Proceedings of the IEEE ICRA 2009
Workshop on People Detection and Tracking
Kobe, Japan, May 2009

Stream Field Based People Searching and Tracking Conditioned on SLAM

Kuo-Shih Tseng and Angela Chih-Wei Tang

***Abstract*—People searching and tracking (SAT) is a key technology for interactive robots since the tracked people are sheltered by environments frequently. For robots, it is a tracking problem given that the target is observable, but otherwise it is a searching problem. Traditional tracking algorithms may lead to divergent estimation of object position when moving objects are unobservable. Moreover, SAT conditioned on simultaneous localization and mapping (SLAM) is complex since it aims at estimating people position, robot position, and map under sensor uncertainty. Motivated by this, we propose a novel stream functions and Rao-Blackwellised particle filter based SAT algorithm in this paper. This laser based algorithm is conditioned on simultaneous localization and mapping (SLAMSAT) to search and track people. With this, the position of the targeted person sheltered by the environment can be successfully estimated by the virtual stream field in a mapped environment. Our experimental results show that this algorithm can search and track people effectively.**

I. Introduction

In a dynamic environment, the robot navigation problem becomes interactive and it includes leading, following, intercepting, and obstacle avoiding. For most applications, a robot should be capable of tracking, following, self-localization, and obstacle avoidance in an unknown environment. Most tracking algorithms aim at correctly estimating the position, velocity, and acceleration of moving objects based on the past and sensor measurement [1]. Object tracking can be realized with Kalman filter (KF) with constant velocity model and/or constant acceleration model [2]. With particle filter (PF), objects with nonlinear states, non-Gaussian probability distribution, and multi-hypotheses are tracked with higher accuracy although the price is its high computational complexity. SLAMMOT uses scan matching and EKF with laser range finders to simultaneously estimate robot position, map, and object state [3]. The conditional PF estimates people motion conditioned on the probability model of robot position with a previously mapped environment [4].

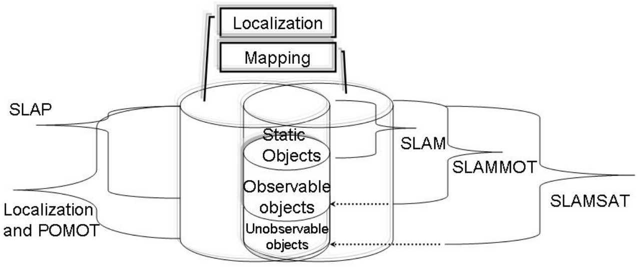


Fig.1. The relationship among SLAM, SLAP, SLAMMOT and SLAMSAT.

The tracking problem will turn into the searching problem if moving objects are unobservable. In [5], a map-based tracking algorithm using Rao-Blackwellised particle filter (RBPF) models the physical interaction between the ball and the wall even if the ball is unobservable. However, this algorithm can only track passive objects. Dynamic action spaces can be utilized to search and explore a moving object which goes toward one of the known destinations [6]. SAT techniques autonomously search and track objects using Bayesian estimator [7]. However, these algorithms cannot achieve simultaneous localization, tracking, and searching in an unknown environment. Motivated by this, a self-localization and partially observable moving object tracking (POMOT) algorithm is proposed in [8]. This algorithm is designed for a static and known environment. However, a robot has to localize itself, map, and search and track objects in most applications.

As shown in Fig. 1, objects can be static or dynamic. If the dynamic object is out of sight, it will be an unobservable object. Otherwise, it will be an observable object. Simultaneous localization and people tracking (SLAP) is to estimate robot and people position [4]. Localization and POMOT is to estimate robot and people position even if the person is unobservable [8]. SLAMMOT is to estimate robot, map and moving objects position [3].

In this paper, we propose a novel stream field based SAT algorithm conditioned on SLAM called SLAMSAT. SLAMSAT is to estimate robot, map and people position even if the person is unobservable. With stream field, we model interactions among goal position, updated environmental features, and people position. Traditional tracking algorithms deemed that objects move actively with velocity and acceleration generated themselves. But from the viewpoint of the stream field, object motion is passive due to the attraction and rejection forces resulted from the goal and environment. Based on this, we can still keep SAT the object

Manuscript received Feb. 2, 2009.

