

# 모바일 감시 로봇을 위한 실시간 움직임 추정 알고리즘

## Real-Time Motion Estimation Algorithm for Mobile Surveillance Robot

한철훈 · 심귀보\*

Cheol-Hoon Han and Kwee-Bo Sim\*

중앙대학교 전자전기공학부

### 요 약

본 논문에서는 파티클 필터(Particle Filter)를 사용한 모바일 감시 로봇을 위한 실시간 움직임 추정 알고리즘을 제안한다. 파티클 필터는 몬테카를로(Monte Carlo) 샘플링 방법을 기반으로 사전분포확률(Prior distribution probability)와 사후분포확률(Posterior distribution probability)을 가지는 베이직안 조건 확률 모델(Bayesian conditional probabilities model)을 사용하는 방법이다. 그러나 대부분의 파티클 필터에서는 초기 확률밀도(Prior probability density)를 임의로 정의하여 사용하지만, 본 논문에서는 Sum of Absolute Difference (SAD)를 이용하여 초기 확률밀도를 구하고, 이를 파티클 필터에 적용하여 모바일 감시 로봇 환경에서 임의로 움직이는 물체를 강인하게 실시간으로 추정하고 추적하는 시스템을 구현하였다.

### Abstract

This paper presents the motion estimation algorithm on real-time for mobile surveillance robot using particle filter. the particle filter that based on the monte carlo's sampling method, use bayesian conditional probability model which having prior distribution probability and posterior distribution probability. However, the initial probability density was set to define randomly in the most of particle filter. In this paper, we find first the initial probability density using Sum of Absolute Difference(SAD). and we applied it in the partical filter. In result, more robust real-time estimation and tracking system on the randomly moving objet was realized in the mobile surveillance robot environments.

**Key Words** : Object Tracking, Particle Filter, SAD, 모바일 감시 로봇, Bayesian Filter

## 1. 서 론

영상이 움직이는 물체를 포함한 경우 이 물체의 이동은 연속된 프레임(frame) 사이에서의 밝기(intensity)의 변화로 나타난다. 이러한 밝기의 변화로부터 움직임을 찾아내는 것을 움직임 추정(motion estimation)이라 한다. 움직임 추정에서 추정 대상을 사람으로 한 응용 분야는 감시·보안 시스템, HCI(Human Computer Interface), 가상 현실 등이 있다. 컴퓨터 비전 시스템의 기술이 발전함에 따라 신분확인, 표정인식, 제스처인식, 환경인식 등의 인식분야에 대한 연구가 많아지면서 인식을 적용할 때 영역의 분할(segmentation)의 목적으로 추정 알고리즘이 필수적으로 요구된다.

영상 부호화에 사용되는 대표적 이동 추정 방법으로는 블록 정합 알고리즘(Block Matching Algorithm:BMA)과 화소 재귀적 알고리즘(Pel-Recursive Algorithm:PRA)을 들 수 있다. 블록 정합 알고리즘에서는 영상의 한 프레임을 동일한 크기의 여러 개의 블록들로 분할하고 이들 각 블록

에 대해 정합 방법을 사용하여 움직임 벡터를 찾는다. 즉 현재 영상의 한 블록을 이전 영상의 특정 범위에 정합시켜서 가장 유사한 블록을 찾아내어 이 블록과 현재 블록과의 위치 차이를 움직임 벡터로 삼는다. 이렇게 블록 단위로 움직임 벡터를 구할 경우에 영상에 대한 기본 가정은 같은 블록 내의 화소들은 모두 같은 방향으로 병진 운동(translational movement)을 한다는 것이다. 블록의 크기가 작을수록 이러한 가정의 신뢰도가 높아지나 움직임 벡터의 계산량과 전송량이 증가하게 된다. 한편 화소 재귀적 알고리즘에서는 화소 단위로 움직임을 추정하게 되는데 먼저 인접한 화소(pixel)의 움직임 벡터를 사용하여 현재 화소의 움직임 벡터를 예측하고 이를 경사법(steepest descent method)을 사용하여 보정한다. 일반적으로 화소 재귀적 알고리즘은 블록 정합 알고리즘에 비해 복잡한 움직임을 잘 반영한다고 한다. 왜냐하면 화소단위로 이동 벡터를 구하므로 블록 정합 알고리즘에서와 같이 한 블록 내의 화소들이 모두 같은 방향으로 병진 운동을 한다고 가정할 필요가 없기 때문이다. 하지만 경사법을 사용하여 이동 벡터를 보정하기 때문에 움직이는 물체의 가장자리에서는 물체의 움직임을 잘 반영하지 못하며 빠른 움직임을 갖는 영상에서 계산량이 많아지는 문제점을 갖는다[1][2].

