

# 에지 검출을 이용한 움직임 검출기 설계

이승준\*, 권병헌\*\*, 최명렬\*

\*한양대학교 전자전기제어계측공학과

\*\*유한대학 정보통신과

e-mail : [lee30806@asic.hanyang.ac.kr](mailto:lee30806@asic.hanyang.ac.kr)

## Design of a Motion Detector Using Edge Detection

Seung-Joon Lee\*, Byong-Heon Kwon\*\*, Myung-Ryul Choi\*

\*Dept. of EECL, Hanyang University

\*\*Dept. of Information and Telecommunication, Yuhan College

### 요 약

본 논문에서는 조명의 변화 및 피사체의 움직임 특성을 이용한 움직임 검출방식을 제안한다. 피사체의 움직임만을 검출하기 위해 현재 프레임과 이전 프레임 간의 차이 정보 그리고 현재 프레임과 다음 프레임 간의 차이 정보 및 현재 프레임의 피사체의 에지를 검출하고, 검출된 에지와 프레임간의 차이 정보를 이용하여 현재 프레임 내의 피사체의 움직임 정도를 판단한다. 제안된 방식에 의한 움직임 검출은 조명의 변화와 같은 영상 변화를 피사체의 움직임으로 보던 기존 알고리즘보다 더 정확한 움직임 검출을 할 수 있음을 컴퓨터 시뮬레이션을 통해 알 수 있었다.

### 1. 서론

최근 새로운 디스플레이 시스템으로 각광 받고 있는 LCD(Liquid Crystal Display)와 같은 FPD(Flat Panel Display) 시스템에 사용되는 디지털 영상 확대 기법은 입력 영상의 움직임 정도에 관계없이 일률적으로 사용되어 이미지 열화 현상을 가져오기도 한다. 이러한 문제점을 극복하기 위해 많은 움직임 적응형 확대 기법들이 제안되고 있다[1]. 그렇지만 효과적인 움직임 적응형 스케일링을 위해서는 우선 신뢰할만한 움직임 검출이 선행되어야 한다. 일반적으로 실시간 처리를 고려하여 제안된 가장 간단한 움직임 검출법은 입력 영상의 프레임 사이의 픽셀들간의 차이값으로 움직임을 판단하는 것이다[2]. 하지만 이러한 입력 영상의 차이값으로만 움직임을 판단하게 되면 조명의 변화와 같은 영상 변화도 피사체의 움직임으로 판단하게 된다. 따라서 본 논문에서는 이러한 움직임 검출의 오류를 줄이기 위해서 에지 검출을 이용한 움직임 검출 방법을 제안하였다.

본 논문에서는 에지 검출을 이용하기 위해 다음과 같은 가정 하에서 이루어졌다. 그것은 피사체의 움직임은 피사체의 에지의 움직임으로 결정된다는 것이다.

즉, 피사체의 에지의 움직임이 곧 피사체의 움직임이 된다는 것이다. 이러한 가정 하에서 픽셀 값들의 차이 정보와 에지 검출 정보를 모두 고려하면 효과적인 피사체의 움직임을 검출 할 수가 있다.

또한 본 논문에서 제안된 움직임 검출기는 하드웨어 구조가 단순하여 실시간 처리가 가능하다.

### 2. 기존의 움직임 검출 및 판단 방법

기존에 제안했던 움직임 검출 방법은 그림 1에서와 같이 (N-1)번째 프레임과 (N)번째 프레임 사이의 보간 대상 픽셀 주위의 4x5 윈도우 내에 있는 픽셀들의 차에 임계치를 두어 판단한다[3]. 움직임 검출은 식 (1)과 식 (2)에 의해 서로 다른 픽셀들의 개수를 세어 일정 임계치(CV) 이상이면 움직이는 부분으로 판단하고 그 이하면 정지된 부분으로 판단한다. 실험적으로 이 임계치(CV)은 4로 정하였다. 식 (1)에서 PD는 Pixel Difference 이고 PV는 Pixel Value 이다.

$$PD=|(PV(i, j, t-1)-PV(i, j, t)| \quad (1)$$

$$\begin{aligned}
 & \text{if}(PD > CV_1) & (2) \\
 & \quad C_1 = C_1 + 1 \\
 & \text{else} \\
 & \quad C_1 = C_1
 \end{aligned}$$

기존의 움직임 판단 방법은 그림 2와 같이 수행된다. 4x5 윈도우 내의 움직임이 검출된 픽셀의 전체 개수를 세어 그 수가 임계치(CV<sub>2</sub>) 이상이면 동영상으로 그렇지 않으면 정지영상으로 처리하였다.

