# USING 3D URBAN MODELS FOR PEDESTRIAN NAVIGATION SUPPORT

**Jürgen Bogdahn and Volker Coors**

HFT Stuttgart - University of Applied Sciences
Schellingstaße 24, 70174 Stuttgart, Germany
{juergen.bogdahn, volker.coors}@hft-stuttgart.de
http://mona3d.hft-stuttgart.de

**KEY WORDS:** pedestrian navigation, 3D city models, facade textures, urban space

**ABSTRACT:**

Mobile navigation is usually focused on car navigation which implies a limitation on a street network. Due to this, route instructions in most systems are based on distances and directions given by the underlying line graph. This type of navigation is not an optimal solution for pedestrians, as they are not necessarily bound to streets, walkways or other polylines to walk on. This paper suggests a concept that subdivides urban space into 'walkable', 'semi-walkable' and 'non-walkable' space that can be used by pedestrians to a certain extent. Advantages as well as shortcomings of this concept will be outlined and discussed. Due to the 'walkable space' concept the procedures of way-finding and route instructions need to be adapted because of the absence of an underlying line graph. In the presented approach instructions are going to be based on landmarks along the route that can be used by pedestrians for way-finding. The use of landmarks leads to a further concept that will be outlined in this paper: procedural façade texturing for 3D urban models. As pedestrians navigate using landmarks, respectively the appearance of the façades as the most prominent parts, flexible and adjustable building textures are needed to support intuitive navigation and orientation. Whereas this is a task-driven scenario models do not need to be photo-realistic and façades can be used to provide a mixture of realism and abstraction (e.g navigation hints or thematic information) along the route. A basic concept for procedural texturing will be presented and a brief discussion on technical capabilities on mobile devices will be provided.

## 1 INTRODUCTION

Navigation systems nowadays can be regarded as standard for cars, motorcycles and just recently Google has introduced their map application as a mobile version, which can be used by pedestrians. Nevertheless, navigation for pedestrians is still under development and there are some issues to be taken into account when it comes to pedestrian navigation in urban space. Using a "car navigation approach" for pedestrian navigation can result in certain shortcomings that the authors are going to discuss and present concepts addressing these problems.

Navigation systems normally provide route instructions based on a street network, respectively a line graph, because vehicles generally navigate on streets. This concept results in route instructions based on this line graph. The concept regarding navigation in such a network is to tell the user when to change directions and into which direction to turn next. Current navigation systems normally provide this information by a distance to the next relevant node on the graph and the direction in which to turn at this point. The current orientation of the user does not even have to be considered, as the assumption is that the vehicle only moves on the line graph, either forward or backwards. Therefore the degree of freedom in terms of probable moving directions is very restricted and simplifies the navigation instructions that are required in order to find the way to the target point. However, pedestrians do not tend to 'walk the line'. Normally pedestrians would navigate in a different way and would not necessarily navigate on a line graph, because they are not bound to navigate on a street, in contrast to cars. Pedestrians can use open spaces, like squares, parks, etc., where they can walk freely. They also use manifold ways of orientation, therefore they are not restricted to street names and other "vehicle based" concepts, like distances and directions on a street network. Pedestrians navigate using the environment they are surrounded by: urban space. When asking someone for the directions to a specific place, people would tend to describe a route based on landmarks and other prominent aspects along the way and hardly use street names or distances. They provide "visual" route instructions that describe the way. Pedestrians actually have the "right speed" in order to use these visual instructions as they have the time to find the specific landmarks and make a decision. For car drivers the direction/distance instruction is just the right information in order to make the next decision, besides all the other tasks that are involved driving a car. Therefore distances and directions seem not to be appropriate for pedestrian route instructions as they can hardly be related to a line graph, whereas visual route instructions might better support them in terms of way finding. The concept of visual route instructions might be backed-up by a line graph based data set, but other concepts might be possible. In section 2 the authors will outline situations regarding open spaces where vector based data sets might be inappropriate and other concepts have to be found for pedestrian navigation.

