Development of Augmented Reality Indoor Navigation System based on Enhanced A* Algorithm

Dexiang Yao[1], Dong-Won Park[1], Syung-Og An[1] and Soo Kyun Kim[1]*
[1] Department of Game Engineering, PaiChai University, *Daejeon, Korea*
[e-mail: kimsk@pcu.ac.kr]
*Corresponding author: Soo Kyun Kim

Received January 8, 2019; revised March 7, 2019; accepted March 30, 2019; published September 30, 2019

Abstract

Nowadays modern cities develop in a very rapid speed. Buildings become larger than ever and the interior structures of the buildings are even more complex. This drives a high demand for precise and accurate indoor navigation systems. Although the existing commercially available 2D indoor navigation system can help users quickly find the best path to their destination, it does not intuitively guide users to their destination. In contrast, an indoor navigation system combined with augmented reality technology can efficiently guide the user to the destination in real time. Such practical applications still have various problems like position accuracy, position drift, and calculation delay, which causes errors in the navigation route and result in navigation failure. During the navigation process, the large computation load and frequent correction of the displayed paths can be a huge burden for the terminal device. Therefore, the navigation algorithm and navigation logic need to be improved in the practical applications. This paper proposes an improved navigation algorithm and navigation logic to solve the problems, creating a more accurate and effective augmented reality indoor navigation system.

Keywords: A* algorithm, Augmented Reality, indoor navigation, path search algorithm, navigation logic

This work was supported by research grant of PaiChai University in 2019. The work is excerpted from parts of first author master's thesis (Dexiang Yao, 2017, "Navigation System based on Augmented Reality for Complex Indoor Environment," PaiChai University Master's Thesis).

http://doi.org/10.3837/tiis.2019.09.016 **ISSN : 1976-7277**

1. Introduction

The modern day cities are growing bigger everyday. Buildings are larger than ever and the interior structures are even more complex. This drives a high demand for precise and accurate indoor navigation systems. Although the existing commercially available 2D indoor navigation system (as shown in **Fig. 1 (a)**) can help users quickly find the best path to their destination, it does not intuitively guide the user to their destination. In contrast, the indoor navigation system combined with AR (augmented reality) technology (as shown in **Fig. 1 (b)**) can quickly guide the user to the destination in real time.

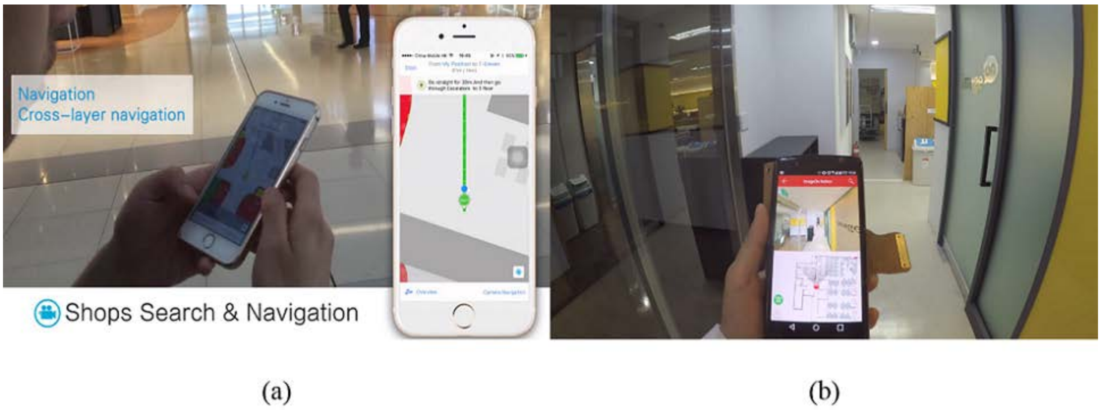


Fig. 1. Comparison between commercial 2D indoor navigation and indoor navigation with AR technology: (a) Commercial 2D indoor navigation in Hong Kong International Airport [1]; (b) Indoor navigation with AR technology in Korean company imageOn [2].

Such practical applications can have various problems such as positioning accuracy, positioning point drift and calculation delay, which can cause errors in the navigation route and result in navigation failure. During the navigation process, frequent calculation and correction of the displayed path will be a great burden for the terminal device. Therefore, the navigation algorithm and navigation logic must be optimized. The paper will introduce the technology of Augmented Reality and indoor positioning in Section 2, and describe in detail of the improved navigation algorithm and navigation logics in Section 3. Lastly, Section 4 provide the results of the real application of the proposed technique, followed by the conclusion section.

2. Augmented Reality and Indoor Positioning

Augmented reality technology [1-2] is a technique that captures a target image in real time through a camera [3-4], which calculates the target’s position and angle and adds corresponding images, videos, and 3D models. Augmented reality technology can not only display real-world information and the corresponding virtual information at the same time, but can also intertwine them to each other. This technique greatly increases the relevance of information, as well as the efficiency of reading related information [5-9].

Augmented reality technology has a profound role in the navigation field, such as the ConnectedDrive HUD system[10] in BMW's concept car. The system uses augmented reality technology to overlay virtual markers on real objects in the external surroundings. It can display navigation information on the exact position of the road ahead of the vehicle, and can prominently display other cars or safety-related objects according to actual conditions.

The advancement of the mobile Internet is increasing the demand for location services. The current indoor positioning technology is in a stage of rapid development. Researchers proposed a number of mature indoor positioning technologies, such as iBeacon Bluetooth positioning technology [11-13], Wi-Fi indoor positioning technology [14-17], and base station indoor positioning technology [18-20]. However, there can be application limitations to different indoor positioning technologies according to their positioning performance, and there is no universal indoor positioning technology that meets all the requirements of the current indoor positioning services.

This article mainly introduces two of the indoor positioning technologies: Wi-Fi indoor positioning technology and base station indoor positioning technology.

Wireless Local Area Network (WLAN) is a mature data transmission system that enables complex positioning, monitoring and tracking services in a wide range of applications. The currently commercialized Wi-Fi indoor positioning technology is a positioning solution based on the wireless local area network standard IEEE802.11. The method measures the intensity of the collected signal of the AP (Wireless Access Point), and uses the triangulation algorithm or the RSSI based fingerprint positioning method to locate the target object. The positioning accuracy is in the range of 3 to 8 meters. The popularity of Wi-Fi networks is driving the positioning accuracy to reach the meter level, lowering the positioning cost, windening the signal receiving range, and strengthening the applicability. However, the augmented reality indoor navigation has more advanced requirements for indoor positioning accuracy, and the positioning accuracy of the Wi-Fi indoor positioning technology still does meet the needs.

