A Map Matching algorithm based on a particle filter

Karim El Mokhtari[1]* , Serge Reboul[2], Monir Azmani[1], Jean-Bernard Choquel[2],
Salaheddine Hamdoune[1], Benaissa Amami[1], Mohammed Benjelloun[2]
[1] Laboratoire d'Informatique, Système et Télécommunications,
Abdelmalek Essaadi University, Faculty of Sciences and Technology,
Ancienne Route de l'Aéroport, Km 10, Ziaten. BP: 416. Tangier, Morocco
[2] Laboratoire d'Informatique Signal et Image de la Côte d'Opale,
Littoral Côte d'Opale University, Univ Lille Nord de France,
50, rue Ferdinand Buisson BP 719 - 62228 Calais Cedex, France
Email: karim@elmokhtari.com - serge.reboul@univ-littoral.fr

***Abstract*—Map matching is the process of finding a match for each GPS point in a vehicle's trajectory to roads on a digital map. Extensive research has been conducted during the last years yielding many algorithms based on different approaches. One of the challenges that face those algorithms is the interruption of GPS signals that occurs specially in dense urban environments. In these cases on-board sensors like odometers and accelerometers can be used temporarily for positioning, however due to the poor accuracy of these methods, the quality of map matching decreases significantly. In this paper, we propose a method that improves the quality of map matching when GPS signals are not available. This method is based on a particle filter using heading and velocity measurement. We evaluate this method through its integration with an existing topological map matching algorithm. We compare the performances when this algorithm is used alone and when associated with the particle filter.**

***Keywords*—Map matching; Particle filter; GPS; circular filter; localization; tracking**

I. Introduction

The increasing use of the GPS devices has helped during the last decades to localize moving objects and people. However, due to the intrinsic errors related to the GPS system and the used sensors, the difference between the measured and the real trajectory of the vehicle could become significant. In order to reduce this difference, we try to associate each GPS point to a road on a digital map. This process is commonly known as *Map matching*.

The general purpose of map matching is to identify the real road on which a vehicle or a person is traveling. It's a key process in many applications such as : fleet management, Intelligent Transport Systems (ITS) and accident detection. During the last years, many classes of map matching algorithms have been developed. The first class takes into account the geometrical relationship between the GPS position and the map elements (nodes, roads) and are referred to as *geometrical algorithms*. The second class, called *topological algorithms*, considers, in addition, the topology of the roads network and the history of the GPS positions. It has been demonstrated in [1] that this class gives more improved results than the first one by discarding aberrant cases like a matched point jumping suddenly between parallel roads when considering only the proximity between the GPS point and the road. Other approaches have been applied to map matching like the probabilistic approach [2], fuzzy logic [3] and advanced techniques [4] [6]. All these algorithms fall under the category of *local/incremental* algorithms that rely on the last matches to evaluate the current one. The *global* algorithms [7], in the another hand, proceed differently by trying to find the best fit of the whole trajectory with the road network. Generally these algorithms may achieve a better matching quality but at a higher computational cost [8].

Several restrictions limit the accuracy of most map matching algorithms, namely :

- The matching of the first point of the vehicle's trajectory : a bad initial matching alters significantly the quality of the subsequent matching that are in most cases based on this initial match
- Map matching in urban environments where the road network is so dense that it becomes difficult to find the correct match to the GPS position specially in Y-shape junctions, roundabouts and close parallel roads. In addition, the accuracy of the GPS positioning in such environments is often altered by phenomena like signal multiple paths, reduced satellites visibility and Dilution Of Precision (DOP)
- The GPS sampling rate : Most map matching algorithms need a high GPS sampling rate (one point every 1 to 30 seconds). The accuracy of those algorithms drops significantly with lower measurements rates (1 to 3 minutes) which is typically the case in applications like remote tracking of vehicles by GPRS or in dense urban environments where the GPS signal may become unavailable for many minutes.

