

〈論 文〉

능동 보모델을 이용한 영상추적 알고리즘

이진우* · 이재웅* · 박광일* · 오준호**

(1994년 8월 13일 접수)

Visual Tracking Algorithm Using the Active Bar Models

Jin-Woo Lee, Jea-Woong Yi, Kwang-Il Park and Jun-Ho Oh

Key Words : Active Bar Models(능동 보모델), Visual Tracking(영상추적), Snake Algorithm
(스네이크 알고리즘)

Abstract

In this paper, we consider the problems of tracking an object in a real image. In evaluating these problems, we explore a new technique based on an active contour model commonly called a snake model,⁽¹⁾ and propose the active bar models to represent target. Using this model, we simplified the target selection problems, reduced the search space of energy surface, and obtained the better performances than those of snake model. This approach improves the numerical stability and the tendency for points to bunch up and speed up the computational efficiency. Representing the object by active bar, we can easily obtain the zeroth, the first, and the second moment and it facilitates the target tracking. Finally, we present the good result for the visual tracking problem.

1. 서 론

본 논문에서는 1대의 CCD 카메라에서 얻어진 물체의 영상정보를 이용하여 이동하는 표적체를 계속 주시할 수 있는 영상 추적문제를 고려하고자 한다. 영상 추적은 비행체의 궤적 추종수행, 항법변수추출 및 유도탄 개발 등의 분야에서 요구되는 문제로 이의 해결을 위해 실시간 표적인식 및 추적기법 개발이 필수적이다. 본 논문에서는 능동 보모델링기법을 이용하여 표적체를 모델링하였으며 이를 에너지 최적화방법으로 풀어 실시간으로 표적체 인식 및 중심좌표 추출을 수행하였다.

에너지 최적화에 의한 물체의 경계추출방법으로 1987년 Kass, Witkin과 Terzopoulos⁽¹⁾등은 snake

algorithm을 제안한 바 있다. 그들은 물체의 경계추출을 위해 active contour(일명 snake)를 구성하고 이를 에너지로 정량화하여 변형하는 물체의 경계를 추출하였다. snake algorithm은 에너지 최적화 과정에서 skake의 불안정한 거동과 수축현상을 보이는 것이 지적되고 있으며^(9,10) 이를 개선하고자 Cohen,⁽⁵⁾ Williams,⁽⁶⁾ Amini^(9,10)등은 snake모델에 구속조건을 첨부하였고 다른 종류의 최적화방법을 사용하여 해결을 시도하였다.

본 논문에서는 다수의 능동 보모 표적체를 모델링하여 snake 모델에서 발생하는 불안정성 및 수축현상을 개선하고자 하였다. 능동 보는 에너지 함수에 의해 정의되는 변형 가능한 에너지 최적 막대로 snake 모델보다 적은 계산량으로 최적해 및 표체의 0차, 1차, 2차 모멘트를 구할 수 있다. 능동 보모델을 통해 최적 에너지 탐색방향을 1차원으로 제한되며 이를 통해 간단한 최적화방법으로도 최적

*한국과학기술원 기계공학과

**정회원, 한국과학기술원 기계공학과

해를 구할 수 있다. 능동 보모델을 사용하여 구현된 추적 알고리즘을 사용하여 본 논문에서는 이동하는 표적체의 중심계좌를 실시간으로 추출하였다. 한편 능동 보모델에 면적구속 및 경계구속을 가함으로써 배경 및 폐색에 대해서도 강인한 결과를 얻을 수 있다.

본 논문의 구성으로는 2장에서 능동 보에 의한 표적체의 모델링 및 에너지함수를 기술하였고 3장에서는 능동 보의 에너지함수에 대한 최적화과정 및 표적체의 모멘트량 계산식, 추적 알고리즘의 구성을 설명하였다. 4장에서는 추적 알고리즘을 통해 얻은 결과를 제공하였고 마지막으로 5장에서 결론을 맺었다.

2. 능동 보모델

2.1 능동 보모델

능동 보는 에너지 범함수로 정의되는 변형 가능한 에너지 최적 막대로 에너지 차이에 의해 초기 형태로부터 변형하여 최소 에너지를 갖는 형태를 유지하게 된다. 능동 보는 식(1)의 벡터 형태로 표현되며 Fig. 1에 개념도를 나타내었다. 식(1)에서

