

컴퓨터 비전 시스템을 위한 추적 알고리즘의 개발

鄭 聖 煥* · 裴 廷 孝 · 金 南 哲

慶北大學校 工科大学 電子工學科

(1987. 6. 29 접수)

Development of Tracking Algorithm for Computer Vision System

Sung-Hwan Jung, Jeoung-Hyo Bae, Nam-Chul Kim

(Received Jun 29, 1987)

Abstract In this paper, a target tracking algorithm of the centroid extraction from moving edges is proposed for computer vision system. It aims to avoid the difficulty of image segmentation in case of the centroid extraction from one frame. The performance of the proposed algorithm for noisy and occluded images is discussed. Finally, it is also applied to a computer vision system.

I. 서 론

최근 영상 처리 기술과 반도체 소자의 급속한 발달로 인하여 입력된 영상으로부터 물체를 인식하고 해석하는 컴퓨터 비전 시스템을 설계 구현함에 있어 많은 진전을 보게 되었다. 이러한 컴퓨터 비전은 공정의 자동화, 인공지능(AI)뿐만 아니라 자동목표 추적, 형태인식, 교통량 제어등 그 응용 범위가 광범위하여 현재 국내에서도 활발히 연구되고 있다.^{1,2)}

본 논문에서는 컴퓨터 비전의 여러 기능중에서 움직이는 물체를 추적하는 기능에 대한 연구 개발을 위하여 새로운 추적 알고리즘을 제안하였다. 제안된 방법은 컴퓨터 비전 시스템의 카메라로부터 입력된 시변영상에서, 이동하는 물체의 이동 경계를 추출한후 이 이동경계의 무게 중심을 추적하는 알고리즘이다.³⁾ 여기서 추출한 이동경계의 정보는 컴퓨터 비전의 여러 분야에서 형태인식에도 이용될 수 있다. 제안된 방법은 1 프레임만을 이용하여 추적하는 기존의 무게 중심 추적 알고리즘에 비해 비교적 간단하고 쉽게 영상을 분리할 수 있으며 빠른 이동과 다양한 이동에도 추적이 가능하다. 본 연구에서는 물체와 배경사이에 어느정도 밝기차이가 있으며 단일 물체가 이동한다

고 가정하였다. 끝으로 제안된 알고리즘의 타당성을 확인하기위해 FORTRAN 언어와 NOVA 어셈블리어로 실제 컴퓨터 비전 시스템에 적용하였다.

II. 무게중심 추출법

카메라로부터 입력된 관측 영상 $g(x,y)$ 에서 물체가 포물면 분포를 가지고 가산된 잡음이 Gauss 분포를 가질때 물체 및 배경의 조건부 확률에 대한 유사성비(likelihood ratio)를 최대로 하는 결합 위치는 다음과 같은 물체의 밝기중심이 된다.⁴⁾

$$x_0 = \frac{\sum_x \sum_y x g(x,y)}{\sum_x \sum_y g(x,y)}$$
$$y_0 = \frac{\sum_x \sum_y y g(x,y)}{\sum_x \sum_y g(x,y)} \quad (1)$$

여기서 (x_0, y_0) 는 유사성비를 최대로 하는 화소의 위치를 나타낸다. 이렇게 구한 중심을 이용하여 추적할 경우 매 화소 마다 곱하기 연산을 해야하므로 오랜 계산시간이 요구되며 하드웨어를 구현할때 회로가 복잡하게 된다. 따라서 이를 간단히 하기위해 관측영상 $g(x,y)$ 를 먼저 배경과 물체로

영상 분할한 후, 이러한 이진영상 $f(x, y)$ 에서 중심을 구하여 이를 이용하여 추적한다. 이렇게 하면 실제 밝기의 중심이 아닌 물체 2차원 면적의 중심을 추적하게 되지만 그 중심이 크게 변하지 않으므로 물체를 추적하는데는 크게 영향을 미치지 못한다. 이러한 이진 영상에서의 무게중심 추출식은 다음과 같다.

