Journal of Global Positioning Systems (2013)
Vol.11, No.1 :116-126
DOI: 10.5081/jgps.11.2.116

Journal of Global Positioning Systems
2012
© CPGPS

Evaluations on 3D Personal Navigation based on Geocoded Images in Smartphones

Yiwu Wang[1], Ruizhi Chen[2,3], Yuwei Chen[2,4], Ling Pei[2], Hannu Hyyppä[1], Juha Hyyppä[4], Lingli Zhu[4], Kirsi Virrantaus[1]
[1]*Department of Real Estate, Planning and Geoinformatics, Aalto University, Finland*
[2]*Department of Navigation and Positioning, Finnish Geodetic Institute, Finland*
[3]*Conrad Blucher Institute for Surveying & Science, School of Engineering and Computer Science, Texas A&M University Corpus Christi, USA*
[4]*Department of Remote Sensing and Photogrammetry, Finnish Geodetic Institute, Finland*

Abstract

3D personal navigation is becoming a standard feature in smartphone platform, which develops in a fast speed nowadays. However, the hardware restrictions of smartphone may degrade the 3D rendering performance, and such real-time operation is not an energy-efficient procedure on smartphone, because heavy computation consumes a lot of power, which is crucial for a smartphone equipped with limited capacity battery. This paper presents a novel solution utilizing geocoded images instead of 3D models to mitigate these technical restrictions on the smartphone. To demonstrate the performance and the improvement of the proposed solution, evaluations are carried out in term of positioning accuracy, resource consumption, efficiency, visualization, and labour costs. The results show that the proposed solution has overwhelming advantages in all these comparisons. This solution also has the capability of achieving a higher frame rate and has a better visualization performance as well. In addition, the proposed solution provides an optional way to decrease the labour costs and hardware investment to build up a similar but quick application by utilizing photos instead of complex 3D model construction for a small-scale area personal navigation application.

Keywords: personal navigation, geocoded images, evaluation, smartphone

1. Introduction

Navigation in smartphones has been in spot light for the past decade. 3D personal navigation technology gives smartphone users totally new experience, which can help mobile users be familiar with surroundings and find their destinations easier comparing with traditional 2D navigation. Therefore, it has a huge market potential in the future.

In the traditional 2D navigation application, road information displayed is composed with points, lines, polygons and texts abstractly. A foreigner, who cannot understand the relevant text indication of the 2D navigation application, will be easily lost his/her right positions. However, even a stranger, who is completely unfamiliar with the visited city and doesn't understand the local language, can easily recognize surroundings from a realistic 3D street views, thus to find her/his location. Fig. 1 briefly shows the difference between 2D and 3D navigation.

Figure 1: Comparison between 2D navigation and 3D navigation

3D technology in mobile devices has been remarkable maturity and popularity in recent years. Several developers have implemented or prototyped various 3D navigation applications in smartphone platform.

In 2001, Rakkolainen (Rakkolainen, et al., 2001) proposed a system that applied 3D Virtual Reality Modelling Language (VRML) models of a city centre with a 2D map of related area on a personal digital assistant (PDA) terminal. The project studied the effects of 3D graphics on navigation and way finding in a city environment. But due to slow rendering speed in the PDA, the project was simulated on a laptop.

In 2005, Burigat (Burigat, et al., 2005) investigated a 3D navigation application for location-aware presentation in a smartphone for tourists. However, the computational capability of the smartphone could not meet the requirements for keeping the good performance and right representation simultaneously. Furthermore, the image displayed did not adequately correspond to the actual viewpoint in some situations because of low accuracy of on-board GPS data.

Nurminen (Nurminen, 2006) developed a solution which turned a photorealistic VRML model into an efficient real-time 3D map running on a smartphone without hardware acceleration in 2006. The 3D models utilized in smartphone were downloaded via 3G networks in an optimized, progressive download scheme. However, the download scheme required strictly on the networking transferring speed and that might not an economical solution for a foreign user because of expensive date transmission fee.

In 2007, Coors (Coors, et al., 2007) presented a mobile navigation application by using 3D city models. The application provided navigation support with cognitive semantic route descriptions by using 3D landmarks. However, the project did not implement the navigation function in a smartphone, but simulated the virtual navigation on a computer.

In 2010 Shanghai EXPO, an application named as Three-Dimensional Personal Navigation (3D PN) and Location-Based Service (LBS) in a smartphone was demonstrated successfully in the Shanghai EXPO Park (Liu, et al., 2010). The project generated a 3D city model through aerial photogrammetry images and point clouds collected by a car-borne laser scanner (Kukko, et al., 2007). We developed a 3D indoor and outdoor seamless navigation and positioning system, which consisted of GPS receiver, multiple Micro-electromechanical Systems (MEMS) sensors, wireless network positioning chip and map matched technology (Chen, et al., 2009b). A Nokia 6710 smartphone, which was installed with the application, could accurately navigate a mobile user to the destination in the Expo Park (Pei, et al., 2011). However, due to the hardware restriction, the insufficient rendering speed causes several problems, e.g. low refreshing rate, and long initialization time. Based on the 3D PN LBS program which is based on 3D model, we propose a geocoded image based solution to mitigate the hardware restrictions to present a better user experience.

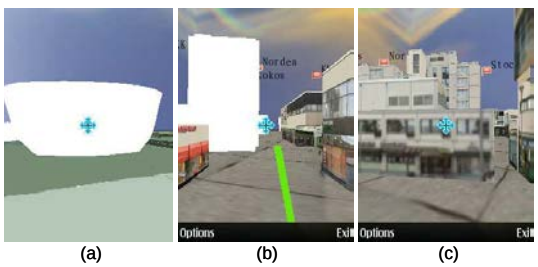
This paper is organized as follows. Section 2 addresses the motivation of this study. Section 3 describes the architecture of the navigation component and discusses the procedure of the geocoding and visualization. Section 4 presents the field test. Section 5 provides and evaluates the test results. The conclusions are made and the further studies are outlined in Section 6.