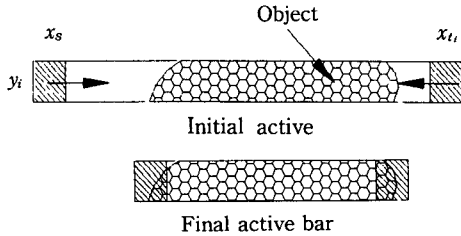


Fig. 1 An active bar model

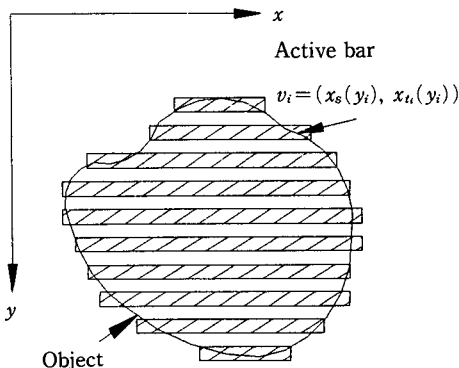


Fig. 2 Object model by the multiple active bar models

x_s 와 x_t 는 각각 영상 좌표계의 x 축에서 능동 보의 시작좌표 및 끝좌표를 나타내며 y 는 y 축 좌표를 의미한다. Fig. 2는 능동 보에 의해 물체를 표시한 결과를 나타낸다.

$$v(y) = (x_s(y), x_t(y)) \quad (1)$$

2.2 에너지모델

1987년 Kass는⁽¹⁾ “Snake : Active Contour Models”에서 에너지 최적화방법에 의해 물체의 경계를 추출한 결과를 제시하였으며 물체의 경계 추출을 위한 에너지함수로 내부에너지, 구속에너지 및 외부 에너지함수를 제공하였다. 이번 절에서는 Kass에 의해 제안된 에너지함수를 능동 보에 대해 적용한 결과 및 이를 공간축과 시간축에 대해 이산화한 식을 제공할 것이다.

능동 보는 구속에너지, 내부에너지, 외부에너지로 구성되며 전체 에너지는 식(2)와 같이 표현할 수 있다. 각 에너지함수에 대한 설명은 다음 절에서 자세히 기술하기로 한다.

$$E_{bar} = \int_y [E_{con}(v(y)) + E_{image}(v(y)) + E_{ext}(v(y))] dy \quad (2)$$

수치계산을 통해 최적해를 구하기 위해서는 능동 보를 이산화하는 것이 필요하며 이는 공간축에 대한 이산화와 시간축에 대한 이산화로 구분할 수 있다. 공간축 상의 이산화를 통해 능동 보는 다음과 같은 벡터로 표현되며,

$$V = \{v_1, v_2, v_3, \dots, v_N\} \\ v_i = (x_{s_i}(y_i), x_{t_i}(y_i)) \quad (3)$$

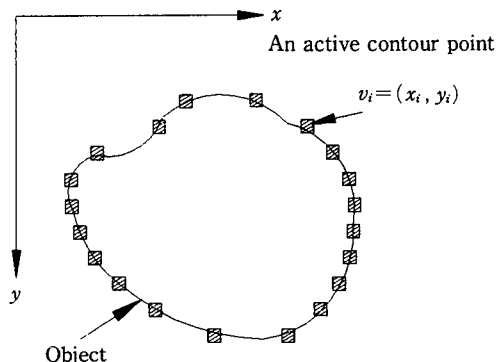


Fig. 3 Representing an object by the snake model

이를 다시 시간에 대해 이산화하여 시각 t 에서 다음과 같은 식을 얻을 수 있다.

$$V^t = \{v_1^t, v_2^t, v_3^t, \dots, v_N^t\}$$

$$v_i^t = (x_{s_i}^t(y_i), x_{t_i}^t(y_i)) \quad (4)$$

식(4)를 식(2)에 적용함으로써 이산화된 최종 에너지 모델을 다음과 같이 얻어진다.

$$E_{bar}(V^t) = \sum_i^N [E_{con}(v_i^t(y_i)) + E_{image}(v_i^t(y_i)) + E_{ext}(v_i^t(y_i))] \quad (5)$$

2.2.1 구속에너지

영상정보가 매우 빠른 샘플링 시간을 가지고 입력된다면 새로 입력된 영상정보와 이전 영상정보 사이에서 발생하는 표적체의 변화는 거의 없다고 가정할 수 있다. 따라서 표적체의 형태 변화를 구속하기 위해 본 논문에서는 구속에너지를 구성하였으며 이를 다시 면적 구속에너지와 경계 구속에너지로 구분하였다.

면적 구속에너지는 표적체의 면적변화를 구속하는 에너지 항으로 식(6)으로 표현할 수 있으며 이를 최소화함에 의해 능동 보는 일정 면적을 유지하려는 성질을 가진다.

$$E_{area}(v) = \left| \frac{dA}{dt} \right|^2 \quad (6)$$

경계 구속에너지는 샘플링 구간사이에서 발생하는 표적체의 형태 변화가 미소하다는 가정을 정량화한 에너지를 항으로 이를 최소화함에 의해 능동 보는 급격한 변화를 제한 받게 된다. 경계구속에너지는 연속에너지와 곡률에너지로 구분되며 각각은 식(7)과 같다.