본 논문에서는 영상에서 물체의 움직임을 추정하기 위해 블록 정합 알고리즘의 SAD(Sum of Absolute Differences)를 사용하였다. SAD 알고리즘은 연속적으로 입력되는 영상에서 현재영상과 이전영상의 차 영상을 구하여 차 영상의

접수일자 : 2009년 3월 6일

완료일자 : 2009년 6월 6일

\* 교신 저자

감사의 글 : 이 논문은 중소기업청 · 서울시의 연구비지원에 의한 2008년도 산학연공동기술개발사업 연구비 지원을 받아 수행된 연구입니다. 연구비지원에 감사드립니다.

화소값(Pixel value)이 임계치가 넘는 경우를 움직임이 발생했다고 가정한다[3][4]. 이렇게 블록단위로 움직임 벡터를 구할 수 있고 움직임 벡터가 가장 큰 블록의 대비(Contrast)와 색상(Color), 좌표(Coordinate) 그리고 에지(Edge) 정보 등을 이용하여 확률모델을 사용하는 파티클 필터(Particle Filter)에 초기 파라미터(Initial Parameter)인 사전분포확률에 적용할 수 있다.

파티클 필터(Particle Filter)는 몬테카를로(Monte Carlo) 샘플링 방법에 기반하여 사전분포확률(Prior distribution probability)과 사후분포확률(Posterior distribution probability)을 가지는 베이저안 조건 확률 모델(Bayesian conditional probabilities model)을 사용하는 방법이다. 파티클 필터를 이용하여 움직이는 모바일 로봇의 카메라의 환경에서 임의로 움직이는 물체를 강인하게 추정 및 추적이 가능하게 하였다. 파티클 필터는 컴퓨터 비전분야에서 Object tracking에 많이 사용되고 있다. 하지만 불필요한 샘플링과 계산의 복잡성에서 상태 벡터(State vector)를 구하는데 다차원의 데이터를 처리하여야 한다. 그리고 물체의 추적을 위해 사전 확률(Prior Probability)이 꼭 필요하며 사전확률의 정확도에 따라 파티클 필터의 성능이 좌우될 수 있다. 따라서 본 논문에서는 이러한 초기의 사전확률의 정확도를 높이기 위해 앞서 언급한 블록 정합 알고리즘의 SAD 방법을 사용하여 파티클 필터의 사전확률밀도함수를 생성하였다. 그림 1은 본 논문에서 제안하는 전체적인 시스템의 구성도이다.

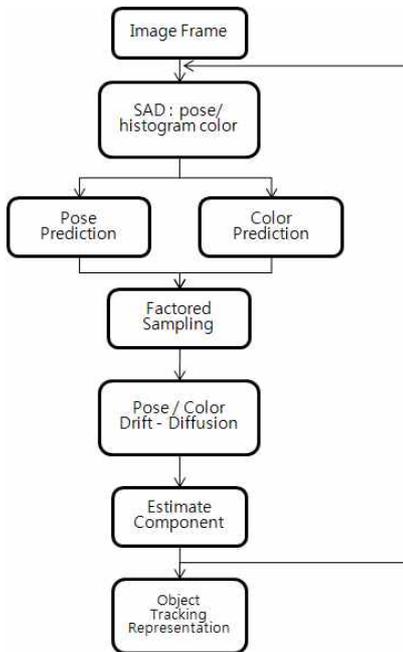


그림 1. 전체적인 시스템 구성도.  
Fig. 1. Overall System Diagram

입력된 영상에서 SAD방법을 사용하여 초기의 움직임 벡터에 대한 좌표와 색상의 히스토그램 분포를 확인한다. 그리고 얻어진 좌표와 히스토그램에 대한 파티클을 생성하는 샘플링 과정을 거친 후 파티클 필터를 통하여 움직임 벡터의 좌표와 색상 히스토그램의 분포를 추정하게 되고 최종적으로 움직이는 물체에 대해 트래킹(Tracking)을 수행하게

