

논문 2010-47SP-1-2

# 주변 전경 픽셀 전파 알고리즘 기반 실시간 이동 객체 검출

## ( A Real-time Motion Object Detection based on Neighbor Foreground Pixel Propagation Algorithm )

응웬탄빈\*, 정 선 태\*\*

( Thanh Binh Nguyen and Sun-Tae Chung )

## 요 약

이동 객체 검출은 입력 영상에서 배경과 다른 전경 객체를 찾는 것을 말하는 것으로 지능 영상 감시, HCI, 객체 기반 영상 압축 등의 여러 영상 처리 응용 분야에서 필요한 과정이다. 기존의 이동 객체 검출 알고리즘은 상당한 계산량을 요구하여 다 채널 영상 감시 응용, 또는 임베디드 시스템에서의 단일 채널의 실시간 응용에 사용하는 데 애로가 많다. 보다 정확한 이동 객체 검출을 위하여 필요한 과정인 전경 마스크 정정은 보통 열림, 닫힘 등의 모폴로지 연산을 통해 수행된다. 모폴로지 연산은 계산량이 적지 않고 게다가 프로세싱 방법이 달라 이동 객체 검출의 다음 단계인 연결 요소 레이블링 루틴과 동시에 처리되기 어렵다. 본 논문에서는 먼저 모폴로지 연산과는 달리 연결 요소 레이블링 루틴에서 사용되는 주변 픽셀 점검 과정을 활용한 전경 마스크 정정 알고리즘인 “주변 전경 픽셀 전파”을 고안하고, 이를 활용하여 전경 마스크 정정과 연결 요소 레이블링이 동시에 수행될 수 있는 이동 객체 검출 방법을 제안한다. 실험을 통해, 제안된 이동 객체 검출 방법이 기존의 모폴로지 연산을 사용한 방법 보다 정확하게 이동 객체를 검출하였으며, 대상 실험 영상 프레임 및 비디오에 대해서는 최소 4배 이상 신속하게 처리됨을 확인하였다.

## Abstract

Moving object detection is to detect foreground object different from background scene in a new incoming image frame and is an essential ingredient process in some image processing applications such as intelligent visual surveillance, HCI, object-based video compression and etc. Most of previous object detection algorithms are still computationally heavy so that it is difficult to develop real-time multi-channel moving object detection in a workstation or even one-channel real-time moving object detection in an embedded system using them. Foreground mask correction necessary for a more precise object detection is usually accomplished using morphological operations like opening and closing. Morphological operations are not computationally cheap and moreover, they are difficult to be rendered to run simultaneously with the subsequent connected component labeling routine since they need quite different type of processing from what the connected component labeling does. In this paper, we first devise a fast and precise foreground mask correction algorithm, “Neighbor Foreground Pixel Propagation (NFPP)” which utilizes neighbor pixel checking employed in the connected component labeling. Next, we propose a novel moving object detection method based on the devised foreground mask correction algorithm, NFPP where the connected component labeling routine can be executed simultaneously with the foreground mask correction. Through experiments, it is verified that the proposed moving object detection method shows more precise object detection and more than 4 times faster processing speed for a image frame and videos in the given the experiments than the previous moving object detection method using morphological operations.

**Keywords :** 이동 객체 검출, 전경 마스크 정정, 블롭 분할, 연결 요소 레이블링

\* 학생회원, \*\* 정회원, 숭실대학교 정보통신전자공학부

(School of Electronic Engineering, Soongsil University)

※ 본 연구는 숭실대학교 교내연구비 지원으로 수행되었습니다.

접수일자: 2009년7월28일, 수정완료일: 2009년11월4일

### I. 서론

이동 객체 검출은 입력 영상에서 배경과 다른 전경 블록을 찾는 것을 말하는 것으로 지능 영상 감시, HCI, 객체 기반 영상 압축 등의 여러 영상 처리 응용 분야에서 필요한 과정이다<sup>[1~3]</sup>. 예를 들어 지능 영상 감시에서는 관심 객체의 행위 분석이 요구된다. 관심 객체는 이동 블록 검출로부터 도출되며 지능 영상 감시의 성능은 얼마나 빠르고 정확하게 이동 블록을 검출할 수 있는지에 달려 있다는 것이 잘 알려져 있다<sup>[1]</sup>. 그런데 기존의 블록 검출 알고리즘은 상당한 계산량을 요구하여 다채널 영상 감시 응용, 또는 임베디드 시스템에서의 단일 채널의 실시간 응용에 사용하는데 애로가 많다.