Kuo-Shih Tseng is with MSRL at the Industrial Technology Research Institute (ITRI), Hsinchu, Taiwan. (phone: 886-4-22779363; fax: 886-4-22782670; e-mail: seabookG@gmail.com).

Angela Chih-Wei Tang is with the Communication Engineering Department, National Central University, Jhongli, Taiwan. (e-mail: cwtang@ce.ncu.edu.tw).

position based on the virtual stream field. The remainder of the paper is organized as follows. Section II describes the stream field based motion model for the RBPF based SAT proposed in Section III. Section IV gives our AdaBoost based leg detection. In Section V, we present the proposed SAT algorithm which combines the stream field and RBPF conditioned on the EKF SLAM algorithm. The experimental results are given in Section VI. Finally, Section VII concludes this paper.

II. Motion Model using Stream Field

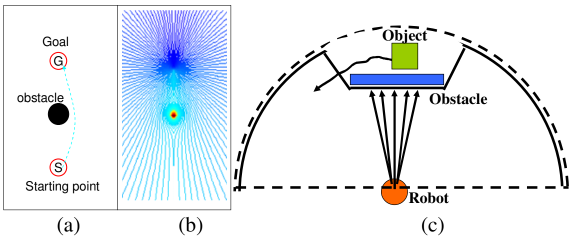


Fig.2. An example of a real environment and its virtual stream field. (a) Obstacle avoidance. (b) Stream field. (c) Real environment.

Complex potential is often employed to solve fluid mechanics and electromagnetism problems [9]. For an irrational and incompressible flow, there exists a complex potential consisting of the potential function $\phi(x, y)$ and stream function $\psi(x, y)$, where (x, y) is the 2-D coordinate. Although the complex potential has been studied quite extensively in motion planning and obstacle avoidance due to its high efficiency, it is seldom considered in object tracking. Motion model plays a key role in probability based tracking algorithms. To achieve on-line prediction of motion model according to the estimated map and virtual goal, we adopt stream field based motion model proposed in [8] for SAT algorithm in this paper. In the following, we give a brief description of this motion model. More details can be found in [8].

Stream field consists of a sink flow $\psi_{\sin k}(x, y)$ and a doublet flow $\psi_{doublet}(x, y)$ by

$$\begin{aligned}\psi(x, y) &= \psi_{\sin k}(x, y) + \psi_{doublet}(x, y)\\ &= -C\tan^{-1}\left(\frac{y-y_s}{x-x_s}\right) + C\tan^{-1}\left(\frac{\dfrac{a^2(y-y_d)}{(x-x_d)^2+(y-y_d)^2}+(y_d-y_s)}{\dfrac{a^2(x-x_d)}{(x-x_d)^2+(y-y_d)^2}+(x_d-x_s)}\right),\end{aligned} \tag{1}$$

where (x_s, y_s) is the center of sink, (x_d, y_d) is the center of doublet, a is the radius of doublet, and C is the constant proportion to the flow velocity. If the number of doublet flows is more than one, the stream field will be superposed by the sink flow and doublet flows. Details of stream fields can be found in [10]. Stream functions will be computed if the robot position, object goal, and obstacle positions are known. The object velocities are computed by the derivative of (1).

In typical tracking algorithms, the object position at time *t* is modeled by $\mathbf{x}_t = f(\mathbf{x}_{t-1}, \mathbf{v}_{t-1})$, where $\mathbf{v}_{t-1}$ is object motion at time *t*-1, and *f* is the object motion model. A robot cannot track a moving object successfully when the object is unobservable. By (1), we assume that objects will avoid a known obstacle (doublet) and move toward a virtual goal (sink) as in the stream field (Fig. 2(c)). Since the stream field constructs an active field where an object is moved inactively by attraction and rejection forces, we can predict object and goal position and construct search path using the known stream field.

A stream field is constructed by a virtual sink and a doublet resulted from a known environment, and then the object motion is predicted by

$$\mathbf{V}_{t-1} = \begin{bmatrix} v_{o,t-1} \\ v_{o,t-1} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial(\psi_{\sin k}(x_{o,t-1}, y_{o,t-1}) + \psi_{doublet}(x_{o,t-1}, y_{o,t-1}))}{\partial y_{o,t-1}} \\ -\dfrac{\partial(\psi_{\sin k}(x_{o,t-1}, y_{o,t-1}) + \psi_{doublet}(x_{o,t-1}, y_{o,t-1}))}{\partial x_{o,t-1}} \end{bmatrix}, \tag{2}$$

where $(x_{o,t-1}, y_{o,t-1})$ is the object position at time *t*-1. To estimate the position of virtual goal of an unobservable moving object, a probability based tracking algorithm with multi-hypothesis would work better than that with single hypothesis. Thus, Section III adopts RBPF to estimate *N* possible positions of an object goal.

