Incorporating Landmarks with Quality Measures in Routing Procedures

Birgit Elias and Monika Sester

ikg – Institute of Cartography and Geoinformatics, University of Hannover, 30167 Hannover, Germany

Abstract. In this paper we present an approach to providing landmark-based routes using a shortest-path algorithm. We start from the assumption, that at one junction there can be several landmarks to choose among, in order to find an optimal description of a route. The landmark selection used for describing the route is optimized taking the quality measures for the landmarks into account. Therefore, it is necessary to define quality measures. In the paper different types of quality measures are introduced and their integration in the route graph, as well as in a routing algorithm is presented. The usability of the approach is demonstrated using test data.

1 Introduction

The improvement of navigation systems and automatically generated routing description is an area of considerable research interest.

Most automated navigation systems rely on finding the shortest path in an underlying database network. But numerous cognitive studies have shown that human navigators use more than this single concept to select a path — criteria like least time, fewest turns and others play a vital role in human route planning.

Another finding of spatial cognition research is the importance of landmarks in the wayfinding process. Route descriptions are a sequence using two basic elements: instructions (actions at critical points along the route) and describing elements (characteristics and position of landmarks). Therefore, incorporating landmarks into automatically generated route instructions is essential to improving the usability of navigation systems.

It is well-known that humans try to minimize the complexity of route descriptions in order to reduce their cognitive load. For example, the capacity of short-term memory restricts it to a limited number of information pieces given in a route description. So, reducing the number of instructions is one possibility to decrease the description complexity. Furthermore, the linguistic complexity (e.g. number of words used) of each instruction or the cognitive complexity of the task described in the instruction has to be optimized.

In this paper we introduce an approach to optimizing the route selection considering the cognitive complexity of the route. For that purpose we incorporate point-like landmarks in the routing process. A graph using the geometric network and the landmark information is built up. At each junction in the road network, a variable number of landmarks can occur (0. . . n). The landmarks are introduced with quality measures to generate abstract route-specific route directions that take the dependency between each landmark and chosen path segment as well as interdependence between different landmarks into account.

The paper is structured as follows: First, a review of the relevant literature dealing with landmarks in directions and aspects of route optimization is given in Sect. 2. In Sect. 3 the idea of incorporating landmarks with quality measures in a shortest-path calculation is introduced. In Sect. 4 the data model and some results using a test data set are presented. The paper closes with discussion of results and outlook to future work in Sect. 5.

2 Related Work

2.1 Landmarks in Directions

Directions are routing instructions, mostly given in verbal form. They are generated to help someone finding his way who is unfamiliar with the environment. This process consists of three different cognitive stages [1]: first, the mental representation of this area is activated. Secondly, the optimal path considering different criteria like shortest path, fewest turns etc. is selected. Thirdly, the abstract conceptualization is transformed to verbal (or graphical) output. For conveying the wayfinding information via speech it is necessary to break the route down in single, sequential segments. These consist of describing elements (position and characteristics of landmarks) and moving actions (especially at critical points along the route) [2, 3].

There are two different approaches to provide local landmarks for route directions. In the first approach, a formal model of landmark saliency grounding on the characterization of Sorrows and Hirtle [4] is specified: the measures 'visual', 'semantic' and 'structural' attraction for building objects are established and assessed via hypothesis testing [5]. As an additional factor the advance visibility of objects while approaching a decision point is taken into account [6]. In a last step, the structural component is enhanced considering the *spatial chunking* of elements into 'higher order route description elements' (HORDE) [7, 8].

The second approach deals with the extraction of building landmarks from existing spatial databases. Therefore, the task is split into two subtasks: in a first step for each potential decision point salient objects are identified as *potential landmarks*. These are determined using data mining methods to detect the salient objects in the local neighborhood of a junction [9, 10]. Subsequently, the selection of potential landmarks has to be reduced according to the characteristics of a specific chosen route to the *route-specific landmarks*. For that step aspects of landmark visibility and perception, landmark quality, as well as cognitive and linguistic optimization of the communication process play an important role [11].

2.2 Aspects of Route Optimization

In automated navigation systems shortest-path calculations applying the Dijkstra algorithm [12] (or the A* algorithm) are used. Therefore, the road database is stored as a graph network with nodes and edges and so the shortest-path problem is translated to a graph-theoretical problem. The distances between the nodes are used as 'travel costs' and are applied as weights to the edges of the network. It is possible to substitute the distance with travel time or other weights and create further route solutions (e.g. fastest route, *simplest paths* [13], *clearest route* [14]). The use of additional node weights is possible without modifications of the routing algorithm. Handling costs of turns requires the storage of edge-edge relations in the graph. For that purpose, the concept of a line graph is introduced [15]. This graph represents all pairs of consecutive edges and allows individual weighting.

