

# 효과적인 실시간 배경 모델링을 위한 환경 변수 결정 방법

(Determining Method of Factors for Effective  
Real Time Background Modeling)

이 준 철 † 류 상 료 †† 강 성 환 ††† 김 승 호 †††  
(Juncheol Lee) (Sangryul Ryu) (Sunghwan Kang) (Sungho Kim)

**요 약** 다양한 환경을 포함하고 있는 동영상에서 움직이는 객체를 추출, 인식하기 위해서는 배경 모델링이 중요하다. 이러한 객체 인식을 위한 전처리 과정인 배경 모델링을 위한 여러 방안이 제안되었다. 그 중 큐 기반 배경 모델링으로 대표되는 Kumar의 방법이 있다. 하지만 이는 프레임의 갱신검사 주기가 고정되어 있어 여러 시스템에 적용시키는데 한계점이 있다. 본 논문은 큐 기반 배경 모델링 기법을 이용하고 이때 주요한 환경 변수가 되는 슬라이딩 윈도우의 크기 및 영상의 밝기 단계에 따른 그룹핑 크기, 프레임의 갱신검사 주기를 배경 모델에 따라 적응적으로 결정하는 방법을 제안한다. 배경 모델에 따른 환경 변수를 결정하기 위해 객체 검출율, 객체 오검출율, 갱신율을 평가 기준으로 삼는다. 제안된 방법으로 실시간 처리에 부적합한 기존의 배경 모델링 방법을 개선하여 보다 효과적으로 객체를 인식할 수 있다.

**키워드** : 배경 모델링, 객체 인식, 감시 시스템, 실시간 이미지 처리

**Abstract** In the video with a various environment, background modeling is important for extraction and recognition the moving object. For this object recognition, many methods of the background modeling are proposed in a process of preprocess. Among these there is a Kumar method which represents the Queue-based background modeling. Because this has a fixed period of updating examination of the frame, there is a limit for various system. This paper use a background modeling based on the queue. We propose the method that major parameters are decided as adaptive by background model. They are the queue size of the sliding window, the size of grouping by the brightness of the visual and the period of updating examination of the frame. In order to determine the factors, in every process, RCO (Ratio of Correct Object), REO (Ratio of Error Object) and UR (Update Ratio) are considered to be the standard of evaluation. The proposed method can improve the existing techniques of the background modeling which is unfit for the real-time processing and recognize the object more efficient.

**Key words** : Background Modeling, Object Recognition, Surveillance System, Realtime Image Processing

## 1. 서 론

일상생활에서 사용하는 많은 종류의 정보들 중에서

동영상과 같은 시각 정보가 차지하고 있는 비율은 매우 크며, 가장 효과적인 정보 전달력을 가지고 있다. 최근에는 영상 해석을 통하여 동영상에서 상대적으로 많은 움직임을 보이는 객체를 추출하거나 특징을 분석하여 컴퓨터 비전을 이용한 자동화 시스템에 응용하는 방향으로 동영상 처리 기술이 발전하고 있다[1]. 객체 인식은 감시 시스템에서 객체의 특징을 추출하여 행동을 인식 및 분석하기 위해 이용된다. 그리고 더 나아가 객체의 의미 해석 또는 영상 내에서의 이벤트 탐지 등에 활용 할 수 있다[2-5].

배경 모델링은 빛의 변화, 객체의 이동 등과 같이 실

† 정 회 원 : 부천대학 인터넷과 교수

jclee@bc.ac.kr

†† 정 회 원 : 청운대학교 컴퓨터학과 교수

rsr@chungwoon.ac.kr

††† 학생회원 : 경북대학교 컴퓨터공학과

shkang@mmlab.knu.ac.kr

†††† 중신회원 : 경북대학교 컴퓨터공학과 교수

shkim@knu.ac.kr

(Corresponding author임)

논문접수 : 2006년 4월 25일

실사완료 : 2006년 11월 20일

내의 환경에서 변화하는 요소로부터 객체를 정확하게 인식하기 위한 전처리 과정이다. 실시간 감시 시스템은 나뭇가지의 흔들림 등의 급격한 조도의 변화 등의 요소를 고려한 배경 모델링 과정을 통해 추출된 객체를 인식하고 그 의미를 해석한 것을 이용한다[2-5]. 감시 시스템처럼 실시간으로 동작하기 위해서는 다양한 환경을 포함하고 있는 동영상 및 감시 시스템을 동작시킬 시스템 환경에 적응적인 환경 변수의 설정이 중요하다.

배경 모델링을 통한 객체를 추출하는 방법은 Hari-taoglu[6]에 의해 제안된  $W^A$ 의 통계적 방법(statistical method)과 Lai[7]에 의해 제안된 스코어보드 알고리즘(Scoreboard algorithm)의 큐 기반 방법(Queue-based method), Kumar[8]에 의해 제안된 방법이 있다.  $W^A$ 는 입력되는 흑백 영상으로부터 각 픽셀의 밝기 값에 대한 통계적으로 접근하여 배경 모델링을 수행한다. 그러나  $W^A$ 는 흑백 영상을 기반으로 하여 칼라 영상보다는 적은 정보를 포함하고 있기 때문에 실외 환경의 다양한 변화 속에서 객체를 정확히 추출하기에는 한계를 가지고 있다. 스코어보드 알고리즘은 러닝 모드(running mode)와 러닝 평균 알고리즘(running average algorithm)의 방법에 대한 각각의 장점을 이용하였다. P. Kumar의 방법은 큐를 사용한 실시간 배경 모델링 방법으로  $W^A$ 와 마찬가지로 객체가 포함된 영상에서 배경 모델링을 수행 할 수 있다. 하지만 이는 프레임 갱신 검사 주기를 고정적으로 정하고 있어, 여러 시스템에 적용적으로 적용시키기에 부적합하다.

본 논문에서는 기존의 배경 모델링 기법과 비교해서 효과적인 객체 인식을 위해 고화질, 저용량의 실시간 처리가 가능하도록 적응적인 환경 변수의 결정 방법을 제안한다. 입력 동영상에서 배경 영역과 비교하여 움직임이 많은 객체를 추출하는 배경 모델링 과정을 통해 배경 영상과 객체를 구분한다. 그리고 배경 모델링 과정과 주기적인 갱신 과정의 실시간 처리를 위해 입력 영상의 각 픽셀의 색상 값을 그룹화하고 중간 값을 이용함으로써 영상 정보의 크기를 감소시킨다. 다음으로 입력 영상과 배경

모델링된 영상의 차 영상을 구하여 객체를 추출한다.

