

Kinematic GNSS Shadow Matching Using a Particle Filter

Lei Wang

UCL Engineering, University College London, London, United Kingdom

BIOGRAPHY

Mr Lei Wang is a PhD student in the Space Geodesy & Navigation Laboratory at University College London. He received a Bachelor's degree in Geodesy and Geomatics from Wuhan University in 2010. His research interests include the fields of GNSS-based positioning techniques for urban canyons, and indoor positioning techniques. (lei.wang.10@ucl.ac.uk).

ABSTRACT

The poor performance of GNSS user equipment in urban canyons is a well-known problem and is particularly inaccurate in the cross-street direction. However, the accuracy in this direction greatly affects many applications, including vehicle lane identification and high-accuracy pedestrian navigation. Shadow matching was proposed to help solve this problem by using information derived from 3D models of buildings. Though users of GNSS positioning typically move, previous research has focused on *static* shadow-matching positioning. In this paper, for the first time, *kinematic* shadow-matching positioning is tackled. Kalman filter based shadow matching is proposed and also, in order to overcome some of its predicted limitations, a particle filter is proposed to better solve the problem.

Real-world kinematic experiments were conducted using a smartphone, with four different walking routes taken in a built-up area in London. Conventional GNSS solution exhibits a mean error of 11.25m, compared with single-epoch shadow matching's 6.54m (a 42.0% reduction) and particle filter shadow matching's 2.41m (a 78.6% reduction). The new particle filter shadow-matching algorithm achieves 2-meter accuracy 72.4% of the time and 5-meter accuracy 90.7% of the time. The results also show, as expected, that Kalman filter shadow matching smooths the cross-street positioning error as compared with single-epoch shadow matching, but does not consistently nor significantly improve the accuracy. The accuracy improvement when using the proposed particle filter, on the other hand, shows its potential to improve urban positioning from street level to lane level.

1. INTRODUCTION

GNSS positioning in urban canyons is inaccurate, particularly in the cross-street direction. This is because the high-rise buildings along the streets block the line-of-sights to GNSS satellites (Groves, 2011). Figure 1

illustrates this. However, the accuracy in the cross-street direction is important to lane detection in vehicle navigation, pedestrian navigation, intelligent transportation systems (ITS) and many other applications (You et al., 2008, Broll et al., 2008, Rashid et al., 2005). As 3D building models are becoming more accurate and widely available (Bradbury et al., 2007), some research uses 3D city models to detect and eliminate non-line-of-sight (NLOS) or multipath errors, in order to improve GNSS positioning accuracy (Peyraud et al., 2013, Groves et al., 2012, Obst et al., 2012, Peyret et al., 2011); some research further uses them to correct NLOS or multipath errors (Bourdeau and Sahnoudi, 2012, Suzuki and Kubo, 2013). 3D models can also be used to render scenes and match with the real-world images for localization (Chen et al., 2012). Another line of research that use 3D city models evaluates GNSS performance by employing 3D ray tracing or ray intersection techniques (Ji et al., 2010, Kim et al., 2009, Kleijer et al., 2009, Suh and Shibasaki, 2007, Bradbury, 2007, Bradbury et al., 2007, Wang et al., 2012).

Shadow matching has been proposed to improve the GNSS accuracy in the cross-street direction using buildings shadows as signals of opportunity for positioning (Groves, 2011, Tiberius and Verbree, 2004, Yozevitch, 2012). The expectation for which signals are available can be predicted using a 3D city model. Consequently, by determining whether a signal is being received from a given satellite, the user can localize their position to within one of two areas of the street. Figure 2 illustrates the concept of this solution.

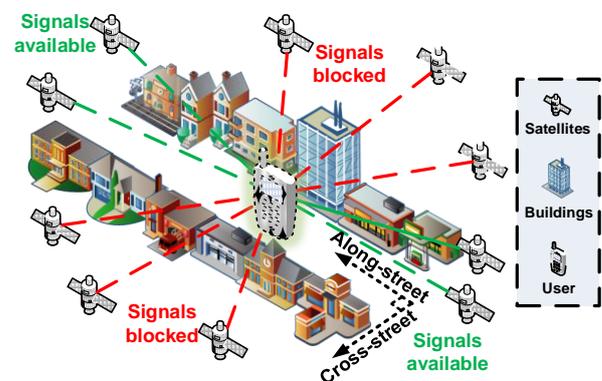


Figure 1: In urban areas, the GNSS positioning inaccuracy results from poor satellite geometry

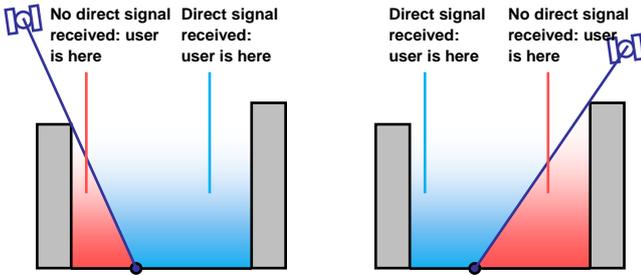


Figure 2: The shadow-matching concept: using direct signal reception to localise position (adapted from Groves (2011)).

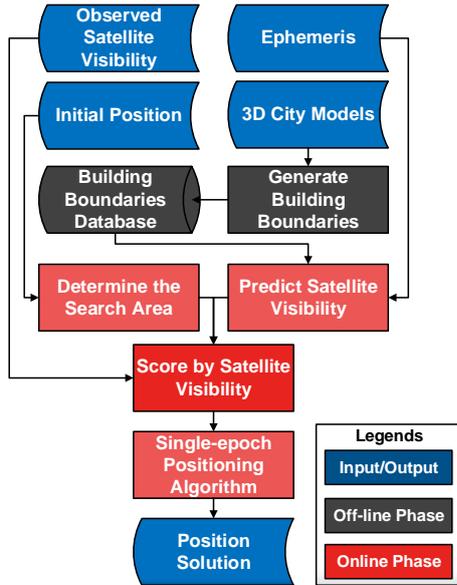


Figure 3 Flowchart of single-epoch shadow matching

The work of this paper is based on the author's doctoral research since 2010, when the shadow matching principle was proposed at UCL (Groves, 2011). The performance of GNSS in urban canyons was first evaluated and verified by 3D city models (Wang et al., 2012). Then, a preliminary shadow-matching algorithm was developed and demonstrated the ability of identifying the correct side of the street (Wang et al., 2011). Furthermore, a scoring scheme has been proposed to account for the effects of satellite signal diffraction and reflection. A full search grid was also implemented and tests were conducted at over 40 locations (Wang et al., 2013b). Moreover, shadow matching has been adapted to work with post-processed smartphone GNSS data (Wang et al., 2013a). Recently, a real-time prototype system has been developed for the Android mobile operation system, which demonstrates the efficiency of the shadow-matching algorithm (Wang et al., 2013c).

This paper is built on the previous work, but it presents a significant progress for shadow matching: two new filtering schemes are proposed that enable the shadow-matching technique to deal with dynamic scenarios. The motivation of the work in this paper comes from the fact that navigation is typically kinematic, whereas the single-

epoch shadow-matching algorithm is valid for static positioning, but not optimized for kinematic cases. This is because in single-epoch shadow matching, GNSS data in each epoch is individually processed, without taking advantage of any knowledge from previous epochs. Previous research (Suzuki and Kubo, 2012) on multi-epoch shadow-matching positioning is not optimized for kinematic positioning. Given that the update rate of a mobile GNSS device is normally 1 Hz, pedestrians and vehicles are not likely to move so fast that the environments change dramatically between consecutive epochs. The single-epoch shadow-matching techniques are thus ignoring important information. In summary, the single-epoch shadow matching techniques are not suitable for kinematic positioning. This is also the scope of this paper, which proposes and implements two different estimation schemes, the Kalman filter and the particle filter, for kinematic shadow-matching positioning.

