

논문 2007-44SP-5-8

# 능동윤곽모델과 움직임 추정을 결합한 실시간 객체 추적 기술

## ( Combined Active Contour Model and Motion Estimation for Real-Time Object Tracking )

김 대 희\*, 이 동 은\*\*, 백 준 기\*

( Daehee Kim, Dongeun Lee, and Joonki Paik )

### 요 약

본 논문에서는 능동윤곽모델에 기반을 둔 스네이크 알고리즘을 움직임 추정과 결합하여 안정적인 객체 추적 기술을 제안하였다. 초기 영상에서 목표 객체의 초기 윤곽을 지정한 후 스네이크 알고리즘을 사용하여 객체의 경계 영역을 찾아내고, 동시에 움직임 추정 기술을 사용하여 객체의 이동 방향과 거리를 예측하여 초기값을 갱신한다. 연속되는 다음 영상에서는 스네이크 알고리즘을 같은 방법을 사용하여 객체 영역을 추정한다. 스네이크 알고리즘은 배경과 객체를 구분하는 역할을 수행하고, 움직임 추정 알고리즘은 객체의 이동 방향과 변위를 찾아낸다. 제안된 기술은 기존의 형태모델에 기반을 둔 추적 기술에 비해 상당히 계산량이 줄기 때문에 실시간 객체 추적이 가능하며 복잡한 배경에서도 추적의 정확도를 유지하는 장점이 있다.

### Abstract

In this paper we proposed a combined active contour model and motion estimation-based object tracking technique. After assigning the initial contour, we find the object's boundary and update the initial contour by using object's motion information. In the following frames, similar snake algorithm is repeated to make continuously estimated object's region. The snake algorithm plays a role in separating the object from background, while motion estimation provides object's moving direction and displacement. The proposed algorithm provides equivalently stable, robust, tracking performance with significantly reduced amount of computation, compared with the existing shape model-based algorithms.

**Keywords** : snake algorithm, motion estimation, object tracking

### I. 서 론

영상 합성, 모션캡처, 보안 감시 시스템, human computer interaction(HCI) 등 많은 분야에서 비디오에서 객체를 인식하고 추적하는 기술들이 널리 응용되고 있다. 추적 시스템은 군사적인 목적으로 처음 개발 되었으나 현재에 와서는 은행, 공항, 대형 빌딩 등과 같은

공공장소에서 보안을 위한 기능으로 많이 사용이 되고 있으며, 로봇 시각을 위한 객체의 움직임 추정 및 인식을 위한 연구가 활발하게 수행되고 있다.

기존의 모델 기반 컴퓨터 시각 기술은 주로 정형(rigid) 객체를 대상으로 하여 그 성능을 인정받았으나<sup>[1]</sup>, 움직이는 사람과 같이 추적하는 물체의 형태가 변하는 경우에는 비정형(non-rigid) 모델을 이용하는 기술이 필요하다<sup>[2~4]</sup>. 비정형 모델 혹은 변형가능(deformable) 모델을 이용한 추출 및 추적 시스템은 객체의 변형에 강건하고 가림현상(occlusion) 등을 극복할 수 있는 기능이 요구된다.

비정형 객체를 위한 강건한 추적 기술의 대표적인 것으로 능동형태모델(active shape model; ASM)이 있다. 그러나 ASM은 초기 훈련집합의 기준점에 따라 principal component analysis(PCA) 결과값이 달라지며, 이는 최종적으로 부정합된 객체 형태를 초래한다. 또한

\* 정희원, \*\* 학생회원, 중앙대학교 첨단영상대학원 영상공학과

(Dept. of Image Engineering, Graduate School of Advanced Imaging Science, Multimedia, and Film, Chung-Ang University)

※ 이 연구는 과학기술부 국가지정연구실 사업과 서울시 산학연협력사업과 정보통신부 및 정보통신 연구진흥원의 대학 IT연구센터(홈네트워킹연구센터) 육성·지원 사업으로 수행되었음

접수일자: 2007년5월11일, 수정완료일: 2007년8월17일

많은 계산량으로 실시간 추적에 어려움을 갖는 단점을 수반한다.

다른 방법으로서 영상분할에 많이 사용되는 능동윤곽모델(active contour model; ACM)은 객체의 윤곽 에너지 정보를 바탕으로 객체의 형태를 찾는 방법이다. 그러나 객체 내부만이 아닌 주변에 있는 배경 에너지의 영향을 함께 받기 때문에 적절한 전처리 과정이 없으면 배경 윤곽 에너지를 최소화하고 잡음을 제거 할 수 없는 특성이 있다.

본 논문에서는 ASM 혹은 스네이크 알고리즘을 사용하여 임의의 객체에 대한 형태를 수동으로 지정한 후 보다 정확한 객체의 형태를 자동으로 추출한다. 그리고 움직임 추정 기법을 결합하여 비정형 객체를 추적하는 기술을 제안한다.

본 논문의 구성은 다음과 같다. II장에서는 기존 스네이크 알고리즘 기술과 문제점을 제시하고, III장에서는 영상 전처리 과정의 필요성과 방법, 그리고 움직임 추정 기법을 적용한 스네이크 알고리즘에 대해 제시한다. IV장에서는 실험 결과를 정리하고, 마지막으로 V장에서 결론을 제시한다.

