

# Particle filter를 이용한 이동 물체 추적 알고리즘

## Mobile Object Tracking Algorithm Using Particle Filter

김세진 · 주영훈\*

Se Jin Kim and Young Hoon Joo\*

군산대학교 제어로봇시스템공학과

### 요 약

본 논문에서는 Particle filter를 이용한 특징 벡터 기반 이동 물체 추적 알고리즘을 제안한다. 이를 위해, 첫 번째, RGB 칼라 모델을 이용하여 초기 이동 물체의 움직임 영역(blob)을 추출하고, KLT-알고리즘을 이용하여 입력 영상에 대한 특징 벡터를 구한다. 그 다음, 초기 추출된 이동 물체의 움직임 영역에 이 특징 벡터를 매칭시켜 1차 특징 벡터를 구한다. 두 번째로, RGB와 HSI 칼라모델을 이용하여 이동 물체의 움직임 영역을 추출하고, 앞서 구한 1차 특징 벡터에 Snake 알고리즘을 적용함으로써 새로운 특징 벡터를 구한다. 그 다음, 2차 추출된 이동 물체의 움직임 영역에 이 새롭게 구한 특징 벡터를 매칭시켜 2차 특징 벡터를 구한다. 최종적으로, 2차 특징 벡터에 Particle filter를 적용함으로써 본 논문에서 제안한 이동 물체를 추적하는 알고리즘을 완성한다. 마지막으로, 본 논문에서 제안한 알고리즘은 복잡하고 다양한 환경에서 실험을 통해 그 응용 가능성을 증명한다.

**키워드** : 물체 추적, 물체 인식, 특징 벡터, 탐색 모델, Snake algorithm, Particle filter

### Abstract

In this paper, we propose the mobile object tracking algorithm based on the feature vector using particle filter. To do this, first, we detect the movement area of mobile object by using RGB color model and extract the feature vectors of the input image by using the KLT-algorithm. And then, we get the first feature vectors by matching extracted feature vectors to the detected movement area. Second, we detect new movement area of the mobile objects by using RGB and HSI color model, and get the new feature vectors by applying the new feature vectors to the snake algorithm. And then, we find the second feature vectors by applying the second feature vectors to new movement area. So, we design the mobile object tracking algorithm by applying the second feature vectors to particle filter. Finally, we validate the applicability of the proposed method through the experience in a complex environment.

**Key Words** : Object tracking, Object recognition, Detection model, Snake algorithm, Particle filter

## 1. 서 론

현재 가장 각광을 받고 있는 인간-컴퓨터 상호작용(HCI, Human Computer Interaction), 인간-로봇 상호작용(HRI, Human Robot Interaction)기술은 인간의 생각이나 감정 그리고 의사 표현을 컴퓨터나 로봇과 공유하고 상호 작용하기 위해 이용하고 있는 기술이다. 이와 같은 기술 중에서 카메라에 입력되는 영상을 처리하여 인간으로부터 표출되는 의사 및 행위를 인식하고 상호 작용하며 정보를 공유 할 수 있는 기술을 바로 비전 기술이라고 한다. 현재 비전 기술은 인간과 컴퓨터/로봇간의 상호 작용 뿐만 아니라 인간의 편의를 도모하기 위한 자동화 로봇 시스템과 생물학, 문서 처리, 공장 자동화, 의료 진단 영상 시스템, 비디오/필름(서비스업 분야)등에 다양하게 이용되고 있으며, 특히 군사적 보안(서비스 보완)분야에서 연구가 매우 활발하게 진행

되고 있다. 게다가 몇몇 분야에서는 실생활에 직접 이용되고 있기도 하다[1-2]. 그리고, 현재 보안 시스템에서의 비전 기술 이용 및 적용도가 매년 증가하고 있으며, 점점 세분화 되어 가고 있는 실정이다. 일반적으로, 보안 시스템에서 비전 기술은 입력 영상에 진입한 물체에 대한 정보를 추출하여 움직임 패턴을 분석/인식하여 이동 물체를 추적하거나 알람 및 메시지를 전송하는데 이용 하고 있다. 이와 같은 보안 시스템에서 가장 많이 이용되고 있는 비전 기술은 바로 물체 추적 및 감시 기술이다. 물체 추적 및 감시 기술의 기존 연구 내용을 보면, 칼만 필터 기반의 깊이 평가를 이용한 방법과 최소 제곱 평가를 칼만 필터에 적용한 방법 [3-4], 적응적 칼라 모델을 이용한 물체 추적 방법과 칼라 히스토그램을 이용한 빛의 변화에 대한 물체 인식 등의 많은 연구가 진행되어 왔다[5-6]. 하지만 이와 같은 방법론들도 해결 하지 못한 몇 가지 문제점들이 있다. 상기 방법론들은 기본적으로 칼라 모델을 기반으로 빛의 변화에 강인하지만, 이동 물체에 대한 특징 벡터 또는 특징 영역을 부분적으로 고려하지 않아서 물체 추적 시 이용할 수 있는 정보량이 제한적이다. 그리고 깊이 정보와 칼라 정보의 최소 제곱 평가 방법을 필터에 직접 적용하여 물체 추적에는 성공하였지만 다중 이동 물체 추적에서 겹침 현상이 발생하는

