

DYNAMIC WIFI-BASED INDOOR POSITIONING IN 3D VIRTUAL WORLD

S. Chan^{a,*}, G. Sohn^a, L. Wang^a, W. Lee^b

^a GeoICT Laboratory, York University, 4700 Keele Street, Toronto, M3J 1P3, Canada
(sol122, gsohn, wangly)@yorku.ca

^b KICT, SOC Research Institute, 283 Goyangdae-Ro, Goyang-Si, Gyeonggi-Do 411-712, Republic of Korea
wslee@kict.re.kr

KEY WORDS: 3D Modeling, Indoor Positioning, Tracking, System Design, Visualization, GIS

ABSTRACT:

A web-based system based on the 3DTown project was proposed using Google Earth plug-in that brings information from indoor positioning devices and real-time sensors into an integrated 3D indoor and outdoor virtual world to visualize the dynamics of urban life within the 3D context of a city. We addressed limitation of the 3DTown project with particular emphasis on video surveillance camera used for indoor tracking purposes. The proposed solution was to utilize wireless local area network (WLAN) WiFi as a replacement technology for localizing objects of interest due to the wide spread availability and large coverage area of WiFi in indoor building spaces. Indoor positioning was performed using WiFi without modifying existing building infrastructure or introducing additional access points (AP)s. A hybrid probabilistic approach was used for indoor positioning based on previously recorded WiFi fingerprint database in the Petrie Science and Engineering building at York University. In addition, we have developed a 3D building modeling module that allows for efficient reconstruction of outdoor building models to be integrated with indoor building models; a sensor module for receiving, distributing, and visualizing real-time sensor data; and a web-based visualization module for users to explore the dynamic urban life in a virtual world. In order to solve the problems in the implementation of the proposed system, we introduce approaches for integration of indoor building models with indoor positioning data, as well as real-time sensor information and visualization on the web-based system. In this paper we report the preliminary results of our prototype system, demonstrating the system's capability for implementing a dynamic 3D indoor and outdoor virtual world that is composed of discrete modules connected through pre-determined communication protocols.

1. INTRODUCTION

In recent years, 3D visualization tools such as Google Earth have become essential for street views and visualization of 3D city models. One can browse through cities and search for useful geospatial information in a web-based virtual environment from our desktop or mobile devices. However, these visualization tools provide a stationary virtual world which limits their use in applications where object dynamics is important. Cities are dynamic, living environments with various kinds of real-time information and moving objects (people and vehicles). To bring such dynamic information into these virtual globe platforms will give users a more immersive context to feel and interpret the real dynamic world.

We are motivated to produce a "dynamic virtual 3D world" with emphasis on the indoor environment. This can be done by fusing sensors with 3D indoor building models, where various kinds of environment information and human positions can be integrated with 3D virtual world and connected with real-time, location-based intelligence. In [3], the authors introduce methods for augmenting aerial visualizations of Earth with dynamic information obtained from videos. Different approaches are proposed to analyze videos of pedestrians and cars under differing conditions to extract dynamic information. The information are then modeled and projected to Aerial Earth Map (AEM) to render the related dynamics. The research shows promising results and demonstrates the feasibility of integrating

movements detected from videos and 3d virtual world. However, it just showed the integration of outdoor dynamic movements detected from videos.

In this paper, we present an approach to integrate dynamic human position in an indoor space with real-time sensor information, for example temperature, in an integrated 3D indoor and outdoor virtual world to provide real-time, location-based intelligence. To achieve this goal, several technical problems need to be addressed.

First, a web-based system that can provide visualization of the dynamic virtual 3D world needs to be designed and developed. The system should have these functions: real-time data acquisition on moving objects from WiFi-enabled devices, real-time sensor data exchange, GIS data and real-time data management, and 3D dynamic data visualization.

Second, we need to develop an approach to integrate indoor and outdoor data and information in 3D context. In [10], we proposed an integrated framework to reconstruct full 3D building models, based on which integrated 3D building indoor and facade models can be reconstructed using Terrestrial Laser Scanning (TLS) data and floor plans. Based on these models, further development needs to be done to create a 3D continuous indoor and outdoor GIS platform to support the whole system.

