Journal of Korea Spatial Information Society Vol.21, No.2 : 1-9, April 2013
http://dx.doi.org/10.12672/ksis.2013.21.2.001 ISSN 2287-9242(Print)
ISSN 2287-9250(Online)

FT-IndoorNavi: 토폴로지 분석 및 실내 경로 네트워크 분석에 기반한 실내 네비게이션을 위한 유연한 네비게이션 알고리즘

FT-Indoornavi: A Flexible Navigation Method Based on Topology Analysis and Room Internal Path Networks for Indoor Navigation

주 건* 이 연** 이 순 조*** 배 해 영****
Jian Zhou Yan Li Soon Jo Lee Hae Young Bae

요 약 최근 이동통신 기술의 발전으로 실내에서의 위치 획득 기술이 용이해지면서 실내 내비게이션 시스템에 대한 연구가 각광을 받고 있다. 최적화된 실내 내비게이션은 크고 복잡한 실내에서의 활동을 위해 빠른 경로를 제공한다. 그러나 기존의 실내 내비게이션 알고리즘 연구들에서는 공간 토폴로지 또는 실내 네트워크에 기반하여 대략적인 경로를 도출한다. 본 논문에서는 공간 토폴로지 분석과 실내 내트워크를 통합 분석하여 실내 경로를 계산하는 알고리즘을 제안한다. 제안 알고리즘은 전문적인 실내지도가 아닌 간소화된 실내 지도 데이터를 활용하여 실내 네트워크와 공간 토폴로지의 혼합 분석을 통해 실내 경로의 길이를 줄일 수 있다. 성능평가를 통해 FT-Indoornavi 알고리즘이 기존의 엘라스틱 알고리즘과 iNav 알고리즘보다 더 빨리 실내 경로를 계산할 수 있음을 보여준다.

키워드 : 공간 토폴로지 분석, 실내 경로 네트워크, 간소화 실내 지도

Abstract Recently many researches have focused on indoor navigation system. An optimal indoor navigation method can help people to find a path in large and complex buildings easily. However, some indoor navigation algorithms only calculate approximate routes based on spatial topology analysis, while others only use indoor road networks. However, both of them use only one of the spatial topology or network information. In this paper, we present a navigation method based on topology analysis and room internal networks for indoor navigation path. FT-Indoornavi (Flexible Topology Analysis Indoornavi) calculate internal routes based on spatial topology and internal path networks to support length-dependent and running-time optimal routing, which adapt to complex indoor environment and can achieve a better performance in comparison of Elastic algorithm and iNav.

Keywords : spatial topology analysis, the internal path network, complex indoor environment

1. Introduction

LBS (Location Based Service) system[17,12] is usually combined with digital maps to provide the users their location and related routing information. Now it is widely used in many fields such as traffic navigation[6], social networking[3], public safety, mobile commerce[11]. With the

* Jian Zhou, Master Student, Dept. of Computer & Information Engineering, Inha University. zhoujiancqupt@gmail.com
** Yan Li, Ph.d Student, Dept. of Computer & Information Engineering, Inha University leeyeon622@gmail.com
*** Soon-Jo Lee, Professor, Dept. of Computer Education, Seowon University. sjlee@seowon.ac.kr(Corresponding Author)
**** Hae-Young Bae, Professor, Dept. of Computer & Information Engineering, Inha University. hybae@inha.ac.kr

continuous development of digital city construction and the gradual popularization of mobile terminals, there is an increasing demand for the location service of some large and complex indoor environments[13] such as the shopping centers, car parks and museums. However, when comparing outdoor and indoor space, it is not difficult to find that the indoor internal structures are more complex than the outdoor ones, such as the doors, corridors, rooms which do not exist in outdoor space. And the complicated structures result in a complicated path network. It is not easy for even the familiar persons to find a place in the complicated indoor space. If the person is unfamiliar with indoor space, it is very likely to get lost. Therefore, the demand for proper guidance[15] in complex indoor environments is becoming even stronger than ever. This leads to the rapid development of indoor routing services. At present, many LBS systems are mainly designed for routing in outdoor space. But less for indoor space. Specialized indoor routing solutions are becoming urgent needs. So it is necessary to make a high performance, fast and continuously updating indoor navigation system.

