3D Navigation Systems based on Synthetic Texturing

[1]Behnam Alizadehashrafi
[2]Alias Abdul Rahman
Department of Geoinformatics
Faculty of Geoinformation Science and Engineering University Teknologi Malaysia 81300
[1]behnam5us@gmail.com
[2]alias@utm.my

Volker Coors
Stuttgart University of Applied Science
Schellingstr. 24
70023 Stuttgart, Germany
volker.coors@hft-stuttgart.de

Thorsten Schulz
Stuttgart University of Applied Science
Schellingstr. 24
70023 Stuttgart, Germany
thorsten.schulz@hft-stuttgart.de

ABSTRACT

Navigation system is the order of the day to pilot people to their destination. This paper focuses on the uses of mobile devices such as PDA's and smart phones without additional hardware to direct pedestrians based on synthetic texturing along with 3D modeling. Each facade is reconstructed by arraying small sized textures in respect to their geometries in different layers. In the process of texture generation cropping, rectifying, removing disturbing objects and exposure setting should be done in advance. Unlimited number of layers with different priorities and their horizontal and vertical pulse functions and texture files can be utilized for creating a simple square looking facade. Each 3D model is created by mapping the synthetic textures on the 3D geometries of each building's 3D model. The processes to create the synthetic textures as well as their usage in mobiles' context are described in detail in this paper.

Keywords: Synthetic texturing, pulse function, 3D urban modeling, navigation system, lightweight geometry

Introduction

Different kinds of road presentations in navigation system devices are in use. They are based on 2D map, text, voice and pictures. People like to identify their world in the navigation system display to compare with their position and environment. MoNa3D[1] is carrying out a project by use of synthetic texturing and pulse functions for 3D navigation purpose. The two main goals of the project are providing a cognitive semantic route description by using landmarks in 3D, which allow context dependent personalized navigation support and developing an approach to create suitable non-photorealistic building textures using image processing methods and synthetic textures along with a corresponding compression system for an efficient storage and transfer of 3D building models (Coors and Zipf, 2007). In particular, the generation and usage of synthetic textures are addressed in this paper.

The Motivation

The idea of having 3D navigation software on smart phones and PDA's along with limited resources on these devices, leads us to generate a small program with less overhead and high performance by means of combining photo realistic with pure synthetic approaches. For this issue an efficient rule of thumb is repeating a small and high quality texture in arbitrary vertical and horizontal direction. The texture image file even can be a pixel which is actually a color map. The impression of having unlimited number of layer with different priorities,

[1] MObile NAvigation 3Dimentional

gives us the opportunity to generate any kind of complex and historical facades. The aim can be achieved from the behavior of raster based applications.

Related works

Using landmarks in mobile navigation systems could improve way finding approach. In this field enriching way finding instructions by means of local landmarks is a reliable method. Simple instructions are defined like geometric data for street network and shape, color, visibility for a determinate facade as visual attraction parameters. Semantic attraction parameters of outstanding landmarks are defined as cultural and historical aspects, Level of importance and explicit marks e.g. street signs. By using these definitions and information, landmarks are extracted from dataset and provide Point of Interest (POI). In fact the POI is a hard coded and predefined data which is geo coded in spatial datasets see Raubal and Winter (2002).

The decision for implementing 3D content into navigation system rose at the end of the nineties after a huge progress in hardware designs. First of all, generating 3D model of the urban area is necessary which can be achieved by creating wire frames out of filtered point clouds. Nowadays this procedure can be done semi-automatically out of laser data or stereo image pairs and photogrammetric methods (e.g. Epipolar match points and so on). High quality and rectified photographs from geometries are necessary for mapping on the wire frames. For this aim, disturbing objects must be removed from shadow less photographs. Finally mapping process can be done see Schulze and Horsel.

Figure 1: Mapping the images after rectification, obstacles removal and radiometric adjustment on the light weight geometry of Building 2" of Stuttgart University of Applied Science in VRML 2.0

The problems of conventional photogrammetry methods are disturbing objects, leaning geometries, shadows, and reflection, time consuming, expensive and "*heavy*" models, which make it unfit for the intention of navigation system devices. Another project for 3D navigation system purpose on smart phones, is M-LOMA[1] Nurminen, A and Tuominen, J (2008). This application which is programmed in C++, can be installed on smart phones and PDA's for running VRML file. The visible part of the model can be rendered on the screen. Moreover lightweight geometries are used for the 3D modeling. In this issue VRML parsing is the first step in implementation process for only visible area on the screen. Texture processing for different LOD's comes afterwards which are created and stored separately. In the next step the visibility calculations based on PVS[2] algorithm is done. Then the visibility list encoding can be followed by geometry files packaging and compressing them in binary format.