The use of on-board sensors like odometers, gyroscopes and accelerometers can help in localizing the vehicle during the absence of GPS signals. This process is widely known as *Dead Reckoning* (DR) where the current position is calculated based on the last one. However, the high level of noise on those sensors reduces the quality of this kind of localization and results in a drift of the calculated position at each time-step.

The vehicle's heading sensing improves significantly the quality of finding good matches for GPS positions on the road network. Quddus proposed in his topological algorithm described in [9] to identify the correct link of the GPS fix with a weighted combination of several criteria : the vehicle's heading, the proximity of the GPS fix to the link and the angular relative position of the link. He concluded that the vehicle's heading has more influence on the quality of the matching than any other factor. However, he admitted that the heading readings from GPS receivers become less accurate for low vehicle's velocities and completely unavailable for insufficient number of visible GPS satellites.

To overcome these shortcomings, the heading measurement could be performed using a magnetometer that offers high measurement rates and an autonomy regarding GPS satellites availability. We provided in [10], [5] another improvement on the heading measurement through the implementation of circular filters. We showed that this filter gives better estimates of the heading and is unaffected by transitions between $-\pi$ and π. We also showed that this filter can provide a knowledge of the vehicle's behavior through model probability calculation.

In this paper, we propose to insure the continuity of the map matching process even when GPS signals are unavailable and this will be performed with a novel method based on a particle filter applied to heading and velocity measurement. We will follow this organization : after this brief introduction, we'll describe the theory behind our particle filter then a system overview. In the experimental part, we'll compare the performances of an existing map matching algorithm on a synthetic data set when used alone and when associated with the particle filter. In the last section, we summarize our contributions and discuss the rooms for future works.

II. Particle Map Matching

A. Classical Approach

When GPS signals are available we perform the map matching using the improved topological algorithm described by Quddus in [9] that we denote ITA in this paper. This algorithm uses 3 weighted criteria to evaluate the correct segment : the difference between the vehicle's heading and the segment direction(WS_H), the proximity of the GPS fix to the segment (WS_{PD}) and the relative position (WS_{RP}). The score of a segment (TWS) is the sum of these 3 weightings. The segment with the higher score is chosen to map the current GPS fix. The matched position is the perpendicular projection of the GPS fix on that segment. When GPS signals are not available, we apply the Dead Reckoning technique where the heading and velocity are used to estimate the next vehicle's position as follows:

$$x(k+1) = x(k) + v(k).\Delta t.\cos\theta(k) + dx(k) \quad (1)$$

$$y(k+1) = y(k) + v(k).\Delta t.\sin\theta(k) + dy(k) \quad (2)$$

where : $x(k)$, $y(k)$, $x(k+1)$ and $y(k+1)$ are the horizontal and vertical components of the vehicle's position at time k and $k+1$ respectively, $v(k)$ and $\theta(k)$ are the vehicle's velocity and heading respectively, $dx(k)$ and $dy(k)$ the error on the horizontal and vertical component of the vehicle's position respectively.

It has been demonstrated in [9] that $dx(k)$ and $dy(k)$ are expressed as follows:

$$dx(k) = dv(k).\Delta t.\cos\theta(k) - v(k).\Delta t.\sin\theta(k).d\theta(k) \quad (3)$$

$$dy(k) = dv(k).\Delta t.\sin\theta(k) + v(k).\Delta t.d\theta(k).\cos\theta(k) \quad (4)$$

where : $dv(k)$ and $d\theta(k)$ are the errors on velocity and heading at time k.

Due to these errors, the uncertainty on the vehicle's position $x(k)$ and $y(k)$ increases at each time step resulting in worse matchings of the vehicle's position on the map.

B. Particle Map matching principle

When GPS signals are available we perform the classical map matching described in the preceding section using the improved topological algorithm [9]. When the GPS signal is not available we propose a Map matching filter that uses the measurements of directions as observations and the vehicle speed as control command.