$$\begin{aligned} x_0 &= \frac{\sum_x \sum_y x f(x, y)}{\sum_x \sum_y f(x, y)} \\ y_0 &= \frac{\sum_x \sum_y y f(x, y)}{\sum_x \sum_y f(x, y)} \end{aligned} \quad (2)$$

여기서 $f(x, y)$ 는 물체와 배경으로 분리된 이진영상이다. (2)식에서 볼때 X 축의 무게중심은 이진영상 $f(x, y)$ 가 "1"일때만 x 위치를 총합하여 물체 pixel의 총합으로 나누면 되며, 같은 방법으로 y 축의 중심도 구할 수 있다. 이렇게 하면 더하기 연산만으로 무게중심을 추출할 수 있어 빠르게 처리할 수 있으며 하드웨어 구현이 간단하게 된다.

또한 이진영상 $f(x, y)$ 로 부터 수평, 수직 방향으로 투영된 투영함수를 사용하면 물체의 여러가지 정보를 얻을 수 있다. 그림 1에서와 같이 $f(x, y)$ 를 x 축 및 y 축에 각각 투영한 투영함수 (projection function)는 다음과 같은 수식으로 나타낼 수 있다.^{5,6)}

$$\begin{aligned} P_x(y) &= \sum_x f(x, y) \\ P_y(x) &= \sum_y f(x, y) \end{aligned} \quad (3)$$

여기서 $P_x(y)$ 및 $P_y(x)$ 는 각각 y 및 x축에 $f(x, y)$ 를 투영한 투영함수들이다. 이렇게 투영된 함수를 사용하여서 다음과 같이 물체의 무게중심을 구할 수도 있다.

$$\begin{aligned} d_{ox} &= \frac{1}{N} \sum_x x P_y(x) \\ d_{oy} &= \frac{1}{N} \sum_y y P_x(y) \end{aligned} \quad (4)$$

여기서 (d_{ox} , d_{oy})는 무게중심의 좌표이며 N은 그들 갯수의 총합이다. 이러한 투영함수를 사용하면 물체의 위치뿐만 아니라 방향, 구조등도 알 수 있다.

앞에서 구한 무게중심을 이용하여 물체를 추적하는 방법에 대해 살펴보면, 기존의 무게중심 추적법은 시변영상의 1 프레임만을 사용하기 때문에 배경과 물체를 분리해내기 위한 영상 분할을 하는데 많은 어려움이 따른다. 따라서 정확히 분할하기 위해서는 비교적 복잡한 분할 알고리즘이 필요하게 된다. 실례를 들면, 그림 2와 같이 여러 가지 창을 사용하여 분할하는 예가 있다.⁷⁾ 여기서는 영상을 분할하는데 background window, target window, edge window등을 사용하여 각각의 창 내의 pixel 들을 해석하여 비교적 복잡하게 영상을 처리하고 있다. 또한 이 경우에 복잡한 배경과 시간이 변함에 따라 물체와 배경사이에 밝기차가 변할 때에는 영상분할이 용이하지 못하며, 특히 background window 나 edge window내에 물체와 같은 밝기의 grey level이 존재할 때는 영상분할 하는데 매우 어려워진다. 따라서 이런 경우에는 무게중심 추출이 대단히 어렵게 된다.