When a person pass through space, he can be generalized to finding the shortest path between two nodes in a network. There are some well proven algorithms, such as Dijkstra[4], A*[8], Elastic algorithm[2], iNav[20,7] and Context-Aware[1,3]. Those models consider only the topological relationship of indoor geometries[16], or the internal path networks of geometry, or the relationship of link-node[14] to get navigation route. At present, there is existing following problems in the indoor navigation system. Firstly, they neglect combining the internal path networks[21,9] with the topology relationship of indoor geometries, so the navigation path is not the shortest one. Secondly, because road network represents relationship between points and points, so not only the indoor map but also the connection information of points should be updated when indoor environment has been changed. However, this will lead to increase the amount of updated data, and the more data will increase more errors. Last, Since the indoor road network information is not enough, part or even the whole of the navigation path may be wrong.

In order to solve those problems mentioned above, This paper proposed FT-Indoornavi (Flexible Topology Analysis Indoornavi) algorithm, which combined with the information of rooms, room internal path networks, corridors and so on in indoor space is proposed. FT-Indoornavi abandons the road network which used in Dijkstra algorithm. A model is proposed in this paper to support length dependent optimal routing based on the geometry of indoor structures (rooms, room internal paths, lobbies and the others), and make full use of the relation of rooms, room internal path networks and lobbies. It takes the spatial topology analysis[18] and room internal path networks to get the navigation path according to the location and destination of user. One shortest route represents a finite sequence of path segments which is obtained by applying a shortest path algorithm in indoor space. The detailed routing information, such as turns and the length of the route, can be obtained from the path segment.

This paper is organized as follows: Section 1 introduces the concept of indoor navigation algorithm, and section 2 reviews related work on the indoor navigation algorithm. In section 3, an indoor network is built and discussed about its benefits of navigation. In section 4, an experiment of FT-Indoornavi is showed. Finally, we conclude our work and put forward future research direction in section 5.

2. Related Work

In this section, a set of key concepts will be introduced, such as indoor space object topology relation, link-nodes, road-networks[10], rooms, doors, corridors, lifts, lobbies and stairways. Several models have been proposed to support human-oriented indoor navigation using those cells. Purely symbolic models are based on a labeling system without considering the object topology relation of the indoor space. Elastic algorithm is based on indoor space object relation without considering the rooms and doors of the indoor space, while iNav is based on rooms, doors and corridors without considering the object topology relation of the indoor spaces.

Purely symbolic models like Dijkstra is a typical of node-link model algorithm, this kind of algorithm must use the relationship of link-node to calculate navigation route. There are two shortcomings in these models: one is the aspect of mistake tolerance, when there exists error between the relationship of link road and node, the navigation system will crash or not run normally. For example, the information of a link which connects two nodes, either of them is not existed. Or the information of nodes which is in the link table does not exist in node table. Such mistakes will affect the normal operation of the navigation system. The second shortcoming is that it needs to update the indoor spatial data quickly. In some places like large-size shopping malls, facilities are often changed. So it must update the link-node's relationship and navigation route. In this case we need update twice data (indoor spatial data and link-node navigation data), and we should pay more attention to the link-node's relationship when the relationship is changed. These operations undoubtedly increase the difficulty of updating data.

Another type of indoor navigation algorithm is Elastic algorithm, this kind of algorithm only considers topology relationship among geometries without using the relationship between link and node. It is based on indoor space object topology relation, and then obtain navigation route according to the coordinate information of spatial geometries. As Figure 1.a shows, the coordinate points(S->A->B->C->T) in navigation route from S to T are obtained based on information of geometry. This algorithm is suitable not only in indoor environment but also in outdoor environment. This algorithm just only uses spatial geometries, but it does not check whether there is an internal road existing in spatial geometry or not. So if the internal geometry has common road, we can obtain better navigation route using it.

