Hochschule für Technik Stuttgart

MOBILE PEDESTRIAN NAVIGATION USING 3D CITY MODELS AND PROCEDURAL FAÇADE TEXTURES

Jürgen Bogdahn, Volker Coors

Abstract: Mobile navigation is usually focused on car navigation and works mainly on a street network represented by a line graph. Due to this, route instructions in most systems are based on distances and directions referring to the underlying line graph, which is appropriate for car navigation systems. Pedestrians, as they are not necessarily bound to streets, walkways or other 'polylines', might need another set of instructions for suitable navigation and way-finding. One concept is the integration of landmarks into route instructions, which are visualized using a digital 3D urban model. The use of landmarks leads to a further concept that will be outlined in this paper: procedural façade texturing for 3D urban models. As façades are the most prominent parts of a building from a pedestrian perspective, flexible and adjustable building textures are needed to support intuitive navigation and orientation. As the 3D model in this case is not only used for photo realistic visualization purposes but in a task-driven scenario, the visualization needs to be adapted and abstracted in order to support the user in solving the navigation task. A basic concept for procedural texturing will be presented and a brief discussion about the scenario specific use will be provided.

Keywords: 3D city model, urban environment, pedestrian navigation, procedural textures

// MOBILE FUSSGÄNGERNAVIGATION MIT 3D-STADTMODELLEN UND PROZEDURALE FASSADENTEXTUREN

// Zusammenfassung: Das Thema mobile Navigation bezieht sich heutzutage hauptsächlich auf den Bereich der Fahrzeugnavigation und arbeitet meist im Bezug auf einen Liniengraphen, der das Straßennetz repräsentiert. Aufgrund dessen basieren in fast allen Systemen die Fahranweisungen auf Entfernungen und Richtungen bezogen auf den zugrunde liegenden Graphen, was für Fahrzeugsysteme durchaus ausreichend ist. Da sich Fußgänger allerdings nicht zwingend auf Straßen, Fußwegen oder anderen „Kanten" in einem Graphen bewegen müssen, brauchen diese besser geeignete Anweisungen um entsprechend gut navigieren zu können. Ein Lösungsansatz liegt hier in der Verwendung von Landmarken, um sich im urbanen Raum zurechtzufinden. Die Verwendung von Landmarken führt hierbei zu einem weiteren Konzept, das in diesem Artikel vorgestellt werden soll: Prozedurale Fassadentexturen für 3D-Stadtmodelle. Da Fassaden aus der Fußgängerperspektive die auffälligsten Flächen darstellen, sind flexible und anpassungsfähige Texturen zur intuitiven Orientierung und Navigation notwendig. Da in diesem Fall das 3D-Modell nicht nur als photorealistische Darstellung verwendet wird, sondern in einem aufgabenorientierten Szenario, muss die Visualisierung abstrahiert und an das Szenario angepasst werden, um den Anwender bei seiner konkreten Aufgabe zu unterstützen. In diesem Artikel wird also das grundlegende Konzept zur prozeduralen Textur und kurz auch die anwendungsspezifische Verwendung beschrieben.

Schlüsselwörter: 3D-Stadtmodelle, Urbaner Raum, Fußgängernavigation, Prozedurale Texturen

Anschrift der Autoren

Dipl.-Ing.(FH) Jürgen Bogdahn
Hochschule für Technik Stuttgart
Schellingstr. 24
D-70174 Stuttgart
E: juergen.bogdahn@hft-stuttgart.de

Prof. Dr. Volker Coors
Hochschule für Technik Stuttgart
Schellingstr. 24
D-70174 Stuttgart
E: volker.coors@hft-stuttgart.de

1. INTRODUCTION

The majority of current commercial navigation systems have the focus on car based navigation. Systems for pedestrians are in development or part of research projects. The current navigation approach is basically divided into two phases: way-finding and user guidance. For car based systems the way-finding or routing is based on the line graph that represents the street network. Algorithms for finding the shortest path from a start point to a destination point in such a graph are well known. (Dijkstra, 1959; Hart, et al. 1968). Alternatives for pedestrian navigation, which not necessarily depend on a line graph, can be found in the research field of robotics and Artificial Intelligence (AI), for example (Kallmann, 2005). This article will discuss the questions about suitable route instructions and appropriate guidance for pedestrians.

Route instructions for car drivers use the underlying street network, essentially the derived line graph, and base their instructions and guidance on it. These systems assume that a car can only move on an edge of the network and is orientated along this edge. Therefore two simple pieces of information are necessary in order to find ones way through the graph to reach the destination node: the distance to the next node (decision point) and the action to take at that point (e.g. turn left or right). These instructions are quite sufficient when navigating the street network and are adequately adjusted to the very complex task of driving. The described information (distance and direction) can be visualized or communicated to the user in very different ways (Coors et. al, 2005), from plain text, to 3D maps. Latest developments in car navigation systems also make use of 3D city models or landmarks. Nevertheless, these models and landmarks are not necessarily reflected in the route instructions.

Pedestrians tend to navigate in a different way and using the car based approach might lead to an inconvenient way of navigation. Different types of pedestrian way-finding and guidance have been discussed in literature and propose that pedestrians have a very different concept to navigate urban space, as well as providing and understanding route instructions. Gaisbauer and Frank (2008) suggest that pedestrians navigate in decision scenes rather than decision points. This concept better reflects that pedestrians do not navigate on a line graph, although this graph can be part of route-finding, but that they move in open spaces. Therefore it is sensible to use existing algorithms on a line graph to actually find the way from start to end point but to provide navigation instructions, which are based on landmarks. Especially in open spaces like squares lines and edges in a graph can hardly cover the freedom that pedestrians have to navigate this open space. Landmark based instructions make more sense here, although route finding can still be based on the 'classical' algorithms, which can be found in car navigation as well.

Visualization of the *decision scenes* is also an important aspect when navigation instructions are based on landmarks. Completely photorealistic rendering of scenes in this scenario can be counter-productive when the provided 'landmarks' in decision areas are not significantly different to their surrounding. The difference between the buildings can also be based on attributes like usage, age, etc. and not necessarily on the visual difference, e.g. colour or height. Therefore non-photorealistic rendering, especially for the prominent façades of buildings, is an issue that needs to be addressed to give appropriate navigation instructions to pedestrians. However, the level of abstraction needs to be adjustable to maintain a certain degree of realism; otherwise the user would have problems to recognize the building in the real world. The visualization of landmarks needs to be achieved in fine grained steps between photo-realism and total abstraction (e.g. integration of navigation hints). This adjustability of the level of realism, especially for façades, needs to be addressed by pedestrian navigation systems especially when using landmark based instructions.

This article will discuss the use of landmarks for pedestrian route instructions and their appropriate visualization in section 2. Section 3 will describe the technical approach for a flexible and light-weight façade rendering. Section 4 will outline the architecture for the technical system using 3D city models and landmarks for pedestrian navigation and section 5 will conclude the paper.