Studies in the field of spatial cognition indicate that different criteria are used for path selection in human wayfinding: in addition to shortest distance and least time, also fewest turns, most scenic or straightest route criteria (minimizing angular deviation) and others are taken into account [16–18]. For this reason different route optimization approaches are introduced recently. They differ in optimizing a given criteria for the route selection or for its description.

Simplest Paths. Very often not the shortest but the simplest route is needed to give wayfinding instructions to someone who is unfamiliar with the environment. Cognitive studies indicate that people choose the straightest possible routes as opposed to more meandering routes [18]. That behavior leads to complexity reduction of the environment and ease the requirements of human short-term memory capacity: following the findings of Miller [19] people find it easy to remember (short-term) up to seven items (give or take two). So, reducing the number of turn elements in a route minimizes the needed memory capacity and therefore the complexity of the route.

Duckham and Kulik [13] propose an algorithm that can be used to select routes that minimize the complexity of instructions. This idea to 'ease the description' was first introduced from Mark [20] and assumes that the number and type of turns burden the route with a specific weight. While Mark [20] uses a weighting function to join metric distances with the instruction complexity, which are represented by a number counting the instruction elements needed, Duckham and Kulik [13] completely rely on the measure of instruction complexity. Nonetheless, the simplest paths are on average only 16% longer than the shortest paths.

Clearest Route. The Landmark Spider approach is similar to the simplest paths as far as it uses only specific weights and no geometric distance information in the shortest-path calculation [14]. But these weights represent the relevance of landmarks at each node with respect to the traveler's movement. Therefore, the combination of the salience of a landmark, its distance to the node, and the orientation of the landmark with respect to the traveler's heading build up the cost function. So, this approach uses a subset of all available landmarks, which are most prominent and easy to find, and determines the *clearest route* in terms of spatial reference.

Until now there has been no testing with real data to assess the performance of this approach. It is expected to be identical to the shortest path solution in the best-case scenario, but a low landmark density will lead to problems [14].

Context-specific Route Directions. This approach provides a formal model that focus on the optimization of the route directions taking the surrounding context into account [21, 22]. For a given route the minimal number of distinct parts in the abstract route directions on the highest granularity levels possible are calculated. Thereby one-one relations between decision point/action pairs and route instruction represent a low granularity. A high granularity stands for a many-one relation expressed by one route instruction covering more than one decision point of the route. Using this criterion supports the idea that the number of description elements in a route is connected to the complexity of the entire route directions.

These different granularity levels for the abstract route elements are produced by applying spatial chunking to them. This method groups several actions at decision points into one segment called 'higher order route directions elements' according to the three types 'numerical', 'landmark', and 'structural' chunking [8]. This approach aims at producing the cognitive simplest routes, because the conceptualization process of the route is eased. It complements the simplest paths approach providing an optimal description for their routes.

3 'Landmark Route'

Here, we want to introduce an approach that optimizes the route selection according to landmarks. The approach simplifies the conception of landmarks in only dealing with point-like landmarks like buildings. From the network and the landmark information a graph is constructed which supports a different routing option besides the shortest-path solution: the chosen route takes landmark quality and distance information into account and also considers landmark chunking rules.

3.1 Modelling Quality Measures for Landmarks in a Graph

To optimize the landmark selection, quality measures for the given landmark set are needed. Therefore, it is necessary to distinguish between different quality measures for landmarks representing the quality of the landmark object itself, the interdependencies between incoming route segment and landmark, and the quality of the landmark with respect to a turn. In the end, the final landmark quality is a summarization of all single quality aspects.

If each landmark is considered as a point-like feature, then it is possible to assign them to a junction in a path network and a 'landmark graph' can be built up. The landmarks represent the nodes of the graph. Their location corresponds to the geographic position of the junctions of the road network, which they are assigned to. Because more than one landmark at one junction is allowed, the connection between each neighbored landmark pair, which has a corresponding connection in the road network, is represented by an edge. So, each node and edge from the road network is multiple modeled in the landmark graph depending on the number of potential landmarks given for each node (see Fig. 1).

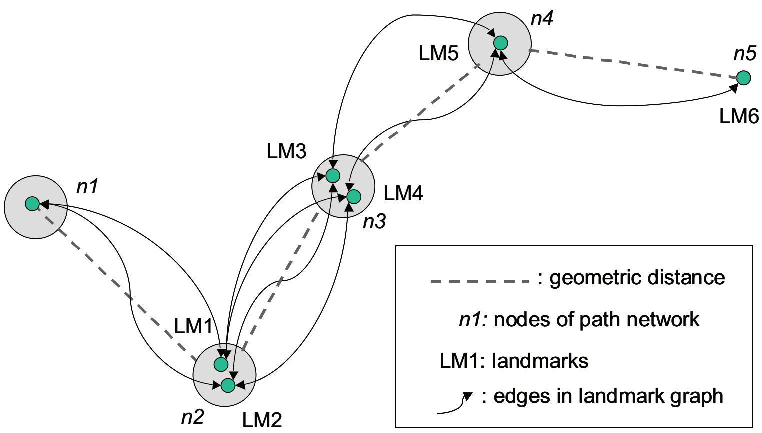

Fig. 1. Different landmark quality measures

Quality measures for landmarks can be assigned to different elements of the graph according to their impact on node-, edge-, edge-edge- or chunked edge-weights (see Fig. 2):