본 논문의 구성은 다음과 같다. 2장에서 객체 추출을 위한 배경 모델링 방법에 대해 다루고 3장에서는 본 논문에서 제안하는 세 가지 환경 변수, 즉 큐 크기와 그룹핑 크기, 갱신검사 주기를 결정하는 방법에 대해 다룬다. 4장에서는 3장에서 제안하고 있는 방법을 실험을 통해 환경변수들의 값을 다양한 입력 영상과 적용한 시스템에 대해 적용적으로 결정하고, 기존의 방법과 비교한다. 마지막으로 5장에서는 결론을 도출한다.

## 2. 객체 추출을 위한 배경 모델링 방법

객체를 추출하기 위해서는 배경 모델링이 필요하다. 그림 1은 전체 시스템 구조도를 나타낸 것으로 객체 추출을 위한 배경 모델링 방법에 대해 기술하고 있다. 입력 영상으로부터 배경 모델링을 하는 과정에서 큐의 크기와 영상의 밝기 단계에 따라 그룹핑 크기를 조정한다. 그룹핑 크기에 의해 그룹 레벨이 정해지고 그룹핑이 이루어진 배경 영상에서 갱신검사 주기를 조절하고, MSE(Mean Square Error)와 임계치(Th)를 구한다. MSE가 임계치 이상일 때만 배경 영상을 갱신한다. 이렇게 생성된 배경영상에서 다음 프레임으로 입력되는 배경 영상과의 차감 연산을 통해 객체를 추출한다 [9-14].

다양한 영상과 적용된 시스템에 적합한 배경 모델링을 위해 입력 영상에서 적절한 큐 크기를 결정하고, 각 픽셀의 밝기 정보로 그룹핑을 결정하여 계산 복잡성을 줄인다. 그리고 배경 갱신검사 주기를 판단하여 배경 모델링 시 불필요한 영상의 갱신을 줄여준다.

초기 배경 모델링 이후 입력 프레임은 시간의 흐름과 햇빛, 구름, 주변광 등의 환경에 따라 배경 변화가 발생하기 때문에 정확한 객체 검출을 어렵게 하는 문제가 발생한다. 따라서 이러한 문제 해결을 위해서는  $N+1$ 번째 프레임 이후에도 지속적으로 배경의 변화를 반영해야 하는 시점에 배경 영상을 갱신해야 한다. 새로운 프레임이 입력될 때 마다 가장 오래된 프레임의 정보를

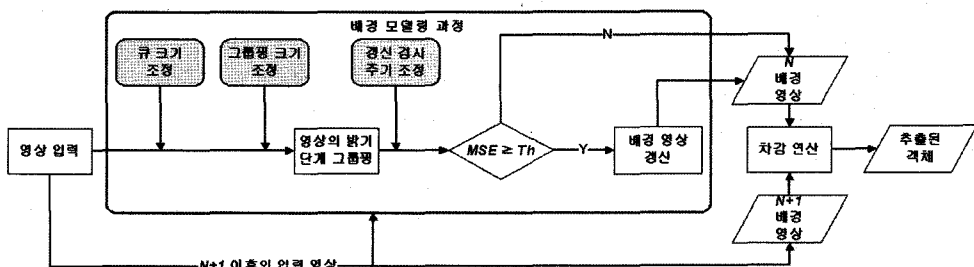


그림 1 전체 시스템 구조도

버리고 새 프레임의 정보를 넣는 슬라이딩 윈도우 (Sliding Window) 방법을 사용한다. 그림 2는  $N$ 개의 큐에서  $N+1$ ,  $N+2$ 번째의 새로운 프레임이 들어 왔을 때의 큐 갱신 과정을 보여준다.

움직임이 없는 건물을 고정된 위치에서 촬영하게 되면  $N$ 개의 프레임 동안 특정 픽셀의 밝기 변화를 면밀히 관찰하면 미세한 차이를 발견 할 수 있지만, 사람의 눈은 미세한 변화를 인식하지 못하여 하나의 정지된 영상으로 보게 된다. 이러한 특성을 이용하여 사람이 인식하는데 원영상과 차이를 느끼지 못할 정도의 밝기 단계를 크기  $g$ 로 그룹핑한 영상을 만들어 이용한다. 영상에서 밝기 단계의 그룹핑이란 연속적인  $l$ 개의 밝기 단계를 한 개의 단계로 사상하는 것이다. 예를 들어 그룹핑 크기  $g$ 를 3으로 두면 밝기 단계  $l=0,1,2$ 는 그룹 레벨  $g_1$ 으로 사상할 수 있다. 배경의 밝기 값으로 사용할 때 디안은  $N$ 개의 프레임 동안 그룹핑된 레벨에서 각 레벨

별로 카운트 정보를 증가시키고 카운트가 0이 아닌 레벨을 추출하여 가장 낮은 그룹 레벨부터 카운트를 합산하여  $N/2$  이상이 되는 시점의 그룹 레벨을 선택한다. 그림 3은  $N$ 개의 연속된 프레임의 임의의 위치  $(x,y)$ 에서 픽셀 값들을  $g=3$ 으로 그룹핑하고 메디안을 구한 후 배경 모델링하는 과정을 보여준다.

영상간의 MSE와 임계치를 사용하여 배경 변화에 적용하면서 갱신 여부를 확인하여 필요할 때에만 배경 갱신을 수행한다. 그림 4는 기간에 따른 MSE 계산과 갱신 여부에 대한 과정을 나타낸다. MSE와 임계치는 최초의 배경 모델링을 위해 입력한  $N$ 개의 프레임들 통해 생성된 배경 영상과 이후에 입력된  $k$ 개의 영상을 사용하여 식 (1)과 식 (2)와 같이 구한다.

