

논문-02-07-2-11

객체 추적 카메라 제어를 위한 고속의 움직임 검출 및 추적 알고리즘

강 동 구*, 나 종 범*

A Fast Motion Detection and Tracking Algorithm for Automatic Control of an Object Tracking Camera

Dong-Goo Kang* and Jong Beom Ra*

요 약

능동 감시 카메라에서 얻어진 연속 영상에는 카메라의 움직임에 의해 발생하는 전역 움직임과 이동 물체의 국부 움직임이 동시에 존재한다. 따라서 이동 물체의 자동 추적을 위한 영상 기반의 실시간 감시 시스템의 구현을 위해 이동 물체의 국부 움직임만을 검출하고 추적할 수 있는 효과적인 알고리즘이 요구된다. 이 논문에서는 연속 영상의 차영상을 이용하는 빠르고 효율적인 움직임 검출 및 추적 알고리즘을 제안한다. 이 알고리즘은 우선 물체의 속도를 고려하여 이전 영상을 선택하고, 현재 영상과 선택된 이전 영상에 존재하는 전역 움직임을 빠르고 정확하게 추정하기 위해 신뢰성있는 소수의 정합 블록만을 선택하여 사용한다. 마지막으로 현재 영상과 전역 움직임이 보상된 이전 영상의 차영상을 얻고, 현재 영상과 차영상의 상관 관계를 이용하여 차영상에 존재하는 강한 잡음을 효과적으로 제거하여 이동 물체 영역을 추출한다. 팬틸트 유닛과 AMD 800MHz 프로세서가 내장된 PC로 구성된 능동 카메라 시스템에 제안한 알고리즘을 적용하였다. 이 시스템은 320×240 크기의 영상을 처리하며 수평 수직 방향의 ±20 탐색 영역에서 전역 움직임을 추정할 때 약 50 frames/sec의 속도로 움직임 검출이 가능하므로 빠른 이동 물체의 실시간 추적에 적합하다.

Abstract

Video based surveillance systems based on an active camera require a fast algorithm for real time detection and tracking of local motion in the presence of global motion. This paper presents a new fast and efficient motion detection and tracking algorithm using the displaced frame difference (DFD). In the proposed algorithm, first, a previous frame is adaptively selected according to the magnitude of object motion, and the global motion is estimated by using only a few confident matching blocks for a fast and accurate result. Then, a DFD is obtained between the current frame and the selected previous frame displaced by the global motion. Finally, a moving object is extracted from the noisy DFD by utilizing the correlation between the DFD and current frame. We implement this algorithm into an active camera system including a pan-tilt unit and a standard PC equipped with an AMD 800MHz processor. The system can perform the exhaustive search for a search range of ±20, and achieve the processing speed of about 50 frames/sec for video sequences of 320×240. Thereby, it provides satisfactory tracking results.

I. 서 론

자동화된 동영상 기반의 감시 시스템은 감시를 위한 값

비싼 노동력을 대신할 수 있기 때문에 관심의 대상이 되어 왔다. 특히 적은 수의 카메라를 이용하여 보다 넓은 영역을 감시하기 위하여 수평 수직 방향으로 회전 가능한 팬틸트 유닛(pan-tilt unit)이 장착된 능동(active) 카메라를 사용하는 것이 가격 측면에서 유리하다. 능동 카메라에서 얻어진 연속 영상에는 카메라의 움직임에 의해 발생하는 전역 움

* 한국과학기술원 전자전산학과
Dept. of EECS, KAIST

직입과 이동 물체의 국부 움직임이 동시에 존재한다. 따라서 이동 물체의 자동 추적을 위한 영상 기반의 실시간 감시 시스템의 구현을 위해서는 이동 물체의 국부 움직임을 검출하고 추적할 수 있는 고속의 알고리즘이 요구된다.

최근 동영상 정보를 이용하여 이동 물체의 영역을 검출하는 다양한 방법이 제안되었으며, 배경 차이, 색 혹은 모양 특징값, 그리고 움직임 정보를 이용하는 방법 등이 대표적이다. 배경 차이를 이용하는 방법^[1]에서는, 움직임을 수행하기 전에 서로 다른 각도에서 얻은 배경 영상을 하나의 모자이크 영상으로 미리 합성해 둔다. 그 후 모자이크 영상으로부터 현재 영상의 배경 영상을 찾아내고 이를 현재 영상과 비교하여 움직임 영역을 검출한다. 이 방법은 이동 물체의 속도, 모양에 관계없이 완전한 분할 결과를 제공하지만, 시간이 지남에 따라 배경 영역에 변화가 생기는 경우 실제 배경 영상과 이를 표현하는 모자이크 영상으로부터 찾아낸 배경 영상이 서로 일치하지 않으므로 올바른 검출 결과를 얻을 수 없다. 따라서 주기적으로 모자이크 영상을 갱신해야 하며 이는 많은 계산량을 요구한다. 물체의 색 혹은 모양 정보를 이용하는 방법^[2]에서는 색이나 모양 특징값 등을 이용해 이동 물체를 모델링하고 이 모델을 이용하여 가장 적합한 영역을 이동 물체의 영역으로 결정한다. 이 방법은 비교적 간단하지만 정확한 초기 이동물체 분할이 어렵고 시간에 따라 이동 물체의 색이나 모양이 변화하는 경우 추적에 실패하기 쉬울 뿐 아니라 한번 잘못 예측된 모델이 다음 추적 오류의 원인이 되는 오류 전파의 문제를 발생시킨다. 움직임 정보를 이용한 방법 중 optical flow를 이용하는 방법^[3]이 있다. 이 방법에서는 현재 영상의 optical flow를 모두 계산한 뒤 이를 전역 움직임 field와 국부 움직임 field로 나누어 이동 물체를 분리하며, 움직임 정보만을 사용하므로 물체의 색이나 모양의 변화에 강인한 결과를 얻을 수 있다. 하지만 정확한 움직임 field를 얻기 위하여 많은 계산량이 필요하고 특별히 설계된 하드웨어 없이는 실시간 구현이 매우 어렵다. 움직임 정보를 이용하는 또 다른 방법으로 차영상을 이용하는 방법이 있다^{[4]-[6]}. 두 장의 연속 영상을 이루는 화소의 시간에 따른 차이 값으로 정의되는 움직임 에너지가 큰 영역을 이동 물체 영역으로 결정하는 방법이다^[4]. 이 방법은 오류의 전파가 없고 구현이 매우 간단하여 널리 사용되고 있는 방법이다. 하지만 차영상을 얻기 위해 필요한 전역 움직임 추정 과정을 위해 여전히 많은 계산량이 요구되며 차영상에 존재하는 강한 잡음은 부정확한 검출 결과의 원인이 된다. 그리고 일정한 시간 간격의 연속 영상을 이용하여 차영상을 얻는 경우 느리거나 정

지한 물체의 정확한 검출을 기대하기 어렵다.