된다.

본 논문의 구성은 다음과 같다. 2장에서 절대차의 합 알고리즘(SAD)에 대해서 설명하고, 3장에서는 세부적으로 파티클 필터(Particle Filter)에 대해 기술한다. 그리고 4장에서 제안한 알고리즘을 이용한 물체 추적 실험을 기술하고, 마지막으로 5장에서 결론을 맺는다.

## 2. 절대차의 합 알고리즘

대부분의 블록 정합 움직임 추정에서 가장 정합이 잘되는 위치를 결정하기 위한 방법으로 블록 정합 오류 평가기준으로 SAD(Sum of Absolute Difference)알고리즘을 사용하고 있다. 영상에서  $(x, y)$ 에 위치하는  $N \times N$  크기의 블록에 대한 SAD는 식 (1)과 같이 계산되어 진다.

$$SAD(dx, dy) = \sum_{m=x}^{x+N-1} \sum_{n=y}^{y+N-1} |I_t(m, n) - I_{t-1}(m+dx, n+dy)| \quad (1)$$

여기서  $I_t$ 는 현재 프레임,  $I_{t-1}$ 는 이전 프레임이며,  $(dx, dy)$ 는  $(x, y)$ 로부터 변위 즉 움직임 벡터이다. 그림 2는 영상 프레임에서 시간  $t$ 와  $t-1$ 영상의 차를 이용하여 움직임 벡터를 추출한 영상을 보여준다. 위쪽에 위치한 이미지는 사람이 시간에 따라 움직이는 테스트 영상이다. 이 두 영상을 이용하여 차 영상을 얻으면 아래의 이미지처럼 차 영상을 이용한 움직임 벡터를 얻을 수 있다.



그림 2. 차 영상을 이용한 움직임 벡터 검출.  
Fig. 2. Detection of motion vector using Difference image

그림 3은 탐색 영역 내에서 전역 탐색을 수행하여 각 위치의 SAD 값의 분포를 나타낸 것이다. 그림에서 보는 것과 같이 영상 프레임  $I_t$ 와 다음 프레임의  $I_{t-1}$ 의 정합 오류가 뚜렷하게 나타나며 이 영역을 움직임의 벡터로 표현할 수 있다.

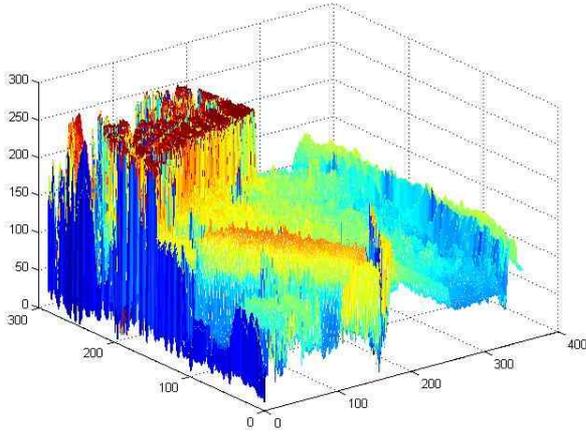


그림 3. 전역탐색을 통한 SAD 결과 예.  
Fig. 3. SAD Result through the full scanning

### 3. 파티클 필터

#### 3.1 배경

본 논문에서 감시시스템을 위해 움직이는 물체를 추적하기 위한 알고리즘으로 파티클 필터를 사용하였다. 최근에 입의 물체를 추적하는데 파티클 필터가 강력한 성능을 발휘하며 그만큼 많이 사용되고 있다.