Particle filters, also known as Sequential Monte Carlo methods, help in many applications to estimate the posterior density of the state space by implementing the Bayesian recursion equations with no restrictive assumption about the dynamics of the state-space of density function. The state-space model can be non-linear and the initial state and noise distributions can take any form. One of the known limitations of the particle filter is that its performance degrades quickly for high-dimensional systems. The particle filter uses a set of particles to represent the posterior density. Each particle has a weight that represents the probability of that particle being sampled from the probability density function. A particle filter is used here to find the best match of the current vehicle's position when GPS signals are temporarily unavailable by using only the vehicle's velocity and heading.

The first particles are delimited by the GPS error circle whose radius is the precision of the last available GPS fix as shown on figure 1. This circle becomes an ellipse if the horizontal and vertical precision of the GPS device aren't identical. The chosen segments should be inside the error circle and the difference between their direction and the vehicle's heading (red vector) shouldn't be more than $\pm 45°$.

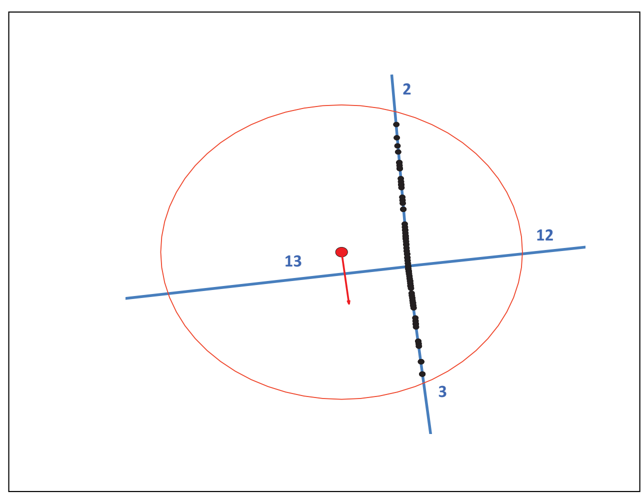

Fig. 1. Initial distribution of the particles

The particles are created uniformly on the retained segments, then they are affected a weight according to a normal distribution centered on the last GPS fix and with a variance that is the GPS reading's variance. Then we re-sample these particles by giving more probability to the those with more weight. In such a way, we obtain a new set of particles more concentrated near the GPS fix as depicted on figure 1.

The particle filter evolves in four steps : prediction, update, estimation and re-sampling.

In the prediction step, each particle moves along the segment on which it's located by a distance of $d = v(k).\Delta t$, where $v(k)$ is the vehicle's velocity at time k and Δt is the odometer's sampling period.

If a particle reaches the end of a segment, many cases can arise and are depicted in figure 2 where a particle P is at the end of segment 1. Given the vehicle's heading $\theta(k)$ represented by the green vector, the next position of this particle is probably on segment 4, and less likely on segments 2 or 6 that are perpendicular or opposed to the vehicle's movement respectively. However the vehicle may possibly take the path of segments 3 or 5 that are in the same direction of the vehicle's movement even if there is an angular difference between the vehicle's heading and the segment direction (denoted here $\Delta\theta_1$ to $\Delta\theta_3$). That's why, in order to decrease the calculation time, we can choose to keep the segments for which this difference is no more than 45° in absolute value: $|\Delta\theta_i| < 45°$ which is a reasonable value for the maximum angular difference. Otherwise even if all the following segments are retained, the filter will automatically discard the less probable ones in the update step as explained in section II-C.

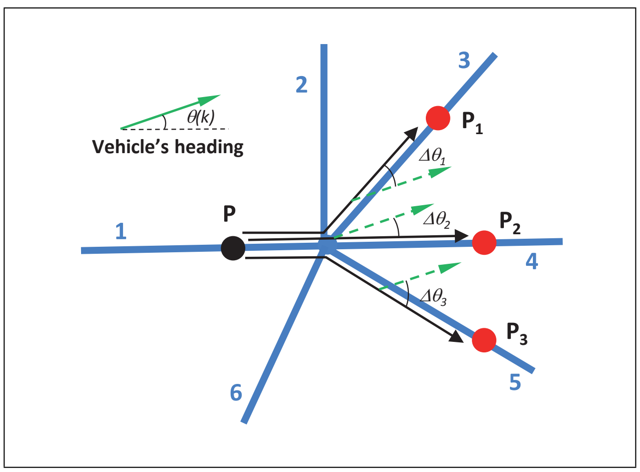

Fig. 2. Particles at a segment's end

By applying this rule, segments 2 and 6 are discarded and segments 3, 4 and 5 retained. We create new particles (P_1, P_2 and P_3) on the retained segments thus considering them as new probable paths for the vehicle.