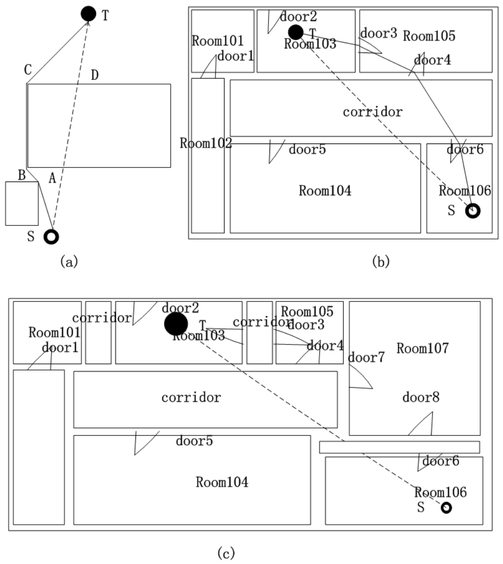

Figure 1. Consider geometry (a), Consider doors and rooms (b), Example for consider geometry, doors, rooms(c)

iNav and Context-Aware are a type of algorithm which concernes room internal paths and indoor space object relation. It only considers doors, rooms and the common part room internal path networks, so this algorithm considers in simple indoor space, as Figure 1.b shows, the navigation route is (S->door6->door4->door3->T). However, there is many information of geometry in complex indoor space, for example large-sized shopping malls (Figure 1.c). There are so many channels in geometry's inner part or among the geometries. In this case, we must consider topology-based indoor space objects first, and then the inner path of geometries.

3. FT-Indoornavi Path Planning Algorithm

In order to solve the problems in section 2, this paper has put forward a new shortest route algorithm model which combines the room internal path networks with the indoor space object topology relation. In large-sized complex indoor space, we consider the topology relationship among geometries firstly, and then make use of inner channel of geometry. So it can obtain more ideal and optimized indoor navigation shortest path. The following three problems are solved by this method:

(1). FT-Indoornavi does not use link-node, so it is not necessary to update link-node data. It just needs to update the indoor space spatial geometries data.

(2). It makes use of indoor space object topology relation and the room internal path networks to obtain the shortest navigation route.

(3). It is suitable not only in the simple indoor space but also in large scale complex indoor space, even when indoor map is not clear to navigate.

3.1 Indoor Space Based On Unit Structure

The base units of room have a number of different kinds of architecture. Although some of the rooms have similar shapes, but they may serve different room internal path networks. And some of them are totally different in shapes but may play the same role during selecting the route. For example, there is only one door in the room, it means there is no room internal path networks. So we just consider this room which people cannot pass through the room internal. In this case, someone only goes into this room. When one room has two doors or more doors, there are more than one path inside of the room. So we can use those paths to get in one optimal route. The following items show three base unit room types:

(1). One door in a room: One door room is a unit which is closed by walls. If someone wants to go into this room, he only needs access one point (door). So this one door unit is a very special architecture which can a constraint of the routing. This type of base unit room has just only one door, there is no internal path in the room, so it can only be considered as a start object or an end object. As shown in Figure 2.a, there is just only one door which can be the entrance, the black point is the door and the lines are walls.

(2). Multiple doors in a room: Multiple doors room is a unit that which includes two doors or more. There are more than one internal path in the room. Therefore, when we calculate navigation path, we must consider the room internal. It is very important for us to get the optimal navigation path. As shown in Figure 2.b, the black point is the door and the lines are the walls.

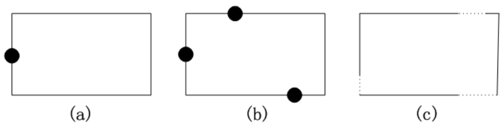

Figure 2. One door for the Room (a), Multiple doors for the Room (b) and Open door for Room(c)

(3). Multiple open doors in a room: One open door for room we just think it is one door for room. Because there is just one way can get into the room. As shown in Figure 2.c, multiple open doors room includes two or more than open doors, the dotted lines are the open doors and the full lines are walls. Because the open door is very wide, we think those open doors consist of the start point and the end point.

3.2 Indoor Graphic Data Structure

This paper stores indoor graphic data structure with the Mysql[22]. Room includes doors, open doors and their spatial data. Before we calculate the Indoor navigation path, we need to deal with the indoor graphic spatial data at first. The following figure is the structure of the indoor graphic data.

Indoor graphic data include following respects: room spatial data, MBR(minimum bound rectangle)[5], DoorPoints.

Room spatial data: we put the rooms as polygons like Figure 3.B. We store those polygons with the data structure like Figure.3.E. This data structure is the Mysql store spatial data structure.