파티클 필터는 베이즈 이론(Bayes' theorem)의 반복적인 수행을 통해 계산되어지는 Bayesian filter를 사용한다. 추적하는 물체의 모든 데이터는 상태  $X_t$ 에 대해 사후(Posterior) 확률밀도함수(PDF: Probability Density Function)  $f_{X_t|Z_{1:t-1}}(X_t|Z_{1:t-1})$ 와 영상프레임에서 관측된 데이터  $Z_{1:t}$ 를 이용하여 유도될 수 있다. 파티클 필터에서 사후확률은 N개의 상태 집합과 샘플링을 통한 파티클  $\{\epsilon^1_t, \dots, \epsilon^N_t\}$ 로 나타낼 수 있다[5].

파티클 필터는 크게 두 단계로 구분할 수 있는데, 예측(Prediction)단계와 관측(Observation)단계이다.

예측단계(Prediction)에서 시간  $t$ 에서 사전확률(Prior PDF)  $f_{X_t|Z_{1:t-1}}(X_t|Z_{1:t-1})$ 는 시간  $t-1$ 에서의 사후 확률로부터 유도될 수 있다.

$$f_{X_t|Z_{1:t-1}}(X_t|Z_{1:t-1}) = \int f_{X_t|X_{t-1}}(X_t|X_{t-1})f_{X_{t-1}|Z_{1:t-1}}(X_{t-1}|Z_{1:t-1})dX_{t-1} \quad (2)$$

식 2는 예측단계에서의 사전확률을 유도하는 과정을 수식으로 나타낸 것이다. 또한 상태정보  $X_t$ 는 이전 상태와 독립적인 현재 상태의 노이즈(Noise)를 포함하고 있다.

관측(Observation)단계에서 상태  $X$ 에 대한 관측데이터는 다음의 수식으로 표현 할 수 있다.

$$Z_t = h(X_t, V_t) \quad (3)$$

여기서  $V_t$ 는 상태  $X_t$ 와 독립적인 관측데이터의 노이즈를 나타낸다. 그리고 관측데이터  $Z_t$ 는  $f_{Z_t|X_t}(Z_t|X_t)$ 를 이용하여 유도할 수 있다. 다음의 그림 4는 파티클 필터를 한번 수행했을 경우의 과정을 도시화한 것이다.

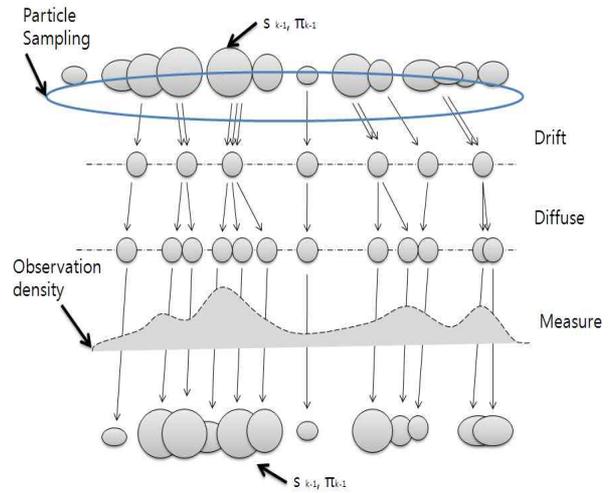


그림 4. 파티클 필터 알고리즘 단계  
Fig. 4. Once step in the Particle Filter

#### 3.2 샘플링 단계

초기의 영상에서 SAD를 거쳐 생성된 히스토그램 색상분포와 좌표를 이용하여 새로운 샘플링 집합을 생성하게 된다. 색상의 분포는 SAD를 설정할 때 분할한 블록의 중심을 구하기 위해  $x$ 축과  $y$ 축의 반을 이용하여  $H_x$ 와  $H_y$ 로 표현한다. 위치  $y$ 에서의 색상 분포는 다음과 같이 정리할 수 있다.