2. Motivation

Although techniques of smartphones and other handheld devices develop rapidly, physical conditions still limit the 3D visualization application in those devices. A desktop or a workstation often relies on dedicated hardware and architecture design, e.g. Intel Accelerated Graphics Port (AGP) interface, for 3D graphics to achieve real-time performance. It is still a challenging task for a handheld device to render 3D graphics as the applications in 3D are extremely resource-demanding programs (Pulli, 2006).

Comparing with computer platforms, smartphone platforms have the various hardware restrictions such as CPU power, memory size, and bus bandwidth. The hardware capabilities of a smartphone nowadays are similar to those of the desktop PCs about 10 years ago (Chun, et al., 2009). Low bus bandwidth in smartphone affects the vertex per polygon count and frame per second (FPS) in applications (Chehimi, et al., 2006). And the data transferring from CPU to graphics routines are also restricted due to low bus bandwidth of smartphones. Meanwhile, many enticing 3D effects such as 3D wrapping, involve floating-point arithmetic. But most smartphones do not have floating-point units in their processors. Instead, such floating point operations are emulated in software which spend 50 times CPU load comparing with integer operations (Wagner, et al., 2007).

In the previous 3D PN LBS application, its visualization engine employs a simplified 3D model with compressed textures in order to accelerate the rendering speed. Even though, the rendering performance is still restricted by the hardware limitations of smartphone platform. Insufficient rendering causes a long period of initialization. In the case of Shanghai EXPO, it takes more than 30 seconds to complete the initialization of the application, which includes more than 200 diverse, unique and complex 3D pavilion models. The texture of the building is temporary missing because of the insufficient rendering speed as shown in Fig. 2 (a).



(a) (b) (c)

Figure 2: Screen snapshots in the 3D PN application which is based on 3D models.

The similar situation happens during navigation procedure. In order to decrease the resource consumption, 3D models and its textures are loaded tile by tile. In some worse case (see Fig. 2 (b)), 3D models cannot be rendered in time because of the limitation of computation power. Such display lag degrades the user experience of navigation because it is difficult to obtain textual information in time.

Although the compressed texture and the simplified model are employed to accelerate the rendering speed, it results in low resolution model and texture, which makes the building facade too vague to be identified as Fig.2 (c) present (Wang et al., 2012).

To achieve a better visualization performance and enhance energy efficiency by mitigating the technical restrictions of the 3D model based solution in a smartphone for small-scale area application, this paper presents a new approach whose visualization engine is no longer dependent on 3D models, but utilizes the geocoded images extracted from 3D models on a computer platform (Wang, et al., 2012, Chen, et al., 2012c). Geocoding is a procedure of transforming the objects associated with georeference from other geographic data (Goldberg, 2008). The motivation of this paper is to develop a system using geocoded images, and build up a workable 3D pedestrian navigation solution on the smartphone platform. Furthermore, a series of tests are carried out to prove the image-based solution has advantages than the previous 3D models based solution.

3. System Development

By modifying the navigation component of the 3D PN LBS application, the proposed geocoded image based personal navigation system is developed on the Nokia Series 60 (S60) smartphone platform. The system consists of a locator component, a navigation component, a visualization component, and a LBS component (Pei et al., 2009a).

The locator component transfers positioning information to the navigation component in the software. Such positioning information includes the location and heading information, obtaining either from the built-in positioning sensors in the smartphone, such as accelerometer, digital compass, or from radio opportunity signal, for example, a GPS receiver, a WLAN (Wireless Local Area Network), or a Bluetooth chip. The navigation component combines the positioning information with road data, and navigates users to find their destination. The visualization component displays the street view of current position. The LBS component constructs the interface to operate the landmarks. For detailed information about this 3D model based personal navigation system, please refer to (Chen, et al., 2008, Chen, et al., 2009b, Chen, et al., 2010a, Wang, et al.2012, Pei, et al., 2010, Pei, et al., 2011).

3.1 Navigation component

The flow chart of navigation component is shown in Fig. 3. The navigation component fulfils the functionalities like querying, map matching and route planning.

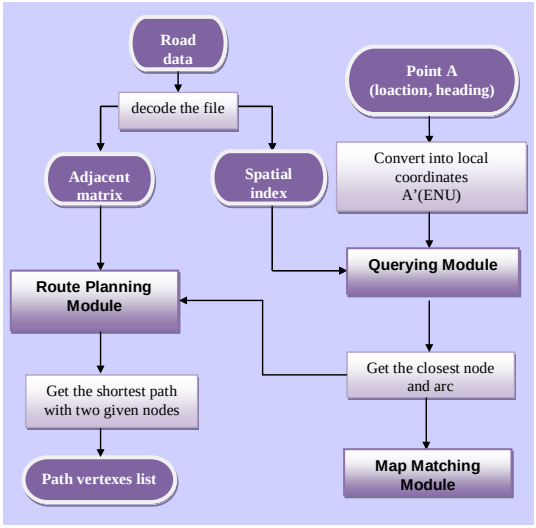


Figure 3: Flow charts of the navigation component.

The navigation component decodes the road data into an adjacent matrix and a spatial index. The adjacent matrix stores the relationship between every two nodes, which is utilized in the route planning module for calculating a shortest path; and the spatial index (Pei, et al., 2009b), which is generated based on the Minimum Boundary Rectangle (MBR) property of each arc, is used for querying module.

The querying module searches the closest road and the closest node for a given position point. The route planning module calculates the shortest path between two existed nodes, and the map matching module locates the given position point in the road data, and fetches the correct picture.

3.2 Querying module

In the querying module, as illustrated in Fig. 3, the related position of point A is transferred into the navigation module with its coordinates, heading and velocity. The coordinates of the given point A are converted into point A' in local East-North-Up (ENU) coordinates as cyan dot in Fig.4 presents, then a buffer square (yellow square in Fig. 4) centred on the given point A' is set with a given buffer value as shown in Fig. 4.

