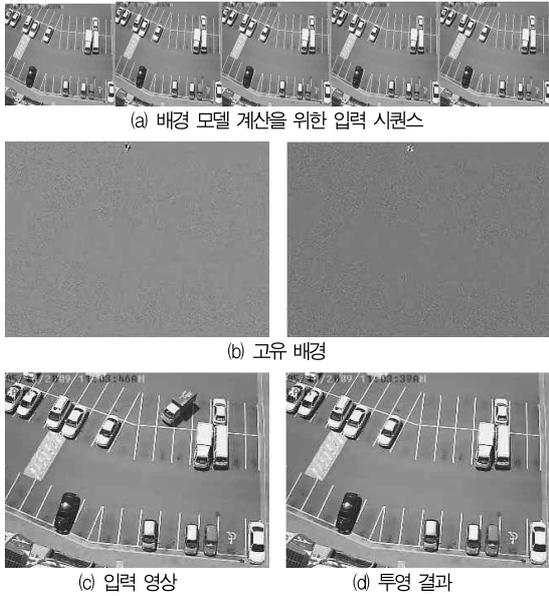


그림 1. 고유 배경을 사용한 생성한 배경 영상  
Fig. 1. A background image by using background model

### III. 고유 배경의 동적 모델링

입력 프레임 영상의 움직이지 않는 영역들도 시간이 지남에 따라 날씨나 조명의 영향에 의해 변한다. 따라서 배경 모델도 시간에 따라 갱신되어야만 한다. 본 논문에서는 R-SVD 방법을 사용하되 모델 갱신 시점을 제어하여 처리 시간을 보다 단축하고 움직이는 객체가 유사한 위치에 나타나는 경우 발생하는 문제점을 개선할 수 있도록 한다.

고유 배경의 갱신을 위해 일정 프레임마다 주기적으로 고유 벡터를 계산하는 것은 과도한 계산량을 요구한다. R-SVD는 크기가 큰 행렬의 SVD를 수행하는데 매우 효율적이다.  $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ 를  $MN \times L$  크기인 행렬  $\mathbf{X}$ 의 특이값 분해라고 하자. 이 때 행렬  $\mathbf{X}$ 의 각 열벡터는 입력 프레임 영상이다. 크기가 큰 행렬  $\mathbf{X}^* = [\mathbf{X}, \mathbf{E}]$ 에 대한 R-SVD은 다음과 같이 수행한다[2]. 이 때  $\mathbf{E}$ 는  $P$ 개의 추가적인 입력 프레임 영상들로 구성된  $MN \times P$  크기의 행렬이다.

- ① Gram-Schmidt와 같은 정규직교화(orthonormalization) 과정을  $[\mathbf{U}, \mathbf{E}]$ 에 적용하여 정규직교 행렬인  $\mathbf{U} = [\mathbf{U}, \tilde{\mathbf{E}}]$ 을 구한다.
- ② 행렬  $\mathbf{V}' = \begin{pmatrix} \mathbf{V} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_K \end{pmatrix}$ 을 생성한다. 이 때  $\mathbf{I}_K$ 는  $K$  차원의 단위행렬(identity matrix)이다.

$$\begin{aligned} \textcircled{3} \quad \mathbf{\Sigma}' &= \mathbf{U}'^T \mathbf{X}^* \mathbf{V}' = \begin{bmatrix} \mathbf{U}^T \\ \tilde{\mathbf{E}}^T \end{bmatrix} [\mathbf{X}, \mathbf{E}] \begin{bmatrix} \mathbf{V} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_K \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{U}^T \mathbf{XV} & \mathbf{U}^T \mathbf{E} \\ \tilde{\mathbf{E}}^T \mathbf{XV} & \tilde{\mathbf{E}}^T \mathbf{E} \end{bmatrix} = \begin{bmatrix} \mathbf{\Sigma} & \mathbf{U}^T \mathbf{E} \\ \mathbf{0} & \tilde{\mathbf{E}}^T \mathbf{E} \end{bmatrix} \end{aligned}$$