접수일자 : 2009년 6월 1일

완료일자 : 2009년 8월 2일

+ 책임저자

본 논문은 지식경제부의 지원으로 수행한 에너지자원인력 양성사업의 연구결과입니다.

경우에 약한 면을 보인다[7-8]. 또한, 이동 물체의 불투명성과 배경의 단조로움 때문에 복잡한 환경에서 배경의 작은 변화에 민감하게 반응하는 약점도 가지고 있다.

본 논문에서는 상기 문제를 해결하기 위해 복잡한 환경에서 강인한 이동 물체 추적 알고리즘을 제안한다. 먼저, RGB 칼라 모델을 이용하여 초기 이동 물체의 움직임 영역(blob)을 추출하고[9], KLT-알고리즘을 이용하여 입력 영상에 대한 특징 벡터를 구한다[10-11]. 그 다음, 초기 추출된 이동 물체의 움직임 영역에 이 특징 벡터를 매칭시켜 1차 특징 벡터를 구한다. 두 번째로, RGB와 HSI 칼라모형을 이용하여 이동 물체의 움직임 영역을 추출하고[12], 앞서 구한 1차 특징 벡터에 Snake 알고리즘을 적용함으로써 새로운 특징 벡터를 구한다[13]. 그 다음, 기 추출된 이동 물체의 움직임 영역에 이 새롭게 구한 특징 벡터를 매칭시켜 2차 특징 벡터를 구한다. 최종적으로, 2차 특징 벡터에 Particle filter를 적용시켜 이동 물체를 추적하는 알고리즘을 제안한다[14-15]. 마지막으로, 본 논문에서 제안한 알고리즘은 복잡하고 다양한 환경에서 실험을 통해 그 응용 가능성을 증명한다.

## 2. 물체 추적 시스템

본 논문에서 제안한 시스템은 기존에 제안된 배경 제거 기법을 기본으로 하여 설계 한다. 제안하는 시스템은 그림 1과 같다.

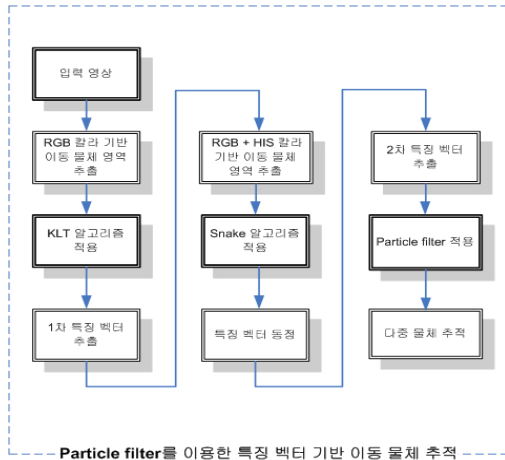


그림 1. 시스템 블록도  
Fig 1. System block diagram

본 논문의 시스템은 크게 세 단계로 구성되어 있다. 첫 번째 단계로 영상에 진입한 물체를 추출한다. 먼저, RGB 칼라 모델을 이용하여 초기 이동 물체의 움직임 영역을 추출하고, KLT-알고리즘을 이용하여 입력 영상에 대한 특징 벡터를 구한다. 여기서, 추출된 특징 벡터들은 입력 영상에서 이동하는 물체의 대한 정보를 완벽히 가지고 있다고 가정 한다. 그 다음, 초기 추출된 이동 물체의 움직임 영역에 이 특징 벡터를 매칭(특징 벡터 추출 방법)시켜 1차 특징 벡터를 구한다. 두 번째 단계로, 1차 특징 벡터들에 대한 보정을 한다. RGB와 HSI 칼라모형을 이용하여 이동 물체의 움직임 영역을 추출하고, 앞서 구한 1차 특징 벡터에 Snake 알고리즘을 적용함으로써 새로운 특징 벡터를 구한다. 그

다음, 기 추출된 이동 물체의 움직임 영역에 이 새롭게 구한 특징 벡터를 매칭(특징 벡터 추출 방법)시켜 2차 특징 벡터를 구한다. 마지막으로, 2차 특징 벡터에 Particle filter를 적용시켜 이동 물체에 대한 개별적인 움직임을 추적한다.

