

## 순차적인 몬테카를로 필터를 사용한 차량 추적

# Vehicle Tracking using Sequential Monte Carlo Filter

\* 이원주, \* 윤창용, \*\* 김은태, \*\*\* 박민용  
 Won-ju Lee

**Abstract** - In a visual driver-assistance system, separating moving objects from fixed objects are an important problem to maintain multiple hypothesis for the state. Color and edge-based tracker can often be "distracted" causing them to track the wrong object. Many researchers have dealt with this problem by using multiple features, as it is unlikely that all will be distracted at the same time. In this paper, we improve the accuracy and robustness of real-time tracking by combining a color histogram feature with a brightness of Optical Flow-based feature under a Sequential Monte Carlo framework. And it is also excepted from Tracking as time goes on, reducing density by Adaptive Particles Number in case of the fixed object. This new framework makes two main contributions. The one is about the prediction framework which separating moving objects from fixed objects and the other is about measurement framework to get a information from the visual data under a partial occlusion.

**Key Words** :Color-based tracking, Monte Carlo Filter, Multiple hypothesis, Particle Filter, Optical Flow.

### 1. 장 Introduction

한 픽셀에 담겨있는 부분적인 정보에서 범위적 분포에 관한 정보까지 넓게 연구되고 있는데 정보란 여러 모델에 의한 색상이나 모양<sup>[1]</sup>, 흐름<sup>[2]</sup>의 유사성 등이 있다.

본 논문에서는 노이즈에 강인한 성격을 가진 Tracking 방법 중 Sequential Monte Carlo (SMC) Filter를 사용한다. 흔히 비교되는 Kalman Filter와는 비선형일 경우 선형화 단계가 필요 없으며 노이즈의 분산이 커짐에도 영향을 덜 받는다<sup>[3]</sup>. Brightness의 분포를 사용하는 Optical Flow와 색상의 분포를 접목하여 Measurement의 성능을 향상시킬 것이다. 반면에 Prediction과정은 도로 혹은 표지판과 같이 지면에 고정된 물체가 자동차의 움직임에 따라 영상에서 얻어지는 위치를 미분방정식으로 얻는다. 하지만 영상정보만으로 위와 같은 처리가 힘들기 때문에 Radar를 통해 Hypothesis Generation (HG)을 얻으며 물체에 대한 거리를 얻는다고 가정한다. 이렇게 얻은 물체와의 거리를 통해 예상된 정보를 얻는다.

### 2. 장 Measurement

#### 2.1 절 밝기정보의 유사성

우리는 프레임이 변해도 픽셀의 Brightness가 변하지 않는다는 전제로 Constraints를 유도해나갈 것이다. 먼저  $t$ 의 시간에  $x, y$  위치에 있는 픽셀의 Brightness를  $E(x, y, t)$ 라 정의한다. 그러면 가정에 의해 아래 식으로 표현된다.

$$\frac{dE}{dt} = 0 \quad (1)$$

위 식에 Chain rule을 사용하면,

$$\begin{aligned} E(x, y, t) &= E(x + dx, y + dy, t + dt) \\ &= E(x, y, t) + \frac{\partial E}{\partial x} dx + \frac{\partial E}{\partial y} dy + \frac{\partial E}{\partial t} dt + (\dots) \\ &\approx E(x, y, t) + \frac{\partial E}{\partial x} dx + \frac{\partial E}{\partial y} dy + \frac{\partial E}{\partial t} dt \\ 0 &= E_x u + E_y v + E_t \end{aligned} \quad (2)$$

$$\text{여기서, } u = \frac{dx}{dt}, v = \frac{dy}{dt}, E_x = \frac{\partial E}{\partial x}, E_y = \frac{\partial E}{\partial y}, E_t = \frac{\partial E}{\partial t} \quad (3)$$

첫 번째 Constraints 식 (2)는 Linear equation이므로 원하는 픽셀 움직임인  $u$ 와  $v$ 를 찾으려면 정보가 더 요구된다. 따라서 두 번째 Constraints를 얻기 위해 Smoothness (평활화) 방법을 사용한다. 이 방법은 Flow velocity의 미분 값의 제곱을 최소화하는 식 (4)로 얻을 수 있다.

$$\left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial u}{\partial y} \right)^2 + \left( \frac{\partial v}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial y} \right)^2 \quad (4)$$

이제 남은 건 위 두 Constrains (2), (4)를 최소화하는 것이다. Constraints 식 (2), (4)는 실제 영상에서는 가정처럼 0이 되지 않으므로 얻어진 값이 Error가 된다. 따라서 Total Error  $\Psi$ 는 식 (6)와 같이 표현된다.

$$\Psi^2 = \iint (\alpha^2 \Psi_c^2 + \Psi_b^2) dx dy \quad (5)$$

여기서,

$$\begin{aligned} \Psi_b &= E_x u + E_y v + E_t \\ \Psi_c &= \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial u}{\partial y} \right)^2 + \left( \frac{\partial v}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial y} \right)^2 \end{aligned} \quad (6)$$

식(6)의  $\alpha^2$ 는 Brightness의 미분 값이 적을 때 사용하며  $\Psi_c$ 의 노이즈에 추정하는 값을 가져야 한다.

