

Particle Filtering에 근거한 낙하하는 꽃잎의 운동궤적의 통계적 추정

이 재 우^{*†}

* 와세다대학교 창조이공학 연구과

Statistical Estimation of Motion Trajectories of Falling Petals Based on Particle Filtering

Jae Woo Lee^{*†}

* Dept. of Creative Science and Technology, Waseda Univ.

(Received December 22, 2015 ; Received April 30, 2016 ; Accepted May 22, 2016)

Key Words: Particle Filter(파티클 필터), Bayes Filter(베이지스 필터), Bio System(생물체 시스템) Motion Estimation(모션 추정), Petal Flying(꽃잎의 비행)

초록: 이 논문은 꽃잎들, 나비나 민들레 씨앗들과 같은 생물체 시스템의 불규칙한 운동을 파티클 필터링 이론에 근거하여 예측하고 추적하는 유용한 방법을 제안한다. 생물체 모사 시스템 설계에 있어서, 생체 시스템의 운동에 대한 관측과 생체 시스템 운동학에 대한 새로운 설계원리가 어떻게 자연스럽게 운동하는가에 대한 인상을 얻는데 중요하다. 공기 중에서 비행하는 꽃잎에 대한 시스템 모델링이 베이지안 확률 규칙을 사용하여 수행되었다. 실험결과는 제안된 방법이 공기의 난류로부터 유도된 랜덤한 외란이 있는 경우에도 잘 예측함을 보여준다.

Abstract: This paper presents a method for predicting and tracking the irregular motion of bio-systems, – such as petals of flowers, butterflies or seeds of dandelion – based on the particle filtering theory. In bio-inspired system design, the ability to predict the dynamic motion of particles through adequate, experimentally verified models is important. The modeling of petal particle systems falling in air was carried out using the Bayesian probability rule. The experimental results show that the suggested method has good predictive power in the case of random disturbances induced by the turbulence of air.

- 기호설명 -

1. 서론

$p(x_t|z_{0:t})$: 사후확률(posteriori probability)
 $p(z_t|x_t)$: 우도(likelihood)
 $p(x_t|x_{t-1})$: 운동모델(motion model)
 $x_0^i \sim p(x_0)$: 확률 p 에서 x_0^i 를 샘플링 함
 N : 파티클의 수
 w_t^i : t 시간에서의 weight factor
 N_{eff} : 유효 샘플의 크기

생물체의 능력을 모방하여 로봇의 자연에 대한 대응 능력을 향상시키는 방법들이 점점 더 많은 주목을 받고 있다. 생물체의 탁월한 능력은 오랜 세월이 걸친 진화에 의하여 자연에 적합하게 선택되었기 때문에 로봇이 이러한 능력을 가질 수 있다면 좀더 탁월한 능력을 낼 수 있다.^(1,2) 예를 들어 나비와 같은 곤충이나 바람을 타고 멀리 이동할 수 있는 식물의 꽃씨와 같은 마이크로 생물체의 운동을 비행체에 적용할 수 있다면 좀더 자연에 대한 이해가 높은 시스템의 설계가 가능하다.⁽³⁾

† Corresponding Author, jwlhee@gmail.com

© 2016 The Korean Society of Mechanical Engineers

본 논문에서는 위와 같은 소형로봇시스템 설계의 선행연구로서 꽃잎과 같이 바람의 영향을 심하게 받는 작은 생물체의 불규칙한 운동을 예측하고 추적할 수 있는 방법을 제시하고자 한다.

물체의 움직임을 추적하는 이론으로 주목 받았던 Kalman filter는 Bayesian recursive filter의 일종으로서 1960년 Rudolf Emil Kalman에 의해 체계화된 이론으로서 실제로 미사일이나 비행체의 추적과 같은 실제의 응용문제를 해결하는 데 많은 기여를 하였다.⁽⁴⁾

Kalman filter의 장점은 과거의 데이터를 근거로 하되 현재의 센서 값을 반영하여 미래의 운동을 예측할 수 있다는 데 있었다. 하지만, Kalman filter는 대상시스템의 상태 방정식이 선형이라는 가정을 하므로 불규칙하게 변동하는 비선형의 시스템에서는 오차가 커지는 단점이 있었다. 선형화의 단점은 극복하지 못했으나 순간 순간의 운동을 예측할 때 비선형 시스템을 순간적으로 선형으로 볼 수 있다는 Extended Kalman filter의 이론이 나와 선형화에서 발생하는 문제점을 어느 정도 해소시켜 주었다.⁽⁵⁻⁸⁾ Extended Kalman filter는 가장 많이 적용된 이론이지만 완전한 비선형을 전제로 하는 것이 아닌, 특정 순간의 선형화를 이용하는 방법이어서 역시 완전히 불규칙한 시스템에서는 한계가 있다.