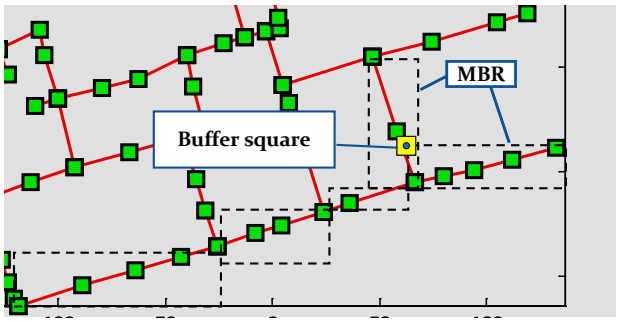


Figure 4: The procedure of querying

Every MBR in the spatial index as a requested geometry is tested against the query geometry, this buffer square, to determine the spatial relationships between two geometries. The Dimensionally Extended 9-Intersection Model (DE-9IM) (Clementini, et al., 1993) is used to present the pair-wise spatial relationship. There are seven possible relationships in this case: contains, overlaps, touches, within, equal, disjoint, and covers. The querying list is re-filtered to exclude the arc if a 'disjoint' relationship is predicated. Among the filtered querying list, the querying module calculates the closest arc and node apart from the given point A'.

3.3 Map matching module

The map matching module is important for the personal navigation in the smartphone, because the positioning accuracy of the on-boarded low-cost GPS is poor. It is not surprising that the smartphone even locates user on a place tens of meters away from its true position in some GPS degraded environments. However, the requirements of positioning accuracy for smartphone based personal navigation are stricter than car navigation. The positioning error requires to be mitigated under several meters for most LBS applications.

A simple map matching algorithm is applied to minimize the calculation demanding and mitigate the position error efficiently simultaneously. As the position obtained from a built-in GPS which might deviate from the true position for several meters, to achieve better accuracy, we correct the position point P(x, y) by projecting the it onto the point P'(x', y') which locates on the centre of the road as Fig. 5 displays. The correcting steps are as following: A) calculates the angle of the arc AB to the true north as α ; B) calculates the angle θ which is the deviated angle from arc AP to the arc AB by cosine formula; C) multiplies the length of AP with $\cos\theta$ to get the length of AP'; D) projects point P(x, y) to the arc AB by formula (1)

$$\begin{cases} x' = x_1 + AP' \times \sin\alpha \\ y' = y_1 + AP' \times \cos\alpha \end{cases} \quad (1)$$

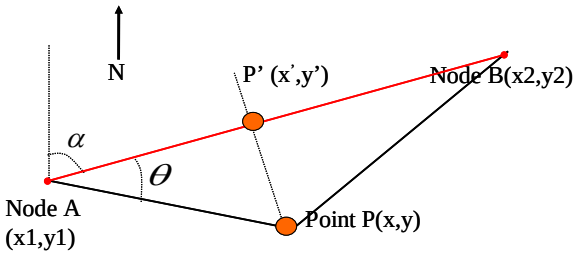


Figure 5: The diagram of projecting the deviated point P to the closest arc AB.

Meanwhile, the heading output from the smartphone is not stable and not entirely consistent with the direction of the road which is calculated from the road data. If the angle between the heading and the road direction is less than 90 degrees, the algorithm assigns the heading with the direction of the road. Otherwise, the opposite direction is assigned. More details can be found in (Chen, et al., 2012c).

3.4 Route planning module

After the quarrying module successfully finds the closest arcs and nodes of the start node and the destination node respectively, the shortest path is calculated using Dijkstra's algorithm (Cormen, et al., 2001). The algorithm uses an adjacent matrix of the graph V which represents the connectivity of the road data and contains the distance between two arbitrary nodes if they are connected. A traversal algorithm based on the adjacency matrix is applied for the route searching.

The algorithm proceeds as the pseudo-codes shown in Table 1, and returns the queue Q as the shortest path from the destination to the start node.

Table 1: Pseudocodes of route planning module

```
/* Initialization: set every distance to INFINITY except the
start node; */
for v_i ∈ V  // V is the adjacent matrix of the graph
    dist [i] = INFINITY;
    previous[i] = NULL;
end for;
dist [start] = 0 ;
while Q is not empty:  // Q is the set of all nodes in Graph
        u = vertex in Q with smallest distance in dist [];
        if dist [u] = INFINITY
            break;  // which means no vertices can access to
start node
        end if;
        remove u from Q;
        for each neighbor v of u // v∈Q
            if(dist[u] + length(u, v) < dist[v])
                dist[v] = dist[u] + length(u, v);
                previous[v]=u;
                update Q;  // Reorder v in the Queue
            end if;
        end for;
end while;
```

3.5 Geocoding images

Images utilized in the solution are obtained from the 3D model constructed on a computer with 3D Max software. Considering the speed of a pedestrian is normally 3-7 km/h, the interval of images generated is set as one meter to be more comfortable for human's eyes. The resolution of the image is scaled to fit the window of the smartphone as 240x320 pixels, and the average size of the image is 6 KB.

The images are associated with geographic information by the nomination based on the following premises: A) each arc is numbered by a unique natural number; B) every arc has two directions. The images are geocoded according to the following criteria: A) the arc belonged; B) the start node of the arc, C) the distance apart from the start node. After that, a unique geocoded filename is assigned to each image according to the IDs of the belonged arc and start node as ArcID_StartNodeID_ImageID. In this way, any real-time position of a pedestrian on a path can be located in the relevant geocoded image. (Wang et al., 2012)

4. Field Test

To evaluate the positioning accuracy and the performance of the proposed methods and algorithms, we carried out an outdoor field test on the 22nd Jul, 2011 at Tapiola centre in Espoo, Finland.

A tester carried a backpack platform which was firmly installed with a NovAltel Synchronized Position Attitude Navigation (SPAN) system, and took two experimental Nokia 6710 mobiles in hands. The mobiles were installed the model-based application and the image-based application respectively. An assistant took a laptop to download and archive the data from the SPAN for post-processing. NovAtel's SPAN technology tightly couples precision GPS receiver with robust Inertial Measurement Units (IMUs) to provide reliable, continuously available measurements including position, velocity and attitude even though short periods of time when no GPS satellites are available. Data from the SPAN were post-processed to generate the reference trajectory with centimetre level accuracy. Such accuracy can meet the experiment's requirements for evaluating the performance and the improvement of the proposed map matching algorithm.