2. LANDMARKS FOR PEDESTRIAN NAVIGATION

In order to find their way through the aforementioned open spaces pedestrians use landmarks to navigate and to describe routes (Raubal, Winter, 2002). These landmarks can be very different to landmarks in current 3D car navigation systems, as these mainly represent points of interest in a city wide context (*global landmarks*). Landmarks for pedestrian navigation can have a more local context and are defined by Raubal and Winter (2002) as *local landmarks*, as these landmarks can be prominent buildings in the current surrounding, which help the user to find the way. However, these objects would hardly be recognized as 'global' landmarks or POIs in terms of the whole cityscape. In this local scenario the identification of a landmark is much more difficult then for 'global' landmarks, as the saliency compared to other buildings needs to be determined according to specific characteristics (Ganitseva, Coors, 2010). Raubal and Winter (2002) defined a conceptual map that is based on 'visual attraction' of a building compared to nearby buildings along with their semantic and structural aspects. Schulz, et al. (2009) developed a tool for identifying local landmarks related to the concept of Raubal and Winter (2002) extending the conceptual map by navigation specific aspects (e.g. relative position to the street network). Elias and Sester (2002) use a buffer zone around the route geometry to extract objects with certain semantics and object types from an ALKIS dataset. However, after identifying the local landmarks for a calculated route, the question would be how to communicate this part of route instructions to the user in an efficient way. One method that is emerging at the moment is the use of Augmented Reality (AR), where additional information is superimposed over a video capturing the real world. This approach can be realized even on smartphones nowadays. Examples are 'Layar' (Layar, online 2010), Enkin, (Enkin, online 2010) and the 'nearest tube'-application of 'accrossair' (accrossair, online 2010). Nevertheless, the AR approach relies on very accurate positioning and orientation information. GPS and accelerometers are already built in most recent smartphones, however accuracy is normally not high enough to provide more sophisticated AR content. That might be the reason why the aforementioned projects mostly use tags or bubbles in order to provide information about objects/landmarks. Highlighting an entrance of a building by superimposing a frame around the door would be a very difficult task when using the information of built-in sensors in the latest smartphone generation.

Another method of presenting enriched information to pedestrians is the use of a digital 3D city model on mobile devices. This approach provides a complete virtual urban environment that can show the realistic representation of the real city, but also has the capability to show additional information or an abstracted representation of the surrounding that is adjusted to user's needs. One benefit over the AR method is the ability to navigate freely in the digital environment. This means the user cannot only get superimposed information from his perspective. The system can also allow free navigation in the scene or other pre-set perspectives to show certain aspects in a way that enhances comprehension (e.g. bird's eye view). Another advantage of digital 3D urban models is the utilization of the user as the most advanced sensor. By comparing the 3D representation on the mobile device with the close-range surrounding the user can orientate and position himself. Thereby he compensates the inaccuracy of GPS signals, for example, or the complete absence of orientation information (in case the device does not have the required sensors).

In order to support pedestrians when using a 3D city model for navigation purposes, a photo-realistic model would not be very helpful. If a situation in the real world is complicated in terms of a decision point during navigation a photo-realistic model would just depict this situation in the same way and would still make the decision complicated. Therefore the model needs to be enhanced by additional information and by an abstracted visualization that emphasises landmarks at decision points. These landmarks need to be part of the navigation instructions, of course. As Raubal and Winter (2002) identified building façades as one very prominent part of a landmark and contributing to the saliency significantly, it can be consequently concluded that façades would be used by pedestrians for orientation and navigation. This would support the hypothesis that integrating additional information and navigation hints into façade textures can intuitively support pedestrian navigation. It would also be possible to mix or adjust realistic and abstract representation with the intention to highlight certain buildings by changing the façade appearance and at the same time keep a factor of recognisability (Figure 1).

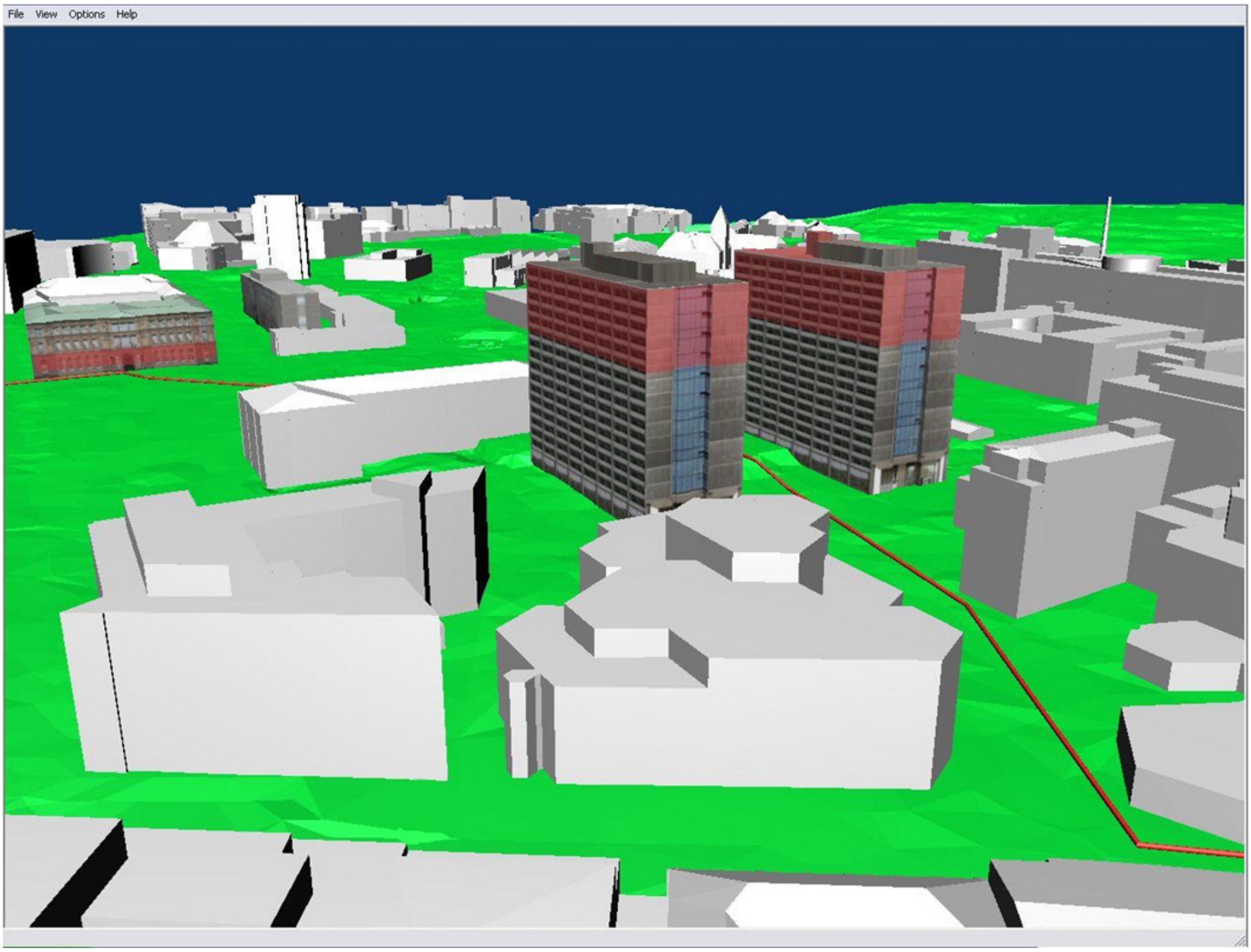

Figure 1: Landmarks with abstracted (highlighted) façade elements (mixture between real and abstracted representation).

Sorrows and Hirtle (1999) identify non-visual criteria (e.g. importance, meaning, use) that define landmarks, which can have a small visual saliency. In this case an abstraction of the 3D model is essential for the user as he cannot recognize this type of landmark by its visual appearance (e.g. 'turn left at the university building') (Figure 1). The next section will describe a technical approach to realize flexible façade textures in order to realize the necessary capabilities of the 3D city model.

