

# 다중카메라 기반의 특정객체 추적시스템의 설계 및 구현

민병묵\*, 이광형\*\*, 민소연\*\*\*

\*나노웨어(주)

\*\*서일대학 인터넷정보과

\*\*\*서일대학 정보통신과

## 요 약

본 논문은 박물관이나 고가의 물품을 판매하는 곳에서 특정한 객체를 감시하고, 특정 객체의 도난을 방지하기 위하여 다중카메라를 설치하여 카메라 상호간의 정보를 교환함으로써 움직이는 객체를 추출하고 추적하는 시스템의 구현이다. 감시대상이 되는 객체의 상단과 정면에 카메라를 설치하여 움직임 객체가 감시대상 객체에 접촉하게 되면 두 대의 카메라가 동시에 움직임 객체를 추출하고 추적하게 된다. 먼저 정면의 카메라는 움직임 물체의 얼굴부분을 캡춰하고 지속적으로 얼굴영역을 확대 / 캡춰하면서 추적을 시작한다. 상단의 카메라는 많은 객체들의 움직임 속에서 특정한 객체만을 추적할 수 있으며, 추적은 방향예측을 통하여 수행하고 객체의 특징정보를 저장한다. 저장된 특징정보는 카메라의 범위를 벗어났을 때 인접한 카메라에 정보를 전송하고 지속적인 추적이 이뤄질 수 있도록 한다.

## 1. 서론

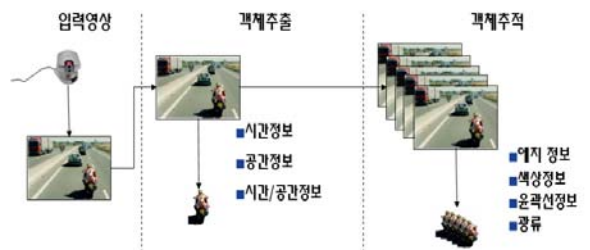
최근 컴퓨터비전 기술의 발달에 따라 영상감시 시스템에 대한 연구와 응용이 활발하게 이루어지고 있다. 영상감시 시스템의 응용연구는 주로 카메라를 통해 입력받은 영상에서 특정 객체를 추출하여 추적하는데 초점이 맞추어지고 있다. 그러나, 객체를 추출하기 위해서는 많은 연산량과 시간이 소요되고, 객체를 추적할 때는 추적대상 객체가 다른 객체에 의해 가려졌을 경우 지속적인 추적이 어려워지게 된다. 본 연구에서는 다중객체의 겹쳐짐 현상에서도 특정객체를 추적할 수 있는 다차원 객체추적 시스템을 제안한다. 감시대상 객체가 있는 영역의 상단과 전면에 각각의 카메라를 설치한다. 상단의 카메라가 특정객체를 감시하다가 객체의 이동정보가 식별되면 특정객체 정보를 전면의 카메라에 전송하면서 전면 카메라의 동작을 요청한다. 전면 카메라는 넘겨받은 객체정보를 이용해 적은 연산량으로 신속히 특정객체를 인식하고 추적을 개시한다. 추적 중인 객체가 다른 객체에 의해 가려지게 되면 상단 카메라로부터 추적대상 객체 정보를 도움 받아 지속적으로 추적을 진행한다. 제안 시스템의 실험결과 연산에 참여하는 픽셀 수가 현저히 줄어들고, 객체의 인식률도 향상되었다. 그리고 기존 방법과 비교하여 객체의 움직임을 검출하는 정확성은 물론 움직임 검출에 소요되는 시간도 단축되어 개선된 성능을 보였다. 고가이거나 희귀한 물건의 감시에 응용하면 사전에 도난으로부터의 예방이 가능할 것으로 보이며, 감시 영역의 폭을 넓히기 위해 이 시스템을 병렬로 연결하는

연구가 필요하다.

## 2. 객체추적 시스템

### 2.1. 객체추적시스템의 개요

객체 추적 시스템은 추적 객체의 특징을 다양한 각도에서 해석하여 연속된 비디오 시퀀스 시간의 흐름에 따라 변화를 감시하고 추적정보를 변환하는 기법을 통해 객체의 위치를 추적하게 된다. 객체 추적 시스템은 [그림 2-1]과 같이 크게 영상 입력부, 객체추출부, 객체추적부로 구성된다.