III. RBPF based SAT using Stream Field Based Motion Model

To improve the accuracy of motion prediction in search case, we adopt stream field based motion model. However, the major problems of tracking with multi-hypothesis (e.g. PF) using this motion model are its heavy computational load and the requirement of precise probability distribution for prediction in the searching case. Since RBPF is capable of reducing the heavy load due to multi-hypotheses and approximating the probability distribution function more precisely [11], we adopt RBPF for SAT. Our RBPF based SAT algorithm using stream field based motion model is quite different from traditional ones where prediction will diverge if the moving object is unobservable.

Let the stream sample set $\mathbf{S}_k = \{\mathbf{s}_k^i, w_k^i \mid 1 \le i \le N\}$, and $\mathbf{s}_k^i = \langle O_k^i, G_k^i, D\rangle = \langle\langle O_{x,k}, O_{y,k}, \Sigma_{O,k}\rangle^i, \langle G_{\phi,k}, U_k\rangle^i, D\rangle$, where O_k^i is object state of the *i*th particle at time *k* including the mean $(O_{x,k}, O_{y,k})$ and covariance $\Sigma_{O,k}$. Object goal G_k^i consists of direction $G_{\phi,k}$ and intensity U_k. D is doublet position generated by map features. In our RBPF based SAT, PF estimates goal state G_k^i and KF estimates object state O_k^i.

We factorize stream distribution into goal set distribution, object set distribution, and stream set distribution at time *k*-1 as follows.

$$\begin{aligned} bel(\mathbf{S}_k) &= P(\mathbf{S}_{1:k} \mid z_{1:k}) = P(O_k^i, G_k^i, O_{1:k-1}^i, G_{1:k-1}^i, D \mid z_{1:k}) \\ &\overset{DBN}{=} \underbrace{P(G_k^i \mid O_k^i, G_{k-1}^i)}_{Goal \text{ set distribution}} \underbrace{P(O_k^i \mid O_{k-1}^i, D, G_{k-1}^i, z_k)}_{Objec\ set\ distribution} \times \\ &\underbrace{P(O_{k-1}^i, G_{k-1}^i, D \mid z_{k-1})}_{bel(\mathbf{S}_{k-1})} \end{aligned} \tag{3}$$

After sampling goal positions, object set distribution is divided into two cases. It will be the tracking case if the robot detects object successfully, otherwise it will be the searching case.

$$P(O_k^i \mid O_{1:k-1}^i, G_{1:k-1}^i, D, z_{1:k}) = \begin{cases} P(O_k^i \mid O_{1:k-1}^i, G_{1:k-1}^i, D, z_{1:k}^O), & \text{tracking case.} \\ P(O_k^i \mid O_{1:k-1}^i, G_{1:k-1}^i, D), & \text{searching case.} \end{cases} \tag{4}$$

For the tracking case, object set distribution is derived from Bayes theorem and updated by KF as follows.

$$P(O_k^i \mid O_{1:k-1}^i, G_{1:k-1}^i, D, z_{1:k}) \overset{Bayes}{=} \eta \underbrace{P(z_k \mid O_k^i)}_{object\ Correction} \underbrace{P(O_k^i \mid O_{1:k-1}^i, G_{1:k-1}^i, D, z_{1:k-1})}_{object \text{ prediction}} \tag{5}$$

Finally, we resample stream sample set after computing the particle weightings. Based on the predicted object goal position, the algorithm can keep predict the object position when the object features are occluded or are fragmentation.

IV. Leg Detection

Detection is a necessary stage prior to tracking. Algorithms of laser based people detection usually work well only if the scan data of one or two of human legs is available. In [14], AdaBoost based leg detection is proposed for people detection. However, such algorithm will fail if the scan data of either leg is not available due to the sheltering effects resulted from environments. Another possible solution is to detect based on motion. However, it is difficult to distinguish slower legs from static objects.