In order to better support visual navigation for pedestrians it would be useful to provide a digital 3D model on the mobile device. There are some approaches of navigation system manufacturers to include specific 3D landmarks into their navigation systems. Nevertheless, the navigation instructions are still based on the street graph providing distances and directions. In order to support a landmark based, visually oriented instruction approach, it would be necessary to provide all buildings along a route, not only the landmarks. This would make it easier for the user to compare the real world to the virtual model, locate his own position, use the visual hints and to follow the route instructions. In order to provide 3D city models on a mobile device, in contrast to just place landmarks, needs a very efficient way of data management and transmission. The focus in this paper will be on textures for 3D buildings as this data set is usually on of the largest. Another issue about façade textures is their flexibility. Normally 'static' photo images are used in order to produce façade textures. In a scenario where users navigate according to visual hints and landmark based route instructions it might be sensible to include these hints into the prominent façade textures in order to support the user. Including these hints would make it necessary to change the façade texture according to the calculated route. This might be very complicated using image based textures. In this paper the authors will outline a concept for procedurally created façade textures that are flexible enough to be adapted to the navigation context and the requirements in this scenario.

The remaining sections of this paper will be organized in the following way: Section 2 will discuss shortcomings of line graph based navigation for the pedestrian use case focusing on open spaces and will outline a possible concept to overcome these problems. Section 3 will introduce the concept of procedural façade textures for a 3D-model-based navigation approach and discuss the benefits of the developed concept. Section 4 will describe the MoNa3D project investigating the use of 3D city models for pedestrian navigation and section 5 will conclude the paper and give a brief outlook on future work.

## 2 ROUTING FOR PEDESTRIAN NAVIGATION

### 2.1 Car-based Approach

When we talk about navigation systems at the moment, car navigation systems in particular, the navigation concept of these systems is mainly similar. Navigation takes place on a street network. This concept is quite reasonable and straight forward, as cars mainly use streets. They are bound to the line graph as they are supposed to stay on the street. Therefore the orientation of the car is given too, it can only move forward or backward on the line segment. This restricted degree of freedom in terms of navigation only requires a minimal set of instructions in order to follow a route through the network. It is sufficient to provide instructions into which direction to turn at the next node of the network and in which distance this node is going to appear (see example in fig 1).


Figure 1: Navigation system example for car navigation [1]

This approach also seems to be beneficial for the scenario of car navigation, as the driver would hardly be able to observe more information as he needs to concentrate on many other things that are incorporated into the process of driving a car. Although there are several concepts integrating 3D buildings into these systems, using landmarks, half-transparent building blocks as well as more detailed textured 3D city models for city centres, these 3D representations are not part of route instructions yet.

[1] from http://www.navigon.com/, copyright Navigon AG

This 'graph-based' approach is also used for pedestrian route descriptions at the moment (see figure 2). One problem that can be recognized using this graph based approach is that the vector data for pedestrian foot paths, bridges, underpasses and other urban 'elements' that can be used by pedestrians are not entirely modelled in the graph(compare (Schilling et al., 2008)). And for open spaces like squares, parks, parking areas it might not be feasible to model these elements by line graphs, because they cannot define all possible paths a pedestrian can take. In the example in figure 2 the system suggests a route to the north entrance of the central station around the parking area. This parking area could be easily crossed by a pedestrian approaching the entrance directly.

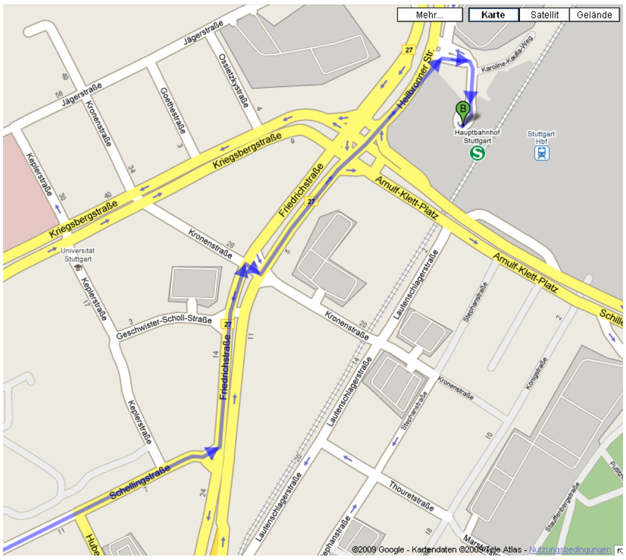



Figure 2: Pedestrian route from HFT Stuttgart to Stuttgart Main Station (from GoogleMaps)

Another issue would be that a local citizen would not cross the huge road intersection at the lower left corner of the parking area, as there are no pedestrian traffic lights (see figure 5). In contrast a local citizen would walk in south east direction and take another entrance of the main station building using a pedestrian crossing or underpass. Most of these aspects are not modelled in most of the current data sets which leads to route suggestions that pedestrian would hardly chose in a real scenario (e.g. main roads with no/limited sidewalks, no traffic lights, etc.). It might be complicated to model the missing data into the line graph and further investigations might have to be made in order to generate a data set that provides all the information relevant for pedestrian navigation in a suitable form.