$$E_{boundary}(v(y)) = \frac{1}{2} \cdot (E_{cont}(v(y)) + E_{curv}(v(y)))$$

$$= \frac{1}{2} \cdot [|\alpha(y) \cdot v_y(y)|^2 + |\beta(y) \cdot v_{yy}(y)|^2] \quad (7)$$

연속에너지 $E_{con}(v)$ 은 y 에 의한 능동 보의 일차 미분항으로 능동 보의 불연속을 의미하고, 곡률에너지 $E_{curv}(v)$ 은 y 에 의한 능동 보의 이차 미분항으로 능동 보의 격임을 나타낸다. 각 에너지를 최소화함에 따라 능동 보는 완만한 형태를 선호하려는 성질을 갖는다. 식(7)에서 $\alpha(s)$ 와 $\beta(s)$ 는 가중치 계수를 각 에너지의 상대적 크기를 조절하기

위해 사용되며 능동 보의 시작점과 끝점에 대해 $\alpha(s) = (\alpha_s, \alpha_t)$, $\beta(s) = (\beta_s, \beta_t)$ 의 형태를 갖는다.

이산화를 통해 식(6)과 식(7)은 다음과 같이 구해진다.

$$E_{area}(v_i) = |A(v_i^t) - A(v_i^{t-1})|^2 \quad (8)$$

$$E_{int}(v_i) = \frac{1}{2} \cdot \left[\left| \alpha_i \cdot \frac{dv_i}{dy} \right|^2 + \left| \beta_i \cdot \frac{d^2 v_i}{dy^2} \right|^2 \right] \quad (9)$$

$$\left| \frac{dv_i}{dy} \right|^2 \approx |v_i - v_{i-1}|^2$$

$$= (x_{s_i} - x_{s_{i-1}})^2 + (x_{t_i} - x_{t_{i-1}})^2 \left| \frac{d^2 v_i}{dy^2} \right|^2$$

$$\approx |v_{i-1} - 2 \cdot v_{i-1} + v_{i+1}|^2$$

$$= (x_{s_{i-1}} - 2 \cdot x_{s_i} + x_{s_{i+1}})^2 + (x_{t_{i-1}} - 2 \cdot x_{t_i} + x_{t_{i+1}})^2$$

본 논문에서는 계산속도의 증대를 위해 구속에너지의 이차식을 일차식으로 근사화하여 사용하였다.

$$E_{area}(v_i) \approx |A(v_i^t) - A(v_i^{t-1})| \quad (10)$$

$$E_{int}(v_i) = \left| \alpha_i \cdot \frac{dv_i}{dy} \right|^2 + \left| \beta_i \cdot \frac{d^2 v_i}{dy^2} \right|^2$$

$$\approx (\alpha_{s_i} \cdot |x_{s_i} - x_{s_{i-1}}| + \alpha_{t_i} \cdot |x_{t_i} - x_{t_{i-1}}| + (\beta_{s_i} \cdot |x_{s_{i-1}} - 2 \cdot x_{s_i} + x_{s_{i+1}}| + \beta_{t_i} \cdot |x_{t_{i-1}} - 2 \cdot x_{t_i} + x_{t_{i+1}}|) \quad (11)$$

2.2.2 외부에너지

외부에너지는 능동 보를 인위적으로 변형하기 위해 사용자에게 의해 제공되는 에너지항을 나타내며 실제 적용에서는 마우스 등과 같은 포인팅 디바이스를 사용함으로써 생성된다. 외부에너지는 상반되는 두 개의 에너지 항으로 구성되며 이들은 각각 Spring과 Volcano라 한다. 이들은 사용자에게 의해 지정된 영상 좌표로부터 능동 보까지의 거리에 비례 또는 반비례하는 에너지 항으로 능동 보에 인장력과 압축력을 가하게 된다.

2.2.3 영상에너지

영상에너지는 영상의 특정 부위를 추출하기 위해 사용되며 표적체의 경계를 추출하기 위해서는 영상의 밝기 변화율을 에너지함수로 사용한다.

$$E_{image}(v_i) = -(|\nabla I(x_{s_i}, y_i)|^2 + |\nabla I(x_{t_i}, y_i)|^2) \quad (12)$$

식(12)에서 I 는 영상의 밝기값을 나타내며 $E_{image}(v_i)$ 를 최소화함에 따라 능동 보의 에지(edge)를 선호하게 된다.