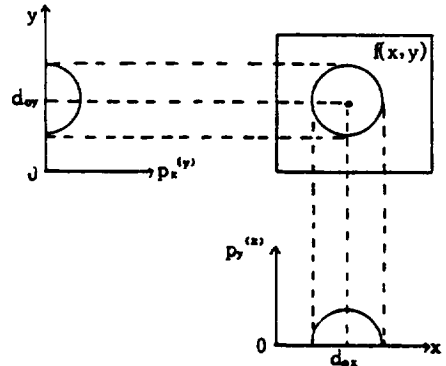


그림 1. 이진 영상을 투영한 그림

Fig.1. Projected figure of binary image

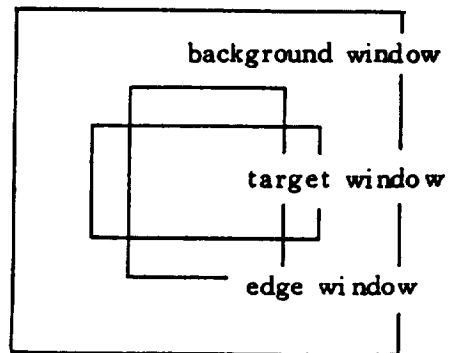


그림 2. 추적 창 영역

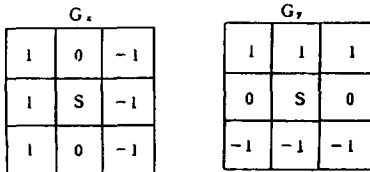
Fig.2. Tracking window region

Ⅲ. 이동경계의 무게중심 추출

제안된 방법에서는 기존의 방법에서 나타나는 영상분할의 어려움을 피하기 위해 연속되는 영상 두 프레임에서 시공간경사^{8~12)}를 이용하여 이동 경계를 추출, 해석함으로써 간단한 알고리즘으로 영상분할을 할 수 있도록 하였다. 여기서 이동 경계라 함은 시변영상에서 물체가 이동할 때 두 프레임으로부터 시간경사 및 공간경사를 구한후 각 역치와 비교할 때 값이 큰 영역을 말하며 다음과 같이 표현된다.^{13,14)}

$$\text{이동경계} = |s| \cdot |T(s)| > T_{th} \text{ and } |G(s)| > G_{th} \quad (5)$$

여기서 시간경사 T(s)는 위치벡터 s에서의 현재 프레임의 밝기 I(s, m)와 과거 프레임의 밝기 I(s, m-1)차를 말하며 공간경사 G(s)는 그림 3에서와 같이 x 방향의 공간경사와 y 방향의 공간경사를 구한후 각각의 절대치를 취하여 더한 값이다. 그리고 T_{th} 와 G_{th}는 각각 시간경사 및 공간경사의 역치들이다. 여기서 사용한 경계 연산자는 곱하기 연산이 필요없는 그림 3과 같은 Prewitt 연산자를 사용했다. 이렇게 시간경사를 이용함으로써 배경영향이 배제될 뿐만 아니라, 공간경사를 적용시킴으로 물체의 이동경계만을 간단히 추출할 수 있어 영상 분할을 쉽게 할 수 있게 된다. 또한 시간경사의 역치보다 큰 부분에서만 공간 경사를



G_x (spatial gradient of x direction)

$$= \{I(i-1, j-1) + I(i-1, j) + I(i-1, j+1) - [I(i+1, j-1) + I(i+1, j) + I(i+1, j+1)]\}$$

G_y (spatial gradient of y direction)

$$= \{I(i, j-1) + I(i, j) + I(i, j+1) - [I(i-1, j+1) + I(i, j+1) + I(i+1, j+1)]\}$$

|G| = |G_x| + |G_y|

그림 3. Prewitt 연산자
Fig.3. Prewitt operator

구하므로 처리속도가 빠르다. 그리고 이동 경계의 무게중심을 추적함으로써 물체의 병진운동(translation)뿐만 아니라 회전운동(rotation), 접근(approach) 그리고 후퇴(recession) 등의 다양한 운동에도 이동경계의 중심은 크게 변하지 않으므로 안정된 추적이 가능해진다. 제안된 방법의 전체적인 흐름도는 그림 4와 같다. 흐름도에서 x 및 y는 이동경계일 때의 각각 위치를 나타내며, count는 안정된 추적이 이루어지도록 물체가 움직였을 때만 window update하기 위한 것이다.

