

동영상에서 적응적 배경영상을 이용한 실시간 객체 추적

최내원[†] · 지정규^{**}

요 약

동영상에서 객체 추적은 몇 년간 컴퓨터 비전 및 여러 실용적 응용 분야에서 관심을 가지는 주제 중 하나이다. 본 논문에서는 보안 및 감시 시스템 분야에 적용할 수 있는 실시간 객체 추적과 얼굴영역 추출 방법을 제안한다. 이를 위해 카메라가 고정되어 있고 배경영상의 변화가 거의 없는 환경으로 제한하고, 입력영상과 배경영상의 차를 이용하여 객체의 위치를 탐지하고 움직임을 추적한다. 보다 안정적인 객체 추출을 위해 적응적 배경영상을 생성하고, 객체 위치 탐지 시 그물식 탐색방법을 이용하여 객체의 내부점을 추출한다. 추출된 점들을 이용하여 MBR(Minimum Bounding Rectangle)을 설정하여 객체의 실시간 추적을 가능하도록 하였다. 또한 설정된 MBR 내에서 얼굴영역을 추출함으로써 보안 및 감시 시스템의 효율성을 향상시켰다. 그리고 실험을 통하여 제한된 환경 하에서 실시간으로 빠른 객체의 추적을 보인다.

Real-time Object Tracking using Adaptive Background Image in Video

Nae-won Choi[†] and Jeong-gyu Jee^{**}

ABSTRACT

Object tracking in video is one of subject that computer vision and several practical application field have interest in several years. This paper proposes real time object tracking and face region extraction method that can be applied to security and supervisory system field. For this, in limited environment that camera is fixed and there is seldom change of background image, proposed method detects position of object and traces motion using difference between input image and background image. The system creates adaptive background image and extracts pixels in object using line scan method for more stable object extraction. The real time object tracking is possible through establishment of MBR(Minimum Bounding Rectangle) using extracted pixels. Also, effectiveness for security and supervisory system is improved due to extract face region in established MBR. And through an experiment, the system shows fast real time object tracking under limited environment.

Key words: 동영상, 배경영상, 객체 추출, 실시간 객체 추적, MBR

1. 서 론

동영상에서 실시간으로 객체 추적을 위해 컴퓨터 비전 기술을 이용하여 구현하기란 매우 어려운 일임에도 불구하고, 컴퓨터 성능의 발달로 인해 영상 처

리 기법의 발전과 더불어 객체 인식과 객체 추적에 관한 연구가 활발히 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].

하지만, 기존의 실시간 객체 추적 시스템의 구현

접수일 : 2002년 8월 30일, 완료일 : 2002년 12월 23일

[†] 정회원, 명지전문대학 교수

^{**} 한국학술진흥재단

에는 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 연산량을 필요로 하는 알고리즘 등을 적용하여 객체 추출에는 뛰어난 성능을 보이지만 실시간 객체 추적에는 만족스런 결과를 얻지 못하고 있다.

컴퓨터 비전 시스템을 이용한 인공 시각, 컴퓨터와 인간의 상호 작용, 영상 기반 제어 장치, 감시 시스템 분야에서 사용되는 객체 추적 방법은 각각 특정 환경에서 적합한 동작을 보이도록 몇 가지 제약 조건을 두어 설계되고 있다. 카메라의 동작을 제한하여 객체와 배경의 분리를 용이하게 하는 방법, 추적하고자 하는 객체를 미리 정하여 그 객체의 특징과 형태를 미리 학습시켜 추적하는 방법, 또는 추적하고자 하는 객체의 수를 제한하는 방법 등이 그것이다[2].

본 연구에서는 보안 및 감시 시스템 분야에서 적용할 수 있는 방법을 제안하기 위하여, 저가형 PC 카메라의 움직임이 고정되어 있고, 외부 환경이 아닌 실내 환경이며, 배경영상의 변화가 거의 없다는 특수 환경으로 제약 조건을 가진다.

먼저 시변하는 잡음을 제거하고 조명에 강인한 객체 추출이 가능하도록 적응적 배경영상을 생성한다. 생성된 적응적 배경영상과 설치된 PC카메라로부터 실시간으로 입력되는 입력영상의 차를 이용하여 객체의 크기와 위치를 탐지하여 객체를 추출한다. 실시간으로 객체 추출을 하기 위하여 완전 차영상을 이용한 방법이 아닌 그물식 탐색방법을 이용한 객체 추출 방법을 사용한다. 그물식 탐색방법으로 추출된 객체의 내부점을 이용하여 MBR(Minimum Bounding Rectangle)을 설정하고, 이를 통해 객체의 실시간 추적을 가능하도록 하였다.

또한 설정된 MBR 내에서 얼굴 영역을 추출하기 위한 방법을 적용한다. 실시간으로 입력되는 영상으로부터 바로 얼굴 영역을 추출함으로써 보안 및 감시 시스템에서의 효용성을 향상시켰다.