Extended Kalman filter를 적용하기 위해서는 운동 방정식의 계수 행렬의 자코비안을 계산해야 한다. 자코비안을 계산하기 위해서 시스템모델이 필요하다. 시스템모델을 알 수 없을 때는 사용하기가 어렵다는 게 단점이다. 이에 대한 대안으로서 시스템모델을 사용하지 않고 확률분포를 샘플링방법을 통해 구하는 Particle filter와 같은 방법이 제시되었다.⁽⁹⁻¹⁴⁾

Particle filter는 Kalman filter와 달리 샘플링을 이용하여 분포를 구한다. 따라서 Kalman filter나 Extended Kalman filter에서와 같은 선형화에 대한 문제가 발생하지 않는다. Monte Carlo Sampling 기법이 많이 사용되는데 이 방법의 문제는 디제너레이션(degeneration)이라 불리는 샘플의 재선택에 의한 퇴화가 발생한다는 점이다.⁽¹⁵⁾ 이를 해결하기 위하여 그전에 샘플링에 사용되었던 샘플들을 다시 샘플링 하는 Resampling방법을 채택한다.

본 논문의 구성은 다음과 같다. 2장에서는 Particle filter의 이론을 설명하고 이를 사용하여 Filter를 설계하는 방법을 제시하고 3장에서는 설계된 Filter에 필요한 모델링 데이터를 실험을 통해 구한다. 4장에서는 실험 결과에 대해 분석하고

토의한다. 5장에서는 결론을 도출한다.

2. 파티클 필터 이론

Particle filter는 Recursive Bayesian filter의 한 종류이다. Bayesian filters에서는 사후확률(a posterior)이 최대가 되는 상태를 최적의 상태로 표현한다. 사후확률은 Bayes 확률이론에 의하면 사전확률(a priori)과 우도(likelihood)의 적으로 표현된다. 즉, 과거의 데이터에 현재의 측정치를 반영하여 현재 상태의 확률을 계산하는 Recursive Bayesian Filter의 한 종류이다.

Kalman filter와 달리 Particle filter는 다수의 sample들을 가지고 사후확률(a posterior)을 표현한다. 이는 Kalman filter가 가우스 분포(Gaussian distribution)를 파라미터의 형태(평균과 분산)로 반영하는 것과 차이가 있다. 이 때문에 Particle filter는 Bayes filter의 비 파라미터적인 표현방법 가운데 하나이다.

Bayes theory에 의하면 사후확률(a posterior)은 사전확률(a priori)과 우도(likelihood)의 적(multiplication)으로 식 (1)과 같이 표현할 수 있다.

$$p(x_t|z_{0:t}) = \eta p(z_t|x_t) \int p(x_t|x_{t-1})p(x_{t-1}|z_{0:t-1})dx_{t-1} \quad (1)$$

여기서 x_t 는 t 시간에서의 시스템의 상태이며 $x_t = (h, v)^T$ 와 같이 위치와 속도로 상태변수를 구성한다. $z_{0:t}$ 는 시스템의 상태 관측치이며, 이 논문에서는 꽃잎이 떨어지는 것을 촬영한 동영상을 사용한다. $p(x_t|z_{0:t})$ 는 우리가 알고 싶은 상태에 대한 사후 확률, $p(x_t|x_{t-1})$ 는 시스템의 $t-1$ 시간에서의 상태에서 t 시간의 상태로의 천이 즉, 상태의 운동을 나타낸다. $p(x_{t-1}|z_{0:t-1})$ 는 그 이전 시간에서

Table 1 Algorithm Particle filter

1	$\bar{x}_t = \chi_t = \phi$
2	for m = 1 to M do
3	sample $x_t^{[m]} \sim p(x_t x_{t-1}^{[m]})$
4	$w_t^{[m]} = p(z_t x_t^{[m]})$
5	$\bar{x}_t = \bar{x}_t + \langle x_t^{[m]}, w_t^{[m]} \rangle$
6	endfor
7	for m = 1 to M do
8	draw i with probability $\propto w_t^{[m]}$
9	add $x_t^{[i]}$ to χ_t
10	endfor
	return χ_t

의 시스템의 상태를 표현한다. Particle filter에서는 (1)의 우변의 적분을 particle이라 불리는 samples들과 그 particles의 가중치의 적(multiplication)으로 표현한다. Particle filter 알고리즘을 pseudo code로 표현하면 Table 1과 같다.⁽¹³⁾

위의 Table 1의 내용을 순서대로 설명하면 아래와 같다.