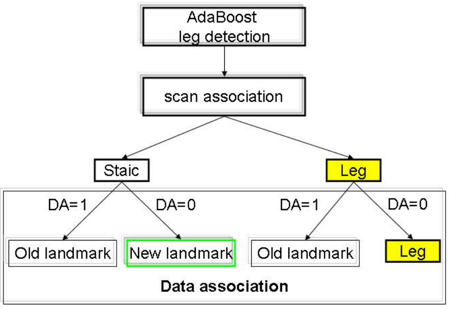


Fig.3. Decision flow of landmarks and leg.

In [15], the authors propose a multi-hypothesis leg-tracker for occlusion problem under known map. In SLAMSAT, a robot has to distinguish leg features (moving object) from static ones. Static features will be considered to be added into maps while leg features will be the candidates of tracked targets. The robot will search the dynamic feature based on the virtual stream field if there is not any dynamic feature detected.

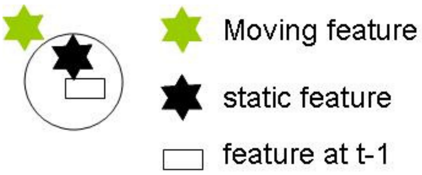


Fig.4. Thresholds for feature classification in scan association.

In this section, we design a leg detection algorithm which is composed of AdaBoost, scan association, and data association of old landmarks. The decision flow of this detection algorithm is shown in Fig. 3. After applying AdaBoost based detection proposed in [14], all features are divided into leg features and non-leg features. Then, scan association determines whether a feature is static or dynamic by comparing scanned features at time *t* with those at time *t*-1. Let the Euclidean distance of a feature position at time *t* and a feature position at time *t*-1 be *d*. Please note that the feature at time *t*-1 could be static or dynamic. As shown in Fig. 4, a feature will be classified into a static one if *d* is smaller than the radius of black circle. Otherwise, this feature will be classified as a dynamic one.

Data association distinguishes whether a feature is for mapping or tracking based on the decision of scan association and AdaBoost based leg detection. If the static feature determined by scan association is not associated with any old landmark, it will be deemed as a new landmark. If a static or dynamic feature is associated with an old landmark, it will be deemed as an old landmark. Otherwise, the leg feature is still deemed as the leg.

It is a simplified task to distinguish static features from dynamic features using data association with a known map. However, such task will become difficult if a robot must simultaneously explore new landmarks with unknown map and search unobservable people. For example, when the dynamic features are sheltered by environments and a new feature is observable, it is difficult to distinguish a new landmark from a dynamic feature. A possible solution for such problem is employing multi-hypotheses based estimators instead of a single-hypothesis based one.

V. The Proposed SAT Algorithm Conditioned on SLAM

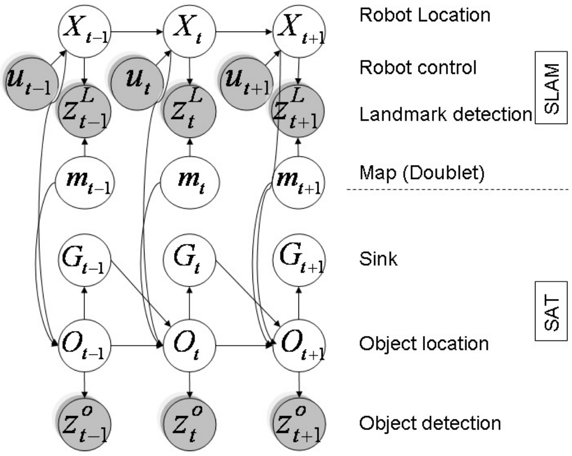


Fig. 5. Dynamic Bayesian Networks (DBNs) of SLAMSAT.

Effective search of sheltered object relies on robust localization, mapping, and tracking. To improve prediction accuracy, a robot has to move toward the sheltered zone and get more object information. This section proposes a scheme where the robot can simultaneously localize it, map environment features, and SAT a moving object (Fig. 5).