3. 영상추적

3.1 추적 알고리즘

영상 추적 알고리즘은 추적대상의 선정 및 표적

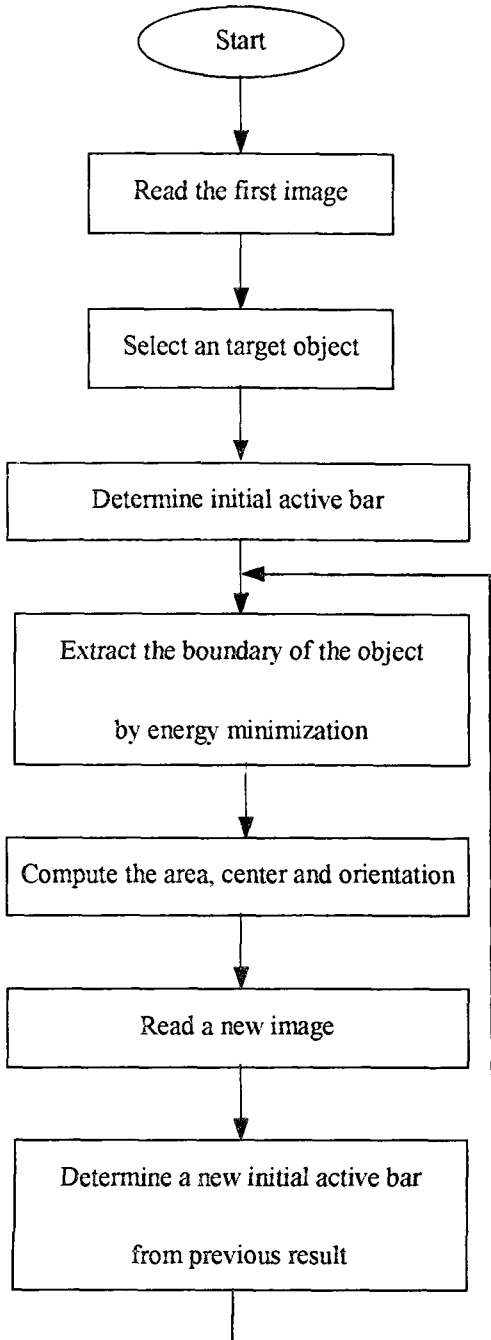


Fig. 4 Flow chart of the tracking algorithm

체 인식, 표적체의 중심좌표 계산 등의 일련 과정을 포함한다. 추적 알고리즘은 사용자가 추적대상을 포함하는 윈도우를 형성함으로써 시작하여 이 윈도우로부터 초기 능동 보가 결정되고 이어서 식 (5)의 에너지를 최적화하여 표적체의 경계추출 및 중심좌표 계산이 수행된다. 중심좌표 계산이 끝나면 새로운 영상이 입력되며 이전 단계에서 추출된 표적체의 경계로부터 새로운 능동 보를 구성하여 일련의 과정을 계속 수행한다. 전체 구성 알고리즘을 Fig. 4에 도식화하였다.

3.2 에너지 최적화

표적체의 경계 추출은 식(5)로 구성된 능동 보 에너지의 최적해를 구함으로써 이루어진다. 능동 보는 길이 방향으로 팽창과 수축만이 가능하므로 에너지 탐색방향을 길이방향으로 제한할 수 있다. 따라서 단순한 최적 알고리즘으로도 식 (5)의 최적해를 구하는 것이 가능하며 2차원 영역에 대해 최적해를 탐색하는 Snake 모델에 비해 계산량은 제곱근의 비율로 줄어든다. 만약 N_B 를 능동 보의 수, L 를 능동 보의 위치가능 영역의 길이라 할 경우 능동 보모델에 대한 계산량은 $2N_B L$ 에 비례한다. 한편 Snake 모델에 대해 dynamic programming을 사용하여 경계를 추출할 경우 계산량은 $N_s (L^2)^3$ 에 비례하며^(9,10) greedy algorithm을 사용하여 해를 구할 경우 $N_s (L^2)$ 에 비례한다.⁽⁸⁾ 여기서 N_s 는 Snake point의 개수이고 L^2 는 Snake point의 위치가능 영역의 면적이다.

본 연구에서는 능동 보의 x_s 와 x_t 에 각각 Brent's method⁽¹⁹⁾를 적용하여 표적체의 경계를 구하였다.

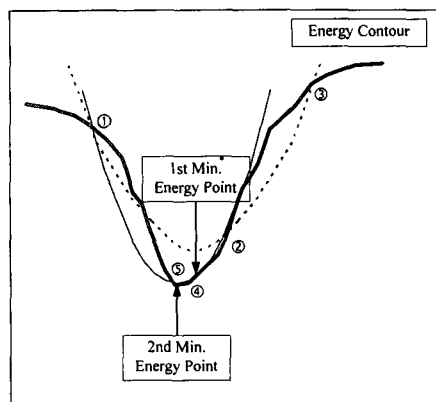


Fig. 5 Brent's method

Brent's method는 세점으로 주어진 탐색 구간으로부터 포물선의 꼭지점을 구하고 이를 포함하는 새로운 탐색 구간을 설정하여 최소치를 구해가는 방법이다. Fig. 5에서는 ①, ②, ③과 ①, ②, ④의 순서로 탐색구간을 축소하여 가며 최소치를 찾는 과정을 도해하였다. 한편 세점 $(a, f(a))$, $(b, f(b))$, $(c, f(c))$ 으로 포물선을 형성하였을 때 꼭지점의 좌표는 식(13)과 같이 얻을 수 있다.⁽¹⁹⁾

$$x_{\min} = b + \frac{1}{2} \cdot$$

$$\frac{(b-a)^2 \cdot [f(b) - f(c)] - (b-c)^2 \cdot [f(b) - f(a)]}{(b-a) \cdot [f(b) - f(c)] - (b-c) \cdot [f(b) - f(a)]} \quad (13)$$

