

THE AUTOMATIC GENERATION OF AN ADAPTIVE NAVIGATION MODEL FOR INDOOR MAP MATCHING

Pengyuan Wang^{1,2}, Jianga Shang^{1,2*}, Zhiyong Zhou³, Yijie Wu^{1,2}, Weixin Sun⁴

¹ School of Geography and Information Engineering, China University of Geosciences, Wuhan, China
- (pengyuanwang, jgshang, wuyijie)cug.edu.cn

² National Engineering Research Center for Geographic Information System, Wuhan, China

³ GIScience Center, Department of Geography, University of Zurich- zhiyong.zhou@geo.uzh.ch

⁴ Beijing Satellite Navigation Center, Beijing, China- swxgis@163.com

Commission IV, WG IV/5

KEY WORDS: Indoor positioning, Indoor map matching, Navigation model, Space representation, Automatic generation

ABSTRACT:

Indoor map matching has been an important technique to improve the indoor localization accuracy because it takes the advantage of available indoor building data and effectively decreases the localization cost. One of the spatial model involved in the indoor map matching, adaptive navigation model, could balance the accuracy and complexity of the model representation required by map matching by combining the medial axes and fine-grained grids according to the movement characteristics of pedestrians in open and narrow areas. In order to reduce the manual effort of producing a large number of models and update them, we propose an algorithm to automatically generate this model. Furthermore, we use this algorithm to generate this model for three lab architectures, and the evaluation of the results of the ANM generated by the algorithm proves that the algorithm meets the requirements.

1. INTRODUCTION

Indoor map matching is a process of matching raw positioning data to indoor maps or navigation models. This algorithm can be divided into geometric analysis, topological analysis, probabilistic map matching algorithms and advanced map matching algorithms (Quddus et al., 2007). The geometric analysis (Bernstein et al., 1996) only uses geometric information to match, which is relatively simple and easy to implement, because it does not consider the topological relationship, it does not allow to correct the error in time and has the poor stability. Topological analysis (Quddus et al., 2003) uses geometric information, topological relationships and historical collection point information, etc. This algorithm has some improvement in matching efficiency and accuracy but is still susceptible to noise. Probabilistic map matching algorithms (Honey et al., 1989) sets a confidence region for each signal point, calculates the probability value according to the distance of the positioning point in the confidence region, and determines the best matching path according to the size of the value. This approach allows pedestrians to deviate from the road, but it is difficult to implement and computationally expensive. The advanced map matching algorithm including Kalman filter (Obradovic et al., 2006), the application of Bayesian inference (Pyo et al., 2001) and the hidden Markov model (Pink, Hummel, 2008), etc, considers comprehensive information such as noise in the trajectory data. These algorithms have higher accuracy but still perform poorly at low sampling rates.

Different map matching algorithms require different navigation models (Taneja, 2013) including grid-based, centerline-based, metric-based, which calls for an unified model for indoor map matching. Hilsenbeck (Hilsenbeck et al., 2014) proposed an

adaptive navigation model (ANM) based on the characteristics of pedestrian movement trajectory with more degrees of freedom in open areas and is similar to the medial axis of space in narrow spaces. ANM combines the movement characteristics of pedestrians to degenerate the freedom of movement of pedestrians in narrow areas to one-dimensional, and still uses two-dimensional representation in open areas. This navigation model not only ensures the representation of geometric information and topological relations in the open area but also reduces the complexity of the model in the narrow area.

However, Hilsenbeck did not make an accurate definition of ANM nor did it propose an automated generation method for this model. Manually generating such a navigation model for large buildings is time-consuming, the refurbishment and retrofit buildings have an average annual 30% change in floor space over their life cycle (Roper, Payant, 2014), the navigation model needs to be updated with the renewal of the building. Therefore, we also proposed an algorithm for automatically generating the navigation model. The algorithm consists of three parts: unit classification, centerline generation and grid generation.

2. RELATED WORK

2.1 Centerline generation algorithm

Since Blum proposed the concept of medial axis transformation (Blum, 1967), many researchers have made outstanding contributions in this field. These researchers have made various definitions of the medial axis based on their own fields, and they all contain the fact that the medial axis is a curve. These researchers have proposed a number of medial axis transformation algorithm, including: Voronoi

*Corresponding author

method (Lee, 1982), topological thinning (Latecki, Ma, 1996), morphological thinning (Jang, Chin, 1990), simulations of grassfire propagation (Blum, 1967), distance transformation (Remy, Thiel, 2005) and so on. The advantage of these medial axis transformation algorithm is that they can perform well for any complex polygon, but their disadvantage is that the generated medial axis is a curve.