이동 객체 검출에서 필요한 단계인 전경 마스크 정정은 보통 열림/닫힘 등의 모폴로지 연산이 전처리 단계로 처리되어 수행된다<sup>[4~6]</sup>. 한편, 전경 마스크는 다수의 블록을 포함할 수 있으므로 이들 블록을 구분하여 분할을 해낼 필요가 있으며 대부분 연결 요소 레이블링 알고리즘을 이용하여 처리된다. 그런데, 모폴로지 연산은 계산량이 적지 않고 게다가 프로세싱 방법이 연결 요소 레이블링 루틴에서 사용되는 연산들과는 달라 다음 단계인 연결 요소 레이블링 연산들과 동시에 처리되기 어렵다.

본 논문에서는 새로운 전경 마스크 정정 알고리즘을 고안하고 이를 활용한 실시간 이동 객체 검출 방법을 제안한다. 본 논문에서 고안된 새로운 전경 마스크 정정 알고리즘인 “주변 전경 픽셀 전파”는 8개 주변 픽셀 점검을 이용하는 데, 8개 주변 픽셀 점검은 연결 요소 레이블링 절차에서도 활용된다. 따라서 “주변 전경 픽셀 전파” 알고리즘이 연결 요소 레이블링 처리 루틴 과정에 포함되어 동시에 처리될 수 있게 된다. 따라서 전경 마스크 정정은 연결 요소 레이블링 처리 과정에서 함께 처리되게 되므로, 모폴로지 연산을 이용한 전경 마스크 정정 과정을 먼저 처리하고 이 후 연결 요소 레이블링 과정을 순차적으로 수행하는 기존 이동 객체 검출 방법들에 비해 처리 시간의 개선을 가져온다.

실험을 통해 제안된 이동 객체 검출 방법이 기존 방법보다 정확하고 신속하게 처리됨을 확인하였다.

본 논문의 구성은 다음과 같다. II장에서는 블록 검출, 모폴로지, 연결 요소 레이블링 등 본 논문의 이해에 필요한 기술적 배경을 설명한다. III장에서는 제안된 이동 객체 검출 방법이 기술된다. 먼저, 전경 마스크 정정

처리 알고리즘인 “주변 전경 픽셀 전파”에 대해 설명하고, 이를 활용한 제안된 이동 객체 검출 방법에 대해 기술한다. IV장에서는 실험 결과가 마지막으로 V장에서는 결론이 맺어진다.

### II. 기술적 배경

#### 2.1 움직임 분석에서의 이동 객체 검출

지능 영상 감시, HCI, 객체 기반 동영상 압축 등의 움직임 분석 응용에서 장면의 배경과 다른 이동 전경 객체를 검출할 필요가 있다.

이동 전경 객체 검출은 보통 그림 1의 여러 단계의 처리 과정을 거친다.

전경 마스크는 전경 픽셀들은 ‘1’로 배경 픽셀은 ‘0’의 값을 가지도록 처리된 이진 영상을 의미한다. 입력 영상 프레임의 각 픽셀이 전경 픽셀인지 배경 픽셀인지의 판별은 다양한 방법으로 이루어진다<sup>[1~3]</sup>.

추출된 전경 마스크에는 원래 전경 픽셀인데, 배경 픽셀로 판별되어 추출되지 못했거나, 배경 픽셀인 데 전경 픽셀로 잘못 추출될 수 있다. 따라서 전경 마스크에 구멍이나 돌출 부분이 고립된 픽셀부분들이 생성될 수 있기 때문에 구멍은 메우고 돌출 부분이나 고립된 부분들은 제거하는 전경 마스크 정정 과정이 필요하게 된다. 전경 마스크 정정은 보통 열림, 닫힘 등의 모폴로지 연산이 전처리 단계로 처리되어 수행된다<sup>[4~6]</sup>.