The base station indoor positioning technology is an indoor positioning technology that is still in the testing stage. This techinique installs small base stations in complex indoor environments and collects signal data of multi-story buildings through RSSI based fingerprint location algorithm. The technology can accurately distinguish floors by analyzing and comparing data, and also has a high positioning accuracy of 2 to 4 meters. Also, the base station has a large signal receiving range, a powerful ability to penetrate obstacles, and can directly locate the mobile phone with the SIM card. However, in order to enhance the positioning accuracy of the method, it is necessary to increase the density of the base station.

3. Indoor Navigation System

The indoor navigation system is composed of three separate modules: the path search module, indoor positioning module, and navigation module (as shown in **Fig. 2**). A brief overview of indoor positioning technology is provided in section 2. However, this paper will mainly focus on the path search module and navigation module.

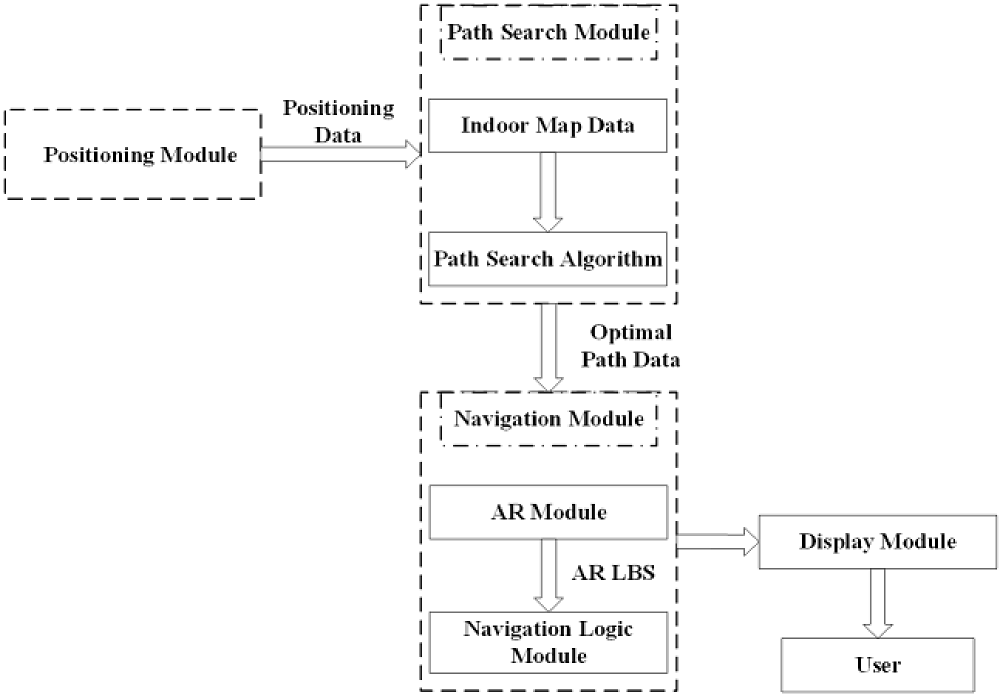


Fig. 2. Indoor navigation system module structure

3.1 Path search module

After the location point of the mobile phone is given by the positioning module, the path search module will then find the optimal paths to the destinations for the users [21]. This subpart focuses on how to find the best path.

3.1.1 Preparatory work

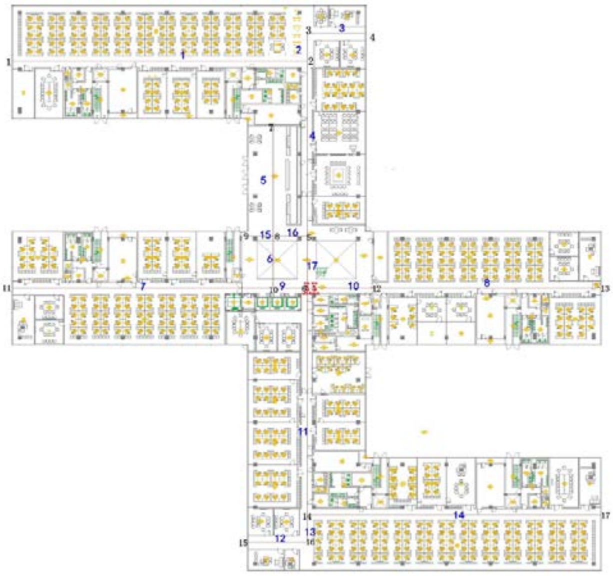


Fig. 3. Indoor environment map. The map used here is generated by the Auto-CAD file of a real building.

Before introducing the details of the naviation system, the groundwork data for the next generation of navigation system must be collected. The preparation steps are as follows:

1) All possible roads are drawn on the map using the red line, forming a road network. In this road network, the crossing point of two or more roads and the end point of a road are defined as the road nodes, marked by the black dots. The path node numbers are marked with black digits and the road numbers with blue digits, respectively (as shown in **Fig. 3**).

2) In order to quickly identify the user's location, the map is equally divided into several different regions. Each region contains various complete roads (as shown in **Fig. 4** - the white lines divide the indoor space into several non-overlapping regions). Therefore, the search can be conducted from coarse to fine to speed up the searching process.

3)

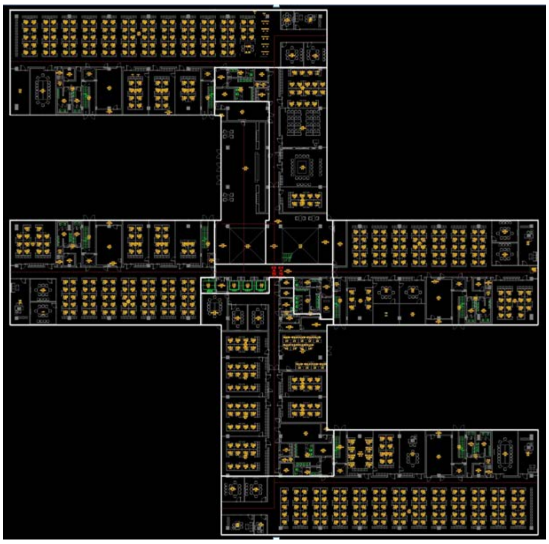


Fig. 4. Indoor space partition map

4) Finally, all the different regions, the roads in each region, and the path nodes must be entered into a table. This data will be utilized as the input data for the path search algorithm.