Third, utilizing WiFi technologies for locating a person in an indoor environment is now feasible [1,5,9]. However, measuring

* Corresponding author.

WiFi signals in indoor space can still prove to be a challenge. The inability of most signals to penetrate through objects result in attenuation of the signals. Additionally the multipath propagation of WiFi caused by reflection and refraction off objects can further complicate any attempts to locate a person [4].

The final component to be considered is how to visualize real-time information in a 3D virtual environment. Our primary interest is the visualization of real-time indoor positions of pedestrians as well as data collected from indoor sensors that provide valuable environment data such as room temperature.

The rest of the paper is structured as follows: section 2 gives an overview of the system architecture and main functions; section 3 describes the approaches to integrate and visualize sensor and indoor position data; section 4 describes our proposed method to integrate indoor and outdoor data and model based on 3D terrestrial laser scanning data and floor plans, and the creation of the dynamic seamless 3D GIS database; section 5 explains localization will be performed using WiFi; section 6 shows the test results of the prototype system; section 7 presents some comments and future works.

2. SYSTEM ARCHITECTURE

In order to achieve our goals, we developed a Web-based dynamic virtual 3D GIS system, which features the integration of indoor position information and real-time sensor information (i.e. temperature) with a seamless 3D indoor and outdoor GIS database. The system was built based on our U50 iCampus system, which is built in celebration of the 50th anniversary of York University, and later it was further refined under the title of 3D Town. 3D virtual photorealistic building models and campus-wide GIS database of the university's Keele and Glendon campuses were created, which enables 3D interactive visualization, navigation and fly-about the campuses. The 3D building models and the GIS database were used as a basis for developing the dynamic virtual 3D GIS.

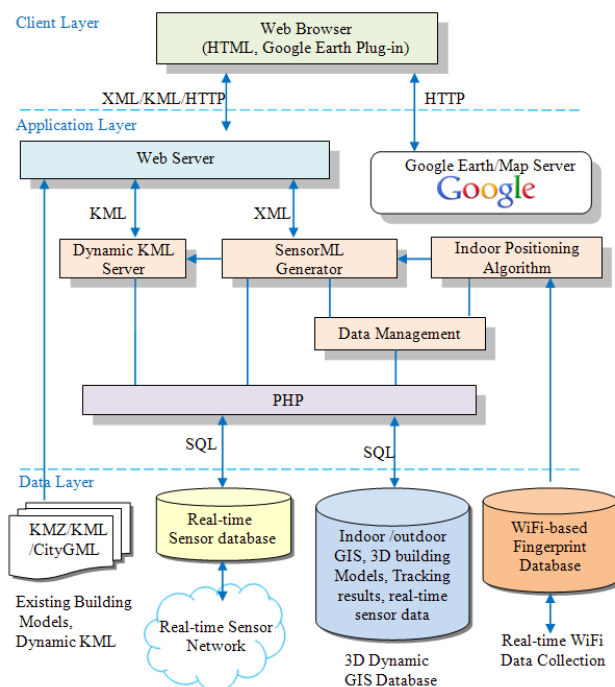


Figure 1. General system architecture for implementing dynamic 3D virtual world.

Figure 1 shows the general system architecture of our proposed system for implementing dynamic virtual 3D world. The system was designed with three functional layers: client layer, application layer, and data layer, which conforms to a multi-tiered layered architecture typical of internet based GIS applications.

We use Google Earth plug-in as the 3D display engine of the system, which allows users to navigate and explore geographic data on a 3D globe using a web browser. Using its JavaScript API, many system display functions like data loading, view and animation control can be implemented.

In application layer, in addition to Web Server and PHP, which are common parts of most internet applications, there are four main parts: SensorML generator, dynamic KML server, indoor positioning algorithm, and data management module.

2.1 SensorML Generator

The OpenGIS Sensor Model Language Encoding Standard (SensorML) specifies models and XML encoding that provide a framework within which the geometric, dynamic, and observational characteristics of sensors and sensor systems can be defined [7]. However, there are many different sensor types and most of existing sensor systems does not support SensorML. Therefore, in order to provide a unified data exchange interface between the system and different sensor systems, we developed a SensorML generator to encode all the dynamic information obtained from indoor position information obtained from real-time WiFi data collection, and real-time sensors. We also made small modifications to the SensorML format so as to make it more suitable for our study.