3. SYNTHETIC TEXTURES

For mobile navigation it is necessary to provide a light-weight model due to the limited resources on the mobile device. Restrictions in bandwidth, memory and other hardware dependent capabilities are common and need to be taken into account. On the other hand, pedestrian navigation is a task-driven scenario, which demands flexible visualization to support the user rather than a photo-realistic representation. Other examples of research on non-photorealistic visualization supporting this theory can be found in Glandner&Döllner (2009) and Plesa and Cartwright (2008) and Plesa and Cartwright (2008). In order to take these requirements into account synthetic textures are used in the approach presented in this paper. For buildings that are no distinctive landmarks, in contrast to e.g. statues or fountains, we use a methodology for non-photorealistic texturing that can still provide an appropriate level of realism and the necessary flexibility with the purpose of including extra information like routing hints.

Currently most texturing in terms of 3D city models is done using photo images that are applied to the façade geometry. This approach tries to provide a maximum of realism to represent the real world scene in the most accurate way. This photo-realistic visualisation is not necessary in terms of pedestrian navigation, as the model only needs to provide enough similarity with the real world scene, so that the user can identify the relevant buildings and landmarks in his real world surrounding. This allows the use of tile based textures, which are generated by the application, ideally on-the-fly. Using standard tiles for windows, doors, etc. that are repeatedly used to rebuild the overall façade texture, results in a reduction of data size for the texture as repeated elements only need to be stored once. In the process of rebuilding the texture additional information can be included into the texture. This leads to abstraction, e.g. integration of arrows as navigation hints, which can support the user in identifying landmarks or making navigation decisions.

Synthetic textures are a way to make façade textures of 3D city models more flexible in terms of ratio between thematic information and photo-realistic appearance. Normally, in current 3D city models the emphasis and the purpose of textured buildings/façades are the photo-realistic representation of the real world building. In order to achieve this task most models apply a photo of the real world façade to the building geometry. However, the photo just represents the real world appearance of the building at the time the picture was taken. Besides the issues about updating the whole façade texture when a part of the real world façade changes (adverts, shop signs, etc.), the façade "content" is restricted to the content of the image. It can hardly be changed in order to adapt to scenario specific requirements.

The use of tile based procedural or synthetic textures can increase flexibility of the façade texture content and can be used to generate a more abstract representation of landmarks in the pedestrian navigation scenario. The authors' concept for procedural façade textures in this paper consists of following 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 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.

The description of the overall façade texture in the presented approach is based on functions that define pulses (Parish, Müller, 2001) in x- and y-directions (Figure 2). Where these pulses overlap they define an active zone where a specific tile will be applied.

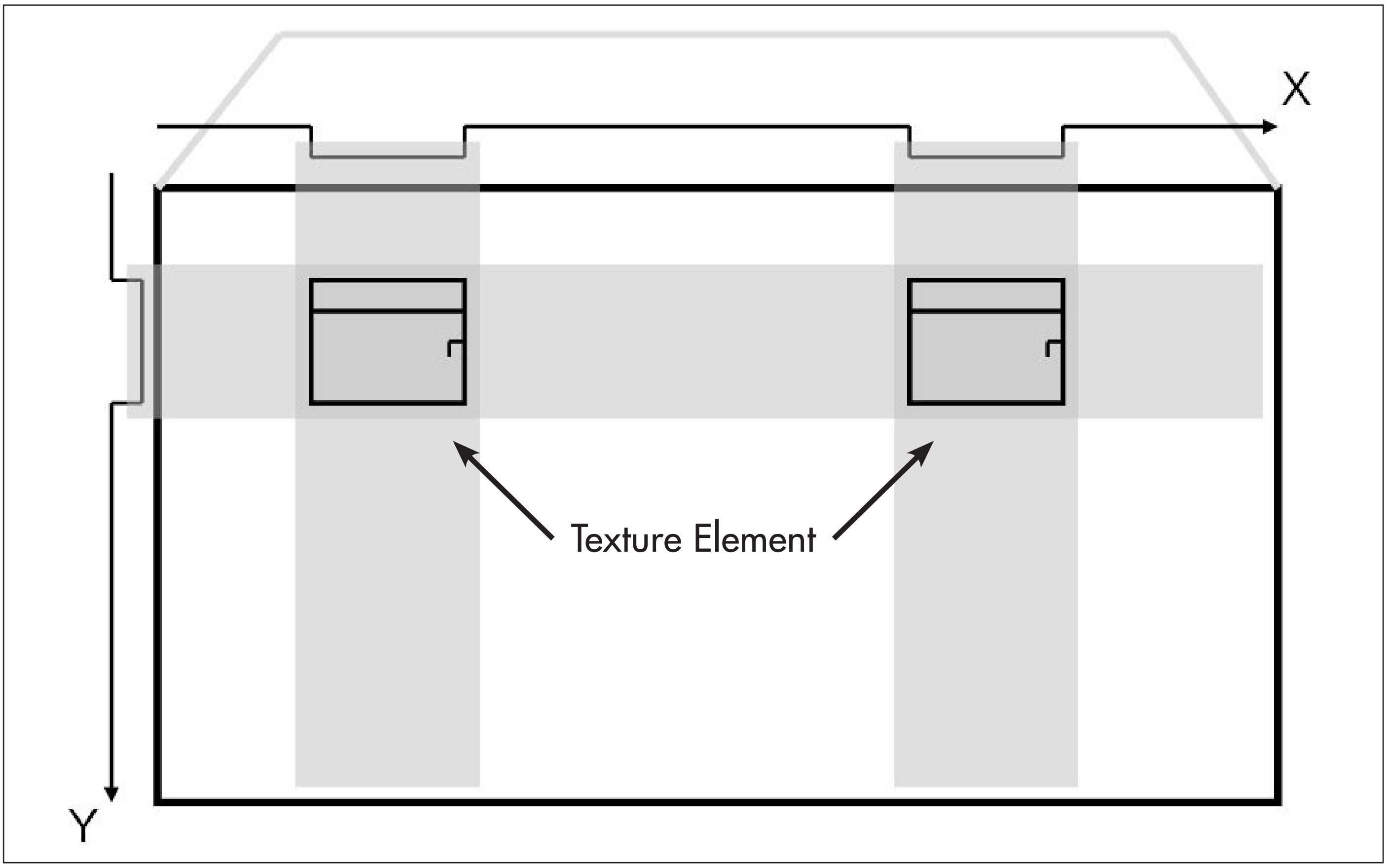

Figure 2: Pulse function defining active zones where façade elements are placed.

The tile can be scaled to the actual size of the active zone or applied in any other way, e.g. repeated in x-direction. It is also possible to arrange pulse functions in layers, comparable to a GIS or CAD system. Hence, it is possible to arrange tiles by their content. One can have layers for doors, windows, signs, etc and arrange them in a specific z-order. Therefore content can be behind or in front of other items in the façade. The layer-based concept can also be combined with a priority value, which determines if elements in the back are overlapped by the element in front or if the element in the back is removed completely. This concept prevents overlapping of doors and windows for example and helps the author of the façade description to define certain priority rules (Figure 3).