3.2 모멘트 계산

능동 보모델에 의해 표적체의 경계가 추출이 이루어지고 나면 이로부터 표적체의 면적 및 중심좌표, 방향을 계산하는 것이 가능하다. 표적체를 능동 보모 모델링 하였으므로 이들 계산은 다음과 같이 쉽게 얻을 수 있다.⁽¹⁷⁾

○면적계산

$$A = \int_s dA = \sum_{i=1}^N (x_{ti} - x_{si} + 1) \quad (14)$$

○도심계산

$$\begin{aligned} x_{cg} &= \frac{1}{A} \int_s x dA \\ &= \frac{1}{A} \sum_{i=1}^N \left[\frac{(x_{ti} + x_{si}) \cdot (x_{ti} - x_{si} + 1)}{2} \right] \end{aligned} \quad (15)$$

$$y_{cg} = \frac{1}{A} \int_s y dA = \frac{1}{A} \sum_{i=1}^N [x_{ti} - x_{si} + 1] \quad (16)$$

○주축계산

$$\begin{aligned} M_z^2 &= \frac{1}{A} \int_s y^2 dA \\ &= \frac{1}{6A} \sum_{i=1}^N [x_{ti} \cdot (x_{ti} + 1) \cdot (2x_{ti} + 1) \\ &\quad - x_{si} \cdot (x_{si} - 1) \cdot (2x_{si} - 1)] \end{aligned} \quad (17)$$

$$\begin{aligned} M_z^x &= \frac{1}{A} \int_s x^2 dA \\ &= \frac{1}{A} \sum_{i=1}^N [(x_{ti} - x_{si} + 1) \cdot y_i^2] \end{aligned} \quad (18)$$

$$\begin{aligned} M_z^{xy} &= \frac{1}{A} \int_s xy dA \\ &= \frac{1}{2A} \sum_{i=1}^N [y_i \cdot (x_{ti} + x_{si}) \cdot (x_{ti} - x_{si} + 1)] \end{aligned} \quad (19)$$

$$a = \int_s x^2 dA - A \cdot x_{cg}^2$$

$$b = \int_s 2 \cdot x \cdot y dA - 2A \cdot x_{cg} \cdot y_{cg}$$

$$c = \int_s y^2 dA - A \cdot y_{cg}^2 \quad (20)$$

능동 보에 의해 표적체의 경계가 추출되었을 경우 표적체의 2차 모멘트는 식(17), (18), (19)과 같으며 이를 식(20)과 (21)에 대입하여 표적체의 주축방향 θ 를 얻을 수 있다.⁽¹⁸⁾

$$\theta = \frac{1}{2} \tan^{-1} \left[\frac{b}{a - c} \right] \quad (21)$$

4. 결 과

능동 보모델의 성능시험을 위해 실제로 이동하는 모형 표적체를 대상으로 영상추적을 시도하였으며 추적의 결과를 Fig. 6, Fig. 7, Fig. 8, Fig. 9에 나타내었다. 각 결과는 제안된 알고리즘을 통해 486급 pc를 사용하였을 경우 $N=20$ (능동 보의 개수)에 대해 10 frame/sec의 계산속도로 얻어진 결과이다.

Fig. 6은 이동하는 모형자동차를 대상으로 얻은 영상추적 결과를 나타낸다. (a), (c)는 대상 표적체의 초기 선정작업으로 사용자가 두 점을 지정하여 윈도우를 생성하는 과정이고, (b), (d)는 영상추적의 결과로써 얻은 중심의 궤적을 나타낸다. 다른 물체에 의한 폐색은 없는 것으로 가정하였으며 배경과의 경계구분이 비교적 명확한 경우에 대한 실험결과를 나타낸다. 이 경우 1 frame 사이의 표적체 이동이 탐색영역을 벗어나지 않는다면 지속적인 표적추적이 가능하였다.

Fig. 7은 형태가 일정하지 않은 물체의 경계 추출에 관한 결과를 보여 준다. 실험은 30초 동안 수행되었으며 지면관계상 그 중 일부의 결과만을 제공하였다. 이 결과로부터 형태의 변형 및 이동하는 표적에 대해서도 경계 추출이 잘 이루어짐을 알 수 있다.

Fig. 8는 식(20)의 계산결과로부터 구한 주축 방향을 표시한다. 이 결과로부터 물체의 방향성이 요구되는 상황에 능동 보모델의 적용이 가능함을 알 수 있다.