## II. 기존 스네이크 알고리즘 기술 및 문제점 제시

영역 분할기법 중 하나인 윤곽선 추출 기법은 특정한 물체의 형태를 추출하기도 하고, 그 면적이나 주위의 길이를 측정하며, 두 화상간의 차이를 구할 수도 있다. 영상에서 윤곽선은 물체가 가지는 특징으로서 배경과 이루는 경계로 표현되기도 하고, 물체 자체를 식별하는 특징 값으로 사용된다. 본 논문에서는 추적 대상인 객체의 최외곽 영역은 그 특징상 폐곡선을 이루고 있으며 또한 이 폐곡선이 동적 특징을 포함하고 있기 때문에 변형모델인 능동윤곽선 모델을 적용하여 윤곽선 검출의 정확성을 향상할 수 있다.

### 1. 기존 스네이크 알고리즘

입력영상으로부터 적당한 기준점을 객체의 윤곽선에 할당한다. 이 작업은 주로 수동으로 이루어진다. 바람직한 기준점은 한 영상에서 기준점으로 선택된 특징점이 다음 영상에서도 특징점으로 연결되어 있어야 한다. 2차원 영상에서 우리는  $n$ 개의 기준점을  $2n$ 차원 벡터를 사용해서 다음과 같이 표현할 수 있다.

$$x = [x_1, \dots, x_n, y_1, \dots, y_n]^T, \quad (1)$$

본 논문에서 적용될 실험에 사용하는 기준점의 수는  $n=32$ 로 했다. 기준점을 자동적으로 결정하는 방법은 Tian등<sup>[7]</sup>에 의해 논의되었다. 기준점의 역할은 모델의 윤곽 형태를 조절하는 것이다. 좀 더 구체적으로, 초기값으로 설정된 기준점은 계속 유지되는 것이 아니라, 각각의 기준점에서 주변 윤곽 에너지 값에 따라 추적하는 실제 객체 윤곽과의 편차를 최소화하는 방향으로 갱신된다.

영역 분할기법 중 하나인 윤곽선 추출 기법은 특정한 물체의 형태를 추출하기도 하고, 그 면적이나 주위의 길이를 측정하며, 두 화상간의 차이를 구할 수도 있다. 영상에서 윤곽선은 물체가 가지는 특징으로서 배경과 이루는 경계로 표현되기도 하고, 물체 자체를 식별하는 특징 값으로 사용된다. 본 논문에서는 추적 대상인 객체의 최외곽 영역은 그 특징상 폐곡선을 이루고 있으며 또한 이 폐곡선이 동적 특징을 포함하고 있기 때문에 변형모델인 능동 윤곽선 모델을 적용하여 윤곽선 검출의 정확성을 향상할 수 있다.

스네이크 알고리즘은 ACM으로서 찾고자 하는 객체 주위의 폐곡선을 사용하여 윤곽선을 찾는 방법이다. M. Kass가 제안한 이 알고리즘은 물체의 윤곽선을 능동적으로 찾아주기 때문에 영상분할이나 객체 추적과 같은 분야에 널리 적용되고 있다. 물체의 에너지 특성을 이용한 스네이크 알고리즘은 크게 외부 에너지와 내부 에너지로 구성 하여, 이들 사이의 관계로서 모델의 동적인 모습을 표현한다. 내부 에너지는 모델의 곡률에 따른 에너지로서 원형에 근사할수록 값이 감소하며, 외부 에너지는 스프링 함수로서 모델의 변화가 외부의 점과 갖는 위치관계로 한 점에 근사할수록 값이 감소한다. 그러므로 스네이크 알고리즘은 두 에너지의 값이 최소화되는 과정에서 물체의 윤곽선을 추출하는 알고리즘이다. 스네이크 알고리즘은 추적하는 객체의 주변에 가깝게 초기값을 정해줄수록 더 정확하게 물체의 윤곽선을 찾게 된다. 본 논문에서는 초기값을 수동으로 32개로

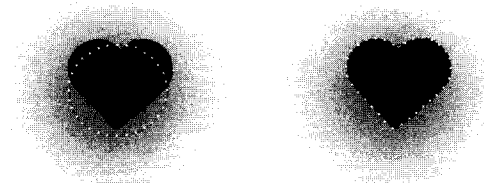


그림 1. 초기값을 이용한 윤곽 곡선 ( $n=32$ )  
Fig. 1. control points defining the contour curve. ( $n=32$ )

설정하였다. 초기의 위치를 연속적인곡선  $S$ 를 이산적인 점들로 표현하면 그림 1과 같다.

이것은 몇 개의 점으로 연결된 모든 곡선을 표현하게 된다. 스네이크 알고리즘은 에너지 함수를 이용하여 원하는 곡선을 나타낸다.