Our aim is to generate a noble solution e.g. by enabling the creation of high quality and easy to describe facades for each object in respect to low memory and storage requirements.

Concept of synthetic textures and pulse functions Parish & Müller (2002)

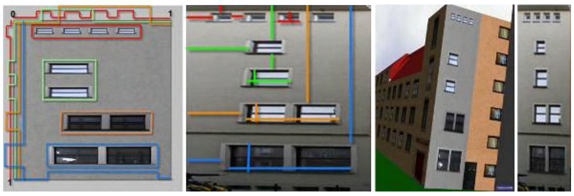
Each building consists of polygons and for each polygon there are some pulse functions in x and y axes which controls the process of texture generation. Pulse function can be easily defined with the logical values. In intersection of X and Y axes, TRUE means insert and FALSE means not to insert the texture (Coors and Zipf, 2007). In order to increase the flexibility, the number of layers of textures and pulse functions are unlimited and each of them has its own priority in respect to others. In the process of creating database for synthetic texturing method, we have to define lots of fields like number of facades which are generating an identical building, texture file name and path or directory; pulse functions and their related parameters and so on. In MoNa3D, for this issue we have defined an XML schema and we applied the same schema in this work Bauer et al (2008). Furthermore, we have created a user friend interface by means of JavaScript and ActiveX objects of Microsoft windows which can receive the parameters and generates XML file fitting to this schema. By using this JavaScript program (see Figure5(c), the distances between similar textures in horizontal and vertical direction are computed semi automatically in respect to the real geometry for pulse functions. The number of the layers for each facade can be defied by operators in respect to the

[2] Mobile LOcation Aware Messaging Application

[2] potentially visible set

 ISBN 978-80-86943-95-4

requested quality or number of different and sparse textures. Figure 2 shows the generation of facade (in five layers).



(a) (b) (c) (d)

Figure 2: (a) illustrates 4 different layers of windows along with their related pulse functions and (b) is the real resized and rectified image of the facade used for measuring process and (c) is the final model without leaning problem and (d) is the rectified form of real image of facade.

Synthetic texturing

Identifying and describing the texture

We can extract texture heuristically according to the form of the facade. Normally it is possible to find a section on every building facade which is repeating exactly with the same size, form and shape. This part should repeat in horizontal or vertical direction with the same distance and often equal size. It depends on the taste of the operator and there is no strict rule and automatic method for selecting textures. It can be a vertical slice which is repeating horizontally or horizontal slice which is repeating vertically with a distance equal to zero or any calculated value in respect to the generated XML schema. Texture of the wall can be stretched in vertical direction or horizontal direction or can be a color map or a texture which is repeating in both directions [Bau06]. The texture of the window has also the same behaviors except stretching in both directions(see Figure3 (b)).

(a) (b) (c)

Figure 3: (a) is the rectified and resized facade (c) is the texture which is extracted from upper part of (a) and repeated in the upper part of (b) in vertical and horizontal direction to generate the model.

Cropping and rectifying

After deciding which part can be extracted as our texture we have to follow cropping and rectifying procedure. For instance Figure3 (c) is the window texture which is extracted from upper part of Figure 3 (a) and repeated in both directions in order to generate the model (see Figure3 (b)). In many cases we need rectangle texture and the single point perspective effect of the camera can be removed by means of cropping along with rectification of the rectangle texture shape. In order to avoid leaning problems, it is also possible to crop and rectify the facade of the building and then crop the selected texture which is as perpendicular as possible to the exposure angle. For instance the window texture in Figure 3(c) was cropped from the rectified and cropped facade which is as perpendicular as possible to the exposure angel (see Figure 3(a)).

Removing the disturbing objects

To have a high quality texture we need to remove disturbing objects and undesired parts of the texture. For example the flying birds, moving cars, flags, signs and moving pedestrians should be omitted from the selected texture.

(a) (b)

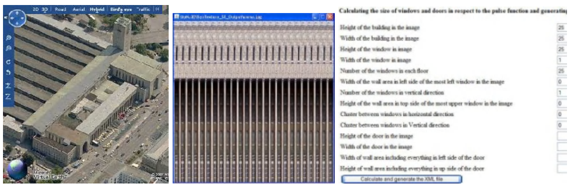
Figure4: (a) is a vertical slice texture which is cropped and rectified from the perpendicular picture of the facade and (b) is the same texture after removing the disturbing objects (e.g. flying birds, moving cars and moving pedestrians and so on). To reduce the complexity, the inside geometries were neglected as well.