Fig. 9은 일시적인 폐색이 발생하였을 때의 결과를 보여주며 폐색이 제거된 후에 다시 표적체의 중

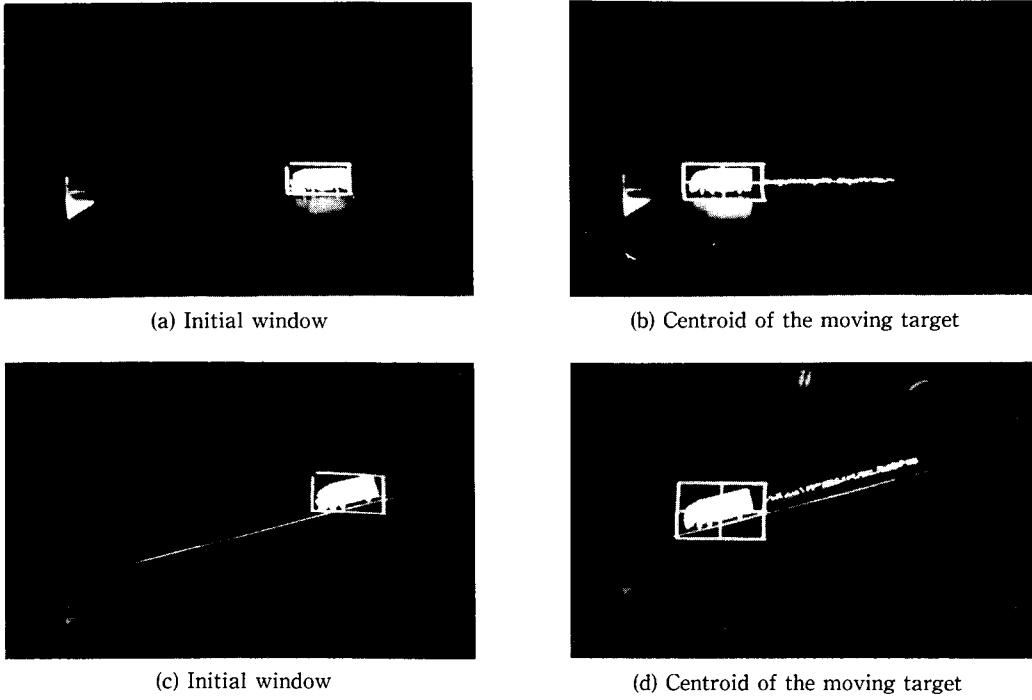


Fig. 6 Visual tracking result

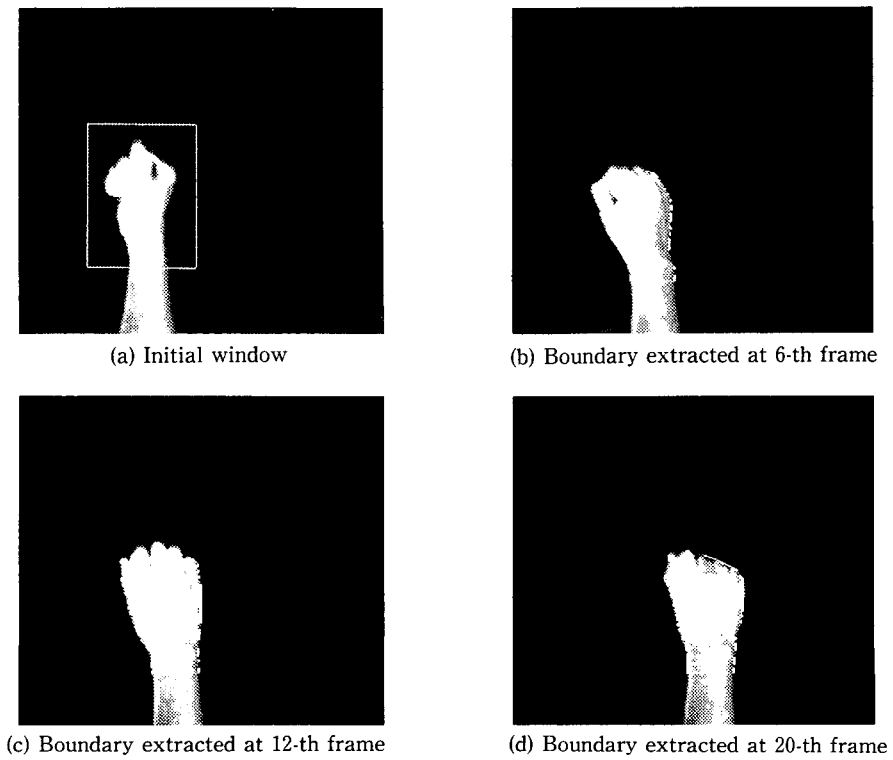
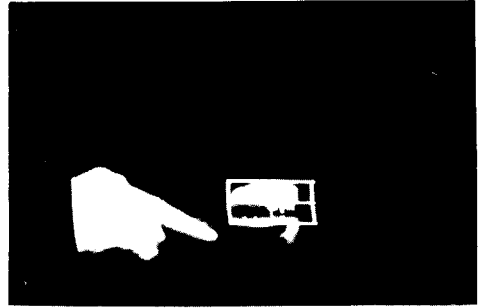