## 3. Particle filter를 이용한 특징 벡터 기반 이동 물체 추적 알고리즘

### 3.1 이동 물체 영역 추출 - 칼라 모델

입력 영상으로부터 물체를 추출하기 위해서 칼라 모델을 이용한다. 본 논문에서는 RGB 칼라 모델과 HSI 칼라 모델을 배경 제거 기법에 적용시켜 초기 전경을 추출한다. 먼저, RGB 칼라 모델을 이용한다. RGB 모델의 목적은 가장 일반적으로 색의 명세를 손쉽게 하기 위함이다. 또한 칼라 모델은 3차원 좌표 시스템의 명세이며 각각의 색을 한 점으로 나타낸다.

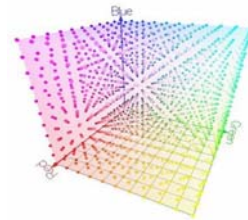


그림 2. RGB 칼라 모델  
Fig 2. RGB color model

일반적으로 RGB 모델의 색채 부분 공간은 그림 2와 같이 입방체로 이루어졌으며 RGB 값은 세 꼭 지점이 된다. 이 모델에서 명암도는 검은색부터 흰색 두 점을 연결하는 선을 따라 전개되고, 색은 원점으로부터 확장되는 벡터로 정의되는 정육면체 상이나 또는 안쪽의 점이다. 모든 R, G, B 값의 범위는 [0, 1]로 가정한다. RGB 색 모델에서의 영상은 각 원색 당 하나씩 모두 3개의 독립적인 영상 평면으로 다음과 같이 구성되어 있다. 그 다음으로 HSI 칼라 모델을 이용한다. HSI 칼라 모델은 색을 인식하는 속성인 색상(Hue), 채도(Saturation), 구성되어 있어 밝기 성분을 별도로 분할하여 사용할 수 있다. HSI 칼라 모델은 아래 그림 3과 같이 상하 대칭인 원뿔 형태로 이루어져 있으며 HSI 값은 원뿔의 꼭지점과 단면의 원주 각 끝점이 된다.

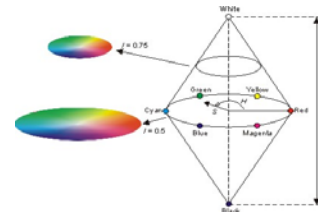


그림 3. HSI 칼라 모델  
Fig 3. HSI color mode

HSI 칼라 모델은 RGB 칼라 모델보다 조명에 덜 민감하기 때문에 RGB 칼라 모델을 HSI칼라 모델로 변환하여 사용하면 조명뿐 만 아니라 빛의 갑작스러운 변화에도 강인함

을 갖는다. HSI 칼라 모델은 칼라 공간을 사용할 때, 어떤 칼라를 만들어 내기 위해서 몇 퍼센트의 파란색이나 녹색이 필요한지를 알 필요가 없다. 진한 빨간색을 분홍색으로 바꾸기 위해 단순히 채도만을 조절하기 때문이다. 어두운 것을 밝게 하려면 명도를 조절 하면 된다. 이러한 이유로 많은 응용 분야에서 HSI 칼라 모델을 사용한다. 그림 3과 같이 이중 원뿔 모델을 사용한다. 색상은 0°에서 360°의 범위를 가진 각도  $\theta$ 로 표현된다. 채도는 0에서 1까지의 범위를 가지는 반지름에 해당하며, 명도는 z축에 해당하는데 0일 때는 검정 색을, 1일 때는 흰색을 나타낸다. S=0일 때, 칼라는 I의 명암도를 갖는다.

$$r = \frac{R}{(R+G+B)}, g = \frac{G}{(R+G+B)}, b = \frac{B}{(R+G+B)}$$

$$i = \frac{1}{3}(r+g+b)$$

$$s = 1 - \frac{3}{(r+g+b)} [\min(r,g,b)]$$

$$h = \cos^{-1} \left[ \frac{\frac{1}{2} [(r-g) + (r-b)]}{\sqrt{(r-g)^2 + (r-b)(g-b)}} \right]$$

$$i \in [0,1], s \in [0,1], h \in [\pi, 2\pi] \text{ for } b > g$$

$$range \rightarrow i = [0,255], s = [0,100], h = [0,360]$$

$$H = h \times 180/\pi, S = s \times 100, I = i \times 255$$

위의 식 (1)과 식(2)는 RGB칼라 공간을 HSI칼라 공간으로 변경할 때 이용되는 수식이다.