### 2.2 Pedestrian Navigation Issues

When it comes to pedestrian navigation the concept of line graphs seems to be too restricted. One can find areas in urban space that are both accessible and comfortable for this user group in comparison to cars, e.g. squares, parks and other open spaces. But these open spaces represent urban elements that can hardly be covered by lines, because pedestrians not always 'walk the line'. As you can see in figure 3 an open space in GoogleMaps is approximated by certain vectors, but hardly any pedestrian would restrict himself to these options to cross the square if it is part of his route. This set of vectors is the attempt to cover the square with the 'car-based' approach, which is certainly not the optimal concept for pedestrian navigation.

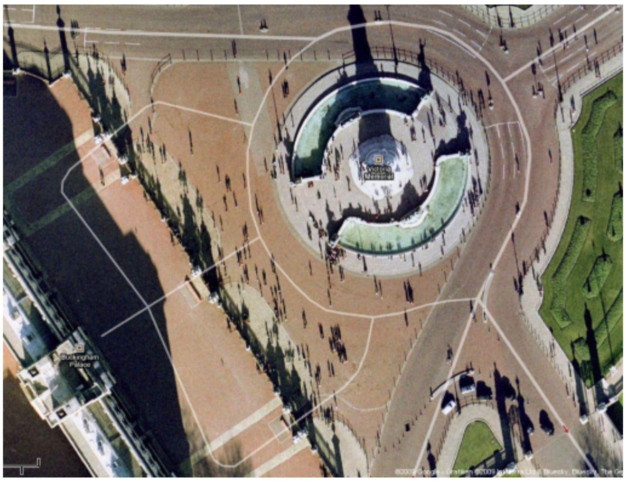


Figure 3: Street network example for open space in London (from GoogleMaps)

As the authors have tried to outline in the previous sections, a street network might not be the optimal basis for pedestrian navigation. And also route instructions based on distances, directions and street names (figure 4) might not be the optimum as pedestrians have a higher degree of freedom in terms of navigation compared to cars: they are not bound to the street network. Direction instructions like ”turn left” are also inaccurate as the orientation of the user is not given, as he does not necessarily align himself to a vector.

**Fußweg nach/zu Schellingstraße 24, 70174 Stuttgart**
**1,0 km** – ca. **13 Minuten**

(A) Hauptbahnhof Stuttgart

| | | |
|---|---|---|
| 1. | **Nordost** auf **Arnulf-Klett-Platz** Richtung **Karoline-Kaulla-Weg** | 59 m |
| 2. | **Links** halten bei **Kurt-Georg-Kiesinger-Platz** | 13 m |
| 3. | Nach **links** abbiegen, um auf **Kurt-Georg-Kiesinger-Platz** zu bleiben | 29 m |
| 4. | Bei **B27/Heilbronner Str. links** abbiegen<br>Weiter auf B27 | 0,3 km |
| 5. | Bei **Kronenstraße rechts** abbiegen | 19 m |
| 6. | Bei **B27/Friedrichstraße links** abbiegen | 0,2 km |
| 7. | Bei **Schellingstraße rechts** abbiegen<br>Das Ziel befindet sich rechts | 0,4 km |

(B) Schellingstraße 24
70174 Stuttgart

Figure 4: Pedestrian route instructions from HFT Stuttgart to Stuttgart Main Station (from GoogleMaps)