2.2 Dynamic KML Server

In order to visualize dynamic or real-time data in Google Earth API, the data has to be described in KML format. Using PHP, we developed a dynamic KML server which can read in the XML sensor data encoded by SensorML generator and retrieve data from GIS database to deliver dynamic KML data to the client in a real-time request-response manner.

2.3 Indoor Positioning Algorithm

We developed an indoor positioning algorithm based on a hybrid deterministic and probabilistic approach. The first method is finding k-Nearest Neighbours (k-NN) for generalizing the possible estimated locations. Then a probabilistic algorithm based on Maximum a Posteriori (MAP) estimation is applied to probabilistically select the final location of the person. Using signal propagation model such as Log-distance Path Loss (LDPL) model, distances from the two nearest APs will be used to determine the location. The indoor position results are exported to SensorML generator to be converted into the unified XML format; and meanwhile, the results are also stored in the database by the data management module.

2.4 Data Management Module

The main function of the data management module is to store the dynamic movement tracking results and the real-time sensor information into the 3D dynamic GIS database, and update that information in a near real-time manner.

In data layer, the main input datasets to the system are dynamic WiFi data collected for indoor positioning and real-time indoor sensor data. In our current system, the real-time sensor is mainly temperature sensor. It certainly can be extended to integrate other kinds of sensors. The reason that we consider indoor positioning data and real-time sensor separately is because the data processing approaches for them are different. The system is

designed to integrate dynamic information from real-time WiFi data to position moving objects, but in our prototype system, the demonstration was done using pre-recorded WiFi data for determining the indoor position.

The core of the data layer is the seamless dynamic 3D indoor and outdoor GIS database. It not only store indoor and outdoor GIS data and 3D building models, it also stores the real-time data (i.e. real-time room temperatures) and dynamic information (i.e. trajectories of moving objects). The creation of indoor GIS and its integration with existing GIS and 3D building models are explained in section 3. Other than the existing 3D building models and existing data in KML or CityGML (which will require conversion into KML) formats can also be integrated in the system.

3. INTEGRATE AND VISUALIZE SENSOR DATA

The sensors used in this study are the Xbow MTS310 sensors. Each sensor node consists of a microphone circuit for acoustic measurements, a light sensor with maximum sensitivity at a wavelength of 690nm, a magnetometer, an accelerometer, and a temperature sensor. However, only the temperature measurements will be used for this study as a mean to visualize the indoor room temperatures of the Petrie Science and Engineering (PSE) Building. Because the temperature of a room normally doesn't change very frequently, the update interval is set to 3minutes.

In addition to the Xbow sensors for monitoring the indoor temperature, the York University Weather Station has also been connected with the dynamic sensor database allowing near real-time update on the campus temperature. The weather station is programmed to update once every 5 minutes. The weather station collects a wide range of meteorological data such as wind velocity, air temperature, relative humidity, precipitation, air pressure, etc., from which only the temperature was used in the prototype system.

KML Network link is an effective way to make dynamic KML, which can automatically update the linked KML data at a given interval. When used together with a dynamic KML server which can retrieve data from any database or real-time data sources, real-time data extraction and visualization can be achieved. The main procedures are:

1. The user loads a KML file which includes a network link to a PHP script, which is part of the dynamic KML server.
2. The PHP script gets the request and read the XML temperature sensor data generated by SensorML generator.
3. Meantime, the PHP script retrieves room polygon data from the 3D dynamic GIS database. In the polygon data, each room polygon has an attribute 'SensorID' which indicates the sensor deployed in it and an attribute of the height of the room.
4. According to the height attributes, the PHP automatically generates a 3D cube for each room.
5. According to the sensor IDs and the corresponding temperature read from the XML data, the PHP script find the temperature for each room cube and assign a color corresponding to this temperature.
6. The PHP script then generates KML code representing these colorized room cubes the deliver the generated KML data to the Google Earth plug-in for visualization.
7. For visualization of outdoor temperature, the temperature of the campus is represented as a large colored-dome covering the vicinity of the campus. When viewed from the ground

level of the virtual campus model, the dome replaces the virtual sky and thus users can easily identify the temperature of the campus.