3.1.2 Path search algorithm

The application of the path search algorithm is necessary in proposed system in order to detect the shortest path that connects the starting point and the target point [22]. There are two main categories of search methods: the method of blind search and the method of heuristic search. Blind search method, also called non-heuristic search, is a type of information-free search that is generally only suitable for solving simple problems. Blind search usually searches according to a predetermined search strategy without considering the characteristics of the problem itself, such as the Dijsktra algorithm. In contrast, Heuristic search method uses the heuristic information possessed by the problem to guide the search, aiming to reduce the search scope and the complexity of the searching process. Examples of heuristic search include the best priority search method [23-24].

Generally, the heuristic search method performs the search for a path more efficiently than the blind search method. However, the path resulting from the heuristic search method is not always the most optimal path. Though the blind search method usually takes longer than the heuristic search method, the result is always the most optimal path possible. Therefore, it is necessary to detect a new technique which combines the advantages of both.

The A* algorithm, that can be treated as a mix of the heuristic search method and the blind search method, is the most effective direct search method for finding the shortest path in a static road network. It is also a commonly used heuristic algorithm for many other problems. The A* algorithm guarantees the shortest path [25] in the road network. A* can be expressed as:

$$f^*(n) = g^*(n) + h^*(n) \tag{1}$$

$f^*(n)$ represents the cost for best path from the starting node *s* to the end node. $g^*(n)$ represents the cost for the way from the starting node *s* to the node *n*. $h^*(n)$ represents the cost for the optimal path from node *n* to the end target node.

$f^*(n)$ is usually unknown, and is used as its approximate estimate:

$$f(n) = g(n) + h(n) \tag{2}$$

$g(n)$ shows the cost of finding for the best path from all the paths which pass through *n*. The heuristic function $h(n)$ shows the estimated cost from *n* to the end target node. $h(n)$ is calculated by estimating the cost of the unsearched paths. The accuracy of $h(n)$ depends on the heuristic information. The lower the estimated cost of $h(n)$, the more optimal the path. Therefore, priority should be given to the resulting path. The basic steps of heuristic search are as follows: calculate the estimated cost for the new node emerged in the search process, select the node with the lowest cost on every search, and start a new search from that node. This is an optimal priority search, and it depends on the cost estimate of the node.

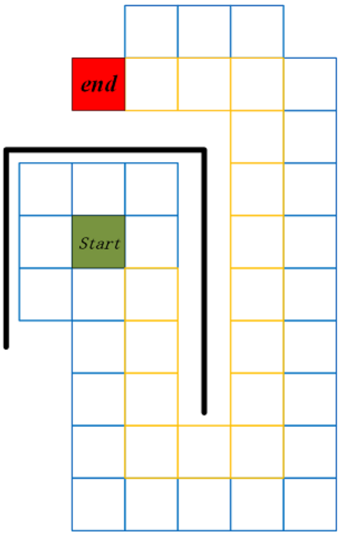


Fig. 5. The meshed indoor map

The standard A* algorithm usually needs to mesh the map with a proper grid. It can find the best path from a starting point to an target point based on this mesh grid (shown by the blue mesh gird in **Fig. 5**). During this searching process, the algorithm will begin from the start point and only move forward one grid on the mesh to approach the end point at each searching step. There are two tables used to store the grids that have gone through the searching process. One table is an $OPEN$ table which stores the grids that have been generated but have not yet been expanded. The other is a $CLOSE$ table which stores the grids that have been generated and expanded. Both $OPEN$ and $CLOSE$ tables store the corresponding values of g and f for the grids.

The working steps of standard A* algorithm are shown as follows:

1) Add the start node s into the $OPEN$ table (expressed as $ADD(OPEN, s)$); add the start node s to the $CLOSE$ table, (expressed as $ADD(CLOSE, s)$); the cost function of the start node s is:

$$f(s) = 0 + h(s) \tag{3}$$

2) Go loop: if $OPEN = NULL$, then $EXIT$. This means if the $OPEN$ table is empty, ends the search algorithm;
3) Expand the eight active grids around the start node s (as shown in **Fig. 6**). Add eight grids into the $OPEN$ table and find the grid n with smallest cost according to **Eq. (2)**. Remove the grid n in the $OPEN$ table, $REMOVE(OPEN, n)$; add the grid n to the $CLOSE$ table, $ADD(CLOSE, n)$;
4) The search process ends, if n is the target node. Pick out the shortest path from node s to node n according to the sequence of the nodes stored in the $CLOSE$ table.
5) If n is not the target node, generate the active grid set M for grid n,which contains the 8 effective grids around grid n;
    a) If the grid M_i in the active grid M *of* set does not exist in the $OPEN$ table or the $CLOSE$ table, add it to the $OPEN$ table, $ADD(CLOSE, M_i)$;
    b) If there is a duplicate grid k in the $OPEN$ table for M_i, and also have $g(M_i) < g(k)$, then remove the grid k in the $OPEN$ table and add the grid M_i, $REMOVE(OPEN, k)$, $ADD(OPEN, M_i)$;
    c) If there is a duplicate grid k in the $OPEN$ table for M_i, and also have $g(M_i) < g(k)$, then remove the grid k in the $CLOSE$ table and add the grid M_i, $REMOVE(CLOSE, k)$, and $ADD(CLOSE, M_i)$; and calculate the values of g and f for the descendant grids in the $OPEN$ table and the $CLOSE$ table;
6) Sort the grids in the $OPEN$ table with increasing order of f;
7) Repeat step 2 to 6 until it ends.

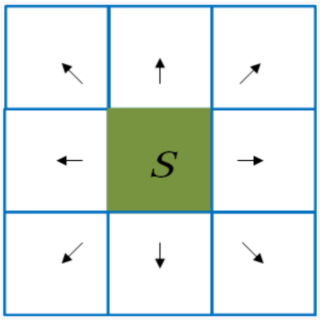


Fig. 6. Eight active grids around the starting node *s*

Fig. 7 shows the $OPEN$ table and $CLOSE$ table for the A* algorithm, when the program has ended. **Fig. 7 (a)** shows the grids stored in the $OPEN$ table and the $CLOSE$ table. Blue boxes and yellow boxes denote the grids that are stored in $OPEN$ table and $CLOSE$ table, respectively. **Fig. 7 (b)** shows each grid in **(a)** and **(c)** with three values: the upper left one denoting the value of $f(n)$, the lower left one denoting the value of $g(n)$, and lower right one denoting the value of $h(n)$. The grid sequence for the optimal path can be found in the final $CLOSE$ table. The grids with red dots in **Fig. 7 (c)** forms the grid sequence of the optimal path.