- ▷ If a quality criterion is only affecting the landmark object itself, the resulting 'landmark object quality weight' is assigned to the corresponding node in the graph (see Fig. 2, top left: weights for *n1*, *n2* and *n3*).
- ▷ If the landmark quality criterion is depending on the incoming route segment, it is assigned as 'landmark quality segment weight' to the directed edge. This stands for the dependency between the landmark quality and the direction the landmark is approached. (See Fig. 2, top right: the weight assigned to edge *e12* (representing the connection from node *n1* to *n2*) models the landmark quality of *n2* with respect to the edge *e12*).
- ▷ If the quality measure depends on the turn configuration, an edge-edge relation has to be modeled in the graph (for example using a line graph [15]). Onto these kind of edges the 'landmark turn quality weight' can be applied (see Fig. 2, down left: the weight applied to 'double edge' *de123* models the

cost of the turn from edge *e12* onto *e23* visiting landmark *n2*). This represents the fact, that the configuration of incoming and outgoing edge is relevant for the weight – this is relevant to model turning restrictions or preferences. However, for using this kind of specific edges in most navigation systems the routing is not performed on the node-edge-graph, but on the dual construct: the edges are modeled as nodes, turns then represent edges; all edge (and node) weights have to be merged into one combined weight and assigned to this new dual edge.

▷ If the quality measure depends on a concatenation of chunked elements, the 'landmark chunk quality weight' is applied to the chunk element (see Fig. 2, down right: Spatial chunking operations result in a HORDE modeled as edge *chunk13* and representing the connection from *n1* to *n3* via *n2*. Therefore, a completely new edge in the graph is created. Its geometric representation is the sequence of the chunked segments, its graph representation is the direct connection between *n1* and *n3*. The weight of the edge depends on the sum of weights of all chunked edges belonging to the chunk multiplied with a factor taken the improvement achieved by the chunking into account.

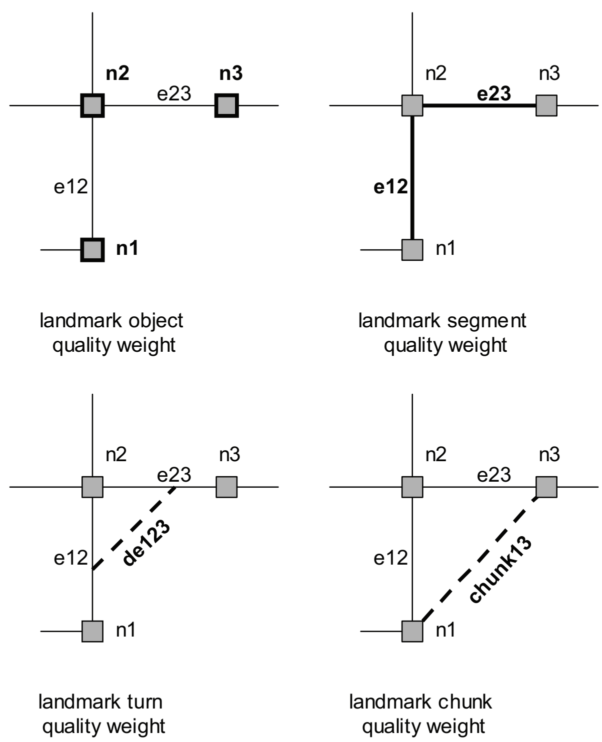

Fig. 2. Different Landmark Quality Measures

3.2 Establishing Weights for Landmark Quality

In Elias [10, 11] a procedure to select potential landmarks is described. In that approach buildings were used as potential landmarks. The selection process is based on a visibility analysis and a Minimum Description Length Principle by selecting objects as potential landmarks that can be described in a compact way. The result is a list of equal suited potential landmarks for a junction. After the selection, the question arises, which aspects have to be modeled as 'quality measures' for these objects to determine the best fitting object as a route-specific landmark in a particular navigation context. In this paper, the modeling of quality aspects is tailored to the characteristics of potential landmarks.