The SLAMSAT sample set is $\mathbf{X}_k = \{r_k, m_k, \mathbf{S}^i_k \mid 1 \le i \le N\}$, and

$$\mathbf{X}_k = \langle r_k, m_k, \mathbf{S}^i_k \rangle = \langle \langle r_{x,k}, r_{y,k}, r_{\theta,k}, \Sigma_{r,k}, m_k \rangle, \langle O_{x,k}, O_{y,k}, \Sigma_{O,k} \rangle^i, \langle G_{\phi,k}, U_k \rangle^i \rangle$$

where r_k is the robot state at time *k* and m_k is the map state at time *k*. Our SLAMSAT factorizes states into goal set distribution, object set distribution, robot state distribution, and the previous state set distribution as follows.

$$bel(\mathbf{X}_k) = P(\mathbf{X}_{1:k} \mid u_{1:k}, z_{1:k}) \overset{DBN}{=}$$
$$\underbrace{P(G^i_k \mid O^i_k, G^i_{k-1})}_{Goal\ \text{set}\ distributi\ on} \times \underbrace{P(O^i_k \mid O^i_{1:k-1}, G^i_{1:k-1}, r_k, m_{1:k}, u_{1:k}, z_{1:k})}_{Object\ \text{set distributi on}} \times$$
$$\underbrace{P(r_k, m_{1:k} \mid r_{1:k-1}, u_{1:k}, z_{1:k})}_{Robot\ set\ \text{distributi on}} \times$$
$$\underbrace{P(O^i_{1:k-1}, G^i_{1:k-1}, r_{1:k-1}, m_{1:k-1} \mid u_{1:k}, z_{1:k})}_{bel(\mathbf{X}_{k-1})}. \tag{6}$$

SAT conditioned on robot position and map is

$$P(O^i_k \mid O^i_{1:k-1}, G^i_{1:k-1}, r_{1:k}, m_{1:k}, u_{1:k}, z_{1:k})$$
$$= \eta \underbrace{P(z^O_k \mid O^i_k)}_{Object\ correction} \underbrace{P(O^i_k \mid O^i_{1:k-1}, G^i_{1:k-1}, r_{1:k}, m_{1:k}, u_{1:k}, z_{1:k-1})}_{Object\ \text{prediction}}. \tag{7}$$

Thus, we simplify (6) to be an EKF localization problem

$$P(r_k, m_k \mid O^i_{1:k-1}, G^i_{1:k-1}, r_{1:k-1}, u_{1:k}, z_{1:k})$$
$$= \eta \underbrace{P(z^L_k \mid r_k, m_k)}_{Robot\ Correction} \underbrace{P(r_k, m_k \mid r_{1:k-1}, u_{1:k}, z_{1:k-1})}_{Robot\ \text{prediction}}. \tag{8}$$

Details of EKF localization can be found in [12].

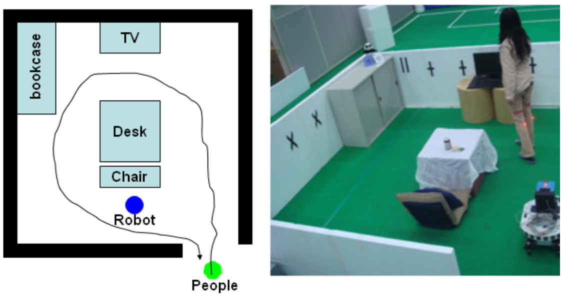
Table I : SLAMSAT algorithm

```
1.  Inputs:
      $S_{k-1} = \{\langle G^{(i)}_{k-1}, O^{(i)}_{k-1}, D \rangle \mid i = 1,...,N\}$ posterior at time k - 1
      $u_{k-1}$                     control mesurement
      $z_k$                         observation
2.  $S_k := \varphi$                 // Initialize
3.  $\bar{\mu}_k = g(u_k, \mu_{k-1})$     // Predict mean of robot and map postion
4.  $\bar{\Sigma}_k = G_k \Sigma_{k-1} G^T_k + R_k$   // Predict covairance of robot and map postion
5.  for m := 1,....,M do          // EKF SLAM update
6.      for c := 1,....,C do
7.          if $d^L_m < d^L_{th}$ do      // if $d_m < d_{th}$, $z_i$ is landmark m
8.              $K^c_k = \bar{\Sigma}_k H^{cT}_k (H^c_k \bar{\Sigma}_k H^{cT}_k + Q_k)^{-1}$
9.              $\mu_k = \bar{\mu}_k + K^c_k (z^c_k - h^c_k(\bar{\mu}_k))$
10.             $\Sigma_k = (I - K_k H^c_k) \bar{\Sigma}_k$
11.         else do
12.             $z^o_c = z_c$              // $z_c$ is a dynamic feature
13. $w^{(i)} := 0$
14. for i := 1,....,N do                 // RBPF Tracking
15      $G^i_k \sim p(G^i_k \mid O^i_k, G^i_{k-1})$        // virtual goal smapling
16.     $O^i_k \sim p(O^i_k \mid O^i_{1:k-1}, G^i_{1:k-1}, r_{1:k}, D, u_{1:k}, z_{1:k-1})$  // see (1)
17.     for j := 1,....,J do        // data association
18.         if $d^o_m < d^o_{th}$ do
19.             $O^i_k$ := kalman update($O^i_k$)  // update object
20.             $w^i_k := p(z^o_{k,j} \mid O^i_t)$         // compute weighting
21.             $S_k := S_k \cup \{\langle G^{(i)}_{k-1}, O^{(i)}_{k-1} \rangle\}$     // insert $S_t$ into sample set
22. Discard smaples in $S_t$ based on weighting $w^i_t$ (resampling)
23. return $S_t, \mu_t, \Sigma_t$
```