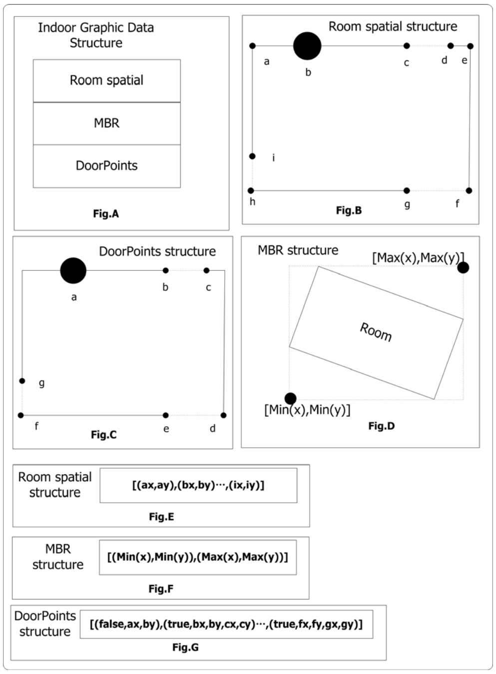

Figure 3. Indoor Graphic Data Structure

MBR: The minimum bounding rectangle (MBR),also known as bounding box or envelope, is an expression of the maximum extents of a 2-dimensional object (e.g. room) within its 2-D (x, y) coordinate system, in other words min(x), max(x), min(y), max(y). The MBR is a 2-dimensional case of the minimum bounding box. As the Figure 3.D shows, the room's MBR is the dotted line rectangle, and we record the dotted line rectangle the min(x), max(x), min(y), max(y). Then we set the MBR data structure like the Figure 3.F.

DoorPoints: Because room spatial includes two kinds of door types. We set DoorPoints structure have two types of structures. When DoorPoints unit has not the open door, we set the first data is false, and the next is one point. If the door is an open door, we change the first data to true and the next data includes two points(the open door's start point and the open door's the end point). We store all of the data with clockwise order. The Figure 3.C is the DoorPoints example, The Figure 3.G shows the data structure of the DoorPoints.

3.3 FT-Indoornavi Path Basic Planing

When we are in a strange indoor space, shops are here and there. Let's make a hypothesis, we want to go to T place from S now, we generally move by the direction of our destination. In this case, a wall hinders us, we will choose route (S->A->B->C->D->T) to bypass the room's wall like the route described in Figure 4.(a1,b1,c1). This method does not consider indoor room internal although there is a common path. But as shown in many cases, there is a room internal in an indoor room, and we can use these room internal channels to get shorter path. For example in Figure 4.a1 there is no room internal channel, so we can use the (S->A->B->T) arrive the T point. But in Figure 4.(b2, c2), there is a room internal channel and we can choose the right indoor room channels to make the navigation path shorter. Like Figure 4.(b2, c2), the path is (B2:S->A->B->C->T), (C2:X->A->B->T). According to the Figure 4.(a2, b2, c2), we can use room internal channel to get shortest navigation path, so according to this model, this paper presents a new optimal shortest navigation path algorithm.

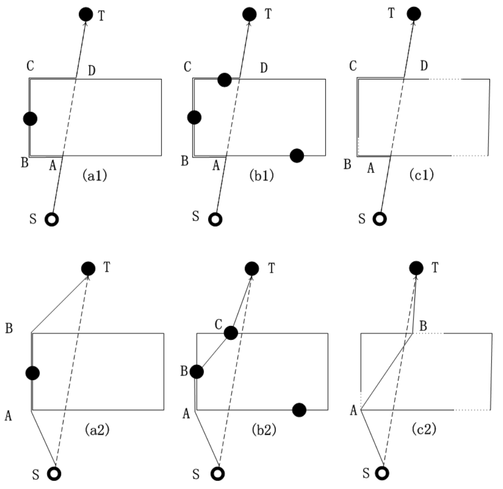

Figure 4. Basic Indoor Navigation Path

```
Process list<Point> getPath
(startpoint,endpoint)
{
    pStartA = startpoint;
    pEndB = endpoint;
    pPaths.add(new Point(pStartA));
    // get the nearest Room for A
    Indoor_Room pTemp_Room =
    getNearestRoom(pStartA, pEndB);
    if (pTemp_Room.GID != null) {
        // get the room after cutting
        //the recent all point information
        List<Point> pPnts =
        getNearestRoomPath(
        pStartA,EndB, pTemp_Room);
        // the small part of the
        //room Point Information
        List<Point> roomSmallParts =
        getRoomPartInfo(pPnts);
        // according to the
        //points get path
        pPaths.add(
        getPartRoomPath(roomSmallParts));
        getPath(pStartA, pEndB);
    }
    pPaths.add(new Point(pEndB));
    Return pPaths;
}
```

Figure 5. Pseudo Code Of The Algorithm

The above description is a simple example of a single room. Now there are a lot of rooms on the same layer. We will pass a lot of rooms from S point to T point. This paper uses relationship between rooms, and room internal path to get the best navigation path.