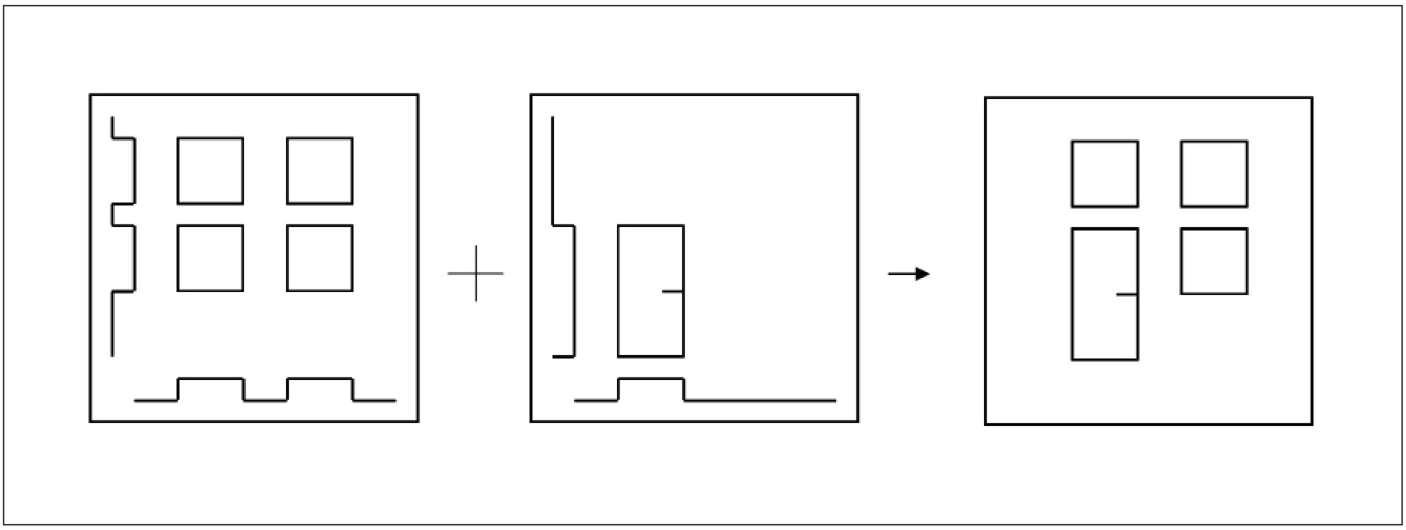

Figure 3: Door layer with higher priority over windows. Window intersecting door is removed.

Another aspect that makes procedural textures more flexible than static photo images is the possibility to control the level of realism (LoR) of the façade. For navigation applications a photorealistic appearance of buildings is not necessary and abstracted representations can be more helpful. Ferwerda (2003) defined a *functional realism* to support task driven visualizations. This type of realism seems an appropriate way of rendering for pedestrian navigation, as well. The layered approach enables to control the content of the façade texture by switching layers on and off. In that way a distinct level of realism (LoR) can be achieved, as it is already possible for geometry by the definition of the level of detail (LoD) (Open Geospatial Consortium, 2008a). By using flexible layers it is also possible to mix realistic elements and thematic information in the façade texture. The thematic information can be modelled by additional pulse layers (Figure 4a). Pulses can be linked to a texture tile containing "abstract" information like arrows for route instructions (Figure 4b) or can be linked to colours in order to represent attribute values (Figure 4c).

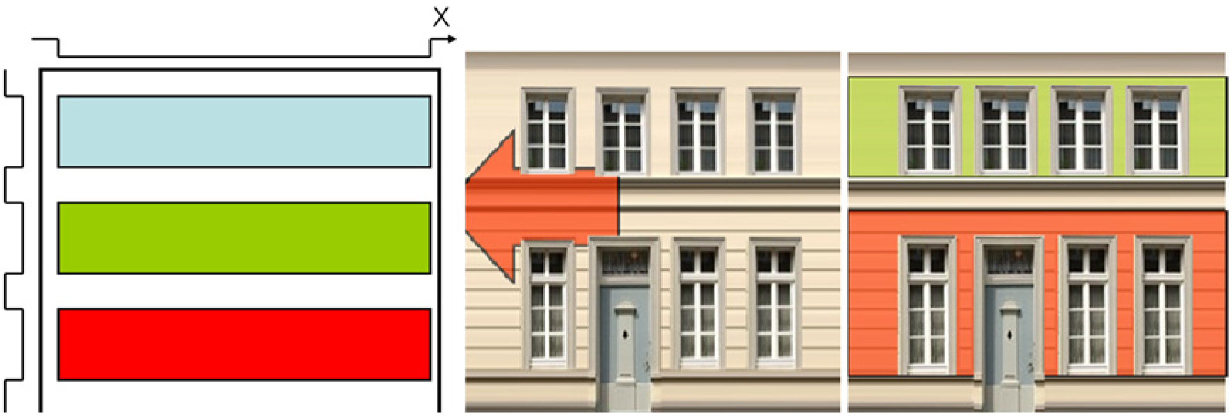

Figure 4: a) thematic elements defined by additional pulse layer; b) route instructions in façade; c) pulses with thematic information integrated into the façade (e.g. building usage per floor).

The arrangement of the tiles to build the overall façade texture can be done in various stages of a client-server system. Dependent on the capabilities of the client device it is possible to build the façade texture on server side, on client side or even on the client's graphics hardware if it is providing a programmable rendering pipeline. The last option is the most flexible, as it can change the façade appearance during the rendering process, if that is supported by the shader programme. In an existing reference implementation it is possible to change the LoR in several steps without reloading texture information from the server. The programme can add and remove pulse layers as part of the 3D rendering process.

For rebuilding the façade texture directly on the graphics board it is necessary to transfer the description of the façade onto the graphics chip and make it accessible by the shader programme. In the presented scenario it is necessary to provide the information about pulses, 'texture-layers' and 'texture-tiles', which need to be applied to areas where pulses overlap. This information is transferred in form of two textures: a texture atlas with the necessary texture-tiles and a 'data texture' that encodes the façade description.

The data texture (Figure 5) encodes information about layers and pulses. It also holds information on the texture tiles in the texture atlas and about the associated layers. In the data texture, which is a floating point texture, all four channels (R, G, B, Alpha) are used to transmit values to the shader programme. In Figure 5 some of the elements are depicted. Three rows build the information for one texture layer. They contain the y/x-functions (pulses), and the linked tiles from the texture atlas. If a pixel is found that is located inside a pulse the shader will look up the colour information in the associated texture tile. By setting a 0/1 at the 'active' value a layer can be switched on and off completely.

