

논문 2004-41CI-4-10

객체 추출 및 추적을 이용한 실시간 웹기반 영상감시 시스템

(Web-based Video Monitoring System on Real Time using Object Extraction and Tracking out)

박 재 표*, 이 광 형**, 이 종 희*, 전 문 석***

(Jae-pyo Park, Kwang-hyoung Lee, Jong-hee Lee, and Moon-Seog Jun)

요 약

실시간 영상에서 객체 추적은 수년간 컴퓨터 비전 및 여러 실용적 응용 분야에서 관심을 가지는 주제 중 하나이다. 하지만 배경영상의 잡음을 객체로 인식하는 오류로 인하여 추출하고자 하는 객체를 찾지 못하는 경우가 있다. 본 논문에서는 실시간 영상에서 적응적 배경영상을 이용하여 객체를 추출하고 추적하는 방법을 제안한다. 입력되는 영상에서 배경영역의 잡음을 제거하고 조명에 강인한 객체 추출을 위하여 객체영역이 아닌 배경영역 부분을 실시간으로 갱신함으로써 적응적 배경영상을 생성한다. 그리고 배경영상과 카메라로부터 입력되는 입력영상과의 차를 이용하여 객체를 추출한다. 추출된 객체의 내부점을 이용하여 최소사각영역을 설정하고, 이를 통해 객체를 추적한다. 아울러 제안방법의 성능에 대한 실험결과를 기존 추적알고리즘과 비교, 분석하여 평가한다.

Abstract

Object tracking in a real time image is one of interesting subjects in computer vision and many practical application fields during the past couple of years. But sometimes existing systems cannot find all objects by recognizing background noise as object. This paper proposes a method of object detection and tracking using adaptive background image in real time. To detect object which is not influenced by illumination and to remove noise in background image, this system generates adaptive background image by real time background image updating. This system detects object using the difference between background image and input image from camera. After setting up Minimum Bounding Rectangle(MBR) using the internal point of detected object, the system tracks object through this MBR. In addition, this paper evaluates the test result about performance of proposed method as compared with existing tracking algorithm.

Keywords : 실시간 영상 감시(real time Video Monitoring), 객체 추적(object tracking out),
객체 추출(object extraction)

I. 서 론

실시간 영상에서 객체 추적을 위해 컴퓨터 비전 기술을 이용하여 구현하기란 매우 어려운 일임에도 불구하고, 컴퓨터 성능의 발달로 인해 영상 처리 기법의 발전과 더불어 객체 인식과 객체 추적에 관한 연구가 활발

히 진행되고 있다. 객체 추적은 카메라로부터 입력된 영상에서 움직임을 보이는 객체를 인식하고, 그 움직임을 추정하여 추적하는 것이다. 객체를 추적하는 방법은 보안, 의료, 군사, 교통, 제어분야 등 여러 분야에 응용될 수 있어 그동안 많은 연구와 개발이 이루어지고 있다^[1,5]. 하지만, 기존의 실시간 객체 추적 시스템의 구현에는 많은 어려움이 존재한다. 우선 강인한 실시간 객체 추적 시스템을 구현하기 위하여 고가의 장비를 요구한다. 움직임을 감지하기 위한 센서를 장착한 카메라나 pan/tilt로 움직이는 카메라 등의 요구이다. 또한 객체 추출이나 추적에 많은 연산을 필요로 하는 알고리즘 등을 적용하여 객체 추출에는 뛰어난 성능을 보이지만 실

* 정회원 국립 한경대학교 컴퓨터공학과
(Department of Computer Engineering, Hankyong National University)

** 학생회원, *** 정회원 숭실대학교 컴퓨터학과
(Department of Computing, Soongsil University)

접수일자: 2003년11월3일, 수정완료일: 2004년5월25일

시간 객체 추적에는 부적합하다.

기존의 객체 추출 및 인식 기법에서의 객체 추적과 얼굴 추출에 우수한 성능을 보이는 방법은 상대적으로 처리 속도가 떨어짐을 볼 수 있다. 반대로 처리 속도가 빨라 많은 프레임을 처리할 수 있는 방법은 객체 추출과 얼굴 추출에 뛰어난 성능을 보이지 못하고 있다^[2,6].

본 논문에서는 보안 및 감시 시스템 분야에서 적용되어 질 수 있는 방법을 제안하기 위하여, 저가형 PC카메라의 움직임이 고정되어 있고 외부 환경이 아닌 실내 환경이며 배경영상의 변화가 거의 없다는 특수 환경으로 제약 조건을 가진다.

제안하는 시스템은 시간의 경과에 따라 배경영역에서 잡음생성을 줄이고자 객체영역 이외의 영역을 배경영역으로 갱신하여 항상 최신의 배경영상을 유지하도록 하였다. 이와 같이 생성된 적응적 배경영상과 설치된 PC카메라로부터 실시간으로 입력되는 입력영상의 차를 이용하여 객체의 크기와 위치를 탐지함으로써 객체를 추출한다. 이때 연산의 양을 줄이기 위하여 그물식 탐색방법을 이용한다. 그리고 추출된 객체의 내부점들에 MBR(Minimum Bounding Rectangle)을 설정하여 객체를 실시간으로 추적한다. 또한 RGB를 정규화하고, 정규화된 RGB 색상정보를 이용하여 MBR내의 객체로부터 얼굴영역을 추출한다.

본 논문은 실시간으로 입력되는 영상으로부터 적응적 배경영상으로 잡음을 현저히 줄였고 실시간으로 얼굴 영역을 추출함으로써 보안 및 감시 시스템으로 효용성을 향상시켰다.

II. 관련 연구

1. 기존 연구

가. 저 수준 해석(Low-Level Analysis)

저수준 해석은 외곽선이나 모서리에 해당하는 에지(edges) 정보, 명암으로 객체를 구분할 수 있는 그레이(gray) 정보, RGB 색상값을 이용하는 컬러(color) 정보, 비디오 시퀀스를 사용하여 움직이는 객체들의 위치를 파악할 수 있는 움직임(motion) 등이 객체에서 객체를 인식하고 추적하는데 많이 이용되고 있다.