이 논문의 구성은 다음과 같다. II에서는 기존의 차영상 기반의 알고리즘을 소개하고 문제점을 지적한다. III에서는 II에서 지적한 문제점을 보완한 고속의 움직임 검출 및 추적 알고리즘을 제안한다. 이 알고리즘은 세 부분으로 이루어져 있다. 첫째, 이동 물체의 속도에 관계 없이 안정적으로 검출할 수 있도록 적절한 이전 영상을 선택한다. 둘째, 현재 영상과 선택된 이전 영상 간의 전역 움직임을 빠르고 정확하게 추정하기 위해 신뢰성있는 소수의 블록만을 선택하여 블록 정합을 수행한다. 셋째, 현재 영상과 차영상의 상관성을 이용하여 차영상에 존재하는 강한 잡음을 효과적으로 제거한다. IV에서는 제안한 방법이 기존의 방법에 비해 효과적임을 보이고 이를 실제 시스템에 적용한 결과를 제시한 후, V에서 결론을 맺는다.

II. 기존의 차영상 기반의 움직임 검출 및 추적 알고리즘^{[4]-[6]}

기존의 움직임 검출 및 추적을 위한 차영상 기반의 알고리즘은 크게 전역 움직임 추정(global motion estimation: GME), 전역 움직임 보상(global motion compensation), 그리고 움직임 영역의 국부화(localization)의 세 과정으로 이루어진다. 즉, 일정한 시간 간격을 가지는 두 연속 영상을 각각 이전 영상, 현재 영상이라고 할 때, 카메라의 움직임에 의해 발생한 두 영상 사이의 전역 움직임을 추정한다. 그리고 이전 영상과 전역 움직임이 보상된 이전 영상을 구하고 이 영상과 현재 영상의 차영상을 구한 뒤, 이를 이용하여 움직임 영역을 검출하는 국부화 과정을 거친다. 그러나 이러한 기존의 방법은 적어도 다음과 같은 세가지 문제를 가지고 있다.

첫째, 연속 영상에 나타난 이동 물체의 속도에 따라 정확한 이동 물체 영역의 검출이 불가능한 경우가 있다. 기존의 방법에서는 시간 간격이 일정한 연속 영상을 이용하여 차영상을 얻으므로 물체가 너무 빠르게 이동하면 차영상의 움직임 에너지가 너무 크게 나타나 물체의 영역 뿐만 아니라 배경 영역도 검출 결과에 포함될 수 있으며, 반면 물체가 너무 느리게 이동하면 차영상의 움직임 에너지가 너무 작게 나타나 물체의 일부분만 검출된다는 문제가 있다. 따라서 이동 물체의 속도에 적응적으로 적절한 이전 영상을 선택할 수 있는 방법이 요구된다.

둘째, 기존의 전역 움직임 추정 방법은 많은 계산량을

필요로 하며 실시간 추적 시스템에 적용하기 어렵다. 전역 움직임 추정 과정은 카메라의 움직임에 의해 연속된 두 영상에 나타난 정지한 배경 영역의 움직임을 2계수 혹은 6계수로 구하는 과정이다. 이 때, 일반적으로 현재 영상의 모든 화소를 이용하여 이전 영상의 일정한 탐색 영역에서 전역 탐색(full search)을 수행하기 때문에, 가장 간단한 2계수 전역 움직임 모델을 사용하더라도 많은 계산량이 요구된다. 이것을 프레임 정합(frame matching) 방법이라고 한다[5]. 현재 영상의 모든 화소를 정합에 이용하는 프레임 정합 방법과 달리 현재 영상의 일정한 격자 안에 위치하는 화소만을 정합에 이용하여 계산량을 줄일 수 있는 고정 격자 방법(fixed grid matching)이 제안되었다[6]. 여기에서 고정 격자란 일정한 간격을 두고 위치한 일정한 크기의 블록들의 집합을 의미한다. 하지만 이처럼 영상의 특징을 고려하지 않은 채 일부의 화소만을 정합에 사용하는 경우 정확한 움직임 추정이 불가능하다(III. 2 참조). 한편, 실시간 동영상 압축 등의 응용 분야에서는 화질의 저하를 감수하여 전역 탐색 방식을 사용하는 대신 탐색 영역에서의 탐색점수를 줄일 수 있는 다양한 고속 움직임 추정 방법을 많이 사용하고 있다. 하지만 탐색점수를 줄이면 전역 탐색의 결과보다 움직임 추정의 정확도가 떨어지며 이는 움직임 검출 실패의 직접적인 원인이 될 수 있어 바람직하지 않다. 따라서 정확하면서도 계산량이 적은 전역 움직임 추정 기법이 요구된다.

셋째, 차영상에 자주 나타나는 강한 잡음은 움직임 검출 실패의 주요 원인이 된다[4]-[5]. 차영상의 밝기값(intensity value)이 큰 영역이 움직임 영역으로 간주되므로 정지한 배경 영역의 밝기값은 작을수록, 움직임 영역의 밝기값은 클수록 정확한 검출이 가능하다. 그러나 배경 영역을 완벽히 보상하는 것은 불가능하며 작은 전역 움직임 추정의 오류에도 비교적 큰 밝기값을 가지는 강한 잡음이 자주 발견된다[4]-[5]. 따라서 이 잡음을 제거하기 위해 다음과 같은 후처리 방법이 주로 사용된다[4]-[5]. 먼저 차영상의 각 화소를 적당한 문턱값(threshold value)으로 이진화(binartization)하여 영상을 간략화 한 후, 이진화 영상에 dilation 과 erosion의 연속 연산으로 정의되는 형태학적 열기(morphological opening) 연산을 수행하여 작은 오류 영역이나 에지 주변에 존재하는 큰 오류 영역 등을 제거한다. 그러나 형태학적 필터링 연산은 계산량이 많아 실시간 연산에 부담이 될 뿐 아니라 강한 열기를 수행하면 이동 물체의 일부 영역이 제거되기 쉽고 약한 열기를 수행하면 에지를 따라 분포하는 큰 오류 영역을 제거하지 못하게 된다. 특히 배경 영역이

복잡하여 에지 성분이 많을수록 차영상의 오류는 더욱 제거되기 어렵다. 따라서 차영상에 나타나는 강한 잡음에 강인한 움직임 검출 방법이 요구된다.

III. 제안한 알고리즘

차영상을 이용하는 움직임 검출 및 추적 방법은 구현이 간단하고 오류 전파의 문제가 없으며 이동 물체의 색이나 모양 등의 특징에 관계 없이 적용할 수 있는 장점이 있으나 II에서 이미 지적했듯 여러 문제점들을 가지고 있다. 제안한 알고리즘에서는 이러한 문제점들을 해결하여 실시간 추적 시스템에 적용할 수 있도록 하였다. 그림 1은 제안한 알고리즘의 흐름을 보여준다. 먼저 현재 영상을 획득하여 저장한 후 이동 물체의 속도를 고려하여 이미 저장되어 있는 적절한 이전 영상을 선택한다. 그리고 현재 영상과 선택된 이전 영상 사이의 전역 움직임을 적은 계산량으로 추정된 뒤 현재 영상과 전역 움직임이 보상된 이전 영상의 차영상을 구한 후, 차영상에 존재하는 강한 잡음에 강인한 방법으로 이동 물체를 검출한다.