$$MSE_N = \frac{\sum [f_N(i,j) - f'(i,j)]^2}{W \times H} \quad (1)$$

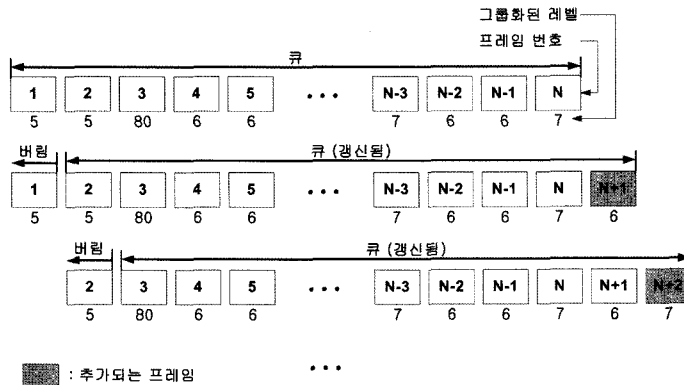


그림 2 큐 갱신 과정

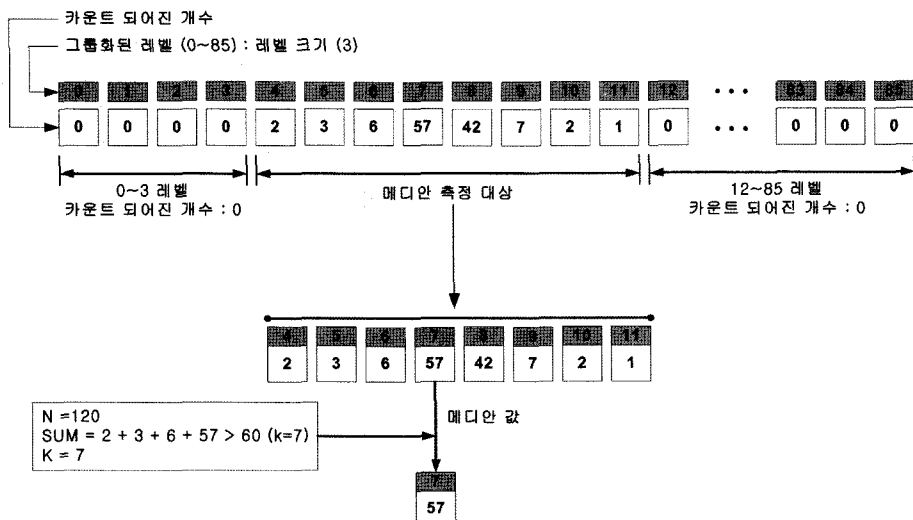


그림 3 메디안 값의 결정 예

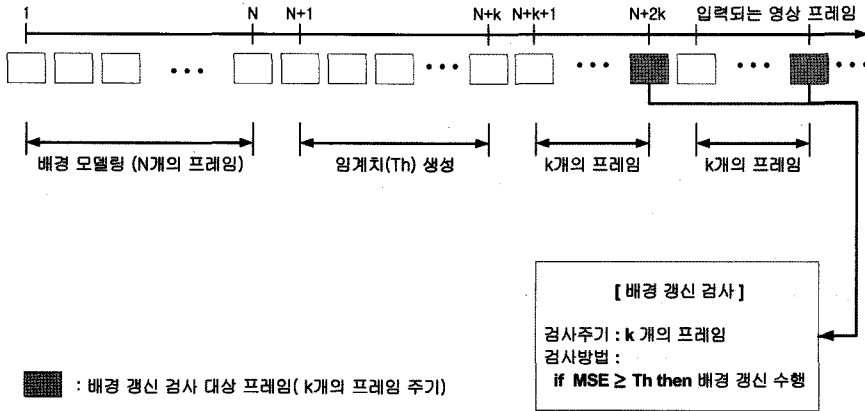


그림 4 시간에 따른 배경 갱신검사

$$Th = \left( \frac{\sum_{i=N+1}^{N+k} MSE_i}{k} \right) - \beta \quad (2)$$

$W$ 와  $H$ 는 입력 영상의 가로와 세로의 크기이다.  $f(i, j)$ 는 현재 입력된 영상의  $(i, j)$  위치에 있는 픽셀 값이며,  $f'(i, j)$ 는 이미 구하여 사용 중인 배경 영상의  $(i, j)$  위치에 있는 픽셀 값이다.  $\beta$ 는 임의의 상수 값이다. MSE는 현재 입력 영상과 배경 영상과의 밝기 차이 값을 의미하며,  $Th$ 는 일정 구간  $k$ 에서 MSE의 평균으로 배경 갱신 판별의 임계치로 사용된다. MSE는 배경 갱신 여부를 결정하기 위해  $k$ 주기마다 임계치와 비교되며, 만약 임계치 이상이 되면 배경 영상을 갱신하고 임계치를 새로 구한다.

3. 큐 크기와 그룹핑 크기, 갱신검사 주기 결정

다양한 환경과 적용된 시스템에 적응적인 대응을 위하여 2장에서 소개한 배경 모델링과 배경 갱신 과정에 사용되는 세 가지 환경 변수로 큐 크기와 그룹핑 크기, 갱신검사 주기 결정 방법을 제안한다. 본 논문에서 적용된 시스템에 적응적으로 대응한다는 것은 특정 규격으로 제한되지 않고 다양한 하드웨어를 수용 할 수 있도록 구현되어 임의의 환경에 적합한 시스템을 의미한다. 각 환경 변수는 나무 등의 영향으로 배경이 자주 변하거나 배경의 변화가 거의 없는 경우처럼 입력 영상의 상태 또는 시스템의 메모리 용량 및 속도에 따라 차이를 가진다. 따라서 서로 다른 환경의 시스템에 동일한 환경 변수 값을 적용하면 성능이 떨어지는 문제점이 발생한다.

3.1 객체 검출율과 객체 오검출율, 갱신율

이러한 문제점을 해결하기 위해 다양한 시스템에 적합한 환경 변수 값을 결정하고 적용시키는 것이 중요하다. 제안하는 방법은 각 영상의 배경 상태와 시스템 상

태에 따라 큐 크기, 그룹핑 크기, 그리고 갱신검사 주기를 적합한 값으로 결정할 수 있다. 적합한 환경 변수 값 결정을 위해서 객체 검출율 (Ratio of Correct Object, RCO), 객체 오검출율 (Ratio of Error Object, REO), 그리고 갱신율 (Update Ratio, UR)을 평가 기준으로 사용한다. 객체 검출율과 객체 오검출율은 Ground Truth 영상을 정의한 후 검출된 객체와의 비교를 통해 구한다.

Ground Truth 영상은 입력된 영상에서 잡음이 전혀 들어 있지 않은 정확하게 객체의 정보만을 포함한 영상을 의미한다. 이 영상은 사람이 직접 추출한다[15]. 그림 5(a)는 임의의 입력 영상에서 추출한 Ground Truth 영상이다. 객체 검출율과 객체 오검출율은 본 논문에서 제안한 객체 추출 과정을 통해 구해진 영상(그림 5(b))와 Ground Truth 영상을 비교하여 구한다.



(a) Ground Truth (b) 인식 결과

그림 5 Ground Truth 영상과 객체 인식 결과 영상

$G_T$ 는 Ground Truth 영상에서 객체의 픽셀 집합을 의미하고,  $R_I$ 는 배경 모델링과 객체 인식 과정을 수행한 후 객체의 픽셀 집합을 의미한다.

각 평가 기준은 다음과 같이 정의된다.