또한 얼굴과 신체 부분들을 포함하는 움직임은 실무 애플들이 누적된 프레임 차이를 임계치와 함으로써 추출되거나, 얼굴영역 이외에도 얼굴 특징들의 위치를 결정하기 위해서 프레임 차이를 채택하는 기법들이 시도되었다^[1,3,7].

나. 특징 해석(Feature Analysis)

저수준 해석으로부터 생성되는 특징들은 애매하기 십상이다. 예를 들어서 컬러 모델을 사용하여 객체 영역들의 위치를 결정하는 것에서 유사한 컬러의 배경 객체들 또한 추출될 수 있다. 이것은 더 높은 수준의 특징 해석에 의해서 해결될 수 있는 고전적인 다대일 정합문제이다. 많은 객체추출 기법들에서 객체형태의 지식은 그것들의 애매한 상태에서부터 다양한 특징들을 특성화하고 이어서 입증하기 위해서 채택되었다^[2,4,7].

다. 능동 형상 모델(Active Shape Models)

능동 형상 모델들은 특징들의 실제적인 물리적 및 더 높은 수준의 외관을 묘사한다^[3,8,9]. 일단 특징에 대해 가까운 근접 내에서 방출되면 능동 형상 모델은 국소 이미지 특징들(모서리들, 밝기)과 상호작용하고 점차적으로 특징의 형상을 얻기 위해서 변형한다^[5].

라. 선형 부분공간 방법(Linear Subspace Methods)

1980년에 사람 얼굴들을 효과적으로 표현하기 위해서 PCA를 사용하는 기법들을 개발하였다^[6]. 다른 얼굴 이미지들의 조합이 주어지면 고유벡터들에 의하여 표현되는, 얼굴들의 분포의 주성분들을 찾는다. 그 후에 얼굴집합에 있는 각 개별 얼굴이 적절한 가중치들을 사용하여 더 일반적으로는 고유 얼굴들로 나타내지고 가장 큰 고유벡터들의 선형조합에 의해서 근사화될 수 있다^[10,11]. 최근에는 얼굴인식을 위하여 이 기법을 발전시켰다. "얼굴공간으로부터의 거리(DFFS: distance from face space)"는 거리지도에서 전역 최소값의 관찰을 통해서 얼굴 존재성의 좋은 지표를 제공한다^[12]. Pentland은 훈련 집합에 있는 다양한 안면 특징 템플릿들로부터 얻어진 고유특징들로부터 생성되는 DFFS를 사용하는 얼굴 특징 추출기를 제안하였다^[13].

2. 기존 연구 분석

기존의 많은 방법들이 객체 추적 및 얼굴 추출을 위하여 제안되어 왔다. 발표된 논문의 결과와 구현을 통한 각각의 방법들을 비교한 결과가 <표 1>에 나타나 있다. 객체 추적과 얼굴 추출에 우수한 성능을 보이는 방법은 상대적으로 처리 속도가 떨어짐을 볼 수 있다. 반대로 처리 속도가 빨라 많은 프레임을 처리할 수 있는 방법은 객체 추출과 얼굴 추출에 뛰어난 성능을 보이지 못하고 있다.

표 1. 기존 알고리즘 성능 분석

Table 1. efficiency Analysis for Existing Algorithm.

구 분	FPS	Detection rate	Tracking rate
저수준 해석	◎	○	○
특징 해석	○	△	△
능동 형상 모델	○	◎	○
선형 부분 공간 방법	△	◎	○
신경망	△	◎	○
통계적 방법	△	○	◎

(◎ : 우수, ○ : 양호, △ : 보통)

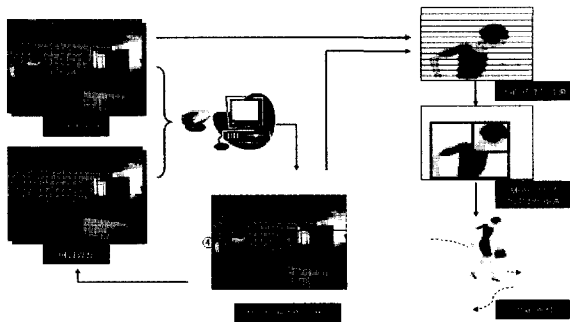


그림 1. 시스템 처리 흐름도

Fig. 1. System Processing Flowchart.

III. 실시간 객체 추적 및 얼굴영역 추출

1. 시스템 구조

본 논문에서도 효율적인 실시간 객체 추적 및 얼굴영역 추출을 위하여 다음과 같은 제약 조건을 가진다. 센서 장착이나 움직임이 가능한 고가의 카메라가 아닌 저가형 PC카메라를 이용하는데 이는 카메라의 움직임이 고정됨을 의미한다. 또한 공원이거나 거리 등의 야외 환경이 아닌 사무실이나 연구실, 기자재실, 자재창고, 주차장, 은행 365일 코너 등 배경영상의 변화가 미미한 특수 환경으로 환경을 제한한다.

그림 1은 제안하는 방법의 전체적인 처리 흐름도이다. PC카메라로 실시간 받아들여지는 영상 중에서 객체의 출현이 감지되지 않는 영상을 초기의 배경영상으로 설정하여 준다. 이 후, 연속적으로 입력되어지는 영상과 초기 배경영상의 비교를 통하여 강인한 객체 추출을 할 수 있는 적응적 배경영상을 생성한다. 계속적으로 갱신을 반복하는 배경영상과 실시간 입력되는 영상에서 그물식 탐색 방법을 이용하여 객체의 출현을 감지하고, 감지된 객체의 내부점을 추출한다. 추출된 객체 내부점을 이용하여 객체를 포함하는 최소 사각형인 MBR을 설정해 주고, 연속되어지는 영상으로부터 일련의 처리

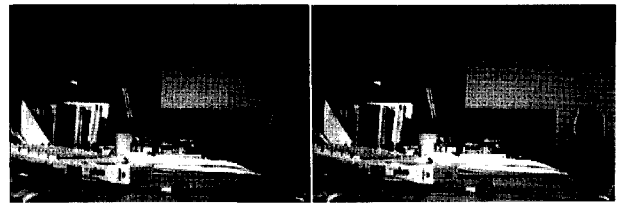


그림 2. 초기 배경영상

그림 3. 180초 후 배경영상

Fig. 2. Default Background.