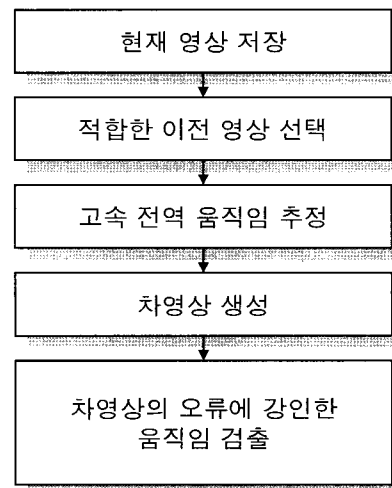


그림 1. 제안한 방법의 흐름도
Fig. 1. Flowchart of the proposed algorithm

1. 이동 물체의 속도를 고려한 이전 영상의 선택 알고리즘

그림 2는 이전 영상을 선택하는 방법을 보여준다. 그림 2(a)는 이동 물체의 속도에 관계없이 현재 영상과 일정한 시간 간격을 가지는 이전 영상을 선택한 결과이다. 예를 들

어 14번 현재 영상내의 움직임 영역을 검출하기 위해 일정한 시간 간격을 가지는 13번 이전 영상을 선택해 차영상을 구한다. 이 방법은 매우 간단하지만, 14번 영상에 나타난 이동 물체는 매우 느리게 운동하므로 인접한 13번 영상과의 차영상에는 움직임이 거의 발견되지 않아 결국 움직임 검출에 실패한다. 반면 그림 2(b)는 이동 물체의 속도를 고려하여 이전 영상을 택한 결과를 보여준다. 예를 들어 14번 현재 영상의 움직임 영역을 검출하기 위해 이동 물체의 속도가 느린 것을 고려하여 영상간 이동 물체의 충분한 움직임이 보장되는 9번 영상을 이전 영상으로 선택해 차영상을 구하므로 움직임의 검출이 가능해진다.

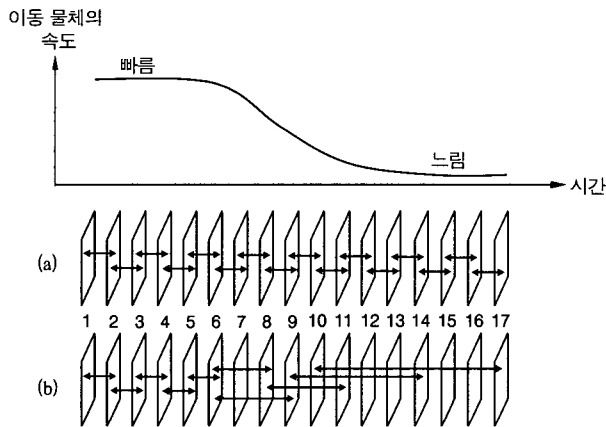


그림 2. 이동 물체의 속도를 고려한 이전 영상의 선택 (a)이동 물체의 속도에 관계 없이 일정 시간 간격을 가지는 이전 영상을 선택하는 경우 (b)이동 물체의 속도에 적응적으로 이전 영상을 선택하는 경우. 물체의 속도가 느리면 물체의 속도가 빠른 경우보다 시간적으로 멀리 떨어진 영상을 이전 영상으로 선택한다.

Fig. 2. Selection of a previous frame (a)Interval between the current and the selected previous frames is constant. The previous frame is selected regardless of object velocity (b)The interval between the current and the selected previous frames is adaptive to object velocity. The frame interval becomes longer (shorter), if the object velocity is slower (faster).

이 절에서는 현재 영상과 이전 영상 사이의 이동 물체가 항상 적절한 움직임을 갖도록 이전 영상을 선택하는 방법을 제안한다. 먼저 배경 영역의 전역 움직임을 2계수로 모델링하여 표현할 수 있다고 가정하자. 임의의 두 영상 f_1 과 f_2 사이의 이동 물체 영역의 움직임 에너지 E_{f_1, f_2} 를 다음과 같이 정의하자.

$$E_{f_1, f_2} = MV_{obj} - MV_g \quad (1)$$

단, MV_{obj} 와 MV_g 는 각각 영상 f_1 과 f_2 사이에서 존재하는 이동 물체의 국부 움직임 벡터와 배경 영역의 전역 움직임 벡터를 나타낸다. MV_g 가 0인 경우 MV_{obj} 는 연속 영상 사이에 나타난 물체의 실제 움직임 벡터를 의미한다. 식 (1)은 능동 카메라 시스템에서 이동 물체의 실제 움직임의 크기를 표현하며 차영상에 나타나는 움직임 에너지의 양을 의미한다. 이 때, 정확한 국부 움직임의 크기 MV_{obj} 를 구하기 위해 많은 계산량이 요구된다. 그러나 만일 카메라의 연속적인 이동 물체 추적이 가능하다고 가정하면 이동 물체는 항상 현재 영상 중심의 근처에 위치하게 되므로 $MV_{obj} \approx 0$ 으로 가정할 수 있으며, 이 가정에 의해 식 (1)은 $E_{f_1, f_2} \approx MV_g$ 로 근사화 될 수 있다. 제안한 방법에서는 이중 버퍼를 사용하여 항상 적절한 E_{f_1, f_2} 을 가지는 이전 영상을 선택할 수 있도록 하여 차영상에 항상 적당한 움직임이 나타나도록 하였다. 이제 영상을 저장하기 위한 두 버퍼를 test 및 reference 버퍼라고 부르고, f_t 와 f_r 을 각각 test 버퍼와 reference 버퍼에 저장된 이전 영상이라고 하자. 여기에서 f_t 는 앞으로 이전 영상이 될 후보 영상을 의미하며 f_r 은 현재 움직임 검출을 위해 사용되는 이전 영상을 의미한다. 또한 GMV_t 와 GMV_r 는 각각 현재 영상과 f_t 와 f_r 사이의 전역 움직임 벡터(global motion vector: GMV)를 의미하며 GMV_b 는 f_t 과 f_r 사이의 GMV를 의미한다. 또한 PMV_t 와 PMV_r 는 각각 GMV_t 와 GMV_r 을 얻는 과정에서 계산량을 줄이기 위해 사용되는 PMV (predicted motion vector)를 의미한다. 제안한 방법은 다음과 같이 요약된다.

Step 1: (초기화) 첫 영상을 test 및 reference 버퍼에 저장하고 PMV_t , PMV_r , GMV_t , GMV_r , 그리고 GMV_b 를 0으로 설정한다.

Step 2: 현재 영상 f_c 를 얻고 PMV_t 를 이용하여 f_c 와 f_t 간의 GMV_t 를 계산한다. 그 후 $PMV_t = GMV_t$, 그리고 $PMV_r = GMV_b + GMV_t$ 로 설정한다.