$$p_y^{(u)} = f \sum_{i=1}^I k \left( \frac{\|y - x_i\|}{a} \right) \delta[h(x_i) - u] \quad (4)$$

위 식 4에서  $I$ 는 그 영역에서의 픽셀(Pixel)의 개수이며  $\delta$ 는 크로네커 델타 함수(Kronecker delta function),  $h(x_i)$ 는 위치  $x_i$ 에서 히스토그램 함수를 나타낸다. 그리고 파라메타  $a = \sqrt{H_x^2 + H_y^2}$ 로 블록의 크기를 적용하기 위해 사용한다. 따라서 각 샘플의 집합은 다음과 같이 나타낼 수 있다[6].

$$s = \{x, y, H_x, H_y, q\} \quad (5)$$

여기서  $x, y$ 는 블록의 위치를 나타내며  $q$ 는 히스토그램을 나타낸다.

샘플링 단계에서 사용자 설정에 의한 N개의 파티클 샘플은  $[0, 1]$ 의 범위에서 임의로 선택된다. 이러한 샘플의 선택 방법은 진화 알고리즘(GA)의 룰렛휠 선택(roulette wheel selection)방식과 비슷한 방법이다. 추정 대상이 되는 물체의 평균 위치는 다음 단계의 대상 물체의 중심을 갱신하기 위해 사용된다.

#### 3.3 예측 단계

예측 단계에서 상태  $X_{t-1}$ 에서  $X_t$ 로 진행될 때 추적되는 물체(Object)의 운동이 확률적으로 예측이 가능하다면 이를 반영한다. 각 샘플을 확률적 동적 모델에서 샘플링(Sampling)하여 새로운 샘플  $s_t^i$ 을 얻는다.

Dynamic model의 응용을 통해 샘플 파티클 집합이 유도된다.

$$s_t = A s_{t-1} + W_{t-1} \quad (6)$$

여기서  $A$ 는 서로 다른 위치에 있는 각 입자들의 비선형 drift를 표현하며  $W_{t-1}$ 는 가우시안 랜덤 분산(Gaussian random variable)을 나타낸다. 또한  $A$ 는 일차 모델(First order model)을 사용하였다.

파티클 샘플 집합의 가중치(Weight)를 갱신하기 위해 목표 히스토그램과 좌표를 샘플 파티클에 의해 예측된 히스토그램과 좌표와 매칭(Computation)을 사용한다. 각각 샘플 파티클의 상태 벡터  $s^{(n)}$ 에 대해 히스토그램  $q$ 와 예측된 히스토그램  $p_{s^{(n)}}$  그리고 좌표  $g$ 와 예측된 좌표  $d_{s^{(n)}}$ 는 다음의 식 4에 의해 계산 되어질 수 있다. 따라서 목표 모델의 가중치는 다음 식 7과 같이 분산  $\sigma$ 을 갖는 가우시안 함수로 나타낼 수 있다.

$$\pi^{(n)} = \frac{1}{\sqrt{2\pi}\sigma} \exp - \frac{((g - d_{s^{(n)}})^2 - (q - p_{s^{(n)}})^2)}{2\sigma^2} \quad (7)$$

### 3.4 관측 단계(Measurement Stage)

파티클 필터(Paticle filter)알고리즘에서 사용하는 확률 분포는 칼만 필터(Kalman filter)에서 사용한 확률 분포를 기초로 해서 더 일반화된 경우에도 추정이 가능하도록 하였다. 이 방법은 영상에 심한 Visual Clutter에서 원하는 물체의 추적 가능하게 한다. 여기서 초기의 움직임 벡터에 대한 히스토그램 색상과 좌표는 확률 밀도함수(PDF)로 나타내고, PDF는 가중치를 가지는 샘플 파티클 집합으로 표현된다.

$$E[s_t] = \sum_{n=1}^N \pi_t^{(n)} s_t^{(n)} \quad (8)$$

위 식 8에서  $s_t$ 는 시간  $t$ 에서 샘플 집합을 나타내며  $\pi_t^{(n)}$ 는  $n$ 번째 샘플에서 가중치(Weight)를 나타낸다.