Fig. 3. Background After 3 Min.

를 반복하게 된다. 연속된 MBR의 설정을 추적함으로써 객체의 실시간 추적을 가능토록 하였다. 추적과 동시에 전체 영상이 아닌 MBR 내에서 얼굴 영역을 추출하는 방법을 사용하여 실시간 얼굴영역을 추출한다.

2. 시스템 구현 방법

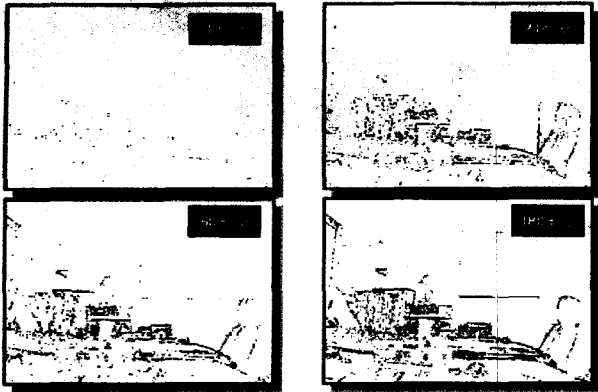
(1) 적응적 배경 영상 생성

그림 2와 그림 3은 PC카메라로부터 입력되어진 초기 설정된 배경영상과 객체의 움직임이 감지되지 않은 상태에서 180초 후의 배경영상이다. 사람의 감지 능력으로는 두 그림이 똑같아 보이지만, 시스템에서는 둘을 다르게 인식한다.

그림 4가 시스템이 이 두 그림을 다르게 인식함을 보이는데, 어떠한 변화도 없는 배경영상이 조명의 영향 등으로 시간의 경과에 따른 잡음 발생의 현상을 보인다. 본 시스템에서 사용되어지는 카메라는 저가이며 고성능이 아닌 일반 PC카메라이다. 저가의 시스템 구축을 위한 선택이었는데 고성능의 카메라가 아니기 때문에 빛에 대해 상당히 민감한 반응을 보인다. 1초, 20초, 60초, 180초 후 영상에서 각각 71개, 1657개, 2061개, 3891개의 최초 배경영상과의 차에서 임계값을 넘어가는 픽셀의 개수가 나타난다. 배경영상과 입력영상의 차이가 균집을 이루고 있으면 배경영상임에도 불구하고 새로운 객체로 인식하는 오류를 범할 수 있다. 여기서 우리는 잡음을 제거하여 보다 정확한 객체의 추출을 위해 적응적 배경영상을 생성한다.

전체적인 배경영상의 갱신은 많은 연산량을 필요로 하기 때문에 실시간 객체 추적 성능을 저하시킨다. 제안하는 시스템에서는 객체영역인 $n * m$ 의 영역을 제외한 나머지 영역을 배경영상으로 갱신하면서 객체 추적을 동시에 이루어질 수 있도록 하였다.

그림 5는 적응적 배경영상 생성 방법을 보인다. 이전의 배경영상과 입력영상에서 객체 추출 후 객체영역을 제외한 나머지 영역을 새로운 배경영상으로 대체한다. 그림 5는 객체의 위치에 따른 취할 수 있는 배경영상



[초당 10프레임 처리시]

그림 4. 시변하는 배경영상에서의 잡음
Fig. 4. Noise of Changeable Background.

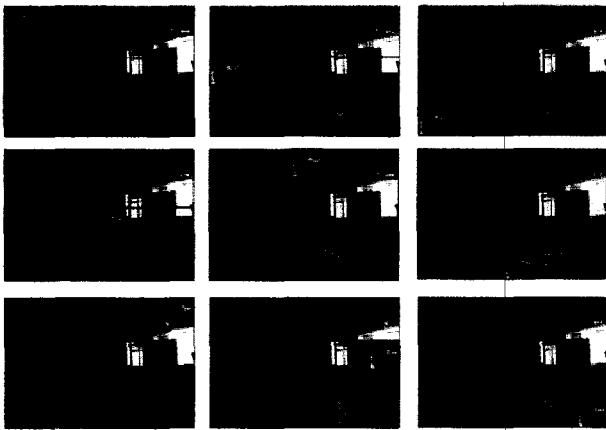


그림 5. 적응적 배경영상 생성
Fig. 5. Creation of Adaptive Background.

의 집합이다. 입력영상에서 객체를 찾아내고 객체의 위치에 따라 객체부분이 아닌 사각영역을 배경영상으로 판단한다. 즉 객체의 위치에 따라 ①~④ 부분으로 분류하고 입력영상을 배경영상으로 대체한다.

제안한 적응적 배경영상 방법을 이용하여 실험한 결과, 시변하는 배경영상내의 조명 등에 의한 잡음을 객체 내부에 포함되어 있는 배경영역을 제외한 나머지 영역에서 95% 이상 제거할 수 있었다. 객체내부에 포함되어 있는 배경영역 또한 객체의 움직임에 따라 바로 배경영상으로 적용되기 때문에 배경영상 내의 잡음을 객체로 인식하는 오류를 범하지 않는다.