Step 3: 만일 $E_{f_c, f_t} = GMV_t > d$ 이면 f_c 와 f_t 를 각각 test 및 reference 버퍼에 저장한다. 이때, d 는 적절히 선택된 양의 상수이다. 그리고 $GMV_b = GMV_t$, $PMV_t = 0$, $PMV_r = GMV_t$ 로 설정한다.

Step 4: 만일 이 시점에서 움직임 검출을 하고자 한다면 현재 영상을 f_c 로 하고 이전 영상을 f_r 로 선택하여 f_c 와 f_r 을 대상으로 $PMV_t \neq 1$ 의 범위에서 GMV 를 구한 후 이 전역 움직임이 보상된 f_r 과 f_c 의 차영상을 얻어 움직임 검출을 수행한다.

Step 5: Step 2, 3와 4를 반복한다.

이동 물체가 한 방향으로 이동할 경우 위의 과정을 수행하는 동안 항상 $E_{f_s, f_r} \approx d, 0 < E_{f_s, f_r} < d$ 을 만족하며 결과적으로 $d < E_{f_s, f_r} < 2d$ 의 식을 만족시킨다. 이 식은 이동 물체가 이동하다 정지한 상황에서도 유효하다. 여기에서, d 값은 차영상에 나타나는 평균 움직임 에너지와 비례하며 d 값이 클수록 차영상의 움직임 에너지의 변화폭이 커진다. 따라서 안정적인 결과를 제공하는 최소의 d 값을 실험적으로 결정할 필요가 있다. 결국, 제안하는 알고리즘은 기존의 방법과 달리 이동 물체의 속도에 관계없이 적절한 이전 영상을 선택함으로써 항상 적당한 크기의 움직임 에너지를 가지는 차영상을 얻을 수 있다.

2. 전역 움직임 추정을 위한 고속의 알고리즘

이 절에서는 현재 영상과 선택된 이전 영상 사이에 존재하는 전역 움직임을 추정하기 위한 고속의 알고리즘을 제안한다. II에서 이미 지적하였듯이 프레임 정합 방법은 영상의 모든 화소를 정합에 이용하므로 정확한 결과를 얻을 수 있으나 많은 계산량을 요구하며 고정 격자 방법은 일정한 위치에 있는 블록 안의 화소만을 정합에 이용하므로 계산량은 줄일 수 있으나 정확한 결과를 기대할 수 없다. 제안한 방법은 블록 정합 알고리즘에 기반을 두고 있으며 몇 개의 신뢰성있는 블록만을 선택적으로 사용함으로써 정합에 사용되는 화소의 수를 크게 줄여 필요한 계산량을 크게 줄이면서도 정확한 결과를 기대할 수 있다. 이 논문에서는 전역 움직임을 2계수(2차원 움직임 벡터)로 표현할 수 있다고 가정하였으며 정합을 위한 유사 척도(similarity criterion)로서 MAD(mean absolute difference)를 사용한다. 만일 하나의 블록 B만을 사용하여 예측한 전역 움직임 벡터 GMV는 다음과 같이 표현할 수 있다.

$$GMV = \underset{(mv_x, mv_y)_{x, y} \in B}{\operatorname{arg\,min}} \sum |I_t(x, y) - I_{t-\delta t}(x + mv_x, y + mv_y)| \quad (2)$$

단, $I_t(x, y)$ 와 $I_{t-\delta t}(x, y)$ 는 각각 현재 영상과 이전 영상의 (x, y) 에서의 휘도(luminance)를 나타낸다. 그러나 식 (2)를 이용하여 잘못된 GMV를 얻을 수 있는데, 정합 블록 B가 간극 문제(aperture problem)를 가지거나 또는 이동 물체의 영역을 포함하고 있는 경우가 이에 해당한다. 한 블록을 이용하여 움직임을 추정할 때 이차원 움직임 벡터를 유일하게 결정할 수 없을 때 그 블록은 간극 문제가 있다

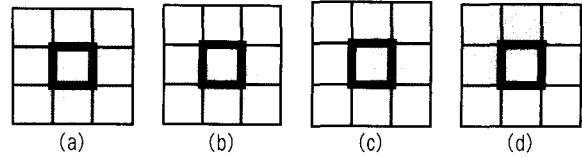


그림 3. 간극 문제를 갖는 블록의 예
Fig. 3. Examples of blocks having an aperture problem

고 말할 수 있다^[7]. 그림 3은 간극 문제를 가지는 블록의 예를 보여준다. 그림 3(a), (b), (c), (d)의 블록들을 이용하여 움직임을 추정할 때 각각 모든 방향에서, 수평 방향에서, 수직 방향에서, 그리고 45도 방향에서 움직임 벡터를 유일하게 결정할 수 없다. 만일 어떤 블록 내부 화소의 밝기값이 모든 방향에서 변화가 크다고 가정하면, 극소점을 중심으로 매우 급한 경사를 가지는 MAD 곡선을 얻을 수 있고 움직임 벡터를 유일하게 결정할 수 있으므로 이 블록에는 간극 문제가 존재하지 않을 것이다. 그러나 이러한 특징을 가지는 블록은 실제 영상 안에서 거의 발견되지 않는다. 따라서 제안한 방법에서는 두 개의 블록, 즉 수평 방향의 에지 성분이 강한 블록(B_h)과 수직 방향의 에지 성분이 강한 블록(B_v)을 선택하고, 이들을 동시에 이용하여 전역 탐색을 수행한다. B_h 와 B_v 를 결정하기 위해서 임의의 블록에 대하여 각 화소 위치에서 수평 수직 방향의 gradient magnitude의 합을 계산한 뒤 이들을 비교한다. B_h (B_v)를 이용하여 전역 움직임을 추정할 때 식 (2)을 적용하여 블록 정합을 수행하면 수직(수평) 방향으로 급한 경사를 가지는 MAD 곡선을 얻을 수 있으므로, B_h 와 B_v 를 동시에 정합에 사용하면 모든 방향에서 급한 경사를 가지는 MAD 곡선을 얻을 수 있으며 간극 문제를 피할 수 있다. 그림 4는 이러한 사실을 설명해 준다. 이 때, 블록 쌍 B_h 와 B_v 를 동시에 이용하여 예측한 GMV는 다음과 같이 표현된다.

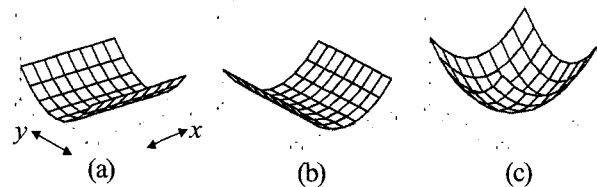


그림 4. (a) B_h 와 (b) B_v 의 MAD 곡선. (c) B_h 와 B_v 를 동시에 정합할 경우 얻어지는 MAD 곡선
Fig. 4. MAD curves of (a) B_h and (b) B_v , and (c)the MAD curve obtained from joint matching of B_h and B_v

$$\begin{aligned} \text{GMV} = \arg \min_{(mv_x, mv_y)} & \sum_{x, y \in B_h} w_h |I_t(x, y) - I_{t-\delta}(x + mv_x, y + mv_y)| \\ & + \sum_{x, y \in B_v} w_v |I_t(x, y) - I_{t-\delta}(x + mv_x, y + mv_y)| \end{aligned} \quad (3)$$

여기서, w_h 와 w_v 는 블록 B_h 와 블록 B_v 의 MAD 곡선에 해당하는 가중치 상수이며, 해당 블록의 각 화소에서의 gradient magnitude의 합의 역수로 정의된다. 이때 가중치 상수를 도입함으로써 각 블록의 MAD 곡선이 최종 MAD 곡선에 동등하게 기여할 수 있다. 만일 정합에 사용하는 블록 쌍의 개수를 증가시키면, 알고리즘의 정확도는 향상되나 복잡도는 증가하게 될 것이다.