A new *kinematic* shadow-matching technique is presented in Section 2. Detailed algorithms descriptions of the Kalman filter and the particle filter are given in Section 3 and 4, respectively. A comprehensive assessment of real-world experiments is presented in Section 5, with different criteria applied to compare the performance between the conventional GNSS navigation solution, the single-epoch shadow-matching system solution, and the two new shadow-matching system solutions. Finally, in Section 6 and Section 7, conclusions are drawn and future work is discussed.

2. KINEMATIC SHADOW MATCHING

The single-epoch shadow-matching algorithm, which is suitable only for static positioning scenarios, is briefly reviewed in Section 2.1. Section 2.2 then introduces the system architecture and algorithm flowchart of kinematic shadow matching. Finally, Section 2.3 gives detailed descriptions of the kinematic shadow-matching system.

2.1 Single-epoch shadow matching

The single-epoch shadow matching is valid for static positioning. There are two phases in the single-epoch shadow-matching positioning algorithm – the offline phase (the preparation step), as illustrated in Figure 3 in grey, and the online phase (the positioning process), as illustrated in Figure 3 in red. The input/output are noted in blue. More details of the single-epoch shadow matching can be found in an earlier paper (Wang et al., 2013c).

2.2 Kinematic shadow-matching system architecture and algorithm overview

There are different implementations of a kinematic shadow matching system. In a full implementation, there is a server interacting with a smartphone user. The smartphone first sends a positioning request with an initial position to the server. The initial position may come from GNSS, or Wi-Fi positioning, assuming its accuracy is a few tens of meters. The server then gathers the enhanced map data

(building boundary) that enables in shadow-matching positioning, according to the user's initial position, and sends them to the user. Finally, the smartphone performs the shadow-matching algorithm to acquire a positioning solution. The overall architecture of the shadow-matching system is illustrated in Figure 4.

In the kinematic shadow-matching algorithm, a Kalman filter and a particle filter are added. This new flowchart of the shadow matching algorithm is illustrated in Figure 5.

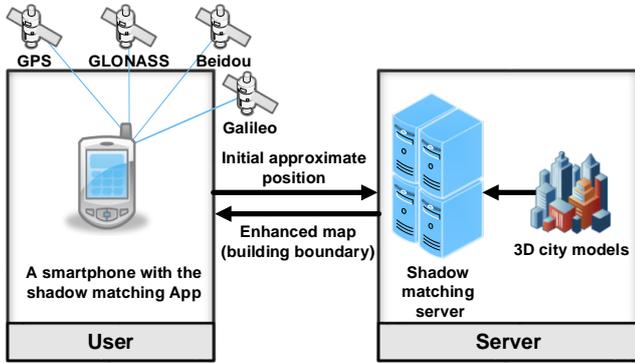


Figure 4: The overall system architecture design

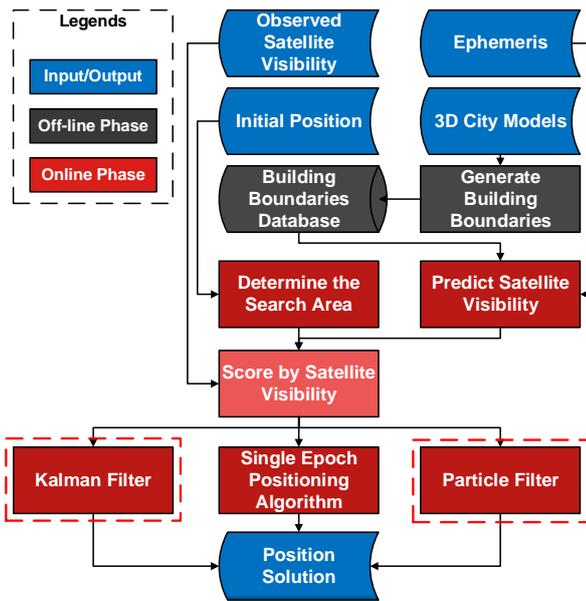


Figure 5 Flowchart of kinematic shadow matching (proposed steps in the paper are surrounded by red frames)

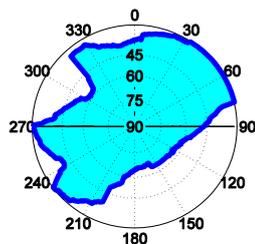


Figure 6: An example of a building boundary as azimuth-elevation pairs in a sky plot. (The centre of the plot correspond to a 90° elevation or normal incidence)

2.3 Kinematic shadow-matching algorithm

This subsection introduces detailed algorithm design of the new kinematic shadow-matching algorithm. There are two phases in the algorithm. The off-line phase generates a grid of building boundaries. The boundaries are from a GNSS user's perspective, with the building's edge determined for each azimuth (from 0 to 360°) as a series of elevation angles. The results from this step show where the building edges are located within an azimuth-elevation sky plot. Figure 6 shows an example of a building boundary computed from a candidate user location. Once the building boundary has been computed, it may be stored and reused in the online phase to predict satellite visibility by simply comparing the elevation of a satellite with the elevation of the building boundary at the same azimuth.

There are four steps in the online phase. At the beginning of the online phase, the search area is defined for the shadow-matching position solution, based on an initial GNSS/Wi-Fi position. A simple implementation might draw a fixed-radius circle centred at the initialized position, but more advanced algorithms might use the knowledge of satellite geometry to optimize the search area.

In the second step, performed at each candidate position, each satellite's elevation is compared with the building boundary elevation at the same azimuth. The satellite is predicted to be visible if the satellite is above the building boundary. With pre-computed building boundaries, this step is computationally efficient.

For the third step, the similarity between predictions and observations of the satellite visibility is evaluated. The candidate positions with the better matches will then be weighted higher in the shadow-matching positioning solution. There are two stages for calculating a score for a candidate position. Firstly, each satellite above the elevation mask angle is given a score, calculated based on the predicted and observed visibility. Secondly, the position scoring function evaluates the overall degree of match between predicted and observed satellite visibility for each possible user position. This is illustrated in (1).

$$f_{pos}(j) = \sum_{i=1}^n f_{sat}(i, j, SS) \quad (1)$$

where $f_{pos}(j)$ is the position score for grid point j ; $f_{sat}(i, j, SS)$ is the score of satellite i at grid point j using a scoring scheme SS . SS is the scoring scheme which defines a score based on predicted and observed satellite visibility. n is the number of satellites above the elevation mask angle.

Different scoring schemes can be applied at this stage, discussed in (Wang et al., 2013b). Essentially, they are designed to mitigate the effect of NLOS reception to

shadow matching. However, they are all based on empirical values and thus may not be appropriate.

A new scoring scheme trained from large amount of real GNSS data is designed (Wang et al.). In this new scheme, the probability of a received signal being a LOS signal is modelled using its SNR value with quadratic fitting.

$$P_o(LOS) = \begin{cases} P_{\min}^o, & S_i \leq S_{\min} \\ a_2 S_i^2 + a_1 S_i + a_0, & S_{\min} \leq S_i \leq S_{\max} \\ P_{\max}^o, & S_i > S_{\max} \end{cases} \quad (2)$$

where P_{\min}^o and P_{\max}^o is the minimum and maximum percentage of $P(LOS)$, respectively; S_i is SNR of the satellite of interest, S_{\min} and S_{\max} is the minimum and maximum SNR with values outside of this range being assigned the closest value; a_0, a_1, a_2 are parameters trained from the large datasets. In this work, the values of these parameters are set as follows: $P_{\min} = 20\%$, $P_{\max} = 90\%$, $S_{\min} = 17$, $S_{\max} = 41$, $a_0 = -1.3$, $a_1 = 0.11$, $a_2 = 0.0013$.

The probability that a satellite prediction matches with the observation can be computed using the following formula:

$$P_m = 1 - P_o(LOS) - P_p(LOS) + 2P_o(LOS)P_p(LOS) \quad (3)$$

where P_m represents the probability that the observation matches the prediction; $P_o(LOS)$ is the probability that an observed signal is a line-of-sight (LOS) signal; $P_p(LOS)$ is the probability that the signal is predicted to be a LOS signal.