When performing edge-edge map matching, the track segment to be matched is a straight line segment, and the exist of the curve in the medial axis will cause the map matching of the track segment difficult. In order to meet the requirements of map matching for straight medial axis, Lee et al. proposed a straight medial axis transformation (S-MAT) algorithm (Lee, 2004). This algorithm generates an angle bisector from the vertices of the polygon, and continues to generate the next level bisector at the intersection of the angle bisectors, iterating the process until the final two bisectors intersect. This works well with convex polygons, but ends early with "I" concave polygons (Figure 1(a)). Tenja (Taneja et al., 2016) modified the rule of generating angular bisector at the concave corner of S-MAT algorithm and proposed a modified straight medial axis transformation (M-MAT) algorithm, which overcomes the disadvantages of S-MAT algorithm, but generates jagged medial axis at the concave corner of "L" shape polygon (Figure 1(b)). The advantage of these straight medial axis algorithm is that it can generate the straight medial axis directly, but the disadvantage is that it can not execute some concave polygons correctly.

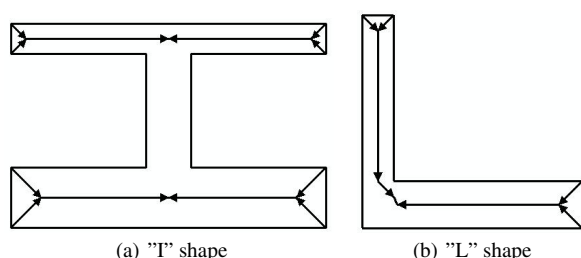


Figure 1. The defect of the straight medial axis algorithm

2.2 Grid generation algorithm

The grid consists of grid edges and grid nodes. There are two types of grid: construction grid and non-construction grid (Ho-Le, 1988). Correspondingly, the grid generation algorithm can be divided into two categories: mapping function method and non-mapping function method. The mapping function method includes the mapping element approach and the conformal mapping approach (Ho-Le, 1988). Conformal mapping approach is difficult to control grid shape and grid density and is rarely used. The mapping element approach requires that the target area be manually divided into a number of simple sub-areas that are advantageous for mapping, and then the mapping function is defined to map the irregular areas into a regular area. After meshing on the regular area, the results is inversely mapped to the original irregular area. The disadvantage of this approach is that it requires manually define simple subregions and mapping functions. Non-mapping function method can automatically generate grids, including delaunay triangulation (Bowyer, 1981), advancing front method (George, Seveno, 1994), finite quad (-oc)-tree method (Legrain et al., 2011) and so on.

The grid generation algorithm now mostly uses the

non-mapping function method, the finite quad (-oc)-tree method is used to generate the uniform shape and size grids, and the delaunay triangulation is used to generate the grids of irregular shapes and sizes. Irregular grids cannot be used for map matching, so this paper does not discuss them. The grids generated by these algorithms are mainly used for path planning and shortest path queries. The advantage of the non-mapping function method is that grid generation can be performed automatically for all target areas, but the grid generated by the non-mapping function method (Figure 2) has the following disadvantages in the map matching field:

1. According to the proportion of each grid occupied by obstacles, the grid is divided into navigable and unnavigable, and the judgment error increases with the increase of grid size.
2. Non-navigable grids cannot be matched and should be deleted.
3. In a complex building, there are units with inconsistent tilt directions. The mesh division of the target area as a whole will cause inconsistencies in grid density in these units.

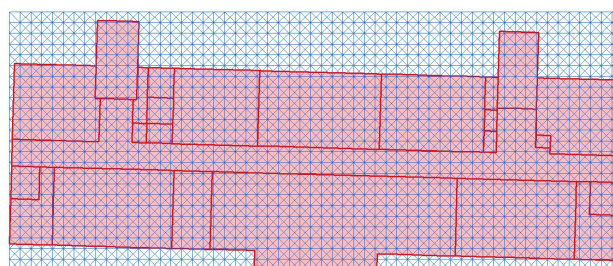


Figure 2. Grid generated by non-mapping function method

The above disadvantages lead to the quality of the grid generated by the non-mapping function method is insufficient to meet the requirements of map matching.

3. THE DEFINITION OF ANM

In this section, we define AMN based on the movement characteristics of pedestrians in indoor spaces. On the one hand, pedestrians tend to stay at a certain distance from physical obstacles when walking. In narrow areas (such as corridors), people need to maintain a safe distance from the walls on both sides and most of the moving trajectories of pedestrians are located near the medial axis (Figure 3(a)), so that the centerline can be used for indoor map matching. On the other hand, the appearance of other passers-by on the way of walking makes pedestrians deviate from the original route, because pedestrians have subconscious insecurities and discomfort when they meet strangers (Helbing, Molnar, 1995). In open areas (such as halls), people have more space to avoid other pedestrians on the way, and the pedestrian's movement trajectory has no obvious directionality (Figure 3(b)), which requires a grid to provide more geometry and topological relationship information. Here, we define narrow spaces and open spaces:

Definition 1 (narrow space) A space in which the moving trajectory of pedestrians is mostly located near the medial axis and approximately linear, with less freedom of movement

Definition 2 (open space) A space in which pedestrians have greater freedom of movement.