본 논문의 구성을 보면, 먼저 2장에서는 관련 연구를 통해 기존의 연구방법을 분석하여 특징 및 문제점을 기술한다. 3장에서는 적응적 배경영상 생성방법과 그물식 탐색방법을 통해 객체의 내부점을 추출하고, 이를 이용하여 MBR을 설정한 다음 실시간으로

객체를 추적하는 방법과 얼굴영역을 추출하는 제안 방법에 대해 기술한다. 그리고 4장에서는 제안한 방법을 구현하고 실험하여 분석한 성능평가에 대해 기술하고, 마지막으로 5장에서는 결론 및 향후 연구 방향을 기술한다.

2. 관련 연구

기존의 추적 알고리즘으로는 3차원 모델 기반의 방법(Model-based Tracking)[4], 영역 기반의 방법(Region-based Tracking)[5], 윤곽선 기반의 방법(Contour Tracking)[6], 특징기반의 방법[7], 시공간 경사법(spatio-temporal gradient method)[8], 무게 중심법(centroid method)[9], 정합법(matching method)[10] 등이 있다.

3차원 모델 기반의 방법은 정확성이 높은 모델과 궤적을 복원하는 것으로, 상세한 기하학적 물체의 모델이 주어져야 하는 단점이 있는데, 그것은 모든 움직이는 물체에 대해 상세한 모델을 기대하는 것이 비현실적이기 때문이다.

영역 기반의 방법은 연속 영상에서 연결된 영역을 구하고 상관관계(cross-correlation) 측정을 이용하여 움직이는 물체를 추적하는 방법으로, 현재의 배경을 측정하여 입력되는 영상과의 차영상에서 물체를 검출한다. 이 방법은 차영상에 임계값 이상의 화소를 연결한 영역은 찾을 수 있으나 혼잡한 물체의 상태에서 각각의 물체를 분할해야 하는 단점이 있다.

윤곽선 기반의 방법은 물체의 경계인 윤곽선을 표현하고, 그것을 동적으로 갱신하면서 추적하는 방법으로, 이 방법의 이점은 영역 기반에 비해 복잡한 계산이 줄어든다는 것이나 부분적인 가려짐이 발생할 경우 물체의 추적이 불가능하다는 단점이 있다[3].

물체의 이동으로 인하여 생기는 밝기의 시간적 변화도와 공간적 변화도 사이의 상호관계로부터 이동 변위를 추출하는 시공간 경사법은 추적 물체가 회전 운동을 하거나 물체의 움직임이 클 경우 추적의 어려움이 있다.

각 시변 영상을 표적과 배경으로 분리하여 이진화한 후 표적의 중심을 추출하여 그 중심의 변화로부터 표적의 이동 정보를 검출하는 무게 중심법은 비교적 계산이 간단하며 계산량 감소를 위해서 물체의 최대 이동 추정 변위에 제한을 둘 필요는 없으나 시변 영

상을 표적과 배경으로 정확히 분리하는데 어려움이 있다.

그리고 틀 영상에서 화소 자체의 정보나 또는 물체의 특징을 추출하여 탐색 영역을 이동하면서 유사성이 최대인 정합점을 찾는 정합법은 시변 물체의 밝기변화, 물체의 확장 및 축소, 그리고 물체의 회전에 적절히 대응하지 못하는 단점이 있다[11].

3. 실시간 객체 추적과 얼굴영역 추출

3.1 제안 시스템의 구조

그림 1은 제안한 실시간 객체 추적 및 얼굴 영역 추출 시스템의 전체적인 구조도이다.

PC카메라로부터 실시간 동영상을 입력받아 시스템 전처리를 수행한다. 시스템 전처리는 1단계로 조명과 잡음에 강인한 객체 추적을 위해 배경영상과 계속적으로 입력되어지는 영상을 이용하여 적응적 배경영상을 생성한다. 그리고 2단계로 생성된 적응적 배경영상과 입력영상 사이에 그물식 탐색방식을 이용하여 객체 내부점을 추출한다. 마지막으로 3단계에서는 추출된 객체 내부점으로부터 MBR을 설정해 주게 된다. 시스템 전처리가 끝나면 실시간 객체 추적을 통해 얼굴영역 추출을 처리하게 된다.

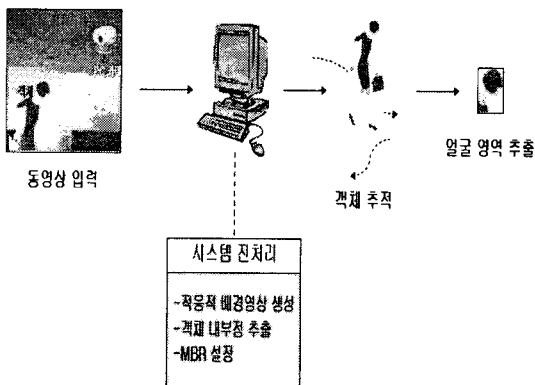


그림 1. 시스템 구조도

컴퓨터 비전 시스템을 이용하여 객체를 추적하는 방법은 각각 특정 환경에서 적합한 동작을 보이도록 몇 가지 제약 조건을 두어 설계되어져 왔다. 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 공원이나 거리 등 야외 환경이 아닌 사무실이나 연구실, 기자재실, 자재창고, 주차장, 은행 365일 코너 등 배경영상의 변화가 미미한 특수 환경으로 환경을 제한한다.