Our RBPF based SAT algorithm conditioned on SLAM is summarized in Table I. The EKF SLAM predicts and corrects robot position using EKF (lines 3-10). Laser measurements are represented as line features using the least square algorithm. The feature is either associated with a known landmark (line 7) or a leg feature (line 12). Goal states G^i_k are sampled and then the *N* kinds of object states O^i_k are predicted according to (1) (lines 15-16). Based on stream set distribution at time $k-1$, we assume the distance between the object and the goal is fixed at 200 cm so that we only randomly sample sink flow direction $G_{\phi,k}$ and sink flow intensity U_k for efficiency. If the *i*th particle is associated with a moving object, RBPF will update moving object position O^i_k. This is described as follows. First, the algorithm computes the weighting of the *i*th particle w^i_k and particles will be resampled (lines 20-22). In tracking case, the stream sample set S^i_k including the object sample set O^i_k and the goal sample set G^i_k will converge. In searching case, it will keep predict the object sample set O^i_k based on the previous stream field S^i_{k-1}

VI. EXPERIMENTAL RESULTS

We adopt UBOT with one SICK laser as the mobile robot platform and a 1.6 GHZ IBM X60 laptop with 0.5G RAM as the computing platform. The area of the experimental environment is 3.6m by 3.6m. We use PhaseSpace for the precise ground truth of people and robot trajectories [13]. The LEDs of PhaseSpace are mounted on two legs of the people and Ubot (Fig. 6(b)). The people walks along the line, and the robot follows the people through the remote control (Fig. 6(a)). The person is sheltered by the desk, bookcase and chair frequently so that tracking may turn into the searching case.



(a) (b)

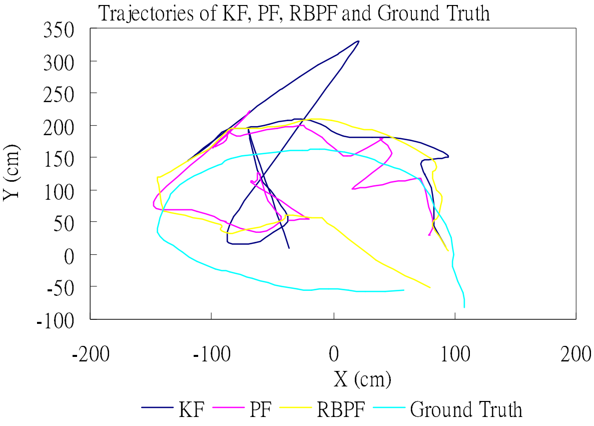
Fig.6. Environment setup: (a) Walking trajectory. (b) The experimental environment.

TABLE II.
Confusion matrix of AdaBoost detection.

