SHORTEST PATH ANALYSES IN RASTER MAPS FOR PEDESTRIAN NAVIGATION IN LOCATION BASED SYSTEMS

V. Walter, M. Kada, H. Chen

Institute for Photogrammetry, Stuttgart University, Geschwister-Scholl-Str. 24 D, D-70174 Stuttgart, Germany
Volker.Walter@ifp.uni-stuttgart.de

Commission IV, WG IV/6

ABSTRACT:

Navigation is one of the main applications in location based systems. In order to navigate to a destination, a shortest path analysis has to be calculated. Shortest path analyses are commonly based on vector maps and calculate the shortest path (or route) between two points in a network. Fundamental research has been done in shortest path analysis for vehicle navigation in street networks, but the algorithms for vehicle navigation are only of limited use for pedestrian navigation. In this paper, an approach based on raster maps is described that calculates shortest paths for pedestrians. It will be shown that the algorithm can be extended very easily in order to support various kinds of applications.

1. INTRODUCTION

Vehicle navigation systems have become a high market penetration. Especially mobile vehicle navigation systems, based on Personal Digital Assistants (PDA), are getting more and more popular. The typical data bases of these systems are vector data, which are acquired in the Geographic Data Files (GDF) format [ISO 2002]. The price of such systems, including PDA, Software, GPS and street data, starts in the range of about $300. Vehicle navigation on street network data is a robust process and the software has become very user-friendly. However, the navigation with such systems has only limited use for pedestrian navigation.

There are two main reasons that existing navigation systems are not suitable for pedestrian navigation. First, the movement behavior of pedestrians can only be inadequately described with vector maps, because unlike cars, pedestrians are not moving along the middle axis of street lanes, but typically use the shortest distance between their actual position and the next visible target of their path, especially when crossing places or in indoor areas. Secondly, vector data for large areas are only available for streets and not for pedestrian or indoor areas. On the other hand, there are many raster maps from inner city areas or from indoor areas of public buildings available. The internet is a huge resource for raster maps of all kinds of areas. In the following it will be shown how these raster maps can be used for pedestrian navigation. First, an overview of the algorithm is given. Then, the partial steps are described in detail and results on different examples are shown. Finally, it will be discussed how the algorithm can be extended to support various kinds of applications.

2. ALGORITHM

The shortest path analysis in raster maps is done in four steps (see Fig. 1). In a pre-processing step (1) the input map is binarised in such a way that areas in which the user can move are represented with '1' and barricades are represented with '0'. Then, the binary raster map is skeletonised with morphological operations and an undirected graph is calculated from the skeleton (2). In the following step, an A* algorithm [Winston 1984] calculates the shortest path in the skeleton from the start point to the end point (3). Finally, a final path is generated by using visibility calculations (4). The result is a 'natural' path that can be used especially for pedestrian navigation.
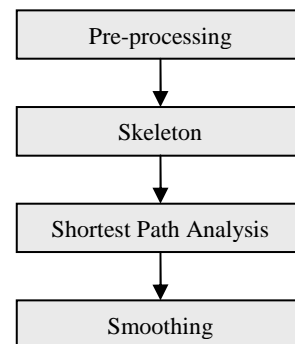