즉, 곡선의 최소값이 되는 에너지 함수  $E(S)$ 를 이용하여 모델을 정의하는 방법이다. 스네이크 알고리즘은 찾고자 하는 물체의 초기 위치를 정해주면, 반복적인 수행을 통하여 3개의 에너지 값이 최소가 되는 지점으로 초기 점들을 이동시키게 된다. 이처럼 스네이크는 초기 값에 의해 다른 물체의 윤곽선을 찾게 되므로, 초기 값의 의존도가 높다. 스네이크 알고리즘 에너지 함수는 다음 식 2와 식 3과 같이 정의된다.

$$E_{snake} = \int (E_{internal}(v(s)) + E_{image}(v(s)) + E_{external}(v(s)))ds, \quad (2)$$

$$v(s) = (x(s), y(s)) \quad (3)$$

여기에서  $v(s)$ 는 스네이크 알고리즘의 위치를 나타내며,  $E_{internal}$ 은 윤곽선을 구부리는 내부 에너지이고,  $E_{image}$ 는 이미지 힘을 나타내고,  $E_{external}$ 는 외부 에너지를 나타낸다. 스네이크 알고리즘은 에너지 최소화 구조이므로 지역적 최소가 가능한 해의 집합을 이루도록 에너지 함수를 만들어야 한다. 에너지함수의 구현에 있어서 알고리즘의 복잡성을 개선하고 속도를 향상시키기 위해 경험적 알고리즘을 이용하며, 이 알고리즘을 사용하면 다음과 같은 에너지 함수의 지역 최소값을 해로 얻는다.

$$E_{snake} = \int (\alpha(s)E_{continuity}(v(s)) + \beta(s)E_{curvature}(v(s)) + \gamma(s)E_{image}(v(s)))ds. \quad (4)$$

이때  $\alpha, \beta, \gamma$ 는 상수이고, 에너지 함수를 구성하는 요소 가운데 연속성(continuity)은 곡선을 수축 시키는 역할을 수행한다. 만일  $n$ 개의 점이 주어졌다면,  $v_i = (x_i, y_i), i = 1, \dots, n$ 로 표현되며, 주어진 점들의 8개의 이웃 화소들의 에너지 값을 구하고, 각 점들 간의 거리를 최소화 해준다. 그러나 윤곽선의 각 픽셀간의 거리를 최소화하는 것은 점들을 한 점으로 모이게 하는 문제를 수반한다. 따라서 두 점간의 거리를 모든 점간의 거리의 평균을 구하여 이 값과의 차이를 이용하여 나타내게 한다. 즉 각 점들의 거리가 평균에 가까운 거리에 있을 경우 에너지는 최소값을 갖는다.

곡률 에너지  $E_{curvature}$ 는 스네이크 알고리즘이 완

만성을 유지하도록 하는 값으로 다음 식 5에 의해 구해진다.

$$E_{curvature} = |v_{i+1} - 2v_j + v_{i-1}| \quad (5)$$

한편 영상 에너지  $E_{image}$ 는 스네이크 알고리즘을 사용하여 표현하고자 하는 곡선의 모양을 나타낸다. 물체의 윤곽선에 존재하는 화소들은 급격한 밝기값의 변화를 가지는 특성을 이용한 것으로, 영상에서  $v(s)$ 의 위치가 물체의 윤곽선이 되도록 정의한다. 영상 에너지는 밝기값의 변화도에 의해서 계산이 되며 식 6과 같다. 이때  $G_{max}$ 는 주변 화소들 중 가장 큰 밝기값의 변화를 나타내며,  $G_{min}$ 은 가장 작은 밝기값의 변화를 나타낸다.

$$E_{image} = \frac{G_{min} - G_j}{G_{max} - G_{min}} \quad (6)$$

## 2. 스네이크 알고리즘 문제점 및 해결방안

본 논문에서는 스네이크 알고리즘의 특성을 이용하여 객체의 형태를 구하고 그 형태를 사용하여 영상에서의 목표 객체를 추적하게 된다. 그러나 스네이크 알고리즘은 여러 가지 최소화 알고리즘이 제안되었음에도 불구하고 다음과 같은 문제점들을 지니고 있다.

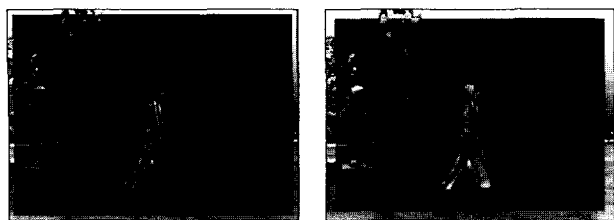
- (1) 내부 에너지 함수의 기능에 의해 복잡한 모양을 지닌 물체의 윤곽선을 추출하지 못한다.
- (2) 정점들이 한 번에 이동할 수 있는 범위가 한정되어 있기 때문에 많은 수행시간을 필요로 한다.
- (3) 초기 스네이크의 위치와 모양에 대해 높은 의존성을 지닌다.

본 논문에서 제안하는 객체 추적 시스템은 실시간으로 수행되어야 하는데, 스네이크 알고리즘은 앞 절에서 제시한 것처럼 몇 가지 문제점들을 가지고 있다. 복잡한 배경에서는 주변 윤곽 에너지로 인하여 객체를 정확하게 인식 할 수 없고, 정점의 이동 범위가 한정되어 있어 많은 수행 시간을 필요로 한다. 그래서 본 논문에서는 움직임 추정 기법을 사용하여 문제점들을 해결하고자 한다.