④  $\mathbf{\Sigma}' = \tilde{\mathbf{U}}\tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^T$ 에 대한 SVD를 계산하면  $\mathbf{X}^*$ 에 대한 SVD는 다음과 같다. 즉  $\mathbf{X}^*$ 에 대한 SVD는 더 작은 행렬  $\mathbf{U}'$ ,  $\mathbf{V}'$ 과 더 작은 행렬  $\mathbf{\Sigma}'$ 의 SVD를 사용해서 계산할 수 있다.

$$\begin{aligned} \mathbf{X}^* &= \mathbf{U}' \mathbf{\Sigma}' \mathbf{V}'^T \\ &= \mathbf{U}' (\tilde{\mathbf{U}} \tilde{\mathbf{\Sigma}} \tilde{\mathbf{V}}^T) \mathbf{V}'^T = (\mathbf{U}' \tilde{\mathbf{U}}) \tilde{\mathbf{\Sigma}} (\tilde{\mathbf{V}}^T \mathbf{V}'^T) \end{aligned}$$

Wang et. al[3]은 일정한 주기마다 새로운 입력 영상을 사용하여 SVD를 수행하여 고유 배경을 갱신하였다. 주기적으로 고유 배경을 갱신하는 경우 배경의 변화가 거의 없는 경우에도 불필요하게 갱신을 수행하여 처리 효율이 감소한다. R-SVD를 사용하면 연산량을 감소시킬 수는 있지만 여전히 많은 연산이 필요하다. 본 논문에서는 주기적으로 고유 배경을 갱신하지 않고 배경의 변화가 필요한 경우에만 갱신할 수 있도록 한다. 이를 위해 기존 고유 배경에 대한 입력 프레임 영상의 변화도(change ratio)를 계산하고 이 값을 기반으로 갱신 여부를 판단하도록 한다. 변화도(CR)는 수식 (8)을 사용하여 계산한다.

$$CR = \frac{\Sigma(\mathbf{x} - \hat{\mathbf{x}})}{MN} \tag{8}$$

$\mathbf{x}$  및  $\hat{\mathbf{x}}$ 은 각각 입력 프레임 영상의 열벡터와 이를 고유 배경에 투영한 후 복원한 결과의 열벡터이다.  $M$ 과  $N$ 은 영상의 가로와 세로의 크기이다. 변화도가 비록 간단하게 정의되었지만  $\hat{\mathbf{x}}$ 은 고유 배경의 갱신에 따라 생성되므로  $CR$ 은 영상 내의 변화 정도를 잘 표현한다. 크기가  $P$ 인 버퍼를 사용하여 입력 프레임 영상을 저장하고  $CR$ 의 값이 임계값을 초과하는 경우 고유 배경을 갱신한다.

한편 여러 프레임에 걸쳐 비슷한 지점에 객체가 나타나는 경우 고유 배경에 객체의 정보가 포함되어 배경으로부터 객체의 정확한 분리를 힘들게 한다. 충분한 개수의 샘플 수를 사용하여 고유 배경을 생성하면 어느 정도 이런 문제를 극복할 수 있지만 연산량에 대한 부담으로 충분한 샘플을 사용하기가 어렵다. 그림 2를 살펴보면 입력 시퀀스에는 한 대의 트럭이 우측 상단에서 좌측 하단으로 이동하고 있다. 이 영상들을 훈련 데이터로 사용하여 고유 배경을 생성하면 트럭의 이동 경로 중에서 트럭이 겹치는 위치에 의해 배경과 함께 트럭에 대한 정보도 함께 고유 배경에 저장된다. 그 결과는 그림 2(c)

에서 쉽게 확인할 수 있다. 그림 2(b)의 입력 영상을 고유 배경에 투영하고 다시 복원한 결과에서는 입력 영상에 존재했던 트럭이 사라졌지만 입력 영상에는 없었던 트럭의 잔상이 복원 영상에 나타나고 있다. 이 잔상은 고유 배경에 저장된 트럭 정보가 복원 영상에 투영된 것이다. 이와 같은 경우에는 추적 대상 객체를 정확하게 분리하는 것이 어려워진다.