The values of $P_p(LOS)$ is set to be 90% if the satellite is predicted to be visible, and 20% if it is predicted to be invisible.

After obtained the probability that a satellite prediction matches with the observation, the scoring result can be calculated using the following formula:

$$f_{sat} = \frac{\log(P_m) - \log(P_{\min}^p)}{\log(P_{\max}^p) - \log(P_{\min}^p)} \quad (4)$$

where P_{\min}^p and P_{\max}^p are the minimum and maximum probability that a signal is predicted to be a LOS signal;

The last step of the shadow-matching algorithm is to generate a positioning solution using the scores from each candidate position. A Kalman filter and a particle filter are used in this step to generate positioning solution. The detailed design of both filters are introduced in Section 3 and 4, respectively.

3. KALMAN FILTER DESIGN

A Kalman filter is used to conduct kinematic shadow-matching positioning over multiple epochs. Combining noisy measurements observed over time should typically be more accurate than using a single noisy measurement. Among the methods that combine multiple measurements, there are two reasons why Kalman filters are firstly chosen in this work to solve the kinematic shadow-matching positioning problem. Firstly, Kalman filters are commonly used in the navigation community to integrate consecutive measurements or data from different sensors, and are proven to be efficient and effective (Groves, 2013). On the other hand, it is frequently observed that in the shadow-matching algorithm, the candidate positions (those that best match predictions) tend to form an approximation of an ellipse, as illustrated in Figure 7. Thus, a Kalman filter should be able to represent this.

The Kalman filter is a state estimation algorithm invented by R. E. Kalman (Kalman, 1960, Brown and Hwang, 1996, Groves, 2013). It is often used to estimate real-time states. In this work, the Kalman filter designed for kinematic shadow-matching positioning consists of 10 steps, as illustrated in Figure 8. There are three phases: initialization, state system propagation and measurement updating, all of which are explained in detail below.

Initialization phase

Step 0: Calculate initial states:

$$\hat{\mathbf{x}}_{k-1}^+ = \begin{pmatrix} n_{k-1}^s \\ e_{k-1}^s \end{pmatrix} \quad (5)$$

where $\begin{pmatrix} n_{k-1}^s \\ e_{k-1}^s \end{pmatrix}^+$ is the last shadow matching solution. When $k-1=0$, it is the conventional GNSS positioning solution.

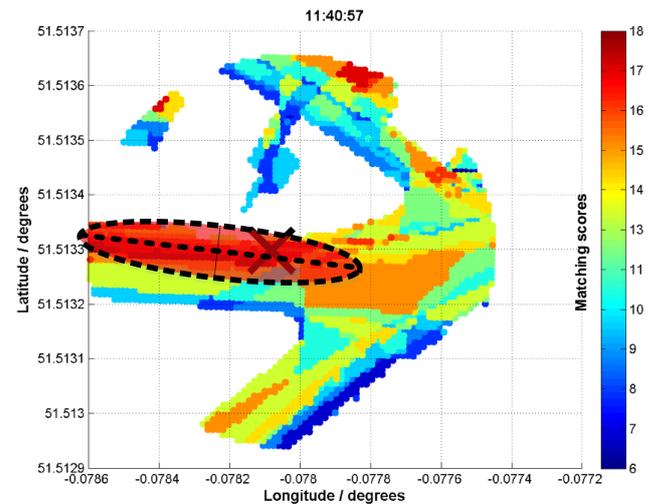


Figure 7. A shadow matching scoring map that shows unambiguous highest-score area, marked in red. (The data was collected at 11:40:57 on 26th, October 2012.)

Calculate error covariance matrix of initial position:

$$\mathbf{P}_{k-1}^+ = \begin{pmatrix} \sigma_{k-1,n}^2 & \sigma_{k-1,ne} \\ \sigma_{k-1,ne} & \sigma_{k-1,e}^2 \end{pmatrix} \quad (6)$$

where $\sigma_{k-1,n}^2$ and $\sigma_{k-1,e}^2$ are the variance of initial position in two horizontal axes (northing and easting), and $\sigma_{k-1,ne}$ is the covariance between the two horizontal axes. E.g. $\sigma_{k-1,n}^2$ and $\sigma_{k-1,e}^2$ can be initialized to be 40 meters, and $\sigma_{k-1,ne}$ are initialized to be 0. This only applies in initialization when $k-1=0$.

System propagation phase

Step 1: State vector time propagation:

In this step, the transition matrix is calculated. The transition matrix is the identity matrix because the states are independent to each other.

$$\Phi_{k-1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (7)$$

Step 2: Error covariance matrix time propagation:

In order to account for the user's movement and for system noise, the error covariance matrix need to be calculated. It can be modelled as:

$$\mathbf{Q}_{k-1} = \begin{pmatrix} S_{vx}\tau_s & 0 \\ 0 & S_{vy}\tau_s \end{pmatrix} \quad (8)$$

where S_{vx} and S_{vy} are the velocity power spectral densities (PSD), for x and y respectively; and τ_s is the time between epochs. In this work, the S_{vx} and S_{vy} are set to be $2\text{m}^2/\text{s}$ (since the user is assumed to be a pedestrian), and τ_s is 1 second.

Step 3: State propagation:

The state vector time propagation can then be conducted using the following formula.

$$\hat{\mathbf{x}}_k^- = \Phi_{k-1} \hat{\mathbf{x}}_{k-1}^+ \quad (9)$$

Step 4: Covariance propagation:

$$\mathbf{P}_k^- = \Phi_{k-1} \mathbf{P}_{k-1}^+ \Phi_{k-1}^T + \mathbf{Q}_{k-1} \quad (10)$$

Measurement update phase

Step 5: Calculate measurement matrix:

The measurement matrix models how the measurement vector varies with respect to the state vector.

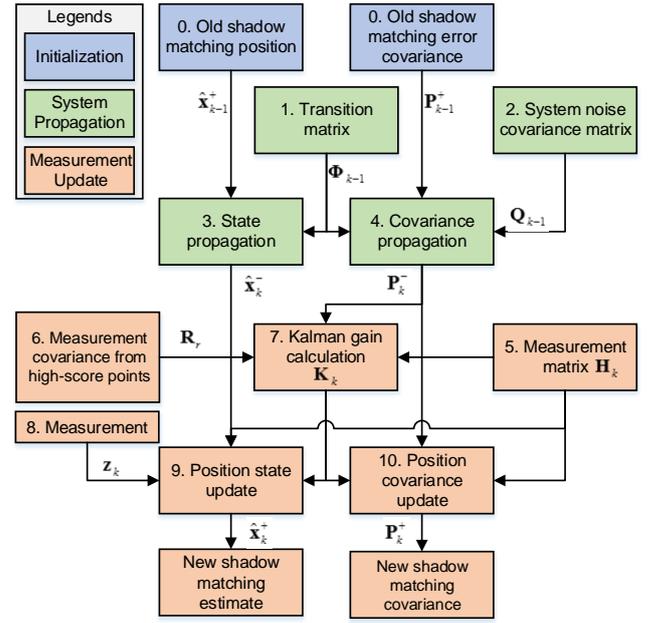


Figure 8: The proposed Kalman filter architecture for kinematic shadow-matching positioning (adapted from (Groves, 2013))

$$\mathbf{H}_k = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (11)$$

Step 6: Calculate measurement variance matrix:

$$\boldsymbol{\mu}_k = \frac{1}{n} \sum_{i=1}^n \mathbf{r}_{C,i} \quad (12)$$

$$\begin{aligned} \mathbf{R}_k &= \begin{pmatrix} \sigma_{k,e}^2 & \sigma_{k,en} \\ \sigma_{k,en} & \sigma_{k,n}^2 \end{pmatrix} \\ &= \frac{1}{n} \sum_{i=1}^n \mathbf{r}_{C,i} - \boldsymbol{\mu}_k \quad \mathbf{r}_{C,i} - \boldsymbol{\mu}_k^T \\ &= \frac{1}{n} \sum_{i=1}^n \mathbf{r}_{C,i} \mathbf{r}_{C,i}^T - \boldsymbol{\mu}_k \boldsymbol{\mu}_k^T \end{aligned} \quad (13)$$

where $\boldsymbol{\mu}_k$ is the averaged candidate grid position; $\mathbf{r}_{C,i}$ is the i -th candidate grid position, these are defined as positions with the highest score in shadow matching; \mathbf{r}_x is the mean of $\mathbf{r}_{C,i}$, n is the number of candidate positions, which is defined as the grid point that has the highest shadow matching score, and \mathbf{R}_k is the measurement variance calculated from high score points, $\sigma_{k,e}^2$ and $\sigma_{k,n}^2$ are the calculated variance of measurement in two horizontal axes, and $\sigma_{k,en}$ is the calculated covariance between the two horizontal axes.