In the actual positioning requirements, for some small space such as elevators, models only needs to provide a point in the unit to match the positioning result to the unit. In order to reduce the complexity of the ANM and improve the accuracy of the algorithm for unit classification, we have added a definition of point space:

Definition 3 (point space) A space whose area is less than $2.25m(1.5m*1.5m)$ is represented by a node in ANM.

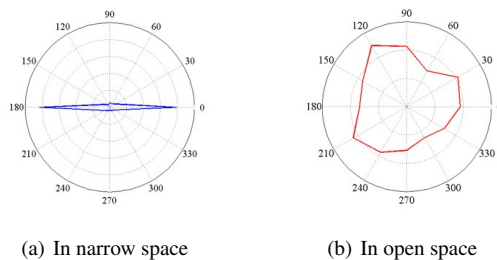


Figure 3. Pedestrian trajectory distribution in different types of spaces (Wang et al., 2016)

Indoor map matching algorithm can be further divided into point-to-point matching, point-to-edge matching and edge-edge matching according to the matching geometric form. In terms of geometry, this paper considers that the essence is to compare a certain time-stamp position point or a trajectory segment of a certain time period with a prior knowledge map to find the position points or track segments with large probability errors and correct them. These methods need to abstract the indoor space into a network of edges and nodes.

In order to reduce the complexity of the matching, the length of the edges of the network needs to be close to the length of the track segment. Therefore, we divide the medial axis by the length of the track segment and set the grid size to the length of the track segment. In this paper, T-length is used to represent the length of the track segment, which is approximately equal to the pedestrian step size. In summary, ANM is a road network consisting of edges of approximately T-length and nodes including their endpoints and the midpoint of the door. Its formal representation is as follows:

$$ANM = (V_a, E_a) \quad (1)$$

where $V_a = \{grid\ nodes, centerline\ nodes\}$
 $E_a = \{grid\ edges, centerline\ edges\}$

4. THE GENERATION ALGORITHM OF ANM

The input of the algorithm is an indoor map of ESRI's ShapeFile format, and the output is an ANM. The algorithm first extracts the indoor unit, then judges the type of the extracted indoor unit, generates a grid for the unit of the open space, generates a centerline for the unit of the narrow space, and stores the generated grids and centerline into ANM. The above process is iterated until all indoor units are processed to obtain the final ANM (Figure 5). The flow chart of the algorithm is shown in Figure 4.

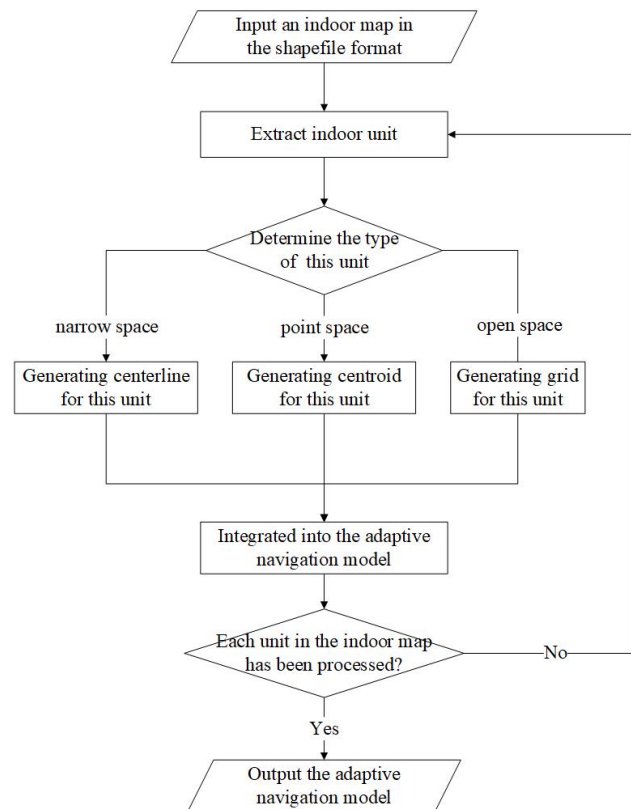


Figure 4. Flow chart of algorithm

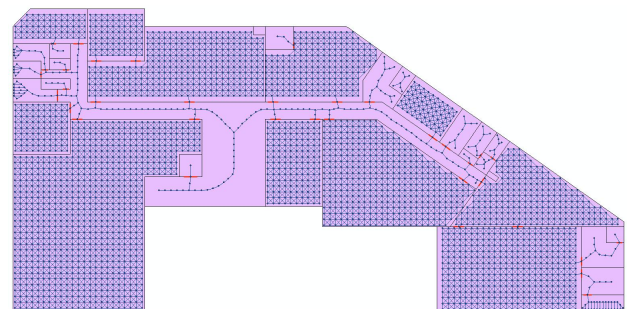


Figure 5. The adaptive navigation model

4.1 Unit classification

The algorithm first determines the unit type from the geometry. The usual practice is to compare the ratio of perimeter and area of the unit. However, for the jagged corridors with several protruding platforms, this method will cause errors in judgment. In the field of Geographic Information System (GIS), a buffer is an area defined by the bounding region determined by a set of points at a specified maximum distance from all nodes along segments of an object (Shashi Shekar, 2011). The buffer is defined by the unit boundary and reflects the geometry information of the unit.