Local average  $\bar{u}$ 와  $\bar{v}$ 의 거리는 Total Error와 비례한다. 따

지자 소개

- \* 연세대학교 전기전자공학과
- \*\* 연세대학교 전기전자공학과 조교수
- \*\*\* 연세대학교 전기전자공학과 정교수

라서 SMC Filter의 Measurement 단계에 적용하면  $X_t^i$ 는 Feature Extraction으로 뽑은 Features와 노이즈를 포함한 값이며 곧 SMC Filter의 Particles이 된다. Multiple Probability  $p(z_t | x_t)$ 은 식 (10)으로 표현될 수 있다.

$$p(Z_{B,t}^i | X_t^i) \propto (2\pi\sigma^2)^{-1} \exp\left\{-\frac{(Z_{B,t}^i - X_t^i)^2}{2\sigma^2}\right\}$$

$$X_t^i = (x_t^{i,m}, y_t^{i,m})$$

$$Z_{B,t}^i = (x_t^i + \delta x_t^i, y_t^i + \delta y_t^i) \quad (7)$$

식 (10)에서  $t$ 는 시간이며  $m$ 은 총  $M$ 개의 Particle Index이다. 또한  $i$ 는  $t$ 시간에 측정된 총  $n$ 개의 결과 Index이다.  $\sigma^2$ 는 측정된 결과에 포함된 노이즈의 분산이다.  $Z_{B,t}^i$ 는  $t$ 시간에 측정된  $i$ 개의 Optical Flow 측정된 결과이며 Gaussian 분포를 사용하여 표현하였다.

## 2.2 절 색상정보의 유사성

식 (11)는 Pérez [5] 의해서 얻은 색상 정보의 유사성이다. 시간에 따른 ROI  $q_t^*$ 의 크기를 식 (15)에서 예측된 크기  $k$ 로 변화시킬 것이다.

$$p(Z_{C,t}^i | X_t^i) \propto \exp\left\{-\gamma_c D^2[q_t^*, q_t^i(x_t^{i,m}, y_t^{i,m})]\right\}$$

$$X_t^i = (x_t^{i,m}, y_t^{i,m}) \quad (8)$$

$\gamma_c$ 는 실험적으로 얻은 상수이며  $q_t^*$ 는 첫 프레임에서 HG를 통해 얻은 ROI이며 시간에 따라 크기가 바뀐다.  $q_t^i(x_t^{i,m}, y_t^{i,m})$ 가 현재 프레임에 대응되는 영역이다. 여기서  $D$ 는 Bhattacharyya similarity coefficient로 얻어진다.  $n$ 은 Histogram의 Bin number이다.

$$D[q_t^*, q_t^i(x_t^{i,m}, y_t^{i,m})] = \left[1 - \sum_{n=1}^N \sqrt{q_t^{i,m}(n)q_t^*(n)}\right]^{1/2} \quad (9)$$

부분적인 Occlusion의 발생시점을 찾는 방법으로는 HG에 선택된 Radar로 두 물체의 거리정보를 얻지 못할 때를 발생시점으로 보며, 다시 물체의 거리정보가 둘로 나뉘었을 때를 종료시점으로 본다.

$$w_t^i = k_t^i p(Z_{B,t}^i | X_t^i) + (1 - k_t^i) p(Z_{C,t}^i | X_t^i) \quad (10)$$

식 (13)에서  $k_t^i$ 는 실험적으로 얻은 Optical Flow와 색상분포의 상대적인 가중치이며 부분적인 Occlusion이 발생한 경우 색상분포의 가중치가 높아지며 일반적인 상황에서는 Optical Flow의 가중치를 높여 광량에 따른 Error를 줄인다.

## 3. 장 Prediction

### 3.1 절 고정된 물체의 예상 움직임

그림 1.는 현재 프레임인 #1 가지고 자동차가  $v\Delta t$ 만큼 이동했을 때 고정된 점  $B$ 가 움직일 예상 위치이다.  $f$ 는 카메라의 focal length이고  $v$ 는 자동차의 속도이며  $\Delta t$ 는 움직인 시간이다. 영상에서 얻어지는 데이터를 Image Plane #1, #2로 놓았다. 고정된 물체는 자동차가 위치  $A$ 에서  $C$ 로 이동할 사이에 영상에서의 위치가 이동하였다.