The installation of SPAN in the field test is shown in Fig. 6. It composed with a GPS antenna, an IMU, a GPS receiver and a battery. The trajectory of GPS measurements of SPAN for Tapiola test is illustrated in Fig. 7 with reference of Google Earth. The route of the test started from A (Handels Bank) to B (Sokos Shopping Centre), and returned from B to A; then started from A again, and turned left at C (Stockmann) and walked to D (Pohjola Insurance Co.,), then returned to C, then went to B, and finally went back to E (Nordea Bank). The GPS gap between location C and location D in Fig. 7 is because in that period, the high buildings in that area blocked the GPS signal. The similar situation also happened in the beginning period of the test when the GPS visibility was poor because of the blockage from southern buildings. However, the IMU provide continuous and precise measurement during GPS outrage.

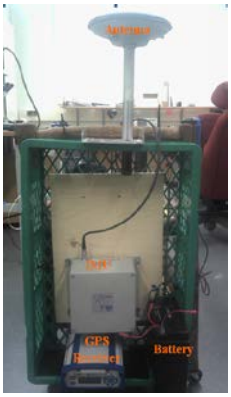


Figure 6: The installation of SPAN.

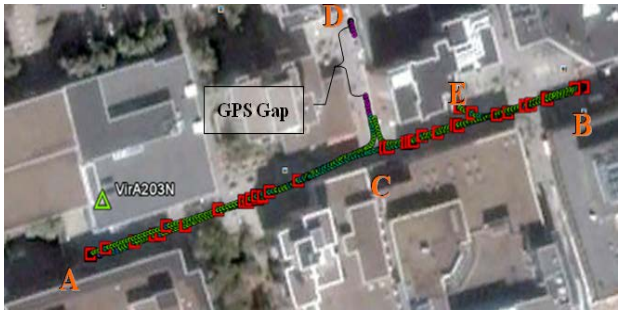


Figure 7: Trajectory of GPS measurements of the SPAN.

5. Results and Evaluations

The evaluation is carried out by comparing in the performance between the model-based solution and the image-based solution in terms of: A) positioning accuracy, B) resource consumption, C) efficiency, D) visualization E) labour cost. Some basic test results have been presented in (Wang et al., 2012, Chen, et al., 2012c); more detailed data could be found in this section.

5.1 Positioning accuracy evaluation

Fig. 8 presents the trajectories from three data sources: built-in GPS data archived by smartphone in blue, map-matched dataset based on the original GPS in green, and the reference trajectory from the SPAN in red in local coordinate. The road data are also presented in black, which are the base for querying and route planning.

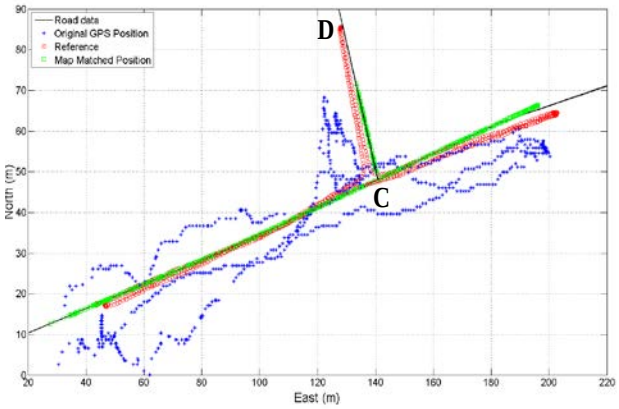


Figure 8: The trajectories of raw GPS data, map matched results and reference achieved by the SPAN

The original GPS data are randomly scattered around the reference trajectory. On the road CD, as the reason mentioned above, it is a GPS unfriendly area where the GPS signal is blocked by intensive high-rise buildings nearby. As a result, the trajectory of original GPS data doesn't reach the end of the road CD. The map matching algorithm matches the GPS data to the centre of the road according to the original position and heading. The map matched trajectory is basically consistent with the reference trajectory, except the part of the GPS-blocked area as Fig. 7 presents.

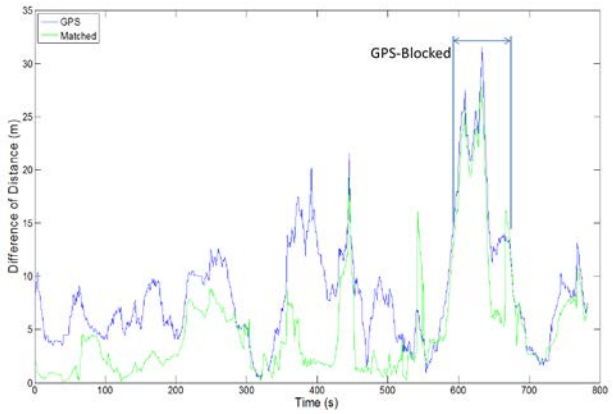


Figure 9: Positioning improvement of map matching algorithm

Table 2: The distance difference of raw GPS and map matched data

	Mean (m)	Max (m)	Min (m)	std	Improvement
GPS	8.41	31.54	0.23	5.33	40%
Matched	5.00	27.79	0.08	5.27	
GPS(50:550)	7.67	21.53	0.23	3.74	54%
Matched(50:550)	3.52	19.45	0.08	3.03	

We can also compare the horizontal error distance of original GPS data and map matched data against the reference trajectory in Fig. 9 and results are listed in Table 2. The maximum horizontal error distance of the original GPS data is 31 meters, with 8.41 meters as mean value. After map matched, the error decreases to 5.00 meters. Therefore, the position accuracy is improved by 42.6%. A considerable error appears from the 550th second to the 650th second, when the tester walks on the GPS unfriendly location - road CD. To check the position error in GPS friendly environment, we select the dataset from the 50th second to the 550th second to analyse. The map matched positioning error in the selected period is 3.24 meters, against 7.67 meters of original GPS data. The position accuracy of smartphone GPS is improved by 54% in GPS friendly environments by proposed map matched algorithm.

We evaluate the heading improvement by comparing the GPS-derived heading and map matched results with reference heading as well. The compared result is displayed in Fig. 10.