초기상태(a priori): 초기상태는 균일하게 분포한다고 가정하고 충분한 양의 N개의 particle을 준비한다. 꽃잎이 낙하하기 전에 초기상태는 위치와 속도가 0인 상태이다. 초기상태에서의 파티클을 샘플링 한다. 초기상태에서는 0위치에서 가우시안 분포를 한다고 가정한다. 파티클의 수가 m이라고 할 때,

$$x_0^{[m]} \sim N(\mu, \Sigma) \quad (2)$$

로부터 파티클 $x_0 = \{x_0^{[1]}, x_0^{[2]}, \dots, x_0^{[m]}\}$ 를 샘플링 한다. 또한 이 상태에서의 weight factor는 모든 파티클에 대해서 균일하다고 가정한다. 따라서 weight factor $w_0 = \{w_0^{[1]}, \dots, w_0^{[m]}\}$ 는 균일한 값을 갖는다.

예측과정(prediction process) : t-1시간에서의 m개의 각각의 particle에 대하여 시스템 모델로부터 random 하게 취한 확률로 샘플링하여 예측 particle들을 생성한다. 식 (1)에서 $\int p(x_t|x_{t-1})p(x_{t-1}|z_{0:t-1})dx_{t-1}$ 은 prediction을 나타낸다. 예측 과정은 과거의 데이터와 운동학적 모델링에 의존하여 현재의 상태를 ‘예측(prediction)’하는 과정이다. 따라서 이 ‘예측’에는 현재의 상태, 여기서는 센서로부터 들어오는 정보는 취합되어 있지 않다. 말 그대로 과거의 데이터와 합리적 가정을 통해 만든 운동학적 모델이 타당하다는 전제하에 현재의 상태를 확인해 보는 과정이다.

x_t 는 꽃잎이 떨어지는 동영상에서 t시간에서의 영상 내의 꽃잎의 위치와 속도이고, x_{t-1} 은 t-1시간에서의 영상 내의 꽃잎의 위치와 속도가 된다. 꽃잎이 낙하하는 속도는 자유 낙하이므로 아래와 같이 나타낼 수 있다. 즉,

$$x_t = \sqrt{2gh_t} \quad (3)$$

로 표현할 수 있다. 이 식은 꽃잎이 자유 낙하할 때 중력의 영향만 받는다는 것을 나타낸다. 현실적으로는 바람의 영향이 존재하므로 바람의 영향

을 받는 경우에 위치의 불확실성이 나타난다. 따라서 꽃잎이 낙하하는 경우의 운동학은 아래와 같이 조건부 확률 $p(x_t|x_{t-1})$ 로서 식 (4)와 같이 근사화할 수 있다.

$$p(x_t|x_{t-1}) = x_{t-1} + \sqrt{2g(x_t - x_{t-1})} + Q(t) \quad (4)$$

이때의 $Q(t)$ 는 바람이나 꽃잎의 상태에 의한 위치추정의 불확실성을 나타내며 가우시안분포로 가정하였다. 본 논문에서는 normal distribution을 나타내는 평균과 표준편차는 실험에 의해 계산한다. 따라서 샘플은 위의 확률로부터 샘플링 된다. 즉,

$$x_t^{[m]} \sim p(x_t|x_{t-1}) \quad (5)$$

따라서 $x_t = \{x_t^{[1]}, x_t^{[2]}, \dots, x_t^{[m]}\}$ 으로 파티클이 생성된다.

추정과정(estimation process) : t시간에서의 센서 측정치 분포로부터 관측 분포를 구한다. 이것이 시스템의 likelihood이다. 예측 과정에서 샘플링 된 particle에 대해 likelihood의 확률로 샘플링 하여 사후 추정치를 계산한다. 꽃잎의 경우, 측정은 영상의 촬영에 의해 이루어진다. 꽃잎의 위치와 속도는 영상 내의 위치와 위치를 미분한 속도 값으로 얻을 수 있다. 따라서 추정치는 다음과 같은 분포를 가진다고 가정한다.