삼각함수와 비례식으로 아래와 같은 방정식을 유도한다.  $\theta_1$ 은 현재 프레임 #1에서의 측정된 위치로 얻는다.

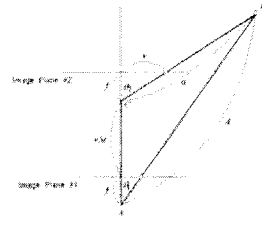


그림 1. 고정된 물체의 Prediction

$$a^2 = (v\Delta t + f)^2 + d^2 - 2d(v\Delta t + f)\cos\theta_1$$

$$\cos C = \frac{a^2 + (v\Delta t + f)^2 - d^2}{2a(v\Delta t + f)}$$

$$\theta_2 = 180^\circ - \text{각도} C$$

$$x = f \tan \theta_2 \quad (11)$$

이렇게 얻은 식 (14)과 움직이는 물체의 예상인 식 (15)를 통해 비교하여 고정된 물체임을 판단한다. 유사성 검사는 3.3 절 ATN에서 자세히 다룬다.

### 3.2 절 움직이는 물체의 예상 움직임

움직이는 물체는 어디로 갈지 예상할 수 없다. 일반적인 방법으로 몇 차례의 프레임을 보고 물체가 현재까지 움직인 거리를 통해 시간에 따른 미분 값으로 움직일 위치를 예측한다. Baoxin Li [1]처럼 아래 같이 얻는다.

$$X = f(\theta, X_0)$$

$$= f(\theta_0, X_0) + J_\theta(\theta - \theta_0) + o(\cdot)$$

$$\approx X_0 + J_\theta(\theta - \theta_0) \quad (12)$$

식 (15)에서  $o(\cdot)$ 는 higher order terms이며  $J_\theta$ 는 Jacobian matrix이다. 시간에 따른 변화가 크지 않다는 전제 하에 higher order terms은 사라질 수 있다.  $\theta$ 는 시작 점  $X_0$ 에서  $X$ 로 변환하는 transformation이다.  $X$ 는 Optical Flow의 결과 좌표로 알 수 있으며  $J_\theta$ 는 움직인 거리의 미분 값이다.

### 3.3 절 Adaptive Particles Number

Multiple Target에서 고정된 물체와 움직이는 물체의 Density를 변화시키기 위해 해당하는 Object의 Particle Number를 조절한다.

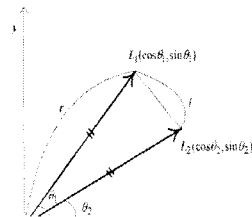


그림 2. 고정된 물체와 움직이는 물체의 방향 비교

또한 실제 움직인 방향을 Measurement를 통해 얻는다. 이렇게 얻은 두 이동 방향을 식 (16)으로 비교할 수 있다.

$$L = \sqrt{(\cos\theta_1 - \cos\theta_2)^2 + (\sin\theta_1 - \sin\theta_2)^2} \quad (13)$$

식 (16)과 그림 2.의  $\theta_1$ 은 고정된 물체의 예상 방향이며  $\theta_2$ 는 실제 움직인 방향이다. 따라서  $r$ 이 1로 가정하면  $L$ 의 최

대값이 2.이며  $L$ 은 0~2사이의 값을 가지며 실험적으로  $-15^\circ \sim +15^\circ$ 인 값인 고정된 물체의 Threshold를 0.26으로 움직이는 물체의 Threshold를 2-0.26으로 정하였다.

### 3.4 절 자동차의 예상 크기

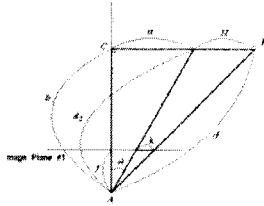


그림 3. 거리를 통해 얻은 자동차의 크기

$$\begin{aligned}
 \text{각도 } B &= 90^\circ - \theta \\
 a &= d \cdot \cos B - H \\
 b^2 &= (a+H)^2 + d^2 - 2d(a+H)\cos B \\
 f : b &= k : H \\
 k &= \frac{fH}{b} \quad (14)
 \end{aligned}$$

그림 3.에서  $f$ 는 focal length이고  $d$ 는 Radar로 얻은 자동차와의 거리이다. 현재 프레임인 #1에서의 자동차 크기  $k$ 를 식 (17)를 통해 추측한다.  $H$ 는 일반적인 자동차의 크기를 알고 있다고 가정한다.