3.2 시스템 처리 흐름도

그림 2는 제안하는 방법의 전체적인 처리 흐름도이다.

먼저 PC카메라로부터 실시간으로 받아들여지는 영상 중에서 객체의 출현이 감지되지 않는 영상을 초기의 배경영상으로 설정하여 준다. 이후, 연속적으로 입력되어지는 영상과 초기 배경영상과의 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 처리를 반복하게 된다. 연속된 MBR의 설정을 추적함으로써 객체의 실시간 추적을 가능토록 하였다.

추적과 동시에 전체 영상이 아닌 MBR 내에서 얼굴 영역을 추출하는 방법을 사용하여 실시간 얼굴영역 추출도 가능토록 하였다.

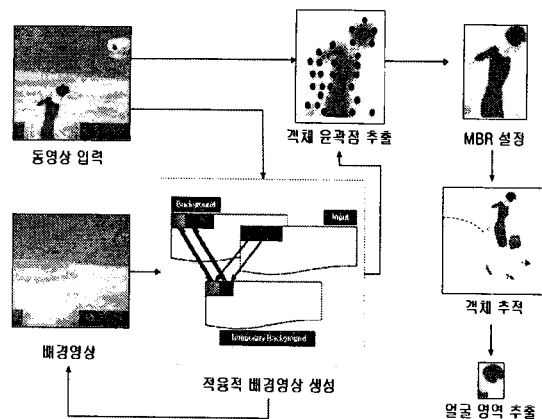


그림 2. 시스템 처리 흐름도

3.3 시스템 구현 방법

3.3.1 적응적 배경영상 생성

그림 3은 PC카메라로부터 입력되어 초기에 설정된 배경영상을 나타내고, 그림 4는 객체의 움직임이 감지되지 않은 상태에서 180초 후의 배경영상을 나타낸다. 사람의 감지 능력으로는 두 그림이 똑같아 보이지만, 시스템에서는 둘을 서로 다르게 인식한다.

그림 5는 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 대상으로 한 선택이었는데 고성능의 카메라가 아니기 때문에 빛에 대해 상당히 민감한 반응을 보인다. 1초, 20초, 60초, 180초 후 영상에서 각각 71개, 1657개, 2061개, 3891개의 최초 배경영상과의

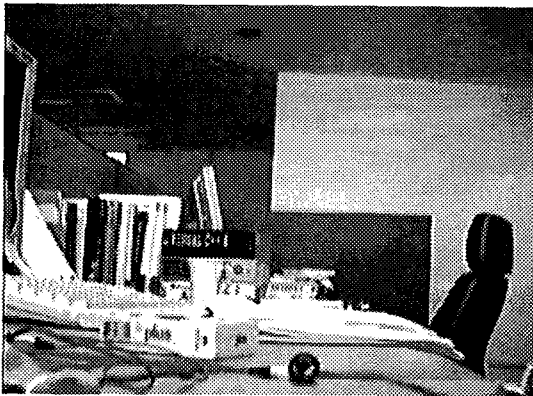


그림 3. 초기 배경영상

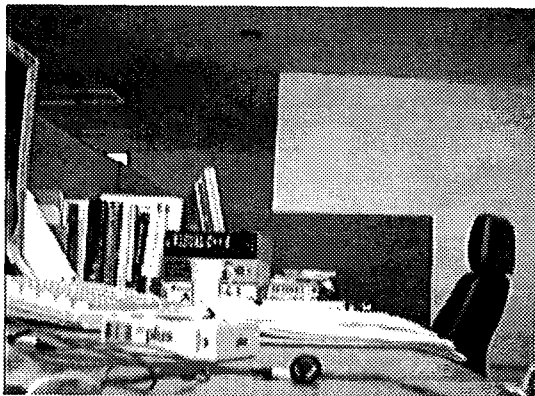
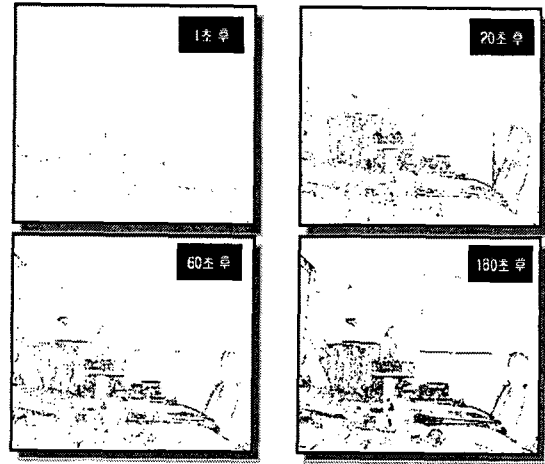


그림 4. 180초 후의 배경영상



[초당 10프레임 처리시]

그림 5. 시변하는 배경영상에서의 잡음

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 $n \times m$ 마스크를 이용하여 계속적인 배경영상의 갱신을 하면서 객체 추적을 동시에 이루어질 수 있도록 하였다.

