Available online at www.sciencedirect.com
ScienceDirect
ICT Express 1 (2015) 82–85
www.elsevier.com/locate/icte

# An adaptive hybrid filter for practical WiFi-based positioning systems

Namjun Park, Sukhoon Jung, Dongsoo Han*

*Department of Computer Science, KAIST, Daejeon, Republic of Korea*

Received 23 January 2015; accepted 21 February 2015
Available online 22 October 2015

**Abstract**

This paper proposes an adaptive hybrid filter for WiFi-based indoor positioning systems. The hybrid filter adopts the notion of particle filters within the prediction framework of the basic Kalman filter. Restricting the predicts of a moving object to a small number of particles on a way network, and replacing the Kalman gain with a dynamic weighting scheme are the key features of the hybrid filter. The adaptive hybrid filter significantly outperformed the basic Kalman filter, and a particle filter in the performance evaluation at three test places: a Library and N5 building, KAIST, Daejeon, and an E-mart mall, Seoul.



*Keywords:* Indoor positioning system; Kalman filter; Adaptive hybrid filter; WiFi fingerprint; Particle filter

## 1. Introduction

In WiFi-based localization, the estimated location is not always accurate and it frequently oscillates even when a user stays at a fixed location. Thus, displaying the location of the user in a stable manner is one of the most challenging issues of WiFi-based indoor positioning systems. In particular, in the single time location estimation of non-moving objects, it is very difficult to cope with this accuracy fluctuation problem without the help of additional sensors such as a gyroscope, a barometer, a compass, and a 3-axis accelerometer. However, in the case of navigation or real-time tracking, we can improve or stabilize the current location estimation to some degree by referring to previous signals and location information (*i.e.*, historic data).

This paper proposes a new location filter for indoor positioning systems. Since the new filter operates in the framework basic Kalman filter (BKF) [1], and it incorporates the notion of the particles filters into the filter, we named it an adaptive hybrid filter (AHF). Like BKF, AHF does not require any additional information from sensors like a gyroscope, a barometer, a compass, or a 3-axis accelerometer. However, when necessary it uses additional information from the sensors.

When we compared the accuracies of the BKF, AHF, and a particle filter measured at a library, N5 building, KAIST, Daejeon, Korea, and an E-mart discount store, Seongsu, Seoul, a significant accuracy improvement was achieved by AHF. At a library, KAIST, around 18.0%, at N5 building, KAIST, 29%, and at an E-mart discount store, around 25.0% accuracy improvements were achieved, respectively. When we compared the accuracy of AHF with that of the particle filter, the AHF showed significantly better accuracy improvement than the particle filter while it showed a greatly improved performance in processing time compared with the particle filter. When we applied the AHF and integrated it with "myCoex", indoor navigation system, which is known as the first full-fledged commercialized indoor navigation system [2], the effect of using the AHF was apparent in improving the accuracy and the stability of location estimation.

## 2. Adaptive hybrid filter

### 2.1. State model

For a clearer understanding of the AHF, we contrast the state models of the AHF with that of BKF. The state model represents an instance of a target object at a certain time. In BKF, the state model is specified using the location and the velocity vector; that is, the state instance at time $k$, $x_k$ is represented by $[C_k^x, C_k^y, V_k^x, V_k^y]^T$, where $C_k^x C_k^y$ are

* Corresponding author.
*E-mail addresses:* njpark@kaist.ac.kr (N. Park), sh.jung@kaist.ac.kr (S. Jung), dshan@kaist.ac.kr (D. Han).

Peer review under responsibility of The Korean Institute of Communications Information Sciences.

http://dx.doi.org/10.1016/j.icte.2015.09.008

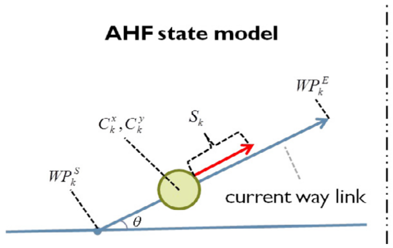


Fig. 1. State models of BKF and AHF.

coordinates of $x$, $y$ at time $k$, and $V_k^x V_k^y$ are the velocities at time $k$ [3].

The state model of BKF has been slightly changed in AHF. The state model of AHF consists of a location vector, speed, and way link; that is, the state instance at time $k$, $x_k$ is represented by $[C_k^x, C_k^y, S_K, WP_k^S, WP_k^E]^T$, where $S_K$ is the speed, and $WP_k^S, WP_k^E$ respectively, are the start and the end points of a way link corresponding to an edge of a way network. In the model, the speed and the way link substitute for the velocity vectors of BKF's state model. Fig. 1 shows the state models of AHF and BKF.