(2) 객체의 내부점 추출

객체의 추적을 위해서 객체의 위치 결정을 위한 객체 추출이 우선적이다. 완전 차영상 방식을 도입한 기존 시스템과는 달리, 제안하는 시스템에서는 그물식 탐색 방식을 이용한다. 그물식 탐색 방식은 영상을 위에서 아래로 δ 픽셀 간격으로 탐색해 나가는 방법으로 완전

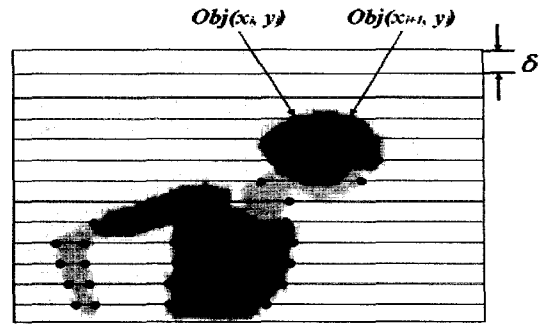


그림 6. 그물식 탐색 방식
Fig. 6. Line Scanning Method.

차영상을 이용하는 시스템보다 처리속도의 높은 향상을 보인다.

그림 6은 그물식 탐색 방식 및 객체 내부점 추출이다. 객체의 내부점 추출은 그물식 탐색을 하면서 배경영상과 입력영상의 차영상을 이용하게 된다. 배경영상과 입력영상의 탐색라인상의 픽셀들을 각각 RGB채널로 분할한다. 식 (1)~(4)에 따라 RGB채널 각각의 차이가 임계값 β 보다 크다면, 객체 내부점 후보군으로 등록하게 된다. 제안하고자 하는 시스템은 보안 및 감시 시스템에 적용되어지기 때문에 사람이라는 객체 이외의 빛의 간섭 등으로 인한 배경에서의 작은 잡음 제거를 위해, 식 (5)를 이용하게 된다. 또한 객체의 일부분이지만 효율적인 잡음 제거를 위하여 어느 정도의 미세한 객체의 일부분은 무시를 한다. 식 (1)~(4)를 만족하는 객체 내부점 후보군에서 연속된 픽셀이 ν 이하라면 잡음으로 판단하여 제거하고, 원하는 객체 내부점 $Obj(x_i, y_i)$ 를 추출하게 된다.

* $Obj(x_i, y_i)$: 객체 내부점

* $Obj_left(top, right, bottom)$: MBR의 최대/최소 좌표

$$Obj_left = \min [Obj(x_i)] - \delta \tag{1}$$

$$Obj_top = \min [Obj(y_i)] - \delta \tag{2}$$

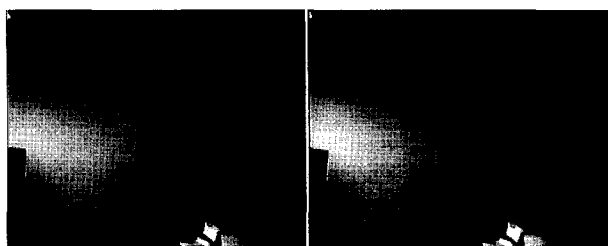
$$Obj_right = \max [Obj(x_i)] + \delta \tag{3}$$

$$Obj_bottom = \max [Obj(y_i)] + \delta \tag{4}$$

$$MBR = [Obj_left, Obj_top, Obj_right, Obj_bottom] \tag{5}$$

그림 7은 계속적으로 갱신을 반복하는 적응적 배경영상과 실시간 입력되어지는 영상을 나타내며, 그림 8은 배경영상과 입력영상에서 그물식 탐색 방식을 이용한 객체 내부점 추출을 나타낸다.

그물식 탐색 방식을 이용하여 객체 내부점을 추출한



(a) 배경영상 (a) Background Image. (b) 입력영상 (b) Input Image.

그림 7. 배경영상과 입력영상
Fig. 7. Background Image and Input Image.



그림 8. 객체 내부점 추출
Fig. 8. Object Inter-point Extraction.

것이 완전 차영상을 이용하여 객체를 추출하는 것에 비하여 강건한 객체 추출을 보이지는 않지만, 실시간 보안 및 감시 시스템에서 적용되어질 제안하는 시스템에서는 객체 추적의 대상인 사람의 추출을 빠르게 하는 목적에는 적합함을 볼 수 있고, 기존 시스템과 비교하여 객체 추출의 속도도 향상됨을 알 수 있다.

(3) MBR을 이용한 객체 추적

추출된 객체 내부점을 이용하여 MBR(Minimum Bounding Rectangle)을 설정하여 준다. MBR은 객체 추적을 위한 객체를 포함하는 최소 사각 영역으로써, 얼굴 영역을 추출할 때 속도를 향상시키기 위한 범위를 제한해 주는 역할을 수행한다.

보다 정확한 MBR의 설정을 위한 전처리 과정으로 Median Filtering을 통해 잡음을 제거하였다. 제안하는 시스템에서는 완전 차영상이 아닌 그물식 탐색 방법을 이용하기 때문에 일반적인 $n * n$ block median filter window를 사용하는 것이 아니라, horizontal median filter window를 이용한다. 또한 컬러 영상에 적용되는 median filtering을 위한 절차를 따른다.

컬러 영상을 필터링 할 때 미디언 값이 가지는 유일한 특징을 이용한다. 미디언 값과 집합에서 모든 다른 값들 사이의 차를 합한 것은 집합에서 임의의 다른 값



그림 9. 설정된 MBR
Fig. 9. Processed MBR.

에 대한 차를 합한 것보다 작다는 것이다. 식 (6)과 같이 각 화소 샘플에 대하여 각각의 RGB 채널에 대한 차들을 합한다. 이중 가장 작은 값을 가진 $Distance_i$ 는 필터의 출력인 x_i 에 대응된다.