According to the algorithm design ideas, this paper puts forward the following steps for indoor navigation path.

1. Connect the starting point(SP) and target point(TP) for line L. Then initialize navigation path coordinates stack S. And add the start point coordinates in S;

2. Get with L at the intersection of room. if there is no intersection of room, push the TP into S, then return S(the navigation path). otherwise, executive step 3;

3. To obtain the nearest point to the starting point of the room;

a) selection is straight-line cutting room smallest part of A, and set the intersection point A1, A2(A1 is near starting point and A2 is near target point);

b) If there is no door or only one door in the part of A, it is executed (e). If A contains two doors, push them into S. Use top of the stack(S) as start point executive step 2. If part A includes two or more doors executed (c). If this part contains two or more open the door executed (d);

c) Compare these doors' coordinate with point A1 and A2. Get A1's the nearest door's coordinate push it into S. Get A2's the nearest door's coordinate push it into S. Use top of the stack(S) as starting point executive step 2;

d) Compare point A1 and point A2 with open doors' D1, D2 (D1 consist of d11 and d12, D2 consist of d21 and d22). If point A1 between d11 and d12 or between d21 and d22, then push A1 into S. If not, compare A1 with d11, d12, d21 and d22, then get the nearest one point push it into S. Make the A2 operation as A1. Then Use top of the stack(S) as starting point executive step 2;

e) The smallest part of the vertex coordinates based on the distance to the starting point. With small to large distance order push them into S. Then Use top of the stack(S) as starting point executive step 2;

According to the algorithm described above steps, the pseudo code of the algorithm is in the Figure 5.

3.4 FT-Indoornavi Algorithm Analysis

The comparison study of FT-Indoornavi algorithm and other link-node algorithms is showed in Table 1.

Table 1. Compare network link-node and FT-Indoornavi

	Update	Fault tolerance
Network link-node	Update geometry(such as rooms, corridors, lifts, lobbies and stairways) and link-node net work	If Connection relations between the link and node have some errors, this link-node algorithm system cannot run, and system will shut down.
FT-Indoornavi	Just update geometry such as rooms, corridors, lifts, lobbies, stairways, doors and so on.	If indoor space geometries have any error data, this algorithm maybe get the path which have not the nearest path, but this system can a navigation path.

Table 2. Use or not Rooms relationship and room internal paths

	Topology Analysis in indoor space	Room internal paths
FT-Indoornavi	Used	Used
Elastic algorithm	Used	Not Used
iNav,Context-Aware	Not Used	Used

FT-Indoornavi algorithm proposed in this research puts forward technology of navigation algorithms which combines the relationship among geometries with inner channel of geometry. We compare it with the kind of algorithm which only uses the relationship among geometries or the inner channel of geometry, Table 2 shows the result.

FT-Indoornavi algorithm model will not involve link-node relationship, it combines the relationship among geometries with inner channel of geometry, can obtain navigation route quickly, acquire improvement on updating data and fault tolerance aspects. This model greatly increases the utilization rate of data which includes indoor room, door, corridor. And will not use to node-link data (network data), so that the algorithm is more optimal.

4. Performance Evaluation

4.1 Experiment Environment

FT-Indoornavi algorithm is based on eclipse platform, java programming language, MySQL database and jdbc(java database connectivity). The indoor map data source comes from WanDa,Chongqing China. Data includes four floors' map, the size of the data is 150 rooms. There are about 37 records on every floor(This data is room information on the floor).The hardware of computer is CPU with Intel(R) Core, 2GHz，Ram 2.00gb,hard disk 500G,Windows XP operating system.