### 2.2. Adaptive hybrid filter

Based on the state models defined above, we can describe the relation between the state instances $x_k$, at time $k$, and $x_{k-1}$ at time $k-1$. In BKF, the relation between the states at time $k$ and $k-1$ is defined by:

$$C_k^x = C_{k-1}^x + \delta T \cdot V_{k-1}^x,$$
$$C_k^y = C_{k-1}^y + \delta T \cdot V_{k-1}^y.$$

On the other hand, in AHF, the relation is defined by:

$$C_k^x = C_{k-1}^x + \delta T \cdot S_{k-1} \cdot \cos\theta,$$
$$C_k^y = C_{k-1}^y + \delta T \cdot S_{k-1} \cdot \sin\theta,$$

where, $\theta$ is the angle of the way link and the base line. The base line is shared by all the way links. Since there are two directions on a straight line, $\theta$ can have two possible values on the straight line: $\theta$ and $\theta + \pi$. At a perpendicular four-way intersection, $\theta$ can have four possible values: $\theta, \theta + \pi/2,\ \theta + \pi,\ \theta + 3\pi/2$.

This means that for an instant on a straight line, we assume there are two possible candidates for predictions: forward and backward directions. For an instant on a four-way intersection, we assume that there are four possible candidates for predictions. Among the candidates, the closest candidate to the measurement is chosen as the final prediction. As long as the angle of the way links connected to the intersections is known, the number and the angle of the way links do not matter in AHF. Note that we can obtain the angle of each way link once the way network is derived from a map. The prediction model of AHF is described by:

$$\begin{bmatrix} C_k^x \\ C_k^y \\ S_k^x \end{bmatrix} = \begin{bmatrix} 1 & 0 & \delta T \cdot \cos\theta \\ 0 & 1 & \delta T \cdot \sin\theta \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_{k-1}^x \\ C_{k-1}^y \\ S_{k-1}^x \end{bmatrix} + \begin{bmatrix} w_k^{C^x} \\ w_k^{C^y} \\ w_k^{S^x} \end{bmatrix}$$

where, $w_k$ is the process noises $w_k \sim N(0, Q_k)$, and $Q_k$ is the process error covariance. In the observation model of AHF, the observation $z_k$ is defined by:

$$\begin{bmatrix} z_k^x \\ z_k^y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} C_k^x \\ C_k^y \\ S_k^x \end{bmatrix} + \begin{bmatrix} v_k^x \\ v_k^y \end{bmatrix}$$

where, $v_k$ is the observation noise $v_k \sim N(0, R_k)$, and $R_k$ is observation error covariance.

Meanwhile, the BKF update is performed using Kalman gain, which is obtained from an error covariance matrix. Kalman gain is used for assigning weights for the combination of a prediction and a measurement. If the measured location is unreliable, it assigns a higher weight to the predicted location. If the measured location is reliable, it assigns a higher weight to the measured location. We leave the details on how to compute the error covariance matrix and Kalman gain to [1].

AHF does not use Kalman gain in the update because Kalman gain usually converges into a specific value. As a result, Kalman gain has limitations in reflecting the accuracies of WiFi-based localization that change dynamically. In AHF, instead of building an error covariance matrix, the accuracy at each location is computed, and then it computes the relative error level of each location. Best Candidate Set (BCS) method [4] was used for the error estimation. In BCS, the error is computed by the distances between the nearest neighbor and the remaining neighbors estimated using a method similar to $k$NN method. After the relative error level is determined, the weights for a prediction and a measurement are defined by:

$$N_k = \frac{1}{\frac{1}{k-1}\sum_{i=1}^{k-1} err_i^e} + \frac{1}{err_k^e} w_k^p = \frac{1}{N_k} \cdot \frac{1}{\frac{1}{k-1}\sum_{i=1}^{k-1} err_i^e}$$

$$w_k^m = \frac{1}{N_k} \cdot \frac{1}{err_k^e},$$

where $x_k^p,\ y_k^p$ are the coordinates of a prediction, which are the results of a prediction phase, and $x_k^m,\ y_k^m$ are coordinates of an estimated location into a real number in the range of 0–1. The weight increases as the average of estimated errors decreases. Once the weight is computed, the filter combines the predictions and measurements by:

$$x_k = w_k^p \cdot x_k^p + w_k^m \cdot x_k^m$$
$$y_k = w_k^p \cdot y_k^p + w_k^m \cdot y_k^m,$$

where, $x_k^p,\ y_k^p$ are the coordinates of a prediction, which are the results of a prediction phase, and $x_k^m,\ y_k^m$ are coordinates of an estimated location.

## 3. Experimental results

### 3.1. Experiment setup

In order to evaluate the validity of AHF for indoor navigation, we collected three datasets: one from the 4th floor of a KAIST library, one from the 2nd floor of N5 building, KAIST,