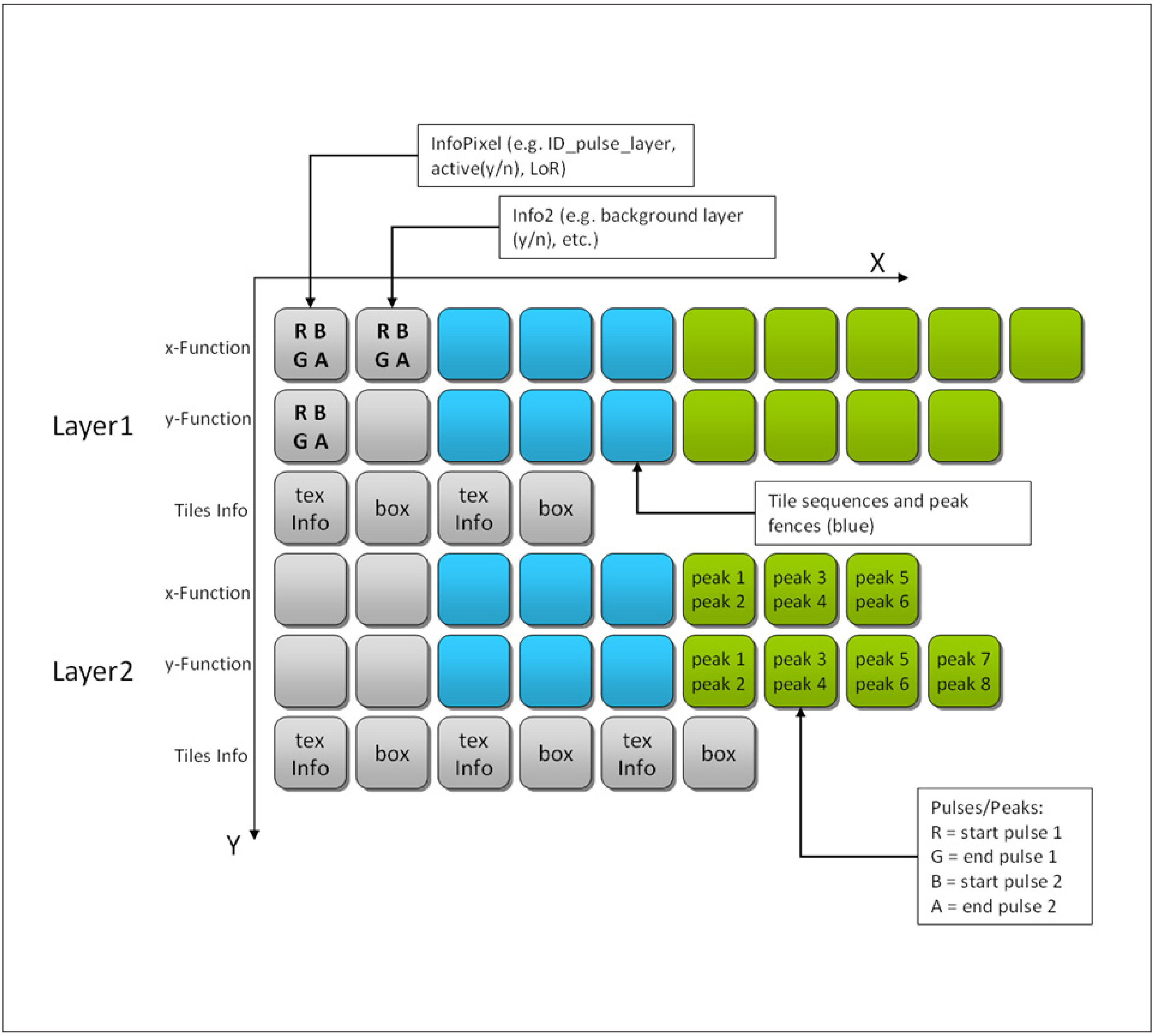

Figure 5: Schematic and simplified structure of the façade description encoded into a 'data texture'.

The rendering in this concept is done in two steps (Figure 6). One shader programme is responsible for generating the procedural texture using the description and the texture tiles. The result is stored in a texture object and a second programme renders the scene with the generated textures applied to the according geometries. This approach has the advantage that the procedural texture needs to be generated only when a new object comes into sight or if a different LoR needs to be rendered. The rest of the time a standard texture mapping is done that is not as computationally complex as the shader for the procedural texture generation.

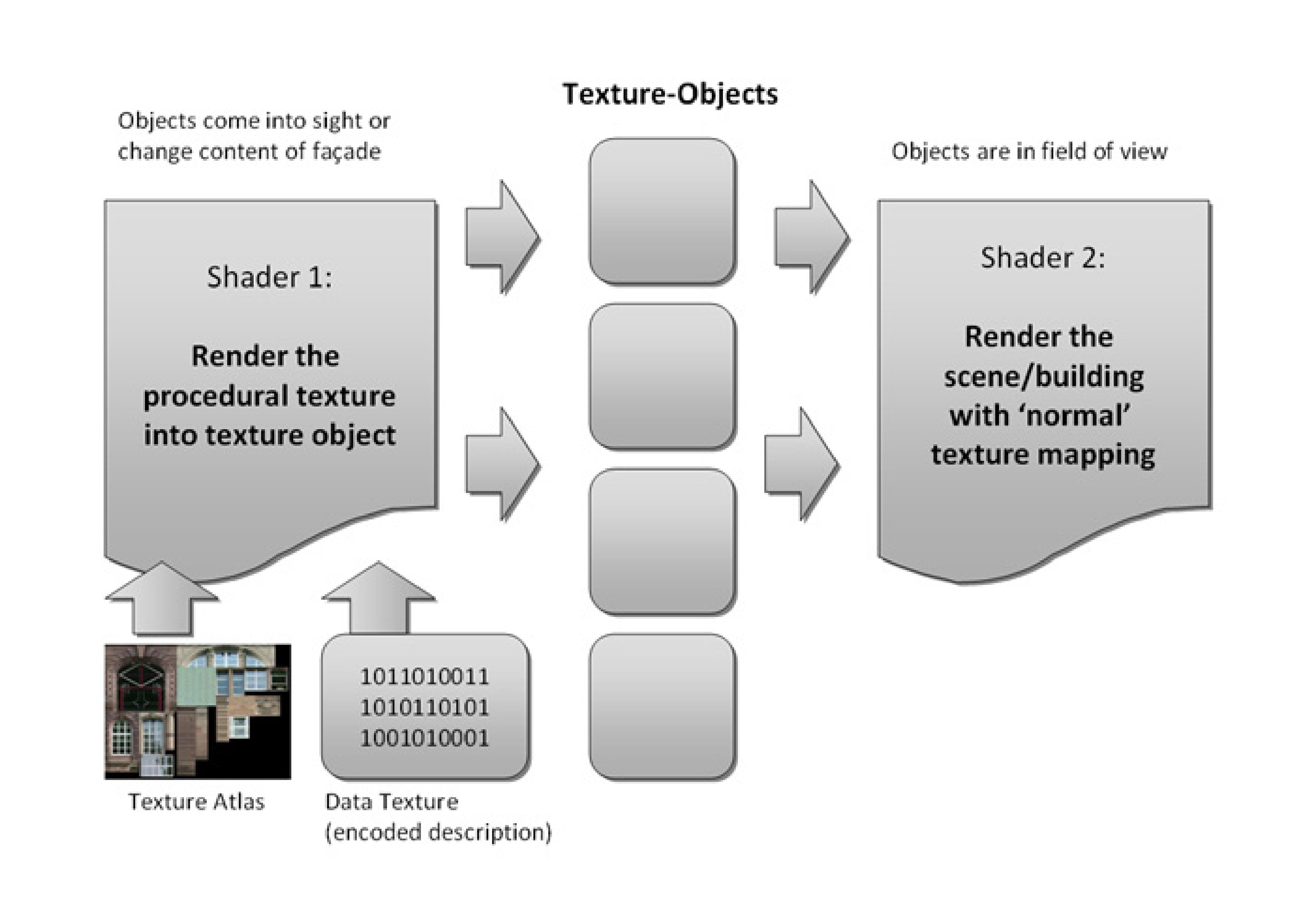

Figure 6: Concept of using two separate shaders for the procedural rendering.

3.1 FUTURE WORK ON PROCEDURAL TEXTURES

One objective of future work is to integrate a fixed layer for real time information. The concept for real-time data is to integrate a layer, which visualizes values, textures or video frames inside the façade. This layer separates the remaining space into foreground and background (Figure 7). Layers holding façade elements can either be in front of the real time layer or behind it. In this way the z-order of the real time data can be adjusted according to the specific scenario and can be changed relatively fast within the rendering process generating the procedural texture. In this case not one procedural texture is rendered, but a foreground and background texture, containing transparent pixels where required. So this approach is more or less a 'procedural multi-texturing'-method. The procedural part produces the three textures (foreground, real-time, background) and the standard rendering shader does the multi-texturing for the façade. This method will combine real-time information with the layered pulse function approach for even more flexibility for façade textures in terms of information visualization.

4. INFRASTRUCTURE

The presented concepts in the previous sections are investigated in the project 'Mobile Navigation 3D – 3D city models for mobile navigation systems' (MoNa3D). The project examines the use of digital urban models and the presentation of landmarks for mobile pedestrian navigation. The project is conducted by HFT Stuttgart – University of Applied Sciences and the University of Applied Sciences Mainz together with partners from industry. Besides the identification of landmarks and their appropriate and flexible visualization, the project also investigates the implementation of a suitable system infrastructure based on open standards.