To determine the characteristics of 'good landmarks' for car navigation systems Burnett et al. [23] conducted a direction-given study. The reasons for using a specific landmark are analyzed and result in a list of factors:

1. **Permanence:** The likelihood of the landmark being present. We assume that here all buildings have the same value. (Probably this aspect gains more attention when it is necessary to determine the weight between different landmark object types, e.g. like buildings and parks.)
2. **Visibility:** The importance of this aspect for the quality of a landmark is approved generally. But besides the existing 'visibility' of objects, also the question of 'visual perception' arises. Cognitive tests reveal the human characteristic to focus his spatial attention according to his expectation [24]. That means, if we expect to make a turn right at the next junction, our visual focus rest on the right side of the street. Landmarks situated on the 'wrong' side will not be noticed.
3. **Usefulness of Location:** Describes whether the landmark is located close to navigational decision points or not. In the extraction process of landmarks this factor is modeled so far that only objects, which are located 'near' a potential decision point (junction), are investigated. But if this junction is not a decision point in the chosen route (means the path is following straight on), the impact of this factor is getting minor.
4. **Uniqueness:** Represents the likelihood of the landmark not being mistaken for other objects. One factor is the highly individual appearance, which, in our case, is already checked in the procedure to select the potential landmarks. The second aspect covers preventing confusion between the landmark and similar looking objects. The landmark must be identified easily and with no mistakes in the environment. Therefore, it is necessary to check whether there are objects in the 'neighborhood' of the route, which are misleading. This local area of attention consists not only of the immediate surrounding of the decision point. Also the incoming route segment has to be inspected, because a misleading object ahead of the correct landmark can lead to a wrong navigation decision [11].
5. **Brevity:** Stands for the conciseness of description associated with a landmark. It is related to the number of terms/words used to refer to the object.

To model these quality aspects in a routing graph it is necessary to link each quality aspect to a type of weight (see Fig. 2). In Table 1 the classification of quality weights is given. Here, we assume that the potential landmarks are determined following the approach of Elias [11], so some of the quality aspects are considered already in that pre-processing step. They are separately marked in the table. Applying the quality weights onto the landmark selection approach of [5] is also possible, but some of the aspects introduced here are already taken into account in the approach itself (e.g. visibility). Therefore, a modification of the weight modeling in respect to the used landmark selection procedure would be needed. Chunk weights are not addressed in this table, because they represent a composition of different quality factors, which are specified at the end of this chapter.

Table 1. Classification of landmark quality weights

Quality Factors	Potential Landmarks	Object Weight	Segment Weight	Turn Weight
1. Permanence	•			
2. Visibility	(•)		•	
2. Perception				•
3. Usefulness of Location	•			
3. Decision Point				•
4. Individual Appearance	•			
4. Uniqueness			•	
5. Brevity		•		

Constituting the different types of weights needs the merging of several aspects in one weight measure. Because no other elements of the graph are affected, considering the brevity of the landmark description builds up the **object weight**. To establish the **segment weight** it is necessary to merge the aspect of visibility and uniqueness into one weight. Both are dependent on the route segment traveled to reach the landmark: the object has to be visible and no confusable objects similar to the landmark itself must be on the route segment leading toward the landmark. For the routing it is necessary that also the distance of a segment is taken into account. Therefore, this segment weight has to be combined with the distance weight and then constitutes a new weight measure for the edge.

To determine the **turn weight** the influence of perception and location has to be combined. Both need an edge-edge relation to be modeled, as the measures depend on the turn instruction and the relative position of the landmark at the junction. Because the turn costs vary dependent on the approaching direction and turn instruction, the combination of incoming and outgoing edge is needed to store this quality weight.

The kind of spatial chunking applied to the data is the so called 'landmark chunking' [8]. It chunks all straight following route segments until a turn instruction is required and located by a landmark ('turn right at the church'). A chunk is stored as a new segment in the graph and its weight is handled as a segment weight accordingly. To define the **chunk weight** it needs a careful consideration of all the influencing factors and their validity for the chunk element. The following options are possible:

- ▷ the geometric distance is the sum of all single segments chunked together
- ▷ the only valid object weight is the measure of the last landmark in the chunk chain (represented by the last node)
- ▷ only the visibility of the end landmark is necessary; the value can be taken from the last segment in the chunk
- ▷ the uniqueness measure has to be determined again, this time taking the complete path of the chunk into account
- ▷ a reduction factor has to be applied on the segment measure representing the simplification achieved by chunking elements (for example depending on the number of elements chunked)

4 Incorporating Quality Weights into Test Graph

To check the usability of using the weights within a shortest-path search algorithm, a few tests with test data are conducted using the Dijkstra Algorithm.

4.1 Graph Modeling Using Time Penalty as Weights

As visualized in Fig. 1 at each junction new nodes have to be introduced that represent the landmarks, and edges between all nodes have to be established. In our scenario, we will describe a setup where different node weights will be modeled, as well as weights for edges and chunking. No tests with the modeling of turn weights are conducted, as they require the dual modeling, which has not been done so far. Therefore, the tests in this paper are restricted to object, segment and chunk weights.