$$p(z_t|x_t) = x_t + R(t) = \text{sensor reading} \quad (6)$$

여기서 sensor reading은 측정치인 t시각에서의 영상 내 꽃잎의 위치를 나타낸다. 식 (6)에서 표현된 것처럼 이 측정치에는 오차가 포함이 되어 있어서 실제의 상태인 x_t 는 R(t)에 의해 가려져 있다고 본다. 여기서 R(t)는 measurement model로서 영상센서가 갖고 있는 센서 고유의 위치 불확실성을 나타낸다. 이 값은 센서의 고유한 값이므로 센서마다 다르다. 이 불확실성도 역시 실험에 의해 측정한다. Measurement model인 센서의 해상도에 의한 위치 측정 불확실성은 아래와 같이 표현할 수 있다.

$$R(t) = N(0, \Sigma_t) \quad (7)$$

식 (7)은 측정값에 중심을 둔 zero centered 가우스 분포 형태를 띈다. 이 값은 영상 장치의 해상도와 관련이 있다. 본 논문에서는 R(t) 역시 실제 실험을 통해서 구한다. 식 (5)에서 샘플링 된 파티클은 추정 과정에서 식 (6)으로 표현되는 measurement

modeling을 이용하여 weight factor를 update한다. 즉,

$$w_t^{[m]} \sim R(t) = N(x_t^{[m]}; 0, \Sigma_t) \quad (8)$$

식 (8)은 일반적으로 가우시안 분포확률을 사용하므로 이를 사용하여 표현하면 아래와 같다.

$$w_t^{[m]} \sim \frac{1}{\sqrt{2\pi\Sigma}} \exp\left(-\frac{(z_t - x_t^{[m]})^2}{2\Sigma}\right) \quad (9)$$

식 (9)에서 샘플링 된 weight factor는 전체 파티클의 weight factor의 총합이 1이 되도록 아래 식 (10)과 같이 normalize 된다.

$$w_t^{[m]} = \frac{w_t^{[m]}}{\sum_{i=1}^m w_t^{[i]}} \quad (10)$$

식 (10)에 의해 최종적으로 계산된 weight factor는 전체 합이 1인 중요도 비율을 나타낸다. 즉, 파티클이 가지는 중요도를 나타내는 index가 된다.

식 (5)와 식 (10)에서 얻은 파티클 예측치와 각 파티클의 weight factor는 하나의 순서쌍을 이룬다. 즉,

$$\mathbf{x}_t = \left\{ (x_t^{[1]}, w_t^{[1]}), (x_t^{[2]}, w_t^{[2]}), \dots, (x_t^{[m]}, w_t^{[m]}) \right\} \quad (11)$$

식 (11)로 표현된 순서쌍 집합 \mathbf{x}_t 는 시각 t에서의 상태 x_t 를 m개의 particle과 weight factor로 표현한 것이다. 이와 같이 Particle filtering 기법은 분포가 Gaussian distribution이라는 가정이 필요 없는 일반적인 분포에도 잘 표현하는 능력이 있으므로 비선형 시스템(nonlinear system)의 dynamics를 잘 표현할 수 있다. 다만, 분포를 표현하는 particle의 수 m이 매우 크면 실제와 닮은 표현능력이 있는 반면 계산량이 많아지는 단점이 있고, m의 수를 줄이면 실제의 분포를 표현하는 능력이 떨어지는 단점이 존재한다.

Resampling과정: Particle filter의 문제 가운데 하나가 몇 번의 반복실행 후에는 가중치가 한 두 개의 중심 샘플에 집중적으로 작용하여 중심 샘플 이외에는 가중치가 무시할 수 있는 수준으로 낮아지는 이른바 샘플의 퇴화(degeneracy)문제가 발생한다는 점이다. 퇴화를 막기 위하여 resampling을 실시한다. Particle의 weight factor를 조사하여 weight factor가 Gaussian distribution으로부터 randomly sampling한 probability 보다 큰 weight factor를 갖는 - 다시 말하면, 더 높은 weight factor를 갖는 -

particle만 선별하여 새로운 particle로 한다. 이것을 Importance based resampling with replacement라고 하는데 이를 통해서 퇴화(degeneracy)의 문제를 개선할 수 있다. 이를 Pseudo-code로 쓰면 다음과 같이 된다.

$$x_t = \{x_t^{[i]}\} \quad \text{where } i = \arg_i \left(\text{rnd}(0,1) \leq \sum_{i=1}^m w_t^{[i]} \right) \quad (12)$$

식 (12)는 cumulative probability of weight factor가 random sampling 값보다도 큰 경우의 $x_t^{[i]}$ 를 새로운 particle로 대체한다는 전략이다. 즉, weight factor에 따른 확률로 상태를 샘플링 함으로써 weight factor의 집중화 현상에 의한 퇴화(degeneracy)를 방지하는 데 목적이 있다.