### 3.2 이동 물체 영역 설정 - 탐색 모델

먼저, 입력 영상에 진입한 이동 물체 탐색 및 특징 벡터 추출을 위한 가상 물체 탐색 모델을 생성한다. 생성된 모델은 그림 4와 같다. 본 모델은 사람 10명의 신체 구조비를 이용하여 생성한 후, Snake 알고리즘을 이용하여 초기화한다.

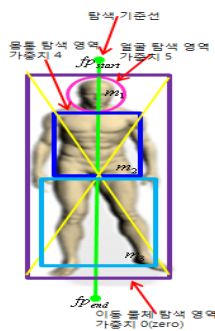


그림 4. 물체 탐색 모델  
Fig 4. Object detection model

생성된 탐색 모델을 초기 추출된 환경에 매칭 시켜 입력 영상에 진입한 물체를 탐색한다. 물체 탐색할 경우 본 모델은 크게 머리, 몸통, 팔-다리 3부분으로 나누어지며, 탐색 공간도 세 부분으로 나누어 탐색 한다. 각 부분에 대한 가중치는 다음과 같이 부여한다. 머리( $m_1$ ) 5, 몸통( $m_2$ ) 4, 팔-다리( $m_3$ ) 3 으로 부여하며 가중치의 총 합은 1이다. 그리

고 본 모델을 이용하여 이동 물체 영역에 특징 벡터를 매칭 시켜 새로운 특징 벡터를 추출한다. 특징 벡터 추출할 경우, 본 모델은 새로운 특징 벡터의  $fp_{start}$ 와  $fp_{end}$  을 제시한다.

칼라모델과 탐색 모델을 이용하여 이동 물체의 영역을 추출하면 그림 5와 같은 결과를 얻을 수 있다.

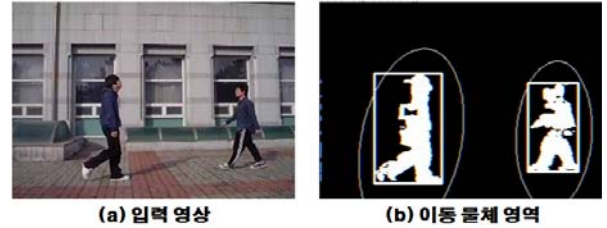


그림 5. 칼라 기반 움직임 물체 추출  
Fig 5. Color-based mobile object extraction

### 3.3 특징 벡터 추출 방법

입력 영상으로부터 추출한 전체 특징 벡터들로부터 이동 물체 영역에 대한 새로운 특징 벡터(7개)를 구한다.

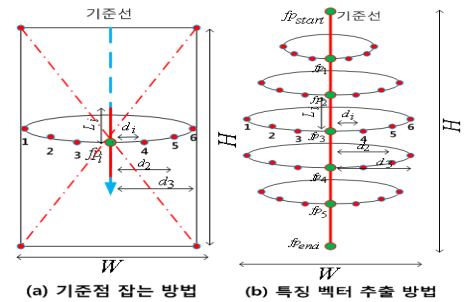


그림 6. 특징 벡터 추출 방법  
Fig 6. feature vector extraction method

먼저, 기준이 되는 특징 벡터를 그림 6(a) 방법을 이용하여 선정과 동시에 초기화 한다. 그리고 선정된 기준점(기준 특징 벡터)으로부터 좌/우측에 추출된 특징 벡터 6개의 개별적인 데이터 값을 이용하여 기준점 값을 갱신한다. 그 후, 그림 6(b)방법을 이용하여  $fp_{start}$  부터 기준점 사이에  $L_i$  만큼씩 간격을 두고 이 방법을 이용하여 2개의 특징 벡터를 추출하고  $fp_{end}$  부터 기준점 사이에  $L_i$  만큼씩 간격을 두고 동일한 방법을 이용하여 나머지 2개의 특징 벡터를 추출한다. 총 7개(이동 물체 특징 벡터(5개) +  $fp_{start}$  (1개) +  $fp_{end}$  (1개))의 특징 벡터를 갖게 되면 작업을 마친다.

### 3.4 1차 특징 벡터 - KLT 알고리즘

입력 영상에 대한 전체적인 특징 벡터를 추출한다. 추출에 이용되는 수식은 다음과 같다.