Step 7: Calculate Kalman gain matrix:

$$\mathbf{K}_k = \mathbf{P}_k^{-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \quad (14)$$

Step 8: Formulate measurement:

$$\mathbf{z}_k = \begin{pmatrix} n_k^s \\ e_k^s \end{pmatrix} \quad (15)$$

where $\begin{pmatrix} n_k^s \\ e_k^s \end{pmatrix}$ is the shadow matching positioning solution at the k -th epoch. This shadow matching solution is calculated using the shadow-matching algorithm described in Section 3, except in step 4, where the positioning solution is generated using a k -nearest neighbour method, as described in more details in a previous paper (Wang et al., 2013c).

Step 9: Update state vector:

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k \mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^- \quad (16)$$

Step 10: Update error covariance matrix:

$$\mathbf{P}_k^+ = \mathbf{I} - \mathbf{K}_k \mathbf{H}_k \mathbf{P}_k^- \quad (17)$$

Repeat 1 – 10.

4. PARTICLE FILTER DESIGN

As well as the Kalman filter, a particle filter based algorithm is used in this paper to improve kinematic shadow-matching positioning. Unlike the Kalman filter, the particle filter is a nonlinear non-Gaussian Bayesian estimation technique (Gordon, 1993, Gustafsson, 2002, Thrun et al., 2005).

The standard Kalman filter is a linear Gaussian estimation algorithm. Although extended Kalman filters (EKF) and unscented Kalman filters (UKF) can adapt the Kalman filter to nonlinear systems, the shadow matching system is not only nonlinear, but also multimodal distributed, i.e. there could be ambiguity from the existence of multiple matching areas. This is illustrated in Figure 7, where there are two main best matching areas, marked in red. A multiple-hypothesis Kalman filter may account for this situation, but the Gaussian approximation to the measurement noise and system noise distribution is still a rough approximation, and it is sometimes difficult to determine how many hypotheses models are needed. For example, it can be seen in Figure 9 that apart from there being two major matching areas, there are also several minor matching areas.

Particle filters have a strong potential to better solve the kinematic shadow-matching problem, because addition to non-linear non-Gaussian nature, more importantly, particle filters can estimate multiple hypothesis models. In fact, each particle can be regarded as a hypothesis model. Thus,

a particle filter is used in this work for kinematic shadow-matching positioning.

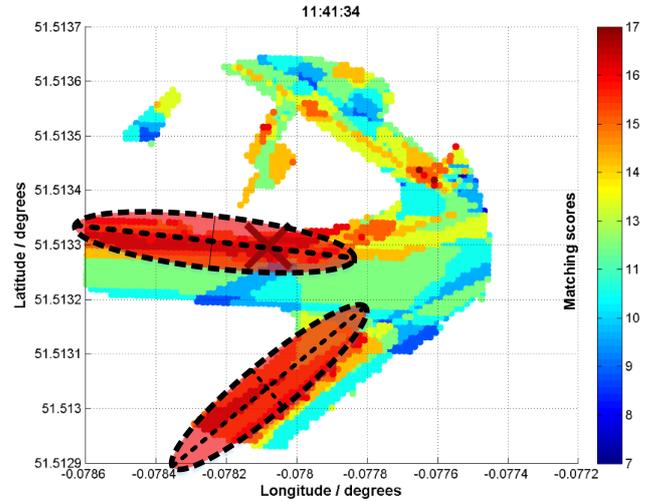


Figure 9. Shadow matching scoring maps that show two ambiguous high-score areas, both marked in red. In this situation, a single-model Kalman filter is not an adequate representation of the multiple distributions; whereas particle filters are adequate. (The data was collected at 11:41:34 on 26th, October 2012.)

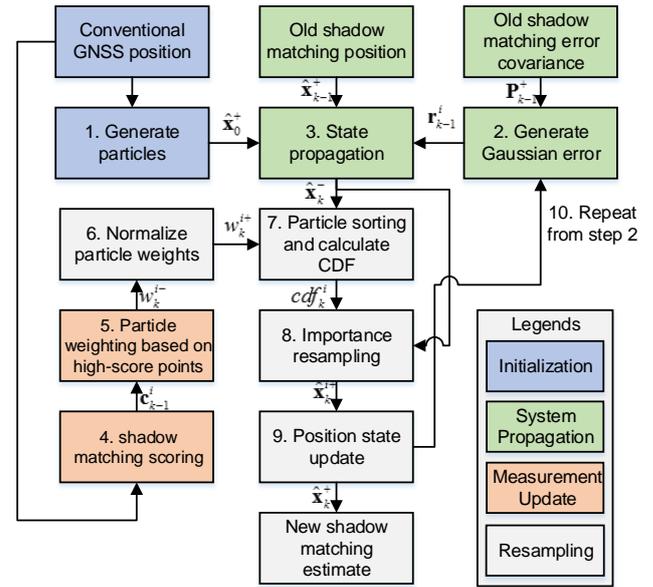


Figure 10: The proposed particle filter architecture for kinematic shadow-matching positioning

An architectural overview of the particle filter is shown in Figure 10. There are four phases, comprising initialization, system updating, measurement updating and resampling. These are detailed in the following descriptions.

Initialization phase

Step 1: Initialization:

Generate n random particles $[\hat{\mathbf{x}}_0^{1+}, \dots, \hat{\mathbf{x}}_0^{n+}]$ in a Gaussian distribution, based on the initial conventional GNSS positioning solution $\begin{pmatrix} n_0^g \\ e_0^g \end{pmatrix}^+$:

$$\hat{\mathbf{x}}_0^+ = \begin{pmatrix} n_0^g \\ e_0^g \end{pmatrix}^+ \quad (18)$$

$$\mathbf{P}_0^+ = \begin{pmatrix} \sigma_{0,n}^2 & \sigma_{0,ne} \\ \sigma_{0,ne} & \sigma_{0,e}^2 \end{pmatrix} \quad (19)$$

$$[\hat{\mathbf{x}}_0^{1+}, \dots, \hat{\mathbf{x}}_0^{n+}] \sim \mathcal{N}(\hat{\mathbf{x}}_0^+, \mathbf{P}_0^+) \quad (20)$$

where in the Gaussian distribution, the initial mean position $\hat{\mathbf{x}}_0^+$ is denoted as $\begin{pmatrix} n_0^g \\ e_0^g \end{pmatrix}^+$, and the error covariance \mathbf{P}_0^+ is denoted as $\begin{pmatrix} \sigma_{0,n}^2 & \sigma_{0,ne} \\ \sigma_{0,ne} & \sigma_{0,e}^2 \end{pmatrix}$. In this work, the $\sigma_{0,n}^2$ and $\sigma_{0,e}^2$ are set to be 20, and $\sigma_{0,ne}$ is set to be 0.