Figure 1. Workflow

3. STEP 1: PRE-PROCESSING

Even though this step consists only of a simple pixel operation, it is difficult to do this in an automatic way, because there is no common agreement on how different object classes have to be represented in raster maps. A human operator has first to select the pixel colour(s) which represent areas on which pedestrians can walk. These areas are filled with '1'-pixels and all other areas with '0'-pixels. Depending on the appearance of the map, it may be necessary to make some interactive editing. For example in indoor maps, walls and doors are very often represented in the same colour. Therefore it may be necessary to cut the doors interactively out in order to enable also the calculation of shortest paths between different rooms. Fig. 2 shows two examples of maps before and after the pre-processing.
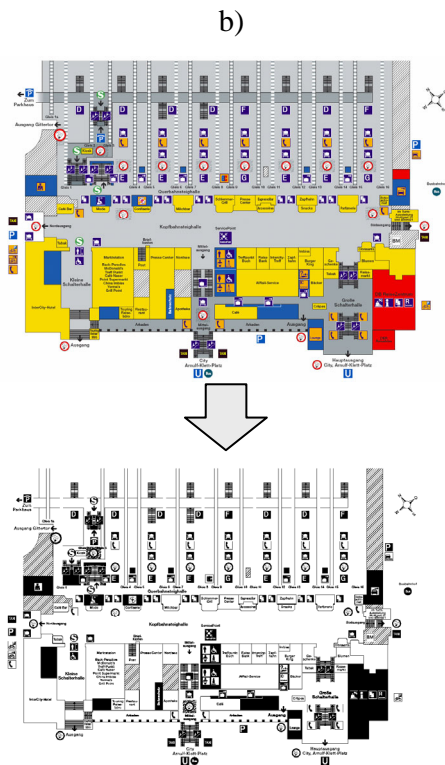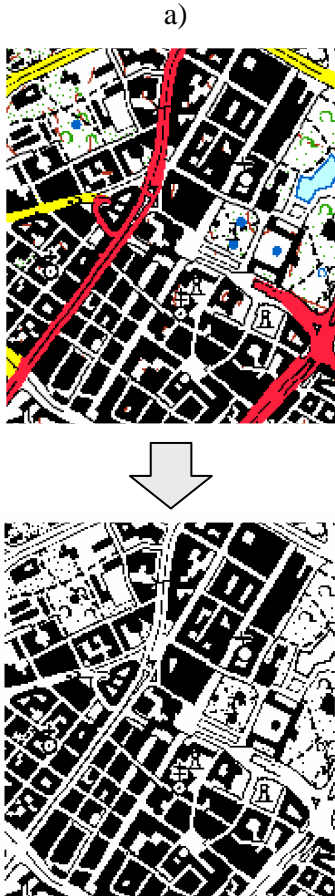
a)

b)

Figure 2. Pre-processing of (a) a topographical map from Stuttgart and (b) an indoor map of Stuttgart main station.

4. STEP 2: SKELETON CALCULATION

The skeleton is the basis for the further shortest path calculations and connects all streets and places. The start and end point of the path are first defined as skeleton pixels in order to guarantee that they are part of the final skeleton. Then, the skeleton is calculated iteratively with morphological operations that thin out the areas on which pedestrian can walk until a skeleton with one pixel width remains. Fig. 3 shows the calculation of the skeleton (green) and the start and end point (red square) of a shortest path query.

Figure 3. Calculation of the skeleton (in green colour) on a city map from Stuttgart. Start and end point: red square.

5. STEP 3: SHORTEST PATH ANALYSIS

The shortest path is calculated with an A* algorithm. The shortest path is that path that connects the start and end point with the skeleton and has the shortest length. Fig. 4 shows an example of a shortest path in blue colour.

Figure 4. Calculation of the shortest path (in blue colour). Start end point: red square

6. STEP 4: SMOOTHING

The shortest path in Figure 5 represents a connection between the start and end point, but it is not a path that a pedestrian will use, because the form of the skeleton is very irregular and follows the silhouettes of the buildings. Furthermore, pedestrians typically do not walk parallel to the streets (at least in pedestrian or indoor areas) but use the shortest way to the next branch of the path.

In order to calculate a shortest path that reflects this behavior, visibility calculations are used. Beginning from the starting point, it is successively tested for each pixel if it is possible to reach that pixel with a direct line without touching or intersecting a background pixel (for example buildings). If this is the case, this direct line is used for the final shortest path instead of the original shortest path. If the pixel cannot be reached with a direct line, a new starting point is set and the algorithm starts from the beginning. This is iteratively repeated until the end point is reached. The result is a smoother and more natural path from the start to the end point.

In order to avoid that the smoothed path touches a background pixel, a morphological operator is used before the smoothing. The operator enlarges all areas, which represent background, with a predefined size. This can lead to the situation that very small streets disappear. Therefore, after the enlargement, the result is combined with an OR operation with the skeleton. Fig. 5 shows the enlargement and the combination with the skeleton at an example.
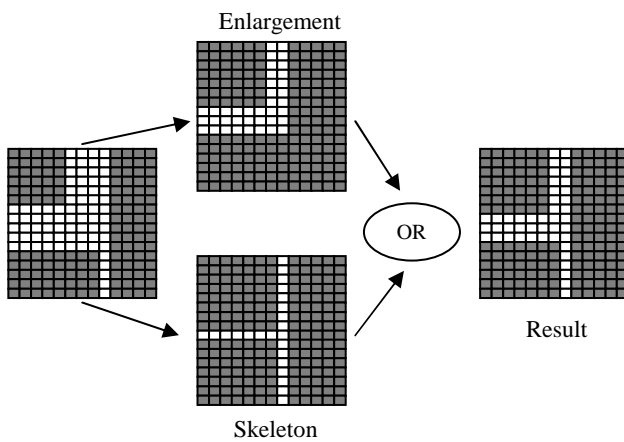