## III. 움직임 예측형 능동운곽모델

### 1. 영상 전처리 과정 필요성 및 방법

제안된 추적 기술의 근간이 되는 스네이크 알고리즘



(a) (b)

그림 2. 가우시안 영상 차분 결과  
 (a) kernel value 3-13, (b) kernel value 3-23.  
 Fig. 2. Gaussian images subtraction results.  
 (a) kernel value 3-13, (b) kernel value 3-23

은 영상의 윤곽 에너지를 이용하기 때문에 잡음의 영향을 많이 받는다. 감시 시스템의 경우 영상이 입력되는 경로에 따라 다양한 잡음에 노출되어 있다. 잡음은 객체의 윤곽이 에너지값보다 크게 작용하여 특징점으로 잘못 인식하는 오류를 발생시키기 때문에 본 논문에서는 가우시안 저역통과필터를 사용하여 잡음을 제거한다. 최적의 저역통과필터는 잡음의 종류에 따라 다른 것이 보통이다.

가우시안 필터는 대표적인 저역통과필터로써 영상의 윤곽 영역을 흐리게 만들어서 배경 에너지의 역할을 감소시킨다. 다른 저역통과필터들과 다른 점은 표준편차인  $\sigma$  값과 커널 크기에 따라 흐려짐의 정도가 달라지는 것이다. 제안하는 추적 알고리즘을 보다 원활하게 수행하기 위하여 진처리 과정으로 서로 다른 커널값을 적용한 영상의 차분을 사용하였다. 다시 말해서  $\sigma$  값과 커널 크기에 따라 흐려짐 정도가 달라지므로 서로 다른 커널을 사용해서 필터링 된 두 영상의 차분으로 윤곽 에너지의 값을 보다 강하게 나타낼 수 있다. 그림 2는 서로 다른 커널을 사용하여 차분한 영상이다. 커널값이 10 이상으로 커진다고 하여도 윤곽 영역을 추출하는 것에는 크게 변화가 없다.

이것은 매우 우수한 대역통과 필터의 역할을 수행하기 때문에 제안하는 알고리즘의 효율성을 높일 수 있다.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (7)$$

수식 7은 2차원 가우시안 필터를 나타내고  $\sigma=1$ 일 경우에는 그림 3와 같은 분포를 갖게 된다. 단순한 필터의 경우 주변 화소값과 평균값을 이용하는 경우가 많은데 이것은 주변 값에 따라 원래 영상의 형태가 달라질 수 있다. 본 논문에서는  $\sigma=19$ , 커널 크기는 각각 3과 13으로 실험하였다.

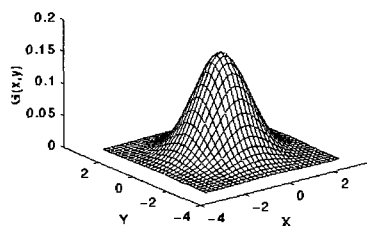


그림 3. 2-D 가우시안 분포도 ( $\sigma=1$ )  
 Fig. 3. 2-D Gaussian distribution chart ( $\sigma=1$ ).

### 2. 움직임 추정 기법

움직임 추정 기법에는 광류(optical-flow), 칼만 필터 등의 알고리즘들이 있으며, 본 논문에서는 칼만 필터를 적용하였다. 칼만 필터란 잡음에 의해 간섭 받는 선형 동적 시스템에서 상태 벡터  $x_k$ 의 최적의 추정값인  $\hat{x}_k$ 를 구하기 위한 재귀 알고리즘이다. 여기서 k는 시간을 나타낸다. 칼만 필터링은 크게 예측, 측정, 수정의 세 단계로 이루어진다.

예측단계에서는 상태 벡터  $\hat{x}_k$ 와 상태 추정 오차의 공분산 행렬  $P_k^-$ 의 예측이 이루어진다. (-)는 아직 측정을 통한 수정 단계가 이루어지지 않았음을 나타낸다. 예측의 단계가 끝나고 나면 측정 단계에서 칼만 이득,  $K_k$ 를 계산한다. 이 칼만 이득의 값과 예측단계의 상태 벡터  $\hat{x}_k^-$ 으로 상태 벡터  $\hat{x}_k$ 를 수정한다. 이때  $z_k$ 는 측정 벡터, H는 상태 벡터와 측정 벡터 사이의 관계를 나

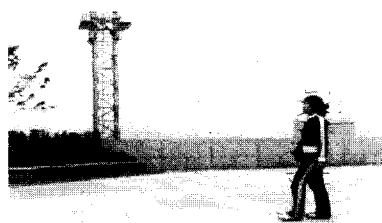


그림 4. 움직임 벡터 추출.  
 Fig. 4. Motion vector extraction.

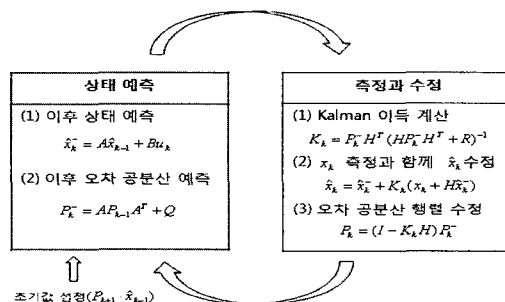


그림 5. 칼만 필터 상태도  
 Fig. 5. Kalman filter state diagram.

타내는 관측 행렬을 나타낸다. 또한 다음 단계의 예측 단계를 위한 오차 공분산 행렬도 수정하게 된다. 이러한 과정이 순환함으로써 칼만 필터가 완성된다. 아래 그림 5는 칼만 필터의 상태를 나타낸 것이다.