- * $Distance_i$: 각 컬러 요소들에 대한 차
- * N : 필터 윈도우에 표현되는 샘플수
- * i : 처리되는 화소
- * j : 다른 화소 샘플

$x_{med} = x_i$ (단, i 는 $Distance_i$ 중 최소값에 대응하는 값)

$$\sum_{i=1}^N |x_{med} - x_i| \leq \sum_{i=1}^N |y - x_i| \quad (6)$$

$$Distance_i = \sum_{j=1}^N (|red_i - red_j| + |green_i - green_j| + |blue_i - blue_j|) \quad (7)$$

객체 내부점 $Obj(x_i, y_i)$ 의 x, y 좌표 중 식 (7)을 통하여 각각의 최대, 최소 좌표를 구하여 객체를 포함하는 최소한의 사각 영역을 설정해 준다. 그림 9는 추출된 객체 내부점을 이용하여 설정된 MBR을 보인다. 이를 통하여 객체의 실시간 추적을 가능하도록 하였다.

(4) 얼굴 영역 추출

HSI 색상계는 색상(hue)과 채도(saturation) 만을 사용함으로써 조명의 영향을 흡수할 수 있다. HSI 색상계는 인간 시각 시스템의 색채 감지 특성에 기초한 영상 처리 알고리즘 개발을 위한 이상적인 도구가 되게 한다. HSI 색상계의 변환에서 가장 복잡한 부분은 색상(hue)의 계산 부분이다. 색상은 RGB의 혼합 비율을 빨강을 기준으로 하는 일반전각으로 표현한다.

얼굴 영역 추출은 식 (8)~(10)에 따라 HSI 좌표계로 변환된 영상의 색상정보에서 피부영역으로 알려진 범위 값에서 추출된 부분을 군집화하여 선택한다.

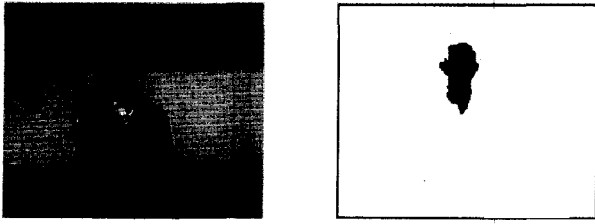


그림 10. HSI 색상계를 이용한 얼굴 영역 추출
Fig. 10. Face region Extraction using HSI.

$$H = \cos^{-1} \left[\frac{\frac{1}{2} [(R-G) + (R-B)]}{\sqrt{(R-G)^2 + (R-B)(G-B)}} \right] \quad (8)$$

$$S = 1 - \frac{3}{(R+G+B)} [\min(R, G, B)] \quad (9)$$

$$I = \frac{1}{3} (R+G+B) \quad (10)$$

색상을 통한 추출 방법은 수행속도가 빠르다는 장점이 있지만 조명을 통해 영상의 변화가 많이 일어나며 명도에 대한 정규화 과정이 필요하다. 이에 HSI 색상과 표계는 색상과 채도 값만을 사용함으로 인해 조명의 영향을 감소시킬 수 있는 장점이 있다. HSI 모델 실험에서 피부색의 범위를 다음과 같이 사용하였다.

$$H_{\min} = 0^\circ, \quad H_{\max} = 50^\circ$$

$$S_{\min} = 0.23, \quad S_{\max} = 0.68$$

카메라를 통해 실시간으로 입력되는 영상의 픽셀당 RGB값을 HSI 좌표계로 변환한다. 변환값에서 피부색 값에 유사한 H, S 범위의 값을 임계값으로 설정하여 벗어나는 경우는 제거한다. 이렇게 추출된 값들 중 이웃하는 픽셀들을 연결하여 군집화하고 가장 큰 군집영역을 1차 얼굴 영역으로 선택한다. 그러나 이 방법은 작은 영역에서도 다양한 군집을 나타내는 문제점을 가지고 있다. 이러한 단점을 해결하기 위해 median filtering 방법을 이용하여 잡음을 제거하여 필요영역만을 군집화하게 된다. 그림 10은 입력된 영상에서 HSI 색상계를 이용한 얼굴 영역 추출 결과를 보이고 있다.

HSI를 이용한 추출 방법은 조명에 덜 영향을 받으면서 얼굴 영역을 추출하는 장점이 있지만, 어두운 영역에서는 얼굴 영역의 추출에 단점을 가지고 있다. 이에 정규화된 RGB영역을 이용하여 추출할 수 있다.

정규화된 RGB를 이용한 방법은 정규화된 값을 사용하여 HSI와 유사한 효과를 거뒀다. 이때 $r'+g'+b'=1$ 이라는 성질을 이용하여 r', b' 의 두 채널 정보만을 사용하여 색을 구분하는 기준으로 사용할 수 있다. RGB 좌표계에서 색상의 정규화 식은 식 (11)과 같다.

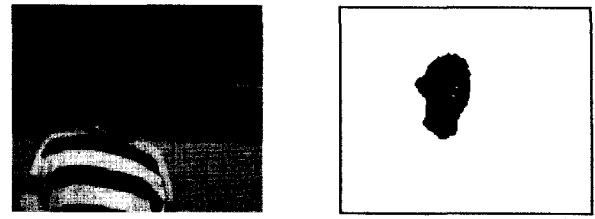


그림 11. 정규화된 RGB를 이용한 얼굴 영역 추출
Fig. 11. Face region Extraction using Normalized RGB.

$$(r', g', b') = \left(\frac{r}{r+g+b}, \frac{g}{r+g+b}, \frac{b}{r+g+b} \right) \quad (11)$$

이 방법은 R, B의 값만을 이용하여 얼굴 영역 추출에 이용할 수 있다. 정규화된 RGB 모델 실험에서 피부색의 범위를 다음과 같이 사용하였다.

$$r'_{\min} = 0.3550, \quad r'_{\max} = 0.4300$$

$$b'_{\min} = 0.2500, \quad b'_{\max} = 0.3050$$

이 방식은 어두운 영역에서 후보 추출에 장점이 있으나 유사 색상영역에서는 구별이 어렵다. 이에 본 논문에서는 두 가지 방법을 혼합한 hybrid 방법론을 이용하였다. 그림 11은 입력된 영상에서 HSI 색상계를 이용한 얼굴 영역 추출 결과를 보이고 있다.