Figure 5. Building enlargement

After the smoothing, it is still possible that there will be still some non-natural shapes in the resulting shortest path. These irregularities are eliminated with a fine-smoothing. For each pair of connecting line elements of the smoothed shortest path, it is tested if it can be replaced with a smoother path that consists of three lines. This fine-smoothing is calculated iteratively until no changes occur in the final result. Fig. 6 shows the concept of the fine smoothing; Fig. 7 the result of the coarse smoothing at an example; Fig. 8 the result of the fine smoothing at an example.
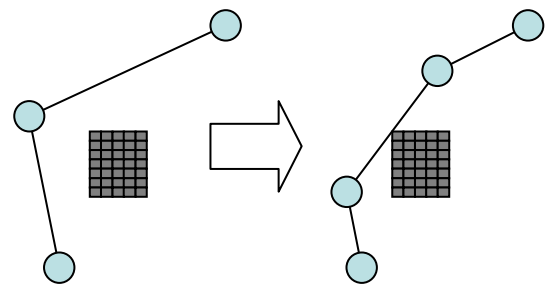

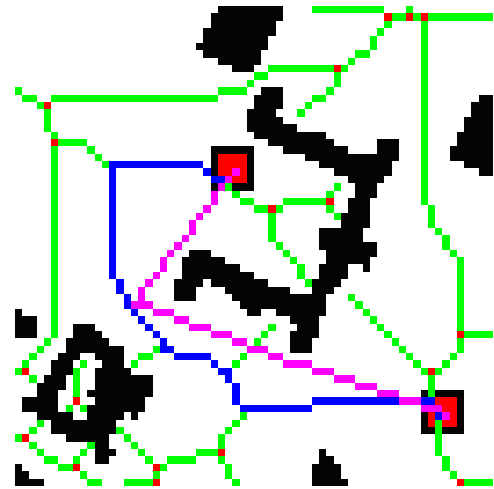Figure 6. Concept of fine smoothing
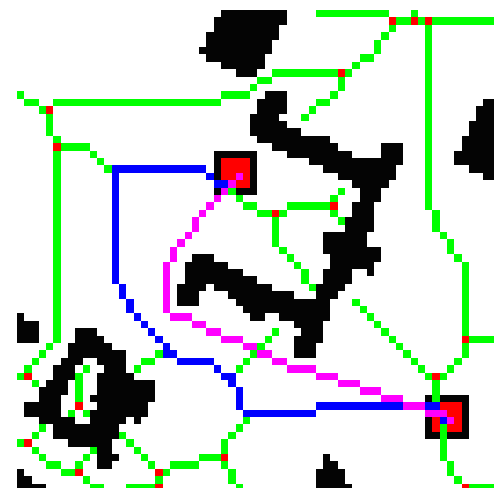

Figure 7. Coarse smoothing

Figure 8. Fine smoothing

7. RESULTS

Fig. 9 (a) shows a shortest path between a place and a street in a Stuttgart city map and Fig. 9 (b) shows a shortest path between the lockers and the entry to the subway in an indoor map of Stuttgart main station. The background pixels are represented with black colour and areas in which pedestrian can walk are represented with white colour. The start and end points are represented with red circles. The calculated skeleton of the white areas is represented with green lines. The shortest path which connects the start and end point by using the edges of the skeleton is represented with a thick blue line. The result of the whole process is represented in magenta colour.

8. DISCUSSION

We have tested the algorithm on a large number of datasets and got very satisfying results. The algorithm does not always find exactly the shortest path, because the shortest path calculation is done on the skeleton which has an irregular shape. Therefore in some special situations, the algorithm finds only the second or third best path – but always a path which has a length that does not differ very much from the shortest path.

The interactive map editing for the pre-processing is a very important factor. For example the indoor map in Fig. 9 (b) contains much text information that is classified as background. This can lead to unnatural results if this text information is along the searched path. However, the interactive editing can be done very effectively, because an operator has only to use a "digital rubber" to remove all unnecessary information from the map, which can be done in a very short time.

In this paper we examined only shortest paths regarding the length of the path. The algorithm can easily be extended if the weights of the graph of the skeleton are changed. For example in pedestrian zones or in areas with special interest for tourists, the graph can get lower weights in order to direct the user to more attractive places. Very different kinds of application can be supported with this approach.

A main important application area for shortest path searches for pedestrians is in the field of Location Based Systems (LBS). This approach will be integrated in the Nexus platform, which is as an open and distributed environment for location based applications [Fritsch and Volz 2003].