If the half width of the unit is larger than the buffer radius, the unit has a portion that is not covered by the buffer, and this part is named openspace (Figure 6). The area of openspace is controlled by the buffer radius and the unit width. For a certain unit, the width is a fixed value, so the area of openspace is determined by the buffer radius. If the buffer radius is determined as a critical value, the part whose width is greater

than this critical value in the unit will generate openspace, while the part whose width is less than this critical value will not generate openspace.

In buildings, narrow space is mainly corridor. In order to ensure that corridor can be accurately judged as narrow space, this critical value is determined as half of corridor width in this paper. Most corridors have been measured to have a width of less than 3 meters, so the critical value is determined to be 1.5 meters in this paper.

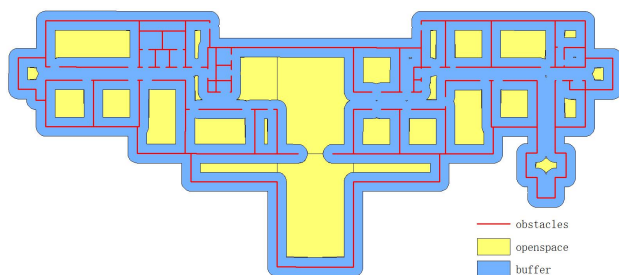


Figure 6. The buffer and openspace of indoor units

If the unit is a simple polygon. If the unit width is smaller than the buffer radius, the buffer completely covers the unit. If the unit width is larger than the buffer radius, the buffer cannot completely cover the unit. The type of the unit can be judged according to whether the unit is completely covered by the buffer.

According to the definition of the buffer, there is a positive correlation between the proportion of the openspace and the proportion of the portion whose width is larger than the buffer radius. Therefore, the ratio of the area of the portion not covered by the buffer to the unit area can be used to represent the ratio of the portion of the unit whose width is larger than the buffer radius. This ratio can be further considered as the ratio of the open part to the unit. When it is larger than a certain value, the open area is more, and the type of this unit is open. When it is smaller than this value, the type of this unit is narrow. This value is the threshold used to determine the type of unit.

In summary, the indicator for judging the type of the unit of the simple polygon and the complex polygon can be unified. The part of the unit that is not covered by the buffer is named openspace. The indicator is the ratio of the area of openspace to the area of the unit. If this ratio is greater than the threshold, the unit has more open parts. The type of this unit is open, otherwise the type is narrow. After experimenting with multiple floors of multiple buildings, this threshold was determined to be 0.3.

4.2 Centerline generation

In this paper, the Voronoi method is used to generate the medial axis. First, the Voronoi diagram of the indoor space is generated. Then the Voronoi edges of the Voronoi diagram are extracted. Finally, the part where the Voronoi edges intersects the boundary of the indoor space and the first level edges are deleted to obtain the medial axis.

For an indoor unit, extract the door and the medial axis segments within the unit. The extracted medial axis segments are tidied up as follows: several medial axis segments with the same trend, end-to-end and no bifurcation points are merged

Algorithm 1: determine unit type

Input: input indoor map with shapefile format

Output: unit type

```

1 generate a buffer with radius of 1.5m for boundary;
2 foreach unit in the indoor map do
3   set opensapcei = the portion of this unit not covered by this
   buffer;
4   set oparea = the area of opensapcei;
5   set unit_area = the area this unit;
6   if unit_area < 0.25 then
7     set sign = "pointspace"
8   else if oparea/unit area < 0.3 then
9     set sign = "narrowspace"
10  else
11    sign = "openspace"
12  end
13 end
14 return sign;
```

into one (such as Figure 7 AB, BC, BD). A short medial axis segment with an endpoint that is an isolated endpoint is deleted. The isolated endpoint is an endpoint whose is not the endpoint of any other medial axis segment and is not on the door.

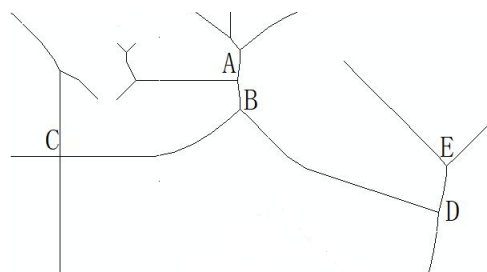


Figure 7. The medial axis to be tidied.