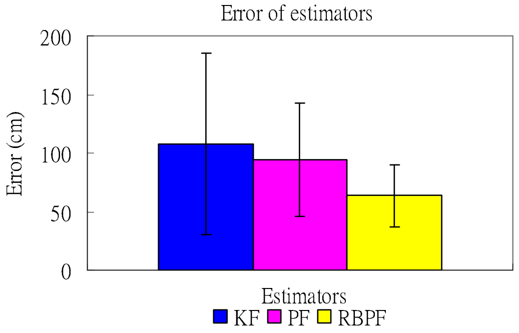
	Detected Label		
Groud Truth	Person	No Person	Total
Person	39 (88.8%)	5(11.2%)	44
No Person	45 (11.4%)	345 (88.6%)	390

TABLE III.
Confusion matrix of AdaBoost, scan association and data association.

	Detected Label		
Ground Truth	Person	No Person	Total
Person	39 (88.8%)	5(11.2%)	44
No Person	23 (5.9%)	367(94.1%)	390



(a)

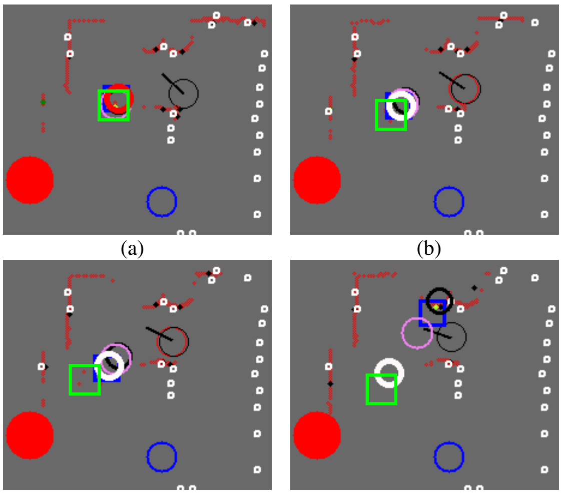


(b)

Fig.7. (a) Trajectories of KF, PF, RBPF, and ground truth. (b) Errors of KF, PF and RBPF.

TABLE IV.
Comparisons of tracking errors.

	Total mean (cm)	Total std. (cm)
KF	107.4	77.59
PF	94.0	48.39
RBPF	63.7	26.30



(a) (b)

(c) (d)

Fig.8. The experimental results of SLAMSAT. Small white circles are mapped landmarks. Black and black points are static and dynamic features, respectively. Black circle and red circle are robot position of odometer and estimated robot position of EKF SLAM, respectively. Blue circle is the original point. Blue square is the estimated people position of KF. Pink circle is the estimated people position of PF. White/Red circle is estimated people position of RBPF when it is the searching/tracking case. Red solid circle is people goal (sink) position of stream field. Green square is the ground truth of people position. (a) 44th frame. (b) 45th frame. (c) 46th frame. (d) 47th frame.

The confusion matrices of detection based on AdaBoost only and detection based on AdaBoost, scan association and data association are presented in Tables II and III, respectively. Obviously, scan association and data association increase accuracy rate of detection.

The tracking trajectories are shown in Fig. 7. In searching case, KF diverges faster than PF while RBPF keeps predicting the object position based on the stream field. Comparisons of average tracking errors among KF, PF, and RBPF are shown in Table IV. The average tracking errors of KF, PF, and RBPF are 107.4cm, 94.0cm, and 63.7cm, respectively.

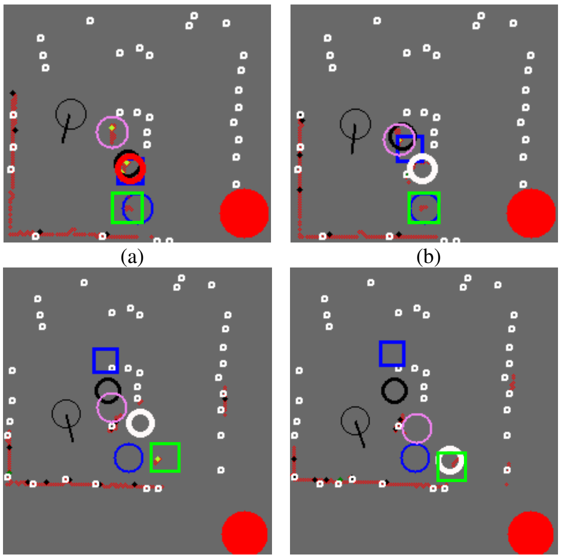


Fig.9. The experimental results of SLAMSAT with false data association. (a) 58th frame. (b) 59th frame. (c) 63th frame. (d) 64th frame.