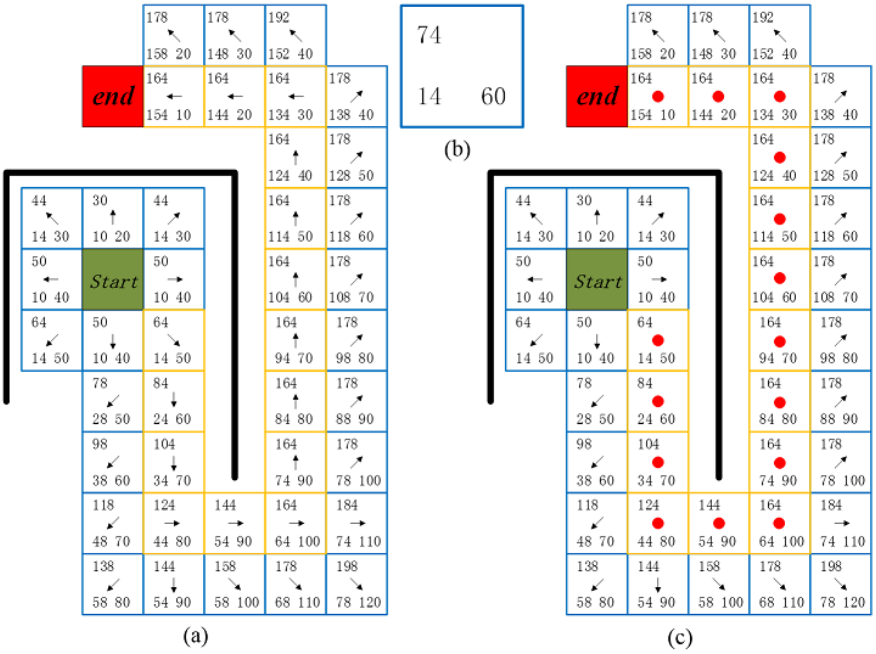


Fig. 7. The final *OPEN* table and *CLOSE* table

For each node, $h(n) \le h^*(n)$ indicates that there are always paths connecting the start node and the end node. When it is satisfied, the A* algorithm will always find the optimal path.

3.1.3 Improved path search algorithm

The heuristic function $h(n)$, which controls the performance of A* algorithm, is critical for finding the optimal path. There are typical cases that $h(n)$ degrades the performance of A* algorithm:

1) In extreme cases, if the value of $h(n)$ is zero, only $g(n)$ is useful. Under these circumstances, A* algorithm will downgrade into the blind search Dijsktra algorithm. Though it can guarantee to find the optimal path, the efficiency is very low.
2) If most of the values of $h(n)$ are smaller than the minimum cost of moving from node *n* to the end node, the best path can be guaranteed to be found by A* algorithm. The smaller the value of $h(n)$, the more nodes which need to be extended in proposed method, and the algorithm runs slower.
3) When each value of $h(n)$ exactly equals the minimum value of moving from the *n* to the end node, the optimal path will be found directly without expanding additional

nodes. In this case, the algorithm runs faster.

4) If most of the values of $h(n)$ are larger than the minimum value of moving from the *n* to the end node, it will not be guaranteed to find the optimal path. In this case, the speed of the algorithm will also be very fast.
5) For another case, if the cost of $h(n)$ is much larger than the cost of $g(n)$, only the value of $h(n)$ will be useful in the algorithm. In this case, A* algorithm becomes a heuristic search known as the Breadth First Search (BFS) algorithm.

The heuristic function $h(n)$ plays an important role for the performance of A* algorithm, affecting running speed and the accuracy in finding the optimal path in a particular indoor environment. However, there are characteristics of these indoor environments that can be used to choose the proper heuristic function $h(n)$. The indoor environment has the following characteristics:

1) Once the environment is determined, the road network in a specific indoor environment will not be changed.
2) The roads in the indoor road network may have multiple branches, which may result in multiple nodes in a single road.
3) Roundabout route may occur.

The A* algorithm is enhanced by improving the heuristic function $h(n)$ for the indoor environment, utilizing the characteristics of the indoor environment.

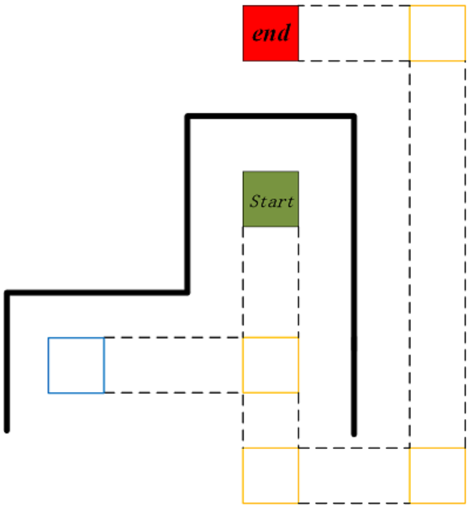


Fig. 8. Node map for indoor environment

The standard A* algorithm searches for the optimal path on a mesh. At every search step, it will only move forward one grid, which is time consuming and impractical. The searching process of standard A* algorithm can be simplified by replacing the meshed map with a node map as shown in **Fig. 8**. The meshed grids are substituted by the nodes of the road, which are denoted by the blue boxes in **Fig. 8.** Based on this simplification procedure, the heuristic function can be established by using node-by-node search based algorithm which can expand the searching from current nodes directly to the next node on the road rather than the eight surrounding grids on the mesh grid. This established heuristic function improves efficiency by eliminating the unnecessary search steps of the A* algorithm. The nodes on multiple branch roads are recorded in the order of expanding. When the nodes are expanded into a wrong direction or at the end of the road, A* algorithm can quickly return to the node on the right road branch to save the time that can be wasted in the wrong search. Therefore, the new heuristic function $h(n)$ cab be expressed as:

$$h(n) = \alpha * distance(n, end), \tag{4}$$

where *distance*(*n*,*end*) is a function that calculates the distance between node *n* and node *end*, $\alpha = \begin{cases} 1, & n \neq end \\ MAX, & n = end \end{cases}$ is the decision coefficient. Through the decision coefficient α, $h(n)$ will control the searching process to return to the previous node and expand nodes in other directions, when encountering end node in a wrong road branch.

Aside from the $OPEN$ table and $CLOSE$ table, a *G* table must also be established to improve the A* algorithm. The *G* table is used to store a list of nodes that have been generated and also have been expanded.