만일 선택된 정합 블록들 중 어느 하나라도 이동 물체를 포함한다면, 이동 물체의 국부 움직임 방향으로 치우친 움직임 벡터를 GMV로 결정하게 된다. 이를 피하기 위하여 일단 바로 이전 이동 물체 영역의 위치에 존재하는 블록은 전역 움직임 추정을 위한 정합 블록으로 선택하지 않는다. 또한 움직임 추정 후 다음의 조건을 만족하지 않는다면 계산된 GMV는 실제 전역 움직임을 구하지 못하고 국부 움직임 방향으로 치우쳤다고 판단한다.

$$\text{MAD}_{\min}(B_{h_1}, B_{v_1}, \dots, B_{h_n}, B_{v_n}) \leq \left(\sum_{k=1}^n \text{MAD}_{\min}(B_{h_k}) + \text{MAD}_{\min}(B_{v_k}) \right) + \alpha. \quad (4)$$

여기서 α 는 적절히 선택된 양의 상수를 나타내고, n 은 전역 움직임 추정에 사용된 수평 수직 블록 쌍의 개수를 의미하며 B_{h_k} 및 B_{v_k} 는 각각 k 번째 블록 쌍을 이루는 수평 및 수직 블록을 의미한다. 또 $\text{MAD}_{\min}(B)$ 는 선택된 블록 B 만을 이용하여 정합할 때 얻을 수 있는 최소의 MAD 값을 나타내고 $\text{MAD}_{\min}(B_{h_1}, B_{v_1}, \dots, B_{h_n}, B_{v_n})$ 는 선택된 모든 블록 쌍을 이용하여 동시에 정합할 때 얻을 수 있는 최소의 MAD 값을 나타낸다. 만일 일부 블록이 이동 물체를 포함하는 경우 각 블록의 움직임 벡터가 서로 달라 우향의 값에 비해 좌향의 값이 크게 증가한다. 따라서 식 (4)를 이용해 선택된 일부의 블록이 이동 물체를 포함하여 전역 움직임 벡터를 잘못 예측하는 것을 막을 수 있다. 여기에서 상수인 α 값을 너무 크게 설정하면 전역 움직임 벡터를 잘못 예측하더라도 식(4)가 만족될 수 있으며 α 값을 너무 작게 설정하면 전역 움직임 벡터를 제대로 예측하더라도 식 (4)가 만족되지 않을 수 있으므로 이를 고려하여 적절한 값으로 설정해야 한다. 만일 식 (4)를 만족하지 못해 전역

움직임 추정이 잘못된 경우로 판단되면 새로운 블록쌍을 선정하여 다시 추정을 시도한다. 식 (4)에서 사용되는 $\text{MAD}_{\min}(B_h)$ 와 $\text{MAD}_{\min}(B_v)$ 의 계산량은 거의 무시할 수 있는데, $\text{MAD}_{\min}(B_h, B_v)$ 를 구하는 과정에서 쉽게 얻을 수 있기 때문이다. 또한 선택된 블록이 영상 안에 골고루 퍼져 있다면 그 블록이 현재 이동 물체 영역을 포함할 확률을 줄일 수 있을 것이다. 따라서 선택 가능한 후보 블록들 중에서 서로의 거리의 역수의 합이 최소가 되는 블록 쌍을 찾는다.

결국 제안한 방법에서는 영상의 특성을 고려하여 전역 움직임 추정에 적합한 최소의 블록들을 선택하여 정합에 사용하므로 계산량 측면에서는 프레임 정합 방법보다 유리하고 정확도 면에서는 고정 격자 방법에 비해 유리함을 알 수 있다. 실험 결과 두 세 개의 블록 쌍을 이용해서 정확한 전역 움직임 추정이 가능함을 확인할 수 있다(IV 참조).

3. 차영상의 잡음에 강한 움직임 검출 알고리즘

전역 움직임 추정에 사용한 움직임 모델(예를 들어 2계수 또는 6계수 모델 등)이 실제 배경의 움직임을 정확히 나타내지 못하는 경우 배경의 에지(edge) 주변에 큰 밝기 값의 강한 잡음이 발생한다. 그리고 배경 조명이 급격히 변화하는 영역에 해당하는 차영상의 밝기값은 커진다. 또한 카메라의 자동초점 조절기능이 동작하여 입력 영상 전체가 갑자기 흐려지거나, 공간 주파수가 큰 영역에서 알리아싱(aliasing)이 발생하는 경우, 그리고 정수 단위의 움직임 추정 기법에서 필연적으로 발생하는 부화소(sub-pixel) 추정 오류 등이 원인이 되어 배경의 에지(edge) 주변에 강한 잡음이 발생하기도 한다. 동영상 압축 분야에서는 이미 이러한 관찰을 기반으로 영상의 gradient magnitude와 움직임 보상 에러(motion compensated error)와의 상관 관계를 도출하고 이를 이용한 효율적인 압축기법을 제안하였다^[8]. 이 논문에서는 gradient magnitude와 전역 움직임 보상 에러와의 상관 관계를 이용하여 차영상의 오류에 강한 움직임 검출 알고리즘을 제안한다.

그림 5는 위에서 설명한 차영상에 나타나는 강한 잡음의 경향을 보여주는 대표적인 예이다. 그림 5(a)는 현재 영상을, 그림 5(b)는 현재 영상과 전역 움직임이 보상된 이전 영상과의 차영상을 나타내며 그림 5(c)는 현재 영상의 gradient magnitude 영상을 나타낸다. 그림 5(b)에서 배경 영역임에도 큰 밝기값으로 나타나는 오류의 대부분이 에지 부근에서 발생되며 또한 gradient magnitude 영상과 비슷