System update phase

Step 2: Generate random noises to account for user motion

In order to account for user motion and unknown changes to the true state, such as mis-modelling and GNSS receiver noise, in the importance weight sampling, random noises $[\mathbf{r}_{k-1}^1, \dots, \mathbf{r}_{k-1}^n]$ for each particle $\hat{\mathbf{x}}_{k-1}^{i+} \in [\hat{\mathbf{x}}_{k-1}^{1+}, \dots, \hat{\mathbf{x}}_{k-1}^{n+}]$ are generated, obeying a Gaussian distribution $\mathcal{N}[\mathbf{0}, \mathbf{P}_{k-1,r}]$, where $\mathbf{P}_{k-1,r}$ is the covariance matrix.

$$\mathbf{P}_{k-1,r} = \begin{pmatrix} \sigma_{k,r,n}^2 & \sigma_{k,r,ne} \\ \sigma_{k,r,ne} & \sigma_{k,r,e}^2 \end{pmatrix} \quad (21)$$

$$[\mathbf{r}_{k-1}^1, \dots, \mathbf{r}_{k-1}^n] \sim \mathcal{N}[\mathbf{0}, \mathbf{P}_{k-1,r}] \quad (22)$$

Step 3: The random noises $[\mathbf{r}_{k-1}^1, \dots, \mathbf{r}_{k-1}^n]$ are then added to the particles $[\hat{\mathbf{x}}_{k-1}^{1+}, \dots, \hat{\mathbf{x}}_{k-1}^{n+}]$ to update their state.

$$[\hat{\mathbf{x}}_k^{1-}, \dots, \hat{\mathbf{x}}_k^{n-}] = [\hat{\mathbf{x}}_{k-1}^{1+}, \dots, \hat{\mathbf{x}}_{k-1}^{n+}] + [\mathbf{r}_{k-1}^1, \dots, \mathbf{r}_{k-1}^n] \quad (23)$$

Measurement update phase

Step 4: for the k -th epoch, perform the GNSS shadow matching algorithm, which is initialized at the last particle-filter positioning solution $\hat{\mathbf{x}}_{k-1}^+$. For the first epoch, i.e. $k=1$, this is the initial conventional GNSS positioning solution $\hat{\mathbf{x}}_0^+$. A grid of m candidate positions with the highest matching scores are chosen from shadow matching, noted as $[\mathbf{c}_{k-1}^1, \dots, \mathbf{c}_{k-1}^m]$.

Step 5: compute the particle weights based on the shadow matching scoring outputs (candidate points). For each particle $\hat{\mathbf{x}}_k^{i-}$ ($1 \leq i \leq n$), its weight w_k^{i-} ($1 \leq i \leq n$) is defined inversely to the shortest Euclidean distance d_k^{i-} ($1 \leq i \leq n$) between this particle and the candidate positions $[\mathbf{c}_{k-1}^1, \dots, \mathbf{c}_{k-1}^m]$, which have been generated in step 4. When the nearest candidate is within 1 meter to the current particle of interest, its distance is considered to be 1 meter.

$$w_k^{i-} = \begin{cases} 1, & (d_k^{i-} < 1 \text{ meter}) \\ d_k^{i-}, & (d_k^{i-} > 1 \text{ meter}) \end{cases} \quad (1 \leq i \leq n) \quad (24)$$

Step 6: Normalize the weights of each particle, so that

$$0 \leq w_k^{i+} \leq 1 \text{ and } \sum_{i=1}^n w_k^{i+} = 1 \quad (1 \leq i \leq n) \quad (25)$$

$$w_k^{i+} = \frac{w_k^{i-}}{\sum_{i=1}^n w_k^{i-}} \quad (1 \leq i \leq n) \quad (26)$$

Importance weight resampling

Step 7: Incrementally sort the particles $[\hat{\mathbf{x}}_k^{1-}, \dots, \hat{\mathbf{x}}_k^{n-}]$ according to their normalized weights w_k^{i+} ($1 \leq i \leq n$), and compute the cumulative density function (CDF), noted as $[cdf_k^1, \dots, cdf_k^n]$, using the following formula:

$$cdf_k^i = \sum_{j=1}^i w_k^{j+} \quad (27)$$

Step 8: Generate n random variables $[s_k^1, \dots, s_k^n]$ in a uniform distribution:

$$[s_k^1, \dots, s_k^n] \sim \mathcal{U}(0,1) \quad (28)$$

For each $s_k^i \in [s_k^1, \dots, s_k^n]$, find the corresponding particle by choosing the first particle in $[\hat{\mathbf{x}}_k^{1-}, \dots, \hat{\mathbf{x}}_k^{n-}]$ for which its cdf_k^i is bigger than s_k^i ; as a result, a new set of particles $[\hat{\mathbf{x}}_k^{1+}, \dots, \hat{\mathbf{x}}_k^{n+}]$ is generated.

Step 9: The average position of these new particles is deemed the positioning solution:

$$\hat{\mathbf{x}}_k^+ = \frac{1}{n} \sum_{i=1}^n \hat{\mathbf{x}}_k^{i+} \quad (29)$$

Repeat 2 – 9.

5. EXPERIMENTS AND RESULTS ANALYSIS

To evaluate the performance of the proposed new algorithms, a number of kinematic experiments were

conducted in central London using a smartphone. Section 5.1 outlines the 3D city model and the experimental routes, and describes the configuration of the shadow-matching algorithm. Section 5.2 compares positioning results between the conventional GNSS positioning, the single-epoch shadow matching, Kalman filter shadow matching and particle filter shadow matching.

5.1 Experimental configurations

A 3D city model of the Aldgate area of central London, supplied by ZMapping Ltd, was used. The model has a high level of building details and decimeter-level accuracy. Figure 11 shows part of the city model used in this work.

Four experimental routes were selected on Fenchurch Street and Leadenhall Street, a built-up area. Figure 12 shows photos taken at the street, showing the urban environments. Two of the routes, numbered route 1 and 2, were located on opposite sides of Leadenhall Street. The other two routes, named route 3 and 4, were on opposite sides of Fenchurch Street. These routes allow system performance comparison between users at different sides of street. Thus, if the proposed algorithm can determine the user’s position no matter which side the user is at, it is probable that the algorithm is not producing the correct answer by chance. All routes were selected on the footpath close to the traffic lanes. Figure 13 shows an aerial view of the experimental routes in a satellite image from Google Earth. The truth model in this experiment is set using the 3D city model. A pedestrian walked in steady speed from the start to the end of each route. Table 1 summarises the experimental configurations for each of the four routes.

Before the experiment, the offline phase generated a grid with 1-meter spacing. Indoor points were then eliminated and building boundaries were determined at outdoor points. The building boundaries were stored in a database.

Using the GNSS data-recording app adapted from earlier work (screenshot shown in Figure 14), a Samsung Galaxy S3 smartphone was used to record GNSS data with a frequency of 1Hz. Both GPS and GLONASS observations were recorded, including satellite visibility information and positioning results from the smartphone GNSS chip. 500 particles are used in the particle filter.

5.2 Positioning performance assessment

In this section, the overall performance of the shadow-matching positioning system is assessed and compared with the conventional GNSS solution from the GNSS chip in the Samsung Galaxy S3 smartphone.

To compare the performance of shadow matching against the conventional GNSS positioning solution, the position errors are transformed from local coordinates (northing and easting) to the along-street and across-street directions. The cross-street direction is the main concern in this paper, because the sensitivity in this direction matters to many

applications, including pedestrian navigation, vehicle navigation, and intelligent transportation systems.

Figure 15 (left) shows the positioning results of the conventional GNSS navigation solution from the smartphone GNSS chip, compared with the three shadow matching algorithms: single-epoch shadow matching, Kalman filter shadow matching, and particle filter shadow matching, expressed as errors in the cross-street direction. The right graphs in the same figure shows the histogram of the error distribution.