블록은 모든 연결된 전경 픽셀들의 집합으로 정의된다. 정정된 전경 마스크엔 다수의 블록들이 존재할 수

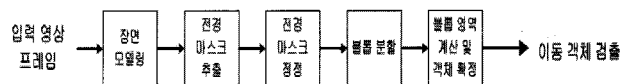


그림 1. 이동 객체 검출 과정  
Fig. 1. Processing steps of moving object detection.

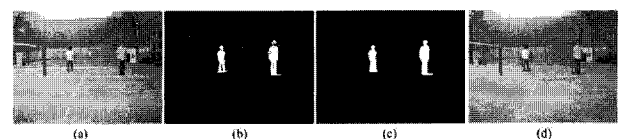


그림 2. (a) 원 영상 프레임, (b) 전경 마스크 이미지, (c) 정정된 전경 마스크 이미지, (d) 이동 객체 영역이 계산된 원 영상 이미지

Fig. 2. (a) Original frame image, (b) Foreground mask image, (c) Foreground corrected mask image, and (d) Original frame image with calculated object regions.

있다. 따라서 이들 블록을 구별하여 검출해내어야 하며, 이 과정이 블록 분할이며, 대부분 연결 요소 레이블링 알고리즘을 이용한다. 블록이 분할된 후에, 각 블록을 포함하는 최소 사각형 영역이 계산되며, 이 사각 영역이 이동 객체 영역으로 확정된다.

그림 2는 이동 객체 검출 단계와 연관된 이미지들을 보여준다.

### 2.2 모폴로지

이진 모폴로지의 기본 아이디어는 간단한 미리 정의된 모양을 갖는 조각 이미지를 이용하여 원래 이미지를 탐색하여 이미지 형태를 변형하여 원하는 영상 처리 작업을 하는 것이다<sup>[4~6]</sup>. 이 조각 이미지를 구조 요소(structuring element)라 하며, 그 자체가 이진 이미지이다. 보통 많이 사용되는 구조 요소에는 디스크, 정사각형, 십자형 등의 있으며 크기는 보통 3x3을 취한다.

이진 모폴로지의 기본 연산에는 팽창(dilation), 침식(erosion), 닫기(closing), 열기(opening) 등이 있다. 팽창 연산은 이미지 영역을 확대하며, 침식은 축소한다. 닫기 연산은 팽창 연산 후 침식 연산을 수행하는 일련의 연산으로 정의되며 이미지 영역 내의 내부 구멍을 채우는 작업을 수행한다. 열기 연산은 침식 연산 후 팽창 연산을 수행하는 일련의 연산으로 정의되며 이미지 영역 경계에서 빠져나온 부분이나 고립된 조그만 픽셀들을 제거한다. 따라서 모폴로지 연산은 전경 마스크 정정 과정에서 많이 이용된다.

### 2.3 유니온-파인드 구조와 연결 요소 레이블링

연결 요소 레이블링(Connected Component Labeling; CCL)은 이미지를 스캔하여 이미지 픽셀들을 픽셀들의 연결에 기반하여 연결된 픽셀들의 모임들로 나누는 작업을 수행한다<sup>[7]</sup>. 전경 마스크 이미지에서 연결 요소는 전경 픽셀을 연결된 주변으로 갖는 전경 픽셀들의 집합이다. 본 논문에서 픽셀의 주변이란 그림 3에서 보이는 바와 같이 해당 픽셀의 8개 주변을 의미한다.

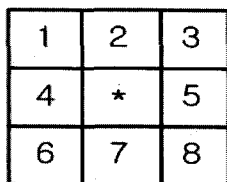


그림 3. 8 주변  
Fig. 3. 8 neighbors.

일단 모든 연결 요소가 결정되면, 각 픽셀은 소속되는 연결 요소의 레이블이 할당된다. 이미지에서 분리된 연결 요소들을 추출하고 이들에게 레이블을 할당하는 작업은 지능형 영상 감시를 포함한 많은 컴퓨터 비전 응용에서 요구되는 중요한 작업의 하나이다<sup>[7]</sup>.

연결 요소 레이블링 문제를 처리하기 위해 많은 알고리즘이 제안되었다<sup>[8~13]</sup>. 이 가운데 가장 유망하고 효율적인 연결 요소 레이블링 알고리즘들은 보통 유니온-파인드 구조(union-find structure)를 이용한 2단계 절차로 이루어진다<sup>[7, 11~13]</sup>.