그림 6은 적응적 배경영상의 생성 방법을 나타낸다.

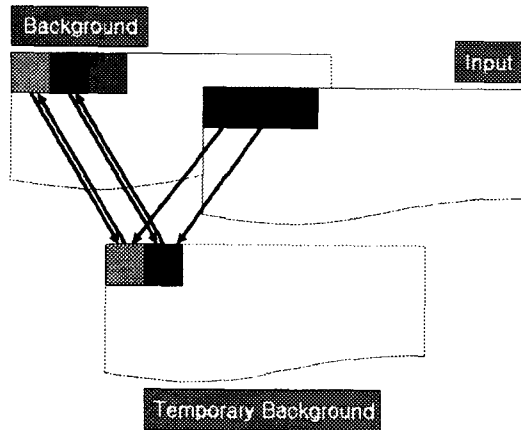


그림 6. 적응적 배경영상 생성

입력영상과 배경영상의 $n \times m$ 마스크 내에서 각 픽셀의 RGB 채널의 각각의 차이값을 구한다. (식 1)에 따라 RGB 채널 각각의 차이값이 임계값 a 보다 작다면, 객체의 부분이 아니라고 판단하여 입력영상에서 픽셀을 선택하여 갱신하여 준다. 그렇지 않다면 새로운 객체의 부분이 출현했다고 판단하여 배경영상의 픽셀을 그대로 사용한다.

```

R_mask = abs (GetRValue(BGImg_mask[i,j])
              - GetRValue(TGImg_mask[i,j]) )
G_mask = abs (GetGValue(BGImg_mask[i,j])
              - GetGValue(TGImg_mask[i,j]) )
B_mask = abs (GetBValue(BGImg_mask[i,j])
              - GetBValue(TGImg_mask[i,j]) )
IF ( (R_mask < a) OR
      (G_mask < a) OR
      (B_mask < a) )
    choice TGImg_mask[i,j]
ELSE
    choice BGImg_mask[i,j]           (식 1)
* BGImg_mask : 배경영상에서의 마스크
* TGImg_mask : 입력영상에서의 마스크
* a : 적응적 배경영상 생성 임계값
    
```

제안한 적응적 배경영상 방법을 이용하여 실험한 결과, 시변하는 배경영상내의 조명 등에 의한 잡음을 91% 이상 제거할 수 있었다.

3.3.2 객체의 내부점 추출

객체의 추적을 위해서 객체의 위치 결정을 위한 객체 추출이 우선적이다. 완전 차영상 방식을 도입한 기존 시스템과는 달리, 제안하는 시스템에서는 그물식 탐색 방법을 이용한다. 그물식 탐색 방법은 영상을 δ 픽셀 간격으로 탐색해 나가는 방법으로 기존 시스템보다 처리속도의 높은 향상율을 보인다.

그림 7은 그물식 탐색 방법을 이용한 객체의 내부점 추출을 나타낸다. 객체의 내부점 추출은 그물식 탐색을 하면서 배경영상과 입력영상의 차영상을 이용하게 된다. 배경영상과 입력영상의 탐색라인상의 픽셀들을 각각 RGB채널로 분할한다. (식 2)에 따라 RGB 채널 각각의 차이값이 임계값 β 보다 크다면, 객체 내부점 후보군으로 등록하게 된다.

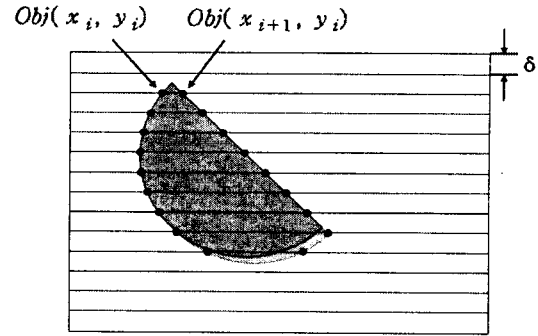


그림 7. 그물식 탐색 방법

제안 시스템은 보안 및 감시 시스템에 적용되어지기 때문에 사람이라는 객체 이외에 빛의 간섭 등으로 인한 배경에서의 작은 잡음 제거를 위해 (식 3)을 이용하게 된다. 또한 객체의 일부분이지만 효율적인 잡음 제거를 위하여 어느 정도의 미세한 객체의 일부는 무시한다.