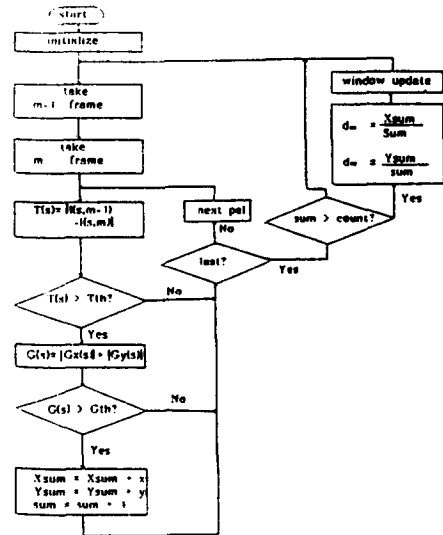


그림 4. 흐름도
Fig.4. Flowchart

Ⅳ. 실험 및 검토

본 알고리즘의 성능평가를 위하여 컴퓨터 시뮬레이션을 행한후 본 연구실에서 제작한 자동 목표 추적장치에 실제로 적용하였다.¹⁵⁾

(1) 컴퓨터 시뮬레이션

객관적인 성능평가를 위해, 먼저 두장의 영상에서 한장을 기준 프레임으로 하고 다른 한장을 임의의 변위를 주어 현재 프레임으로 하여 이동변위를 추출하는 추정성능과 여러장의 영상으로 매 프레임마다 임의의 변위를 주어 그때마다 추정된 변위로 움직이는 물체를 추적하는 추적성능을 조사하였다.

먼저 추정성능을 구하는데 사용된 영상은 120

×120 화소의 CAR와 SPHERE 영상으로 그림 5에 나타내었다. 여기서 CAR 영상은 장난감 자동차가 움직이는 장면을 직접 디지털화한 영상에서 자동차 부분만을 분리하여, $\text{mean}=80$, $\rho_x=0.95$, $\sigma_x=4$ 를 갖는 2-D Markov 신호의 배경에 모자이크한 영상이다. 또한 SPHERE 영상은 합성영상으로 다음과 같은 밝기함수를 갖는 영상이다.

$$I(s, m) = \text{con} \sqrt{R^2 - C^2 + 140}$$

여기서

$$\text{con} = \text{contrast}, R = 30$$

$$C^2 = (i-X)^2 + (j-Y)^2, s = (i, j)$$

$$X = \alpha + 60, Y = \beta + 60$$

α, β : 현재 및 과거 프레임 사이의 실제 이동변위

즉 CAR 영상과 같은 배경에 $\text{contrast}=3$, $R=30$ 인 합성 영상 구를 모자이크한 것이다. 추정 성능은 물체의 이동변위를 비교적 큰 변위¹⁶⁾인 $\pm 5 \sim \pm 10$ 으로 144번 움직여 조사하였으며, 이때 CAR(SPHERE) 영상의 시간경사 및 공간경사의 역치는 잡음이 없을 경우 각각 5, 20(10, 40)이며 잡음이 섞일 경우 각각 30, 80(25, 80)으로 하였다. 성능평가의 척도로는 실제 변위와 추정변위의 절대차의 평균으로 나타나는 MAE(mean absolute error)로 나타내었다.

$$\text{MAE} = \frac{1}{2N} \sum_{i=1}^N (|\hat{\alpha}_i - \alpha_i| + |\hat{\beta}_i - \beta_i|)$$

여기서 (α_i, β_i) 및 $(\hat{\alpha}_i, \hat{\beta}_i)$ 은 각각 실제변위 및 추정변위를 나타낸다. SNR 따른 추정성능을 조사하면 표1과 같다. 여기서 SNR은 최대 신호 전력비 대 잡음 전력비로 나타낸다. 표 1에서 볼때 잡음이 섞일 경우 추정성능이 저하되기는 하나 거의 1 pel을 벗어나지는 않는다.