The working steps of improved A* algorithm are presented as shown as follows:

1. Add the starting node *s* to both the *G* table; and add the starting node *s* to both the $OPEN$ table and the $CLOSE$ table, that is $ADD(OPEN, s)$ and $ADD(CLOSE, s)$; the value of cost function for the starting node *s* is:
$$f(s) = 0 \tag{5}$$
2. Go loop: if $OPEN = NULL$, then *EXIT*. If the $OPEN$ table is empty, the search algorithm fails and the process ends;
3. We expand the starting node to the neighbor path node. We can add the path node to the $OPEN$ table and detect the smallest neighboring node *n* according to **Eq. (6)**. Remove the node *n* from the $OPEN$ table, $REMOVE(OPEN, n)$; add the node to the $CLOSE$ table, $ADD(CLOSE, n)$.
$$f(n) = g(n) + \alpha * distance(n, end) \tag{6}$$
4. If *n* is the end node, the finding process succeeds. Sort out the nodes from *s* to *n* for the shortest path according to the sequence of the nodes stored in the $CLOSE$ table;
5. If *n* is not the target node, it can be further classified into two categories:
   a) Extend node *n* directly to the neighboring node *m*, if it does not a path node of a branch way (as shown in **Fig. 9 (a)**). Remove node *m* from the $OPEN$ table, $REMOVE(OPEN, m)$; but add it into the $CLOSE$ table, $ADD(CLOSE, m)$.
   b) Add node *n* into table *G*, if it is a path node of a branch route (as shown in **Fig. 9 (b)**). Find the neighboring node *m* next node *n* that makes $f(m)$ the smallest according to **Eq. (6)** and add the node *m* into the $OPEN$ table, $ADD(OPEN, m)$.
      i. If *m* of node is not the target node of the route, remove it from the *OPEN* table, $REMOVE(OPEN, m)$; but add it into the $CLOSE$ table, $ADD(CLOSE, m)$.
      ii. If *m* of node is not the target node of the route, detect table *G* to find the node *n* in the nearest branch route and return to the node *n*. Remove all nodes after the node *n* in both the $OPEN$ table and the $CLOSE$ table. Extend to the nodes whose cost of *f* is bigger than $f(m)$ but smaller than that of the

neighboring nodes.

iii. If all of the adjcent nodes of *n* are the target nodes of branch routes, detect *G* to find and return the adjecnt branch route node of *n*.

6. Sort the nodes in the $OPEN$ table;
7. Repeat step 2 to 6 until it ends.

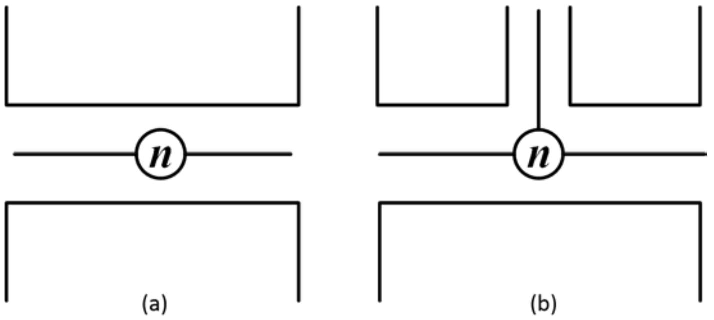


Fig. 9. (a) node *n* is not a branch path node; (b) node *n* is a branch path node.

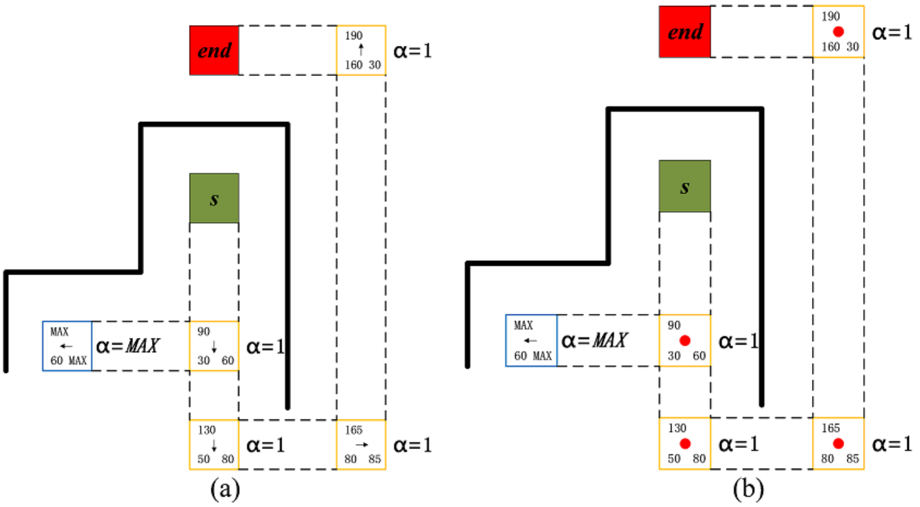


Fig. 10. The final *OPEN* table and *CLOSE* table in the improved A* algorithm.

Fig. 10(a) shows the node stored in the $OPEN$ table and $CLOSE$ table. The blue boxes and yellow boxes represent the nodes that are stored in the $OPEN$ table and $CLOSE$ table, respectively. In **Fig. 10(b)**, the nodes marked with red dots form the node sequence for the optimal path. By comparing **Fig. 10 (b)** with **Fig. 7 (c)**, it is easy to conclude that the searching process for the optimal path is highly simplified by the improved algorithm.

According to the final node sequence stored in the $CLOSE$ table, the start node *s* can be found by back tracking from the final target node when the final target is reached. Thus, the entire node sequence of the optimal path is given. For the next stage of the AR navigation system, this node sequence is utilized to display the road marks by the AR technology to guide the users finding their destination in the real world.

3.2 Navigation module

As discussed in the previous section, the path search module will find an optimal path for the user based on the current positioning location of the user provided by the indoor positioning module. Then the optimal path will be passed to the navigation module in a form of node sequence. The navigation module utilizes the AR-based LBS (Location Based Service – as shown in **Fig. 11**) to provide the users various information, such as information about current locations, the correct direction, distance to the destination, and so on. By using this information, the users is guided to find the destination in a complicated indoor environment.

Fig. 11 shows an example of the AR-based LBS used in the indoor navigation system on a smart phone. The blue sequenced arrows displayed on the screen, which appear imprinted onto the actual floor, help guide the user to the destination. Some other labels also help the user in finding the correct way.