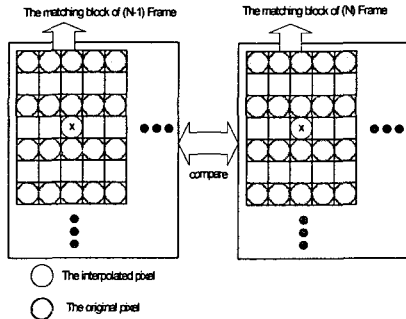


그림 1. 움직임 검출에 사용된 픽셀

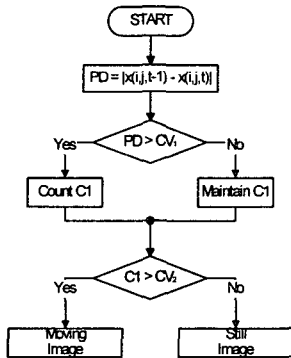


그림 2. 기존 움직임 검출 알고리즘의 흐름도

### 3. 에지 검출 알고리즘

에지 검출 방법은 유사 연산자, 차연산, 유도 연산자, 기울기 연산자, 라플라시안 등이 있다[4,5]. 이 중 본 논문에서는 실시간 에지 검출을 위해 유사 연산자 방식을 사용하였다. 이 방식은 가장 단순하고 빠른 에지 검출 방식으로 식 (3)과 같이 수행된다. 식 (3)에서 EV는 Edge Value를 의미하며 그림 3의 (b)에서 중심 픽셀 A와 주변 픽셀 B에서 I 사이의 각각의 차이값을 구한 뒤 최대값을 선택한다. 여기서 max 함수는 최대값을 구하는 함수이다.

$$EV = \max\{(A-B), (A-C), (A-D), (A-E), (A-F), (A-G), (A-H), (A-I)\} \quad (3)$$

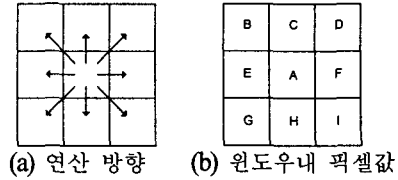


그림 3. 에지 검출 방법

### 4. 제안한 움직임 검출 및 판단 방법

제안된 움직임 검출 방법은 (N-1)번째 프레임과 (N)번째 프레임 사이의 검출 대상 픽셀 및 (N)번째 프레임과 (N+1)번째 프레임 사이의 검출 대상 픽셀 주위의 3x3 윈도우 내에 있는 픽셀들의 차에 의해 움직임을 검출하게 된다. 제안한 움직임 검출 및 판단 방법은 3 단계를 거쳐 움직임을 검출, 판단하게 된다. 먼저 첫번째 단계에서는 (N-1)번째 프레임과 (N)번째 프레임 간의 픽셀 차이값 및 (N)번째 프레임과 (N+1)번째 프레임 간의 픽셀 차이값, 그리고 에지 검출값을 두 번째 단계로 넘겨준다. 두 번째 단계에서는 픽셀 차이값들과 에지 검출값을 서로 AND 연산과 OR 연산을 수행한다. 마지막 단계에서는 두 번째 단계의 출력값이 임계치(CV<sub>1</sub>) 이상이 되는 값들을 세어 그 수가 두 번째 임계치(CV<sub>2</sub>) 이상이 되면 동영상으로 그 이하면 정지화로 판단한다. 실험적으로 CV<sub>1</sub>과 CV<sub>2</sub>는 각각 4와 2로 정하였다.