본 논문에서는 초기형태 이동된 정점들을 측정벡터로 하고 다음 프레임에서 찾아야 할 정점들을 상대벡터로 놓고 칼만 필터 예측 알고리즘을 사용하여 초기값을 형성한다.

### 3. 제안된 알고리즘

앞 절에서 언급한 스네이크 알고리즘의 문제점들을 극복하기 위한 방법으로 다음과 같은 방법을 제안한다. 초기 입력영상으로부터 적당한 기준점을 객체의 윤곽선에 할당한다. 이 작업은 주로 수동으로 이루어진다. 바람직한 기준점은 한 영상에서 기준점으로 선택된 특징점이 다음 영상에서도 특징점으로 연결되어 있어야 한다. 2차원 영상에서 우리는  $n$ 개의 기준점을  $2n$ 차원 벡터를 사용해서 다음과 같이 표현할 수 있다.

$$x = [x_1, \dots, x_n, y_1, \dots, y_n]^T. \quad (8)$$

본 논문에서 적용될 실험에 사용하는 기준점의 수는  $n=32$ 로 했다. 기준점을 자동적으로 결정하는 방법은 Tian등<sup>[7]</sup>에 의해 논의되었다. 기준점의 역할은 모델의 윤곽 형태를 조절하는 것이다. 좀 더 구체적으로, 초기값으로 설정된 기준점은 계속 유지되는 것이 아니라, 각각의 기준점에서 주변 윤곽 에너지 값에 따라 추적하는 실제 객체 윤곽과의 편차를 최소화하는 방향으로 갱신된다. 다음과 같이 초기값을 설정하는 것은 초기 스네이크 알고리즘을 적용하는데 초기값의 위치와 형태가 많은 작용을 하는 것에 대한 문제점을 해결하기 위한 방법으로 제시한다. 그리고 복잡한 배경에서도 보다 정확한 윤곽선을 찾아내는데 강건한 모습을 보여준다.

초기값 지정 이후 스네이크 알고리즘을 적용한 이후 다음 프레임의 영상이 입력되면 움직임 추정 기법을 사용하여 다음 초기값을 설정한다. 객체의 이동 방향과 이동 거리를 움직임 추정 기법을 사용하여 구하고 정점을 이동시킨 후 스네이크 알고리즘을 적용한다. 이것은 복잡한 배경에서도 객체 윤곽 형태 보존률이 높고 정점의 이동 거리가 작아지기 때문에 수행 시간이 짧아지는 장점을 가지고 있다. 식 9는 스네이크 에너지 함수에 움직임 에너지 함수를 더한 것이다.



그림 6. 움직임 벡터를 사용한 다음 영상 초기값 생성 방법.

Fig. 6. Using motion vector to generate initial point of next frame.

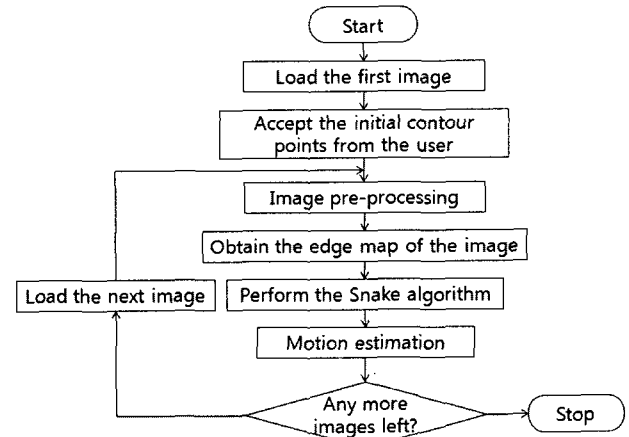


그림 7. 제안한 알고리즘 흐름도

Fig. 7. Proposed the algorithm flowchart.

$$E = \int (\alpha(s)E_{continuity} + \beta(s)E_{curvature} + \gamma(s)E_{image} + \lambda(s)E_{motion}) ds \quad (9)$$

그림 6은 움직임 추정 기법을 사용하여 얻어진 움직임 벡터를 사용하여 다음 초기값을 구하는 방식을 보여준다.

그림 7은 앞에서 설명한 제안된 알고리즘의 흐름도이다. 영상이 입력되면 가우시안 필터 차분을 하고 스네이크 알고리즘을 적용 후 이전 영상을 이용한 움직임 벡터를 얻고 다시 초기값을 설정하는 것을 반복적으로 수행하는 과정을 보여준다.

## IV. 실험 결과

제안된 움직임 예측과 결합된 능동윤곽모델을 실험하기 위해서 실내와 실외에서 획득한  $320 \times 240$  영상을 사용하였다. 그림 8은 실외에서 획득한 영상에 대해서 타원을 초기값으로 사용한 결과를 보인다.