$t+\tau$  시간 후의 영상은 이전 영상에서 모든 점들의 움직임 위치와 같다.

$$I(x,y,t+\tau) = I(x-\xi(x,y,t,\tau), y-\eta(x,y,t,\tau)) \quad (3)$$

$\delta = (\xi, \eta)$ 는 시간  $t$  에서  $t+\tau$  로 변할 때 점  $X=(x,y)$ 에서의 변위이다. 그리고 생성된 수식은 잡음 때문에 다음 픽셀의 위치를 알아내기 힘들어 유사 변환  $\delta = GX+d$  을 실시한다. 여기서  $G$  는 변형 행렬이고,

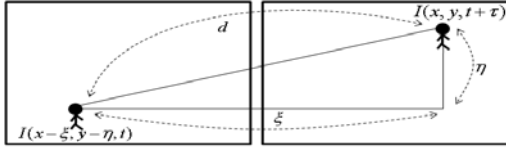


그림 7. 특징 벡터 생성 방법  
Fig 7. Method for creating feature vectors

$$G = \begin{bmatrix} g_{xx} & g_{xy} \\ g_{yx} & g_{yy} \end{bmatrix} \quad (4)$$

$t + \tau$  시간 후의 영상은 이전 영상에서 모든 점들의 움직임 위치와 같다.

$$I(x, y, t + \tau) = I(x - \xi(x, y, t, \tau), y - \eta(x, y, t, \tau)) \quad (3)$$

$\delta = (\xi, \eta)$ 는 시간  $t$ 에서  $t + \tau$ 로 변할 때 점  $X = (x, y)$ 에서의 변위이다. 그리고 생성된 수식은 잡음 때문에 다음 픽셀의 위치를 알아내기 힘들어 유사 변환  $\delta = GX + d$ 을 실시한다. 여기서  $G$ 는 변형 행렬이고,

$$G = \begin{bmatrix} g_{xx} & g_{xy} \\ g_{yx} & g_{yy} \end{bmatrix} \quad (4)$$

$d$ 는 이전 영상과 현재 영상과의 변위를 나타낸다. 이 값들은 특징 영역의 크기와 프레임간의 카메라 영상의 수에 따라 결정된다. 만약, 시간  $t + 1$ 일 때  $X$ 의 좌표의 데이터 값은  $I(AX + d; t + 1)$ 으로 정의되며, 여기서  $A = I + D$ 이고  $I$ 는  $2 \times 2$  단위행렬이다. 연속된 입력 영상에서  $X$  좌표의 움직임 벡터  $d$ 를 찾기 위해서  $\|\Delta d\| \leq \epsilon$ 일 때까지 반복하여 특징 벡터 제공 오차의 합을 계산하면 식(5)로 표현된다.,

$$\epsilon = \iint_{w \times d} [I(X + d; t) - I(X; t + \Delta t)]^2 w dX \quad (5)$$

$$d_j = d_j + \Delta d_j$$

여기서,  $\epsilon$ 는 특징 벡터의 제공 오차 합,  $d_j$ 는 움직임 벡터이고  $w d$ 는 영상의 크기,  $w$ 는 가중치 함수이며, 일반적으로 1로 놓는다.

### 3.5 2차 특징 벡터 - Snake 알고리즘

이동 물체의 1차 특징 벡터( $fp_1, fp_2, fp_3, fp_4, fp_5$ )에 대한 동정에 사용된 스네이크 알고리즘 에너지 함수는 내부 에너지 함수와 외부 에너지 함수로 구성되어 있다. 여기서는 스네이크 알고리즘의 에너지 함수를 최소화 시키는 점을 찾는 방향으로 알고리즘이 수행한다. 먼저, 내부 에너지 함수는 다음 식 (6)과 같다.

$$E_i^{Int} = |L_i - \|fp_j - fp_k\|| \quad (6)$$

여기서,  $fp_j$ 와  $fp_k$ 는  $L_i$ 과 인접한 특징 벡터이다. 다음으로, 외부 에너지 함수는 다음 식(7)과 같다.

$$E_i^{Ext} = - \sum_{x=fp_x-k}^{fp_x-k+m} \sum_{y=fp_y-k}^{fp_y-k+m} \bar{I}_b(x, y) \quad (7)$$

여기서  $k$ 는 특징 벡터와 실루엣 사이의 거리를,  $m$ 은 탐색공간을 나타낸다. 여기서, 이동 물체의 특징 벡터를 동정하면 내부 에너지 함수의 기하학적 관계는 식(8)과 같이

표현할 수 있다.

$$\begin{aligned} E_1^{Int} &= |L_1 - \|fp_{start} - fp_1\|| \\ E_2^{Int} &= |L_2 - \|fp_1 - fp_2\|| \\ E_3^{Int} &= |L_3 - \|fp_2 - fp_3\|| \\ E_4^{Int} &= |L_4 - \|fp_3 - fp_4\|| \\ E_5^{Int} &= |L_5 - \|fp_4 - fp_5\|| \end{aligned} \quad (8)$$

그리고, 외부 에너지 함수의 기하학적 관계를 보면 다음과 같이 식(9)로 표현할 수 있다.