There are a few interesting points that can be observed from this figure. Firstly, the overall characteristics of shadow matching and conventional GNSS solutions are very different. The conventional GNSS solutions are smoother, as smoothing algorithms are commonly used in navigation GNSS chipsets. However, the shadow matching solutions, no matter which version, tend to be closer to zero, which means their accuracy is better, though their consistency is less stable.

Secondly, the Kalman Filter shadow matching sometimes outperforms single epoch shadow matching by smoothing out anomalies, but not consistently better. The Kalman Filter shadow matching is more like a smoother, which in many cases has fewer variations compared to the single-epoch shadow matching. When considering all routes taken in kinematic experiments, the smoothing effect frequently achieves a better accuracy, showing the benefit of using the Kalman Filter shadow matching. In other words, the improvement of the Kalman filter compared to single-epoch shadow matching is in smoothing, though not very significantly in accuracy.

Thirdly, the particle filter shadow matching significantly outperforms all three other methods, including the Kalman filter shadow matching. For all the routes, the particle filter shadow matching positioning results show a clear peak at zero-error.

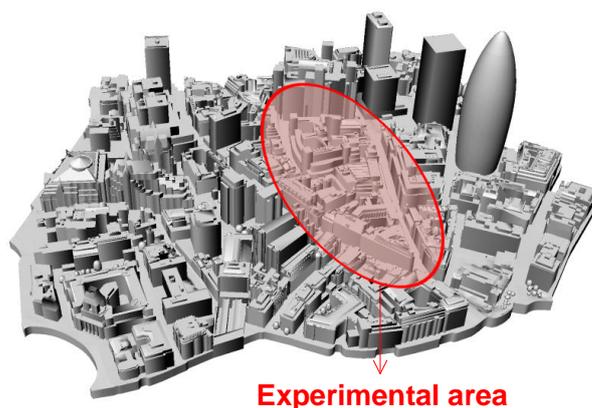


Figure 11. The 3D city model used in shadow matching experiments. The area marked in red is where the four routes of experiments were conducted.

Table 1 Summary of experimental configurations

Route name	Start time	Start position	End time	End position	Azimuth (Degree)
Route 1	12:06:39	A (-0.07889849, 51.51333651)	12:08:06	B (-0.07806907 51.51329938)	97.6
Route 2	11:34:25	C (-0.07891204, 51.51325212)	11:36:11	D (-0.07802733 51.51320084)	97.6
Route 3	11:47:17	E (-0.07903944, 51.51256939)	11:49:23	F (-0.07819828 51.51299566)	62.0
Route 4	11:57:40	G (-0.07899008 51.51250454)	11:59:37	H (-0.07812412 51.51291452)	62.0



Figure 12 A photo taken at the experimental area, showing that it is an urban environment

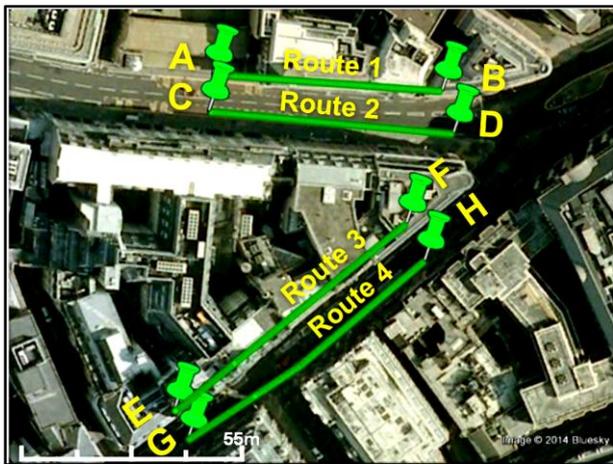


Figure 13 The 4 experimental routes illustrated in a satellite image



Figure 14 A screen shot of the developed Android app which is used to record GNSS data for shadow matching (including satellite PRN, signal-to-noise ratio, azimuth, elevation and conventional GNSS positioning solution).

It is clearly demonstrated that the particle filter shadow matching solution has improved on the conventional positioning error, in the across-street direction, from typically 10 - 40 meters to within 2-3 meters in route 1, 2, and 4. In route 3, the particle filter shadow matching is also better than conventional GNSS solutions in most epochs.

In order to evaluate the performance across all of the epochs, a statistical analysis was performed. Two indicators, mean absolute deviation (MAD) and standard deviation (SD), were used to evaluate the performance from a statistical perspective. MADs for conventional GNSS and shadow matching (single epoch, Kalman filter, particle filter) are compared in Figure 16. Bars in the left sub-figure show MADs for each route and the right sub-figure shows the mean MADs over all routes. The error are shown for using the conventional GNSS, single epoch shadow matching, Kalman filter shadow matching, and particle filter shadow matching algorithms, respectively. It should be noted that the statistics typically cover a 2-minute (120 seconds) observation period, during which the constellation geometry changes slowly, so the results are highly correlated, temporally, allowing consistency of the system to be evaluated in a constantly changing environmental layout.

$$MAD = \frac{1}{n} \sum_{i=1}^n abs(x_i) \quad (30)$$

$$SD = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (31)$$

where x_i is the cross-street positioning error at the epoch, \bar{x} is the mean cross-street positioning error, and n is the number of epochs in that route.

It is shown in Figure 16 that the across street positioning performance of particle filter shadow matching is significantly better than the conventional GNSS positioning. The single epoch shadow-matching algorithm reduced the mean cross-street error, compared with conventional GNSS solutions, from 11.25m to 6.54m – by 42.0%, averaged over routes 1, 2, 3, and 4. Kalman filter shadow matching has similar performance to single epoch shadow matching. The particle filter shadow-matching algorithm reduced the mean cross-street error to 2.41m – by 78.58%, compared with conventional GNSS positioning solutions.

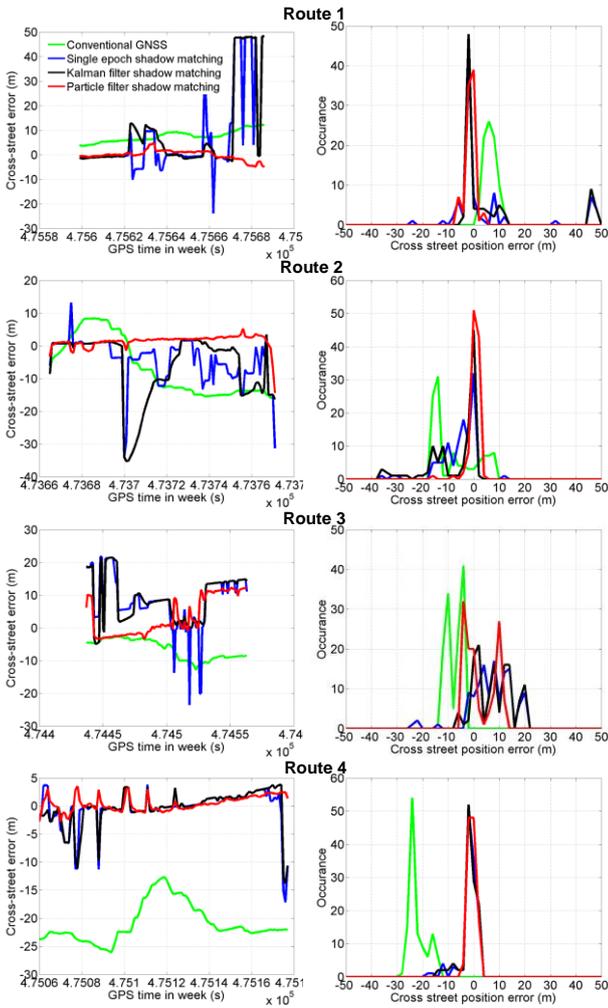


Figure 15 Positioning error of conventional GNSS, shadow matching, Kalman filter and particle filter, in cross-street direction.