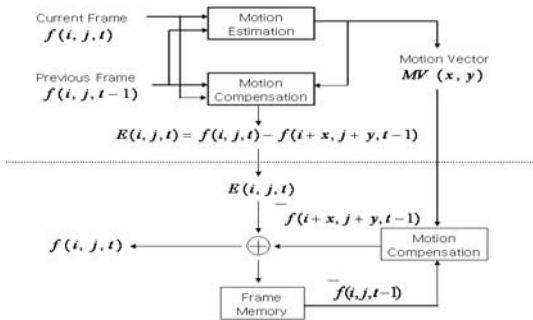
[그림 2-1] 객체 추적 시스템의 구성도

영상 입력부는 객체 감시를 위한 실제 장면을 카메라를 통해 시스템으로 입력한다. 카메라를 통하여 입력되는 영상데이터는 색상, 텍스처, 모양, 경계 등으로 표현을 하게 된다. 객체 추출부는 입력된 영상을 분석하여 특정 객체를 식별해낸다. 객체를 추출하기 위해서는 시간 정보와 공간 정보를 사용한다. 객체 추적부는 객체 추출부에서 추출된 객체가 움직이는 도선을 쫓아 궤적을 추적하게 된다. 특정 객체

의 이동을 추적하기 위해서는 시간적으로 갱신되는 영상에서의 색상 정보, 에지 정보, 윤곽선 정보 등을 사용하게 된다.

**2.2. 객체 추적 알고리즘**

비디오 영상은 연속인 프레임의 작은 변화로 이루어져 있다. 움직임 검출은 프레임 사이의 이러한 작은 변화를 예측하여 시간적인 중복성을 제거하기 위한 것이다. 움직임 검출의 원리는 연속된 프레임의 각각의 화소가 값의 변화 없이 위치만 움직인다는 전제하에 현재 프레임의 화소값을 추정하여, 추정된 화소값과 실제 화소값의 차이와 움직임 정보인 움직임 벡터를 추출하는 것으로 [그림 2-2]와 같다 [1]. 객체의 움직임 검출방법은 이미지의 전체 픽셀을 탐색하는 전역탐색 방법과 일부분의 픽셀을 이용하는 고속탐색 방법 등이 있다.

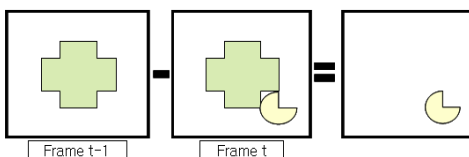


[그림 2-2] 움직임 검출 방법

**(1) 차분영상 기법**

차분영상을 이용한 움직임 검출 기법은 현재 프레임 Frame t-1과 이전프레임 Frame t의 차를 이용하여 움직임을 검출하며, 연속된 두 프레임간의 차를 이용하는 방법과 배경영상과 두 프레임간의 차를 이용하는 방법이 사용된다. 두 프레임간의 차를 이용하는 방법은 현재 프레임과 이전프레임의 차이 값의 임계값을 비교하여 '0'이면 변화된 화소가 없고, 임계값 이상일 경우는 화소가 변화된 것으로 한다. 배경영상과 두 프레임간의 차를 이용하는 방법은 기준 배경영상을 설정하고 현재 프레임과의 차이 값의 임계값을 비교하여 화소의 변화를 측정하는 방법으로 [그림 2-3]은 식(2.1)에 의한 차분영상 방법을 나타낸다 [2],[3],[4].

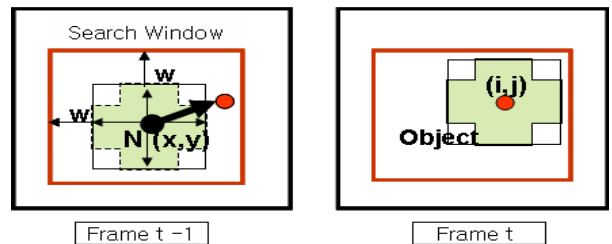
$$Difference_{img}(i, j) = |f_t(i, j) - f_{t-1}(i, j)| \dots \dots \dots (2.1)$$



[그림 2-3] 차분영상 방법

**(2) 블록정합 기법**

블록정합 기법은 [그림 2-4]와 같이 블록 내의 모든 화소가 같은 움직임 벡터를 갖는다는 조건 하에 이전 프레임으로부터 현재 프레임을 블록 단위로 추정하는 방법이다. 프레임을 일정한 크기의 블록(block)으로 나누고 현재 부호화 하고자 하는 블록과 가장 유사한 블록을 이전 프레임에서 찾아 그 블록으로 현재의 블록을 추정하는 방법이다. [5],[6] 그리고 N×N크기의 블록에 대해 프레임 당 최대 움직임 속도를 W화소라 하면 상하, 좌우로 W화소만큼 움직임이 가능하므로 데이터 탐색 영역의 크기는 (2W+N)×(2W+N)이 된다.