(식 2)를 만족하는 객체 내부점 후보군에서 연속된 픽셀이 v 이하라면 잡음으로 판단하여 제거하고, 원하는 객체 내부점 $Obj(x_i, y_i)$ 를 추출하게 된다.

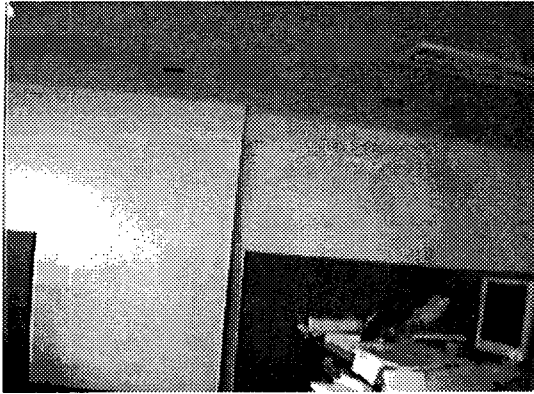
```

R_obj = abs (GetRValue(BGImg[x_i,y_i])
             - GetRValue(TGImg[x_i,y_i]) )
G_obj = abs (GetGValue(BGImg[x_i,y_i])
             - GetGValue(TGImg[x_i,y_i]) )
B_obj = abs (GetBValue(BGImg[x_i,y_i])
             - GetBValue(TGImg[x_i,y_i]) )
IF ((R_obj >= beta) OR
      (G_obj >= beta) OR (B_obj >= beta))
    ObjInit(x_i, y_i) = true
ELSE
    ObjInit(x_i, y_i) = false           (식 2)
* ObjInit(x_i, y_i) : 객체 내부점 후보군
* beta : 객체 내부점 후보군 추출 임계값

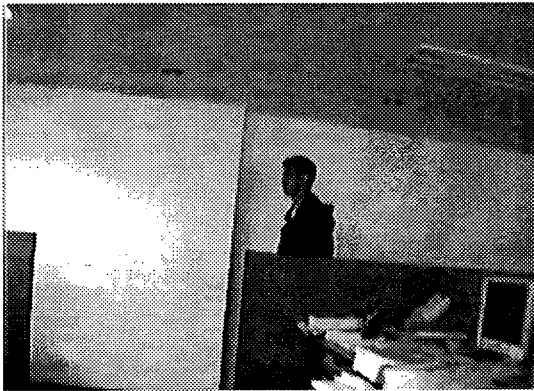
x_{i+1} - x_i >= v                     (식 3)
* v : 객체 내부점 추출 임계값
    
```

그림 8은 계속적으로 갱신을 반복하는 적응적 배경영상과 실시간으로 입력되어지는 영상을 나타낸다.

그리고 그림 9는 배경영상과 입력영상에서 그물식 탐색 방법을 이용한 객체 내부점 추출을 나타낸다.



(a)



(b)

그림 8. (a) 배경영상, (b) 입력영상

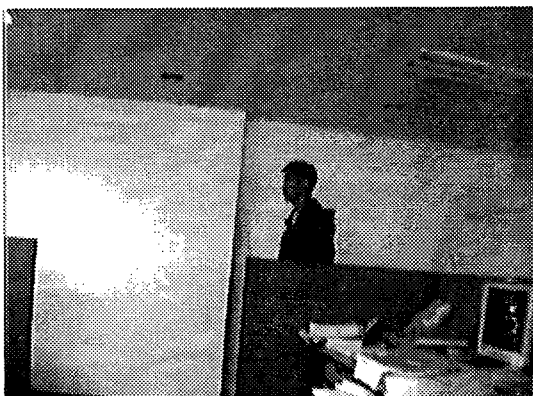


그림 9. 객체 내부점 추출

그물식 탐색 방식을 이용하여 객체 내부점을 추출한 것이 완전 차영상을 이용하여 객체를 추출하는 것에 비하여 강건한 객체 추출을 보이지는 않지만, 보안 및 감시 시스템을 대상으로 한 제안 시스템에서

는 객체 추적의 대상인 사람을 추출하는 목적에는 적합함을 볼 수 있고, 기존 시스템과 비교하여 객체 추출의 속도를 빠르게 수행함을 볼 수 있다.

3.3.3 MBR을 이용한 실시간 객체 추적

추출된 객체 내부점을 이용하여 MBR을 설정하여 준다. MBR은 객체 추적을 위한 객체를 포함하는 최소 사각 영역으로써, 얼굴 영역 추출 시 속도를 향상시키기 위한 범위를 제한해 주는 역할까지 수행한다.