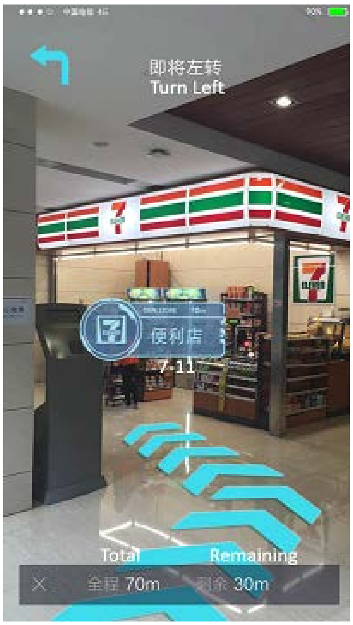


Fig. 11. AR-based LBS

Due to the existence of the positioning error, the positioning point may be far away from the real location of the user, which may cause the path search module to find the wrong path that does not match the actual situation. In this case, the AR-based LBS navigation module will display a wrong path to the user. It will affect the user experience for the indoor navigation system. Navigation logics are used in the navigation module to help correct the impact of the positioning error.

The proposed navigation logics are expressed as follows:

1) Since there is positioning error in the positioning module, the positioning point will randomly locate inside a circle centered on the actual position of the user. Based on the pre-defined road network, positioning point will always be absorbed to the nearby pre-defined road. As shown in **Fig. 12**, both the actual location of the user *X* and its corresponding positioning point *X'* are absorbed onto the points X_1 and X_1' on the pre-defined road (red line), respectively. The new error denoted by the line segment X_1X_1' is definitely smaller than the positioning error denoted

by the white line segment *XX'*. Therefore, the impact of positioning error on the navigation performance can be greatly reduced through the absorbing method.

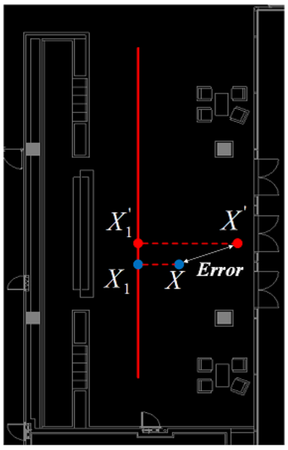


Fig. 12. The error reduction by using the absorption point

2) Sometimes the positioning point may be absorbed onto a wrong road. Two thresholding values β and $\gamma(\beta \le \gamma)$ are used to detect any exceptions.
   a) If the length between the positioning point and the absorption point is smaller than β, the positioning point is absorbed onto the correct road.
   b) If the length between the positioning point and the absorption point is larger than γ, the navigation logic will recall the positioning module to get a new position and do the absorption again.
   c) If the length between the positioning point and the absorption point is smaller than γ and larger than β, the point will be adopted if it is in the forward direction.
3) For every positioning point, a point that is in ahead of the positioning point is selected (as shown in **Fig. 13**). The red dot is the absorption point of the current positioning point. The blue dot is the selected point. Then the angle ε between the path (red line) and the line segment connecting the red dot and blue dot are calculated. According to the sign of ε, we can give the right information for the users about the direction. The turning direction is determind by

$$Inform(\varepsilon) = \begin{cases} turn\,left, & \varepsilon < 0 \\ go\ straight, & \varepsilon = 0 \\ turn\,right, & \varepsilon > 0 \end{cases} \tag{7}$$

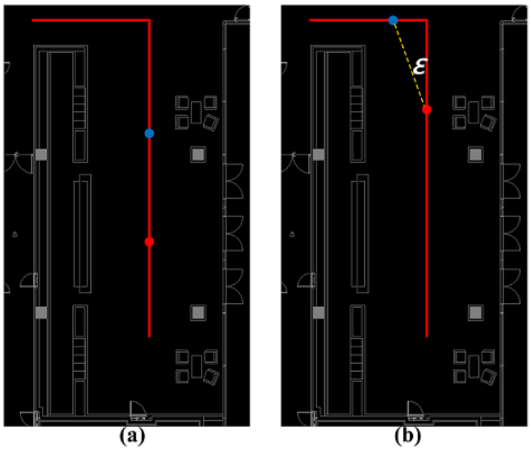


Fig. 13. Turning information. (a) Going straight, (b) Turning left.

Note that by setting proper values of β and γ, the navigation logics can guarantee that the users do not go to the wrong way. For example, when the distance X_1X_1' is bigger than the value of γ, the navigation logic will conclude that the user has been guided to the wrong way. In this case, the user will be informed by the navigation logic to re-search the path, which prevents the user from keep going in a wrong direction without noticing the error.

4. Experiments and Simulation results

The augmented reality indoor navigation system is applied in a practical complex indoor environment to show experimental results. Two different mainstream indoor positioning technologies were configured to compare their navigation effects.

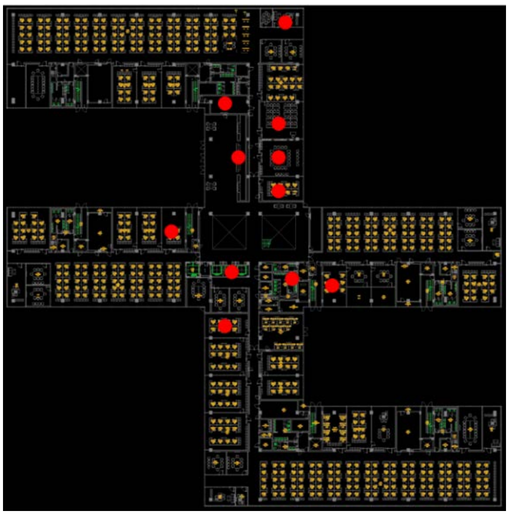


Fig. 14. POI distribution

The test room environment is located on one floor of a large laboratory building. The entire building has a square plan view with a side length of 112 meters and contains multiple test scenarios such as long corridors, circular routes, multiple offices, and large open spaces. As shown in **Fig. 14**, a number of specific POI (Point Of Information) coordinates are preset as navigation destinations that the user can select. The red dot is used to identify it in the picture.

In the test two different indoor positioning technologies are used: Wi-Fi indoor positioning technology and base station indoor positioning technology. Different positioning techniques have different positioning errors, and the fluctuation of the positioning error within the error range is called the “ther error variation”. The smaller the magnitude of the error change, the less likely the navigation will fail.