## 4. 장 Experiment

그림 4.는 3개의 Objects를  $x$ 과  $y$ 좌표만을 SMC Filter로 Tracking한 결과이다.

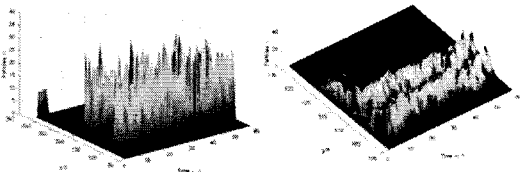


그림 4. 3개의 Objects를 SMC Filter의 Particles 분포 (왼쪽부터 고정된 물체 1개, 움직이는 물체 2개  
1 row:  $x$ 축 Particles 분포, 2 row:  $y$ 축 Particles 분포)

고정된 물체의 경우 예상된 움직임과의 비교를 통해 Particles의 수를 줄여 상대적으로 낮은 Density를 보인다. 오른쪽 2개의 Object는 Noise가 포함되어 있어도 고정된 물체보다 상대적으로 높은 Density를 띄어 구분할 수 있다. 실험에서는 고정된 물체의 Threshold를 0.26으로  $+15^\circ$ 로 정한 그래프이다. 특히 2 row의  $y$ 좌표는 고정된 물체의 낮은 Density로 다른 Objects의 Tracking 결과와 구분 짓기 어렵다.

그림 5.는 3개의 Object의 estimation이다. 그림 5.의 오른쪽 1번째 그래프가 고정된 물체의 estimation이다. 실선인 True값의 측정값이 계속 있음에도 estimation은 0으로 감을 보여준다. 다시 말해 결국 고정된 물체는 움직이는 물체보다 상대적으로 Density가 낮아 Tracking할 물체에서 제외된다.

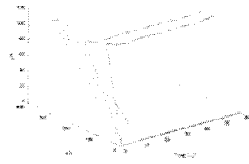


그림 5. 3개의 Object의 estimation 결과 (straight line: True, dot line: estimation)

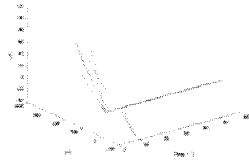


그림 6. 3개의 Object의 estimation error

그림 6.는 estimation error로 시간이 지남에 따라 결국 0에 수렴함을 보여준다. 영상에서의 True값을 직접 얻기 어려우므로 여기서 사용된 True값은 노이즈가 포함된 Measurement이다. 하지만 SMC Filter의 효과로 estimation은 측정 노이즈의 범위로 True값을 따라가게 된다.

## 5. 장 결론

본 논문에서는 SMC Filter를 사용한 Multiple Targets의 Tracking 문제를 다루고 있다<sup>[1]</sup>. Radar를 이용한 Hypothesis Generation 후보들로부터 이진 프레임 정보로 현재의 위치를 예상할 수 있는 고정된 물체와 실제 움직임과의 유사성을 바탕으로 Particles Number를 Adaptive한다. 따라서 움직이는 물체의 Density를 높여 시간이 지남에 따라 고정된 물체는 제외되며 움직이는 물체만을 Tracking한다.

Measurement 단계로 Brightness기반의 Features를 사용한 Optical Flow<sup>[2]</sup>와 시간에 따라 크기가 변하는 ROI의 Color 색상분포로 부분적인 Occlusion에 대응함을 보였다. 두 Measurement는 빛과 Occlusion에 서로 보완적인 관계를 가지며 외란이나 노이즈에 강인한 SMC Filter의 사용으로 전체적인 영상의 흔들림에도 시간이 지남에 따라 True값을 추정해 나감을 보여주었다.<sup>[3]</sup>

## 참고문헌

- [1] B. Li and R. Chellappa, "Simultaneous Tracking and Verification via Sequential Posterior Estimation," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Vol. 2, pp. 110-117, Hilton Head, SC, USA, June 2000
- [2] B.K.P. Horn and B.G. Schunck, "Determining optical flow," *Artificial Intelligence*, Vol. 17, pp. 185-203, 1981
- [3] S. Thrun and W. Burgard and D. Fox, "*Probabilistic Robotics*," The MIT Press, pp. 90-109, 2005
- [4] R. E. Schapire, "A brief introduction to Boosting." In *Proc. the Sixteenth International Joint Conf. Artificial Intelligence*, pp. 1401-1405, 1999
- [5] P. Pérez, C. Hue, J. Vermaak, M. Gangnet, "Color-based probabilistic tracking," in *Proc. Eur. Conf. Computer Vision*, pp. 661-675, Copenhagen, Denmark, May 2002