$$\begin{aligned} E_1^{Ext} &= - \sum_{x=fp_{1x}-m_1}^{fp_{1x}+m_1} \sum_{y=fp_{1y}-k-m_1}^{fp_{1y}-k+m_1} \bar{I}_b(x, y) \\ E_2^{Ext} &= - \sum_{x=fp_{2x}-m_2}^{fp_{2x}+m_2} \sum_{y=fp_{2y}-k-m_2}^{fp_{2y}-k+m_2} \bar{I}_b(x, y) \\ E_3^{Ext} &= - \sum_{x=fp_{3x}-m_3}^{fp_{3x}+m_3} \sum_{y=fp_{3y}-k-m_3}^{fp_{3y}-k+m_3} \bar{I}_b(x, y) \\ E_4^{Ext} &= - \sum_{x=fp_{4x}-m_4}^{fp_{4x}+m_4} \sum_{y=fp_{4y}-k-m_4}^{fp_{4y}-k+m_4} \bar{I}_b(x, y) \\ E_5^{Ext} &= - \sum_{x=fp_{5x}-m_5}^{fp_{5x}+m_5} \sum_{y=fp_{5y}-k-m_5}^{fp_{5y}-k+m_5} \bar{I}_b(x, y) \end{aligned} \quad (9)$$

최종적으로 에너지 함수  $E_i$ 는 내부와 외부 에너지 함수의 볼록 합으로 나타내어진다.

$$E_i = K \bar{E}_i^{Ext} + (1 - K) E_i^{Int} \quad (10)$$

여기서,  $K$ 는 볼록 합 파라미터,  $\bar{E}_i^{Ext}$ 은 정규화 된 외부 에너지 함수 값이다.

$$\bar{E}_i^{Ext} = \frac{E_i^{Ext}}{\sum E_K^{Ext}} \quad (11)$$

여기서,  $K$ 가 크면 실루엣에 더 민감하고,  $K$ 가 작으면 비율에 민감하다. 결국,  $K$ 의 크기에 따라 새로운 특징 벡터를 추출한다. 그리고 탐색 모델의 초기 특징 벡터  $F_i$ 는 다음과 같다.

$$F_i = [F_{ihx}, 240 - F_{ihy} + fp_{start}] \quad (12)$$

여기서,  $L_i$ 는 10명의 신체 비례 정보를 이용하여 얻은 대략적인 중요 신체 관절 간의 거리, 즉 특징 벡터간의 거리를 나타낸다. 수식으로 표현하면,

$$L_1 = L_2 = L_3 = L_4 = L_5 = \frac{38}{27} B_w \quad (13)$$

여기서,  $B_w$ 는 RGB 칼라 모델과 HSI 칼라 모델을 이용하여 추출한 실루엣의 가로 넓이다.

### 3.6 물체 추적 - Particle filter 알고리즘

먼저, 이동 물체의 방향  $x_t$ , 위치  $d_t$ , 상태  $s_t$ , 라고 하면 시간  $t$ 일 때 이동 물체의 위치 예측하는 위치 식(14)는,

$$Bel(x_t) = p(x_t | s_t, d_t, s_{t-1}, d_{t-1}, \dots, s_0, d_0) \quad (14)$$

이와 같은 상태에서 새로운 위치로 이동 할 때 예측 식(15)은 다음과 같다.

$$Bel^-(x_t) \leftarrow \int p(x_t|x_{t-1}, d_t) Bel(x_{t-1}) dx_{t-1} \quad (15)$$

$$Bel(x_t) \leftarrow \eta_t p(s_t|x_t) Bel^-(x_t)$$

여기서,  $p(x_t|x_{t-1}, d_t)$ 는 물체의 움직임,  $p(s_t|x_t)$ 은 현재 이동 물체의 위치에서 나타나는 상태 값이다. 그리고  $\eta_t$ 는 정규화 상수이다. 가중치  $w_t$ 를 가지는  $n$ 개의 샘플의 집합  $P_t$ 는 식(16)로 나타낸다.

$$P_t = \{ \langle x_t^{(i)}, w_t^{(i)} \rangle | i = 1, \dots, n \} \quad (16)$$

여기서,  $x_t^{(i)}$ 는 샘플의 위치와 방향,  $w_t^{(i)}$ 는 각 샘플의 가중치이며 총 합은 1이다. 그리고 샘플을 분포할 경우 샘플 당 오차를 고려(가중치)하여 분포 시켜야 한다. 샘플 당 오차는  $p(x_t|x_{t-1}, d_t)$ 으로 표현되고 오차를 이용하여 각 샘플들에 대해 분포를 구해보면 다음과 같다. 먼저, 예측 분포는 식(17)으로 표현된다.

$$p(x_t|x_{t-1}, d_t) p(x_{0:t-1}|s_{0:t-1}, d_{0:t-1}) \quad (17)$$

또한, 대상 분포는 식(18)과 같이 나타낼 수 있다.