Radiometric adjustment and texture size settings

In order to increase the quality of the model some additional exposure settings, radiometric adjustments, color settings, contrast settings and also texture size settings, were carried out. The texture size is an important issue for generating output image file for different applications. Our Java program generates a square image file according to our Pulse Function and it is possible to change the size of output image file in Java program for each façade even to any shape. In order to have a simple and user-friendly Java program, the size of output image file is defined as a fixed size (256*256 for popular mobile navigation systems and 512*512 for the test on computer screen and VRML.2 environment) in the respective utilized hardware and software environment.

Generating the facade semi automatically

Among the issues in generating facade semi automatically such as the geometries of the selected textures, two different issues can be preceded. Both issues depend on the form of textures and taste of the operator for choosing the texture. In the following examples we will illustrated these aspects in detailed.



(a) (b) (c)

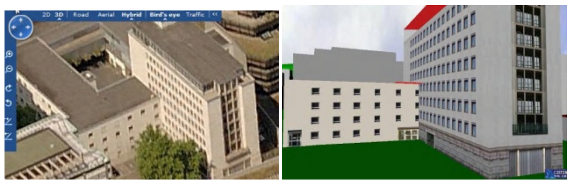
Figure 5: (a) is bird's view option of Microsoft Virtual Earth from main station in Stuttgart. (b) is the generated model of front part without knowing the Geometry of the texture in respect to the façade and (c) is the JavaScript user friendly interface for calculating the parameters and generation XML file which can be run in internet explorer

In fact, the generated texture in Figure 4 is repeated 25 times in horizontal direction. The distance between them are equal to zero. The out put image file is created without having knowledge of any parameters about the selected texture in respect to the facade.

In the next example in order to measure the parameters of texture in respect to the facade, we need to rectify and resize the original picture of the facade taken from any angle (see Figure 6 (b)) without having quality and completeness as shown in Figure 6 (a). Lets assume that window is our texture, now we can use different applications like Photoshop to measure the parameters like width and height of the window, starting point from left side, from upside, vertical and horizontal distance between them, starting point of the door from upside, from left side and so on. Figure 7 shows the comparison of the generated model of Figure 5 (see Figure 7(b)) with bird's view option of Microsoft Virtual Earth (Figure 7 (a)).

(a) (b)

Figure 6: Right image (b) is a picture of facade from any angel and the left image (a) is rectified and resized form of (b) for measuring purpose.



(a) (b)

Figure 7: The comparison between the models of the buildings of University of Stuttgart (b) with bird's view option of Microsoft Virtual Earth (a).

Generating lightweight geometry

For heavy geometry on each surface we need to map a texture. To decrease system overhead for mobile navigation system, generating lightweight geometry is necessary. The geometries as the roofs and floors are less important and in many cases we do not need to map any texture on them except some special cases. For instance The *Mercedes isometric Star* of the main station which is rotating could also be helpful to identify the building (see Figure 9(b)). Normally, a small texture can be repeated or stretched or even one pixel color map would be sufficient to generate the output image file to map on the light weighted geometry of roof. Removing unnecessary geometries is based on the taste of the operator which might be done heuristically. This is the main reason that we couldn't define any semiautomatic method for this issue and this step has been done manually. For instance for the front part of the main station (in our existing VRML file) there were 20 points for the walls and we reduced them to four points in outside which are enough for the mapping purpose (all the models are for outdoor 3D navigation systems and indoor geometries should be omitted). In addition, after removing the small geometries we can represent them with different radiometric adjustment and exposure settings (in this case the user can "feel" them as a small geometry (see Figure 8)).

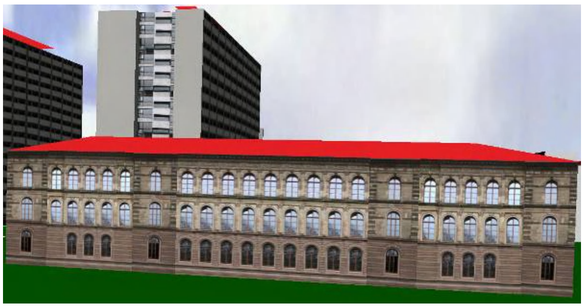


Figure 8: The representation of small geometries with different exposure settings and radiometric adjustments.