For a certain medial axis segment, the strict division according to T-length usually leaves a short edge, which affects the accuracy of map matching, so this short edge should be evenly distributed among other edges. The length of the medial axis segment is rounded to the T-length to get the number of segments it is divided into, which is called the split-number in this paper. The length of the medial axis segment is divided by split-number to get the length of each segment from the subdivision, which is called the split-length in this paper. Then starting from one end of the medial axis segment, the split points is determined by split-length. Finally, the end points and the split points are connected end to end to get the approximate straight medial axis segments (Figure 8).

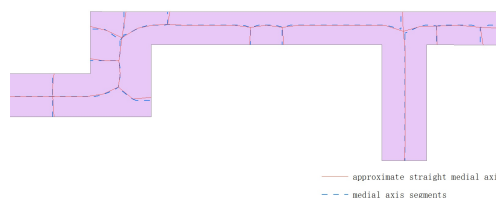


Figure 8. The approximate straight medial axis

In this research, the connectivity of the units is ensured by its door, so it is necessary to connect the approximate straight medial axis within the unit and unit's door. In order to ensure that the length of the edges of the centerline is similar to T-length, the connection between the door and the approximate straight medial axis need to be processed. First calculate the

distance between the midpoint of the door and the endpoints of the approximate straight medial axis segments, and find the nearest endpoint. If the distance is greater than the diagonal length of a square with a length of T -length, that is $\sqrt{2} \cdot T$ -length, the line connecting the midpoint of the door and the endpoint is divided into corresponding segments by the method of dividing the medial axis above. If the distance is less than $(\sqrt{2} - 1) \cdot T$ -length, the coordinates of this endpoint is replaced with the coordinates of the door's midpoint.

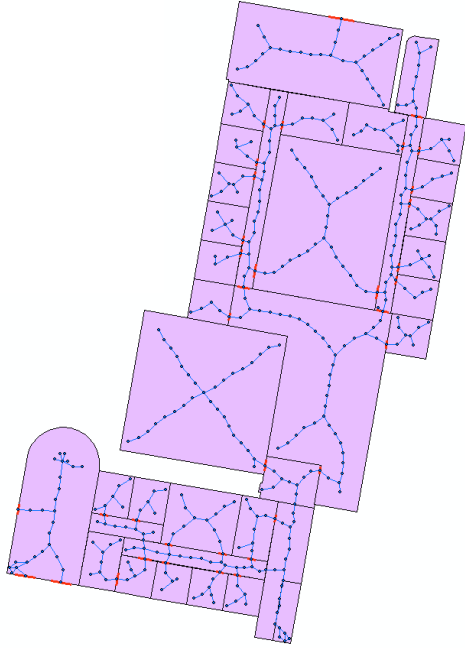


Figure 9. The centerline of indoor units

After performing the above operations on the approximate straight medial axis and the door in the unit, the edges of the centerline is obtained. Extract the endpoints of these edges as the nodes of the centerline (Figure 9). It is necessary to remove duplicate points and points at doors when extracting the endpoints. The door is shared by two units and added after all units have been processed to avoid node duplication.

4.3 Grid generation

The building was designed and built into a relatively independent unit, such as meeting rooms, offices, bathrooms, etc., in order to use the function. Therefore, it is only necessary to determine the mapping function when performing mesh division on the indoor space using the Mapping Element Approach. The minimum bounding rectangle (MBR) is the minimum of the range of all objects (eg, points, lines, polygons) in all directions (Toussaint, 1983). The direction of its long side can represent the main direction of the complex unit, and such a regular area can be used as a range of grid division. Therefore, the process of finding the mapping function can be equivalent to the process of finding the minimum circumscribed rectangle of the unit, and the tilt angle and range of the MBR of the unit are regarded as the tilt angle of the unit and the range of the mesh division, and the grid is meshed within the MBR, which can be completed automatically by the algorithm. In summary, the grid generation algorithm in this paper adopts the Mapping Element Approach.

Algorithm 2: centerline generation

Input: indoor map with shapefile format
Output: centerline

```

1 generate a Voronoi diagram for the indoor map;
2 extract the Voronoi edge as the medial axis;
3 foreach unit in the indoor map do
4   set  $medial\_axis_i$  = the medial axis in this unit;
5   set  $door_i$  = the collection of doors in this unit;
6   tidy the  $medial\_axis_i$ ;
7   foreach media segment in  $medial\_axis_i$  do
8     get split points;
9     connect the endpoints and the split points in turn;
10    edges of the centerline  $\leftarrow$  connection lines;
11  end
12  nodes of the centerline  $\leftarrow$  endpoints of edges;
13  foreach door in  $door_i$  do
14    calculate distance between the nearest node and the door's midpoint;
15    if distance >  $\sqrt{2} \cdot T - length$  then
16      split the connection line between the nearest node and the door's midpoint;
17      edges of the centerline  $\leftarrow$  the splitted connection line;
18    else if distance <  $(\sqrt{2} - 1) \cdot T - length$  then
19      set  $nearest\_node = door's\ midpoint$ ;
20    end
21  end
22 end
23 nodes of the centerline  $\leftarrow$  midpoints of door;
24 removes duplicate points from the centerline's nodes;
25 return centerline;
```