4.2 Performance Evaluation

FT-Indoornavi algorithm in this paper acquires navigation path by combining spatial analysis and room interior channel of special indoor environment. Compared with other navigation algorithms which have the same start point and end point, such as Dijkstra, Elastic, the distance of navigation path computed by the proposed algorithm is shorter than the one acquired by Dijkstra and Elastic. Moreover, cost time of acquiring shortest path by the proposed algorithm is also less than the one by Dijkstra, but slightly more than the one by Elastic. This is because Elastic algorithm calculates the spatial relationship between indoor rooms only. It does not need to compute room interior channel. In order to get the shortest navigation path, the proposed algorithm also involves room interior channel. The following is the comparing data between the proposed algorithm, Dijkstra and Elastic algorithm.

There are eight groups of data in Figure 6, and every group of data represents the navigation path distance acquired by three algorithms which

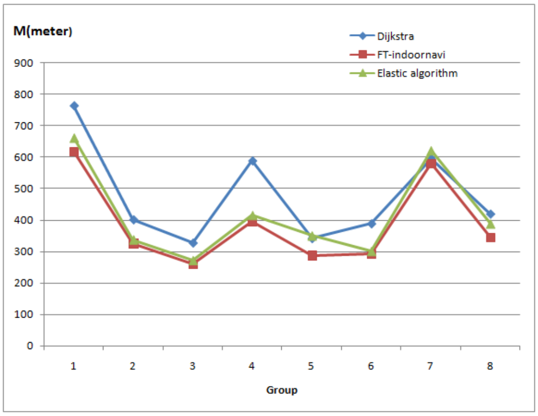

Figure 6. The Evaluation Result Of Path Distance

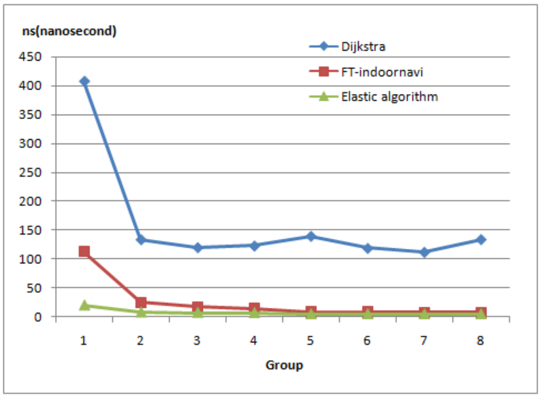

Figure 7. The Evaluation Result Of Path Computation Time

have the same start point and end point. Rhombus represents the typical Dijkstra shortest path distance, triangle represents Elastic one and square represents the FT-Indoornavi. We can see from Figure 6 that the distance of acquired navigation path using the proposed algorithm is shorter than the one using Dijkstra and Elastic. The proposed algorithm is better than Dijkstra and Elastic at the aspect of the navigation path distance.

There are eight groups of data in Figure 7, and every group of data represent running time cost by three algorithms which have the same start point and end point. Rhombus represents the cost time of computing shortest path by the typical Dijkstra, triangle represents the one by Elastic. and square represents FT-Indoornavi. At the aspect of running time, the proposed algorithm is less than Dijkstra. Because Elastic had not considered whether there are viable channels in a room. But FT-Indoornavi algorithm considers, the running time is slightly more than Elastic. Although FT-Indoornavi algorithm is slightly more than the Elastic algorithm, in terms of computing time it can provide users with shorter than Dijkstra and Elastic shortest path selection, which has application value.

5. Conclusions

FT-Indoornavi puts forward a new flexible indoor navigation algorithm which makes good use of indoor facilities (room, corridor, door and so on) in complex and special indoor space. This algorithm will not depend on road network structure. It has many advantages such as high effectiveness and high rate of fault tolerance. And it is suitable for indoor and outdoor environment. In addition it does not need to node-link complex link relationship. It can offer more efficient and superior navigation route. But when the room internal is very complex, we pass the room internal which will spend us so much time. In the future, we want to make this algorithm more widely applied to actual circumstances and use it on smart phone to offer user better service.

References

[1] Afyouni, I; Ray, C; Claramunt, C; 2012, Spatial models for indoor and context-aware navigation systems: A survey. Journal of Spatial Information Science, 4: 85-123.

[2] Chen, Y; Wu, K; Wang, D; Chen, G. 2006, Elastic Algorithm: A New Path Planning Algorithm About Auto-navigation in 3D Virtual Scene, ICAT 2006: 1156-1165.