그림 4의 첫 번째 단계에서는 기존 알고리즘에 의해 움직임을 검출하며 두 번째 단계에서는 에지 검출을 이용해 입력 영상내에 움직임이 검출된 픽셀 중 피사체의 움직임이 아닌 부분에 대해 제거한다. 그리고 마지막 단계에서는 피사체의 움직임으로 검출된 픽셀들에 의해 움직임 정도를 판단한다. 여기서 MV는 Motion Value이다.

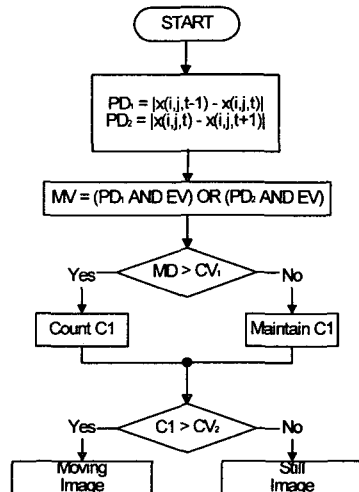


그림 4. 제안한 움직임 검출 알고리즘의 흐름도

### 5. 하드웨어 구성

제안된 움직임 검출 방법에 대한 하드웨어의 기본적인 구성은 그림 5와 같다.

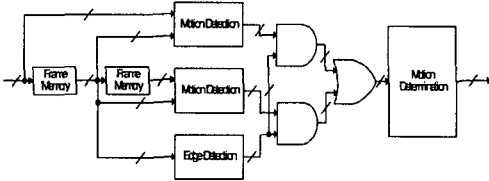


그림 5. 움직임 검출기의 전체 블록 다이어그램

그림 5에서와 같이 하드웨어의 구성은 2개의 프레임 메모리와 에지 검출과 움직임 검출 블록 그리고 검출된 움직임을 판단하는 블록들로 구성되어 있다. 각 블록은 3x3 윈도우를 처리하기 위해 24 비트를 입력으로 한다. 움직임 검출기는 (N-1)번째 프레임의 픽셀과 (N)번째 프레임의 픽셀 그리고 (N+1)번째 프레임의 픽셀들을 입력으로 하며 에지 검출기는 (N)번째 프레임의 첫 번째, 두 번째 그리고 세 번째 라인의 픽셀들을 입력으로 한다. 그리고 움직임 검출기와 에지 검출기의 출력의 동기(Sync)를 맞추기 위해서 움직임 검출기 마지막에 플립플롭(Flip-Flop)을 이용하였다.

그림 6은 에지 검출기의 구조를 나타내고 있으며 그림 7에서는 그림 6의 에지 검출기와 움직임 검출기를 서로 연결해 놓은 구조이다.

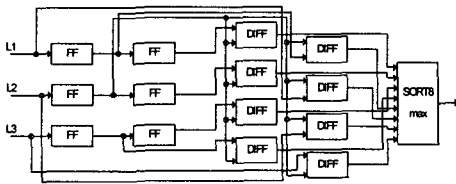


그림 6. 에지 검출기의 구조

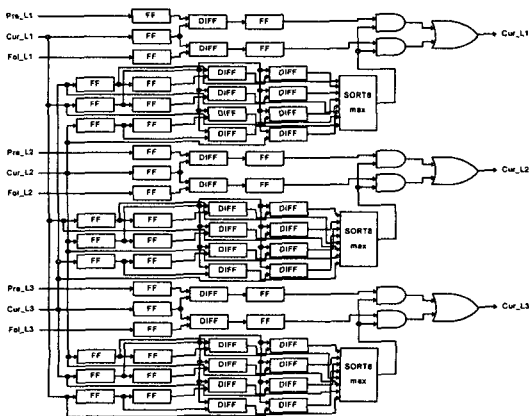


그림 7. 움직임과 에지 검출기의 구조

### 6. 시뮬레이션 결과

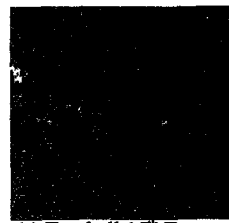
시뮬레이션은 준 동영상인 Salesman 의 30 프레임 (512x480), 동영상인 Football 의 25 프레임(512x480)을 사용하였다. 또한 조명의 변화에 대한 성능 검증을 위해 각각의 이미지의 밝기에 변화를 주어 제안한 알고리즘으로 처리한 후 이미지를 비교하였다. 또한 움직임이 검출된 픽셀의 시각적인 표현을 위해 움직임이 있다고 판단된 픽셀들은 검은색으로 대치되었다.