The experimental results of SLAMSAT are shown in Figs. 8 and 9. As shown in Figs. 8(a) and 8(b), the KF and PF based tracking keep predicting people based on their motion model. However, RBPF can further keep searching people position based on the virtual goal. Figures 8(c) and 8(d) shows the incorrect KF estimation which is resulted from false detection. Parts of the PF particles are associated with the wall feature while others are not. Accordingly, the people position estimated by PF is between those by KF and RBPF. Nevertheless, RBPF can still keep searching people position based on the virtual goal.

In Fig. 9, the algorithm will infer that the chair is the person if the chair is deemed as leg features and the person is out of sight but near the chair. The false alarm of leg detection may result in the incorrect estimation of KF and RBPF (Fig. 9(a)). Also, the incorrect PF estimation is resulted from another false alarm of leg detection. As shown in Fig. 9(b), the incorrect estimation of KF and PF are resulted from false detection. However, RBPF can still keep searching people position based on the virtual goal. In Fig. 9(c), KF and PF diverge when there is no feature near the last estimation. When the leg feature is detected, the estimation of the people position by RBPF is near the feature so that RBPF can estimate the people position correctly. However, the feature position is far from that by KF estimation. Parts of PF particles are associated with the leg feature, but others are not so that the estimation of PF is inaccurate (Figs. 9(c) and 9(d)). Obviously, the experimental results show that our proposed RBPF algorithm is better than KF and SIR PF in the searching and tracking case.

VII. Conclusions

This paper proposes a novel SAT algorithm based on stream functions and RBPF conditioned on SLAM called SLAMSAT. SLAMSAT estimates the moving object position, robot position, and map under sensor uncertainty. The experimental results show our algorithm can search and track moving objects effectively.

References

[1] A. Yilmaz, O. Javed, and M. Shah, “Object tracking: A survey,” *ACM Computing Surveys*, vol. 38, no. 4, 2006.

[2] Y. Bar-Shalom and X.-R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques*, YBS, Danvers, MA, 1995.

[3] C.-C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte, “Simultaneous localization, mapping and moving object tracking,” *The International Journal of Robotics Research*, vol. 26, no. 9, pp. 889-916, Sept. 2007.

[4] M. Montemerlo, S. Thrun, and W. Whittaker, “Conditional particle filters for simultaneous mobile robot localization and people-tracking,” in *Proc. IEEE International Conference on Robotics and Automation*, pp. 695-701, May 2002.

[5] C. Kwok and D. Fox., “Map-based multiple model tracking of a moving object,.” *Robocup Symposium,* 2004.

[6] N. Roy and C. Earnest, ”Dynamic action spaces for information gain maximization in search and exploration,” in *Proc. the American Control Conference*, June 2006.

[7] B. Lavis, T. Furukawa, and H. Durrant-Whyte, ”Dynamic space reconfiguration for Bayesian search and tracking with moving targets,” *Autonomous Robots*, vol. 24, no. 4, pp. 387-399, May 2008.

[8] K.-S. Tseng, “A stream field based partially observable moving object tracking algorithm,” *10th IEEE Intl. Conference on Control, Automation, Robotics and Vision*, Hanoi, Vietnam, 2008.

[9] W. Kaufmann, *Fluid Mechanics*, McGraw-Hill, 1963.

[10] S. Waydo and R. M. Murray, “Vehicle motion planning using stream functions,” in *Proc. IEEE International Conference on Robotics and Automation,* vol.2, 2003.

[11] X. Y. Xu and B. X. Li, “Adaptive Rao–Blackwellized particle filter and its evaluation for tracking in surveillance,” *IEEE Trans. Image Processing*, vol. 16, no. 3, pp. 838-849, March 2007.

[12] S. Thrun, W. Burgard, and D. Fox, Probabilistic Robotics, MIT Press, 2005.

[13] http://www.phasespace.com/

[14] K. O. Arras, O. M. Mozos, and W. Burgard, “Using boosted features for the detection of people in 2D range data,” *Proc. IEEE International Conference on Robotics and Automation*, Rome, Italy, 2007.

[15] K. O. Arras, S. Grzonka, M. Luber, and W. Burgard, “Efficient people tracking in laser range data using a multi-hypothesis leg-tracker with adaptive occlusion probabilities,” in *Proc. IEEE International Conference on Robotics and Automation*, May 2008.