- 객체 검출율(RCO): 배경 모델링과 객체 인식 과정을 수행한 후 얻은 영상에서 객체의 픽셀 개수가 Ground Truth 영상의 객체 픽셀 개수에 포함되어

있는 비율(그림 5(a))

- 객체 오검출율(REO): 객체 검출율과 반대로 Ground Truth의 객체에 포함되지 않은 비율(그림 5(b))
- 갱신율(UR): 임의의 갱신검사 주기 동안 실제 갱신이 이루어진 비율

각 평가 기준은 다음의 특징을 가진다.

- 객체 검출율은 반드시 추출해야 하는 객체의 비율이 되고, 객체 오검출율은 추출하지 않아야 하는 비율이 된다.
- 갱신율이 높으면 갱신검사 주기를 줄여야 하고, 갱신율이 낮으면 갱신검사 주기를 늘여야 한다.
- 세 개의 환경 변수는 객체 검출율은 높고 객체 오검출율과 갱신율은 낮도록 결정되어야 한다.

식 (3)은 객체 검출율을 구하는 수식이고, 식 (4)는 객체 오검출율을 구하는 수식이다. 식 (4)의 객체 오검출율은 100% 이상이 발생할 경우 100%로 나타낸다.

$$RCO = \frac{\sum(RI_{(x,y)} = G_T(x,y))}{\sum G_T(x,y)} \times 100 \quad (3)$$

$$REO = \frac{\sum(RI_{(x,y)} \neq G_T(x,y))}{\sum G_T(x,y)} \times 100 \quad (4)$$

만약,  $REO \geq 100\%$  이면  $REO = 100\%$ 로 한다.

그림 6은 객체 검출율과 객체 오검출율에 대한 예를 보여준다. 그림 6(a)와 6(b)의 각각 윗부분은  $RI$ 를 나타내고, 아래 부분은  $G_T$ 를 나타낸다.



(a) RCO (b) REO  
그림 6 객체 검출율과 객체 오검출율의 예

예를 들어 그림 6의 경우 결과 영상( $RI$ )은 객체의 일부가 잘못 검출되어 있다. 즉, 객체의 머리는 검출되지 않았고 객체의 그림자는 검출되었다. 그러나 객체의 몸과 다리는 정상적으로 검출되었으므로 객체 검출율은 그림 6(a)와 같이 객체의 머리를 제외한 몸과 다리 부분이 되고, 객체 오검출율은 그림 6(b)와 같이 그림자 부분이 된다.

배경 모델링은 식 (1)과 식 (2)의 조건에 따라 갱신 여부를 결정한다. 따라서 갱신율을 구함으로써 불필요한 갱신검사를 하지 않고, 객체 검출율과 객체 오검출율간

의 상호 관계성을 고려함으로써 다양한 시스템에 적합한 환경 변수 값을 얻을 수 있다. 갱신율  $UR$ 은 식 (5)와 같다.

$$UR = \frac{\text{실제 갱신 횟수}}{\text{갱신 검사 횟수}} \times 100 \quad (5)$$

### 3.2 세 가지 환경 변수의 결정 과정

다양한 입력 영상과 적용된 시스템에 적합한 세가지 환경 변수 값을 결정하는 과정은 그림 7, 그림 8, 그림 9와 같다. 환경 변수인 큐 크기의 초기 값은 1로 하고, 그룹핑 크기와 갱신검사 주기의 초기 값은 자원의 최대 허용값으로 설정한다. 그리고 객체 검출율과 객체 오검출율, 갱신율에 대해서 최적화될 수 있도록 매 프레임마다 반복적으로 값을 변화시키면서 최종값을 구한다. 큐 크기와 그룹핑 크기는 객체 검출율이 이전에 구한 값보다 크고 객체 오검출율이 이전 값보다 낮으면 최종값으로 결정된다. 배경 영상의 갱신검사 주기는 갱신율이 이전보다 낮은 경우에 새로운 주기로 결정된다. 단, 큐 크기 및 그룹핑 크기의 결정 방법과 마찬가지로 객체 검출율은 이전보다 높아야 하고 객체 오검출율은 낮아야만 한다. 적용된 시스템에 적합한 환경 변수 값인 큐 크기, 그룹핑 크기, 갱신검사 주기를 결정하는 과정은 다음과 같다.