그림 2. 충분하지 않은 영상 샘플의 사용에 따른 고유 배경의 문제점  
Fig. 2. A problem of eigenbackground because of insufficient of image samples

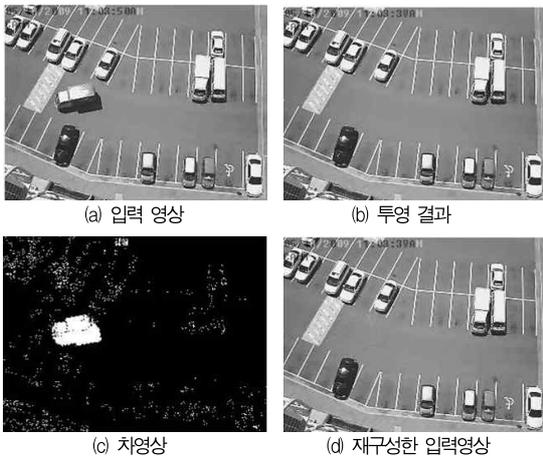


그림 3. 입력 영상의 재구성  
Fig. 3. Reconfiguration of an input image

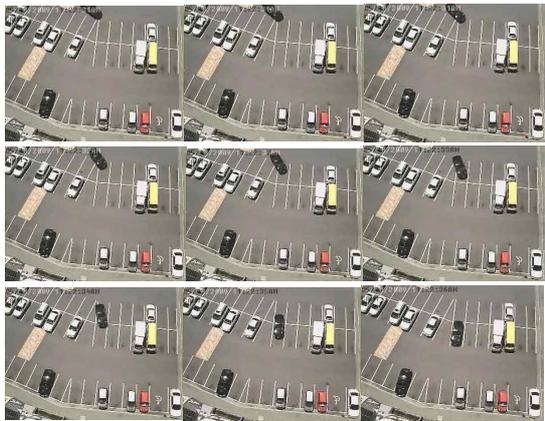
본 논문에서는 적은 수의 입력 영상을 사용하여 고유 배경을 갱신하되 그림 2의 문제가 발생하지 않도록 하기 위하여 입력 프레임 영상에 대해 전처리 작업을 수행한다. 이를 위해 우선 입력 영상을 고유 배경에 투영한 결과 영상과 입력 영상의 차 영상을 구한다. 차 영상은 매우 작은 임계값을 사용하여 이진화(thresholding)를 수행한다. 이진화 결과에서 객체에 해당하는 픽셀(흰색)은 투영 결과 영상에서 복사하고 배경

에 해당하는 픽셀(검은색)은 원 영상에서 복사하여 새로운 입력 영상을 생성한다. 이렇게 생성한 입력 영상은 객체의 영향을 최소화할 수 있다. 이렇게 만들어진 영상은 투영 결과 영상과 비슷하게 보이지만 투영 결과 영상은 원래 영상에서 일부 정보가 상실되었으므로 고유 배경을 갱신하기 위해 사용하기에는 적절하지 못하다.

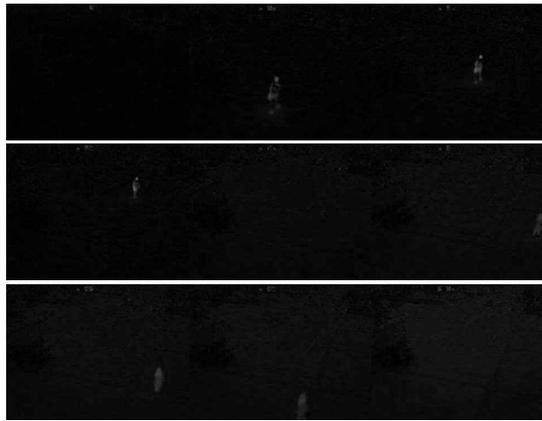
#### IV. 실험 결과

본 논문에서는 제안한 방법의 타당성을 검증하기 위하여 다양한 영상 시퀀스에 대해 실험하였다. 영상 시퀀스는 320 × 240 크기의 칼라 비디오인데 그레이스케일 영상으로 변환하여 사용하였다. 제안한 방법은 2.93GHz의 PC에서 Matlab으로 구현하여 실험하였다.