유니온-파인드 구조는 등가 클래스(equivalence class)를 트리 구조로 효과적으로 구축하고 조작하기 위한 것으로, 서로소 집합(disjoint set)의 모임들을 저장하고 유니온(2개 집합을 하나로 합침)과 파인드(특정 원소가 어느 집합에 속하는 지를 결정)의 연산들을 지원한다. 각 집합은 트리구조로 저장된다. 트리의 노드는 각 집합의 레이블을 나타내고 노드의 값은 부모 노드를 가리킨다. 유니온-파인드의 트리 구조는 벡터 어레이 PARENT를 통해 표현된다. PARENT의 색인(subscript)은 가능한 레이블(노드)을 나타내고 그 값은 부모 노드의 레이블을 타낸다. '0' 값을 갖는 노드는 최상위 노드(root node)를 의미한다.

“유니온-파인드 구조를 이용한 2 단계 절차 연결 요소 레이블링 알고리즘”(CCLUF)의 동작은 간단히 설명하면 다음과 같다<sup>[7]</sup>.

첫 번째 단계에서, 알고리즘(CCLUF)은 이진 이미지를 행별로 왼쪽에서 오른쪽으로 차례로 스캔한다. 스캔 중에 전경 픽셀을 만나면 이미 스캔된 픽셀들 중에 주변 전경 픽셀이 있는 지를 점검하고(주변 전경 픽셀 점검)만일 없으면 해당 전경 픽셀에 이제까지 할당되었던 레이블중 가장 큰 레이블 다음 레이블을 할당하고, 있으면 스캔된 주변 전경 픽셀들의 레이블중 가장 낮은 레이블을 할당한다. 이때, 주변 전경 픽셀들의 레이블들은 모두 같은 등가 클래스에 속하게 된다. 발견된 등가 클래스는 유니온-파인드 구조에 기록된다. 첫 번째 단계의 절차의 마지막에 각 등가 클래스는 완전히 결정되고 유니온-파인드 구조의 트리의 루트로 결정되는 유일한 레이블을 갖게 된다.

두 번째 단계 절차는 이미지를 스캔하여 임시 레이블들을 최종 레이블로 대치된다. 최종 레이블은 각 레이블의 루트 노드의 레이블로 된다.

그림 4는 예제 입력 이진 이미지를 보여준다. 여기서

f			f			f			f
f			f			f			f
f			f			f			f
	f	f					f	f	

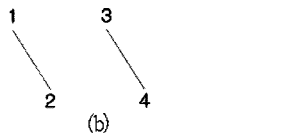
그림 4. 전경 마스크 (f 는 전경 픽셀을 의미)  
Fig. 4. Foreground mask where f means foreground pixel.

1			2			3			4
1			2			3			4
1			2			3			4
	1	1					3	3	

(a)

PARENT

1	2	3	4
0	1	0	3



(b)

그림 5. 1단계 처리 결과  
Fig. 5. The result of 1st pass.

1			1			3			3
1			1			3			3
1			1			3			3
	1	1					3	3	

그림 6. 마지막 2단계 처리 후 결과  
Fig. 6. The final result.

f 는 전경 픽셀을 의미하며, 그 밖의 픽셀은 배경 픽셀이다.

그림 5는 앞에서 기술한 유니온-파인드 구조를 이용한 연결 요소 레이블링 알고리즘의 첫 단계 절차 수행 결과를 보여준다.

그림 6은 2단계 절차후의 마지막 레이블링 결과를 보여준다.

### III. 제안된 이동 객체 검출 방법 : CLNF

닫기/열기 등의 모폴로지 연산과 연결 요소 레이블링 알고리즘의 연산들이 처리 방법이 달라 동시에 처리하기 힘들다는 것을 어렵지 않게 이해할 수 있다. 또한, 모폴로지 연산은 일반적으로 이미지의 모든 픽셀에 대

해 수행하므로 계산량이 적지 않다.