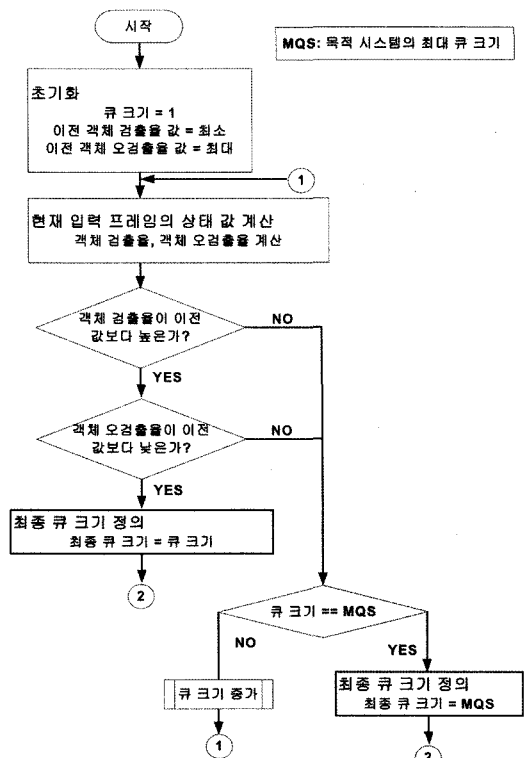


그림 7 목적 시스템에 적합한 큐 크기의 결정 과정

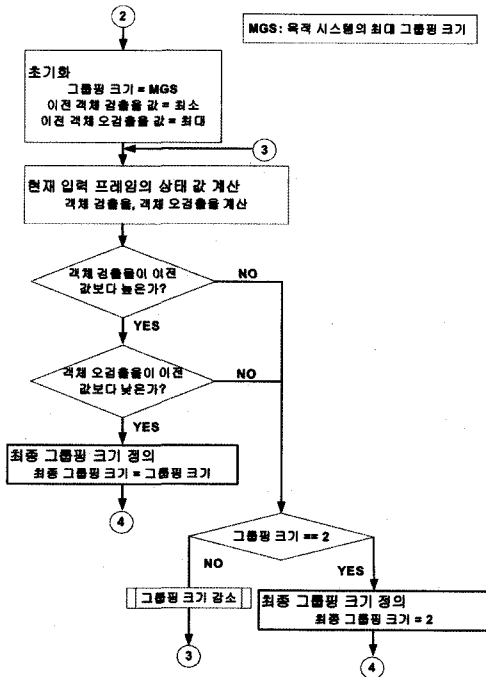


그림 8 목적 시스템에 적합한 그룹핑 크기의 결정 과정

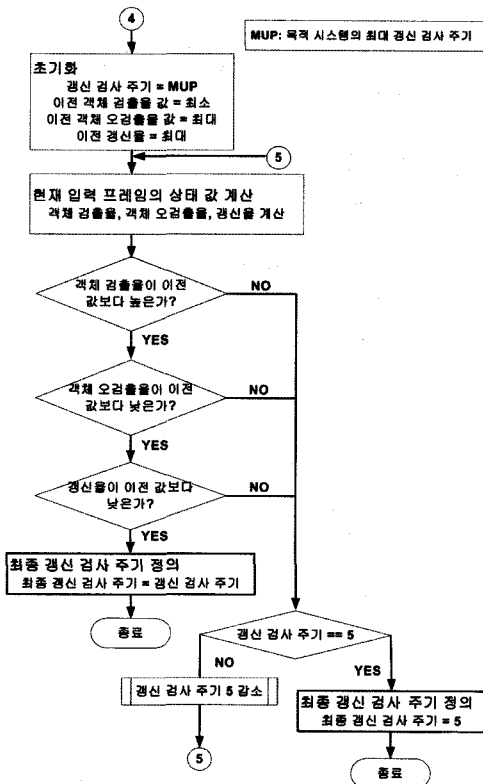


그림 9 목적 시스템에 적합한 갱신검사 주기의 결정 과정

### 4. 실험 및 고찰

3장에서 제안한 효과적인 객체인식을 위한 적응적 환경 변수의 결정 방법에 대한 성능을 분석하고 기존 방법과 비교한다. 다양한 환경의 입력 영상과 적용된 시스템에 적합한 배경 모델링과 배경 갱신을 하기 위해서는 세 가지 환경 변수인 큐의 크기, 그룹핑 크기, 그리고 배경 갱신검사 주기를 결정해야 한다. 환경 변수 값은 3장에서 제안한 객체 검출률, 객체 오검출률, 갱신율을 성능 평가의 기준으로 하여 실험한다. 그리고 결정된 세 가지 환경 변수 값을 이용하여 제안한 방법과 기존의 방법 간의 성능을 분석하였다.

제안한 방법의 성능을 분석하기 위하여 실험 영상으로는 그림 10과 같이 배경 변화가 많은 영상, 배경 변화가 중간 정도인 영상, 배경 변화가 거의 없는 영상으로 세 가지 종류의 영상을 사용하였다. 그리고 실험에 사용된 하드웨어와 소프트웨어의 사양은 표 1과 같다.

4.1절에서는 배경 모델링과 배경 갱신에 필요한 큐의 크기와 그룹핑 크기, 갱신검사 주기에 대한 적합한 값을 산출하기 위한 실험 결과를 보여 준다. 4.2절에서는 기존 방법과의 비교를 통해 제안한 방법의 향상된 성능을 보여 준다.



(a) 배경 변화가 많은 영상 (b) 배경 변화가 중간 정도의 영상



(c) 배경 변화가 거의 없는 영상

그림 10 실험 영상의 예 (각 1,100개 프레임)

표 1 하드웨어 및 소프트웨어 사양

실험환경	사양
중앙처리장치	펜티엄 IV 2.8 GHz
주기억장치	512 Mbyte
구현언어	Microsoft Visual C++
영상크기	360 × 240 (pixel, QVGA)
영상속도	초당 30개 프레임

4.1 큐 크기와 그룹핑 크기, 갱신검사 주기의 결정

큐 크기는 세 가지의 종류의 실험 영상을 배경 모델링한 후 추출한 객체를 대상으로 객체 검출율과 객체 오검출율을 계산함으로써 구하였다. 그림 11은 배경 종류에 따른 객체 검출율과 객체 오검출율을 보여준다.

큐의 크기는 그림 11에서 알 수 있듯이 120일 때 객체 검출율과 객체 오검출율의 모든 기준에서 좋은 결과를 보인다. 객체 검출율은 큐의 크기가 120이 되기 전까지 계속 증가를 한 후 120 이상이 된 후부터 거의 변화가 없었다. 객체 오검출율은 큐의 크기가 40 정도일 때 급격히 떨어진 후 변화가 거의 없다가 120이 되는 지점에서 다시 떨어졌다. 큐의 크기가 40에서 120 사이에서는 객체 오검출율은 크게 낮아지지만 객체 검출율이 계속 증가하기 때문에 적절한 큐의 크기로 결정될 수 없다.

실험을 통해서 큐의 크기가 너무 작으면 객체에 대한 전체 통계 정보가 적어서 객체가 아닌 잡음 등의 영향을 많이 받음을 알 수 있었다. 반면 큐의 크기가 너무 크면 불필요한 메모리의 공간 낭비가 초래되었다. 따라서 각 영상에 대한 배경 모델링에 적합한 큐의 크기는 실험 결과에 따라 120으로 결정하였다.

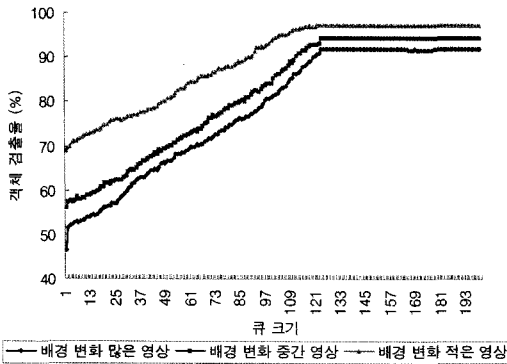
그룹핑 크기는 256 단계의 색상 정보를 최대 그룹핑 크

기(본 실험에서는 8G)로 시작하여 크기를 1씩 감소하면서 객체 검출율과 객체 오검출율을 구하여 결정하였다.

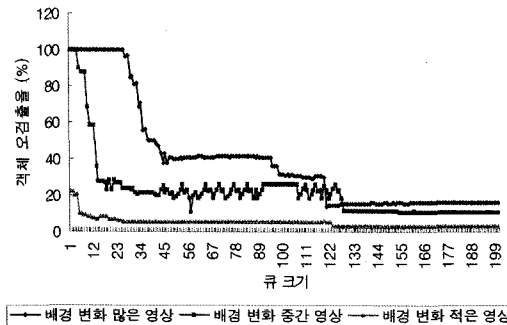
앞에서 결정한 큐의 크기 120에 대해 8G, 7G, ..., 3G, 2G에 대한 성능 평가를 수행한 결과는 그림 12와 같다. 그룹핑 크기는 2G와 3G에서는 객체 검출율과 객체 오검출율은 큰 차이가 나지 않았지만, 4G에서는 객체 검출율은 떨어지고 객체 오검출율은 상당히 증가함을 알 수 있다. 따라서 그룹핑 크기는 3G(86 단계)로 결정하는 것이 처리 속도와 메모리 공간을 줄이는데 바람직한 것으로 알 수 있었다.