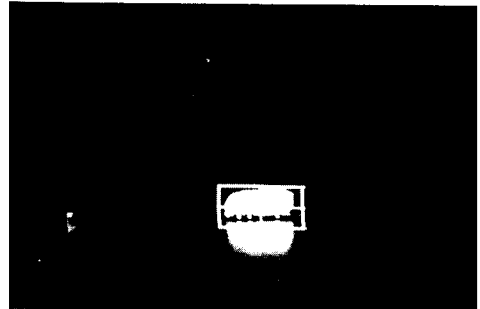
Fig. 7 Visual tracking result for nonrigid object

심이 추출됨을 알 수 있다. Fig. 9의 결과는 면적 구속 에너지 항을 사용하지 않았을 경우의 결과로써 능동 보는 폐색에 의해 표적체의 경계에 영향을 받는다. 따라서 폐색 발생 순간에는 표적체의 경계를 정확히 추출하지 못하게 되며 따라서 이때 구해진 중심좌표는(a)에서처럼 실제 중심에서 벗어난 결과를 보인다. 그러나 폐색이 사라지면 (b)과 같이 표적체의 중심좌표가 추출됨을 알 수 있다.

표적체에 대해 어느 정도의 사전정보가 주어질 경우 능동 보는 보다 강인한 결과를 보일 수 있다. Fig. 10의 결과는 표적대상 선정단계에서 외부 에너지를 사용하여 초기 표적체의 경계를 정확히 추출하고 이 때 구해진 표적체의 면적 및 경계를 사전 정보로 사용함으로써 능동 보는의 면적 및 경계를 구속하여 얻은 결과이다. 이 경우 면적 구속 에너지를 사용하였으므로 Figs. 7, 8, 9의 결과보다 계산 시간은 증가하였으나 복잡한 배경하에서도 표적체 추출이 잘 이루어졌음을 알 수 있다.

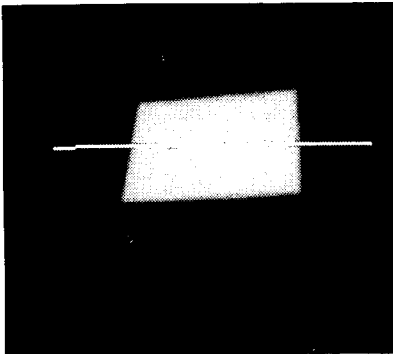


(a) Result for the obscured target

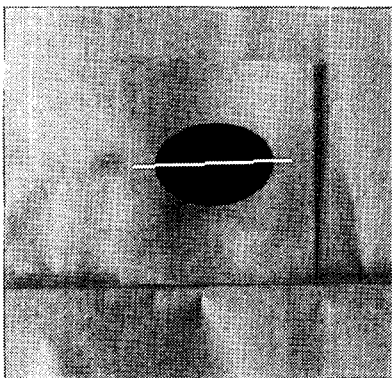


(b) Result after removing the obscured objects

Fig. 9 Result for the partially obscured target

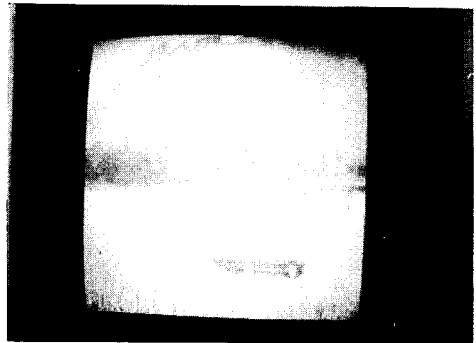


(a) Principal axis of the square object

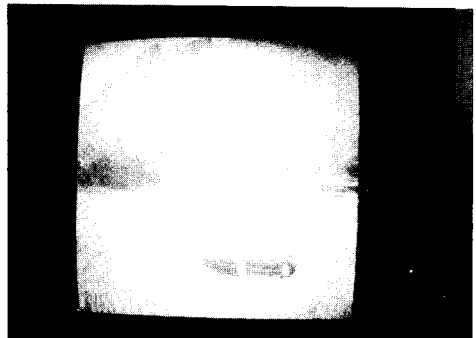


(b) Principal axis of the elliptic object

Fig. 8 Extracting of the principal axis



(a) Boundary trace of the moving target



(b) Centriod of the moving target

Fig. 10 Result using the constraint energy

Table 1 Comparison of processing times

	No. of active bars	Used energy	Computation speed
Fig. 6	20	Image and boundary constraint energy	10 frame/sec
Fig. 8	20	Image and boundary constraint energy	10 frame/sec
Fig. 9	30	Image and boundary constraint energy	7 frame/sec
Fig. 10	12	Image, boundary and area constraint energy	8 frame/sec

Table 1은 각 실험에 대한 능동 보의 수 및 계산속도를 정리한 것이다.