그래서 본 논문에서는 모폴로지 연산과는 다른 연결 요소 레이블링 작업 루틴에 통합되어 연결 요소 레이블링 작업 수행 시에 동시에 전경 마스크 정정이 가능한 새로운 전경 마스크 정정 방법을 제안한다.

#### 3.1 주변 전경 픽셀 전파

자연 객체(사람, 차량, 나뭇가지 등)들은 매우 작은 구멍이나 빠져나온 부분이 있기 힘들다. 또한 고립된 소수 픽셀 집합은 관심 있는 객체일 가능성이 적다. 전경 마스크 이미지의 블롭들은 자연 객체의 블롭들이다. 따라서 전경 마스크 이미지에 보이는 전경 영역에서의 구멍은 추출되지 못한 전경 픽셀일 가능성이 매우 높다. 또한 고립된 전경 픽셀이나 빠져나온 전경 픽셀들은 전경 픽셀로 잘못 추출되었을 가능성이 높다. 그러므로 전경 마스크 이미지의 구멍은 주변에 많은 전경 픽셀을 가질 확률이 높고 빠져나온 전경 픽셀이나 고립된 전경 픽셀은 주변에 배경 픽셀을 가질 가능성이 높다.

여기서, 픽셀 X의 전경 주변 확률 P(X)를 해당 픽셀 X의 8 주변 (그림 3 참조) 중 전경 픽셀일 비율로 정의한다.

이러한 관찰로부터 본 논문에서는 다음과 같이 전경 마스크 정정 방법인 “주변 전경 픽셀 전파” 알고리즘을 제안한다.

#### 주변 전경 픽셀 전파(Neighbor Foreground Pixel Propagation : NFPP)

픽셀의 값이 2가지인 전경 마스크 이미지( 값 ‘1’은 전경 픽셀을, 값 ‘0’은 배경 픽셀을 의미 )에서, 전경 마스크 정정은 다음과 같이 이루어진다.

- 전경 마스크 이미지를 행별로 왼쪽에서 오른쪽으로 스캐닝할 때, 해당 픽셀의 정정은 다음과 같이 전경 주변 확률이 0.5 이상이면, 전경 픽셀(값 1)로, 확률이 0.5 미만이면 배경 픽셀(값 0)로 한다. 즉,

$$\text{값}(X) = \begin{cases} 1; & P(X) \geq \frac{1}{2} \\ 0; & \text{그렇지 않으면} \end{cases}$$

“주변 전경 픽셀 전파” 알고리즘 수행의 예가 다음 그림 7에 나타나 있다.

그림 8은 적응 혼합 가우시안 모델링<sup>[14-15]</sup>을 이용하여 얻어진 전경 마스크에 NFPP 방법을 적용한 결과를 보여 준다. NFPP 적용 후에 전경 마스크의 블롭들이

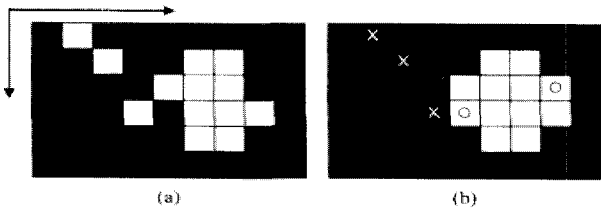


그림 7. (a) 전경 마스크, (b) NFPP 적용후 정정된 전경 마스크 (X 위치는 배경으로 변환된 픽셀들을 나타내며, O 는 전경 픽셀로 변환된 픽셀들을 나타낸다.)

Fig. 7. (a) Foreground mask, (b) Final result after NFPP where X spots are elements that were changed to background (Black), O spots are elements that were turned into foreground (White).



그림 8. (a) 전경 마스크 (b) NFPP 적용후  
Fig. 8. (a) Foreground mask. (b) After NFPP.

더 분명해지고 채워졌음을 볼 수 있다.

NFPP는 간단하면서도 좋은 블롭 정정 성능을 보여 준다. 이러한 좋은 블롭 정정 성질에 더하여 보다 중요한 특성은 연결 요소 레이블링(CCL) 알고리즘 처리 중에 수행될 수 있다는 점이다. 이는 NFPP와 CCL이 모두 주변 전경 픽셀 점검을 활용하기 때문이다.