[그림 2-4] 블록정합 기법

이때, 블록간의 일치 정도를 평가하는 기준으로는 식(2.2)와 같이 MSE(Mean of Squared Error), MAD(Mean of Absolute Difference), SAD(Sum of Absolute Difference)가 있으나, 계산이 간편하고 하드웨어 구현이 용이한 SAD 방법이 주로 사용된다.

$$MSE(x, y) = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N [f(i, j, t) - f(i+x, j+y, t-1)]^2$$

$$MAD(x, y) = \frac{1}{N^2} \sum_{i=0}^N \sum_{j=0}^N |f(i, j, t) - f(i+x, j+y, t-1)|$$

$$SAD(x, y) = \sum_{i=0}^N \sum_{j=0}^N |f(i, j, t) - f(i+x, j+y, t-1)| \quad (2.2)$$

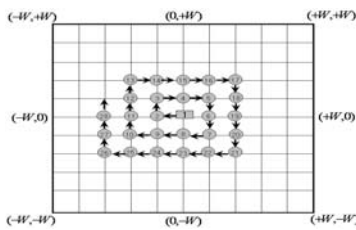
식(2.3)에서 현재 블록과 후보 블록의 크기는 N×N이고, 탐색 영역은 수평 수직 각각 -W~+W사이 값을 가진다. f(i, j, t)와 f(i+x, j+y, t-1)은 각각 현재 프레임의 현재 블록과 탐색 영역 내 후보 블록의 (i, j)번째 위치와 화소값을 나타낸다. 그리고 움직임 벡터는 SAD를 정합 기준으로 사용할 때 SAD(x, y)가 최소가 되는 움직임 벡터 V(x, y)는 식(2.3)과 같으며, 식(2.3)은 탐색 범위 (-W ≤ x, y ≤ +W)에서 SAD(x, y)를 계산하고 계산된 SAD(x, y) 값 중 가장 작은 값을 갖는 좌표 (x, y)가 최종 움직임 벡터 V(x, y)가 된다.

$$V(x, y) = \arg \min_{(-W \leq x, y \leq +W)} SAD(x, y), \quad (2.3)$$

**(3) 전역 탐색기법**

움직임 추정을 위한 가장 보편적인 방법은 전역 탐

색기법(full search)이다. [그림 2-5]와 같이 전역 탐색기법은 현재 프레임의 현재 블록을 이전 프레임의 탐색 영역 범위 전체와 일일이 비교하는 탐색 방법으로서 성능이 뛰어나고, 정합 정도를 계산하는 방법이 규칙적이어서 하드웨어 구현이 용이하다는 장점을 가지고 있다. 움직임 벡터의 범위가 수직, 수평 방향 모두  $-W \sim +W$ 이고, 정합 기준으로 SAD를 사용할 경우, 전역 탐색의 계산량은 현재 블록에 대해서 전역 탐색 알고리즘을 수행하려면  $(2W+1)2$ 번의 차의 절대값의 합 계산이 필요하므로, 프레임 전송 속도가 Frame/sec이고 한 프레임이  $n$ 개의 현재 블록으로 구성되어 있을 때, 차의 절대값의 합 계산은 초당  $(2W+1)2 \times n \times m$ 번 수행된다.



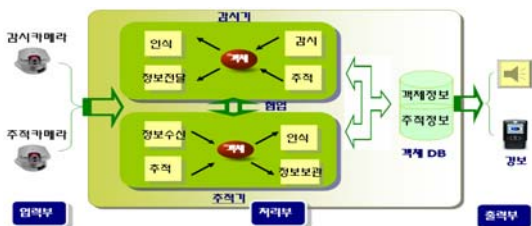
[그림 2-5] 전역 탐색 기법

현재 블록의 크기가  $N \times N$ 만큼의 뺄셈과 절대값 계산, 누산으로 구성되므로, 전역 탐색기법의 블록 탐색 연산량은 각각  $(2W+1)2 \times n \times m \times N^2$ 번이다. 따라서 전역 탐색 기법에서 소요되는 총 연산량은  $3(2W+1)2 \times n \times m \times N^2$ 이다. 이처럼 전역 탐색 방법은 알고리즘이 단순하여 하드웨어 구현이 용이하고 정확한 움직임 벡터를 찾을 수 있지만 영상의 크기나 탐색 범위가 증가하는 경우 발생하는 많은 계산량 때문에 실시간 시스템 구현에 문제점이 많다.