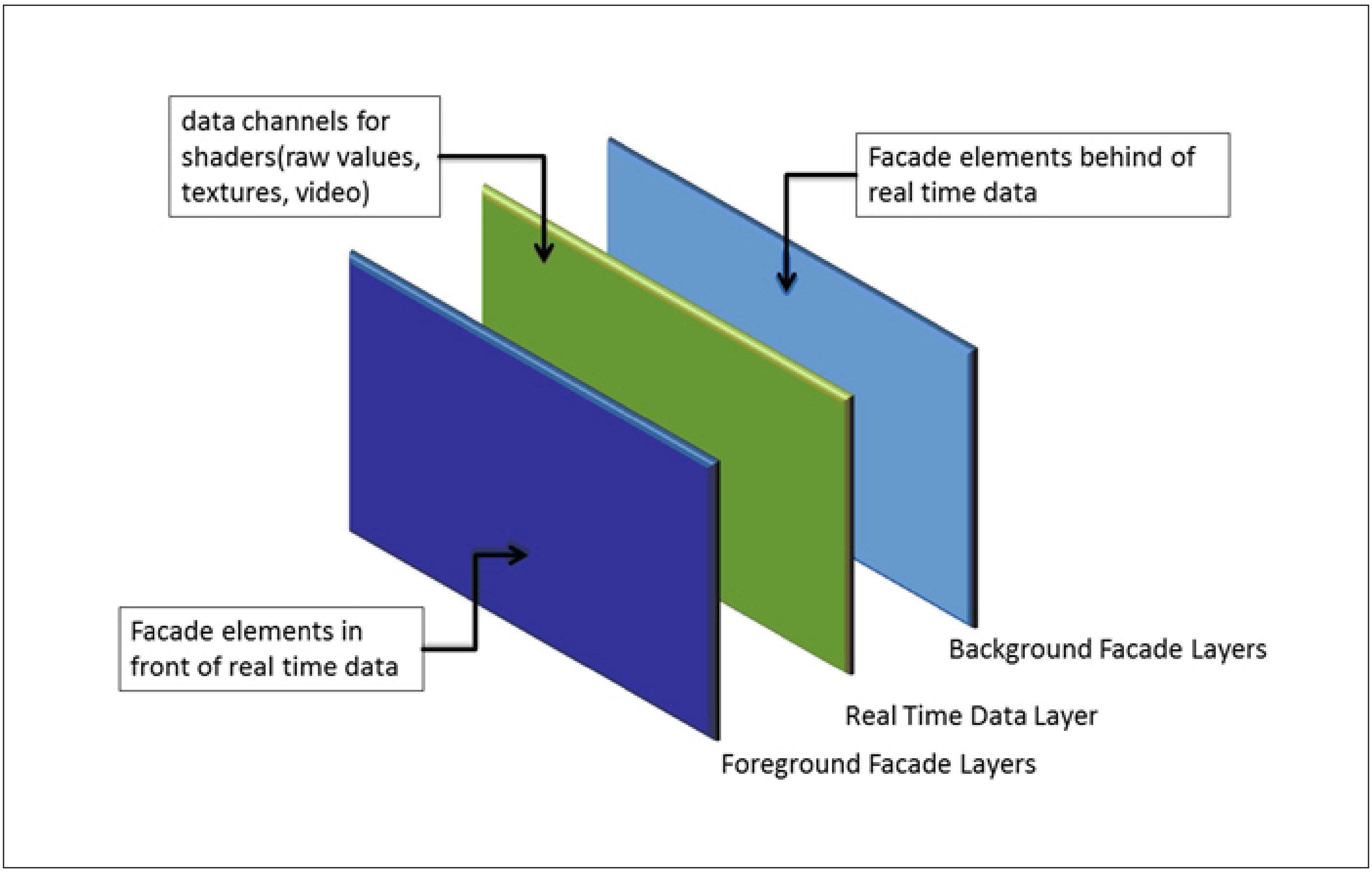

Figure 7: Real time data integration concept.

4.1 SERVER-SIDE SYSTEM

The system 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, 2008b), a Web 3D Service (W3DS) (Open Geospatial Consortium, 2010), a Catalogue Server and others in order to fulfil the task included in the workflow (Figure 8). 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. The workflow for requesting a route with the according instructions is managed by the 'mediator layer'. Firstly, the route is requested from the OpenLS service, which provides the 'classical' route that is calculated based on a line graph, which is the basis for further steps in the process (Neis, et. al., 2007).

This information is enhanced by the 'semantic route service' that identifies the relevant landmarks for the given route and adjusts the route instructions so that they fit to the landmarks. The route and the IDs of the landmarks are then provided to the Web 3D Service. Based on this information the W3DS can query the 3D model and identify the landmarks using the IDs received from the semantic route service. With the given route geometry and the relative position of the landmarks it is possible to generate textures that include appropriate navigation hints and additional information relevant for the user. 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 (Coors, Rossignac, 2004) of the data can take place in order to optimise transmission performance. The format of the mediator's response depends on the client. It would be possible to use standard formats like X3D (ISO, 2007) or proprietary formats and custom viewers.