Having said that pedestrians do not navigate on lines, the authors would introduce the concept of people walking in zones. This is especially true for squares and other open spaces in the urban environment. These zones can basically be navigated freely taking into account certain restrictions. A basic classification into ”walkable”, ”semi-walkable” and ”non-walkable” space might be appropriate and could be refined in the future. In this approach it would be possible to subdivide open spaces into zones rather than approximating them by a path network. Investigations about navigable space in urban areas are also made by (Slingsby and Raper, 2007). Figure 5 shows two zones that would influence the route from the HFT Stuttgart to the main station. There is one ”non-walkable” zone at the road intersection. This zone would be avoided by pedestrians as there is a lot of traffic, no sidewalks and no pedestrian traffic lights. The second zone is ”semi-walkable” representing a parking area which can be crossed by pedestrians. This zone is defined as semi-walkable as for example a mother with a baby carriage or a person in a wheelchair would not preferably use it. Subdividing urban space into zones would also allow to overcome the split between indoor and outdoor navigation, as floor plans could also be subdivided into these zones. In this way the same data concept could be used for indoor and outdoor navigation. The need of connecting indoor and outdoor navigation is also addressed in (Becker et al., 2008)

Conceptually, the surface space is partitioned into a walkable space with obstacles of polygonal shape. Finding the shortest path here is similar to the well-known problem called ”Shortest path for a point robot” in computational geometry. The definition of the problem and an approach to solve it is given in (de Berg et al., 2000):”A point robot is moving among a set S of disjoint simple polygons in a plane. The polygons in S are called obstacles. Our goal is to compute a shortest collision-free path from $p_{start}$ to $p_{goal}$”. Calculating this solution has the complexity $O(n^2 \log n)$. This concept can be extended to buildings as even in buildings, people walk on a surface with obstacle polygons. These surfaces in 3D space are usually connected by stairs, escalators etc.

Figure 5: Zones of ”non-walkable” (red) and ”semi-wakable” (yellow) space for the example from HFT Stuttgart to Stuttgart Main Station (own depiction based on GoogleMaps)

Another possible approach for way-finding in this zone based environment might be to use channel finding algorithms like the one described in (Kallmann, 2005). The computation of the shortest path through the channel might not need to be conducted as the user might find his way through the channel based on visual hints using a 3D city model (see section 3). If required by the scenario the shortest path can still be computed using concepts like the Funnel algorithm (Chazelle, 1982)(Lee and Preparata, 1984). This path can be additionally displayed in the 3D scene for user support. In contrast to the car-based approach, the user does not need to follow this line in order to understand the route instructions, it is more a visual guidance for the shortest path. In an open space, the user might also decide to use a different path to the next landmark. For identifying appropriate landmarks close to the channel this shortest path could be used for spatial requests deriving the closest landmarks to the channel that can be used as navigation points/hints. ”Non-walkable” space in urban areas could be defined as obstacles, for example the building footprints, which can be acquired from the land registry data set. As depicted in figure 6 the whole city would be subdivided in appropriate zones. The required triangulation could be pre computed as a basis for the way finding process. The described algorithm in (Kallmann, 2005) might need to be extended in order to take into account different path costs according to the classification of ”semi-walkabke” and ”walkable” space. As this paper only outlines a possible concept, further investigations are required and will be part of future work.

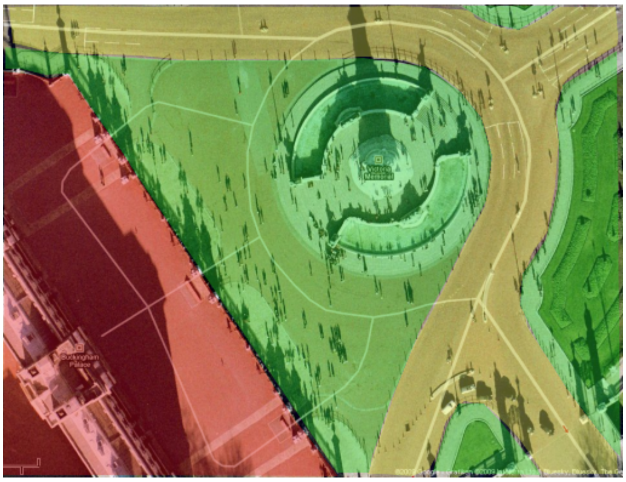


Figure 6: Zone definition for London examples (own depiction based on GoogleMaps)