### 3. 다중카메라 기반의 특정객체 추적시스템

#### 3.1 제안시스템 개요

본 장에서는 제안하는 다차원 객체 추적 시스템에 대해서 기술한다. 다차원 객체 추적 시스템의 구성도를 중심으로 부분별 기능을 살펴본다. 특정 객체 영역을 추출하고 특정 객체를 추적하는 특정 객체 추적 알고리즘에 대해서 설계한다.



[그림 3-1] 다차원 객체 추적 시스템 구성도

#### 3.2. 입력부

입력부는 두 대의 카메라 즉, MC와 TC로부터 입력되어지는 영상을 처리부로 전송하는 부분이다. MC와 TC에서 입력되어지는 영상은 같은 공간을 연속적으로 촬영하는 곳으로 MC에 입력되는 특정객체에 대한  $x$ 값의 넓이는 TC에 입력되는 특정좌표  $x$ 값의 넓이와 같다. 또한 MC에 입력되는 영상에서 특정객체가 분포되어 있는  $y$ 영역은 TC에서 추적하는 특정객체의 원근을 계산할 수 있는  $z$ 좌표가 된다. MC에서 TC로 전달되는 정보는 앞서 언급한 위에서 말한  $x$ 값의 넓이와  $y$ 영역 즉, TC의  $z$ 정보가 된다.

#### 3.2. 처리부

감시기와 추적기로 이루어진 처리부는 입력부로부터 전송되어진 영상을 처리하는 부분으로 감시객체의 유·무와 특정객체가 유입되었을 때 특정객체의 윤곽선처리, 객체의 최소영역계산, MC에서 TC로의 정보 전송, 정보발생, 객체의 추적 등을 담당하는 곳이다.

감시기에서는 최초에 MC로부터 전송되어진 영상에서 감시대상 객체의 특징을 추출하기 위하여 감시대상객체의 크기, 색상정보, 모양정보 등을 저장한다. 감시대상객체의 유·무를 실시간으로 확인하는 기능 또한 감시기에서 MC로부터 전송되어진 영상에 의해 수행된다. 감시대상객체에 특정객체가 유입되는지의 판별은 감시대상객체보다 큰 영역을 감시대상영역으로 설정하고 이 영역에 특정한 객체가 유입되는지를 판별하여 감시대상객체의 도난을 미리 감지한다. 감시대상영역에 특정객체가 유입되면, 감시기에서 유입된 객체의 일부를 통하여 객체의 윤곽선을 추출하게 되고 추출된 윤곽선의 최소  $x1$ 값과 최대  $x2$ 값, 최소  $y1$ 값과 최대  $y2$ 값의 영역을 객체의 크기로 설정하고  $x1$ 과  $x2$ 정보를 추적기로 전송하고  $y1$ 과  $y2$ 의 정보는 좌표값을 전송하지 않고 중심값 즉,  $mid(y1, y2)$  값을 전송하여 추적기에서 객체의 원근에 대한 정보로 활용된다. 추적기에서는 MC에 특정객체가 유입되었을 때 감시기에 의해 TC의 동작을 시작하게 된다. TC의 동작시작과 동시에 출력부에 경보를 통보하고 저장장치에 최초로 입력되어진 TC의 영상정보를 저장하게 된다.

#### 3.1.3 출력부

출력부는 MC에 의해 감시객체영역에 특정객체가 유입되면 TC가 동작을 시작하도록 하는 정보를 전송하고, 시스템은 감시대상 객체의 분실 위험에 따른 경보시스템을 작동하게 된다. 또한 TC가 동작하고 있는 동안에 특정객체를 추적하는 과정에서 획득한 영상은 출력부에서 객체추적영상으로 획득하여 저장

하게 된다. 처리부에서는 추출된 객체영역 부분만을 저장하여 장치의 효율성을 높일 수 있다. 출력부의 구성요소는 객체감시영역에 특정객체의 침입을 알리는 경보장치와 추적객체의 영역을 저장하기 위한 데이터저장 장치로 이루어진다.

**4. 실험 및 분석**

**4.1 객체인식 연산량**

다차원 객체추적시스템에서 객체인식 연산량은 감시대상영역에 유입된 객체의윤곽선 추출에서 부터 객체의 유입여부를 인식하는데 까지의 연산에 참여하는 픽셀수로 정의한다. [표 4-1]은 제안시스템의 객체인식 연산량을 나타낸다. 연산량은 객체감시대상 영역에 객체의 유입여부를 판단하기 위한 연산량과 유입된 객체의 모양을 판단하기 위하여 객체인식에 필요한 연산량을 계산하였다. [표 4-1]에서와 같이 제안시스템은 입력영상 전체에 대해 객체유입을 판단하지 않고 객체감시영역의 변화에 대해서 특정객체의 유입을 판단하므로 연산에 참여하는 픽셀수를 줄일 수 있었다.