마지막으로 배경 갱신검사 주기의 결정은 앞에서 결정된 큐 크기 120, 그룹핑 크기 3G 상태에서 갱신검사 주기를 최대 허용 주기(본 실험에서는 150개 프레임)에서 5 프레임 단위로 감소시키면서 객체 검출율과 객체 오검출율의 성능을 분석하여 결정하였다. 그림 13(a)는 갱신검사 주기에 따른 객체 검출율을, 그림 13(b)는 객체 오검출율의 결과를 보여준다.

배경 영상의 갱신검사 주기는 15개 프레임마다 수행하는 것이 가장 효율적인 것을 알 수 있었다. 갱신검사 주기가 15개 프레임보다 크면 객체 검출율은 서서히 감소하고 객체 오검출율은 증가하고 있음을 알 수 있다.

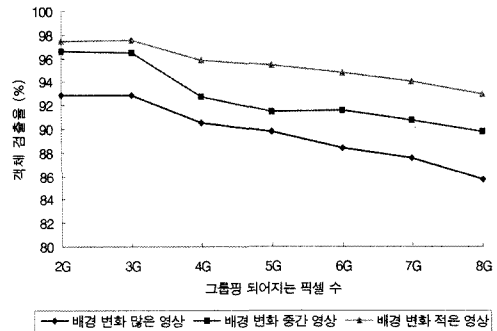


(a) 객체 검출율 결과 (RCO)

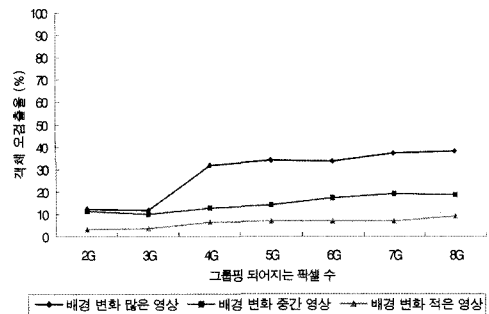


(b) 객체 오검출율 결과 (REO)

그림 11 큐 크기에 따른 객체 검출율과 객체 오검출율

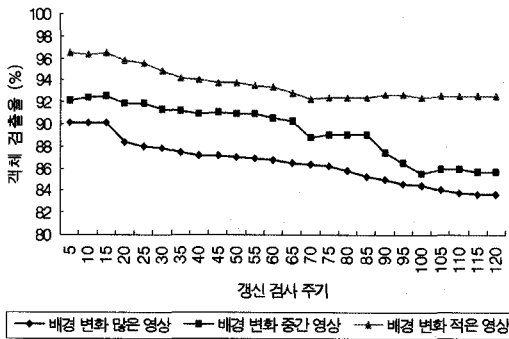


(a) 객체 검출율 결과 (RCO)

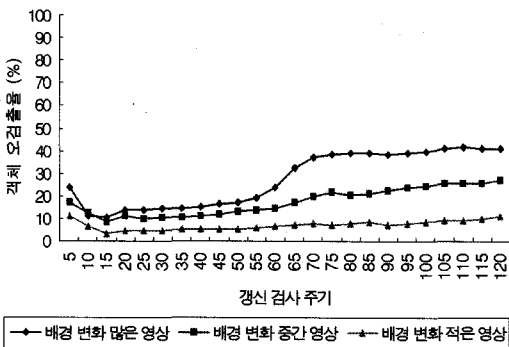


(b) 객체 오검출율 결과 (REO)

그림 12 그룹핑 크기에 따른 객체 검출율과 객체 오검출율 (큐 크기가 120으로 결정된 상태에서 실험)



(a) 객체 검출률 결과 (RCO)



(b) 객체 오검출률 결과 (REO)

그림 13 갱신검사 주기에 따른 객체 검출률, 객체 오검출률 (큐 크기는 120, 그룹핑 크기는 3G로 결정된 상태에서 실험)

그리고 갱신 주기가 60개 프레임 이상이면 객체 오검출률이 급격히 증가하였다. 즉, 갱신검사 주기를 길게 잡으면 배경 변화를 반영해야 하는 배경 모델링 시점을 놓치는 문제가 생긴다. 이 문제로 인해 객체를 배경의 일부분으로 인식하는 오류가 발생하거나 배경 갱신 주기 동안 나타났다가 사라진 객체를 인식하지 못하는 오류가 발생한다. 반면 배경 갱신검사 주기를 너무 짧게(15개 프레임 이하) 잡으면 움직이는 속도가 느린 객체가 배경에 포함되는 잘못된 배경 영상 생성으로 인하여 객체 인식률이 떨어지는 결과를 초래한다. 한편 갱신검사 주기가 80~85일 때 배경 변화가 중간 정도인 영상은 객체 검출률이 단조 감소하지 않고 약간 커지는 것을 볼 수 있다. 이것은 객체의 변화 주기가 갱신검사 주기와 거의 일치하였기 때문인 것으로 해석되었으며, 객체 오검출률에서도 비슷한 현상을 볼 수 있었다.

그림 14는 각 갱신검사 주기별로 실제 갱신이 필요한 비율을 보여준다. 예를 들어 배경 변화가 많은 영상의 경우 갱신주기가 25개 프레임일 때 갱신율은 약 85%이다. 이것은 100번의 갱신검사 횟수에 대해 실제 갱신은

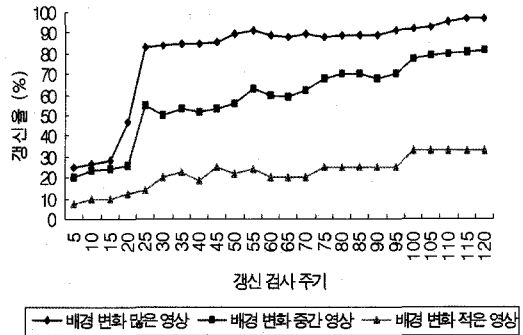


그림 14 갱신검사 주기에 따른 갱신율

85회 일어났다는 것을 의미한다.