To ensure that the grid is evenly distributed within the cell, the grid should be centered within the MBR. In this paper, the grid size is determined as T -length to meet the requirements for indoor map matching. The length and width of the MBR are usually not integer multiples of T -length, which causes a gap between the grid and the MBR. The gaps need to be evenly distributed around the grid, so that the starting point coordinates of the grid are (Figure 10):

$$S_x = B_x + r \cdot \cos \alpha; \quad S_y = B_y + r \cdot \sin \alpha \quad (2)$$

where

$$\begin{aligned}
r &= \sqrt{w^2 + l^2} \\
w &= (|AB| \bmod T\text{-length})/2 \\
l &= (|BC| \bmod T\text{-length})/2 \\
\alpha &= \beta - \gamma \\
\beta &= \arctan(l/w) \\
\gamma &= \arctan(A_y/A_x)
\end{aligned}$$

In a similar way, we can get nodes and edges of the grid from the start node S through loop. Equation 3 gives the coordinates of the point $P_{i,j}$ of the i -th row and the j -th column of the grid.

$$\begin{cases} P_{x,i,j} = S_x + distance_{ij} \cdot \cos \alpha_{ij} \\ P_{y,i,j} = S_y + distance_{ij} \cdot \sin \alpha_{ij} \end{cases} \quad (3)$$

$$\begin{aligned}
\text{where } distance_{ij} &= \sqrt{(i \cdot T - length)^2 + (j \cdot T - length)^2} \\
\alpha_{ij} &= \beta - \gamma_{ij} \\
\gamma_{ij} &= \arctan((i \cdot T - length)/(j \cdot T - length))
\end{aligned}$$

The generated grid needs to be anti-mapped into the original polygon. The grid generated by the non-mapping function method often retains the grid of the obstacle area as a

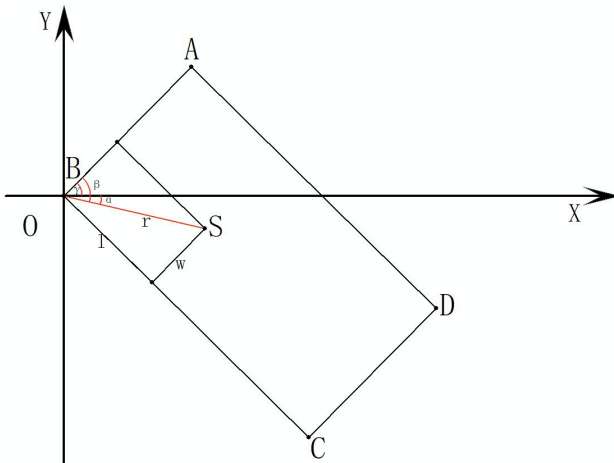


Figure 10. The starting point of the grid

non-navigable grid. Map matching does not need the information of non-navigable grid, hence the process of reflection is to delete the grid that is not in the unit and intersects with the unit boundary. Finally, the door is connected with the nearest grid node by the method of connecting the door mentioned above to ensure the connectivity of the grid (Figure 11).

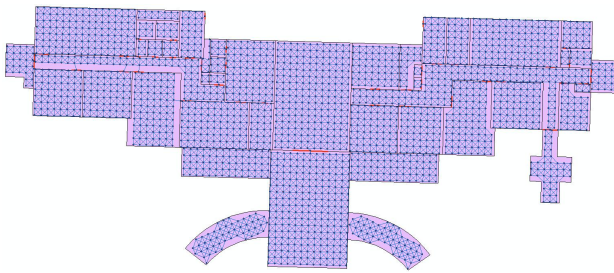


Figure 11. The grid of indoor units

5. EXPERIMENTS AND DISCUSSIONS

The algorithm consists of three parts, so our experiment also includes three parts: the accuracy of judgment of unit type, the representation error of approximate straight medial axis, and the representation of the grid. Three lab buildings were selected as the experimental sites for the above experiment, Figure 11 shows building1, Figure 9 shows building2, and Figure 5 shows building3.