In order to keep the Sensor XML data up-to-date, the SensorML generator keeps querying and reading real-time temperatures from the indoor room sensor databases and the outdoor weather station FTP server every certain interval, which can be set according to specific cases.

3.1 Integrate and Visualize Moving Objects

After the people have been localized in the indoor space, the projected 3D world coordinates can be obtained. Information are then formatted in SensorML format and also stored in the 3D dynamic GIS database. The XML file will be read and parsed by the client JavaScript codes. People can then be visualized using a 2D or 3D sprite for anonymous visualization and to protect their privacy. Figure 2 detailed the overall process from real-time WiFi data collection to indoor position estimation to output data formatting for visualization and rendering on Google Earth.

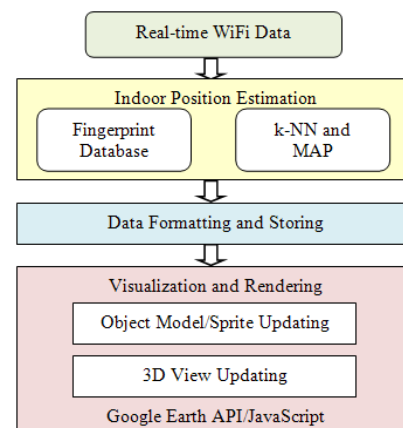


Figure 2. Overview of the process from indoor position estimation to visualization of the indoor positioning data.

4. INTEGRATION OF INDOOR WITH OUTDOOR

For a dynamic virtual 3D world, it is essentially a continuous 3D space through outdoor world to indoor environment. Also, in order for the users to be able to interact with their environment, accurate location and immersive virtual reality information are crucial. In contrast, as the most important data source to provide 3D virtual environment, 3D building models created using current state-of-the-art approaches lack indoor geometric, topologic and semantic information. Researches that address navigation in both outdoors and indoors are limited. Consequently, there is a gap between indoor and outdoor, which has hindered technologies such as mobile augmented reality (MAR) from delivering its promise. The gap can be bridged by the integration of GIS and CAD. On the one hand, 3D Terrestrial Laser Scanning (TLS) data and 2D floor plans are integrated to reconstruct integrated 3D indoor and facade models with both geometric and semantic information; on the other hand, architectural drawings and GIS database are integrated to create a seamless outdoor and indoor navigation database.

4.1 Integration of TLS Data and Floor Plans

Using line template matching, a maximum sequential similarity based algorithm was developed for the co-registration between TLS point clouds and 2D floor plans. First, floor exterior walls line segments are extracted from floor plans. Then a planar surface growing algorithm is used to segment the original TLS

point clouds. The planar information (width, height and the normal direction) obtained during the segmentation process is used to classify the point clouds and the ground points can be distinguished and the noisy non-facade points are removed. A sweep plane is used to cut the extracted facade points into a series of small bands of points with constant height of 0.2 meter from bottom to top. Then TLS wall line segments at every 0.2 height level can be obtained.

Both floor plan wall and floor plan wall line segments are reconnected using virtual lines and decomposed into small matching units with local transformation invariant features and represented using a line matrix. The adapted normalized cross-correlation function is employed to measure the similarity between the line sequence matrices (Equation 1).

$$S = \frac{\sum_{i=1}^4 \sum_{j=1}^n [(T_{ij} - \bar{T})(F_{ij} - \bar{F})]}{\sqrt{\left[\sum_{i=1}^4 \sum_{j=1}^n (T_{ij} - \bar{T})^2 \right] \left[\sum_{i=1}^4 \sum_{j=1}^n (F_{ij} - \bar{F})^2 \right]}} \quad (1)$$

Where: n = number of columns of T, F
 4 = number of rows of T, F
 T_{ij} = value from matrix T at row i , column j
 \bar{T} = average of all digital numbers in T
 F_{ij} = value from matrix F at row i , column j
 \bar{F} = average of all digital numbers in F

A group matching algorithm is applied to simultaneously determine final matching results across floor plans and estimate translation and relative orientation vectors between floor plans and TLS station. The floor plans are then georeferenced and registered to the TLS point clouds.