그림 8(b)는 초기 영상의 초기 윤곽을 타원으로 설정한 후 스네이크 알고리즘을 적용한 결과이며, 8(d)는 수동적으로 초기값을 설정한 후 스네이크 알고리즘을 적

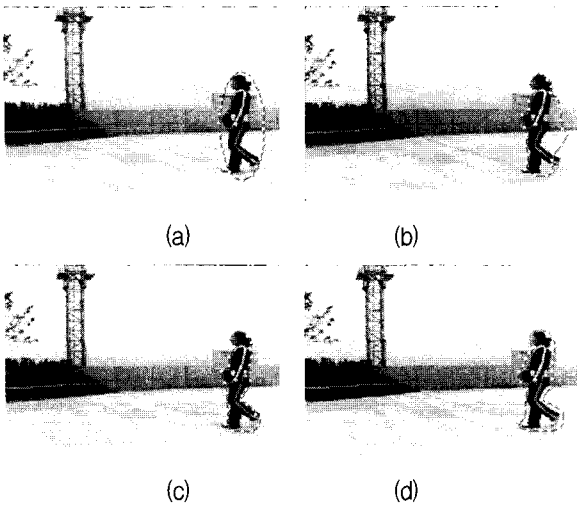


그림 8. 실외 영상 실험결과 (a)타원, (b)초기 스네이크 알고리즘 수행 결과, (c)기준점을 할당한 결과, (d)기준점 할당한 후 스네이크 알고리즘을 적용한 최종 결과.

Fig. 8. Experimental results with outdoor test images.

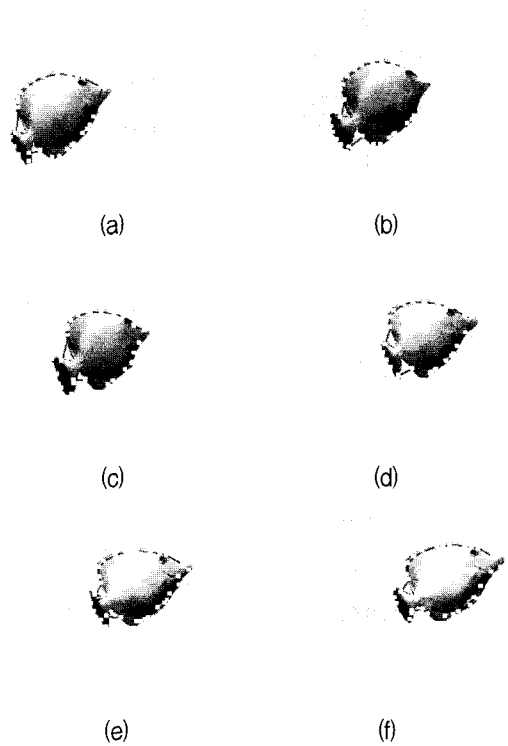


그림 9. 정형 객체에 대한 움직임 추정 적용 결과 (a) 1st frame (b) 5th frame (c) 10th frame (d) 15th frame (e) 18th frame (f) 21th frame.

Fig. 9. Experimental results of rigid object using motion estimation (a) 1st frame (b) 5th frame (c) 10th frame (d) 15th frame (e) 18th frame (f) 21th frame.

용한 결과이다. 위의 결과는 스네이크 알고리즘이 초기 값에 많은 영향을 받는 것을 보여준다.

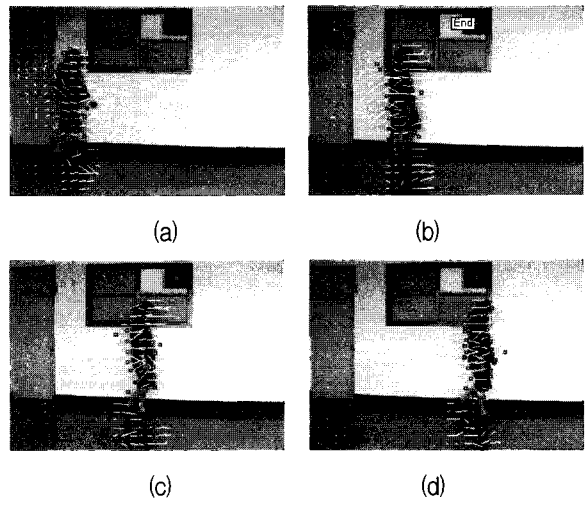


그림 10. 움직임 추정 적용 결과 (a) 4th frame (b) 12th frame (c) 20th frame (d) 28th frame.

Fig. 10. Result of motion estimation (a) 4th frame (b) 12th frame (c) 20th frame (d) 28th frame.

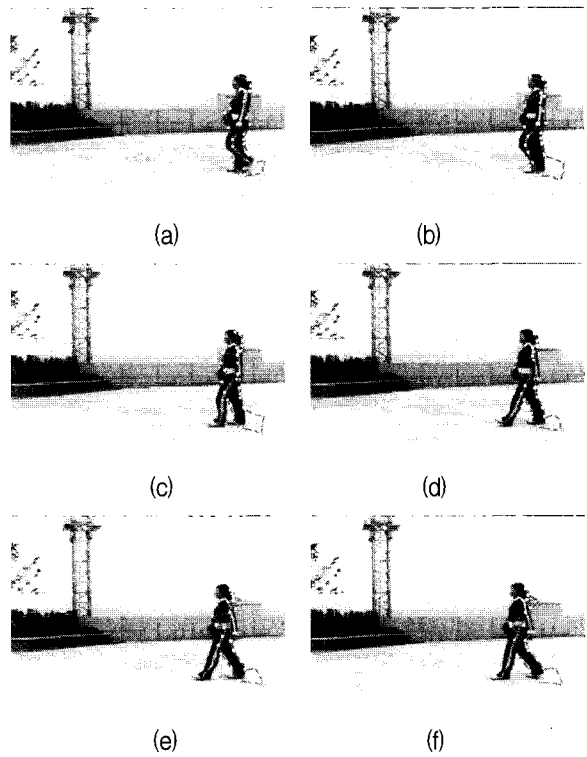


그림 11. 스네이크 알고리즘만을 적용한 결과 (a)-(f) 2nd frame - 6th frame.