보다 정확한 MBR의 설정을 위한 전처리 과정으로 본 연구에서는 Median Filtering을 통해 잡음을 제거하여준다. 제안 시스템에서는 완전 차영상이 아닌 그물식 탐색 방법을 이용하기 때문에 일반적인 $n \times n$ block median filter window를 사용하는 것이 아니라, horizontal median filter window를 이용한다. 또한 컬러 영상에 적용되는 median filtering을 위한 프로시저를 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 값에 대한 차를 합한 것보다 작다는 것이다. (식 4)와 같이 각 화소 샘플에 대하여 각각의 RGB 채널에 대한 차들을 합한다. 이 중 가장 작은 값을 가진 $Distance_i$ 는 필터의 출력인 x_i 에 대응된다.

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

$$Distance_i = \sum_{j=1}^N (|red_i - red_j| + |green_i - green_j| + |blue_i - blue_j|) \quad (식 4)$$

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

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

객체 내부점 $Obj(x_i, y_i)$ 의 x, y 좌표 중 (식 5)를 통하여 각각의 최대, 최소 좌표를 구하여 객체를 포함하는 최소한의 사각 영역을 설정해 준다.

$$Obj_left = \min [Obj(x_i)] - \delta$$

$$Obj_top = \min [Obj(y_i)] - \delta$$

$$\begin{aligned}
 Obj_right &= \max [Obj(x_i)] + \delta \\
 Obj_bottom &= \max [Obj(y_i)] + \delta \\
 MBR &= [Obj_left, Obj_top, Obj_right, \\
 &\quad Obj_bottom] \quad (\text{식 5})
 \end{aligned}$$

- * $Obj(x_i, y_i)$: 객체 내부점
- * $Obj_left(top, right, bottom)$: MBR의 최대, 최소 좌표

그림 10은 추출된 객체 내부점을 이용하여 설정된 MBR을 나타내는데, 이를 통하여 본 논문에서는 객체의 실시간 추적을 가능하도록 하였다.

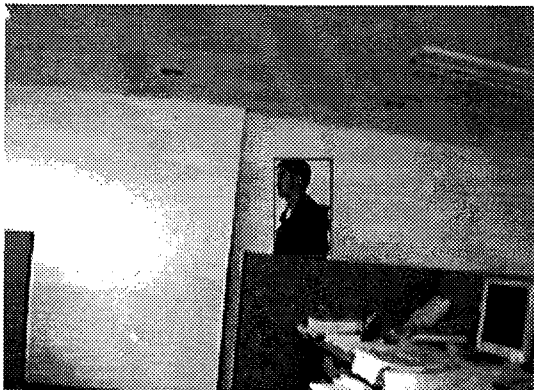


그림 10. 설정된 MBR

3.3.4 얼굴영역 추출

설정된 MBR 내에서 RGB 공간에서의 색상 정보와 모양정보를 이용하여 얼굴 영역을 검출한다. 이때 사용하는 방법은 주성분분석(Principal Component Analysis) 방법을 이용한다. 검출된 얼굴영역을 포함하는 최소 사각 영역(Minimum Bounding Rectangle for Face Region)을 설정해 줌으로써 얼굴영역을 추출한다. 이 영역은 추후 사람 인식 분야와의 결합을 통해 보안 및 감시 시스템의 효율성을 높일 수 있다.

4. 구현 및 성능분석

4.1 실험 환경

본 실험은 카메라의 움직임이 고정되어 있고 배경영상의 변화가 거의 없다는 특수 환경으로 제약하고, 입력영상을 실시간으로 받아들여 객체의 위치를 탐

지한 다음 이동 객체를 추적하는 실험을 하였다.

본 논문에서 제안한 방법의 구현을 위해 프로그래밍 언어 Delphi 6.0을 이용하였고, Intel(R) Pentium 4 CPU 1.60GHz, 256MB RAM을 가진 하드웨어에 Microsoft Windows 2000 운영체제를 탑재한 PC를 사용하여 실험하였다. 그리고 저가형 PC 카메라로부터 받아들이는 배경영상과 입력영상의 크기는 320 X 240의 24bit 컬러영상과 160 X 120의 24bit 컬러영상을 이용하였다.

4.2 시스템 구현

그림 11은 구현된 실시간 객체 추적 및 얼굴영역 추출 시스템으로 PC카메라로부터 입력받은 동영상에서 객체를 추출하고 MBR을 통해 실시간으로 객체를 추적하며, 얼굴 영역을 추출하는 과정을 나타낸다.

인터페이스의 상단부에는 제안한 시스템에서 사용되어지는 임계값과 변수에 대한 값을 입력하여 환경과 상황에 가장 적합한 객체 추적을 가능하도록 하였으며, 입력되어지는 동영상의 크기, 해상도 등은 선택적으로 입력되게 하였다. Background Image 부분은 계속적으로 갱신되고 있는 배경영상을 나타내고, Input Image 부분은 실시간으로 입력되어지는 영상을 보여준다. 왼쪽 하단부는 처리되어진 프레임수와 초당 처리 속도(frame per second) 등을 나타내 준다.