(a) Salesman 7<sup>th</sup> Frame



(b) Salesman 9<sup>th</sup> Frame



(c) Football 15<sup>th</sup> Frame



(d) Football 20<sup>th</sup> Frame

그림 8. 기존의 알고리즘에 의한 시뮬레이션 결과



(a) Salesman 7<sup>th</sup> Frame



(b) Salesman 9<sup>th</sup> Frame



(c) Football 15<sup>th</sup> Frame



(d) Football 20<sup>th</sup> Frame

그림 9. 제안한 알고리즘에 의한 시뮬레이션 결과

그림 9에서와 같이 제안된 알고리즘에 의해 움직임이 검출된 이미지는 움직임이 있다고 판단된 부분이 피사체의 에지에 집중되어 있음을 알 수 있다.



(a) Salesman 7<sup>th</sup> Frame



(b) Salesman 9<sup>th</sup> Frame



(c) Football 15<sup>th</sup> Frame



(d) Football 20<sup>th</sup> Frame

그림 10. 조명 변화에 의한 시물레이션 결과

그림 10은 샘플 이미지의 밝기를 변화시켜 움직임 검출한 결과이다.

그림 9와 그림 10에서 보듯이 조명의 변화가 있더라도 기존의 방법보다 더 정확하게 피사체의 움직임을 검출하는 것을 확인할 수 있다.

## 7. 결론

한 영상 내에서 움직임이 피사체의 에지에서만 발생된다는 가정이 있는 경우 그 외 다른 영상의 변화를 무시할 수 있다면 움직임 검출기는 훨씬 더 정확하게 피사체의 움직임을 검출할 수 있게 된다. 제안한 알고리즘 방법에서는 기존의 알고리즘에서 나타난 피사체의 움직임 외에 다른 영상의 변화에 대해서도 움직임으로 판단하는 오류를 보상해 주고 있다. 하지만 제안된 알고리즘은 기존 알고리즘에 비해 움직임 검출이 피사체의 에지에만 한정되기 때문에 입력 영상의 움직임 정도가 많이 감소하는 단점을 보여 주고 있다. 그러나 이러한 단점에도 불구하고 실제 검출된 이미지를 보면 기존의 알고리즘에 비해 피사체의 움직임 검출이 향상됨을 확인할 수 있다. 이것은 기존의 방법에 의해 검출된 이미지에서 불필요한 부분을 제거 시킨 결과이다. 결국, 하드웨어적인 측면이나 검출 결과로 볼 때 기존의 제안된 방법에 비해 향상된 움직임 검출기를 구현할 수 있다.

본 논문은 산업자원부와 과학기술부에서 시행한 선도기술 개발 사업의 지원을 받았습니다.

## 참고문헌

- [1] A. Murat Tekalp "Digital Video Processing", Prentice Hall, pp. 314-330, 1995.
- [2] 김봉민 외 "조명과움직임의 변화특성을 이용한 동작검출 기법", 대한전자공학회 하계종합학술대회 논문

집, Vol. 21. No. 1, pp. 940-943, 1998.

[3] 김희철 외 "움직임 적응형 멀티프레임 보간 알고리즘", 대한전자공학회 하계종합학술대회 논문집, Vol. 23, No. 1, pp. 54-57, 2000.

[4] Rafael C. Gonzalez, Richard E. Woods "Digital Image Processing", Edison Wesley, pp. 416-476, 1993.

[5] Randy Crane. "A Simplified Approach to Image Processing : classical and modern techniques in C", Prentice Hall, pp. 79-85, 1997.