$$\alpha p(s_t|x_t) p(x_t|x_{t-1}, d_t) p(x_{0:t-1}|s_{0:t-1}, d_{0:t-1}) \quad (18)$$

그 다음, 최종 구해진 예측 분포와 대상 분포를 이용하여 각 샘플 당 가중치  $w_t$ 를 구한다.

$$w_t = \frac{\alpha p(s_t|x_t) p(x_t|x_{t-1}, d_t) p(x_{0:t-1}|s_{0:t-1}, d_{0:t-1})}{p(x_t|x_{t-1}, d_t) p(x_{0:t-1}|s_{0:t-1}, d_{0:t-1})} \quad (19)$$

$$= \alpha p(s_t|x_t)$$

$$\propto \alpha p(s_t|x_t)$$

결과적으로 샘플 당 가중치를 식 (20)으로 결정한다.

$$w_t = p(s_t|x_t) \times w_{t-1} \quad (20)$$

#### 4. 실험 및 결과 고찰

본 실험에 사용된 영상은 320\*240 크기의 24bit 칼라 영상이고 프레임 속도는 15 frame/1sec 이다. 먼저, RGB 칼라 모델을 이용하여 입력 영상에 진입한 이동 물체의 초기 영역을 추출한다. 그 다음, KLT 알고리즘을 이용하여 전체 입력 영상에 대한 특징 벡터들을 추출한다.

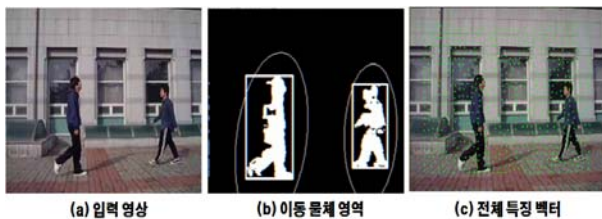


그림 8. 특징 벡터 추출 결과

Fig 8. Feature vector extraction result

추출된 특징 벡터들은 입력 영상에서 이동하는 물체의 대한 정보를 모두 가지고 있다고 이미 위에서 가정 하였다. 결과는 그림 8(c)를 통해 확인 할 수 있다.

그 다음, 초기 추출된 이동 물체의 움직임 영역에 이 특

징 벡터를 매칭시켜 1차 특징 벡터를 구한다. 결과는 그림 9(c)를 통해 확인할 수 있다. 두 번째로, RGB와 HSI 칼라 모델을 이용하여 이동 물체의 움직임 영역을 추출하고, 앞서 구한 1차 특징 벡터에 Snake 알고리즘을 적용함으로써 새로운 특징 벡터를 구한다. 그 다음, 기 추출된 이동 물체의 움직임 영역에 이 새롭게 구한 특징 벡터를 매칭시켜 2차 특징 벡터를 구한다. 결과는 그림 10을 통해서 확인 할 수 있다.

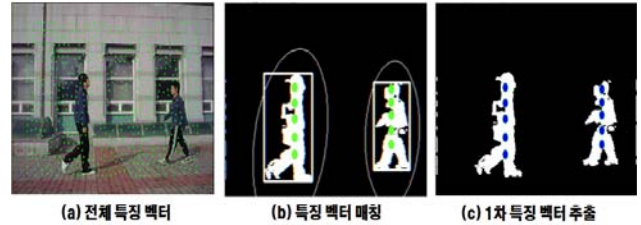


그림 9. 1차 특징 벡터 추출 결과

Fig 9. 1st feature vectors extraction result

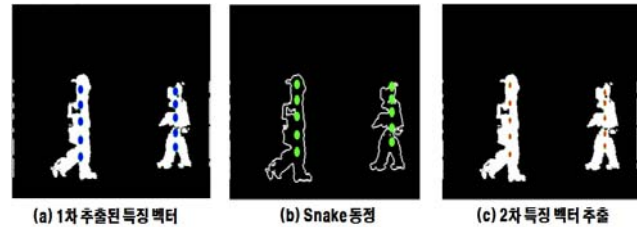


그림 10. 2차 특징 벡터 추출 결과

Fig 10. 2nd feature vectors extraction result

최종적으로, 2차 특징 벡터에 Particle filter를 적용시켜 이동 물체를 추적한다. 결과는 그림 11을 통해 확인 할 수 있다.



그림 11. 물체 추적 결과

Fig 11. Object tracking result

마지막으로, 제안된 물체 추적 알고리즘을 이용하여 입력 영상에 진입한 이동 물체를 실시간 추적한다. 그림 12는 시험 결과를 보여준다. 실험 결과에 의하면 영상에서 두 물체가 겹치는 현상이 발생하더라도 각각의 물체에 대한 개별적인 추적이 가능함을 알 수 있다.