As route instructions based on distances and directions would not work properly in this channel based routing scenario it will be necessary to find other ways of guiding the user through urban space. A landmark based approach (Coors et al., 2003) might be useful here and could guide the user appropriately using instructions that take the degree of freedom for pedestrians into account. By providing a 3D city model this landmark based approach can be realized in a very user friendly way, as the user can compare the real world to the 3D representation and orientate himself accordingly. This is basically possible, compared to the car scenario, because pedestrians have the time and the ’appropriate speed’ to use the 3D model as a navigation hint. Inaccuracy in positioning using GPS enabled smartphones for example can be compensated by the comparison of the virtual model with the real world. Nevertheless, limited resources on mobile devices make it necessary to provide light weight and efficient models. Limited screen size also makes some investigations necessary how visual navigation hints can be integrated into the 3D model. Highlighting of complete buildings using false colours might be a first step. However, in a dense urban environment and for pedestrians it could be necessary to provide more sophisticated visual hints, like highlighting the correct entrance to a big building or a building complex. In some scenarios it might also necessary to give exact visual hints in order not to confuse the user. With a textual instruction like ”pass the two university towers” it might be necessary to visually define if the landmark needs to be passed left, right or between the two buildings. This might be very important because passing the object on the wrong side would result in a position from which the user cannot see the next landmark, though the system would not recognize the slightly wrong position due to the inaccurate GPS position. Of course this could also be solved by enhancing the textual route instructions, nevertheless a visual hint should support the user in addition to the textual description. Other issues about the 3D model will be discussed in the next section, including considerations of how detailed or realistic the model needs to be.

## 3 PROCEDURAL TEXTURING APPROACH FOR PEDESTRIAN NAVIGATION

In order to give a visual aid for landmark-based navigation a 3D city model will be provided on the mobile user device. This 3D model should enable the user to orientate himself in his environment and find the specific landmarks that were identified by the system as suitable way points. The 3D city model in this scenario needs to be very efficient in terms of data model and data size, due to restricted system resources on the mobile device. Therefore, on the one hand a light-weight model needs to be provided which on the other hand can still provide a decent level of realism in order to support the user. In this paper the authors will focus on the façade textures as 1) textures consume a quite relevant amount of the overall data size and 2) building façades are the most prominent surfaces in urban space from a pedestrian perspective. Therefore this part of the data needs to be managed in an efficient way and needs to be flexible in terms of adding additional information. Integrating additional information into the façade texture might only be one approach of information visualization, but as the concept of landmark-based navigation introduces buildings as way points, the integration of information into the textures can be a feasible concept. This would also address the limited screen size and the restricted capabilities of integrating additional information by text boxes, billboards, etc.

The concept the authors are investigating is procedural creation of façade textures for 3D city models (compare (Coors, 2008)). In this concept not the original photo image is applied to the geometry in order to texture the building, but small tiles of the original texture. These tiles are arranged by a parameterized description in order to build the overall texture (figure 7). In that way the tiles can be arranged flexibly and according to user needs or context requirements. Using small pieces of the original texture taking into account repetitive and redundant elements of the façades this approach will also reduce the data size of the model.

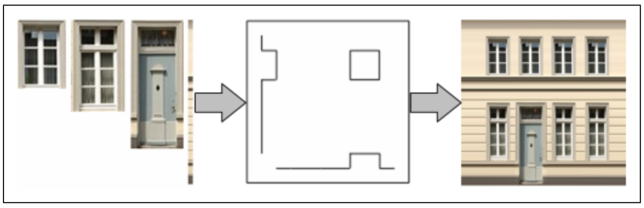


Figure 7: Procedural texture approach using tiles and a description in order to rebuild the overall façade texture

Basically the concept is built upon three components:

- **The programme:** this component implements the logic of the arrangement of the tiles in order to rebuild the façade texture. It reads the parameters of the description and arranges the tiles accordingly. The programme can support specific capabilities of the user’s device and hardware. For example, shaders could be used according to the effects the user wants to integrate into the visualization (e.g. thematic information, etc.).

- **The description:** the description is a set of parameters and definitions to describe for each façade how the tiles need to be arranged and if certain effects should be applied to specific tiles. The size for the tiles can be defined as well. In this way tiles can be scaled in order to fit into the overall façade reconstruction.

- **Texture tiles:** the tiles hold the actual texture information. The texture information is managed in small elements that are arranged in order to rebuild the façade. The texture information can be acquired from different sources, e.g. from a real world image but also managed in a texture library, which includes standard textures.

In the presented approach of procedural façade texturing the authors adapt the concept of (Parish and Müller, 2001) and extend it by further functionality in order to support different Levels of Realism (LoR) and the integration of additional information. These two aspects will be in focus as they are relevant for the navigation scenario outlined in the previous sections. The flexibility of the textures in terms of LoR seem to be useful as the 3D city model is not used for visualization only, where the aim is a maximum of realism. The navigation context is a task driven scenario where a specific LoR and an appropriate level of abstraction can be more beneficial for the user. The concept of adjusting realism to the task of the user is discussed in (Ferwerda, 2003).