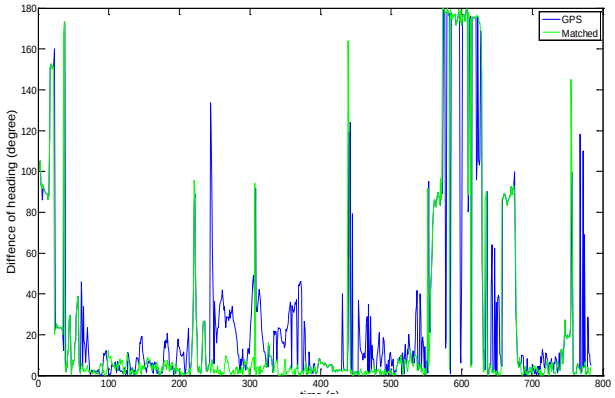


Figure 10: Heading improvement of map matching algorithm

The average deviation of the original GPS-derived heading is 30 degrees, which is 5 degrees higher than that of map matched results. In the GPS friendly environment from the 50th second to the 550th second, the map matched mean deviation decreases to 5 degrees. Such heading accuracy can meet the requirements of most positioning applications in the situations.

Table 3: The deviation of raw GPS heading and map-matched heading

	Mean (deg)	Max (deg)	Min (deg)	std	Improvement
GPS	30	179	0	46	17%
Matched	25	179	0	49	
GPS(50:550)	13	134	0	17	62%
Matched(50:550)	5	164	0	12	

A conclusion can be drawn based on the test results discussed above that the map matching algorithm plays a significant role to improve the position and heading

accuracy of the GPS data obtained from smartphone built-in sensor in the GPS friendly environment.

However, in the GPS-degraded environments like the complex urban city, more solutions are demanded to improve the positioning accuracy, such as mitigation of non-line-of-sight positioning errors (Chen, et al., 2012a, Chen, et al., 2009a), motion recognition based on measurements of self-contained sensor (Chen, et al., 2010b, Pei et al., 2012) and multi-sensor fusion method (Chen et al., 2012b) etc.

5.2 Consumption evaluation

To quantify the improvements on resource consumption of the proposed solution, the value of improvements is calculated using the values between two solutions for each measurement with the equation (2):

$$p = \frac{(v_m - v_0) - (v_i - v_0)}{v_m} \times 100 \qquad (2)$$

where p is the improvement in percentages, v_m is the value of consumed resource of the model-based solution; v_i is that of the image-based solution, and v_0 the value when the smartphone is running at standby state. Tapiola test field is selected for the evaluation.

The consumption evaluation is executed with Nokia Energy Profiler 1.2, which is a stand-alone measurement application for S60 3rd Edition, and offers developers to monitor the energy usage on their target applications in real time. The Nokia Energy Profiler supports multiple measurement views including power consumption, Random Access Memory (RAM), processor, current etc.

The evaluation data consist of four separated scenarios: A) initialization (PI); B) virtual navigation (PV); C) real time navigation (PRN) and D) real time positioning (PRP). The virtual navigation scenario runs the application offline in a similar way as playing a game. No positioning sensor is activated in this scenario because all position information is input by user through pressing the direction keys on the smartphone keyboard. The real time positioning scenario turns on the GPS chip but does not offer navigation service, thus the power consumption should be lower in comparison with real time navigation scenario which activates full navigation service.

In order to evaluate the improvement of the image-based application objectively, we also take the data measured at the standby state into consideration as a baseline for comparison.

The comparison of power consumption between image-based application (ABI) and model-based application (ABM) is shown in Fig. 11 and Table 3. The red curve displays the fluctuation of power consumption of ABM, while the blue one represents that of ABI. From the Fig. 11, we can conclude the image-based solution has significant advantages on consumptions than the model-based solution.

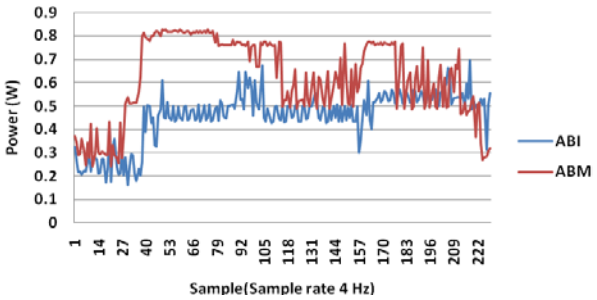


Figure 11: The comparison of power consumption

Average power consumption of ABI is 0.52W and ABM's power consumption is 0.68W. The power consumption at standby state is 0.31w. The imaged-based solution decreased the power consumption by 43 precents in all. Especially in of initialization period, the image-based solution consumes 36% of the power needed for model-based solution. Considering a larger 3D model will be loaded, the energy efficiency for image-based solution is even higher.

The lower consumption of the proposed solution benefits smartphones to be used for a longer time. The tested smartphone will last 5 and half hours when the ABM is running, which is 1.5 more hours comparing with ABI. In the terms of energy, the image-based solution has obvious advantages.

Table 4: Comparison of power consumption

Power Consumption (W)		Mean	Net Increase	Improvement
PI	ABI	0.48	0.17	64%
	ABM	0.80	0.48	
PV	ABI	0.50	0.18	43%
	ABM	0.63	0.32	
PRP	ABI	0.52	0.21	37%
	ABM	0.65	0.34	
PRN	ABI	0.59	0.28	20%
	ABM	0.66	0.34	
Average	ABI	0.52	0.21	43%
	ABM	0.68	0.37	

Remarks: Power consumption in standby status is 0.31W.

The CPU load is presented in the Fig. 12. We introduce two items in the comparisons of CPU load: one is FCR (Full CPU load occupation Rate) which calculates the ratio that 100% CPU load during the test; and another is the ratio that 90% of CPU payload is occupied by the program during the test, abbreviated as FCR90.