[표 4-1] 객체인식 연산량

연산량 실험횟수	연산량	최소 연산 픽셀	최대 연산 픽셀
1~100	147.3 픽셀	40	236
101~200	142.8 픽셀	43	239
201~300	147.9 픽셀	40	238
301~400	136.7 픽셀	40	237
401~500	144.3 픽셀	40	239
평균	143.8 픽셀	40.6	237.8

**4.2 객체추출 정확성**

객체추출의 정확성은 감시대상영역에 유입된 객체의 전체 크기를 인식하는 실험으로 객체가 감시대상영역에 유입되는 시간을 서로 다르게 하여 실험하였다. 실험결과 감시대상영역에 유입속도가 빠르면 객체이외의 영역까지 객체로 오인하는 결과를 얻었으나, 지속적인 추출을 수행하여 객체이외의 영역을 좁힐 수 있었다. [표 4-3]은 움직임 속도에 따라 객체의 영역이 아닌 부분과 객체로 판단하는 픽셀의 개수를 계산하였다. 객체영역의 픽셀 수는 정지영상에서 객체부분이 갖는 픽셀의 총 개수를 나타내었고, 오류영역 픽셀 수는 객체 이외의 영역을 객체로 오인하는 픽셀의 개수를 나타내었다. 객체 추출율은 객체영역의 총 픽셀 수와 오류영역의 픽셀 수의 비율을 나타내었으며, 실제 객체영역과 추출된 객체영역의 매칭은 평균 97.4%를 나타내었다.

[표 4-3] 객체추출의 정확성

연산량 실험횟수	평균 픽셀수	오류영역 픽셀수	객체추출율 [%]
1~10	2,029	101	95.25
11~20	1,843	22	98.82
21~30	2,323	62	97.40
31~40	3,070	42	98.65
41~50	1,889	70	96.42
평균	2,230.8	59.4	97.40

**5. 결론**

본 논문에서는 다차원 객체 추적 시스템을 제안하였다. 상단에서 객체를 감시하고 추적하는 MC와 MC의 x축 정보를 이용하여 전면의 영상에서 연산량을 최소화 하면서 객체를 추적하는 방법을 제안하였다. 제안한 시스템의 특징은 객체의 가려짐 현상이 일어나는 환경에서도 지속적인 객체의 추적이 가능하고 감시 및 추적 카메라를 상호 협업으로 기능케 하여 다차원 추적을 통해 일관성 있는 객체 추적이 가능하다. 또한 객체인식을 위해 전체 대상영상을 이용하지 않고 부분영상을 이용하여 객체인식 시간을 단축율을 높일 수가 있다. 시스템은 박물관, 보석상 등 고가의 상품 도난에 대한 사전 예방 감시가 가능하며, 짧은 시간으로 객체추출과 추적을 실행하므로 실시간 영상보안 시스템으로의 활용이 기대된다.

**참고문헌**

[1] Yen, Ching Chang and Shyang Chang, "A fast estimation algorithm on the hurst parameter of discrete-time fractional brownian motion," IEEE Trans on Signal Processing, Vol. 50, No. 03 pp. 0554~0559, March 2002.

[2] 박창준, 조상현, 최홍문, "잡음에 강건한 주목 연산자의 구현과 효과적인 다중물체 검출," 전자공학회 논문지 : SP , Vol. 8, No.1, pp. 89~95, 2001.

[3] 백주호, 이창수, 오해석, "배경화면 변화를 이용한 객체의 윤곽점 검출," 한국정보처리학회 춘계학술회의, Vol. 10, No. 1, pp. 563~566, 2003.

[4] 이창수, 전문석, "적응적 배경영상과 그물형 픽셀 간격의 윤곽점 검출을 이용한 객체의 움직임 검출," 한국통신학회 논문지, Vol. 30, No. 3, pp. 92~101, 2005.

[5] 최내원의, "동영상에서 배경영상을 이용한 실시간 객체추적," 한국정보과학회 추계학술발표 논문집, Vol. 29, No. 2, 2002.

[6] 지정규, 이창수, 오해석, "적응적 배경영상과 픽셀 간격을 이용한 움직임 검출," 한국데이터베이스학회 논문지, Vol. 10, No. 3, pp. 45~53, 2003.