다음은 파티클 필터(Particle Filter)의 반복적(iteration)으로 수행되는 단계를 간략히 정리한 것이다.

초기에 주어진 파티클 샘플 집합  $\{X_{t-1}^{(i)}, \pi_{t-1}^{(i)}\}$ 에서 새로운 파티클 샘플 집합  $\{X_t^{(i)}, \pi_t^{(i)}\}_{t=1,2,\dots,N}$ 을 구한다. 가중치  $\pi_0^{(i)} = 1/N$ 와 누적 분포(Cumulative distribution)  $c_0^{(i)} = i/N$ 을 초기화 한다.

1. 다음의 방법을 이용하여 샘플  $s_t^{(i)}$ 을 선택한다.
  - 균일하게 분포된 랜덤 값을 발생시킨다.  $r = [0,1]$
  - $c_{t-1}^{(j)} \geq r$  인 가장 작은  $j$ 을 찾는다.
  - $s_t^{(i)} = X_{t-1}^{(j)}$ 을 설정한다.
2.  $s_t^{(i)}$ 을 얻기 위해 샘플링으로부터 예측한다.  $P(X_t | X_{t-1} = s_t^{(i)})$
3. 물체의 특징 파라메타를 측정하고 가중치를 갱신한다.

$\pi_t^{(i)} = P(Z_t | X_t = \hat{s}_t^{(i)} | t_{t-1}) \sum_{i=1}^N \pi_t^{(i)} = 1$ 이 되도록 설정하고 누적 분포  $c_t^{(i)}$ 을 계산한다. 그리고  $\{X_t^{(i)}, \pi_t^{(i)}\}$ 을 갱신한다.

## 4. 실험 및 결과

본 논문에서 제안하는 움직임 추정 알고리즘의 실험을 위한 환경으로 다음의 표 1을 사용하였다. 또한 실험에 사

용한 영상프레임은 카메라를 직접 사용하지 않고 Edinburgh 대학의 CAVIAR 프로젝트에 사용된 동영상들을 사용하여 실험하였다.

표 1. 실험 환경

Table 1. Experimental environments

OS	MS Window XP SP3
CPU	Intel Core 2 Duo 2.8G
RAM	2G Byte
Compiler	MatLab 7.1
Format of Movies	Uncompressed AVI(320×240)

실험에서 파티클 필터를 이용하여 움직이는 물체를 추정(Estimate)하기 위해서 두 가지의 관측 데이터를 사용하였다. 하나는 움직임 벡터에 대한 좌표 정보와 다른 하나는 움직임 벡터 영역에 대한 색상 히스토그램 정보이다.

다음의 그림 5는 MatLab을 이용하여 파티클 필터의 성능을 실험한 결과이다. 각각의 위치와 히스토그램 정보를 이용하여 True Data를 추정하는 결과를 나타낸다. 그림 5에서는 샘플링하는 파티클(Particle)의 생성 개수를 50로 설정하여 실험한 결과이며, 그림 6은 파티클 샘플을 100로 하였고, 그림 7은 생성하는 파티클 샘플을 200으로 하여 실험한 결과를 나타낸 것이다.

실험 결과에서 볼 수 있듯이 샘플링하는 파티클의 개수에 따라서 그 성능이 달라짐을 알 수 있고 관측 데이터에 대한 샘플이 임계치 이상이 되면 그 성능에 크게 영향을 주지 않음을 100개의 샘플과 200개의 샘플을 생성하여 실험한 결과를 통하여 알 수 있다.