In Table 2, the standard deviation (SD) is compared between different methods, showing the new particle filter shadow matching solution also outperforms the other methods. Since the standard deviations of single epoch shadow matching and Kalman filter shadow matching are larger than their averaged MADs, the latter are insignificant, given that the datasets of this experiment are not very large.

Further statistical comparisons were conducted to assess the positioning performance as success rate of achieving a threshold of certain accuracy in the cross-street direction, and the results are shown in Figure 17. As the street is around 10m wide, a positioning accuracy of less than 5m is considered good enough to distinguish sides of streets, while a positioning accuracy better than 2m is considered good enough to distinguish the footpath from a traffic lane.

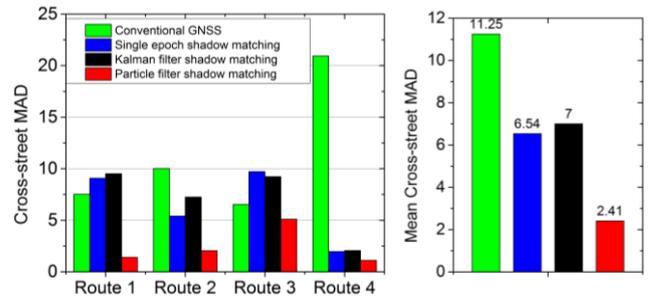


Figure 16 Mean absolute deviation (MAD) of cross-street positioning errors using different methods

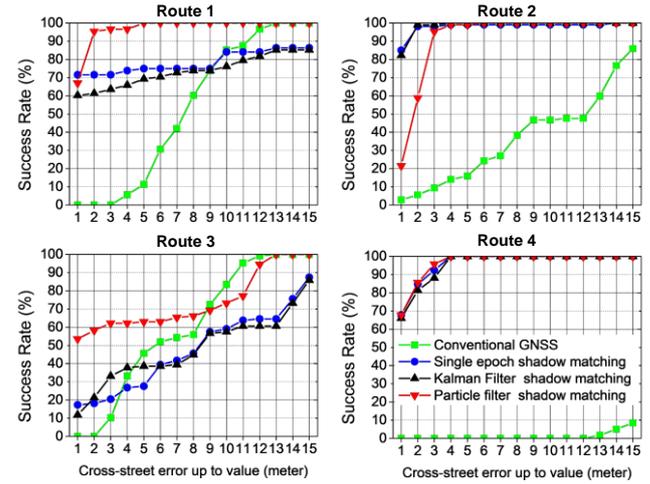


Figure 17 Success rate comparison between different positioning methods in each route

Table 2 Comparison of cross-street error standard deviation between different methods

Route	Conventional GNSS (m)	Shadow matching		
		Single epoch (m)	Kalman filter (m)	particle filter (m)
Route 1	2.32	16.97	17.01	1.92
Route 2	8.76	6.94	9.96	2.15
Route 3	3.04	8.07	7.06	5.92
Route 4	3.43	3.50	3.34	1.34
Average	4.39	8.87	9.34	2.83

It can be seen from Figure 17, Table 3 and Table 4 that, Kalman filter shadow matching performs similarly to single epoch shadow matching, but both better than conventional GNSS. Particle filter shadow matching performs best as, for determining the correct side of a street, its success rate in these results is 72.4%, while for Kalman filter shadow matching it is 65.8%, and for conventional GNSS it is a poor 1.40%. The success rate of distinguishing a footpath from a traffic lane is 90.7% for particle filter shadow matching, 77.0% for Kalman filter shadow matching, and merely 18.2%, for conventional GNSS positioning.

Table 3 Success rate of achieving the 2-meter accuracy threshold in cross-street direction comparison between different methods

Route	Conventional GNSS	Shadow matching		
		Single epoch	Kalman filter	particle filter
Route 1	0	0.716	0.614	0.955
Route 2	0.056	0.981	0.991	0.607
Route 3	0	0.181	0.213	0.520
Route 4	0	0.847	0.814	0.814
Average	0.014	0.681	0.658	0.724

Table 4 Success rate of achieving the 5-meter accuracy threshold in cross-street direction comparison between different methods

Route	Conventional GNSS	Shadow matching		
		Single epoch	Kalman filter	particle filter
Route 1	0.114	0.750	0.693	1
Route 2	0.159	0.991	1	1
Route 3	0.457	0.276	0.386	0.646
Route 4	0	1	1	0.983
Average	0.182	0.754	0.770	0.907

6. CONCLUSIONS

While single-epoch shadow matching works only for static applications, now, for the first time, the kinematic shadow matching is tackled in this paper. Two approaches are proposed: using a Kalman filter and using a particle filter. Compared with single-epoch shadow matching, both enable position estimation of moving objects (pedestrians or vehicles with GNSS enabled devices) using data from multiple epochs.

Real-world kinematic experiments were conducted in an urban area in London, UK. An Android application was developed to record the GNSS data stream on a smartphone. Four different routes, on two different streets, were tested by a pedestrian, providing a performance assessment of the new system. Evaluation and comparison between four methods (conventional GNSS, conventional single-epoch shadow matching, Kalman filter shadow matching, and particle filter shadow matching) was conducted.

Compared with single-epoch shadow matching, the Kalman filter is proven able to smooth the results, as can be seen in Figure 15. In terms of accuracy, Kalman filter shadow matching is similar to single epoch shadow matching. Compared with conventional GNSS, it reduces the mean cross-street positioning error from 11.25m to 7m – by 37.8%. It also improves the success rate of distinguishing the footpath from a traffic lane (2-meter-

error) from 1.40% to 65.8%, and the success rate of distinguishing sides of streets (5-meter-error) from 18.2% to 77.0%.

Since a Kalman filter has its limitations, including linear and Gaussian distribution assumptions, a particle filter, a non-linear non-Gaussian estimator, is tested. Experimental results from the particle filter show a clear boost in positioning performance. Compared with Kalman filter shadow matching (which is very similar in performance to single-epoch shadow matching), particle filter shadow matching increases the 2-meter-error success rate from 65.8% to 72.4%, and the 5-meter-error success rate from 77.0% to 90.7% – and reduces the average cross-street positioning error to 2.84m – by 74.8% compared to conventional GNSS.

In summary, the four experimental routes together prove that the Kalman filter can smooth single-epoch shadow matching, though accuracy is no better than for single-epoch shadow matching. The proposed particle filter, in contrast, boosts positioning accuracy significantly compared with conventional GNSS. Thus, particle filter shadow matching has the potential to improve mobile device positioning in urban areas from street level to lane-level.

7. DISCUSSIONS AND FUTURE WORK

As discussed in the introduction, meter-level cross-street accuracy for GNSS positioning in urban areas would benefit a variety of applications from Intelligent Transportation Systems (ITS) and lane identification in navigation systems, higher resolution location-based advertisement (LBA), step-by-step guidance (for the visually impaired and for tourists) to many other location-based services (LBS).

It should also be noted that the system does not require real-time rendering of 3D scenes or any additional hardware, making it power-efficient and cost-effective. As shadow matching is a highly parallelizable algorithm, parallel processing techniques can be applied to it, exploiting the increasing availability of multi-core processors in smartphones. The system is also extensible to work with Galileo and Beidou (Compass).

The shadow matching positioning system is a suitable complementation to conventional GNSS positioning. As shadow matching improves the cross-street accuracy significantly, it shows a high potential to be combined with conventional GNSS for better overall performance (Groves et al., 2012).

In the future, shadow matching can be implemented as part of an intelligent positioning system (Groves et al., 2012), along with techniques including Wi-Fi positioning, IMU, gyroscopes, pressure sensors, Bluetooth low energy, and context-aware techniques (Groves et al., 2013), in order to improve the overall positioning performance.

Acknowledgement

The author is very grateful to his primary supervisor, Dr. Paul D Groves, for his insights into shadow matching and Kalman filters, and kind support over the 4-year course of this project. The author is also very grateful to his secondary supervisor Prof. Marek K Ziebart for his insights into using 3D city models, his valuable suggestions and providing the funding opportunity to the author to start this project.