4.2 Integration of 3D Building Modeling With GIS Database

After registration, the horizontal extents and locations of façade features like windows and doors are known. Then a sweep line based method is used to find the façade feature outlines and vertical locations. Based on the detected facade feature locations, 3D façade model and indoor model are reconstructed simultaneously and integrated into one model. Figure 3 shows an example of the reconstructed integrated 3D indoor and facade model.

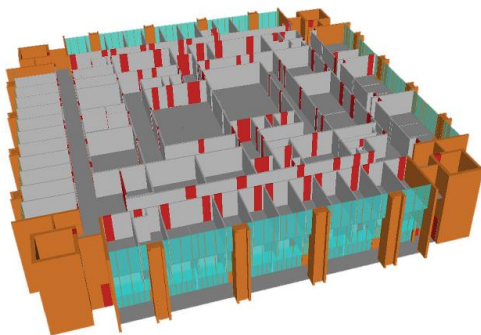


Figure 3. An example of reconstructed integrated 3D building model

From floor plans and other administrative databases, indoor GIS can be created and then integrated with existing outdoor GIS to establish a continuous indoor and outdoor GIS system. The GIS system can then be extended to 3D by integrating the 3D building models. First, room polygons are extracted from the floor plans of the PSE building on Keele campus. Then, attributes, like

building name, floor number, room number, room owner, room type, etc., are created. The values of these attributes can be extracted from floor plans and administrative database. The dynamic and real-time information like room temperature can also be added to this database. All the information are imported to a PostgreSQL/PostGIS database and thus, an indoor building virtual model is established. Because the floor plans were already georeferenced as explained in section 4.1, then the indoor GIS data and existing outdoor GIS data can be seamlessly integrated.

After building the seamless indoor and outdoor GIS database, the 3D integrated building models are also stored in it, which are represented and stored using polygons and multipatches. Then, in the end a seamless dynamic 3D GIS database were established.

5. INDOOR POSITION ESTIMATION

The procedure for WiFi-based indoor position estimation is detailed in this section. The indoor position estimation in this study considers result obtained from a sequential two-steps method. First, a deterministic k-NN method using Manhattan distance estimation was applied to provide a preliminary generalization of the person's location from within a WiFi fingerprint database for the building. This generalization will limit the possible positions of the WiFi user to four locations that are predefined in the database. Then the MAP estimation method will select the most likely position out of the four selected locations by comparing the difference in distance between each estimated location to a pair of known access point locations. The smaller the difference in distance, the higher the likelihood of an accurate estimated position. Distance is obtained by converting received signal strength from nearby access points (AP). The purpose of using this hybrid approach is to increase the effectiveness of a coarsely sampled fingerprint map either due to practical urgency needs for deployment or due to a lack of APs to provide a dense enough WiFi coverage to meet an acceptable indoor positioning precision threshold for a certain application.

5.1 Distance Estimation from WiFi Signal Strength

The Log-Distance Path Loss (LDPL) model is a radio signal propagation model used to estimate path loss of a signal. By using a mathematical model of indoor signal propagation, it can help reduce the dependency on empirical data of the indoor localization algorithm [8]. The model is used to solve the distance d (see Equation 2), between a person at an unknown position and the AP location [2].

$$S = S_{1m} + 10n \log d \quad (2)$$

Where: S = signal path loss between AP and receiver
 S_{1m} = signal path loss at 1 meter away from AP
 d = distance in meter between receiver and AP
 n = path loss exponent of the environment

5.2 Manhattan Distance Estimation

K-nearest neighbour (k-NN) classification is a simple machine learning algorithm that classifies objects based on distance and/or similarity measure. With data points positioned in an input space, the objective of k-NN is to determine which training data are close to it. The function of k-NN is to approximate the data points, the WiFi signal strength measurements in our context, to the closest samples, the pre-recorded data stored in the fingerprint database. It accomplishes this by computing the distance to each data point in the training set using distance estimation methods such as Manhattan distance estimation.