중요 가중치의 변화는 시간에 따라 증가하므로 퇴화 문제를 피하기는 어렵다. 이 때문에 알고리즘의 퇴화가 이루어졌는지를 알아볼 수 있는 기준 척도로서 효율적인 샘플 크기를 의미하는 N을 사용한다. 여기서는 참고문헌 (13)에 제시된 유효 샘플 크기에 관한 수식을 사용한다.

$$N_{eff} = \frac{1}{\sum (w_t^{[i]})^2} \quad (13)$$

3. 통계적 모델링과 실험

본 논문에서는 벚꽃잎이 낙하할 때 벚꽃잎의 형상에 의한 영향, 그때 그때 달라지는 바람의 영향에 의하여 불규칙 운동을 하는 것을 영상으로 취득하여 위치와 속도를 계측하였다. 이 자료를 근거로 하여 위에 설계된 Particle filter를 사용하여 추적실험을 수행하였다.

3.1 실험 방법

낙하실험은 Fig. 1에서 보는 바와 같이 야외에서 실시되었으며 벚꽃잎을 표준 크기의 각각 모양이 다른 것을 10개씩 따서 10회 반복 시험하였다. 꽃잎이 낙하하는데 필요한 시간은 1.0~1.4초의 범위에서 형성되었다. 낙하 꽃잎의 계측은 영상의 촬영과 분석을 이용하여 수행되었다. 스마트폰 카메라를 고정대에 고정시킨 다음 꽃잎을 낙하시켜서 낙하동영상을 촬영하였다. 그 후 저장된 동영상을 PC에 옮기고 Windows의 Moviemaker S/W를 이용하여 0.03초의 time step으로 나누어서 정지영상으로 만들었다. 이렇게 만들어진 정지영상 내의 꽃잎의 위치를 Windows의 그림 판 S/W를 이용하여

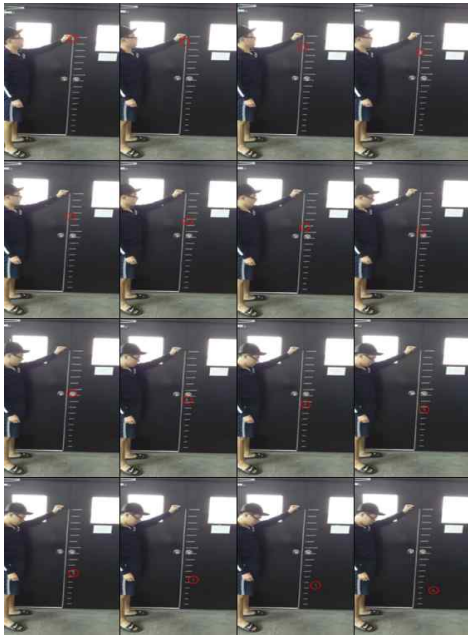


Fig. 1 Experimental view of falling down petals of cherry blossom. The petal falling in the air is shown in the red circle in the Figure. Template was used to calibrate the position of the petal in the air

픽셀 단위로 계측하였다. 실제 치수를 확인하기 위하여 카메라 보정이 필요하였는데 이 실험에서는 Fig. 1에서 보는 바와 같이 일정한 간격을 가진 눈금을 새긴 Template을 만들어서 Template과 함께 촬영하였다.

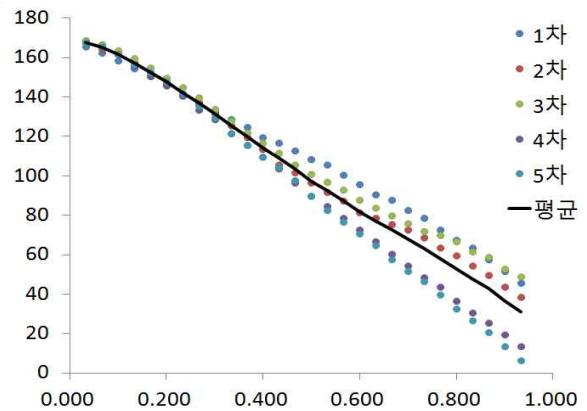
이와 같은 실험을 10회 동일한 내용으로 실시하고 결과로 나온 위치 데이터를 횡수로 평균하여 각 단계에서의 평균 속도와 표준 편차를 구하였다.