갱신율이 낮다는 것은 그 주기 내에 실제 갱신할 횟수가 적다는 것을 의미한다. 반대로 갱신율이 높다는 것은 실제 사용되는 배경 영상의 갱신 시점이 되었음에도 불구하고 갱신검사 주기가 너무 길어서 즉시 배경 갱신을 반영하지 못하고 있음을 의미한다. 이러한 경우 배경이 바뀌었지만 예전의 배경 영상을 그대로 사용하기 때문에 객체의 검출률이 떨어지게 된다. 따라서 본 실험에서는 가장 좋은 결과를 보이는 15개 프레임 단위로 수행하도록 갱신검사 주기를 결정하였다.

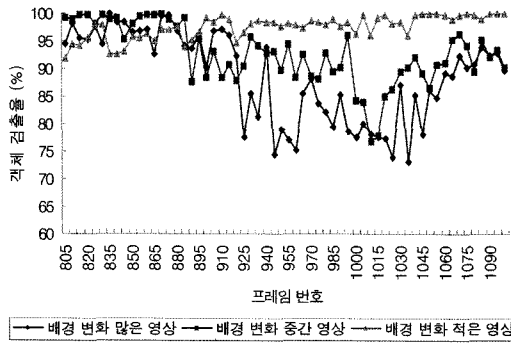
#### 4.2 기존 방법과 성능 비교

4.1절의 실험을 통해 결정된 세 가지 환경 변수를 사용하는 본 논문의 배경 모델링 방법이 기존의 큐 기반 배경 모델링으로 대표되는 Kumar의 방법과 어느 정도의 성능 차이를 가지는지 비교 실험을 하였다. 두 방법의 실험에서 배경 모델링을 위한 초기 프레임의 수는 120개 프레임, 갱신검사 주기는 Kumar의 방법에서는 10개 프레임, 제안한 방법에서는 실험을 통한 결과 값인 15개 프레임을 사용하였다. Kumar의 방법은 배경 모델링 이후 무조건 10개 프레임 주기로 배경 갱신을 수행한다. 그러나 제안한 방법은 15개 프레임마다 MSE를 측정하여 배경 갱신 여부를 결정함으로써 배경의 변화가 거의 없으면 배경 갱신을 수행하지 않는다.

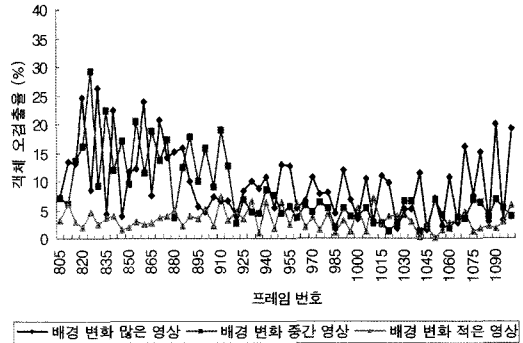
그림 15의 (a)와 (b)는 Kumar가 제안한 방법에 따른 객체 검출률과 객체 오검출률 결과이며, (c)와 (d)는 본 논문에서 제안한 방법에 따른 객체 검출률과 객체 오검출률의 실험 결과이다. 그림 15의 (e)와 (f)는 실험에서 사용한 전체 프레임 1,100개에 대한 객체 검출률과 객체 오검출률의 평균을 보여준다.

그림 15의 (e)와 (f)를 통해 제안한 방법이 Kumar의 방법에 비해 평균 객체 검출률에서는 큰 차이가 나지 않지만, 평균 객체 오검출률에서는 기존의 방법보다 우수한 것을 알 수 있다. 따라서 제안한 방법은 기존의 Kumar의 방법에 비해 5 프레임 긴 갱신검사 주기를 사

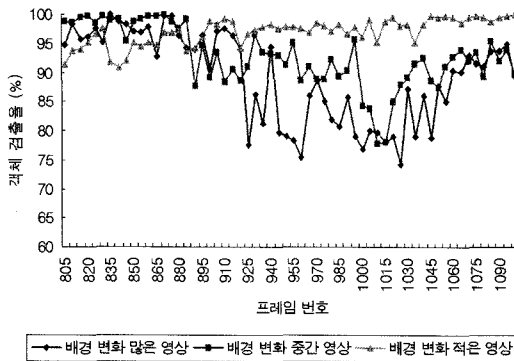




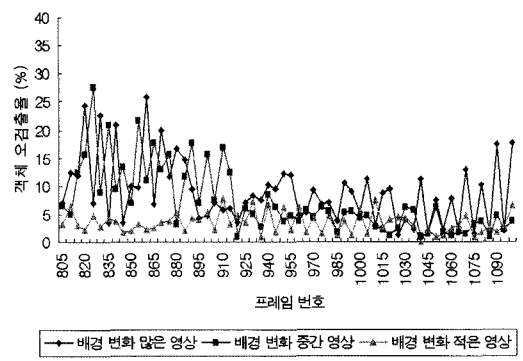
(a) Kumar가 제안한 방법에 따른 객체 검출율 결과



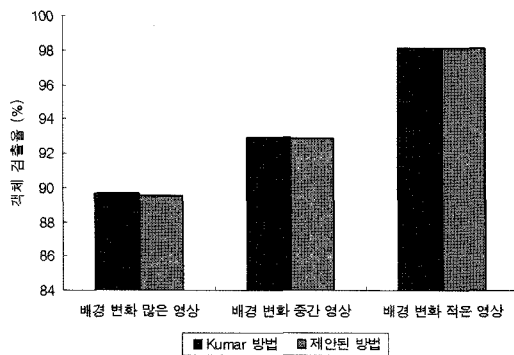
(b) Kumar가 제안한 방법에 따른 객체 오검출율 결과



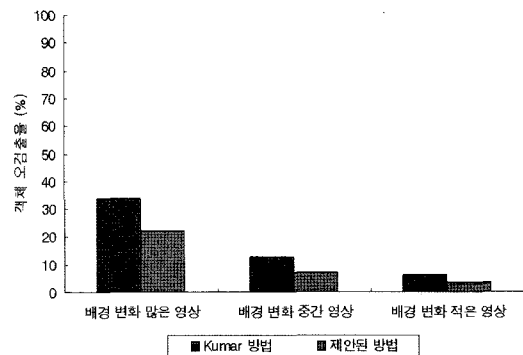
(c) 제안한 방법에 따른 객체 검출율 결과



(d) 제안한 방법에 따른 객체 오검출율 결과



(e) Kumar의 방법과 제안된 방법의 평균 객체 검출율



(f) Kumar의 방법과 제안된 방법의 평균 객체 오검출율

그림 15 배경 모델링을 통한 객체 인식의 기존 시스템과의 성능 비교

용하면서도 비슷한 객체 검출율과, 우수한 객체 오검출율을 얻고 있으므로 실시간에 보다 적합하다고 할 수 있다.