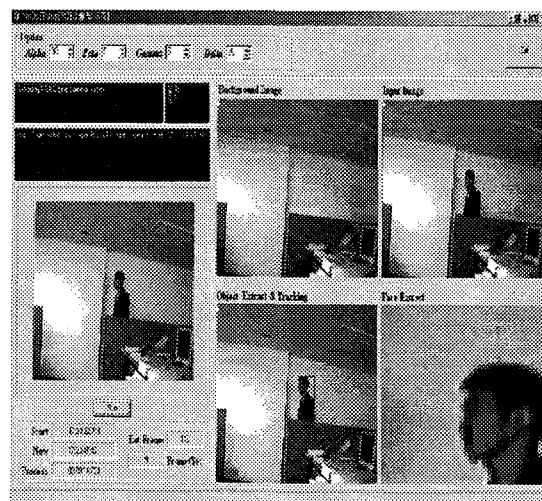


그림 11. 실시간 객체 추적 및 얼굴 영역 추출 시스템

Object Extract & Tracking 부분은 객체를 추출하고 추적하는 모습을 보이는데, 입력된 영상으로부터 객체를 잡음 없이 추출하고 실시간 객체 추적이 가능하다. Face Extract 부분은 작은 영상으로부터 선택되어져 모자이크 현상을 보이지만 내부적으론 얼굴영역을 정확히 추출할 수 있었다.

4.3 실험 결과

표 1은 본 논문에서 제안한 방법을 위한 임계값들과 변수에 대한 최적화 실험 결과이다. 표 1에서 보는 바와 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.

표 1. 임계값과 변수에 대한 최적화 실험 결과

실험값 \ 임계값	1	2	...	5	...	25	...	30	...
α (적응적 배경영상 생성 임계값)	0.023	0.151	...	0.210	...	0.915	...	0.887	...
β (객체 내부점 후보군 추출 임계값)	0.101	0.256	...	0.494	...	0.856	...	0.934	...
γ (객체 내부점 주변 임계값)	0.495	0.531	...	0.926	...	0.349	...	0.324	...
δ (그물식 탐색을 위한 스캔 간격)	0.951	0.903	...	0.851	...	0.595	...	0.496	...

그림 12는 320 X 240의 해상도를 지닌 적응적 배경영상과 실시간으로 입력되어지고 있는 입력영상 중 하나를 나타내고 있다.

그림 13은 입력된 동영상으로부터의 실시간 객체 추적 과정을 하드디스크에 BMP파일로 저장되어진 연속된 파일의 일부를 보이고 있다.

제안한 방법으로 변화된 장소, 카메라의 위치, 높이, 조명의 변화 등 여러 상황으로 40여건의 Dataset에서 실험하였을 때 320 X 240 해상도의 영상은 초당 9~13 프레임을 처리하였으며, 160 X 120 해상도의 영상은 초당 17~23 프레임의 처리 성능을 보였다.



그림 12. 배경영상(좌)과 입력영상(우)

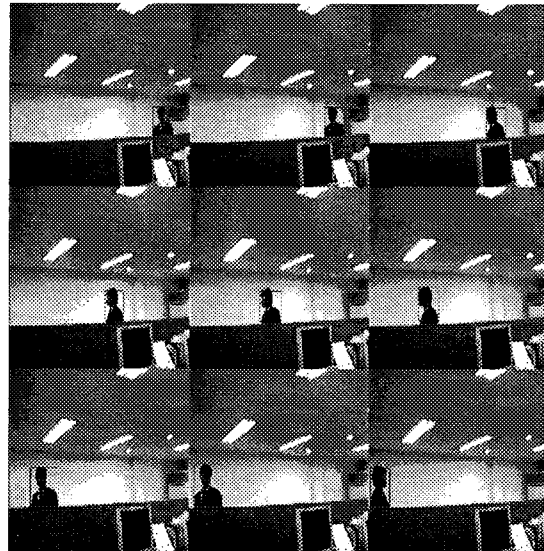


그림 13. MBR을 이용한 실시간 객체 추적

이는 기존의 객체 추적 방법에 비하여 처리 속도가 향상되었음을 나타내며, 제안한 알고리즘을 통하여 실시간 객체 추적 및 얼굴 영역 추출이 가능함을 보인다. 객체 추적 및 얼굴 영역 추출 처리 속도는 추적 대상인 객체의 크기 등에 따라 민감하게 반응하며 객체가 작았을 경우, 각각 최대 11프레임과 20프레임 까지도 처리 속도를 보였다.