In order to achieve this flexibility the procedural texture approach seem to be most appropriate. In the authors approach the aforementioned description is based on a 'pulse function' along the x- and y-axis. The pulses of both functions describe a zone in the texture area where they overlap and a predefined texture tile will be applied (see fig 8). These pulses can also be arranged in layers, where one layer includes one type of façade element. Therefore it is possible to decide which layers, respectively which content, to include into the façade. In that way it is possible to adjust the LoR according to the context and the user needs fulfilling the specific task.

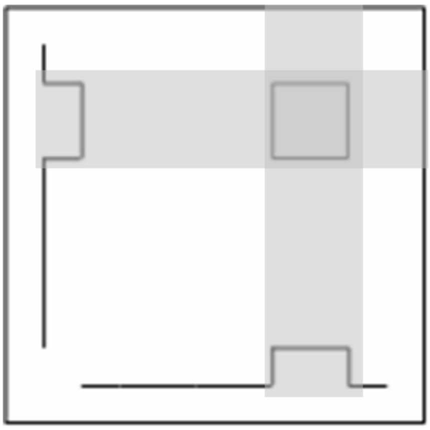
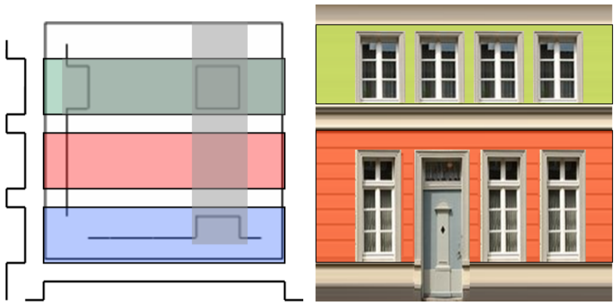


Figure 8: Pulse function concept for procedural texturing of façades

Another possibility of this approach would be to define additional layers with further information besides the 'real world' elements. In that way thematic data can be added into the façade texture. For example, if the use of the specific floors is given it would be possible to define pulses that are linked to a colour code. The pulses can represent the thematic data of the floors and it would be possible to generate a mixture of realistic appearance and abstracted thematic representation (see fig 9(a) and 9(b)).

For the navigation context these additional layers and pulses can also be used to give users visual hints in order to support them in navigating urban space. Using 3D city models the pedestrian navigation system would have to be a server-client application. Therefore it would be possible to generate the navigation hints on the server dynamically according to the computed route. By



(a) Example for thematic pulses (b) An abstracted façade representation

Figure 9: Including thematic information into the procedural façade

computing the shortest path in the determined channel (compare section 2.2 and (Kallmann, 2005)) it would be possible to detect the relevant landmarks. If specific landmarks are close to the route it would be possible to find out where the route passes them and because of the flexible texture concept a change in the appropriate façade textures would be possible (see fig 10).

Figure 10: An abstracted façade indicating a route instruction passing the landmark on the left side.

All in all the procedural concept seems promising in terms of reduction of data size which is a major issue when it comes to mobile applications. It also provides a flexible way of generating façade texture content according to the application scenario. This approach is quite innovative in terms of 3D city models as textures nowadays mostly consist of static image based textures. These textures only allow one fixed representation of the façade, which is sufficient for realistic visualizations. For task driven applications using 3D models as a part of the application concept and probably as a part of the user interface, a more flexible approach is required. Procedural textures appear to be a good concept towards a more sophisticated use of 3D models in task driven scenarios beyond pure visualization purposes.

## 4 MONA3D PROJECT - 3D CITY MODELS FOR PEDESTRIAN NAVIGATION

MoNa3D is the abbreviation for "Mobile Navigation in 3D" which focuses on pedestrian navigation with support by 3D city models. This project is conducted in cooperation between the HFT

Stuttgart, the University of Applied Sciences Mainz and several partners from the navigation systems and location based services industry. The aim of the project is the development of a navigation system that provides 3D city models on the mobile client in order to support pedestrians navigating urban space. As the authors have already outlined in the previous sections the concept of using 3D models for navigation is beneficial for pedestrians when using landmark-based route instructions. The MoNa3D project will investigate most of the aforementioned issues like procedural textures, detection and specification of landmarks as well as implementation of a prototype navigation system for future field tests. The concept of "walkable", "semi-walkable" and "non-walkable" space is not part of the MoNa3D project, although it is considered as part of future work. The computation of the route is still based on a street network. Nevertheless, route instructions based on landmarks and procedural texturing for the 3D buildings is part of the research conducted in MoNa3D. The following sections will describe the general architecture and concept of the MoNa3D project.