추출된 얼굴 영역을 포함하는 최소 사각 영역(MBRF, Minimum Bounding Rectangle for Face Region)을 설정해 줌으로써 얼굴 영역을 추출한다. 이 영역은 추후 사람 얼굴 인식 분야와의 결합을 통해 보안 및 감시 시스템의 효율성을 높일 수 있다.

IV. 구현 및 실험 평가

1. 실험 환경

본 실험은 카메라의 움직임이 고정되어 있고 배경영상의 변화가 거의 없다는 환경으로 제약(일반 사무실)하고 입력영상을 실시간으로 받아들여 객체의 위치를 탐지하고, 이동 객체의 추적을 실험하였다.

본 논문에서 제안한 방법의 구현을 위해 Delphi 6.0을 이용하였고, Intel(R) Pentium 4 CPU 2.4GHz, 256M RAM의 PC에서 Microsoft Windows 2000 운영체제 하에서 실험하였다. 저가형 PC카메라로부터 받아들이는 배경영상과 입력영상의 크기는 320 × 240의 24bit 컬러영상과 160 × 120의 24bit 컬러영상을 이용하였다.

2. 시스템 구현

그림 12는 구현된 실시간 객체 추적 및 얼굴영역 추

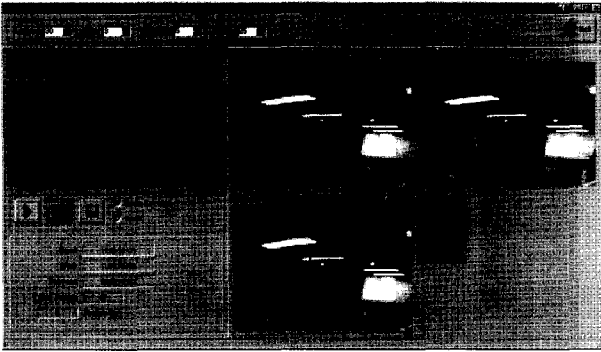


그림 12. 실시간 객체 추적 및 얼굴 영역 추출
Fig. 12. Real Time Object Tracking and Face Region Extraction.

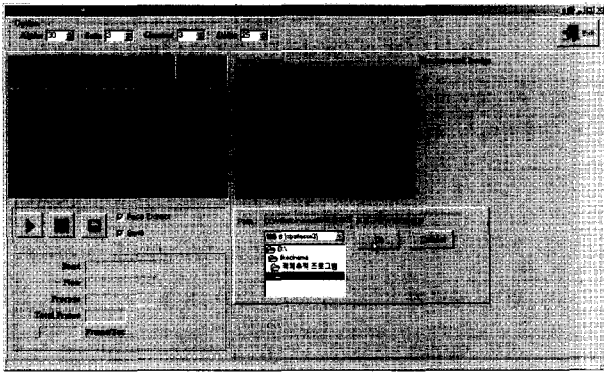


그림 13 실시간 객체 추적과정 저장 화면
Fig. 13. Save of Real Time Object Tracking Process.

출 화면으로 입력받은 동영상에서 객체를 추출하고 MBR을 통한 실시간으로 추적하며, 얼굴 영역 추출 과정을 보인다.

인터페이스의 상단부에는 제안한 시스템에서 사용되어지는 임계값과 변수에 대한 값을 입력하여 환경과 상황에 가장 적합한 객체 추적을 가능하도록 하였으며, 입력되는 동영상의 크기, 해상도 등을 선택적으로 입력되게 하였다. 배경영상 부분은 계속적으로 갱신되고 있는 배경영상을 보이고, 입력영상 부분은 실시간 입력되는 영상을 보여준다. 왼쪽 하단부는 처리되어진 프레임 수와 초당 처리 속도(FPS, Frame Per Second) 등을 나타내 준다. 객체추출 및 추적 부분은 객체를 추출하고 추적하는 모습을 보이는데, 입력된 영상으로부터 객체를 잡음 없이 추출하고 실시간 객체 추적이 가능하다. 얼굴추출 부분은 작은 영상으로부터 선택되어져 얼굴영역을 정확히 추출할 수 있었다.

그림 13은 실시간 객체 추적과정 저장 화면으로 저장 장치에 JPEG 파일형식으로 시간대별 실시간 객체 추적 과정을 저장함으로써 움직임이 추출된 후 객체의 움직임을 추후 검색이 가능하다.

표 2. 임계값과 변수에 대한 최적화 실험 결과
Table 2. Threshold & Value for optimization test.

실험값 \ 임계값	1	2	...	5	...	25	...	30	...
α	0.023	0.151	...	0.210	...	0.915	...	0.887	...
β	0.101	0.256	...	0.494	...	0.856	...	0.934	...
γ	0.495	0.531	...	0.926	...	0.349	...	0.324	...
δ	0.951	0.903	...	0.851	...	0.595	...	0.496	...

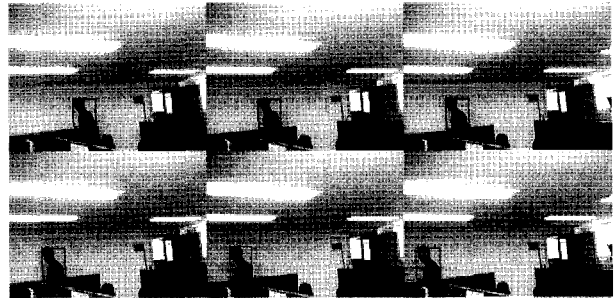


그림 14. MBR을 이용한 실시간 객체 추적(1)
Fig. 14. Real Time Object Tracking using MBR(1).

3. 실험 결과

표 2는 본 논문에서 제안한 방법을 위한 임계값들과 변수에 대한 최적화 실험 결과이다.