In case we find no segment that satisfies the condition $|\Delta\theta_i| < 45°$, the particle is deleted because it evolves probably along an unlikely segment as shown on figure 3.

The update step in the particle filter allows to check if a particle exists on the correct path by assigning a weight that accounts for the difference between the vehicle's heading and the segment direction. Thus for particle i the weight $w_i(k)$ is calculated as a function of :

- $\theta(k)$ the vehicle's heading measured at time k

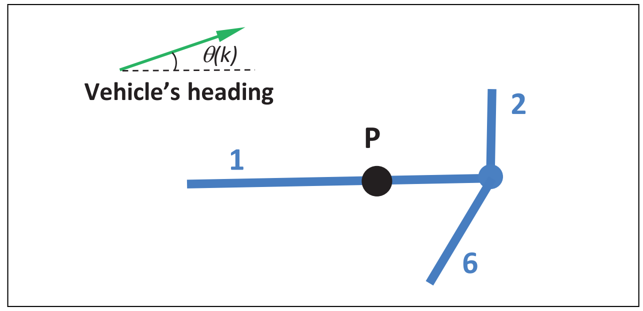

Fig. 3. Particle at a dead end

- $\phi_v(k)$ the direction of the current segment v where the particle is located at time k

This weight emphasizes the particles that evolve on segments more aligned with the vehicle's movement and by the same token disfavors the ones that are opposed or perpendicular to the vehicle's direction of movement.

Then, we estimate the vehicle's position with a weighted sum of the coordinates of each particle.

The last step of the particle filter is re-sampling. In this step, the particles are redistributed according to their weight. We choose randomly a fixed number of particles among the existing ones by giving more probability to particles with higher weight. At the end of this step, we obtain a new set of particles with a higher concentration on the segments where the vehicle is more likely to exist.

We should underline here that the key instants in this algorithm are when the vehicle makes a turn. This allows to eliminate all the particles that are on segments where no turn is possible given the vehicle's heading like the example on figure 3 and concentrate more particles on the segments where a turn on the detected direction is possible.

C. Particle map matching algorithm

Let's denote the last measured GPS position $(x_p, y_p)^T$ obtained with an accuracy defined by the respective error variances v_x, v_y for the horizontal and vertical axis. In the initialization step, the current segment associated to the last available GPS solution is noticed v and V is the set of connected segments. In our system, the filtering algorithm is as follows:

1) *Initialization*: we draw N particles $x_0^i = (X^i(0)Y^i(0))^T$, $i = 1, \ldots, N$. $X^i(0)$, $Y^i(0)$ are drawn from a normal distribution of respective mean x_p, y_p and variance v_x, v_y. The weights are initialized to :

$$w_0^i = \frac{1}{2\pi\sqrt{v_x\ v_y}} \exp(-\frac{(X^i(0)-x_p)^2}{2v_x} - \frac{(Y^i(0)-y_p)^2}{2v_y})$$

2) *Prediction step*: the particles are propagated on the road network with the following rule:
If a particle reaches the end of the current segment, sample uniformly from the set V a segment v, otherwise v is unchanged. Then the particle is propagated

via the following equations :

$$X^i(k) = X^i(k-1) + v(k-1).\Delta t \ \cos(\phi_v(k-1)) + \nu_r \ \cos(\phi_v(k-1)) \quad (5)$$

$$Y^i(k) = Y^i(k-1) + v(k-1).\Delta t \ \sin(\phi_v(k-1)) + \nu_r \ \sin(\phi_v(k-1)) \quad (6)$$

where ν_r follows a normal distribution of mean 0 and variance v_Q. ν_r is a random noise that models the process noise variance and the control command inaccuracy.