이러한 특성은 이동 객체 검출에 계산량을 많이 줄일 수 있다. 이러한 특성에 기반하여 본 논문에서는 새로운 이동 객체 검출 알고리즘 CLNF(CCLUF with NFPP)을 제안한다. CLNF는 기존 블롭 검출 알고리즘 보다 더 빠르고 정확하다.

### 3.2 CLNF (CCLUF with NFPP)

#### 가. 개요

본 논문에서 새로운 이동 객체 검출 알고리즘, CLNF을 제안한다. CLNF는 CCLUF와 NFPP를 결합한 것이다. CLNF의 동작 구조는 그림 9와 같다.

CLNF는 CCLUF와 마찬가지로 2단계 절차로 수행된다. 다만, 모폴로지 연산 전처리후 CCLUF이 수행되던 기존 방법과 달리 CCLUF 수행 중에 전경 마스크 정정 방법인 NFPP가 1단계에서 동시에 수행되도록 되어 계

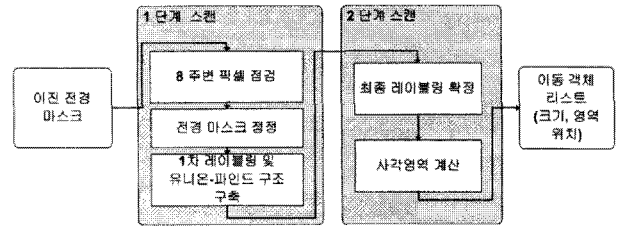


그림 9. CLNF의 동작 구조  
Fig. 9. Work flow of CLNF.

산량이 줄어든다. 1단계에서는 연결 요소 레이블링과 NFPP를 이용한 전경 픽셀 정정, 그리고 유니온-파인드 구조가 구축되며 2단계에서 구축된 유니온-파인드 구조를 이용하여 임시로 할당된 픽셀들의 레이블이 정정된다. 이후, 각 레이블의 블롭을 포함하는 최소 사각 영역이 계산된다.

#### 나. 의사 코드(Pseudo Code)

제안된 블롭 검출 알고리즘 CLNF의 더 명확한 기술을 위하여 CLNF의 의사 코드를 아래에 스케치한다.

##### (1) 1단계 스캔에서 의사 코드

```

-ParentLabel: the label value should be assigned to
current pixel
-ChildLabel: the label which will be replaced by
ParentLabel
-Image[]: binary image
-Labels[]: label map of Image[]
-LabelTree[]: PARENT array in Union-Find Structure
explained in 2.3
-NewLabel:=0;           // the new value of label

With each Image[X]
Start
  With each of 8 neighbors of a pixel Image[X]
  Start
  - Count Foreground Pixel
  - Assign ParentLabel as the smallest label among
  Image[X]'s left or above foreground pixel
  neighbors' labels
  - In case Image[X] has 2 different neighbor labels,
  assign ChildLabel as the bigger label
  End
  If foreground neighborhood probability >=1/2
  Then
  -Image[X] = Foreground;           // Foreground = 1;
  Else
  -Image[X] = Background;           // Background = 0;
  End
  If Image[X] is foreground
    
```

```

Then
  If ParentLabel and ChildLabel are not assigned
  Then
    - Increase NewLabel by 1;
    - Labels[X] = NewLabel;
  Else
    - Recalculate new ParentLabel (parent of
      ChildLabel) rely on LabelTree[];
    - LabelTree[ChildLabel] = ParentLabel;
    - Labels[X] = ParentLabel;
  End
End
End
End
    
```

(2) 2단계 스캔에서의 의사 코드

```

-SizeBlobs[]: list of blob size
-BlobRectangles[]: list of blob rectangle regions

With each Labels[X]
Start
  - Relabeling Labels[X] by its parent note in tree
    LabelTree[];
  - Increase SizeBlobs[Labels[X]] by 1;
  - Calculate BlobRectangles[Labels[X]];
End
    
```

IV. 실험 결과

4.1 NFPP와 모폴로지 연산의 전경 마스크 정정 정확성 비교

먼저, 전경 마스크 정정 능력에 대해 기존 모폴로지 연산과 본 논문에서 제안한 NFPP와 비교하였다. 그림 10은 이 비교 결과를 보여 준다.