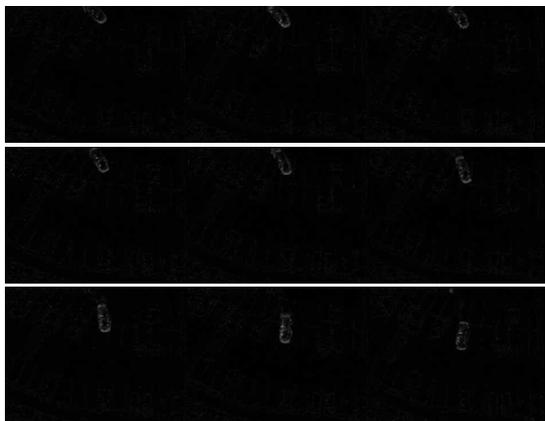
그림 4 및 5는 두 가지 입력 시퀀스에 대한 객체 검출 결과를 나타낸다. 이들 입력 비디오는 각각 1063 및 2022개의 프레임으로 구성된다. 그림 4의 객체 검출 결과는 입력 영상과 배경 모델에 투영한 후 복원한 영상과의 차영상으로 후처리 과정은 수행하지 않은 영상이다. 특히 그림 5는 일몰 시간에 촬영하여 영상의 밝기가 비교적 심하게 변하는 경우에 대한 객체 검출 결과이다. 그림 5(b)에서 그림 5(d)는 각각 Oliver의 초기 고유 배경 모델[1], Wang[3]이 제안한 방법 및 본 논문에서 제안한 방법의 객체 검출 결과이다. 그림 5(b)에서 쉽게 확인할 수 있듯이 배경에 대한 갱신 없이는 일몰에 따른 영상 밝기의 변화를 제대로 수용하지 못하여 시간이 흘러갈수록 추적 객체를 제외한 나머지 영역에서 상당한 노이즈가 발생하고 있다. 이런 상황은 그림 5(e)에서 더욱 쉽게 확인할 수 있다. 그림 5(e)는 일부 객체 검출 결과에 대한 이진화 결과이다. 그림 5(e)의 첫 번째 열은 배경 갱신이 없는 경우이고 두 번째 열은 Wang의 방법이며 세 번째 열은 본 논문에서 제안한 방법이다. 그림 5를 통해 제안한 방법은 Wang의 방법과 유사한 결과를 나타내는 것을 확인할 수 있다. Wang의 방법과 제안한 방법에서는 입력 버퍼의 크기를 동일하게 5로 사용하였다. 즉 Wang의 방법은 5프레임 주기로 배경 갱신을 수행한다. 그리고 제안한 방법에서의 영상 변화도에 대한 설정값은 2로 지정하였다. 본 논문에서 제안한 방법은 영상 변화도를 조절하여 영상의 밝기 변화가 심한 경우에는 배경 모델을 보다 자주 갱신하고 밝기 변화가 적은 경우에는 갱신 빈도를 줄이는 것이 가능하다.



(a) 입력 시퀀스의 일부 영상



(b) Oliver의 방법[1]



(b) 객체 검출 결과

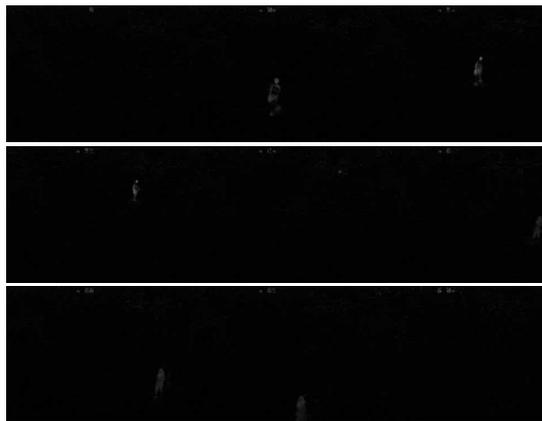


(c) Wang의 방법[3]