For developing a common framework to assess the graph, seconds are chosen as reference dimension units for geometric distances and quality weights. For that purpose the geometric distances (given in meters) are transformed into seconds assuming an average pedestrian speed of 1,5 m/sec. As far as that, the shortest-path in the network is equal to the shortest-time solution of the Dijkstra analysis. The test network used here is shown in Fig. 3.

Integrating landmarks in the process can be regarded as an improvement of the navigation task. If the task is regarded as a process with a specific duration, the navigation process of a route described by 'good landmarks' has to be faster than without any landmark information. Therefore, all segments in the landmark graph without any landmark information at their end node have to be punished with a general time delay. This penalty shall represent the additional time that is needed by a pedestrian to orientate himself at the end of each route segment

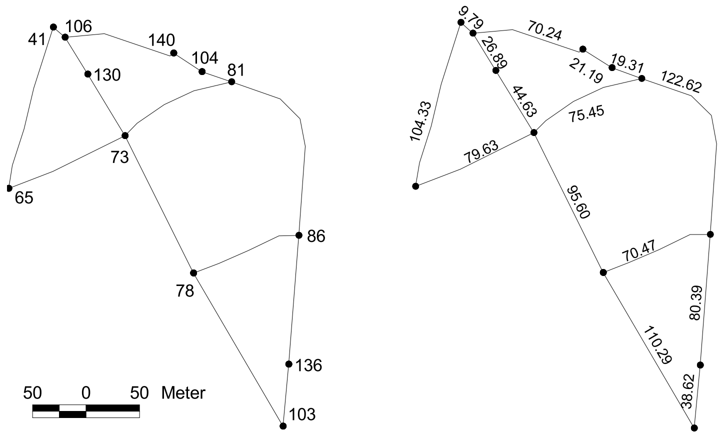

Fig. 3. Test data: nodes of the network (left) and distances (given in seconds) between the nodes (right)

(e.g. causing stops at each decision point). The same holds for the quality of the landmarks: good landmarks get no penalty, whereas bad ones or junctions with no landmark at all are punished with an extra time delay. Thus, additional landmark information leads to a simplification of the navigation task and the time delay decreases. It neutralizes if the landmark is 'good' in all quality aspects, because the navigation process can be completed without any stops or time delay for re-orientation purposes.

This approach is represented in the data model for the landmark graph as follows:

Distances: The metric distances are taken from the underlying path network. They are converted into seconds modelling an average walking speed of 1,5 m/sec of a pedestrian. The global time delay is applied as a penalty to all distances without any landmark information in their end node and here is proposed with 90 sec.

Object Weights: The nodes of the graph represent the potential local landmarks located near the decision points of the route. Because more than one landmark at a decision point is allowed, the nodes and their connections are multiple modeled. The landmark object weight is applied to the nodes of the landmark graph and causes a maximum penalty of 30 sec (that means, if the landmark is not very good, at most 30 sec are added as penalty on the landmark node).

Segment Weights: The edges of the landmark graph are directed edges representing the distances between the landmarks as well as the landmark segment weights. These weights reflect the impact of the landmark depending on its quality according to the segment influencing aspects. If the landmark is clearly visible from the route at an early stage, its value is 0 sec. The maximum penalty for a bad landmark weight is here given with 30 sec for each edge. The distance measures and the segments weights are added up.

Chunking: We assume that a route description that entails identical landmarks in a sequence can be memorized better than a route where the landmarks change from junction to junction. Chunking of instructions can be integrated in two different ways: either by introducing new edges in the graph that serve as 'shortcuts', or by inclusion in the routing algorithm (see Sect. 4.3).

The first option is to introduce new edges in the graph by connecting route segments that can be described by subsequent identical landmark objects. This inclusion can be automated in a kind of region growing approach, known from digital image processing (see e.g. [25]): Starting from a seed node, routes are followed and gathered, that have identical landmark types. After at least two nodes with the same type have been found, a new edge is included that now constitutes a kind of shortcut in the route graph. For this edge, an appropriate weight has to be determined as described above. After all nodes have been visited via all possible edges, the algorithm terminates, as all new shortcut edges are found.

In this paper, the idea of incorporating landmarks and their quality into the routing process is introduced. The quality weights are modeled and applied using a time delay for the traveler. There are no studies or research findings so far examining the actual impact of these aspects with respect to the navigation process. So as a start, we have chosen reasonable penalty values for each weighting type and assume that all types have the same maximum effect.

4.2 Examples for Integrating Object and Segment Weights

In this section an example showing the use of object and segment weights in the landmark graph is introduced (see Figure 4 and Table 2 accordingly). The optimal route between junction 65 and 86 is requested. Here, we investigate the route costs of two different options: the route can lead either from 65 via junction 73 and 78 toward junction 86 (called D1) or from 65 via junction 73 and 81 to 86 (called D2). The shortest-path calculation of both routes using the original Dijkstra algorithm reveals, that the distance of route D1 is shorter than of D2 (see Table 2).