3.2 시스템 모델링

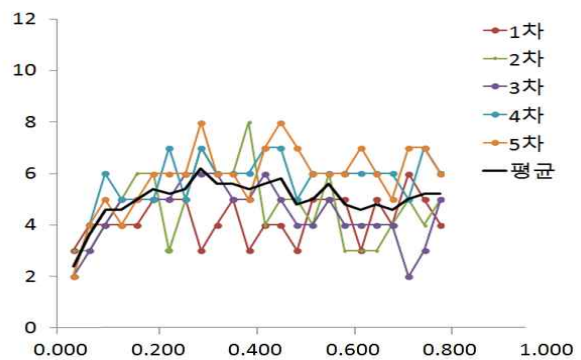
이와 같은 실험을 바람의 상태가 조금씩 다를 때를 기준으로 하여 5회(epoch) 실시하였다. 따라서 총 실험 횟수는 10회/epoch x 5 epoch=50회의 실험을 수행하였다.

위의 측정으로부터 작성된 실험 시스템 모델은 다음 Fig. 2 ~ Fig. 4와 같다. Fig. 2(a)는 꽃잎이 자유 낙하할 때의 위치에 대한 실험 결과이다. Fig. 2(b)는 위치 데이터를 활용하여 구한 속도의 실험 결과 그래프이다.

Fig. 2(b)에서 각 회차마다 바람의 세기가 달라져서 꽃잎의 낙하 평균속도가 회수에 따라 많이 차이가 남을 볼 수 있다. Fig. 3은 자유 낙하하는 꽃잎의 속도 히스토그램이다. 속도 히스토그램에서 볼 수 있는 바와 같이 전체적으로 분포는 가우스 분포를 닮아 있으므로 여기서는 가우스 분포로



(a) Position variation with time



(b) Velocity variation with time

Fig. 2 Distance and velocity graph for the falling petal in the air

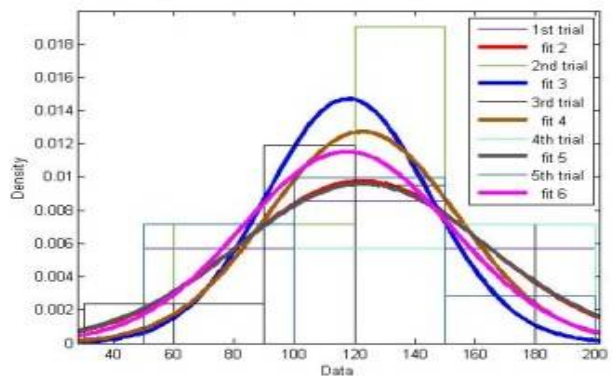


Fig. 3 Velocity distribution of falling petal

근사하였다.

Fig. 4는 실험 회차에 따른 분포의 변동성을 막대 그래프로 표현한 것을 나타낸다.

3.3 Particle filter에 의한 궤적추정

여기서 운동을 기술하는 데 사용된 상태변수는 위치와 속도이다. 즉, $\mathbf{x} = (x, v)$ 이며 x 는 꽃잎의 위치를 나타내며, v 는 꽃잎의 속도를 나타낸다. 상태