3) *Update step*: the weights are computed according to the angle measurements:

$$w_k^i = w_{k-1}^i f_{CN}\left(\theta(k); \phi_v(k), \kappa_R\right) \quad (7)$$

The von Mises distribution f_{CN} is symmetric, unimodal on one period and its density is given by the following expression [12]:

$$f_{CN}(\theta(k); \phi_v(k), \kappa_R) = \frac{1}{2\pi I_0(\kappa_R)} e^{\kappa_R \ cos(\theta(k)-\phi_v(k))}$$

where $\theta(k)$ defines the vehicle's heading, $\phi_v(k)$ segment v direction and κ_R the concentration parameter. I_0 is the modified Bessel function of the first kind and zero order. $I_0(\kappa_R)$ can be considered as a normalizing constant. κ_R defines the accuracy of the measurements obtained with a circular filter [5] used to filter the heading's observations provided by the magnetometer. The weights are processed and then normalized.

4) *Estimation*: the coordinates and the error variances are estimated according to the following formulas:

$$\hat{x}(k) = \sum_{i=1}^{N} w_k^i X^i(k) \quad (8)$$

$$\hat{y}(k) = \sum_{i=1}^{N} w_k^i Y^i(k) \quad (9)$$

$$v_x(k) = \sum_{i=1}^{N} w_k^i (X^i(k) - \hat{x}(k))^2 \quad (10)$$

$$v_y(k) = \sum_{i=1}^{N} w_k^i (Y^i(k) - \hat{y}(k))^2 \quad (11)$$

5) *Re-sampling*: the chosen re-sampling is here the SIR (Sequential Importance Resampling) [11] with effective number of particles as follow:

$$N_{eff} = \frac{1}{\sum_{i=1}^{N} {w_k^i}^2} \quad (12)$$

The selected segment in our map matching process can be chosen as the closest one to the estimated position $(\hat{x}(k), \hat{y}(k))^T$, or the one where the maximum number of particles exists. The latest criteria is used in our experimentation.

III. Experimentation

We experiment the particle map matching on synthetic data. We consider a vehicle equipped with a GPS receiver, an odometer for velocity sensing and a magnetometer for heading sensing. On a digital map, we choose a trajectory in the city of Calais - France that the vehicle will follow as shown on Figure 4. We choose randomly a point of the trajectory and we mask the GPS signals for n consecutive points.

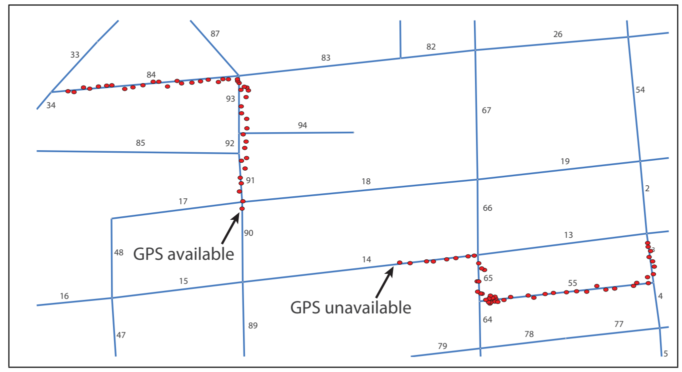

Fig. 4. Vehicle's synthetic trajectory with GPS masking

In this experimentation, we test the performances of ITA alone and associated with the particle filter that we denote ITA-PF. In the ITA-PF algorithm, we use ITA when GPS signals are available and the particle filter when they aren't.