- α : 적응적 배경영상 생성 임계값,
- β : 객체 내부점 후보군 추출 임계값
- γ : 객체 내부점 추출 임계값,
- δ : 그물식 탐색을 위한 스캔 간격

표 2에서 보는바와 같이 α 가 25일 때 적응적 배경영상의 생성으로 많은 잡음을 제거할 수 있었고, β 가 30이었을 때 객체를 성공적으로 추출할 수 있었다. γ 가 5일 때에는 추출된 객체 내부점 후보군에서 많은 잡음 성공률을 보였고, δ 는 보다 조밀한 간격으로 탐색할 때 객체의 추출이 잘 이루어지나, 많은 연산량을 요구해 보다 빠른 탐색을 위하여 5를 선택하였다.

그림 14와 그림 15는 입력된 동영상으로부터의 실시간 객체 추적 결과를 보이고 있다. 제안한 방법으로 변화된 장소, 카메라의 위치, 높이, 조명의 변화 등의 여러 상황에서 120여건의 Dataset에서 실험하였을 때 320×240 해상도의 영상은 초당 9~13 프레임을 처리하였으며, 160×120 해상도의 영상은 초당 17~23 프레임의 처리 성능을 보였다.

이는 기존의 객체 추적 방법에 비하여 처리 속도의 향상을 보이며, 제안한 알고리즘을 통하여 실시간 객체 추적 및 얼굴 영역 추출이 가능함을 보인다. 객체 추적



그림 15. MBR을 이용한 실시간 객체 추적(2)
Fig. 15. Real Time Object Tracking using MBR(2).

표 3. 시스템 성능 평가
Table 3. System Performance Test.

구분	FPS	Detection rate	Tracking rate
저수준 해석	15~18	91%	90%
특징해석	11~16	84%	82%
능동형상 모델	12~15	97%	93%
선형 부분공간 방법	5~9	98%	89%
신경망	6~9	96%	91%
통계적 방법	5~8	92%	95%
제안 시스템	17~23	97%	98%

및 얼굴 영역 추출 처리 속도는 추적 대상인 객체의 크기 등에 따라 민감하게 반응하며 객체가 작았을 경우, 각각 최대 11프레임과 20프레임까지도 처리 속도를 보였다.

4. 성능 평가

표 3과 그림 16은 제안한 시스템과 기존 방법들과의 성능 평가이다. FPS는 처리속도 평가기준으로 초당 처리되는 프레임의 수를 나타낸다. 추출율은 하나의 프레임에서 19,200(160×120 해상도) 픽셀 중 실제 객체영역의 픽셀 수에 대한 추출된 객체영역의 픽셀 수를 나타낸다. 또한 추적율은 이전 프레임의 추출된 객체 영역 픽셀의 다음 프레임에서 추적된 성능을 나타낸다. 성능 평가에 사용된 영상은 160×120 해상도의 24 bit 컬러영상을 사용하였다.

표 3과 그림 16에서 보는 바와 같이 초당 처리하는 프레임 수와 객체 추출율, 추적율의 복합적인 면에서 기존의 방식들에 비해 향상됨을 알 수 있다.

또한, 실시간 영상에서 객체의 추출과 추적에 대한 계산량을 평가하기 위하여 실제 실시간 영상으로 시뮬레이션을 하였다. 먼저 객체의 추출을 위해 배경영상을 획득한 후 최초로 움직임이 감지되는 프레임의 객체를 추출하였고, 객체추출의 정확성을 알아보기 위해 몇 프

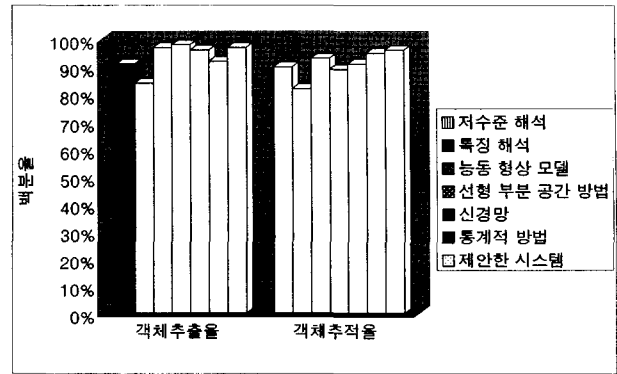


그림 16. 시스템 성능 비교
Fig. 16. Performance Comparison for System.

표 4. 객체추출 시 계산에 참여되는 픽셀 수
Table 4. The Number of calculated Pixel in Object Extraction.

계산에 참여하는 영역		계산	계산량
전체영역에서 객체추출 (기존 시스템)		$320 \times 240 \times 3$	230,400
제안 시스템	첫 번째 영역에 객체가 존재	$320 \times 20 \times 3$	19,200
	두 번째 영역에 객체가 존재	$(320 \times 20 + 240 \times 20) \times 3$	33,600
	세 번째 영역에 객체가 존재	$(320 \times (20 + 20) + 240 \times 20) \times 3$	52,800
	네 번째 영역에 객체가 존재 (최대)	$(320 \times (20 + 20) + 240 \times (20 + 20)) \times 3$	67,200

레이프 후의 객체를 같은 방법으로 추출하였다. 표 4는 입력영상의 크기가 320X240일 때 객체추출영역이 전체일 때와 제안한 시스템을 적용하였을 경우의 연산량을 계산한 것이다.

표 4에서 보여 주고 있는 것과 같이 본 논문에서 제안한 그물식 방법은 최악의 경우라도 일반방법과 계산량의 차이가 아주 많다는 것을 알 수 있다. 최근 하드웨어의 성능 개선에 따라 컴퓨터의 속도가 비약적으로 발전했다고는 하지만 많은 계산량은 실시간으로 영상을 처리 할 경우 전체 시스템의 효율성을 떨어뜨리게 된다. 따라서 본 논문에서 제안한 방법이 효과성에서도 뛰어나다는 것을 증명하고 있다.