Manhattan distance estimation is to determine the closeness of the data points to the sample data as described in equation (3). In

this method, the distance between two points is computed in signal space. The points can be represented by the real-time WiFi signal strength measurement of a person and the signal strength pre-recorded in the WiFi fingerprint database. According to [6], if the database contain M fingerprints and has a set of locations define as $L = \{l_1, l_2, l_3, \dots, l_k\}$, then to calculate the location represented by a set of received signal strengths ($RSSI_l, RSSI_2, \dots, RSSI_n$), we can define the problem as follow:

For each $l \in 1, \dots, M$ calculate the minimum distance \hat{d}_{Man} between $(RSSI_1^l, RSSI_2^l, \dots, RSSI_n^l)$ and $(RSSI_1^u, RSSI_2^u, \dots, RSSI_n^u)$

Where: l denotes a location
 u denotes a user location
 n denotes no. measurements available at location u
 $RSSI_i^l$ is the signal strength in database from i -th AP
 $RSSI_i^u$ is the observed signal strength
 l_{4Man} is a set of four locations closest to the person

$$\hat{d}_{Man} = \sum_{i=1}^n |RSSI_i^l - RSSI_i^u| \quad (3)$$

$$l_{4Man} = \arg \min_{k=4; l_k \in L} (\hat{d}_{Man}) \quad (4)$$

Upon calculating the minimum distances for all locations in the fingerprint database, equation (4) will be used to infer the top four locations based on their \hat{d}_{Man} .

5.3 Maximum a Posteriori Estimation

This study proposes the use of priori information derived from the k-NN algorithm to determine a person's location. Each of the four nearest locations to the person determined from the k-NN method, in conjunction with distances interpolated from RSSI via LDPL model, can be used to generate conditionally independent probability density functions (pdf) to estimate the most probable location to represent the person's indoor position. One primary argument for using a probabilistic method for location inference is that the results, presented as a probability, are often a better indication of the level of confidence in the location estimation algorithm. These probabilistic location estimation methods store information about the WiFi signal strength distributions in a fingerprint map and use probabilistic inference algorithm to compute the most probable position. The following figure illustrates the notations to be used in the MAP estimation procedure under a simplistic scenario, an open corridor. It is assumed that two APs can be detected at any given time.

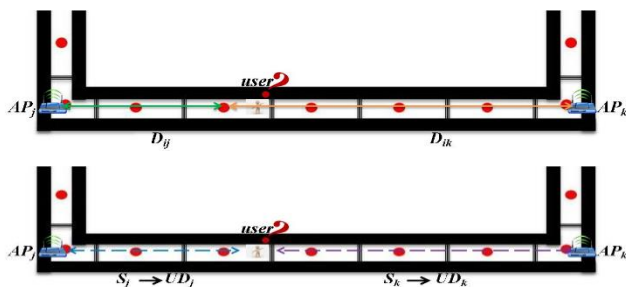


Figure 4. Considering the physical distance between a person and AP (denoted as D) to the WiFi distance and AP (denoted as UD).

The APs where the user is measuring the strongest signals from are depicted as AP_j and AP_k in Figure 4. The red circles are the locations predefined in the fingerprint database, denoted as l_i where $i = 1..4$ in equation (5), selected by Manhattan distance estimation. D_{ij} and D_{ik} represent the physical distances from l_i to the two APs. D_{ij} and D_{ik} can be directly obtained from the database using the coordinates of the APs and locations. Under a corridor scenario, the total distance covered by D_{ij} and D_{ik} should

always be the same for a given corridor. The person who wishes to identify their closest position with respect to the predefined locations in the fingerprint database within the building, will be required to provide two signal values measured from AP_j and AP_k . The two dashed lines in lower Figure 4 represent the user's distances, UD_j and UD_k , connecting the person's position to the two APs. These distances are obtained by interpolating the measured WiFi signal values (S_j and S_k) from the person through the LDPL model. Mathematically Figure 4 is represented below in equation (5).

$$\hat{L} = \underset{l}{\operatorname{argmax}} P(l_{4Man} = l_i | S_j, S_k) \\ = \underset{l}{\operatorname{argmax}} \frac{P(S_j | l_i) \cdot P(S_k | l_i) \cdot P(l_i)}{P(S_j) \cdot P(S_k)} \quad (5)$$