Now, we introduce multiple landmarks at one node (at junction 81 landmarks 14 and 15, at junction 86 landmark 19 and 20) and apply landmark object weights to all nodes of the landmark graph. We assume that only 81_15 and 86_20 are 'good' landmarks in respect to object quality and therefore have no time penalty at all. All other nodes represent landmarks of 'bad' quality in respect to object quality and are punished with a time delay of 30 sec accordingly. Different route alternatives are calculated (called O1–O4) and presented in Table 2. The results show that only a combination of both weighted landmarks 81_15 and 86_20

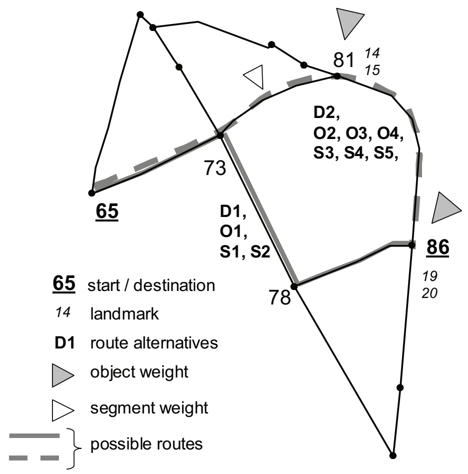

Fig. 4. Example using weights: route alternatives between junctions 65 and 86

accelerates the navigation task and leads to the shortest (or here better 'fastest')-path. All other combinations would still prefer the O1 (that is equal to the route D1).

In a next step segment weights are introduced additionally. They are applied comparable to the node weights: only the 'good' landmarks in respect to the segment aspects are weighted with 0 sec, all other landmarks get a penalty of 30 sec for their 'bad' quality in respect to their segment characteristics. In our case, only the edge from 73 leading to 81_15 receives a good segment weight. Different route alternatives (using different landmark combinations called S1–S5) are processed. The results show, that using the landmark node combination 81_15 and 86_20 leads to the fastest route. This route has to be preferred because it is the best landmark route available.

4.3 Integrating Chunking in Dijkstra Algorithm

Chunking can be included directly into the routing algorithm, in our case into the Dijkstra Algorithm. The idea is that route elements between adjacent nodes of a different type get a penalty as opposed to route elements that link two nodes of the same type. In the original Dijkstra Algorithm those nodes are expanded, that have currently been identified as being accessible on the shortest path from the start node. When the accessible successor nodes are added, not only the edge weights between them are used, but also a penalty value, if the adjacent nodes are of a different type. For the penalty, an appropriate value has to be chosen. In our case we used 60 sec, assuming that this additional time is needed to memorize different types.
In the following example (Fig. 5), the shortest route between junctions 73 and 140

Table 2. Result of Dijkstra Algorithm: Route between junctions 65 and 86 using object and segment weights

	nodes	shortest-path		using object weights				using object and segment w.				
start:	65	D1	D2	O1	O2	O3	O4	S1	S2	S3	S4	S5
	dist	79,63	79,63	79,63	79,63	79,63	79,63	79,63	79,63	79,63	79,63	79,63
73	node			30	30	30	30	30	30	30	30	30
	edge							30	30	30	30	30
	dist		75,45		75,45					75,45		
81_14	node				30					30		
	edge									30		
	dist		(75,45)			75,45	75,45				75,45	75,45
81_15	node					0	0				0	0
	edge										0	0
	dist	95,60		95,60				95,60	95,60			
78	node			30				30	30			
	edge							30	30			
	dist	70,47	122,62	70,47	122,62	122,62		70,47		122,62	122,62	
86_19	node			30	30	30		30		30	30	
	edge							30		30	30	
	dist	(70,47)	(122,62)				122,62		70,47			122,62
86_20	node						0		0			0
	edge								30			30
total [sec]:		**245,70**	277,70	335,7	367,7	337,7	**307,7**	425,7	395,70	457,70	397,70	**367,70**

is searched. At each of these junctions there are a varying number of landmarks of different type. E.g. at node 130, there are landmarks 30, 31, and 32 with landmark types 2, 4 and 1, respectively. The routing between junctions 73 to 140 is possible along all landmark nodes in the graph. Also, there will be four target points in node 140, as that node consists of four landmarks.

Applying the modified Dijkstra Algorithm to this route yields the results given in Table 3.