### 4.1 Client-Server Concept

The MoNa3D architecture is basically a client-server environment as it would not be feasible to store the 3D model and the street network on the mobile device. Therefore the system has got server-side components and the main part of the communication is based on Open Geospatial Consortium (OGC) standard interfaces. The server side system provides an OpenLS service (Open Geospatial Consortium, 2008), a Web 3D Service (Udo Quadt, 2005), a Catalogue Server and many more in order to fulfil the task included in the workflow. A 'mediator service' coordinates all the actions among the different services providing only one interface towards the client device. This prevents the client application to coordinate these actions and to cache intermediate results. Basically the 'mediator layer' is responsible to query the route for a given start- and endpoint from the OpenLS service and to provide the returned route to the Web 3D Service. The semantic route service will also identify the relevant landmarks and provide information about them (e.g. IDs). On the 3D server the appropriate 3D model will be loaded. As there is information about the landmarks available it should be possible to generate procedural textures according to the route geometry, which include navigation hints and additional information relevant for the navigation task (see section 3). The output of the Web 3D Service will then be transmitted to the 'mediator service' which will forward the 3D scene including the navigation instructions to the client. Optionally a compression of the data can take place in order to optimise transmission. The format of the 'mediator' response depends on the client. It would be possible to use standard formats like X3D (ISO/IEC FDIS 19775-1:2008, Extensible 3D (X3D), 2007) or proprietary formats and custom viewers. The rendering of the 3D model and procedural textures will be discussed in the next section.

### 4.2 Rendering on Mobile Platforms

The rendering of the procedural textures using hardware shaders is an open issue at the time of writing this paper. Currently there are no smartphones, PDAs or other mobile devices on the market that support shaders with their graphics hardware. The final goal would be to rearrange the texture tiles on the client device using the programmable rendering pipeline. This would completely exploit the benefits of using small tiles in terms of data size, because the final texture is built during the rendering process. Due to the absence of the appropriate hardware the texture is rebuilt by the client application and the complete texture is kept in memory for rendering. For clients that require a data format that does not

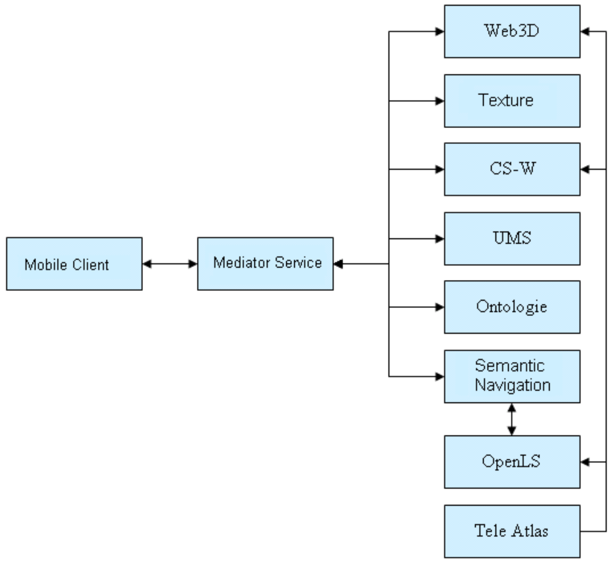



Figure 11: The MoNa3D System Architecture

support the transmission of shader code or cannot provide functionality of analysing the description for procedural textures, the textures could also be rebuilt on the server side, thus transmitting the complete texture to the client. This would also allow including navigation hints and additional information into the texture, as these are computed on server side. However, dynamic changes to textures in a context sensitive manner would only be possible on client side, otherwise too much traffic would be generated. A small advantages would remain even when the textures are rebuilt on the server. Although the complete textures would be transmitted the system would only have to store and load the texture tiles from the database. This results in less data that has to be stored in the database and the amount of data transmitted between the DBMS and the server application.