Then the measured signal strengths will be converted into distance through the LDPL model. Thus S_j and S_k will be converted into UD_j and UD_k respectively. In order to calculate the posterior probability, the UD_j and UD_k are assumed to be normally distributed. Therefore, the final equation will be represented in equation (6).

$$\hat{L} = \underset{l}{\operatorname{argmax}} P(l_{4Man} = l_i | UD_j, UD_k) \\ = \underset{l}{\operatorname{argmax}} N(UD_j; D_{ij}, \sigma_{D_{ij}}^2) \cdot N(UD_k; D_{ik}, \sigma_{D_{ik}}^2) \cdot P(l_i) \quad (6)$$

N is the normal distribution of UD_j and UD_k away from AP_j and AP_k with calibrated standard deviations $\sigma_{D_{ij}}^2$ and $\sigma_{D_{ik}}^2$ for each AP and mean physical distance values of D_{ij} and D_{ik} respectively. It is modelled so that the distances at location l_i and user location u to each of the two APs are independent of one another. Equation (6) is considered as finding the maximum posterior probability out of the four location candidates selected using Manhattan distance estimation method by computing the likelihood of the distances D and UD representing the same location in the fingerprint database.

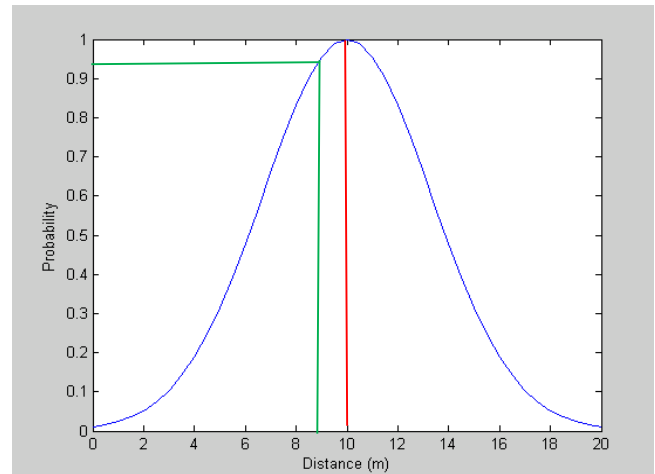


Figure 5. Likelihood function measuring the degree of closest between D and UD .

Figure 5 demonstrates one instance where the physical distance, D (red line) and the user's distance, UD (green line) have approximately 90% probability of being the same location. Since we consider the two likelihood functions separately between a location to each of the two APs, the end result will be affected independently by these two functions. With the lack of dependency of the two functions, the final result will be more reliable. Therefore the final location can be identified by considering the probability of the two likelihood functions (one from each AP to the location) and the prior probability term.

The calculation of the standard deviations $\sigma_{D_{ij}}^2$ and $\sigma_{D_{ik}}^2$ are important as it represent the deviation of D_{ij} and D_{ik} caused by uncertainties such as conversion from signal strength to distance via LDPL model and the fluctuation of signal strength itself. Due to the nature of interacting with two independent APs and subsequently forming two independent user distance pdfs, it is mandatory to have unique standard deviations to characterize the uncertainty of user distance for each AP. Standard deviations of the likelihood functions can be derived from the measured distance discrepancy between D and UD (see Figure 6).

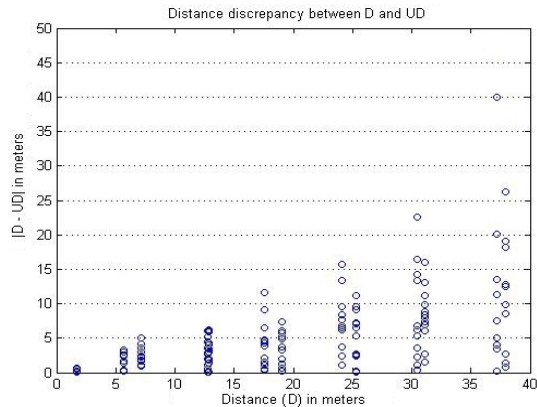


Figure 6. Distance discrepancy measured between the physical distance (D) and the user measured signal distance (UD).

6. SYSTEM TEST RESULTS

We developed a prototype 3D virtual world system for implementing dynamic information based on the above sections. Figure 7 shows the interface of the system. The interface is a 3D virtual campus with several colourized flashing circles indicates the areas having dynamic information. Users can go to the spots to see what is happening by clicking the circles.