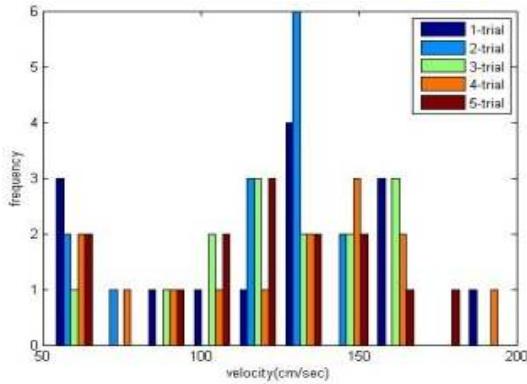


Fig. 4 Velocity distribution of falling petal

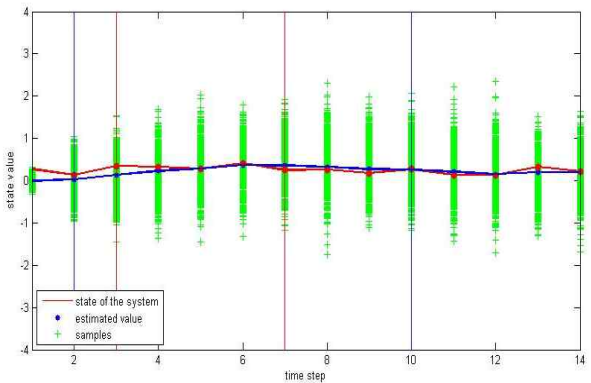
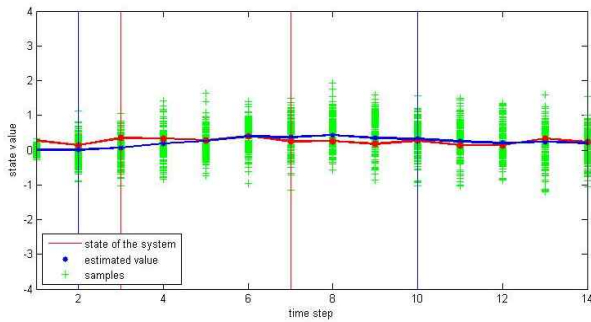


Fig. 5 Estimated vs. true value and error values are normalized. Upper picture: 150 particles, Lower one: 1000 particles)

변수를 사용하여 2절에서 설계한 Particle filter 알고리즘을 적용하였다. Particle filter의 적용 시에 particle의 수는 150~1000개로 변동시키면서 실험하였다. 이때 시스템의 표준 편차는 2%, 측정의 표준 편차는 10% 정도로 하였는데 이는 영상의 측정 정밀도 및 측정의 불확실성을 포함하기 때문이다.

4. 결과 및 고찰

Fig. 5는 꽃잎이 떨어지는 속도의 불규칙성에 대하여 Particle filter가 예측한 다음 추적하는 것을

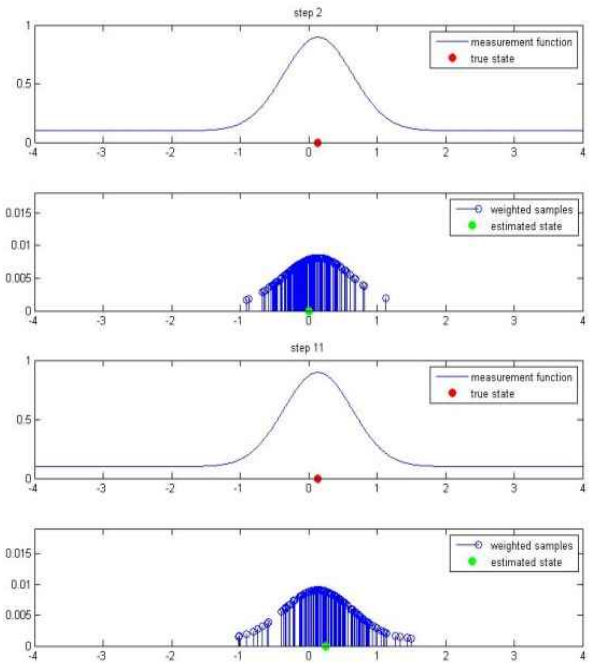


Fig. 6 Weighted samples and estimated samples(particles 150)

나타낸 것이다. Fig. 5에서 알 수 있는 바와 같이 불규칙한 속도 변동에 대해서 상당히 빠르게 추적하는 것을 볼 수 있다.

Fig. 5에서 적색 라인은 꽃잎의 위치를 나타낸다. 청색 점은 Particle filter가 예측한 꽃잎의 추정 위치를 나타낸다. 초록색으로 아래 위로 길게 나 있는 것은 그 위치에서의 particle의 분포이다. 초록색은 그 샘플링 순간의 꽃잎의 상태변수의 불확실성을 나타낸다.

Particle filter의 particle의 수는 150개~1000개로 변동시키면서 실험하였다. Fig. 5에 나타난 바와 같이 particle의 수가 150인 경우(Upper part)에 비해 particle의 수가 1000인 경우(Lower part)에 step이 진행할수록 오차가 더 빠르게 줄어드는 것을 알 수 있다.

Particle의 수는 많을수록 실제 분포에 가깝게 나타나나 너무 많으면 계산 량이 많아져서 속도가 느려진다. 실시간 추적에서는 계산 량이 많아져서 속도가 느려지면 추적할 수 없게 되므로 이를 적절한 수준으로 하지 않으면 안 되는 문제가 있다.