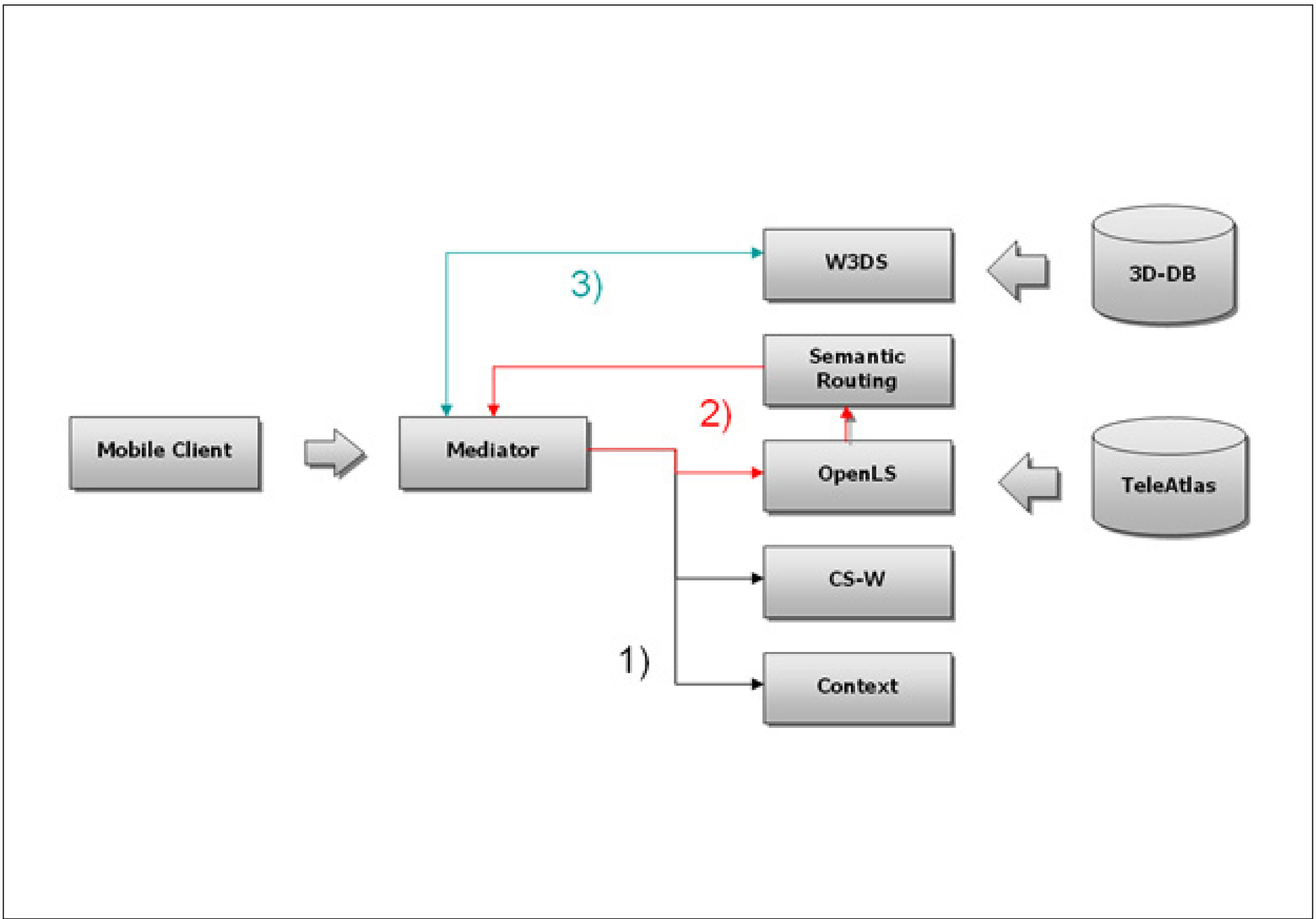

Figure 8: The MoNa3D system workflow: 1) find available servers and user settings 2) find route and appropriate local landmarks 3) query 3D scene.

4.2 XML-BASED DESCRIPTION OF THE FAÇADE TEXTURE

In order to communicate the façade description between the involved system components a XML schema was defined. This schema makes it possible to specify the included layers as well as an optional background colour (Figure 9).

One layer consists of an optional name that enhances human readability, links to the texture elements, the pulse functions in x- and y-direction, as well as hints for applying texture tiles within the defined pulse area. The 'hints'-section is further subdivided into 'wrapping', 'constraints', a texture sequence, the 'operation' that need to be applied and the layer 'priority'.

The 'wrapping' flag in this schema defines how the texture tile is actually applied to the zone that the overlapping pulses define. A tile can be scaled to fit to the pulse or it can be repeated in x- and/or y-direction. The 'constraints' value specifies if an element should be applied under any circumstances or if it is not considered when it has a lower priority and partly overlaps with another element. The 'texture sequence' can actually be used when a layer is linked to more than one tile. In this case it is possible to define in which order the associated tiles should be applied to the specific pulse areas. 'Operation' only supports two values at the moment: 'add' and 'substract'. This value allows controlling the behaviour of the programme in respect to pulse values. Pulses functions can take values 0 or 1. In case of 'add' active areas are defined by overlapping pulses of value 1, whereas 'sub' turns areas active where pulses have the value 0.

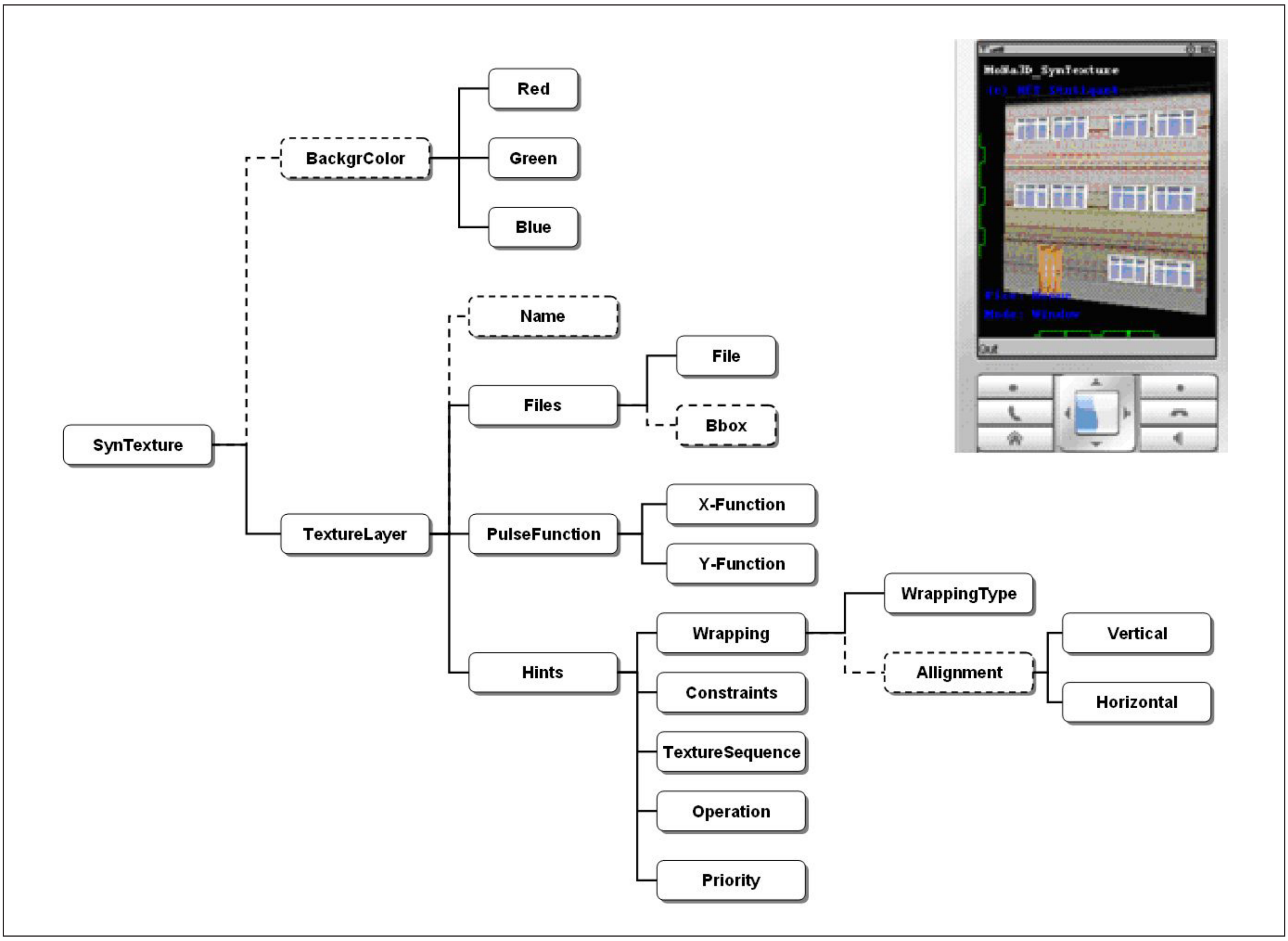

Figure 9: XML-based schema for façade descriptions.

4.3 MOBILE CLIENT PROTOTYPE

Figure 10 provides a sample of the 3D visualization on a mobile device. The client application is realized as a JavaME application. The prototype application allows navigating to specific streets, coordinates or the centre of town. If the exact street name cannot be found by the system a list of possible candidates is provided. On the right side the start and the end point of a route is shown. The 3D city model is based on TeleAtlas data and for routing TeleAtlas and OpenStreetMap data is used.