## 5 CONCLUSIONS AND FUTURE WORK

This paper has outlined a possible concept for pedestrian navigation that is not based on a street network like in today's navigation systems. The described concept is based on navigable zones that are classified according to their suitability for pedestrians. A more fine grained classification can be introduced according to the specific scenario. These zones can be navigated freely by the user, who is not restricted to walking on a street/path network, which is especially true for open spaces like squares, parks, etc. These open spaces can only be approximated when using a line graph and would probably not provide a convenient navigation experience for the user. Because of the fact that this concept does not work on a street network anymore route instructions based on distances and directions are also not appropriate anymore as pedestrian users would find it complicated to orientate themselves as they are not necessarily aligned to a line and find it hard to understand in which direction to turn. The authors have suggested to navigate the aforementioned zones according to landmarks and to support the user by a digital 3D city model. This model needs to be efficient and light-weight in order to be usable on a mobile device. Nevertheless, the model should also be flexible in terms of appearance and LoR in order to be adaptable to the user's scenario and the navigation scenario in general. The procedural texture approach outlined in section 3 seems to be promising in both terms, reducing data and providing the required flexibility. The landmark based navigation concept and the approach of procedural texturing on mobile devices are investigated by the MoNa3D project presented in section 4. At the moment the focus is on data management of the procedural texture on server side and the identification of suitable landmarks. First promising prototypes have been implemented and test were conducted, both in the specific domains. In the future work the main focus is to integrate the components to form the overall system and to start with first field tests as soon as the system prototype is working. Although these field tests can only investigate the feasibility of the system and the involved concepts as well as a general evaluation of acceptance of the identified landmarks for navigation. A detailed comparison between traditional navigation and the use of landmarks in 3D city models in terms of human perception and Human-Computer-Interaction needs more detailed research which is out of scope of this project at the moment.

## ACKNOWLEDGEMENTS

The authors would like to thank the German Ministry for Education and Research for sponsoring the MoNa3D project. We also would like to thank the City of Stuttgart and all industry partners for providing data and support for our test case scenario.

## REFERENCES

Becker, T., Nagel, C. and Kolbe, T., 2008. 3D Geo-Information Sciences. Springer, chapter A Multilayered Space-Event Model for Navigation in Indoor Spaces.

Chazelle, B., 1982. A theorem on polygon cutting with applications. In: Proceedings of the 23rd IEEE Symposium on Foundations of Computer Science, pp. 339–349.

Coors, V., 2008. Synthetic textures for 3d urban models in pedestrian navigation. In: B. et al. (ed.), Techniques and Applications for mobile Commerce, Proceedings of TAMoCo 2008, IOS Press, pp. 46–53.

Coors, V., Elting, C., Kray, C. and Laakso, K., 2003. Presenting route instructions on mobile de-vices - from textual directions to 3d visualization. In: M. Kraak, A. MacEachren and J. Dyke (eds), Geovisualization, Taylor & Francis.

de Berg, M., Cheong, O., Kreveld, M. V. and Overmars, M. (eds), 2000. Computational Geometry - Algorithms and Applications. 2. edn, Springer, chapter 15, pp. 308–310.

Ferwerda, J. A., 2003. Three varieties of realism in computer graphics. In: Proceedings SPIE Human Vision and Electronic Imaging 2003, pp. 290–297.

ISO/IEC FDIS 19775-1:2008, Extensible 3D (X3D), 2007.

Kallmann, M., 2005. Path planning in triangulations. In: Proceedings of the Workshop on Reasoning, Representation, and Learning in Computer Games, Edinburgh, Scotland, pp. 49–54.

Lee, D. T. and Preparata, F. P., 1984. Euclidean shortest paths in the presence of rectilinear barriers. Networks 14(3), pp. 393–410.

Open Geospatial Consortium, 2008. Location services (openls).

Parish, Y. I. H. and Müller, P., 2001. Procedural modeling of cities. In: E. Fiume (ed.), Proceedings of ACM SIGGRAPH 2001, ACM Press, New York, NY, USA, pp. 301–308.

Schilling, A., Lanig, S., Neis, P. and Zipf, A., 2008. 3D Geo-Information Sciences. Springer, chapter Integrating Terrain Surface and Street Network for 3D Routing.

Slingsby, A. and Raper, J., 2007. Advances in 3D Geoinformation Systems. Springer, chapter Navigable Space in 3D City Models for Pedestrians, pp. 49–64.

Udo Quadt, T. H. K., 2005. Web 3d service discussion paper. OGC Website.