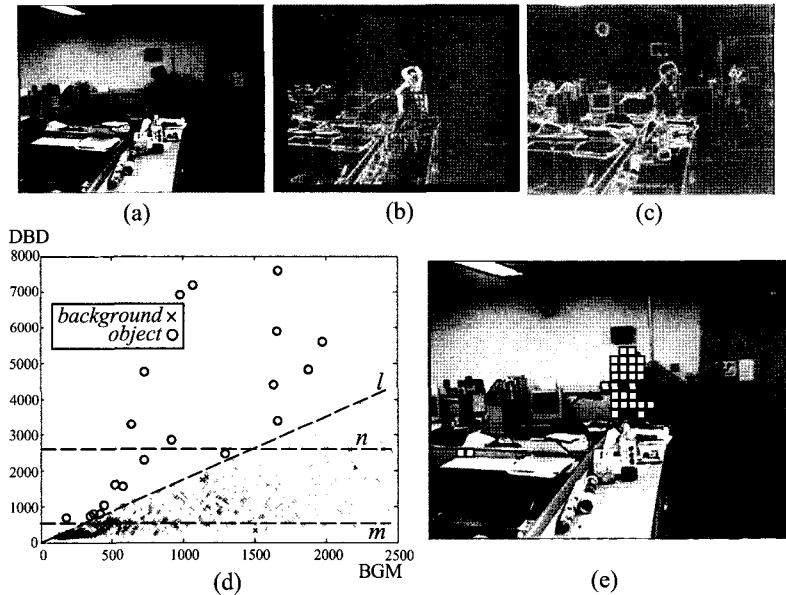


그림 5. (a)현재 영상, (b)차영상, (c)gradient magnitude, (d)(BGM, DBD)의 분포, (e)검출된 8×8 블록들. (d)에서 검출된 블록은 'o'로 검출되지 않은 블록은 'x'로 나타내었다.

Fig. 5. (a)Current frame, (b)DFD, (c)gradient magnitude, (d)distribution of (BGM, DBD), and (e)detected 8×8 blocks. In (d), detected blocks are marked with 'o', and non-detected blocks with 'x', respectively.

한 경향을 가짐을 알 수 있다. 이제 현재 영상 및 차영상을 $N \times N$ 크기의 블록 영상들의 합으로 표현할 수 있다고 하자. 그리고 DBD(displaced block difference) 및 BGM(block gradient magnitude)은, '차영상을 이루는 $N \times N$ 크기의 블록 내부 화소의 밝기값의 합' 과 '현재 영상을 이루는 $N \times N$ 크기의 블록 내부의 gradient magnitude의 합'을 나타낸다고 하자. 그림 5(d)는 배경 영역의 한 블록에 해당하는 순서쌍 (DBD, BGM)이 일정한 기울기를 갖는 직선 l에 의하여 제한되는 영역(하단부)에 주로 분포한다는 것을 보여준다. 즉, 정확한 비례 관계는 아닐지라도 gradient magnitude가 작은 (큰) 배경 영역에서는 차영상의 값도 작게(크게) 나타나는 경향을 발견할 수 있다. 마찬가지로 그림 5(d)로부터 이동 물체의 영역을 포함하는 블록의 DBD 값이 이 블록의 BGM 값과 비슷한 BGM 값을 갖는 배경 영역의 블록의 DBD 값보다 크다는 것을 알 수 있다. 물체의 움직임이 전역 움직임과 달라 대개 이동 물체 영역의 차영상이 배경 영역의 오류 영상보다 큰 값을 가지기 때문이다. 따라서 정지된 배경 영역과 이동 물체 영역을 나누기 위한 기준으로 BGM 축과 수평인 직선 m 또는 직선 n보다 직선 l이 더욱 적합하다. 이진 형태학적 필터링을 포함한 기존의 검출 방법에서는 적당한 상수값(직선 m 또는 n을 의미)으로 차영상의 밝기값을

임계화(thresholding)하기 때문에 강한 잡음에 민감한 특성을 가졌으나 제안하는 알고리즘에서는 그림 5(d)에서처럼 현재 영상의 gradient magnitude 값과 차영상 간의 상관 관계를 이용하여 잡음에 강인한 움직임 검출이 가능하도록 하였다. 제안하는 알고리즘은 다음과 같이 요약된다.

- Step 1: 현재 영상을 $N \times N$ 크기의 블록으로 나누고 각 블록의 BGM이 일정한 문턱값보다 크면 에지 블록(edge block) B_e 로 그렇지 않으면 평탄 블록(smooth block) B_s 로 결정한다. 그리고 에지 블록에 대해서는 DBD_e/BGM_e 의 평균(μ_e) 및 분산(σ_e)을 구하고 평탄 블록에 대해서는 DBD_s 의 평균(μ_s) 및 분산(σ_s)을 구한다 (단, DBD_e 와 DBD_s 는 각각 B_e 와 B_s 의 DBD이고 BGM_e 는 B_e 의 BGM이다). 이때, 바로 이전에 이동 물체의 영역으로 검출된 영역을 포함하는 블록은 위의 과정에서 제외시킨다.
- Step 2: 에지 블록의 경우 해당 DBD_e/BGM_e 값을 ($\mu_e + p\sigma_e$)으로 임계화하고, 평탄 블록의 경우 해당 DBD_s 값을 ($\mu_s + p\sigma_s$)으로 임계화하여 임계값보다 크면 움직임이 있는 블록으로 결정한다. 이때, p와 q는 미리 정해 놓은 양의 실수이다.
- Step 3: 움직임이 있는 블록을 8-neighbor 기준에 의하여 연결하여 서로 다른 그룹으로 묶는다.
- Step 4: 후보 그룹 중 DBD의 합(움직임 에너지의 크기)이 가장 큰 그룹을 선택하여 이동 물체의 영역으로 정한다.

Step 1에서 사용되는 각 블록의 BGM 값은 전역 움직임 추정시 이미 각 화소에서의 gradient magnitude를 계산하여 저장하였으므로 이를 이용하여 계산량을 줄일 수 있다. 그림 6은 위의 과정을 거치면서 얻는 단계별 결과의 예를 보여준다. 그림 6(c)는 제안한 알고리즘을 적용하여 최종적으로 얻은 결과를 보여주며 이동 물체를 포함하는 최소 사각형을 의미한다. 최종 사각형으로부터 이동 물체의 면적 및 위치 정보를 얻고 이를 이용하여 카메라를 제어함으로

써 이동 물체가 화면의 중심에 위치하게 하면서 추적할 수 있도록 한다. 결국 제안한 방법에서는 잡음이 강한 차영상으로부터 움직임 영역을 빠르고 효과적으로 찾기 위하여, 현재 영상의 BGM과 차영상의 DBD간의 상관 관계를 이용한 효과적인 임계화를 수행할 수 있다. 특히 제안한 방법은 매우 복잡한 배경(cluttered scene)을 가지는 환경에서 우수한 성능을 보일 것으로 기대된다.