5. 결 론

본 논문에서는 능동 보모델의 에너지함수를 제공하였으며 이를 이용하여 실시간의 영상추적 결과를 제공하였다. 표적체를 능동 보로 모델링함에 따라 표적체의 수축, 팽창 및 회전에 따른 형태 변형에 무관하게 경계를 추출할 수 있었으며, 단순화된 표적체 모델을 통해 표적체의 면적 및 중심방향을 쉽게 구할 수 있다. 또한 최적 에너지 탐색방향을 1차원으로 제한함으로써 간단한 최적화방법으로도 최적해를 구할 수 있었으며 최적해 탐색공간의 축소로 인하여 표적체 인식에 요구되는 계산 시간을 줄임으로써 실시간으로 표적인식 및 표적체의 이동 변수 추출을 가능하도록 하였다. 한편 표적체의 사전정보를 구축에너지화함으로써 폐색 및 배경효과를 줄일 수 있음을 결과를 통해 제시하였다.

참고문헌

- (1) Kass, M. Witkin, A. and Terzopoulos, D. 1988, "Snakes: Active Contour Models," *International Journal of Computer Vision*, pp. 321~331.
- (2) Terzopoulos, D., Andrew Witkin and Kass, M. 1987, "Symmetry-Seeking Models for 3D Object Reconstruction," *International Journal of Computer Vision*, 1, pp. 211~221.
- (3) Andrew Witkin, Terzopoulos, D. and Michael Kass, 1987, "Signal Matching Through Scale Space," *International Journal of Computer Vision*, pp. 133~144.
- (4) Terzopoulos, D., Andrew Witkin and Kass, M. 1988, "Constraints on Deformable Models: Recovering 3D Shape and Nonrigid Motion," *Artificial Inteligencr* 36, pp. 91~123.
- (5) Laurent D. Cohen, 1991 "Note: On Active Contour Model and Ballons," *CVIGP*, Vol. 53, No. 2, pp. 211~218, March.
- (6) James M. Rehg and Andrew P. Witkin, 1991, "Visual Tracking with Deformation Models," *Proceedings of the 1991 IEEE International Conference on Robotics and Automation Sacramento, California*, April.
- (7) Donna J. Williams and Mubarak Shah, "Edge Contours Using Multiple Scales," *CVGIP*, Vol. 51, pp. 256~274.
- (8) Donna J. Williams and Mubarak Shan, 1992, "A Fast Algorithm for Active Contours and Curvature Estimation," *CVGIP*, Vol. 55, No. 1, pp. 14~26, January.
- (9) Amir A. Amini, Saeid Tehrani and Terry E. Weymouth, 1988, "Using Dynamic Programming for Minimizing the Energy of Active Contours in the Presence of Hard Constraints," *in Proc. Second Int. Conf. Computer Vision*, Tarpon Springs, FL, Dec.
- (10) Amir A. Amini, Terry E. Weymouth, and Ramesh C. Jain, 1990, "Using Dynamic Programming for Solving Variational Problems in Vision," *IEEE PAMI*, Vol. 12, No. 9, September.
- (11) Frederic Leymarie and Martin D. Levine, 1993, "Tracking Deformable Objects in the Plane Using an Active Contour Model," *IEEE PAMI*, Vol. 15, No. 6, June, pp. 617~634.

- (12) Delingette, H. Herbert, M. and Ikeuchi, K. 1992, "Shape Representation and Image Segmentation using Deformable Surfaces," *Image and Vision Computing*.
- (13) John Canny, 1986, "A Computer Approach to Edge Detection," *IEEE PAMI*, Vol. 8, No. 6, November.
- (14) Aggarwal J. K. and Nandhakumar, N. 1988, "On the Computation of Motion from Sequences of Images-A Review," *Proceedings of the IEEE*, Vol. 76, No. 8, August.
- (15) John(Yiannis) Aloimonos and Dimitris Tasikiris, P. 1991, "On the visual mathematics of tracking," *Image and Vision Computing*, Vol. 9, No. 4, August.
- (16) Herbert Freeman, 1974, "Computer Processing of Line-Drawing Image," *ICM Computing Surveys*, Vol. 6, No. 1, March.
- (17) Bajon, J. Cattoen, M. and Kim, S. D. 1985, "A Concavity Characterization method for digital objects," *Signal Processing* 9, pp. 151~161.
- (18) Berthold Klaus Paul Horn, 1987, *Robot Vision*, MIT Press, pp. 48~53, 1987.
- (19) William H. Press, Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling, 1988, *Numerical Recipes in C*, Cambridge University Press, pp. 299~302.