9. REFERENCES

ISO 2002. ISO/FDIS 14825(E), Intelligent Transport Systems - Geographic Data Files – Overall Data Specification (GDF4.0), ISO TC 204/WG3, 10/10/2002.

Fritsch, D. & Volz, S, 2003. Nexus- the mobile GIS environment. In: Peter Farkas (ed.): Proc. of the first Joint Workshop on mobile future and symposium on trends in communications, Bratislava, Slovakia, 26-28, pp. 1-5.

Winston, H. J., 1984. Artificial Intelligence. Addison-Wesley, London.

a)

background pixel

skeleton

shortest path in skeleton
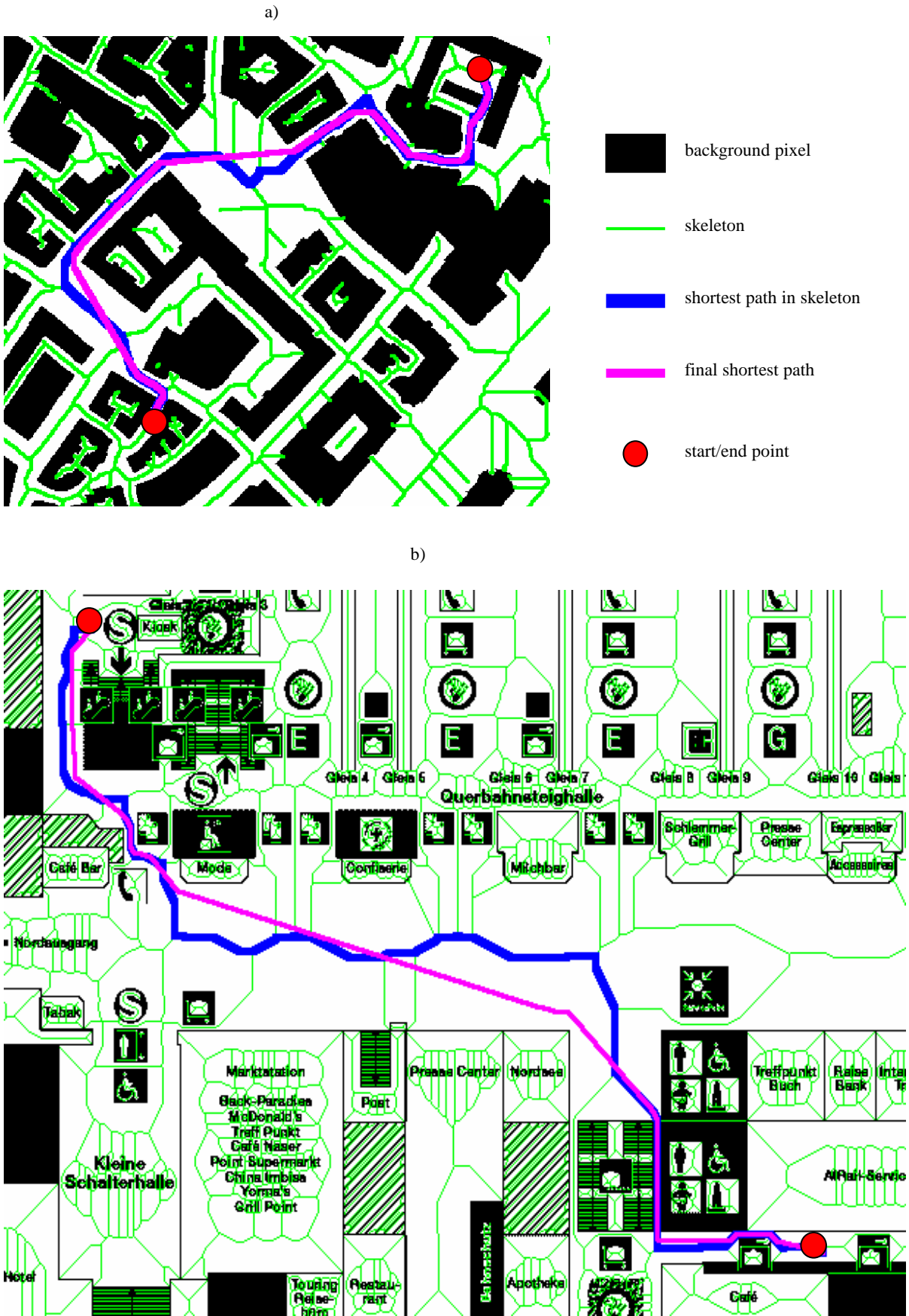
final shortest path

start/end point

b)

Figure 9: Shortest path examples (a) Stuttgart city map (b) Stuttgart main station