Table 3. Result of Dijkstra Algorithm: route between junctions 73 and 140

route nr	start (type)	dist.	via node (type)	dist.	via (type)	dist.	end (type)	penalty (from-to)	total distance
1	9(1)	75,45	15(1)	19,31	26(1)	21,19	37(6)	60 (26-37)	175,95
2	9(1)	75,45	15(1)	19,31	26(1)	21,19	38(1)	0	115,95
3	9(1)	44,63	31(4)	26,89	28(4)	70,24	39(4)	0	141,76
4	9(1)	75,45	15(1)	19,31	26(1)	21,19	40(3)	60 (26-40)	175,95

As shown in the table, the routing from junction 73 to junction 140 yields four different possibilities. Three of them use the junctions 81 and 104 with different selections of their landmarks. One is taking the north route via junctions 130 and 106. Route 2 is the shortest with 115,95 sec. It starts with landmark 9 and

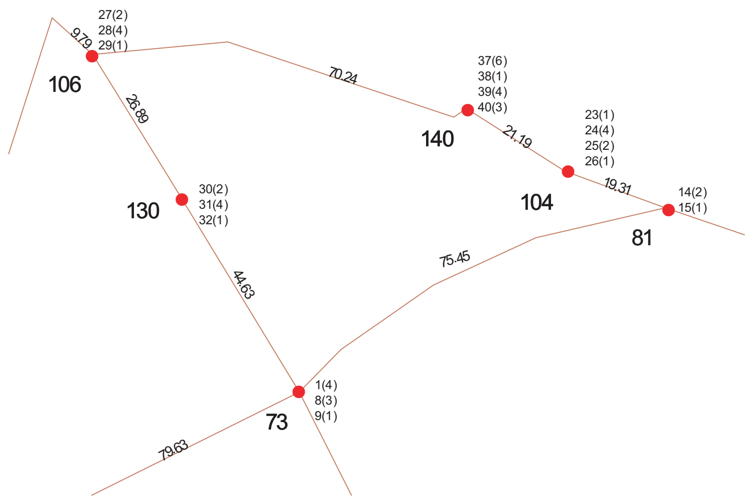

Fig. 5. Example for chunking in Dijkstra Algorithm: junctions with different landmark-nodes, including their type, in brackets.

uses landmarks 15, 26 and 38, that all have the same type (1). Routes 1 and 4 have different end nodes (37(6) and 40(3)) - thus they have to 'pay' for this type-change with 60 sec (see column penalty). Route 3 uses only landmarks of type 4 and thus no penalty is applied.

The best route among the four possibilities is route number 2, which uses only landmarks of type 1: LM1. The derived route description can be as follows: "Start from junction 73, turn left at LM1, go straight at the next LM1, until you reach the following LM1, which is your target."

In the example, chunking was used to join any subsequent nodes of same type. In reality this might not be useful. As stated in [8], chunking is typically used to aggregate descriptions of the same type along a straight line, until a turning instruction appears. Taking such aspects into account is also possible, however, then also the geometric properties of the graph have to be taken into account: only those adjacent nodes of same type that go straight will get no punishing value.

The proposed chunking operations are not verified if they are cognitively plausible. They are introduced to show the feasibility of incorporating the idea of chunking into the routing process by means of creating new edges and time penalty values. The actual chunking strategies has to be analyzed and fit into the proposed model.

5 Conclusion and Outlook

In the paper we presented an automatic approach that is able to generate an optimal route description using landmarks. From a set of potential landmarks assigned to single junctions, the optimal ones are selected that can be used to describe the best route. When transferring the problem to a graph structure, a crucial factor is the modeling of the weights needed for the determination of the optimal route. Here, we try to model all distances and weights by means of seconds, that represent the duration of the different aspects (like moving or re-orientation). We made some assumptions that also led to satisfying results. Here further work is needed to prove the correctness of the penalty concept for all eventualities and specify the time dimension for each weighting factor. Especially the penalty model for the chunking weight has to be adopt to the actual chunking strategy in wayfinding descriptions. Also further investigations and experiments are needed to assess the time delay of a pedestrian. Therefore, user tests have to be conducted to determine the impact of the different aspects in reality.

However, the presented method gives the technical framework for the calculation of shortest paths. It takes different factors into account, as it allows to modeling all the aspects in terms of weights in the graph. Moreover, an approach was shown including the introduction of possible chunks in the routing calculation process of the Dijkstra Algorithm.