4.4 성능 평가

표 2는 제안한 시스템과 기존 방법들과의 성능 비교이다. 성능 비교에 사용된 영상은 160 X 120 해상도의 24 bit 컬러영상을 사용하였다. 표 2에서 보는 바와 같이 그물식 탐색 방법을 도입한 본 시스템이

표 2. 시스템 성능 평가

구 분	처리속도 (FPS)	추출율	추적률
3차원모델기반의 방법	5~8	89%	88%
영역기반의 방법	11~16	94%	91%
윤곽선기반의 방법	14~19	98%	96%
특징기반의 방법	8~11	94%	92%
시공간 경사법	7~12	96%	91%
무게 중심법	13~17	97%	98%
정합법	11~14	95%	94%
제안 시스템	17~23	97%	96%

기존의 방법들에 비해 매우 빠른 처리속도를 보인다. 기존의 몇몇 방법들에 비해 강인한 객체 검출에는 조금 떨어지는 성능을 보이나, 실시간 객체 추적에 있어서는 뛰어난 성능을 보임을 알 수 있다.

5. 결론 및 향후 연구 방향

본 논문에서는 입력되어지는 영상으로부터 실시간으로 객체의 위치를 탐지하고 객체를 추적하여 열골영역을 추출하는 방법을 제안하였다.

실험은 고정 PC카메라와 배경영상의 변화가 거의 없다는 제한된 환경 조건에서 실시간으로 객체의 추적이 안정적임을 보여주었다. 이는 객체의 추출 및 추적 알고리즘이 빠르게 수행되어 객체 인식과 결합하여 객체의 움직임 정보와 인식을 통한 보안 및 감시 시스템 등의 응용분야에 적용되어 질 수 있다는 것을 기대할 수 있다.

그러나 입력영상과 배경영상 각각의 R, G, B 값의 차를 이용하여 객체의 일부분임을 인식할 때, 잡음과 조명의 영향으로 인하여 실험에 실패한 경우도 발생하여 이에 대한 보완이 필요하다.

앞으로 보다 정확한 객체의 추출을 위한 안정적인 시스템의 설계가 필요하며, 다중 객체 추적과 3차원적 추적방법에 대한 연구가 진행 중이다.

참 고 문 헌

[1] 이희영, 최재영, 강동구, 김홍수, 차의영, 전태수,

“배경영상을 이용한 목표물 추적에 관한 연구”, 한국멀티미디어학회 1999년도 춘계 학술발표논문집 (학술발표), Vol.2, No.1, pp.386-390, 1999.

[2] 황본우, 손형진, 이성환, “대화형 하이퍼 비디오 저작을 위한 객체 기반 추적 방법”, 정보과학회 2001년 추계 학술대회, Vol.28, No.2, pp.427-429, 2001.

[3] 이상욱, “윤곽선 모델과 특징을 이용한 이동 물체 추적”, 경상대학교 해양산업연구소보, Vol. 14, pp.42-51, 2001.

[4] D. Koller, J. Daniilidis and H. Nagel, “Model-based Object Tracking in Monocular Image Sequences of Road Traffic Scenes.” Int'l J. of Computer Vision, Vol.10, No.3, pp.257-281, 1993.

[5] P. Salesmbier, L. Torres, F. Meyer and C. Gu, “Region-based Video Coding Using Mathematical Morphology.” Proc. of the IEEE, Vol.83, No.6, pp.843-857, 1995.

[6] M. Isard and A. Blake, “Contour Tracking by Stochastic Propagation of Conditional Density.” In Proc. European Conf. Computer Vision, pp.343-356, 1996.

[7] B. Rao, “Data Association Methods for Tracking Systems.” In A.Black and A.Yuille. editors, Active Vision, pp.91-105, MIT, 1992.

[8] T. Augi, T. Ishihara, H. Nagahashi and T. Nagae, “Contour tracking and synthesis in image sequences.” SPIE '95, pp.834-845, 1995

[9] R. Venkateswarlu, K. Sujata and B. Venkateswara, “Centroid tracker and aim point selection.” SPIE, Acquisition, Tracker and Pointing IV, Vol.1697, pp.520-529, 1993.

[10] Hamid Naseri and John A. Sttler, “Segmentation motion estimation.” ICASSP, pp.1906-1910, 1996

[11] 임용호, 백중환, 황수찬, “퍼지 예측을 이용한 이동물체 추적”, 한국항행학회 논문지, Vol.5, No. 1, pp. 26-36, 2001.



최 내 원

1980년 광운대학교 전자계산학과
(공학사)

1988년 숭실대학교 전자계산학과
(공학석사)

1996년~숭실대학교 컴퓨터학과
박사과정 수료

1983년~현재 명지전문대학 교수

관심분야 : 멀티미디어, 데이터베이스, 영상처리



지 정 규

1987년 서울산업대학교 전자계산
학과(공학사)

1989년 숭실대학교 전자계산학과
(공학석사)

1998년 숭실대학교 전자계산학과
(공학박사)

1978년~1996년 (주)삼호, 서울시
설관리공단 전산실

1996년~현재 한국학술진흥재단

관심분야 : 멀티미디어, 데이터베이스, 영상처리

교신저자

지 정 규 137-170 서울시 서초구 염곡동 304