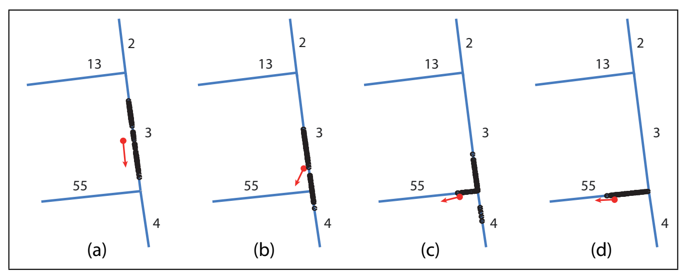

Fig. 5. Particle filter's behavior at a turn

In the example illustrated in figure 5, we observe the behavior of the particle filter on a turn. At step (a), all particles are on segment 3 since its direction is aligned with the vehicle's heading indicated by the red vector, then at step (b), the vehicle begins turning towards segment 55, some particles move to segment 4 since its direction is still not far away by 45° from the vehicle's heading. At step (c), the vehicle finishes its turn, many particles are moving to segment 55 since it's aligned with the vehicle's heading, at the same time the particles on segment 4 are decreasing in number, their weight becoming weaker because of the increasing difference between segment 4 direction and the heading. Finally at step (d), the vehicle moves on segment 55, all the particles are on this segment since its direction is the closest to the heading.

To evaluate to the performance of the two algorithms ITA and ITA-PF, we mask n consecutive GPS points at a random starting point of the trajectory, the first algorithm uses Dead Reckoning while the second uses the particle filter. For each n, we repeat the experiment 100 times, the starting point is chosen randomly each time. Then we calculate the average of

TABLE I. PERCENTAGE OF CORRECT SELECTED SEGMENT AS A FUNCTION OF THE MASKED POINTS (GPS ERROR = 15M)

Masked points	10	20	30	40	50	60
ITA	83.2%	81.6%	78.4%	78.4%	76.0%	73.6%
ITA-PF	84.0%	83.2%	82.4%	84.0%	81.6%	80.8%

TABLE II. PERCENTAGE OF CORRECT SELECTED SEGMENT AS A FUNCTION OF THE GPS ERROR (MASKED POINTS = 30)

GPS error (m)	5	10	15	20	25	30
ITA	90.4%	87.2%	78.4%	72.8%	66.4%	58.4%
ITA-PF	93.6%	90.4%	82.4%	76.0%	68.8%	61.6%

the points where the selected segment was correct. The results are shown on table I.

In a second experiment, we maintain the same number of masked points $n = 30$, but we change the GPS error. Likewise, we repeat the experiment 100 times, choosing each time a different starting point for masking. Then we calculate the average of correct segment selection. The results are shown on table II.

On figure 6, we represent the percentage of correct segment selection for different GPS errors.

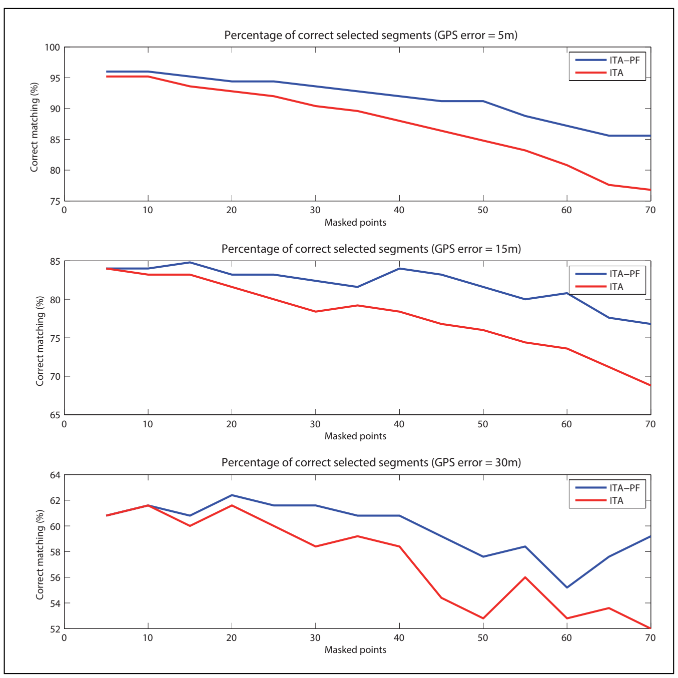

Fig. 6. Percentage of correct selected segments for different GPS errors

We see clearly that the ITA-PF algorithm outperforms the ITA algorithm specially when the number of masked points increases, in this case the ITA-PF shows more performance in selecting the correct segment than the Dead reckoning technique that decreases in precision at each time step.