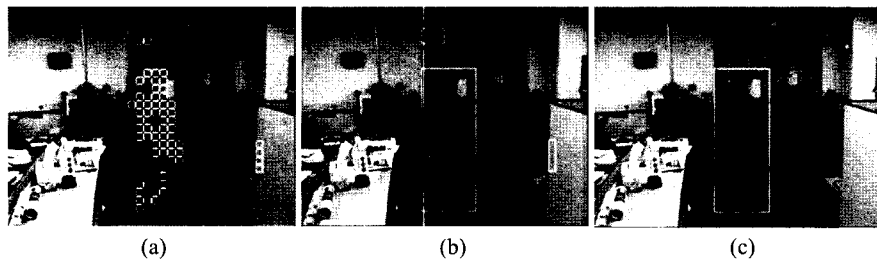


그림 6. 제안한 움직임 검출 알고리즘의 단계별 결과. (a)Step 2의 임계화 결과. (b)Step 3의 결과. (c)Step 4의 결과(최종)
 Fig. 6. Results of the proposed localization algorithm. (a)Result of step 2, (b)result of step 3, and (c)result of final step 4

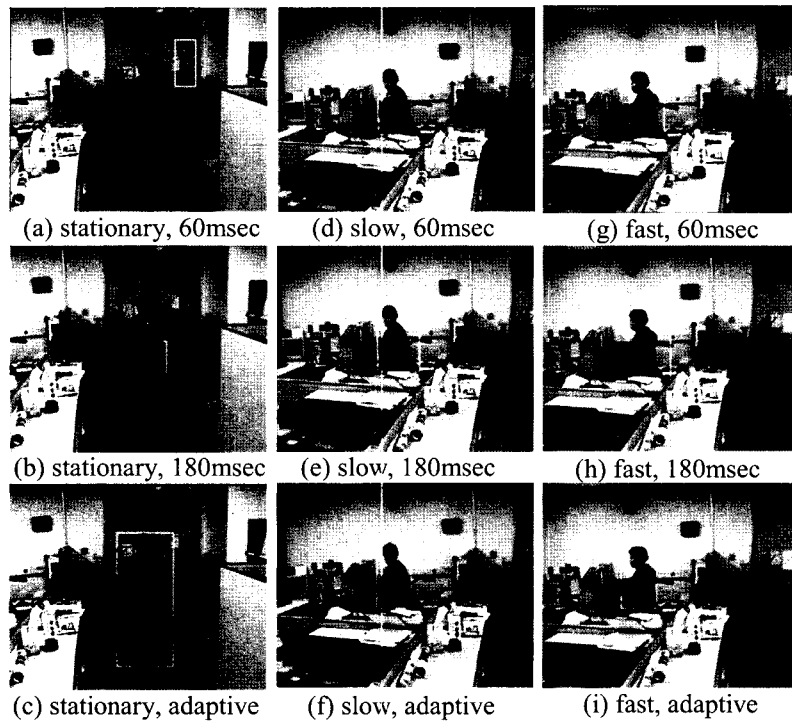


그림 7. 왼쪽 열부터 차례로 각각 정지한, 느린, 빠른 물체의 검출 결과. 처음 두 행은 고정된 시간 간격(60 or 180ms)의 이전 영상을 사용한 경우의 움직임 검출 결과이며 마지막 행은 제안한 선택적 방법에 의해 선택된 이전 영상을 사용한 경우의 움직임 검출 결과이다

Fig. 7. Localization results of stationary, slow, and fast objects from the left to right column, respectively. The first two rows use a previous frame with a fixed interval (60 or 180ms), and the last one uses a previous frame selected by the proposed adaptive scheme

IV. 실험 결과

III. 1에서 제안한 이전 영상의 선택 방법을 적용한 경우와 그렇지 않은 경우 이동 물체의 검출 결과를 그림 7에 나타내었다. III. 1의 Step 3에서 $d = 2$ 로 설정하였다. 그림 7은 제안한 방법을 적용할 때 정지해 있거나 느리거나 빠른 물체에 대하여 정확한 검출 결과를 얻을 수 있음을 보여준다. 반면, 고정된 시간 간격을 가지는 연속 영상을 이용하는 기존의 방법을 적용할 경우 정지한 물체를 전혀 검출하지 못한다. 또한 시간 간격을 길게(180msec) 설정하면 느린 물체는 정확하게 검출되는 반면 빠른 물체의 경우 현재 움직임 영역 이외에 이전 움직임 영역까지 포함된 부정확한 결과를 얻는다. 시간 간격을 짧게(60msec) 설정하면 느린 물체의 경우 부정확한 검출 결과를 얻게 된다.

제안한 전역 움직임 추정 방법을 프레임 정합 방법^[5]과 고정 격자 정합 방법^[6]과 비교하였다. 비교 대상의 모든 알고리즘에서 2계수 전역 움직임 모델을 사용하였고 정확한 정합을 위하여 전역 탐색 방법을 사용하였으며 움직임 추정을 위한 블록의 크기는 16×16 으로 하였다. 300장의 320×240 크기의 영상에 대하여 실험하였다. 충분한 탐색 영

역(± 10)에서 프레임 정합 방법을 사용하여 예측한 GMV를 실제 GMV라고 간주하고 이와 비슷할수록 정확한 결과로 판단하였다. 제안한 방법과 고정 격자 정합 방법의 결과를 실제 GMV와 비교한 결과를 표 1에 나타내었다. 234개 블록을 모두 사용한 프레임 정합 방법과 4개의 블록만을 사용한 제안한 방법의 결과가 약 98% 일치하였으며, 불일치하는 경우에도 실제 GMV와 비교하여 1화소 이하의 차이를 보였다. 또한 고정 격자 정합 방법에 비해 항상 정확한 결과를 얻을 수 있었다.

표 1. ± 10 탐색 영역에서 다양한 전역 움직임 추정 알고리즘의 계산량 및 정확도 비교

Table 1. Complexity and accuracy comparison of various global motion estimation algorithms for a search range of ± 10 .

방 법	사용한 블록 수	탐색점 수	실제 GMV와의 일치율(%)	± 1 오차 범위에서 실제 GMV와의 일치율(%)
프레임 정합 방법	234	103,194	100	100
고정격자 방법	2	882	61.07	83.22
	4	1,764	86.58	99.33
	6	2,646	87.25	99.33
제안한 방법	2	882	96.64	100
	4	1,764	97.99	100
	6	2,646	97.99	100