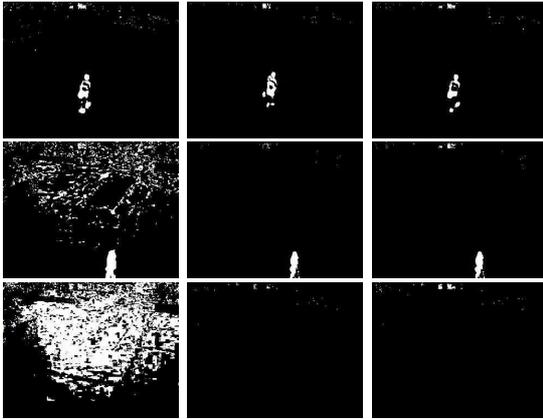
그림 4. 첫 번째 비디오 시퀀스에 대한 객체 검출 결과  
Fig. 4. Object detection results on first video sequence



(a) 입력 시퀀스



(d) 제안한 방법



(e) 처리 방법에 따른 이진화 결과 비교

그림 5. 두 번째 비디오 시퀀스에 대한 객체 검출 결과  
Fig. 5. Object detection results on second video sequence

표 1은 처리 방법에 따른 수행 시간을 비교한 결과이다. 초기 고유 배경 모델에서 배경에 대한 갱신을 수행하지 않고 객체를 검출하는데 소요된 시간은 32.2초로 초당 65.9 프레임을 처리할 수 있지만 주기적으로 배경을 갱신하는 Wang의 방법은 초당 30.1 프레임만을 처리할 수 있다. 반면 본 논문에서 제안한 방법은 초당 52.4 프레임을 처리하여 Wang의 방법과 유사한 검출 결과를 제공하면서 보다 빠른 처리가 가능한 것을 확인할 수 있다.

표 1. 평균 처리속도 비교  
Table. 1. Comparison of average processing speed

방법	처리시간	초당처리 프레임수
Oliver[1]	32.196	65.878
Wang[3]	70.399	30.128
제안방법	40.495	52.376

그림 6은 시간의 흐름에 따라 수식 (8)에서 정의한 영상 변화도값의 변화 추이를 나타낸다. 배경에 대한 갱신을 하지 않는 경우 변화도는 지속적으로 증가하는 것을 확인할 수 있다. 반면 주기적으로 배경을 갱신하는 경우 변화도는 거의 일정한 값을 유지하고 있다. 본 논문에서 제안한 방법은 설정한 변화도값(2.0)을 초과하지 않으며 대체적으로 일정한 값을 유지하고 있는 것을 확인할 수 있다. 다만 입력 초기 과정에서 제안한 방법과 Wang의 방법 사이에 변화도 차이가 일부 발생하고 있는데 이는 프레임의 입력에 따라 Wang 방법은 매 프레임 배경 갱신을 수행하므로 영상 변화도의 값에 큰 변화가 발생하지 않지만 제안한 방법에서는 임계값에 도달할 때

까지 배경 갱신을 수행하지 않으므로 이 구간에서 값의 차이를 나타내고 있다.