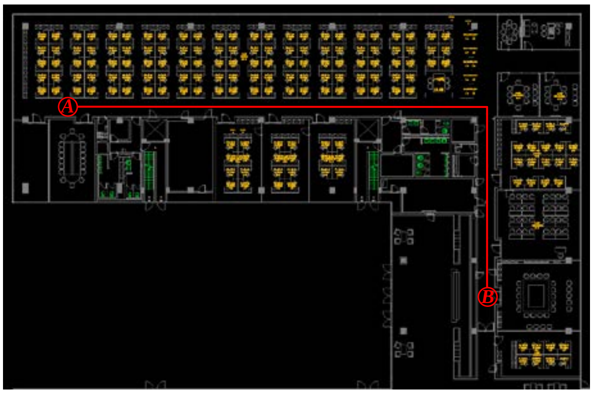


Fig. 15. Test route

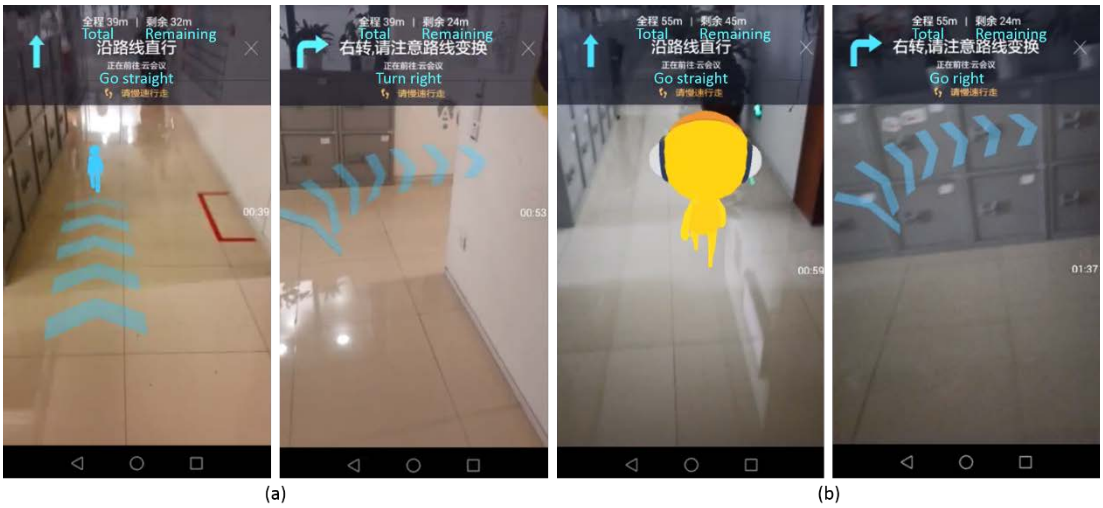


Fig. 16. Navigation performances

The error range of Wi-Fi indoor positioning technology is quite large at 3 to 8 meters. The advantage is that the positioning frequency can be adjusted arbitrarily. As the positioning frequency increases, the variation of error of the positioning point coordinates also increases. Consequently, the positioning accuracy of the positioning point may be affected.

The error range of the base station indoor positioning technology is only 3 to 5 meters. The positioning accuracy of this technology is relatively stable because the error range is relatively small and the highest positioning rate of reccurance which can be set is once per second.

An augmented reality indoor navigation system was implemented in smart mobile devices for Android and iOS. When the system receives the corresponding indoor positioning technology to send the positioning signal, the smart mobile device will display the corresponding augmented reality navigation information for the user.

In **Fig. 15**, we can show the test route from point A to B. Wi-Fi indoor positioning technology and base station indoor positioning technology is used to test the entire augmented reality indoor navigation system. **Fig. 16** shows the test results of the navigation system. (a) and (b) respectively represent the test results of the Wi-Fi indoor positioning technology and the base station positioning technology.

5. Conclusion

This paper presents a stable augmented reality indoor navigation system. The system can achieve similar results by combining different indoor positioning technologies. With the limited computing resources of mobile devices, the system can quickly search for the shortest path to the destination.

However, in order to establish an indoor navigation system for any particular indoor environment, a lot of preparation work is required. For example, collecting indoor environmental information and establishing an indoor map. If new methods to speed up the preparation process are discovered, then the indoor navigation can be spread in a variety of practical applications. This will be the exploration direction of the subject.

Another significant issue in practical applications is the large amount of magnetic field interference sources present in complex indoor environments. The sources of magnetic interference may have an impact on the augmented reality indoor navigation system, causing the navigation path to deviate from the correct direction. Currently there are no viable solutions to remove the effects of magnetic field interference sources. This is also one of the main reasons why the augmented reality indoor navigation system is difficult to commercialize. Going forward, the focus should be on how to eliminate the influence of indoor environmental magnetic interference sources.

References

[1] S. Zlatanova, "Augmented reality technology," *GISt Report No. 17, Delft*, 2002.

[2] Barfield Woodrow, *Fundamentals of wearable computers and augmented reality*, CRC press, Boca Raton, 2015. Article (CrossRef Link).

[3] A.I. Comport, E. Marchand , M. Pressigout, F. Chaumette, "Real-time markerless tracking for augmented reality: the virtual visual servoing framework," *IEEE Transactions on visualization and computer graphics*, Vol. 12, No. 4, pp. 615-628, 2006. Article (CrossRef Link).

[4] F. Zhou, H.B.L. Duh, M. Billinghurst, "Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR," in *Proc. of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, pp. 193-202, 2008. Article (CrossRef Link).

[5] M. Billinghurst, H. Kato, "Collaborative augmented reality," *Communications of the ACM*, Vol.45, No. 7, pp. 64-70, 2002. Article (CrossRef Link).

[6] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, B. MacIntyre, "Recent advances in augmented reality," *IEEE computer graphics and applications*, Vol. 21, No. 6, pp. 34-47, 2001. Article (CrossRef Link).

[7] B. Bach, R. Sicat, J. Beyer, M. Cordeil, H. Pfister, "The Hologram in My Hand: How Effective is Interactive Exploration of 3D Visualizations in Immersive Tangible Augmented Reality?," *IEEE transactions on visualization and computer graphics*, Vol. 24, No. 1, pp. 457-467, 2018. Article (CrossRef Link).

[8] P.H. Wu, G.J. Hwang, M.L. Yang, C.H. Chen, "Impacts of integrating the repertory grid into an augmented reality-based learning design on students' learning achievements, cognitive load and degree of satisfaction," *Interactive Learning Environments*, Vol. 26, No.2, pp. 221-234, 2017. Article (CrossRef Link).

[9] J. Rekimoto, "Matrix: A realtime object identification and registration method for augmented reality," in *Proc. of 3rd Asia Pacific Computer Human Interaction (Cat. No.98EX110)*, pp.63-68, 1998. Article (CrossRef Link).