Mapping the output image on the geometry

In the process of mapping output image on the geometry, we have defined the output image file of Pulse Function as a square shape. We noticed that if the height of building is less than half of the width of the building or width of the building is less than half of the height of the building, we will face with deformation and lack of quality in the process of mapping square output image on rectangle geometry. In order to deal with this problem we generated the output image for the geometry in a part of output image file and then used that section for the mapping and stretching on the geometry (Figure 9). In addition to above mentioned method we can also use different resolutions and shapes or transparent in our Java program to deal with the problem of deformation or lack of quality and strict square shapes.

Figure 9: The mapping and stretching a part of output image for the geometry.

Test environment

25 buildings were modeled from the Stuttgart University of Applied Science to the main train station in Stuttgart and represented in VRML2 as are illustrated in Figure 10 (buildings around white line as a shortest path). The walls of some of models are generated with just one pixel or one color map which is repeated in vertical and horizontal directions. Many of other historical buildings like "Building 1" of the Stuttgart University of Applied Science was generated with a vertical slice of texture (which was repeated in horizontal directions (see Figure 11). In some of buildings' models the pictures of the facades are resized and placed or mapped on the related geometry with respect to the drawing rules and level of details.

Figure 10: The white line represents the path of the 25 buildings, modeled from main station to the Stuttgart University of Applied Science.

Conclusions

Synthetic texturing is one of the outstanding methods for the aim of 3D modeling of urban area which has been proven in this procedure. The pulse functions are very useful in texturing the facades in respect to their own priority in the representation for 3D mobile navigation system devices with high quality and small size for the 3D models. The concept of having unlimited number of layers with their own pulse functions and priorities, provides high flexibility and easy to generate any kind of desired façade for each geometry in the 3D model. Furthermore, this method is not time consuming compared to conventional photogrammetric methods.

 ISBN 978-80-86943-95-4

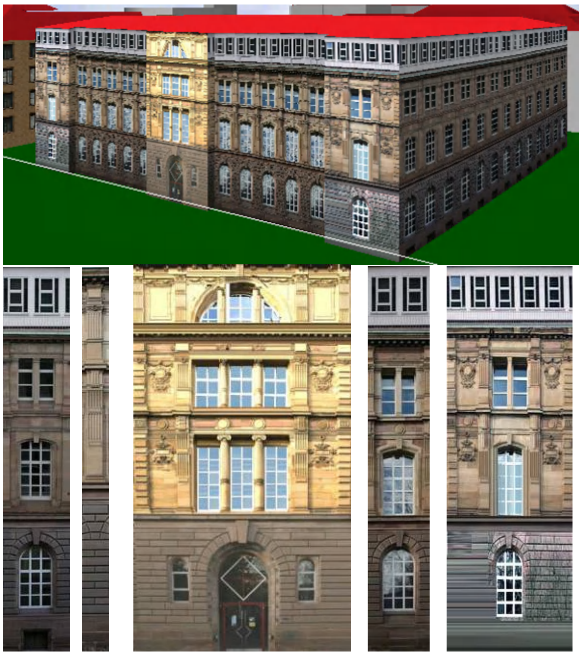


Figure 11: The model for "Building 1" of Stuttgart University of Applied Science via VRML2 and used textures.

ACKNOWLEDGMENTS

We appreciate the Stadtmessungsamt Stuttgart for providing the data and Federal Ministry of Education and Research for enabling the project. We also appreciate the staff of Stuttgart University of Applied Science that they supported us during this study.

REFERENCES

Bauer, M. & V,Coors. & T.Schulz &A.Zipf (2008) Zur Nutzung von 3D-Stadtmodellen für mobile Navigationssysteme

Coors, V. & A.Zipf (2007): MONA 3D -- Mobile Navigation Using 3D City Models

Nurminen, A. & J.Tuominen (2007): Mobile LOcation-aware Messaging Application

Helsinki University of Technology Industrial Information Technology Laboratory-P.O. Box 9204, 02015 HUT, Finland-MSc

Parish, Y.I.H. & P. Müller (2002) "Procedural Modeling of Cities," *Proc. Siggraph,* ACM Press, 2001, pp. 301–308.

Raubal, Martin &S. Winter (2002): Enriching Wayfinding Instructions with Local Landmarks - Institute for Geoinformatics, University of Münster - Vienna University of Technology, Austria

Schulze, M. –Horsel (2007): 3D Landmarks–Generation, Characteristics and Applications. ISPRS Conference – 3D Architecture 2007.