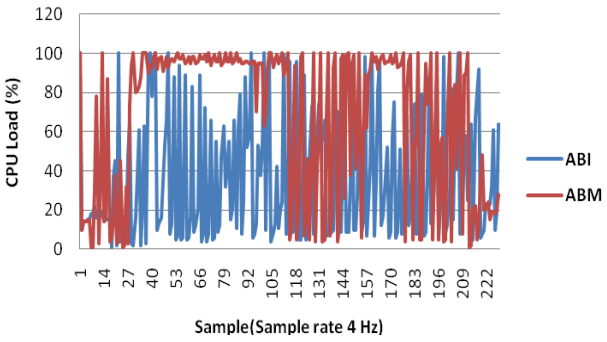


Figure 12: The comparison of CPU load

The average CPU load of ABM in PI is 96%, and FCR90 is 92%, as shown in Table 4. In the procedure of initializing ABM, the application occupies almost full computational power of the CPU, and there is hardly any free time slot available for other applications. That will put more pressure on the system and easily lead to a system crashing. In both real-time procedures (navigation and positioning), the ABM demands higher CPU calculating power to load and render the 3D model in real time, while the FCR and FCR 90 of the ABI are zero. In general, the average CPU load for ABI is 32%, achieves 81% improvement when comparing with that of ABM. Both the data and figures can prove overwhelming advantages on the CPU load for the ABI solution.

Table 5: Comparison of memory consumption

CPU Load (%)		Mean	Net Increase	Improvement (%)	FCR (%)	FCR90 (%)
PI	ABI	38.8	14.5	80	6.9	21.0
	ABM	96.1	71.8		12.5	92.0
PV	ABI	36.4	12.1	71	0.9	8.5
	ABM	66.8	42.5		17.0	45.0
PRP	ABI	26.2	1.9	93	0	0
	ABM	51.2	26.9		18.4	26.0
PRN	ABI	28.5	4.2	87	0	0
	ABM	56.0	31.7		21.0	26.0
Average	ABI	32.5	8.2	81	2.0	7.4
	ABM	67.5	43.2		18.6	47.0

Remarks: The CPU load in the standby state is 24.3%.

The system RAM of smartphone is used by active programs and the system itself, as well as providing "disk" space. The net increase of ABI's occupied RAM is 3.6MB averagely, and is 16.6MB for ABM, improving by 78%. From Fig. 13, we can observe the occupied RAM of ABI increases little comparing with the standby state. Considering the size of the current 3D models is relatively small, simplified with compressed texture, the occupied RAM will have a more significant increase if large, more complex models with higher resolution are applied. On the contrary, the occupied RAM for ABI always keeps fluctuation in a narrow scale regardless of the model size. In this aspect, the image-based solution gets less restriction from hardware viewpoint, and can be applied for more low-end smartphone since most smartphones have external memory cards nowadays.

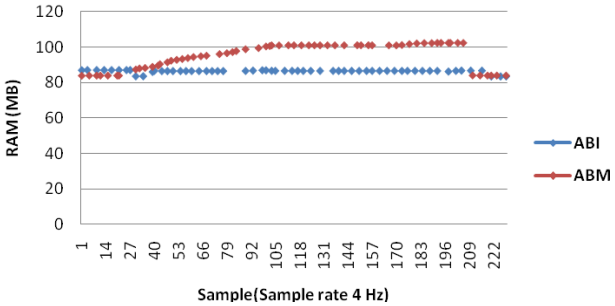


Figure 13: The comparison of RAM consumptions

5.3 Efficiency evaluation

The efficiency of the solutions is evaluated according to the parameter of consuming time, which is the cost time for completing specific functions. The consuming time in program running can be computed in a specific function and the results are archived in a debug file stored on the smartphone memory. The difference between two solutions is observable to be identified even by human's eyes.

Time consuming for two solutions is compared in the Table 6. The full operating cycle includes the procedures of obtaining GPS data, post-processing, querying, routing, map matching, and displaying, which is executed in the program every second. ABI can finish all of these procedures in 0.09 second, while it cost 0.67 second for ABM.

Table 6: Comparison of time consumption

	ABI (seconds)	ABM (seconds)
Initialization	0.45	1.94
Initial rendering	1	15
Full operating cycle	0.09	0.67

The consuming time for model-based solution increases with the large and complex model. Taking Shanghai EXPO 2010 model as an example, it takes 30 seconds to finish the initial rendering after the application starts. On the contrary, the consuming time for the image-based solution is independent of the scale of model. No matter which size the 3D model is, or how large the area to be demonstrated is, the image-based solution keeps the same efficiency. The application with shorter consuming time for initialization offers better user experience.

5.4 Visualization evaluation

The displaying performance is seriously restricted by the CPU load and the rendering speed in ABM solution. Limited by the computational power, the texture rendering or even the display of the 3D model is always temporary missing as Fig. 14 (a) and Fig. 14 (c) presents.

Such situations will never happen in the ABI solution, as Fig. 14 (b) and Fig. 14 (d) show.

Meanwhile, the images applied in the ABI solution are captured from the model constructed on the computer. They keep explicit model and uncompressed texture as those on computer in the visualization. Furthermore, comparing with the simplified background in the ABM solution, the adding texture of sky and finer ground casts a more realistic immersion for the mobile user.

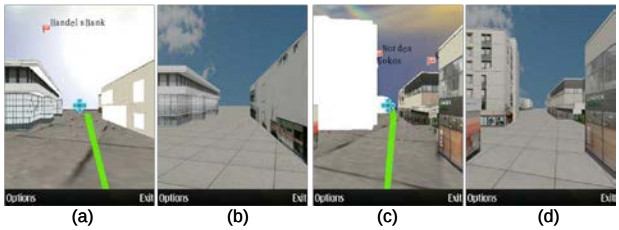


Figure 14: Comparison of the same street views

The frame rate, also called FPS, is the frequency at which the smartphone displays consecutive images. In the ABM solution, 1 FPS already reaches the limit of display engine. Even so, the rendering speed still cannot catch the speed of frame updating. Some textures fail to be rendered in time during the navigation.

On the contrary, there is an obvious advantage for the proposed ABI solution. Taking Table 6 as reference, one single full operating cycle for the ABI costs no more than 0.1 second, thus there is still some potential for increasing FPS of the ABI solution in a certain range, under the premise of the proper operation. Higher refreshing rate results in a smoother visualization which makes humans eyes more comfortable. The frame frequency can be increased from 1 to 2.5 frames per second according to the test.