그리고 추적성능을 구하는데 사용된 영상은 추적에서 사용한 배경을 120×300 으로 확장하여 만든 배경에 CAR을 연속적으로 움직이면서 모자이크하여 얻은 영상이다. 그림 6은 잡음이 없을때 x 방향과 y 방향의 속도에 따른 추적성능을 나타낸 그림이다. 여기서 x 방향의 운동 방정식은 mean

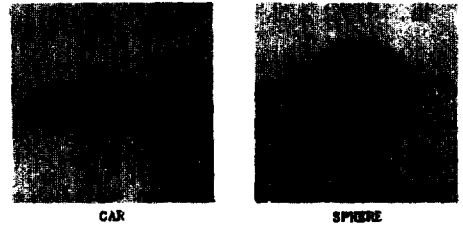


그림 5. 시뮬레이션에 사용된 영상

Fig.5. The image used in simulation

표 1. 잡음(SNR)에 따른 추정성능

Table 1. Estimation performance versus noise(SNR)

noise	CAR	SPHERE
noise free	0.038	0.028
36 dB	0.434	0.514
30 dB	0.490	0.545

$=5$, $\text{variation}=3$ 으로 움직이는 정현파 함수이며 y 방향은 $\text{mean}=0$, $\text{variation}=5$ 인 정현파 운동 방정식을 갖는다. 여기서 볼때 이 운동 방정식의 이동변위는 비교적 큰 변위를 움직이며 이에 따른 추적 성능은 그림 6에 나타난 것과 같이 순간속도가 과소 추정되더라도 다음 순간에 과대 추정함으로써 보상되어 잘 추적함을 보았다. 잡음에 따른 추적성능과 occlusion이 발생한 영상에서의 추적성능을 x와 y의 2차원 위치로써, 그림 7(a), (b)와 그림 8에 각각 도시하였다. SNR=30 dB의 잡음이 심하게 섞인 경우도 큰 오차없이 추적함을 볼 수 있으며, occlusion이 발생한 경우도 가려진 부분이 나타났을때 보상추적하여 물체를 놓치지 않고 잘 추적함을 볼 수 있다. 그림 9는 bar 형태의 전경을 occlusion 시켰을 때의 영상이며 전경은 $\text{mean}=175$, $\rho_x=0.95$, $\sigma_x=4$ 를 갖는 2-D Markov 신호이며 크기는 높이 100pel, 폭 10pel이다.

(2) 추적장치에의 적용

제안된 알고리즘을 그림 10과 같은 실제 컴퓨터 시스템의 추적장치에 적용하였으며 추적 장치의 제원은 표 2와 같다. 고속의 16bit 마이크로 프로

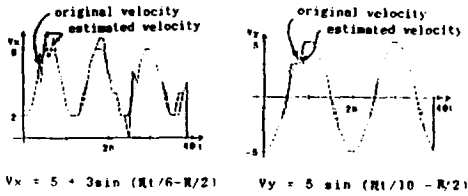


그림 6. X와 Y방향의 속도(잡음이 없을때)
Fig.6. Velocity of X and Y direction(noise free)

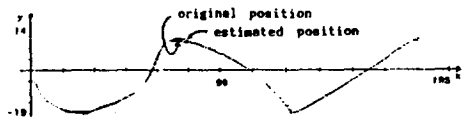


그림 7(a). X와 Y의 2차원 위치(잡음이 없을때)
Fig.7(a). Two dimensional position of X and Y

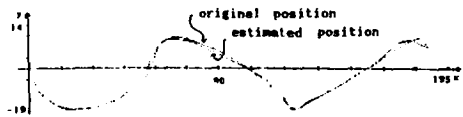


그림 7(b). X와 Y의 2차원 위치(30 dB 일때)
Fig.7(b). Two dimensional position of X and Y direction (SNR=30 dB)

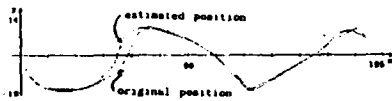


그림 8. X와 Y의 2차원 위치(30 dB, occlusion)
Fig.8. Two dimensional position of X and Y direction (SNR=30 dB, occlusion)