그림 10은 NFPP가 모폴로지보다 방법이 간단하면

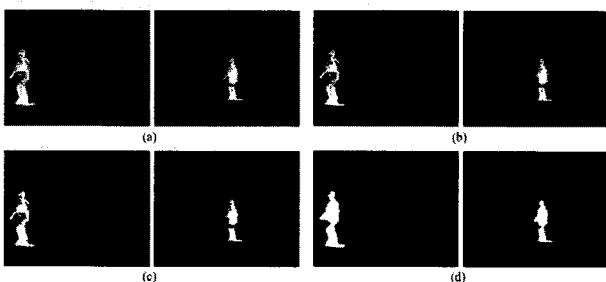


그림 10. (a) 전경 마스크, (b) 열기 모폴로지 연산 적용 후, (c) 열기 및 닫기 모폴로지 연산 적용 후, (d) NFPP 적용 후  
 Fig. 10. (a) Foreground mask, (b) Result after morphology (Open), (c) after morphology (Open + Close), (d) after NFPP.

서도 정정 능력이 더 좋음을 보여 준다.

그림 10(a)에서의 전경 마스크는 적응 혼합 가우시안 모델링<sup>[14-15]</sup>을 이용하여 구하였으며, 그림 11(b)와 (c)의 열기 및 닫기 모폴로지 연산은 3x3 사이즈의 정사각형 구조 요소를 사용하였다. 만일 더 적절한 구조 요소를 사용하면, 모폴로지 연산 이용의 경우도 NFPP 사용 경우와 같이 좀 더 분명한 전경 마스크 정정을 얻을 수 있다. 그러나 이 경우 더 많은 연산 계산량이 필요하다.

4.2 수행시간 시간 비교

기존 모폴로지 이용 이동 객체 검출 알고리즘(이하 기존 방법, CONV)과 제안 이동 객체 검출 알고리즘(제안 방법, CLNF)을 이용한 각각의 경우의 수행시간을 비교하기 위해 2가지 실험을 시행하였다.

첫 번째 실험에서는 그림 10(a)의 전경 마스크 한 프레임의 원래 영상 프레임에 대해 이동 객체 검출 시간을 계산하였고, 2번째 실험에서는 영상 비디오에 대해 연속적인 이동 객체 검출 수행 시간을 계산하여 비교하였다.

2가지 실험의 환경은 3GHz 펜티엄 4 코어 및 3GB DDR2 메모리를 갖는 윈도우즈 PC이다. 또한, 각 영상 이미지 프레임의 해상도는 320x240이며, 영상 비디오 1, 2, 3과 4의 크기는 각각 4.67MB(221 프레임)과 245MB(1117 프레임), 7.67MB(2687 프레임), 47.90MB(4912 프레임)이다.

표 1. 그림 11(a)의 원래 영상 한 프레임에 대한 이동 객체 검출 수행 시간 비교

Table 1. Comparison of the processing time between the conventional moving object detection algorithm (CONV) and the proposed one (CLNF) for original frame of Fig. 11(a).

	수행 시간 (sec)
기존 방법 (CONV)	0.01225
제안 방법 (CLNF)	0.002526

표 2. 동영상에서의 이동 객체 검출 수행 시간 비교  
 Table 2. Comparison of the processing time between the conventional blob detection algorithm(CONV) and the proposed one(CLNF) for two movies.

	비디오1	비디오2	비디오3	비디오4
기존 방법 (CONV)	2.8315	12.9551	29.6298	53.6821
제안 방법 (CLNF)	0.5195	2.2734	6.0848	11.4395

표 1과 2는 각각 첫 번째와 두 번째 실험 결과를 보여준다.

표 1과 2의 결과는 제안된 이동 객체 검출 알고리즘 CLNF가 기존 모폴로지 사용 이동 객체 검출 방법 보다 더 빠르게 처리한다는 것을 보여준다.