[10] J. Loewenau, K. Gresser, D. Wisselmann, W. Richter, D. Rabel, S. Durekovic, "Dynamic Pass Prediction-A New Driver Assistance System for Superior and Safe Overtaking," *Advanced microsystems for automotive applications (Springer Press)*, pp. 67-77, 2006. Article (CrossRef Link).

[11] G. De Blasio, A. Quesada-Arencibia, C.R. Garcia, J.C. Rodríguez-Rodriguez, R. Moreno-Diaz, "A Protocol-Channel-based Indoor Positioning Performance Study for Bluetooth Low Energy," *IEEE Access*, vol. 6, pp. 33440-33450, 2018. Article (CrossRef Link).

[12] X.Y. Lin, T.W. Ho, C.C. Fang, Z.S. Yen, B.J. Yang, F. Lai, "A mobile indoor positioning system based on iBeacon technology," in *Proc. of 37th Annual International Conference of the IEEE*, pp. 4970-4973, 2015. Article (CrossRef Link).

[13] J. Yang, Z. Wang, X. Zhang, "An ibeacon-based indoor positioning systems for hospitals," *International Journal of Smart Home*, Vol. 9, No. 7, pp.161-168, 2015. Article (CrossRef Link).

[14] K. Chen, C. Wang, Z. Yin, H. Jiang, G. Tan, "Slide: Towards Fast and Accurate Mobile Fingerprinting for Wi-Fi Indoor Positioning Systems," *IEEE Sensors Journal*, Vol. 18, No. 3, pp. 1213-1223, 2018. Article (CrossRef Link).

[15] B. Shin, J.H. Lee, T. Lee, H.S. Kim, "Enhanced weighted K-nearest neighbor algorithm for indoor Wi-Fi positioning systems," in *Proc. of 8th International Conference on Computing Technology and Information Management (NCM and ICNIT)*, Vol. 2, pp. 574-577, 2012.

[16] S.H. Jung, B.C. Moon, D. Han, "Performance evaluation of radio map construction methods for wi-fi positioning systems," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 18, No. 4, pp. 880-889, 2017. Article (CrossRef Link).

[17] J. Petajajarvi, K. Mikhaylov, R. Yasmin, M. Hamalainen, J. Iinatti, "Evaluation of LoRa LPWAN technology for indoor remote health and wellbeing monitoring," *International Journal of Wireless Information Networks*, Vol. 24, No. 2, pp. 153-165, 2017. Article (CrossRef Link).

[18] J. Rekimoto, "A hand-held augmented reality system for collaborative design," *Proceeding of Virtual Systems and Multimedia*, Vol. 96, pp.18-20, 1996.

[19] M.M. Buddhikot, S. Sen, D. Samardzija, T. Zhang, S. Walker, U.S. Patent No. 9,553,70, Washington, D.C., U.S. Patent and Trademark Office, 2017.

[20] R. Zhao, B. Zhong, Z.L. Zhu, L. Ma, J.F. Yao, "Overview of Indoor Localization Techniques and Applications," *Electr. Sci. Tech.*, Vol. 27, pp. 154-157, 2014.

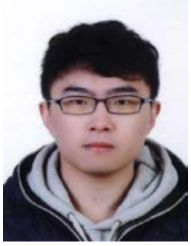
[21] P. Dahne, J.N. Karigiannis, "Archeoguide: System architecture of a mobile outdoor augmented reality system," in *Proc. of IEEE International Symposium on Mixed and Augmented Reality*, pp. 263-264, 2002. Article (CrossRef Link).

[22] H.J. Nilsson, "Patent: Principles of artificial intelligence," *Morgan Kaufmann*, 2014.

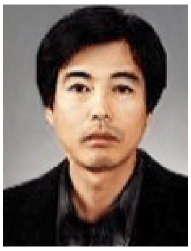
[23] X. Cui, H. Shi, "A*-based pathfinding in modern computer games," *International Journal of Computer Science and Network Security*, Vol. 11, No. 1, pp. 125-130, 2011.

[24] M. Li, Y. Zhang, S. Li, "The Gradational Route Planning for Aircraft Stealth Penetration Based on Genetic Algorithm and Sparse A-Star Algorithm," in *Proc. of MATEC Web of Conferences*, Vol. 151, 2018. Article (CrossRef Link).

[25] W. Zeng, R.L. Church, "Finding shortest paths on real road networks: the case for A," *International journal of geographical information science*, Vol. 23, No. 4, pp. 531-543, 2009. Article (CrossRef Link).



Dexiang Yao received the B.S. and M.A. degree in the Department of Game Engineering at PaiChai University, Daejeon, Korea. He is Ph.D. Candidate in the Department of Game Engineering at PaiChai University, Daejeon, Korea. His research interests include fical animation and 3D fical reconstruction.



Dong-Won Park is a professor in the Department of Game Engineering at Paichai University, Korea. He received Ph.D. in Computer Science Department of Texas A&M University, College Station, Texas, U.S.A, in 1993. He was a visiting scientist at the Real-Time Research Center of Texas A&M University from 2000 to 2001. His research interests include image untderstanding,virtual and mixed reality.

Syung-Og An is a professor in the Department of Game Engineering at Paichai University, Korea. She received M.S. and Ph.D. in Computer Science & Engineering Department of Korea University, Seoul, Korea, in 1984 and 1989, respectively. She was a visiting professor at the University of Pennsylvania State University from 1993 to 1994. Her research interests include multimedia system, computer graphics and database.

Soo Kyun Kim received Ph.D. in Computer Science & Engineering Department of Korea University, Seoul, Korea, in 2006. He joined Telecommunication R&D center at Samsung Electronics Co., Ltd., from 2006 and 2008. He is now a professor at Department of Game Engineering at Paichai University, Korea. Dr. Kim has published many research papers in international journals and conferences. Dr. Kim has been served as Chairs, program committee or organizing committee chair for many international conferences and workshops; Chair of ICCCT'11, ITCS'10, HumanCom'10, EMC'10, ICA3PP'10, FutureTech'10, ACSA'09, Em-Com'09, CSA'09, CGMS'09, ISA'09, SIP'08, FGCN'08 and so on. Also Dr. Kim is guest editor of the International Journal of "IET Image Processing" and "Multimedia Tools and Applications". His research interests include multimedia, pattern recognition, image processing, mobile graphics, geometric modeling, and interactive computer graphics. He is a member of ACM, IEEE, IEEE CS, KACE, KMMS, KKITS and KIIT.