#### 5. 결 론

본 논문에서는 Particle filter를 이용한 특징 벡터 기반 이동 물체 추적 알고리즘을 제안하였다. 이를 위해, 첫 번째, RGB 칼라 모델을 이용하여 초기 이동 물체의 움직임



영역을 추출하였고, KLT-알고리즘을 이용하여 입력 영상에 대한 특징 벡터를 구하였다. 그 다음, 초기 추출된 이동 물체의 움직임 영역에 이 특징 벡터를 매칭시켜 1차 특징 벡터를 구하였다. 두 번째로, RGB와 HSI 칼라모델을 이용하여 이동 물체의 움직임 영역을 추출하였고, 앞서 구한 1차 특징 벡터에 Snake 알고리즘을 적용시켜 새로운 특징 벡터를 구하였다. 그 다음, 기 추출된 이동 물체의 움직임 영역에 이 새롭게 구한 특징 벡터를 매칭시켜 2차 특징 벡터를 구하였다. 최종적으로, 구해진 2차 특징 벡터에 Particle filter를 적용시켜 이동 물체를 추적하였다. 마지막으로, 본 논문에서 제안한 알고리즘을 복잡하고 다양한 환경에서 실험을 통해 그 응용 가능성을 증명하였다.



그림 12. 실험 결과  
Fig 12. Experimental results

### 참 고 문 헌

[1] D. M. Gavrila, L. S. Davis, "Towards 3D model based tracking and recognition of human movement: a multi view approach", Int. Workshop on Face and Gesture Recognition, Vol. 1, pp. 272-277, 1995. 6.

[2] V. I. Pavlovic, R. Sharma, and T. S. Huang, "Visual interpretation of hand gestures for human computer interaction: A review", IEEE, Trans. on PAMI, Vol. 19, No. 7, pp. 677-695, July, 1997.

[3] L. Matthies, T. Kanade, and R. Szeliski "Kalman filter based algorithms for estimating depth from image sequences". IJCV, Vol. 3, No. 3, pp. 209-236, 1989, 3.

[4] H. W. Sorenson, "Least-squares estimation: From gauss to kalman", IEEE Spectrum, Vol. 7, pp. 63-68, 1970.

[5] G. J. Jang and I. S. Kweon. "Robust objects tracking using an adaptive color model" Int. Conf. on Robotics and Automation. Vol. 2, pp. 1677-1682, 2001. 5.

[6] G. Healey and D. Slater, "Using illumination invariant color histogram descriptors for recognition", Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, pp.355-360, 1994.

[7] K. S. Bhat, M. Saptharishi, and P. K. Khosla, "Motion detection and segmentation using image mosaics," in Proc. IEEE Int. Conf. Multimedia and Expo, Vol. 3, pp. 1577-1580, 2000. 6.

[8] G. D. Finlayson, B. Schiele, and J. L. Crowley. "Comprehensive color image normalization". 5th European Conference on Computer Vision, Vol. 1, pp. 475-490, 1998.

[9] 김세진, 주영훈 "가상 모형을 이용한 움직임 추출 알고리즘", 한국 지능 시스템 학회 Vol. 18, No. 6, pp. 731-736 2008. 12.

[10] B. D. Lucas and T. Kanade. "An iterative image registration technique with an application to stereo vision", Int. Joint Conference on Artificial Intelligence, pp. 674-679, 1981, 8.

[11] <http://opencvlibrary.sourceforge.net>

[12] C. Garcia and G. Tziritas, "Face detection using quantized skin color regions merging and wavelet packet analysis", IEEE Trans. on Multimedia, Vol. 1, No. 3, pp. 264-277, 1999, 9.

[13] C. Xu, and J. Prince, "Snakes, shapes, and gradient vector flow", IEEE. Trans. on Image Processing, Vol. 7, No. 3, pp. 359-369, 1998.

[14] U. C. Jung, S. H. Jin, X. D. Pham, J. W. Jeon, J. E. Byun, H. Kang, "A real-time object tracking system using a particle filter", 2006 IEEE/ RSJ Int. Conf. vol. 9, pp. 2822- 2827, 2006. 10.

### 저 자 소 개



김세진(Se Jin Kim)

2008년 : 군산대학교 전자정보공학부 졸업  
2008년~현재 : 동 대학원 전자정보공학부 석사과정

관심분야 : 지능형 로봇, 인간-로봇 인터랙션, 보안 감시 시스템, 지능제어.

Phone : 063-469-4706

E-mail : blessedu@kunsan.ac.kr



주영훈(Young Hoon Joo)

2009년 제 19권 제 3호 참조