According to the definition of narrow space and open space, the type of unit space is the subjective feeling that people derive from proxemics. Although Hall gives a quantitative representation of interpersonal distance, there are complex units including open space parts and narrow space parts in buildings, and it is difficult to define the type of these units. In the design of the building, the unit is divided for the use of its functions (such as meeting, dining and learning). In order to achieve a certain function, the unit needs to be designed as the corresponding type of space. Therefore, the experimenter judges the unit type according to the empirical principle of the relationship between the unit function and the unit type as the criterion of whether the algorithm is correct or not. Table 1 shows the correct rate of the unit type judgment of the three

Algorithm 3: Grid generation

Input: input indoor map with shapefile format

Output: output grid

```

1 foreach unit in the indoor map do
2   set  $mbr_i$  = the MBR of the unit;
3   set  $door_i$  = the collection of doors in the unit;
4   set  $widNumber$  = the number of grid nodes in the
   direction of width of  $minbound_i$ ;
5   set  $lenNumber$  = the number of grid nodes in the
   direction of length of  $minbound_i$ ;
6   get grid start point  $S$  based on  $minbound_i$  and  $T$ -length;
7   get grid nodes through the  $widNumber * lenNumber$ 
   times loop;
8   get the grid edges in the X direction through the
    $widNumber * (lenNumber - 1)$  times loop;
9   get the grid edges in the Y direction through the
    $(widNumber - 1) * lenNumber$  times loop;
10  get the grid edges in the diagonal direction through the
    $(widNumber - 1) * (lenNumber - 1)$  times loop;
11  foreach grid edges and grid points do
12    if in the unit then
13      edges of the grid  $\leftarrow$  grid edges;
14      nodes of the grid  $\leftarrow$  grid points;
15    end
16  foreach door in  $door_i$  do
17    connect this door to the grid;
18  end
19 end
20 end
21 nodes of the grid  $\leftarrow$  midpoints of door;
22 removes duplicate points from the grid's nodes;
23 return grid;

```

buildings. The experimental results show that this indicator has a higher accuracy rate for the unit type judgment.

Table 1. The accuracy of judgment of unit type

building	unit number	correct rate
building1	44	100%
building2	36	97%
building3	33	100%

The representation error of approximate straight medial axis is used to measure the degree of difference between it and the medial axis. The smaller the value, the closer the approximate straight medial axis is to the medial axis, and the more accurate the representation of ANM in the narrow unit is. The representation error of approximate straight medial axis has two aspects, that is, the mean value of the distance from the node on the approximate straight medial axis to the corresponding node on the medial axis, and the root mean square error (RMSE) of this distance. In order to balance the accuracy and efficiency of the evaluation, the approximate straight medial axis and the medial axis are discretized with a length of one tenth of the T -length. The formula for calculating the root mean square error is:

$$RMSE = \sqrt{\sum_{i=1}^n (d_i - \bar{d})^2 / n} \quad (4)$$

where n = number of discrete points in medial axis
 d_i = distance of the discrete point i of medial axis
 \bar{d} = mean of the distances of all discrete points on medial axis

Table 2 shows the mean and RMSE of the distance between the medial axis and the approximate straight medial axis. In the three experimental sites, the mean value of the distance is in the centimeter-level, RMSE is also very small, so the difference between the two medial axes is negligible for indoor pedestrian positioning.

Table 2. The representation error of approximate straight medial axis

building	mean	RMSE
building1	0.010879	0.000811
building2	0.044234	0.002216
building3	0.037955	0.003950

Pedestrians have greater freedom in the movement of open spaces and ANM is essentially a possible movement trajectory of pedestrians. Therefore, the grid should cover the target area as much as possible and have as many nodes and edges as possible in the covered area to fully represent the spatial topological relationship and geometric information. We use the usual non-mapping function method and the mapping function method used in this paper of the three buildings to generate the grid. The results were evaluated from three aspects: the ratio of the area covered by the grid to the building area, the number of grid edges and the number of grid points.

Table 3. The indicator of the grid generated by mapping method and non-mapping method

number	method	building1	building2	building3
grid edges	non-mapping	6082	4402	14682
	mapping	7318	5107	12442
grid nodes	non-mapping	1923	1428	3496
	mapping	2251	2374	4355
covered area ratio	non-mapping	0.719240	0.708697	0.811241
	mapping	0.824686	0.771406	0.812488

Table 3 shows the number of grid points, the number of grid edges and the ratio of the area covered by the grid generated by the mapping function method and the non-mapping function method. As can be seen, the mapping function method used in this paper is better than the grid generated by the non-mapping function method in the number of grid points and the coverage area in the three experimental sites. In the number of grid sides, in addition to building3, the results of other experimental sites have also improved. In general, the grid generated in this paper is better than the non-mapping function method.

6. CONCLUSION AND FUTURE WORK

In this paper, an algorithm for automatically generating ANM is proposed. We first propose an indicator that uses the buffer to determine the unit type according to the geometry. Experiments prove that this indicator has a high accuracy of judgment. Then, combined with the requirements of map matching on the medial axis, the approximate straight medila axis is used to replace the medial axis, which not only meets the requirements of map matching on the straight medila axis but also avoids

some problems of the straight medila axis algorithm. The experiment shows that the error between the medial axis and the approximate straight medila axis is within the acceptable range. In addition, according to the characteristics of indoor space has been divided into units at the time of construction, the grid is generated by using the mapping function method, and the generated grid is roughly evenly distributed within the unit, and the representation of the indoor unit is more accurate than that generated by the non-mapping function method. Our future work will put forward more comprehensive evaluation of the algorithm.