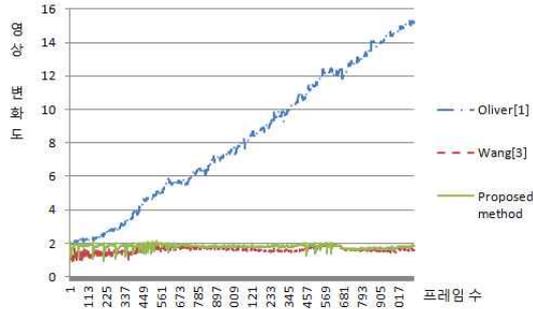


그림 6. 영상의 변화도  
Fig. 6. Change ratios according to methods

#### IV. 결론

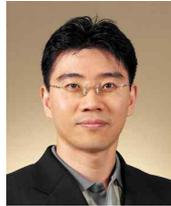
본 논문에서는 고유 배경을 사용하고 이를 동적으로 갱신하여 움직이는 객체를 효과적으로 검출하는 방법을 제안하였다. 고유 배경의 생성을 위해 R-SVD 방법을 사용하였다. 이때 추가적으로 영상 변화도를 정의하고 이를 기준으로 모델 갱신을 동적으로 수행함으로써 기존의 방법에 비해 유사한 검출 결과를 나타내면서 보다 빠른 처리 속도를 얻을 수 있었다. 또한 입력 프레임 영상에서 움직이는 객체가 나타나지 않도록 영상을 재구성하여 작은 훈련 데이터를 사용하여 정확한 배경 모델을 생성할 수 있도록 하였다. 배경에 대한 갱신이 없는 경우[1]에는 초당 처리할 수 있는 프레임 수가 66 프레임 정도 되지만 영상 밝기 변화에 따라 객체 검출 성능이 현저하게 저하되었다. 기존 방법인 Wang[3]의 방법은 좋은 객체 검출 성능을 제공하지만 초당 처리 프레임수가 30 프레임 정도 밖에 되지 않는다. 이에 반해 본 논문에서 제안한 방법 Wang의 방법과 유사한 검출 성능을 제공하면서 초당 52 프레임 정도로 빠른 처리 속도를 나타내었다.

본 논문에서 제안한 방법은 카메라가 고정되어 있는 스마트폰이나 스마트 TV 등의 환경에서 일부 후처리 과정을 추가하여 움직이는 객체는 정확하고 효과적으로 분리하는데 사용할 수 있을 것이다. 또한 객체추적을 통한 증강현실[9] 등의 응용분야에 활용할 수 있다.

## 참고문헌

- [1] Nuria M. Oliver, Barbara Rosario, and Alex P. Pentland, "A Bayesian Computer Vision System for Modeling Human Interactions," *Pattern analysis and machine intelligence*, Vol. 22, No. 8, pp. 831-843, Aug. 2000.
- [2] D. Ross, J. Lim and M.H. Yang, "Adaptive Probabilistic Visual Tracking with Incremental Subspace Update," *Proc. of the 8th European Conf. on Computer Vision*, Vol. 2, pp. 470-482, Nov. 2004.
- [3] Lei Wang, Lu Wang, Qing Zhuo, Huan Xiao and Wenyuan Wang, "Adaptive Eigenbackground for Dynamic Background Modeling," *Intelligent Computing in Signal Processing and Pattern Recognition*, Vol. 345, pp. 670-675, Oct. 2006.
- [4] A. Yilmaz, O. Javed, and M. Shah, "Object Tracking: A Survey," *ACM Journal of Computing Surveys*, Vol. 38, No. 4, pp. 1-45, Dec. 2006.
- [5] Massimo Piccardi, "Background subtraction techniques: a review," *2004 IEEE Int'l Conf. on Systems, Man and Cybernetics*, pp. 3099-3104, Apr. 2004.
- [6] Gilbert Strang, "Linear Algebra and Its Application," Brooks Cole Pub Co, 2005.
- [7] Lirui Xu, "Tracking Pedestrian Trajectory Using Adaptive Background Model," *Engineering Master's Thesis in Kumoh National Institute of Technology*, Feb. 2009.
- [8] A Levy and M Kindenbaum "Sequential Karhunen-Loeve basis extraction and its application to images," *IEEE Transaction on Image Processing*, Vol. 9, pp. 1371-1374, Aug. 2000.
- [9] Hag-hee Kim, "Realization of augmented reality and object tracking using multiple camera," *Journal of The Korea Society of Computer and Information*, Vol. 16, No. 6, pp.89~97, Jun. 2011.
- [10] Young-Suk Ji, Young-Joon Han, Hern-Soo Hahn, "Robust Method of Updating Reference Background Image in Unstable Illumination Condition," *Journal of The Korea Society of Computer and Information*, Vol. 15, No. 1, pp.91~102, Jan. 2010.

## 저자 소개



김성영

2004 : 부산대학교

컴퓨터공학과 공학박사

현재 : 금오공과대학교

컴퓨터공학과 부교수

관심분야 : 컬러 영상 분할 패턴인식

비디오 처리

Email : sykim@kumoh.ac.kr