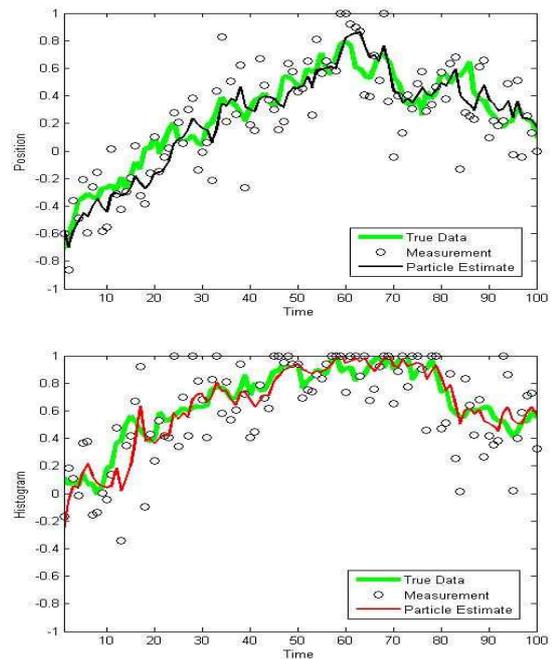


그림 5. 히스토그램과 위치 정보를 이용한 파티클 필터 성능 실험 (N = 50)  
Fig. 5. Experiment of particle filter with Histogram and Position information (N =50)

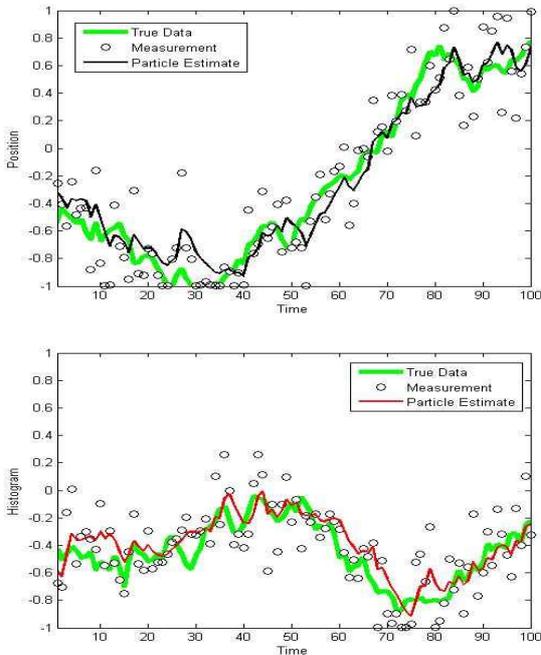


그림 6. 히스토그램과 위치 정보를 이용한 파티클 필터 성능 실험 (N = 100)  
 Fig. 6. Experiment of particle filter with Histogram and Position information (N =100)

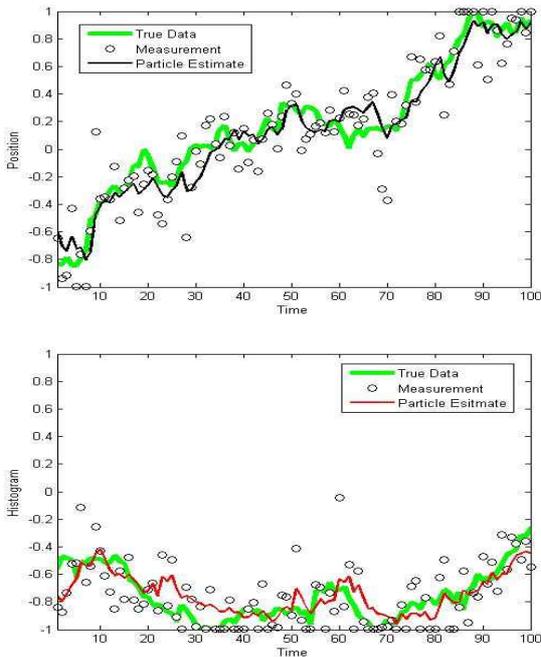


그림 7. 히스토그램과 위치 정보를 이용한 파티클 필터 성능 실험 (N = 200)  
 Fig. 7. Experiment of particle filter with Histogram and Position information (N =200)



그림 8. 히스토그램과 위치 정보를 이용한 움직이는 물체 추적 결과  
 Fig. 8. Experiment of Object Tracking using Both Histogram and Position information

위의 그림 8은 최종적으로 본 논문에서 제안한 알고리즘을 실험한 결과이다. 초기의 움직임 벡터를 갖는 블록을 SAD방법을 이용하여 검출하고 그 검출된 영역의 히스토그램 색상분포 밀도함수를 생성하고 블록의 중심 좌표(Centre)를 이용하여 샘플을 생성하여 관측 데이터(Measurement Data)의 샘플 파티클로 사용하게 된다. 결과에서 볼 수 있듯이 파티클 필터를 이용하여 움직이는 물체를 추적하는데 효율적인 성능을 가짐을 알 수 있다.