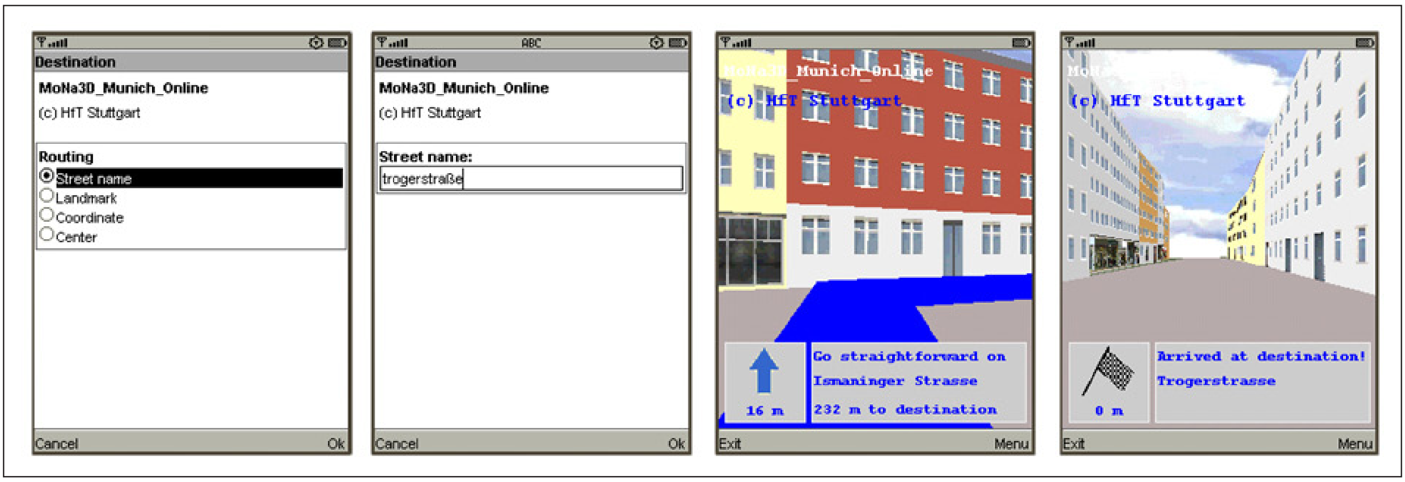

Figure 10: 3D visualization on the mobile device.

Figure 11: Mobile Client running on a Nokia smartphone.

5. CONCLUSIONS AND OUTLOOK

Digital 3D city models can be used to support navigation in urban environments and be a real benefit for pedestrians. They can especially help to compensate possible positioning inaccuracies of the GPS device, as well as the absence of orientation information. By navigating urban space using landmarks, the user can follow an intuitive way of navigation and conveniently choose their own way to the next waypoint/landmark. This seems to be most appropriate in open spaces like squares, which can hardly be approximated by a line graph. For the flexible and situation dependant visualization of landmarks and other objects the concept of procedural façade textures was presented, which can be used to integrate navigation hints into the façade of a landmark. With this approach it is also possible to control the level of realism of the object, as photo realism is not always necessary or even counterproductive.

Future investigations would be necessary in the field of visibility of landmarks, linking indoor and outdoor navigation as well as mobile rendering performance and flexibility. Another field of further research could be the combination of 'classical' route finding based on line graphs and the 'robotics' approach for squares and other open spaces.

ACKNOWLEDGEMENTS

We would like to thank the MoNa3D team, especially TeleAtlas for providing data for the project. We would also like to thank Thorsten Schulz for the implementation of the mobile viewer and Alexander Zipf and his team for developing routing and landmark detection functionality. We would also like to thank the German Ministry for Education and Research for funding the MoNa3D project and the City of Stuttgart for providing test data for our case study area.

References

accrossair, Nearest Tube application, online reference: http://www.acrossair.com/acrossair_app_augmented_reality_nearesttube_london_for_iPhone_3GS.htm, last visited: 09.02.2010

Coors, V., Elting, C., Kray, C., Laakso, K. (2005): Presenting Route Instructions on Mobile Devices: From Textual Directions to 3D Visualization. In: Exploring Geovisualization, Dykes, MacEachren, Kraak (Editors), Elsevier, pp. 529-550.

Coors, V., Rossignac, J. (2004): Delphi: Geometry-based connectivity prediction in triangle mesh compression. In: The Visual Computer, International Journal of Computer Graphics, Vol. 20, Number 8-9, Springer Verlag, Berlin Heidelberg, pp 507-520.

Dijkstra, E. W. (1959): A note on two problems in connexion with graphs. In: Numerische Mathematik. 1, S. 269-271

B. Elias, Sester, M. (2002): Landmarks für Routenbeschreibungen. In: GI-Technologien für Verkehr und Logistik, IfGI PRINTS, Vol. 13, Universität Münster.

Enkin, project on AR compass/pedestrian navigation, online reference: http://enkinblog.blogspot.com/ & http://vimeo.com/843168, last visited: 08.02.2010

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

Gaisbauer, C., Frank, A. U. (2008): Wayfinding Model for Pedestrian Navigation. In: Proceedings of the 11th AGILE International Conference on Geographic Information Science 2008, University of Girona, Spain.

Ganitseva, J., Coors, V. (2010): Automatic Landmark Detection for 3D Urban Models. In: Kolbe, T., König, G., Nagel, C. (Hg.): Proceedings of International Conference on 3D Geoinformation. Aachen: Shaker Verlag GmbH (Int. Archives of Photogrammetry, Remote Sensing and Spatial Information Science, Vol XXXVIII-4/W15), S. 37-43.

Glander, T., Döllner, J. (2009); "Abstract representations for interactive visualization of virtual 3D city models", In: Computers, Environment and Urban Systems 33, pp.375-387

Glander, T., Peters, D., Trapp, M., Döllner, J. (2009): 3D Wayfinding Choremes: A Cognitively Motivated Representation of Route Junctions in Virtual Environments, In: Advances in GIScience, Lecture Notes in Geoinformation and Cartography, 2009, pp. 407-427.

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

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.

Layar, Mobile Augmented Reality Browser, online reference: http://layar.com/, last visited 08.02.2010

Neis, P., Schilling, A., Zipf, A. (2007): 3D Emergency Route Service (3D-ERS) based on OpenLS Specifications. GI4DM 2007. 3rd International Symposium on Geoinformation for Disaster Management. Toronto, Canada.

Open Geospatial Consortium (2008)a: City Geography Markup Language (CityGML).

Open Geospatial Consortium (2008)b: Location Services (OpenLS).

Open Geospatial Consortium (2010): Web 3D Service (W3DS), Discussion Paper.

Parish, Y., Müller, P. (2001): Procedural Modeling of Cities. Proceedings of ACM SIGGRAPH 2001, ACM Press, pp. 301-308

Plesa, M., Cartwright, W. (2008): Evaluating the Effectiveness of Non-Photorealistic 3D Maps for Navigation with Mobile Devices, in Map-based Mobile Services, Meng, Zipf, Winter (Eds.), ISBN-978-3-540-37109-0, Springer-Verlag.

Raubal, M., Winter, S. (2002): Enriching Wayfinding Instructions with Local Landmarks. In: Lecture Notes in Computer Science, Volume 2478/2002, Springer Verlag, pp. 243-259

Schulz, Th., Bauer, M., Coors, V., Zipf, A. (2009): Mobile Navigation mit 3D-Stadtmodellen. Beitrag zu "Ortsbezogene Anwendungen und Dienste", Fachgespräch der GI-Fachgruppe KuVS, 10.-11.September, Bonn.

Sorrows, M., Hirtle, S. (1999): The Nature of Landmarks for Real and Electronic Spaces. In: Spatial Information Theory. Freksa & Mark (Editors), Proceedings of COSIT'99, Springer Verlag, pp. 51-64.