References

1. Daniel, M.P., Denis, M.: Spatial descriptions as navigational aids: A cognitive analysis of route directions. Kognitionswissenschaft **7**(1) (1998) 45–52
2. Denis, M., Pazzaglia, F., Cornoldi, C., Bertolo, L.: Spatial discourse and navigation: An analysis of route directions in the city of venice. Applied Cognitive Psychology **13** (1999) 145–174
3. Tversky, B., Lee, P.: Pictorial and verbal tools for conveying routes. In Freksa, C., Mark, D., eds.: Spatial Information Theory: Cognitive and Computational Foundations of Geographic Information Science, Springer Verlag (1999) 51–64
4. Sorrows, M., Hirtle, S.: The nature of landmarks for real and electronic spaces. In Freksa, C., Mark, D., eds.: Spatial Information Theory: Cognitive and Computational Foundations of Geographic Information Science, Springer (1999) 37–50
5. Nothegger, C., Winter, S., Raubal, M.: Computation of the salience of features. Spatial Cognition and Computation **4**(2) (2004) 113–136
6. Winter, S.: Route adaptive selection of salient features. In Kuhn, W., Worboys, M., Timpf, S., eds.: Spatial Information Theory: Foundations of Geographic Information Science; International Conference, COSIT 2003. Volume 2825 of Lecture Notes in Computer Science., Springer (2003) 320–334
7. Klippel, A., Winter, S.: Structural salience of landmarks in route directions. In Cohn, A.G., Mark, D.M., eds.: COSIT 2005. Volume 3693 of Lecture Notes in Computer Science., Springer (2005) 347–362
8. Klippel, A., Tappe, H., Habel, C.: Pictorial representations of routes: Chunking route segments during comprehension. In Freksa, C., Brauer, W., Habel, C., Wender, K.F., eds.: Spatial Cognition III — Routes and Navigation, Human Memory and Learning, Spatial Representation and Spatial Learning. Lecture Notes in Artificial Intelligence, Springer; Berlin (2003) 11–33

9. Elias, B.: Extracting landmarks with data mining methods. In Kuhn, W., Worboys, M., Timpf, S., eds.: Spatial Information Theory: Foundations of Geographic Information Science; International Conference, COSIT 2003. Volume 2825 of Lecture Notes in Computer Science., Springer (2003) 398–412
10. Elias, B., Brenner, C.: Automatic generation and application of landmarks in navigation data sets. In Fisher, P., ed.: Developments in Spatial Data Handling, Springer (2004) 469–480
11. Elias, B.: Extraktion von Landmarken für die Navigation. PhD thesis, Universität Hannover (2006) (in press).
12. Dijkstra, E.W.: A note on two problems in connexion with graphs. In: Numerische Mathematik. Volume 1. Springer (1959) 269–271
13. Duckham, M., Kulik, L.: "'simplest"' paths: Automated route selection for navigation. In Kuhn, W., Worboys, M., Timpf, S., eds.: Spatial Information Theory: Foundations of Geographic Information Science; International Conference, COSIT 2003. Volume 2825 of Lecture Notes in Computer Science., Springer (2003) 169–185
14. Caduff, D., Timpf, S.: The landmark spider: Representing landmark knowledge for wayfinding tasks. In T. Barkowsky, C. Freksa, M.H., Lowe, R., eds.: Reasoning with Mental and External Diagrams: Computational Modeling and Spatial Assistance, AAAI Press, Stanford, CA, USA (2005) 30–35
15. Winter, S.: Modeling costs of turns in route planning. GeoInformatica **6**(4) (2002) 345–361
16. Golledge, R.G.: Path selection and route preference in human navigation: A progress report. In: Spatial Information Theory: A Theoretical Basis for GIS, International Conference COSIT '95, Semmering, Austria, September 21-23, 1995, Proceedings. Volume 988 of Lecture Notes in Computer Science., Springer (1995) 207–222
17. Golledge, R.D.: Human wayfinding and cognitive maps. In: Wayfinding Behavior. John Hopkins Press (1999) 5–45
18. Dalton, R.C.: The secret is to follow your nose: Route path selection and angularity. Environment and Behavior **35**(1) (2003) 107–131
19. Miller, G.: The magical number seven, plus or minus two: Some limits on our capacity for processing information. The Psychological Review **63** (1956) 81–97
20. Mark, D.M.: Automated route selection for navigation. IEEE Aerospace and Electronic Systems Magazine **1**(9) (1986) 2–5
21. Richter, K.F., Klippel, A.: A model for context-specific route directions. In Freksa, C., Knauff, M., Krieg-Brückner, B., eds.: Spatial Cognition IV. Reasoning, Action and Interaction: International Conference Spatial Cognition 2004. Volume 3343 of Lecture Notes in Computer Science., Springer (2004) 58–78
22. Richter, K.F., Klippel, A., Freksa, C.: Shortest, fastest, - but what next? a different approach to route directions. In Raubal, M., Sliwinski, A., Kuhn, W., eds.: Geoinformation und Mobilität - von der Forschung zur praktischen Anwendung. Beiträge zu den Münsteraner GI-Tagen 2004. IfGIprints, Institut für Geoinformatik; Münster (2004) 205–217
23. Burnett, G., Smith, D., May, A.: Supporting the navigation task: Characteristics of 'good' landmarks. In Hanson, M.A., ed.: Contempory Ergonomics 2001, Taylor and Francis (2001) 441–446
24. Maaß, W.: How spatial information connects visual perception and natural language generation in dynamic environments. In Frank, A., Kuhn, W., eds.: Spatial Information Theory, Springer Verlag (1995) 223–240
25. Gonzalez, R.C., Woods, R.E.: Digital Image Processing. 2. edn. Prentice Hall (2002)