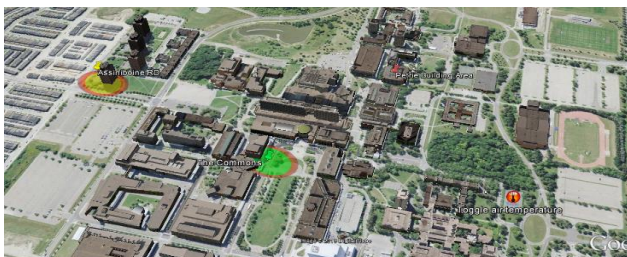


Figure 7. 3D virtual world system interface.



Figure 8. Indoor positioning based on WiFi technique. Detected person can be visualized using simple 3D avatar.

Figure 8 shows the test result of indoor positioning using WiFi-based hybrid technique. The person can be detected using the method presented in section 5 and visualized using simple 3D avatar for anonymity. Additional information can be obtained from the indoor GIS such as room temperature of the nearest room. Table 1 presents some of the preliminary indoor positioning results measured from floor 1 to 3 of the Petrie

Science and Engineering building at York University. The results are represented by the probability of correctly locating a person to the nearest location in the fingerprint database.

Table 1. Indoor positioning results represented by the probability of correctly locating a person to the nearest location in the fingerprint database.

Floor	Positioning Accuracy (k-NN)	Positioning Accuracy (MAP)
1	0.745	0.865
2	0.719	0.909
3	0.745	0.845

7. CONCLUSION AND FUTURE WORK

In this paper, we have presented a web-based system developed based on Google Earth plug-in that brings dynamic indoor positioning information and real-time sensors into an integrated 3D indoor and outdoor virtual world. By integrating TLS data, architectural plans, and existing GIS, a dynamic 3D seamless indoor and outdoor GIS database was built, which acts as the core database to support the system. The system has a flexible and scalable architecture that is open for new developments and applications.

8. REFERENCES

- [1] Bolliger, P.L., 2011. *Robust Indoor Positioning through Adaptive Collaborative Labeling of Location Fingerprints*. Thesis (PhD). ETH Zurich.
- [2] Jung, S., Lee, C., Han, D., 2011. Wi-Fi Fingerprint-based Approaches Following Log-Distance Path Loss Model for Indoor Positioning. *Intelligent Radio for Future Personal Terminals (IMWS-IRSFT), 2011 IEEE MTT-S International Microwave Workshop Series*, 1-2.
- [3] Kim, K., Oh, S., Lee, J., Essa, I: Augmenting Aerial Earth Maps with Dynamic Information from Videos. *Virtual Reality*: 1-16 (2011).
- [4] King, T., Kopf, S., Haenselmann, T., Lubberger, C., Effelsberg, W., 2006. COMPASS: A Probabilistic Indoor Positioning System Based on 802.11 and Digital Compasses. *First ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization (WiNTECH)*, 34-40.
- [5] Kodippili, N.S., Dias, D., 2010. Integration of Fingerprinting and Trilateration Techniques for Improved Indoor Localization. *Seventh International Conference on Wireless and Optical Communications Networks (WOCN)*. 1-6.
- [6] Mautz, R., 2012. *Indoor Positioning Technologies*, Habilitation Thesis. ETH Zurich.
- [7] SensorML, OGC website, <http://www.opengeospatial.org/standards/sensorml>, accessed on May 25, 2011.
- [8] Srinivasa, S., Haenggi, M., 2009. Path Loss Exponent Estimation in Large Wireless Networks. *Information Theory and Applications Workshop*, 124-129.
- [9] Turner, D., Savage, S., Snoeren, A., 2011. On the Empirical Performance of Self-calibrating WiFi Location Systems. *IEEE Conference on Local Computer Networks*, 36, 76-84.
- [10] Wang, L., Sohn, G.: An Integrated Framework for Reconstructing Full 3D Building Models. *Lecture Notes in Geoinformation and Cartography: Advances in 3D Geo-Information Sciences*. Springer-Verlag, Berlin Heidelberg (2011).