Fig. 11. Result of Snake algorithm (a)-(f) 2nd frame - 6th frame.

그림 9는 정형 객체에 대한 결과를 보여준다. 비정형 객체와 비교하면 보다 정확한 형태를 나타낸다.

그림 10은 움직임 벡터값의 방향과 크기를 함께 보여주며 그 결과에 따라 스네이크 알고리즘 결과가 다음과 같이 나오는 것을 보여준다.

표 1. 스네이크 알고리즘 적용한 결과 오차 픽셀과 평균.

Table 1. Snake algorithm apply to error pixel and average results.

(단위: pixel)

	2_frame	3_frame	4_frame	5_frame	6_frame	7_frame
오차 픽셀수	68	182	257	317	361	403
오차 평균	2.125	5.6875	8.03125	9.90625	11.28125	12.59375

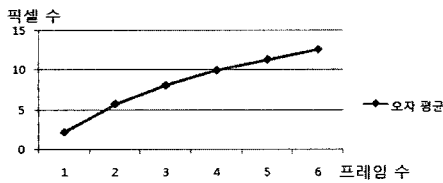


그림 12. 스네이크 알고리즘 적용한 결과 오차 픽셀과 평균 그래프.

Fig. 12. Snake algorithm apply to error pixel and average results graph.

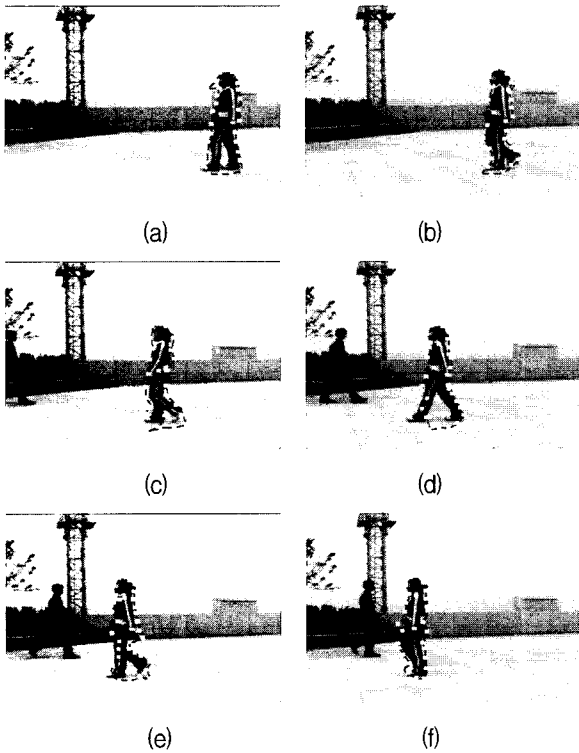


그림 13. 움직임 추정 적용 결과 (a) 4th frame (b) 10th frame (c) 15th frame (d) 22th frame (e) 26th frame (f) 28th frame.

Fig. 13. Result of motion estimation (a) 4th frame (b) 10th frame (c) 15th frame (d) 22th frame (e) 26th frame (f) 28th frame.

그림 11은 실시간으로 입력되는 영상에서 스네이크 알고리즘만을 적용한 영상이다. 초기 영상에서만 초기

표 2. 움직임 예측 스네이크 알고리즘 적용한 결과 오차 픽셀과 평균.

Table 2. Motion estimation snake algorithm apply to error pixel and average results.

(단위: pixel)

	4_frame	10_frame	16_frame	23_frame	27_frame	29_frame
오차 픽셀수	31	35	54	55	48	45
오차 평균	0.96875	1.09375	1.6875	1.71875	1.5	1.40625

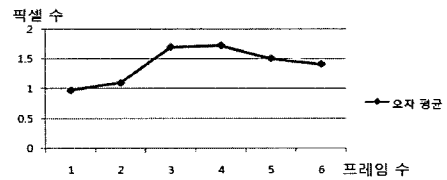


그림 14. 움직임 예측 스네이크 알고리즘 적용한 결과 오차 픽셀과 평균 그래프.

Fig. 14. Motion estimation snake algorithm apply to error pixel and average results graph.

값을 적용한 후 다음 영상부터는 이전 스네이크 알고리즘 결과를 초기값으로 사용한다.

스네이크 알고리즘만을 적용한 결과값은 이전 영역에서 객체가 움직인 이후에도 에너지값이 충분히 높게 유지되면 정점의 위치가 변하지 않게 된다. 따라서 이전 영상의 스네이크 알고리즘 결과를 초기값으로 가지면 배경 에너지에 많은 영향을 받는다. 이와 같은 문제를 해결하기 위해 움직임 추정을 통하여 스네이크 알고리즘의 문제점을 극복하였다.