In one word, the proposed ABI solution can increase frame rate to achieve a smoother visualization, on the premise of keeping even lower consumptions against model-based solution.

5.5 Labour and cost Evaluation

Most of the existing 3D mobile navigation systems utilize 3D model, which requires expensive equipment such as laser scanner for data acquisition which costs more than hundreds thousands of euros, involves complicated model construction and even plentiful manual work.

In this work, the data used in this 3D model construction are collected by a mobile road environment mapping system which is composed with several latest technologies. The instruments are expensive and far from easy access by the public. Meanwhile, the data processing of data is also a redundant procedure. For example, the 3D model of the Tapiola test field is based on laser point cloud collected by a mobile mapping system. And it took a proficient researcher about three months to construct 3D model, and spent a master student nearly one year to import the model into the mobile platform.

Furthermore, to construct a 3D model for indoor environment with a mobile mapping system will be more difficult from the aspect of technology. And it is not an economic solution for small-scale models considering the corresponding cost on the labour and the time.

The proposed geocoded image based solution provides a novel method to 3D visualization in a more efficient and economical way, by implementing visualization with real photos replacing the images taken from 3D model. Such replacement only needs a camera cost several hundred euros and a simple geo-reference system. And the data collection task in the field can be accomplished in a couple days by one person. It is also flexible to set up a small-scale model within a few days. That means it can be easily transplanted into a commercial system for museum, exhibition, and other indoor applications. Comparing with traditional 3D modelling, the cost of the proposed method is largely saved in terms of labour, time and instrument investment.

6. Conclusions and Discussions

This paper introduces a simple, flexible, energy-efficient and cost-effective 3D personal navigation solution, which utilizes geocoded images captured on 3D models to implement 3D personal navigation in a smartphone for small-scale area.

Comparing with the 3D model-based solution, the proposed geocoded image-based solution presents a better performance with higher resolution and refreshing rate and the complex 3D modelling process is not required anymore. As the consuming time for the calculation is reduced, the consumption of power, CPU and RAM decrease dramatically, as well as labour and cost. Because of low consumption of the system's energy and resource, the solution also offers a possibility to be applied to low-end smartphone platform.

For a user-friendly personal navigation application, turning details is preferred as extra information. Such functionality will be added in further study. Other route planning algorithms besides the classic Dijkstra's algorithm will be investigated to higher performance for large road database. Cloud-based image storage strategy will be probed for large-scale area application. The test area will be extended, and indoor scenario will also be considered. The geocoded images applied in further studies will be collected by a camera instead of

snatching from 3D models. Panoramic photo will also be applied to fulfil 360 degrees viewpoint. To improve the positioning accuracy in the GPS-denied environment, a positioning solution based on vision will be considered as auxiliary algorithm to improve the accuracy and reliability of the system (Li, et al., 2010).

Acknowledgments

This work is a part of the INdoor Outdoor Seamless Navigation for SEnsing Human Behavior (INOSENSE) project, funded by the Academy of Finland.

References

Burigat S., Chittaro L. (2005), ***Location-aware Visualization of VRML Models in GPS-based Mobile Guides,*** Proceedings of Web3D 2005: 10th International Conference on 3D Web Technology, ACM Press, New York, 2005, pp. 57-64.

Chehimi F., Coulton P., Edwards R. (2006), ***Advances in 3D graphics for Smartphones.*** In: Proceedings of international conference on information and communication technologies: from theory to applications. Damascus, Syria, 24–28, April 2006.

Chen Liang, Ali-Loytty Simo, Piche Robert and Wu Lenan. (2009a), ***Mobile tracking in mixed line-of-sight/non-line-of-sight conditions: algorithm and theoretical lower bound,*** Wireless Personal Communications, vol. 65, no. 4, pp. 753-771, 2012

Chen Liang and Wu Lenan. (2009b), ***Mobile positioning in mixed LOS/NLOS conditions using modified EKF banks and data fusion method***, IEICE Transactions on Communications, vol. E92-B, no. 4, pp. 1318-1325, 2009.

Chen Liang, Pei Ling, Kuusniemi Heidi, Chen Yuwei, Kröger Tuomo and Chen Ruizhi. (2012a), ***Bayesian Fusion for Indoor Positioning Using Bluetooth Fingerprints.*** Wireless Personal Communications, 2012, DOI: 10.1007/s11277-012-0777-1.

Chen Ruizhi, Kuusniemi Heidi, Hyyppä Juha, Zhang Jixian, Takala Jarmo, Kuittinen Risto, Chen Yuwei, Pei Ling, Liu Zhengjun, Zhu Lingli, Liu Jingbin, Qin Yan, Leppäkoski Helena, and Wang Jianyu. (2010a), ***Going 3D, Personal Nav and LBS,*** GPS World, Feb 2010 issue, pp14-17.

Chen Ruizhi, Hyyppä Juha, Zhang Jixian, Takala Jarmo, Kuittinen Risto, Chen Yuwei, Pei Ling, Liu Zhenjun, Zhu Lingli, Kuusniemi Heidi, Liu Jingbin, Qin Yan, Leppäkoski Helena, Wang Jianyu. (2009b), ***Development of a 3D Personal Navigation and LBS System with Demonstration in Shanghai EXPO in 2010,*** in Proceedings of ION GNSS 2009, Savannah, GA, USA, Sept. 22-25, 2009.

Chen Ruizhi, Wang Yiwu, Pei Ling, Chen Yuwei, Virrantaus Kirsi. (2012c), ***3D Personal Navigation in Smartphone Using Geocoded Images.*** GPS World. October 2012 issue, pp36-42.

Chen Yuwei; Antero Kukko; Chen Ruizhi; Chen Wei; Hyyppä Juha; Kaartinen Harri. (2008), ***Multi-sensor based three-dimensions navigation and its application on pedestrian navigation,*** Proceedings of NaviTec 2008, Section Sensor Fusion 2, 10-12 December 2008, Noordwijk, Netherlands.