The author would like to thank Mr. Henry Martin and Mr. Haraldur Gunnarsson, both are UCL PhD student, for their help in the raw data collection, and Mr. Haraldur Gunnarsson for reviewing the English of the paper. The author also thanks Prof. Ruizhi Chen at Texas A&M University Corpus Christi, for his kind and valuable advices in reviewing the paper, and Dr. Ramsey Faragher at University of Cambridge, for his advices on using particle filters.

The author thanks University College London (UCL) and the Chinese Scholarship Council for jointly providing funding to support this research.

References

- BOURDEAU, A. & SAHMOUDI, M. 2012. Tight Integration of GNSS and a 3D City Model for Robust Positioning in Urban Canyons. *ION GNSS*. Nashville, Tennessee.
- BRADBURY, J. 2007. Prediction of Urban GNSS Availability and Signal Degradation Using Virtual Reality City Models. *Proceedings of the 20th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2007)*. Fort Worth, TX.
- BRADBURY, J., ZIEBART, M., CROSS, P. A., BOULTON, P. & READ, A. 2007. Code Multipath Modelling in the Urban Environment Using Large Virtual Reality City Models: Determining the Local Environment. *The Journal of Navigation*, 60, 95-105.
- BROLL, W., LINDT, I., HERBST, I., OHLENBURG, J., BRAUN, A. K. & WETZEL, R. 2008. Toward Next-Gen Mobile AR Games. *Computer Graphics and Applications, IEEE*, 28, 40-48.
- BROWN, R. G. & HWANG, P. Y. C. 1996. *Introduction to Random Signals and Applied Kalman Filtering*, Wiley.
- CHEN, R., WANG, Y., PEI, L., CHEN, Y. & VIRRANTAU, K. 2012. 3D Smartphone Navigation Using Geocoded Images. *GPS World*.
- GORDON, N. J., D. J. SALMOND, AND A. F. M. SMITH 1993. A Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation. *Proc. IEE Radar Signal Process*, 140, 107-113.
- GROVES, P. D. 2011. Shadow Matching: A New GNSS Positioning Technique for Urban Canyons *The Journal of Navigation*, 64, pp417-430.
- GROVES, P. D. 2013. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems (2nd edition)*, Boston, London, Artech House.
- GROVES, P. D., JIANG, Z., WANG, L. & ZIEBART, M. 2012. Intelligent Urban Positioning using Multi-Constellation GNSS with 3D Mapping and NLOS Signal Detection. *ION GNSS 2012*. Nashville, Tennessee.
- GROVES, P. D., MARTIN, H., VOUTSIS, K., WALTER, D. & WANG, L. 2013. Context Detection, Categorization and Connectivity for Advanced Adaptive Integrated Navigation *ION GNSS+*. Nashville, Tennessee.
- GUSTAFSSON, F. E. A. 2002. Particle Filters for Positioning, Navigation and Tracking. *IEEE Trans. on Signal Processing*, 50, 425-437.
- JI, S., CHEN, W., DING, X., CHEN, Y., ZHAO, C. & HU, C. 2010. Potential Benefits of GPS/GLONASS/GALILEO Integration in an Urban Canyon - Hong Kong. *The Journal of Navigation*, 63, 681-693.
- KALMAN, R. E. 1960. A New Approach to Linear Filtering and Prediction Problems. *ASME Transactions, Series D: Journal of Basic Engineering*, 82, 35-45.
- KIM, H. I., PARK, K. D. & LEE, H. S. 2009. Development and validation of an integrated GNSS simulator using 3D spatial information. *Journal of the Korean Society of Surveying Geodesy Photogrammetry and Cartography*, 27, 659-667.
- KLEIJER, F., ODIJK, D. & VERBREE, E. 2009. Prediction of GNSS availability and accuracy in urban environments - Case study Schiphol Airport. Chapter 23 of "Location Based Services and TeleCartography II" *In: REHRL, G. G. A. K. (ed.). Springer-Verlag, Berlin Heidelberg*
- OBST, M., BAUER, S. & WANIELIK, G. 2012. Urban multipath detection and mitigation with dynamic 3D maps for reliable land vehicle localization. *Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION*.
- PEYRAUD, S., BÉTAILLE, D., RENAULT, S., ORTIZ, M., MOUGEL, F., MEIZEL, D. & PEYRET, F. 2013. About Non-Line-Of-Sight Satellite Detection and Exclusion in a 3D Map-Aided Localization Algorithm. *Sensors*, 13, 829-847.
- PEYRET, F., BÉTAILLE, D. & MOUGEL, F. 2011. Non-Line-Of-Sight GNSS signal detection using an on-board 3D model of buildings. *11th International Conference on ITS Telecommunications (ITST)*.

- RASHID, O., COULTON, P. & EDWARDS, R. 2005. Implementing Location Based Information/Advertising for Existing Mobile Phone Users in Indoor/Urban Environments. *Proceedings of the International Conference on Mobile Business*. IEEE Computer Society.
- SUH, Y. & SHIBASAKI, R. 2007. Evaluation of satellite-based navigation services in complex urban environments using a three-dimensional GIS. *IEICE Transactions on Communications*, E90-B, 1816-1825.
- SUZUKI, T. & KUBO, N. GNSS Positioning with Multipath Simulation using 3D Surface Model in Urban Canyon. ION GNSS 2012, September 17 - 21, 2012 2012 Nashville, TN.
- SUZUKI, T. & KUBO, N. Correcting GNSS Multipath Errors Using a 3D Surface Model and Particle Filter. ION GNSS 2013, September 2013 2013 Nashville, TN. 1583 - 1595.
- THRUN, S., FOX, D. & BURGARD, W. 2005. *Probabilistic Robotics*, MIT Press.
- TIBERIUS, C. & VERBREE, E. 2004. GNSS positioning accuracy and availability within Location Based Services: The advantages of combined GPS-Galileo positioning. *NaviTec*.
- WANG, L., GROVES, P. & ZIEBART, M. Smartphone Shadow Matching for Better Cross-street GNSS Positioning in Urban Environments *In preparation for journal submission*.
- WANG, L., GROVES, P. & ZIEBART, M. 2011. GNSS Shadow Matching Using A 3D Model of London. *European Navigation Conference*. Grange Tower Bridge, London
- WANG, L., GROVES, P. & ZIEBART, M. 2013a. Shadow Matching: Improving Smartphone GNSS Positioning in Urban Environments. *China Satellite Navigation Conference (CSNC) 2013 Proceedings*. Springer Berlin Heidelberg.
- WANG, L., GROVES, P. D. & ZIEBART, M. K. 2012. Multi-Constellation GNSS Performance Evaluation for Urban Canyons Using Large Virtual Reality City Models. *The Journal of Navigation*, 65, 459-476.
- WANG, L., GROVES, P. D. & ZIEBART, M. K. 2013b. GNSS Shadow Matching: Improving Urban Positioning Accuracy Using a 3D City Model with Optimized Visibility Prediction Scoring. *Journal of The Institute of Navigation*, 60.
- WANG, L., GROVES, P. D. & ZIEBART, M. K. 2013c. Urban Positioning on a Smartphone: Real-time Shadow Matching Using GNSS and 3D City Models. *ION GNSS 2013*. Nashville, Tennessee.
- YOU, Y., CHIN, T. J., LIM, J. H., CHEVALLET, J.-P., #233, COUTRIX, L. & NIGAY, L. 2008. Deploying and evaluating a mixed reality mobile treasure hunt: Snap2Play. *Proceedings of the 10th international conference on Human computer interaction with mobile devices and services*. Amsterdam, The Netherlands: ACM.
- YOZEVITCH, R. B. M., B. ; LEVY, H. 2012. Breaking the 1 meter accuracy bound in commercial GNSS devices. *Electrical & Electronics Engineers in Israel (IEEEI), 2012 IEEE 27th Convention of*.