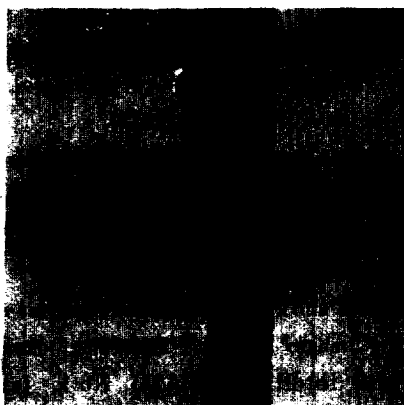


그림 9. occlusion 된 영상
Fig.9. Occluded image



그림 10. 컴퓨터 비전 시스템
Fig.10. Computer vision system

표 2. 자동 목표 추적 장치의 자원
Table 2. Specifications of the automatic target tracker

System spec.	CPU	F9445 16 bit microprocessor
	Instruction execution time	375 nsec
	Clock frequency	16 Mhz
	System memory	32 Kwords
Language	Frame memory	64 Kbytes
		NOVA assembly
Digitization	Speed	30 frames/sec
	Bits per pixel	8 bits
	Size	64 x 64
	Sampling rate	6 Mhz

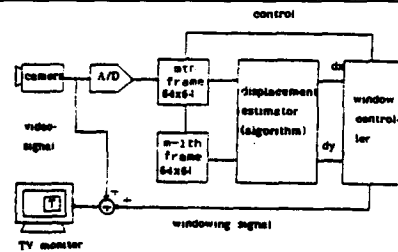


그림 11. 자동 추적 장치의 구조
Fig.11. Structure of the automatic tracker

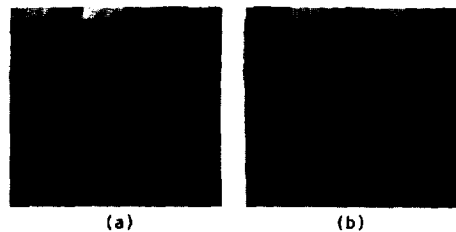


그림 12. 추적 동작 : (a)→(b)
Fig.12. Tracking operation:(a)→(b)

세서를 중앙제어기로 사용하는 실제 추적 장치의 동작원리는 그림 11과 같다. 먼저 카메라로부터 8bit로 64×64 크기의 영상을 디지털화하여 메모리 버퍼에 두 프레임을 저장하여 제안된 알고리즘을 적용시켜 이동변위를 추출하게 된다. window controller는 이 추출된 변위를 이용하여 물체 움직임에 따라 window를 update하여 추적하며 이렇게 update한후 다음의 영상을 받아 들여 이전의

동작을 반복하게 된다. 이 장치에 사용된 언어는 NOVA assembly language¹⁷⁾이다.

실제로 적용한 결과 제안된 알고리즘은 약 6frames/sec을 처리함을 보았으며 contrast가 적게 변하거나, 비교적 복잡한 배경과 물체를 다양하게 이동시켰을 때도 안정된 추적 성능을 확인할 수 있었다. 그러나 contrast가 변할 경우에는 그때마다 시간경사와 공간경사의 역치를 조정해 주어야 한다는 문제점이 있다. 그림 12는 실제 추적할 때 화면에 나타난 장면을 나타낸다. 또한 추적 속도를 자동목표 추적 장치에서의 초당 계산 속도와 최대 추정변위, 화면에서의 추적속도, 카메라로부터 1km 떨어진 위치에서의 실제 속도에 따라 비교하면 표 3과 같다. 여기서 제안된 방법의 최대 추정 변위는 다른 알고리즘에 비해 제한이 적다. 즉 물체가 window 안에만 있으면 큰 변위도 추정이 가능하다. 그러나 컴퓨터 시뮬레이션에서 최대 10pef까지만 실험하였으므로 최대 추정 변위는 10pef이라 가정하였다.