### 5. 결론

제한된 성능의 하드웨어를 가진 영상 처리 환경에서 획득된 영상은 원래의 영상보다 낮은 해상도로 열악한

품질을 가지는 한계를 보이고 있다. 이러한 문제점은 그 다음 단계에서 진행될 후속 과정들을 올바르게 처리할 수 없는 원인이 되고 있다. 본 논문에서는 이러한 문제점을 해결하기 위한 방법으로 객체 인식을 위한 적응적 환경 변수인 큐 크기와 그룹핑 크기, 갱신검사 주기 결정 방법을 제안하였다.

먼저, 다양한 환경의 입력 영상과 적용된 시스템에 실

시간으로 대응하기 위해 배경 모델링 과정에서 세 가지 환경 변수를 결정하여 객체를 추출하는데 적용시켰다. 다음으로 실험을 통해 제안한 방법의 성능을 기존 방식과 비교하여 분석하였다. 그 결과 세 가지 환경 변수는 입력 영상의 환경 또는 적용된 시스템의 메모리 용량 및 속도 등의 환경에 영향을 받아 적응적으로 결정될 수 있었다.

실험은 환경 변수 결정, 배경 모델링 및 배경 갱신 과정을 대상으로 제안 방법의 적합성 및 성능 분석을 목적으로 수행하였다. 먼저 본 논문에서 제안한 큐의 크기, 그룹핑 크기, 갱신검사 주기가 주어진 실험 환경에서 적합한 값을 실험을 통해 제시하였다. 배경 모델링 방법으로 본 논문에서도 적용하고 있는 큐 기반의 대표적 방법인 Kumar가 제안한 방법과 비교하였다. 실험 결과에서는 본 논문에서 제안한 배경 모델링 방법이 Kumar가 제안한 방법에 비해 객체 오검출율을 최대 50%까지 감소시켰다. 따라서 실험에서 제안한 방법이 배경 영상의 갱신 주기와 객체 정보의 정확성 측면에서 기존의 방법보다 우수하다는 것을 확인할 수 있다.

따라서 객체 인식을 위한 적응적 환경 변수의 결정으로 다양한 환경, 적용된 시스템 속에서 객체를 정확히 추출하여 인식할 수 있다. 본 연구 결과는 고화질 저용량의 실시간 처리가 가능하므로 임베디드 시스템에 적용시킬 수 있는 적합한 방법이다. 또한 본 연구는 보다 정확하고 강력한 객체 인식 방법에 대한 추가적인 연구를 통해서 실시간 감시 시스템 등의 응용 분야로 확대 적용할 수 있을 것으로 기대한다.

## 참 고 문 헌

- [1] N. Haering and N. D. V. Lobo, "Visual Event Detection," Kluwer Academic Publishers, 2001.
- [2] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," IEEE Transaction on Systems, Man and Cybernetics, Part C, Vol. 34, Issue 3, pp. 334-352, 2004.
- [3] L. Wang, W. Hu, and T. Tau, "Recent developments in human analysis," Pattern Recognition, Vol. 36, No. 3, pp. 585-601, 2003.
- [4] T. B. Moeslund and E. Granum, "A survey of computer vision-based human motion capture," Computer Vision and Image Understand, Vol. 81, No. 3, pp. 231-268, 2001.
- [5] J. K. Aggarwal and Q. Cai, "Human motion analysis: a review," Proceeding of IEEE Workshop on Motion of Non-Rigid and Articulated Objects, pp. 90-102, 1997.
- [6] I. Haritaoglu, D. Harwood, and L. S. Davis, "W4: Real-Time Surveillance of People and Their Activities," IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 22, No. 8, pp. 809-830, 2000.
- [7] A. H. S. Lai and N. H. C. Yung, "A Fast and Accurate Scoreboard Algorithm for Estimating Stationary Backgrounds in an Image Sequences," International Symposium on Circuits and Systems, Vol. 4, pp. 241-244, 1998.
- [8] P. Kumar, S. Ranganath, and W. Huang, "Queue based fast background modelling and fast hysteresis thresholding for better foreground segmentation," The 2003 Joint Conference of the Fourth ICICS and PCM, Vol. 2, pp. 15-18, 2003.
- [9] D. Gutchess, M. Trajkovic, E. Cohen-Solal, D. Lyons, and A. K. Jain, "A Background Model Initialization Algorithm for Video Surveillance," International Conference on Computer Vision 2001, Vol. 1, pp. 733-740, 2001.
- [10] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," International Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 23-25, 1999.
- [11] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: realtime tracking of the human body," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, pp. 780-785, 1997.
- [12] I. Haritaoglu, D. Harwood, and L. S. Davis, "A Fast Background Scene Modeling and Maintenance for Outdoor Surveillance," International Conference on Pattern Recognition, Vol. 4, pp. 3-7, 2000.
- [13] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric Model for Background Subtraction," Lecture Notes in Computer Science, Vol. 1843, pp. 751-767, 2000.
- [14] S. Hongeng, R. Nevatia, and F. Bremond, "Video-Based Event Recognition: Activity Representation and Probabilistic Recognition Methods," Computer Vision and Image Understanding 96, pp. 129-162, 2004.
- [15] L. Z. Manor and M. Irani, "Event-Based Analysis of Video," IEEE Conference on Computer Vision and Pattern Recognition, 2001.



이 준 철

1988년 경북대학교 자연과학대학 통계학과 졸업. 1991년 경북대학교 대학원 컴퓨터공학과 석사. 1994년 경북대학교 대학원 컴퓨터공학과 박사 수료. 1995년~현재 부천대학 인터넷과 부교수. 관심분야는 감시 시스템, 멀티미디어, 다시점

동영상 등



류 상 료

1988년 경북대학교 공과대학 전자공학과 졸업. 1990년 경북대학교 대학원 컴퓨터공학과 석사. 1997년 경북대학교 대학원 컴퓨터공학과 박사. 1998년~현재 청운대학교 컴퓨터학과 부교수. 관심분야는 알고리즘, 멀티미디어 등



강 성 환

2006년 계명대학교 정보통신대학 컴퓨터공학과 졸업. 2006년~현재 경북대학교 공과대학 컴퓨터공학과 석사과정. 관심분야는 멀티미디어, 다시점 동영상, 감시 시스템 등



김 승 호

1981년 경북대학교 전자공학과 졸업  
1983년 한국과학기술원 전산학과 석사  
1994년 한국과학기술원 전산학과 박사  
1985년~현재 경북대학교 컴퓨터공학과 정교수. 관심분야는 알고리즘, 멀티미디어, 다시점 동영상, 감시 시스템 등