Fig. 6은 14step 가운데 2step과 11step에서의 실제 꽃잎의 위치 그리고 꽃잎의 추정치와 particle의 분포 상태를 보여준다. 2step의 위 그림은 추정치의 분포를 나타내며 적색의 원형마크가 추정치의 참값을 나타낸다. 이 위치는 실험에서 2step에서의 영상을 통하여 계측된 위치이다. 아래 그림은 이

측정치 분포가 가중치로서 반영된 샘플들의 분포를 보여주고 있다. 2 step에서 예측 치는 약간 왼쪽으로 위치해 있는데 particle의 분포밀도도 측정치 쪽으로 약간 기울어져 나타나 있다.

11step에서의 그림에서는 측정치의 방향으로 예측 치가 좀더 이동한 모양을 나타낸다. 초기의 측정치와 예측치사이의 오차는 단계가 지날수록 줄어드는 것을 Fig. 5에서도 확인할 수 있다.

5. 결론

본 연구에서는 마이크로-생물시스템(micro-bio system)이 공기 중에서 자유롭게 운동할 때 이를 예측하고 추적하는 모델을 제시하고 이를 파티클 필터 이론을 사용하여 실제 실험결과와 비교하였다. 이를 통하여

(1) 마이크로-생물체와 같은 자유운동이 가능한 시스템의 운동경로를 수학적 모델을 통하여 통계적으로 관측하였다.

(2) 마이크로-생물체의 운동을 제시된 수학적 모델을 파티클 필터이론에 적용하여 오차범위 내에서 추적이 가능함을 보였다.

(3) 마이크로-생물체의 운동을 이용한 시스템의 제작에 필요한 지식을 획득할 수 있었다.

참고문헌

(References)

- (1) Pines, D. J. and Bohorquez, F., 2006, "Challenges Facing Future Micro-air-vehicle Development," *Journal of Aircraft*, Vol. 43, No. 2, pp. 290~305.
- (2) Murphy, J. T. and Hu, H., 2010, "An Experimental Study of a Bio-inspired Corrugated Airfoil for Micro Air Vehicle Applications," *Experiments in Fluids*, Vol. 49, No. 2, pp. 531~546.
- (3) Xia, F. and Jiang, L., 2008, "Bio-inspired, Smart, Multiscale Interfacial Materials," *Advanced Materials*, Vol. 20, No. 15, pp. 2842~2858.
- (4) KALMAN, R., A New Approach to Linear Filtering and Prediction Problems1.
- (5) Ljung, L., 1979, "Asymptotic Behavior of the Extended Kalman Filter as a Parameter Estimator for Linear Systems," *IEEE Transactions on Automatic Control*, Vol. 24, No. 1, pp. 36~50.
- (6) Julier, S. J. and Uhlmann, J. K., 1997, "In New Extension of the Kalman Filter to Nonlinear Systems," *AeroSense'97, International Society for Optics and Photonics*, pp. 182~193.
- (7) Marins, J. L., Yun, X., Bachmann, E. R., McGhee, R. B. and Zyda, M. J., 2001, "In An Extended Kalman Filter for Quaternion-Based Orientation Estimation Using MARG Sensors," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 4, pp. 2003~2011.
- (8) St-Pierre, M. and Gingras, D., 2004, "In Comparison Between the Unscented Kalman Filter and the Extended Kalman Filter for the Position Estimation Module of an Integrated Navigation Information System," *Intelligent Vehicles Symposium*, pp. 831~835.
- (9) Nummiaro, K., Koller-Meier, E. and Gool, L. V., 2003, "An Adaptive Color-based Particle Filter," *Image and Vision Computing*, Vol. 2, No. 1, pp. 99~110.
- (10) Khan, Z., Balch, T. and Dellaert, F., 2004, "An MCMC-based Particle Filter for Tracking Multiple Interacting Targets," *In Computer Vision-ECCV Springer*, Vol. 2, pp. 279~290.
- (11) Chang, C. and Ansari, R., 2005, "Kernel Particle Filter for Visual Tracking," *IEEE Signal processing letters*, Vol. 12, No. 3, pp. 242~245.
- (12) Nummiaro, K., Koller-Meier, E. and Gool, L. V., 2002, "Object Tracking with an Adaptive Color-based Particle Filter," *In Pattern Recognition, Springer*, pp. 353~360.
- (13) Arulampalam, M. S., Maskell, S., Gordon, N. and Clapp, T., 2002, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, pp. 174~188.
- (14) Dellaert, F.; Fox, D., Burgard, W. and Thrun, S., 1999, "In Monte Carlo Localization for Mobile Robots," *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1322~1328.
- (15) Doucet, A., Godsill, S. and Andrieu, C., 2000, "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering," *Statistics and Computing*, Vol. 10, No. 3, pp. 197~208.