7. ACKNOWLEDGEMENTS

The study is supported by the National Key Research and Development Program of China (No. 2016YFB0502200)

REFERENCES

- Bernstein, David, Kornhauser, Alain et al., 1996. An introduction to map matching for personal navigation assistants.
- Blum, Harry, 1967. A Transformation for Extracting New Descriptors of Shape. *Models for the perception of speech and visual form*, 19, 362-380.
- Bowyer, A., 1981. Computing Dirichlet tessellations. *Computer Journal*, 24, 162-166.
- George, Paullouis, Seveno, Eric, 1994. The advancingfront mesh generation method revisited. *International Journal for Numerical Methods in Engineering*, 37, 3605-3619.
- Helbing, Dirk, Molnar, Peter, 1995. Social force model for pedestrian dynamics. *Physical review E*, 51, 4282.
- Hilsenbeck, Sebastian, Bobkov, Dmytro, Schroth, Georg, Huitl, Robert, Steinbach, Eckehard, 2014. Graph-based data fusion of pedometer and wifi measurements for mobile indoor positioning. *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing*, ACM, 147-158.
- Ho-Le, K., 1988. Finite element mesh generation methods: a review and classification. *Computer-Aided Design*, 20, 27-38.
- Honey, Stanley K, Zavoli, Walter B, Milnes, Kenneth A, Phillips, Alan C, White Jr, Marvin S, Loughmiller Jr, George E, 1989. Vehicle navigational system and method. US Patent 4,796,191.
- Jang, B. K., Chin, R. T., 1990. Analysis of thinning algorithms using mathematical morphology. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 12, 541-551.
- Latecki, Longin, Ma, C. Min, 1996. An Algorithm for a 3D Simplicity Test. *Computer Vision & Image Understanding*, 63, 388-393.
- Lee, D T, 1982. Medial axis transformation of a planar shape. *IEEE Trans Pattern Anal Mach Intell*, PAMI-4, 363-369.
- Lee, Jiyeong, 2004. A Spatial Access-Oriented Implementation of a 3-D GIS Topological Data Model for Urban Entities. *Geoinformatica*, 8, 237-264.

Legrain, Gregory, Allais, Raphael, Cartraud, Patrice, 2011. On the use of the extended finite element method with quadtree/octree meshes. *International Journal for Numerical Methods in Engineering*, 86, 717–743.

Obradovic, Dragan, Lenz, Henning, Schupfner, Markus, 2006. Fusion of map and sensor data in a modern car navigation system. *Journal of VLSI signal processing systems for signal, image and video technology*, 45, 111–122.

Pink, O., Hummel, B., 2008. A statistical approach to map matching using road network geometry, topology and vehicular motion constraints. *International IEEE Conference on Intelligent Transportation Systems*.

Pyo, Jong-Sun, Shin, Dong-Ho, Sung, Tae-Kyung, 2001. Development of a map matching method using the multiple hypothesis technique. *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No. 01TH8585)*, IEEE, 23–27.

Quddus, Mohammed A, Ochieng, Washington Y, Noland, Robert B, 2007. Current map-matching algorithms for transport applications: State-of-the art and future research directions. *Transportation research part c: Emerging technologies*, 15, 312–328.

Quddus, Mohammed A, Ochieng, Washington Yotto, Zhao, Lin, Noland, Robert B, 2003. A general map matching algorithm for transport telematics applications. *GPS solutions*, 7, 157–167.

Remy, E., Thiel, E., 2005. Exact medial axis with euclidean distance. *Image & Vision Computing*, 23, 167–175.

Roper, Kathy, Payant, Richard, 2014. *The facility management handbook*. Amacom.

Shashi Shekar, Hui Xiong, 2011. *Encyclopedia of GIS*. Springer.

Taneja, Saurabh, 2013. A Framework for Evaluation and Generation of Navigation Models for Map-Matching of Indoor Positioning Data. *Dissertations & Theses - Gradworks*, 7, 895–909.

Taneja, Saurabh, Akinci, Burcu, Jr, James H. Garrett, Soibelman, Lucio, 2016. Algorithms for automated generation of navigation models from building information models to support indoor map-matching. *Automation in Construction*, 61, 24–41.

Toussaint, Godfried T., 1983. Solving geometric problems with the rotating calipers. *IEEE Melecon*, 83.

Wang, Di, Shang, Jianga, Cheng, Wen, Li, Xiaolin, 2016. iminer: Sub-room-level poi interaction detection for semantic location history construction. *2016 Fourth International Conference on Ubiquitous Positioning, Indoor Navigation and Location Based Services (UPINLBS)*, IEEE, 182–191.