그림 13은 이전 영상에서 스네이크 알고리즘 적용 후 결과값을 칼만 필터를 적용하여 객체의 움직임을 찾아내고 이동한 방향과 거리만큼 정점의 위치를 이동하여 초기값으로 적용한 결과이다.

표 1, 2 그림 12, 14의 결과를 보면 움직임 예측 스네이크 알고리즘의 결과가 오차율을 많이 줄일 수 있음을 보여준다. 그리고 실험결과 기존 알고리즘을 적용하면 약 2 frame/sec의 속도가 나오고 제안된 알고리즘을 적용하면 약 5 frame/sec의 결과가 나타난다. 이것은 특징값들의 이동거리가 줄어들기 때문에 계산량이 줄어든 것을 보여준다.

## V. 결론

본 논문에서는 움직임 추정 기술을 적용한 스네이크 알고리즘을 이용하여 비정형 객체를 실시간으로 추적하

는 방법을 제시하였다. 초기 영상에서 수동적 초기값을 바탕으로 스네이크 알고리즘 수행을 거친 후 객체의 윤곽을 찾고 이를 바탕으로 칼만 필터를 사용하여 움직임을 예측한 후 이동 방향과 거리만큼 이동하여 초기값으로 설정 후 다시 스네이크 알고리즘 과정을 거친다. 본 논문에서는 스네이크 알고리즘의 문제점을 움직임 추정 기법을 사용하여 해결한다. 움직임 추정 기법인 칼만 필터를 사용하여 이전 영상에서 움직인 방향과 거리를 예측하여 그 이동 거리만큼을 각각의 기준점들에 적용시켜서 이를 스네이크 알고리즘 수행하는 것이 배경 에너지의 영향을 최소화하고 수행 반복 횟수도 적어 실시간이 가능한 결과를 얻을 수 있다.

실험 영상에서 보다 움직임이 많거나 복잡한 영상에 적용하기 위해서는 초기값의 수를 늘리고 배경 에너지와 객체 윤곽 에너지를 구분할 수 있는 정보를 획득하여 보다 정확한 윤곽을 추출해야 한다. 예를 들어 칼라를 사용하여 객체와 배경을 구분하는 기술에 대한 연구가 진행되어야 할 것이다.

## 참 고 문 헌

- [1] Y. Boykov, D.P. Huttenlocher, "Adaptive Bayesian recognition in tracking rigid objects," *Computer Vision and Pattern Recognition*, vol.2, pp.697-704, 2000.
- [2] I. Haritaoglu, D. Hartwood, and L. S. Davis, "Real-time surveillance of people and their activities," *IEEE Trans. PAMI*, vol. 22, pp.809-830, 2000.
- [3] S. J. Mckenna, Y. Raja, S. Gong, "Tracking color objects using adaptive mixture models," *Image and Vision Computing*, vol. 17, pp.225-231, 1999.
- [4] R. Plankers, P. Fua, "Tracking and modeling people in video sequences," *Comp. Vision and Image Understanding*, vol. 81, pp. 285-302, 2001.
- [5] M. Kass, A. Witkin, and D. Terzopoulos, "Snake: Active contour models," *International Journal of Computer Vision*, pp.321-331, 1998.
- [6] A. A. Amini, T. E. Weymouth, and R. C. Jain, "Using dynamic programming for solving variational problems in vision," *IEEE Trans. Patt. Anal. Machine. Intell.*, pp.855-867, 1990.
- [7] S. Tanimoto and T. Pavlidis, "A hierarchical data structure for picture processing," *Comput. Graphics Image Process.* vol. 4, pp. 104-119, 1996.
- [8] D. J. Williams, Mubarak Shah, "A fast algorithm for active contours and curvature estimation," *CVGIP: Image Understanding*, vol. 55, no. 1, pp. 14~26, 1992.
- [9] L. D. Cohen, "On active contour models and balloons," *CVGIP: Image Understanding*, vol. 53, no. 2, pp. 211~218, March 1991.
- [10] S. Kim, J. Kang, J. Shin, S. Lee, J. Paik, S. Kang, B. Abidi, and M. Abidi, "Optical flow-based tracking of deformable object using a non-prior training active feature model," *Proc. PCM 2004, LNCS*, vol. 3333, pp. 69-78, December 2004.



## 저자 소개



김 대 희(정회원)  
 2005년 강남대학교 컴퓨터공학과  
 학사졸업  
 2007년 현재 중앙대학교 첨단영상  
 대학원 영상공학과  
 석사과정  
 <주관심분야 : 객체추적, 신호처  
 리>



이 동 은(학생회원)  
 2005년 한서대학교 컴퓨터공학과  
 학사졸업  
 2007년 현재 중앙대학교 첨단영상  
 대학원 영상공학과  
 석사과정  
 <주관심분야 : 워터마킹, 객체추적,  
 영상압축>



백 준 기(정회원)  
 1984년 서울대학교 제어계측  
 공학과 학사졸업  
 1987년 노스웨스턴대학교 전기 및  
 컴퓨터 공학과 석사졸업  
 1990년 노스웨스턴대학교 전기 및  
 컴퓨터 공학과 박사졸업  
 2007년 현재 중앙대학교 첨단영상대학원  
 영상공학과 교수  
 <주관심분야: 영상복원, 신호처리, 반도체>