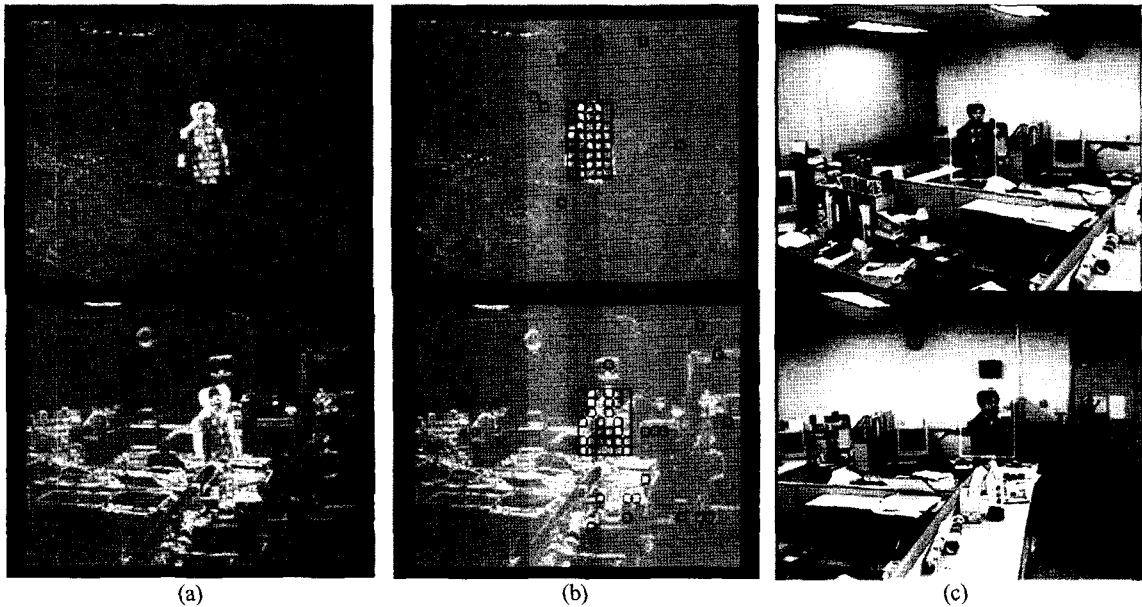


그림 8. 제안한 움직임 검출 알고리즘을 적용한 결과. 첫번째 행은 차영상의 잡음이 비교적 작을 때의 검출 결과를 나타내며 두 번째 행은 차영상의 잡음이 비교적 강할 때의 검출 결과를 나타낸다. (a)차영상, (b)작은 사각형과 큰 사각형은 각각 임계화 결과와 그룹화 결과를 나타낸다, (c)최종 검출 결과

Fig. 8. Results of the proposed localization algorithm. The first low presents the detection results with weak noise of difference image, and the second low presents the noise with strong noise of difference image. (a)Difference image, (b)small and large blocks present results of thresholding and grouping, respectively, (c)final object bounding box

그림 8은 III. 3에서 제안한 움직임 검출 알고리즘의 결과를 보여준다. III. 3의 Step 1와 Step 2에서 $N=8$, $p=q=2$ 로 설정하였다. 차영상에 잡음이 약한 경우와 강한 경우 모두 이동 물체가 성공적으로 검출되었다.

마지막으로 '삼성 SPD-1600' 팬틸트 유닛이 내장된 카메라와 800MHz AMD 프로세서가 내장된 표준형 PC에 제안한 세 알고리즘을 모두 구현하였다. 이 시스템에서 320×240 크기의 영상을 대상으로 팬틸트의 회전 속도가 최대 $40^\circ/\text{sec}$ 일 때 ± 20 의 탐색 영역을 전역 탐색하여 전역 움직임을 추정한 결과, 약 50 frames/sec의 속도로 실시간 움직임 검출이 가능하다.

V. 결론

이 논문에서는, 능동 카메라 환경에서 이동 물체의 검출 및 추적을 수행하기 위한 차영상을 이용하는 효과적인 알고리즘을 제안하였다. 먼저 적절한 이전 영상을 선택하는 방법을 제안하여 임의의 속도를 가지는 이동 물체의 검출을 가능하게 하였고, 몇 개의 신뢰성있는 블록들만을 선정하여 전역 움직임 추정에 이용하는 방법을 제안함으로써 고속의 움직임 추정을 가능하게 하였으며, 마지막으로 현재 영상과 차영상의 상관 관계를 이용하여 잡음이 강한 차영상으로부터 움직임 영역을 효과적으로 분리할 수 있는 방법을 제안하여 안정적인 추적이 가능하게 하였다. 기존의 방법과의 비교 실험을 통하여 제안한 알고리즘의 성능이

우수함을 보였으며 실제 능동 카메라 시스템에 적용하여 빠른 물체에 대하여 실시간 동작이 가능함을 확인하였다.

참고 문헌

- [1] K. S. Bhat, M. Saptharishi, and P. K. Khosla, "Motion detection and segmentation using image mosaics," in Proc. IEEE Int. Conf. Multimedia and Expo, Vol. 3, pp. 1577-1580, New York, USA, Jul. 2000.
- [2] D. Comaniciu and V. Ramesh, "Mean shift and optimal prediction for efficient object tracking," in Proc. 2000 IEEE Int. Conf. Image Processing, Vol. III, pp. 70-73, Vancouver, Canada, Sept. 2000.
- [3] L. Wixson, "Detecting salient motion by accumulating directionally-consistent flow," IEEE Trans. Pattern Anal. Machine Intell., Vol. PAMI-22, No. 8, pp. 774-780, Aug. 2000.
- [4] D. Murray and A. Basu, "Motion tracking with an active camera," IEEE Trans. Pattern Anal. Machine Intell., Vol. PAMI-16, No. 5, pp. 449-459, May 1994.
- [5] K. W. Lee, S. W. Ryu, S. J. Lee, and K. T. Park, "Motion based object tracking with mobile camera," Electron. Lett., Vol. 34, No. 3, pp. 256-258, Feb. 1998.
- [6] L. Marcenaro, G. Vernazza, and C. S. Regazzoni, "Image stabilization algorithms for video-surveillance applications," in Proc. 2001 IEEE Int. Conf. Image Processing, pp. 349-352, Thessaloniki, Greece, Sept. 2001.
- [7] B. K. P. Horn and B. G. Schunck, "Determining Optical Flow," Artificial Intelligence, Vol. 17, pp. 185-204, 1981.
- [8] B. Tao and M. T. Orchard, "Gradient-based residual variance modeling and its applications to motion-compensated video coding," IEEE Trans. Image Processing, Vol. CSVT-10, No. 1, pp. 24-35, Jan. 2001.

저 자 소 개



강 동 구

- 1996년 3월~2000년 2월 : 서강대학교 공과대학 전자공학과 (공학학사)
- 2000년 3월~2002년 2월 : 한국과학기술원 전자전산학과 (공학석사)
- 2002년 2월~현재 : 한국과학기술원 전자전산학과 박사과정
- 주관심분야 : 영상처리, 의학 영상 시스템



나 종 범

- 1971년 3월~1975년 2월 : 서울대학교 공과대학 전자공학과 (공학학사)
- 1975년 3월~1977년 2월 : 한국과학기술원 전기 및 전자공학과 (공학석사)
- 1979년 3월~1983년 2월 : 한국과학기술원 전기 및 전자공학과 (공학박사)
- 1977년 3월~1980년 3월 : 광운공과대학 전자공학과 전임강사
- 1981년 1월~1981년 12월 : 미국 뉴욕 콜럼비아 대학교 방문 연구원
- 1983년 7월~1987년 6월 : 미국 뉴욕 콜럼비아 대학교 연구 조교수
- 1993년 2월~1994년 2월 : 미국 뉴욕 콜럼비아 대학교 방문 교수
- 1987년 7월~현재 : 한국과학기술원 전자전산학과 교수
- 1984년 12월~1985년 7월 : Philips Medical Systems 기술 자문
- 1990년 6월~1993년 5월 : 주식회사 문화방송 기술 자문위원
- 주관심분야 : 영상처리, 영상 시스템, Digital TV, 삼차원 영상 가시화