표 3. 자동 목표 추적장치에 적용한 방법들의 비교
Table 3. Comparison of algorithms applied to the automatic target tracker

		the proposed	a fast matching (hierarchical)	spatio-temporal gradient
processing speed (frames/sec)		6	6	7
max. estimated displacement (pels/frame)		10	7	4
max. estimated velocity	(pels/sec)	60	42	28
	at a distance of 1Km (Km/h)	313	219	146

IV. 결 론

본 논문에서는 컴퓨터 비전의 한 중요기능인 추적에 대한 연구개발을 위해 새로운 추적 알고리즘을 제안하였다. 제안된 방법은 기존의 무게중심 추적방법에서 나타나는 영상분할의 어려움을 피하기 위해, 두 프레임에서 시공간 경사를 이용하여 이동경계를 추출한후 이 이동경계의 무게중심을 추적하는 알고리즘이다. 이때 시공간 경사를 사용함으로써 배경의 영향을 배제하였으며, 기존의 방법에 비해 비교적 간단한 알고리즘으로 영상 분할을 쉽게 하였다. 또한 전체 알고리즘이 더하

기 연산만으로 이루어졌기 때문에 빠르게 처리할 수 있으며 하드웨어를 구현할 때 간단히 할 수 있다. 그리고 제안된 방법은 무게중심을 이용함으로써 물체의 다양한 운동에도 잘 추적할 수 있다. 제안된 알고리즘을 실제 컴퓨터 비전 시스템에 적용하여 그 타당성을 확인하였다.

참 고 문 헌

1. B.K.P.Horn, Robot Vision, The MIT Press, (1986).
2. D.H.Ballard and C.M.Brown, Computer Vision, Prentice-Hall Inc., (1982).
3. T.Pavlidis, Algorithm for Graphics and Image Processing, Computer Science Press., (1982).
4. W.L.Wolfe and G.J.Zissis, The Infrared Handbook, Office of Naval Research, Department of navy, Washington, DC, pp.22-63-22-85, (1978).
5. A.L.Gilbert and et al, "A Real-time Video Tracking System," IEEE Trans.on Pat. Anal. Machine Intel., PAMI-2, pp.47-56, (1980).
6. A.L.Gilbert, "Video Data Conversion and Real - Time Tracking," IEEE Computer, pp.50-56, Aug. (1981).
7. Serk,M.S., A Study on the Automatic Target Tracking System, Final Report, KAIST, (1983).
8. W.N.Martin and J.K. Aggarwal, "Dynamic scene Analysis: A Survey," CGIP.Vol.7, No.3, pp.356-374, June, (1978).
9. W.B.Thompson, "Combining Motion and Contrast for Segmentation," IEEE Trns.Pat. Anal.Machine Intell., Vol. PAMI-2, pp.543-549, Nov. (1980).
10. M.Yachid and et al, "Automatic Analysis of Moving Image," IEEE Trans. Pat. Anal. Machine Intell., Vol.PAMI, No.1, Jan. (1981).
11. Jang,S.H.,A Displacement Vector Estima-

- tion and Moving Object Extraction Using Region and Boundary of Difference Picture, M.S. Thesis, KAIST, (1986).
12. Lee, J.B., A Study on the Segmentation of Time-Varying Imagery Using Motion Information, M.S.Thesis, KAIST, (1986).
 13. Koh.B.S., A Study on the Segmentation of Moving Objects by Use of Moving edges, M.S.Thesis, KAIST., (1985).
 14. Chon, I.S., Real-Time Automatic Target Tracking Using a Subtempate of Moving Region, M.S.Thesis, Kyungpook Nat.Univ.,Dec. (1986).
 15. Kong, T.H., Hardware Implementation of a Multi-Function Image Processing System, M.S. Thesis, Kyungpook Nat. Univ. June, (1986).
 16. R.Jain, W.N.Martin and J.K.Aggarwal, mentation through the Detection of Changes Due to Motion," Computer Graphics and Image processing 11, pp.13-14, (1979).
 17. Data General, How to Use the NOVA Computers, (1974).