[3] Dao, E. A. 2012, Spatial models for con-

text-aware indoor navigation systems: A survey, International Journal of Geographical Information Science, 26 (3), 469-494.

[4] Dijkstra, E. W. 1959, A note on two problems in connexion with graphs[J], Numerische Mathematik, 1(1): 267-271.

[5] Faloutsos C; Ranganathan M; Manolopoulos Y. 1994, Fast Subsequence Matching in Time-Series Databases. Proceedings of ACM SIGMOD Int'l Conference on Management of Data, Minneapolis, Minnesota, pages 419-429.

[6] Guha, R. K; Chen, W. 2009, A distributed traffic navigation system using vehicular communication, Proc. of 1st. IEEE Vehicular Network Conference.

[7] Goetz, M; Zipf, A. 2011, Formal Definition of a User-Adaptive and Length-Optimal Routing Graph for Complex Indoor Environments, Geo-spatial Information Science, 14(2): 119-128.

[8] Hart, P. E; Nilsson, N. J; Raphael, B. 1968, A formal basis for the heuristic determination of minimum cost paths[J], Transactions on Systems Science and Cybernetics SSC, 4(2): 100-107.

[9] Hoppenot, P; Pradel, G; Caleanu, C; Perrin, N; Sommeilly, V. 2003, Towards a Symbolic Representation of an Indoor Environment, IEEE CESA 2003.

[10] Hector, G; Han J, W; Li, X. L; Myslinska, M; Sondag, J. P. 2007, Adaptive Fastest Path Computation on a Road Network: A Traffic Mining Approach. VLDB 2007: 794-80.

[11] Lu, E. H; Lee, W. C; Tseng, V. S. 2011, A Framework for Personal Mobile Commerce Pattern Mining and Prediction, IEEE Transactions on Knowledge and Data Engineering, v.24 n.5, p.769-782.

[12] Lee, K; Lee J. 2013, A Geocoding Method on Character Matching In Indoor Space, Journal of Korea Spatial Information Society, 21(1):87-100.

[13] Lee, S; Lee J. 2011, Navigable Space-Relation Model for Indoor Space Analysis, Journal of Korea Spatial Information Society, 19(5):75-86.

[14] Park, I; Jang, G. U; Park, S; Lee, J. 2009, Time-Dependent Optimal Routing in Micro-scale Emergency Situation. 10th Int. Conf. on Mobile Data Management: Systems, Services and Middleware, pages 714-719.

[15] Raubal, M; Egenhofer, M. 1998, Comparing the complexity of wayfinding tasks in built environments[J], Environment and Planning B-Planning & Design, 25(6): 895-913.

[16] Shen, J. W; Wu, M. G; LV, G. N; Wen, Y. N; Wang, X; Zhang, H. P. 2010, Topological relationships calculation for 3D curves data set based on monotone chains. In Geoinformatics, 2010 18th International Conference on, pp. 1-5.

[17] Wang, X. l ; Pang, X. Y;Luo, Y. 2010, W. LBS-p: A LBS Platform SupportingOnline Map Services, Vehicular Technology Conference FSpatial models for context-aware indoor navigation systems: A survey, International Journal of Geographical Information Scienceall (VTC 2010-Fall): 1~5.

[18] Xu, T. Y; Zhou, Y. C. 2012, Topology Analysis of Rural Power Distribution Network Based on Spatial Database, Power and Energy Engineering Conference (APPEEC), 2012 Asia-Pacific.

[19] Yu, Z. W; Liang, Y. J; Xu, B. K; Xu, Y; Guo, B. 2011, Towards a Smart Campus with Mobile Social Networking, in Proc. 2011 4th International Conference on Cyber, Physical and Social Computing, pp:162-169.

[20] Yuan, W. J; Markus, S. 2010, iNav: An Indoor Navigation Model Supporting Length-Dependent Optimal Routing. In Proc. of AGILE, pages 299-314.

[21] Yuan, W; Schneider, M. 2010, Supporting 3d route planning in indoor space based on the lego representation. In: 2nd ACM SIGSPATIAL Int, Workshop on Indoor Spatial Awareness (ISA), pp. 16 - 23.

논문접수 : 2013.02.02
수 정 일 : 2013.03.11
심사완료 : 2013.03.25