## V. 결 론

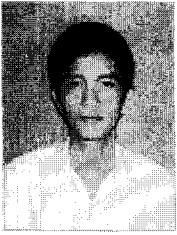
본 논문에서는 기존의 모폴로지 연산을 이용한 이동 객체 검출 방법 보다 정확성 및 수행 속도에서 성능이 개선된 새로운 실시간 이동 객체 검출 방법을 제안하였다. 모폴로지 전처리 과정을 통하여 전경 마스크 정정을 수행하였던 기존 이동 객체 검출 알고리즘과 달리, 제안된 개선된 이동 객체 검출 방법은 전경 마스크 정정이 연결 요소 레이블링 알고리즘 수행 과정 중에 처리할 수 있는 간단하면서도 효율적인 전경 마스크 정정 알고리즘인 “주변 전경 픽셀 전파(NFPP)” 알고리즘을 제안하고 이를 이용함으로써 수행 시간 개선을 이루었다.

실험을 통하여 본 논문에서 제안한 이동 객체 검출 알고리즘의 수행 시간 및 정확성 개선 효과를 확인하였다.

## 참 고 문 헌

- [1] T. Chen, H. Haussecker, A. Bovyryn, R. Belenov, K. Rodyushkin, A. Kuranov, V. Eruhimov, “Computer Vision Workload Analysis: Case Study of Video Surveillance Systems,” *Intel Technology Journal*, May 2005.
- [2] Z. Zivkovic, “Improved adaptive Gaussian mixture model for background subtraction,” *ICPR* pp. 28-31, 2004.
- [3] C. Stauffer C, W.E.L. Grimson, “Adaptive background mixture models for real-time tracking,” *IEEE CVPR*, pp. 244 - 252, 1999.
- [4] P. Soille, “Morphological Image Analysis: Principles and Applications,” *Springer-Verlag Telos*, Berlin, 2003.
- [5] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, “Digital Image Processing using MATLAB,” *PrenticeHall, UpperSaddleRiver*, NewJersey, 2003.
- [6] E. R. Dougherty and R. A. Lotufo, “Hands-on Morphological Image Processing,” *SPIE Press*, Bellingham, 2003.
- [7] L. Shapiro and G. Stockman, “Computer Vision,” *Prentice Hall*, Upper Saddle River, New Jersey, 2001.
- [8] A. Rosenfeld, J.L. Pfaltz, “Sequential Operations in Digital Picture Processing,” *JACM*, pp. 417-494, 1966.
- [9] M. Onoe, “Real Time Parallel Computing Image Analysis,” *Plenum Press*, New York, 1981.
- [10] R. Lumia, L. Shapiro, and O. Zniga, “A New Connected Components Algorithm for Virtual Memory Computers,” *Computer Vision and Image Processing*, pp.287-300, 1983.
- [11] C. Fiorio and J. Gustedt, “Two linear time Union-Find strategies for image processing,” *Theoretical Computer Science, Elsevier Science Publishers Ltd*, pp. 165-181, 1996.
- [12] R. E. Tarjan, “Efficiency of a Good But Not Linear Set Union Algorithm,” *Journal of the ACM*, pp. 215-225, 1975.
- [13] M. B. Dillencourt, H. Samet, and M. Tamminen, “A General Approach to Connected-Component Labeling for Arbitrary Image Representations,” *JACM, ACM*, pp. 253-280, 1992.
- [14] C. Stauffer C, W.E.L. Grimson, “Adaptive background mixture models for real-time tracking,” *IEEE CVPR*, pp. 244 - 252, 1999.
- [15] P. KadewTraKuPong and R. Bowden, “An improved adaptive background mixture model for real-timetracking with shadow detection,” in *Proc. 2nd European Workshp on Advanced Video-Based Surveillance Systems*, 2001.

저 자 소 개



응웬탄빈(학생회원)  
 2005년 10월 University of  
 natural science VNU-HCM,  
 Information Technology  
 Department 학사.  
 2008년 9월~현재 송실대학교  
 정보통신공학 석사  
 재학 중.

<주관심분야 : 영상감시, 소프트웨어 공학, 정보  
 공학, 임베디드 멀티미디어>



정선태(정회원)-교신저자  
 1983년 2월 서울대학교  
 전자공학과 학사.  
 1990년 12월 미국 미시간대학교  
 (앤아버) 전자 및 컴퓨터  
 박사.  
 1991년 3월~현재 송실대학교  
 정보통신전자공학부 교수.

<주관심분야 : 생체인식, 지능형 영상감시, 임베  
 디드 멀티미디어>