Chen Yuwei, Chen Ruizhi, Pei Ling, Kröger Tuomo, Chen Wei, Kuusniemi Heidi and Liu Jinbin. (2010b), ***Knowledge-based Error Detection and Correction Method of a Multi-sensor Multi-network Positioning Platform for Pedestrian Indoor Navigation,*** Proceedings of IEEE/ION PLANS 2010, pp873-879, Indian Wells, USA, May 4 - 6, 2010.

Chun B.-G. and Maniatis P. (2009), ***Augmented Smartphone Applications Through Clone Cloud Execution.*** In Proc. Of the 8th Workshop on Hot Topics in Operating Systems (HotOS), Monte Verita, Switzerland, May 2009.

Clementini Eliseo, Paolino Di Felice and Peter van Oosterom. (1993), ***A small set of formal topological relationships suitable for end-user interaction.*** In Abel, David; Ooi, Beng Chin. Advances in Spatial Databases: Third International Symposium, SSD '93 Singapore, June 23–25, 1993 Proceedings. Lecture Notes in Computer Science. 692/1993. Springer. pp. 277–295.

Coors, V. and Zipf, A. (2007), ***MoNa 3D --- Mobile Navigation using 3D City Models***. LBS and Telecartography 2007. Hongkong.

Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L.; Stein, Clifford. (2001), ***Section 24.3: Dijkstra's algorithm. Introduction to Algorithms (Second ed.).*** MIT Press and McGraw-Hill. pp. 595–601. ISBN 0-262-03293-7.

Goldberg. Daniel W. (2008), ***A Geocoding Best Practices Guide***. University of Southern California, GIS Research Laboratory, 2008.

Kukko, A., C.-O Andrei, V.-M. Salminen, H.Kaartinen, Y. Chen, P. Rönnholm, H. Hyyppä, J.Hyyppä, R. Chen, H. Haggrén, I. Kosonen, and K.Capek. (2007), ***Road Environment Mapping Systemof the Finnish Geodetic Institute – FGI Roamer,*** International

Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences,Vol. 36(3/W52), pp. 241-247.

Li Xun, Jinling Wang; Olesk, A.; Knight, N. ; Weidong Ding (2010), ***Vision-based Positioning with a Single Camera and 3D Maps: Accuracy and Reliability Analysis.*** Proceeding of UPINLBS 2010, Helsinki, Finland, Oct. 2010.

Liu Zhengjun, Zhang Jixian, Yan Qin, Chen Ruizhi, Kuittinen Risto, Ye Guangen, Pei Ling, Zhu Lingli, Liu Jingbin. (2010), ***Implementation of a 3D Personal Navigation Visualization Engine and its Demonstration in Shanghai World Exposition 2010,*** 7th International Symposium on LBS & TeleCartography, Guangzhou, China, 20-22, Sept. 2010.

Nurminen Antti. (2006), ***m-LOMA - a mobile 3D city map.*** In: Denis ,Gracanin (editor). Proceedings of the 11th International Conference on 3D Web Technology (Web3D 2006). Columbia, Maryland, USA. 18-21 April 2006, pp 7-18.

Rakkolainen I., Vainio T.. (2001), ***A 3D City Info for Mobile Users,*** Computers & Graphics. Volume 25, Issue 4, pp 619-625, August, 2001.

Pei Ling, Guinness Robert, Liu Jingbin Chen Yuwei, Kuusniemi Heidi, Chen Ruizhi. (2012), ***Using LS-SVM Based Motion Recognition for Smartphone Indoor Wireless Positioning***. Sensors, Vol 12(5), 6155-6175.

Pei Ling, Chen Ruizhi, Liu Jingbin, Liu Zhengjun, Kuusniemi Heidi, Chen Yuwei, Zhu Lingli.(2011), ***Sensor Assisted 3D Personal navigation on a Smart Phone in GPS Degraded Environments,*** Proceedings of the 19th International Conference on Geoinformatics, Shanghai, China, June 24-26, 2011.

Pei Ling, Chen Ruizhi, Liu Jingbin, Tomi Tenhunen, Heidi Kuusniemi, Chen Yuwei. (2010), ***Inquiry-Based Bluetooth Indoor Positioning via RSSI Probability Distributions***, In Second International Conference on Advances in Satellite and Space Communications, (SPACOMM 2010), Athens, Greece, 13-19 June, 2010, pp.151-156.

Pei Ling, Chen Ruizhi, Chen Yuewei, Leppakoski Helena, and Perttula Arto. (2009a), ***Indoor/outdoor seamless positioning technologies integrated on smart phone.*** In First International Conference on Advances in Satellite and Space Communications,(SPACOMM 2009), Colmar, France, 20-25 July, 2009, pp. 141-145.

Pei Ling, Wang Qing, Gu Juan. (2009b), ***MLGL: A Spatial Data Model for Mobile Geoinformatics and Cartography,*** Proceedings of the 5th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM 2009), Beijing, China, September 24-26, 2009.

Pulli. K. (2006), ***New APIs for Mobile Graphics,*** Proc. SPIE Electronic Imaging: Multimedia on Mobile Devices II, SPIE, 2006, pp. 1–13.

Wagner, D., Schmalstieg, D. (2007), ***ARToolKitPlus for Pose Tracking on Mobile Devices,*** Proceedings of 12th Computer Vision Winter Workshop (CVWW'07), pp. 139-146, 2007.

Wang Yiwu, Chen Ruizhi, Pei Ling, Chen Yuwei, Virrantaus Kirsi. (2012), ***3D Personal Navigation in Smartphone using Geocoded Images,*** Proceeding of IEEE/ION PLANS2012, pp:584-589, Myrtle Beach, USA, 23-26 April 2012.

Biography

Yiwu Wang, (born in 1978, sindywyw@hotmail.com) graduated from Department of Oceanic Engineering, Zhejiang University in China in 2001, with bachelor degree of hydrology and water engineering. She continued her master of Geoinformatics in the Department of Real Estate, Planning and Geoinformatics, school of Engineering, Aalto University in Finland. In 2011, she completed her graduate thesis in Finnish Geodetic Institute with the title “3D Personal Navigation in Smartphone using Geocoded Images”.