We also notice that the performance of both algorithms decreases for higher GPS errors, but the ITA-PF remains always better as shown on figure 6.

Figure 5 shows the reason behind the gain of accuracy with the particle filter : even if, by the time, the particles are spread due to the errors on velocity measurement, they are concentrated again during the vehicle's turns. These turns act like key points that help discarding particles on less probable segments and recreate them on the most probable one. This is not the case in Dead Reckoning where the errors keep increasing continuously.

IV. CONCLUSION

We propose in this paper a new map matching method based on a particle filter. This method is applied when the GPS signals are not available and uses the velocity and heading to estimate the current vehicle's position. We compare the performance of an existing map matching algorithm when used alone and with the particle filter. We show in the experimentation that the use of the particle filter improves significantly the quality of map matching specially when the GPS signals are masked for a long period.

In our future work, we plan to improve the accuracy of our method through the integration of the circular interacting multi-model filter proposed in [10] that offers a better heading's estimation and a detection of the vehicle's behavior.

REFERENCES

[1] J.S. Yang, K.S. Chon, and S.P. Kang, "The map matching algorithm of GPS data with relatively long polling time intervals", Journal of the Eastern Asia Society for Transportation Studies, 6:2561-2573, 2005

[2] J. Pyo, D. Shin, and T. Sung, "Development of a map matching method using the multiple hypothesis technique", in IEEE Proceedings on Intelligent Transportation Systems, pages 23-27, 2001

[3] M.A. Quddus, R.B. Noland, and W.Y Ochieng, "A high accuracy fuzzy logic based map matching algorithm for road transport", Journal of Intelligent Transportation Systems, 10(3):103-115, 2006

[4] D. Obradovic, H. Lenz and M. Schupfner, "Fusion of Map and Sensor Data in a Morder Car Navigation System", Journal of VSLI Signal Processing 45, pp. 112-122, 2006

[5] G. Stienne, S. Reboul, M. Azmani, J.B. Choquel, M. Benjelloun, "A multi-temporal multi-sensor circular fusion filter", Information Fusion, article in press, 2013.

[6] O. Pink and B. Hummel, "A Statistical Approach to Map Matching Using Road Network Geometry, Topology and Vehicular Motion Constraints", in proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems, 2008

[7] S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk, "On map-matching vehicle tracking data", In 31st VLDB Conference, Trondheim, Norway, pages 853-864, 2005

[8] Lou, Y., C.Y. Zhang, Y. Zheng and X. Xie, "Map-Matching for Low-Sampling-Rate GPS trajectories", in Proceedings of ACM SIGSPATIAL Conference on Geographical Information Systems (ACM GIS 2009)

[9] M.A. Quddus, "High Integrity Map Matching Algorithms for Advanced Transport Telematics Applications", PhD thesis, Centre for Transport Studies, Department of Civil and Environmental Engineering, Imperial College London, 2006

[10] K. El mokhtari, S. Reboul, M. Azmani, J.B. Choquel, S. Hamdoune, B. Amami, M. Benjelloun, "A circular interacting multi-model filter applied to map matching", in Proceedings of the 16th International Conference on Information Fusion, Istanbul, Turkey, pages 1473-1478, 2013

[11] S. Arulampalam, S. Maskell, N. Gordon and T. Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking", IEEE Transactions on Signal Processing,Vol 50, No 2, pp. 174-188, 2002.

[12] S.R. Jammalamadaka and A. SenGupta, "Topics in Circular Statistics", World Scientific Publ, New Jersey, 2001.