V. 결론

본 논문에서는 입력되는 영상으로부터 배경영상의 실시간 갱신을 통해 객체의 위치를 탐지하여 객체를 추적하고, 객체의 내부점을 이용한 MBR에서의 얼굴영역을 추출하는 방법을 제안하였다. 실험은 고정 PC카메라와 배경영상의 변화가 거의 없다는 제한된 환경 조건에서 실시간으로 배경영상의 갱신과 객체의 추적이 안정적인 모습을 보였다. 이는 객체의 추출 및 추적 알고리

즘이 빠르게 수행되어 객체 인식과 결합하여 객체의 움직임 정보와 인식을 통한 보안 및 감시 시스템 등 응용분야에 적용될 수 있다는 것을 기대할 수 있도록 하여 준다. 하지만 입력영상과 배경영상 각각의 R, G, B 값의 차를 이용, 객체의 일부분임을 인식할 때, 잡음과 조명의 영향으로 인하여 실험에 실패한 경우도 발생하여 이에 대한 보완이 필요하다. 향후 연구로는 레이블링을 이용한 다중객체의 추출과 추적이 실시간으로 이루어질 수 있도록 연구가 요구된다.

참 고 문 헌

- [1] E. Hjelm and B.K. Low, "Face Detection: A Survey," *Computer Vision and Image Understanding*, Vol. 83 No. 3, pp. 236-274, 2001.
- [2] T.V. Pham, M. Worring, and A.W.M. Smeulders, "Face detection by aggregated Bayesian network classifiers," *Technical Report 2001-04, Intelligent Sensory Information Systems Group*, University of Amsterdam, 2001.
- [3] M. Rogers and J. Graham, "Robust active shape model search," *Proceedings of the 7th European Conference on Computer Vision*, Vol. 4, pp. 517-530, 2002.
- [4] K. Okada and C. von der Malsburg, "Analysis and synthesis of human faces with pose variations by a parametric piecewise linear subspace method," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 1, pp. 761-768, 2001.
- [5] 이희영, 최재영, 강동구, 김홍수, 차의영, 전태수, "배경영상을 이용한 목표물 추적에 관한 연구," *한국멀티미디어학회 춘계학술발표논문집*, 제2권, 제1호, 386-390쪽, 1999년
- [6] 황본우, 손형진, 이성환, "대화형 하이퍼 비디오 저작을 위한 객체 기반 추적 방법," *정보과학회 추계학술발표논문집*, 제28권, 제2호, 427-429쪽, 2001년
- [7] 정연권, 하상석, 정선태, "다채널 영상감시 시스템을 위한 다중 포맷 동영상 저장 DirectShow Filter 설계 및 구현," *대한전자공학회 02 하계종합학술대회 논문집(4)*, 제4권, 113-116쪽, 2002년 6월
- [8] 서우성, 송명렬, "WWW를 이용한 실시간 영상감시 시스템의 설계 및 구현," *대한전자공학회 00 추계종합학술대회 논문집(1)*, 417-420쪽, 2000년, 11월
- [9] 오종안, 정영기, "능동카메라 환경에서의 특징기반의 이동물체 추적," *대한전자공학회 02 하계종합학술대회 논문집(4)*, 141-144쪽, 2002년, 6월
- [10] 박수형, 염동훈, 고기영, 김두영, "레벨 세트와 히스토그램을 이용한 이동 물체의 추적," *대한전자공학회 02 하계종합학술대회 논문집(4)*, 137-140쪽, 2002년, 6월
- [11] 김은태, "퍼지제어 이론을 이용한 샘플된 비선형 시스템의 추적제어에 대한 연구," *전자공학회논문지*, 제40권, 6호, 2003년 11월
- [12] 박기태, 문영식, "Dominant 컬러쌍 정보와 Color Correlogram을 이용한 객체기반 영상검색," *전자공학회논문지*, 제40권, 제2호, 2003년 03월
- [13] 이경미, 김달배, 이창우, 김향준, "비디오 감시시스템을 위한 영역 기반의 움직이는 물체 분할," *전자공학회논문지*, 제40권 제1호, 2003년 01월

저 자 소 개



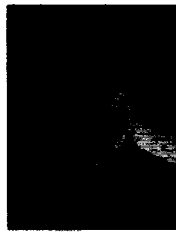
박 재 표(정회원)
 1996년 송실대학교 컴퓨터학부
 (공학사)
 1998년 송실대학교 컴퓨터학과
 (공학석사)
 2001년 송실대학교 컴퓨터학과
 (박사과정 수료)

1998년~현재 국립 한경대학교 컴퓨터공학과 강사
 <주관심분야: 멀티미디어 보안, DRM, PKI>



이 광 형(학생회원)
 1998년 광주대학교 전자계산학과
 (공학사)
 2002년 송실대학교 컴퓨터학과
 (공학석사)
 2004년 송실대학교 컴퓨터학과
 (박사과정 수료)

<주관심분야: 멀티미디어 보안, 멀티미디어 데이터베이스, DRM>



이 중 희(정회원)
 1998년 한밭대학교 전자계산학과
 (공학사)
 2000년 송실대학교 컴퓨터학과
 (공학석사)
 2004년 송실대학교 컴퓨터학과
 (공학 박사)

2000년~현재 국립 한경대학교 컴퓨터공학과 강사
 <주관심분야: 멀티미디어 보안, 유비쿼터스 컴퓨팅, e-Learning>



전 문 석(정회원)
 1981년 송실대학교 전자계산학과
 (공학사)
 1986년 University of Maryland
 Computer Science
 (공학석사)
 1989년 University of Maryland
 Computer Science
 (공학박사)

1989년 3월~7월 Morgan State University 조교수
 1989년~1991년 New Mexico State University
 Physical Science Lab 책임 연구원
 1991년~현재 송실대학교 부교수
 <주관심분야: 전자상거래 보안, 인터넷 보안, 멀티미디어 보안, 인증 시스템>