### 5. 결 론

베이시안 필터(Bayes Filter)의 확률밀도를 기반으로 하는 파티클 필터는 Object Tracking에 많이 사용되고 그 성능이 효율적임은 매우 잘 알려져 있다. 하지만 확률밀도를 계산하는 복잡성과 초기의 사전확률밀도(Prior probability density)를 정의하여야 한다는 문제점이 있었다. 이 문제를 극복하기 위해서 본 논문에서 영상 프레임간의 차 영상을 이용하는 블록 정합 알고리즘(Block Matching Algorithm)에서 SAD 방법을 사용하여 초기의 영상에 대한 움직임 벡터(Motion Vector)를 갖는 블록의 색상 히스토그램과 중심 좌표를 이용하여 초기 확률밀도를 설정하는 움직임 추정 알고리즘을 제안하였다. 실험 결과에서 볼 수 있듯이 제안한 알고리즘이 움직이는 물체에 대해 실시간으로 추적이 가능하여 실질적인 시스템에 적용이 가능함을 확인할 수 있었다.

향후 연구 계획은 본 논문에서는 제안한 알고리즘을 시스템에 적용하기 위해 직접 모바일 로봇에 탑재되어있는 카메라를 통해서 실험하고 이를 통해 개선·보완할 계획이다.

참 고 문 헌

[1] A.M. Tourapis, O.C.Au, and M.L.Liou, "Predictive motion vector field adaptive search technique (PMVFAST)- Enhancing block based motion estimation," *Proc. SPIE Visual Commun. Image Process.*, San Jose, CA, Jan. 2001.

[2] Yu-Wen Huang, Shao-Yi Chien, Bing-Yu Hsieh, and Liang-Gee Chen, "Global Elimination Algorithm and Architecture Design for Fast Block Matching Motion Estimation," *IEEE Transactions on circuits and systems for Video technology*, Vol. 14, No.6, June 2004.

[3] Cheng Du, Yun He, and Junli Zheng, PPHS: A Parabolic Prediction-Based, Fast Half-Pixel Search Algorithm for Very Low Bit-Rate Moving-Picture Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, No.6, June 2003.

[4] Bo Zhao and Jian Chen, "A Fast Two-Step Search Algorithm for Half-Pixel Motion Estimation," *Dept. of Electronic Engineering, Shanghai Jiaotong University*, Shanghai 2003, China

[5] K. Choo, and D.J. Fleet, "People Tracking with Hybrid Monte Carlo." *International Conference on Computer Vision*, Vol.2, 2001, pp.321-328.

[6] Katja Nummiaro, Esther Koller-Meier, Luc Van Gool, "An adaptive color-based particle filter," *Image and Vision Computing 21, 2003*, pp.99-110.

[7] Jayesh H.Kotecha and Petar M. Djuric. "Gaussian particle filtering." *IEEE Transactions On Signal Processing*, 51, 2003.

저 자 소 개



한철훈(Cheol-Hun Han)

2008년 : 중앙대학교 전자전기공학부  
공학사

2008년 ~ 현재 : 중앙대학교 대학원  
전자전기공학부 석사과정

관심분야 : Computer vision system, Emotion  
Recognition, Intelligent Robot 등



심귀보(Kwee-Bo Sim)

1990년 : The University of Tokyo  
전자공학과 공학박사

1991년 ~ 현재 : 중앙대학교  
전자전기공학부 교수

[제19권 제2호 (2009년 4월호) 참조]

E-mail : kbsim@cau.ac.kr  
Homepage URL : <http://alife.cau.ac.kr>