BETWEEN PHOTO-REALISM AND NON-PHOTO REALISTIC RENDERING – MODELING URBAN AREAS FOR REAL TIME VR

Guenter Pomaska

Bielefeld University of Applied Sciences, Faculty of Architecture and Civil Engineering, Germany
gp@imagefact.de - www.imagefact.de

KEY WORDS: Architectural, Building, Virtual Reality, Multimedia, Panorama, Visualization

ABSTRACT:

Non-photo realistic rendering techniques generally admits as painterly rendering, line art rendering, computer sketching and half toning. These techniques are widely used for producing stylized artificially looking drawings. Computer models in architecture are often targeted to photo-realism. Rendering techniques like ray tracing or radiosity can provide realistic looking results. The quality depends on the definition of material, light and environment. Applying real time virtual reality is still a compromise between navigation speed and photo-realism.

Using image textures from photos, rectifications or orthophotos, and mapping them to a generalized geometry does not give sufficient results. The first impression of a so-called photo realistic model might be nice. A closer look displays discontinuities in shadows, radiometric distortions or perspective errors. Objects occur in the scenery, which should not be there. Photo realistic computer models provided in this way are only useable for visualization. Improved solutions exist in the use of a pre-processor, preparing the CAD model for good quality rendering and high-speed interactive navigation.

Taking into consideration that non-photo realistic rendering techniques are used in real time VR as well, we should ask the question why trying photo-realism? Under architectural modeling, we define a technique that represents a building with a certain degree of abstraction but still providing the characteristics of the buildings.

This contribution reports about modeling an urban area by converting CAD models into a VR data structure. It is designed for PC workstations and Web. The fourth dimension is the time, the epochs of the emergence of buildings and site construction.

1 SCENE MODELLING

1.1 Geometric Modeling

During the construction process surfaces, solids or NURBS representing the geometric description of a computer model. Conversion to real time viewing environments, Cortona for VRML 97 or Octoga for X3D for example, translates the structure into meshes or triangles, more or less unstructured and not as well suited as efficient scene graph structures, used as optimized geometry for real time VR.

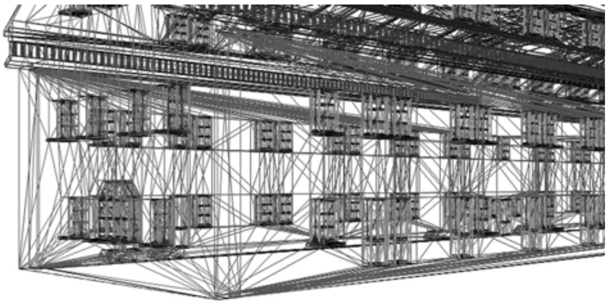


Figure 1: Wire frame rendering mode

It is obvious that conversion from CAD to VR by simply clicking one button runs into loss of performance. This aspect is subordinated here; the focus in this paper is more directed to rendering results.

Appearance of geometric elements is defined by color and / or texture.

1.2 Material Modeling by Color

Color is a specific distribution of wavelengths. Color becomes visible by reflection of wavelengths from an object and the perception of color of the human eye. The light, that is illuminating the scene, must contain the wavelengths the object is able to reflect. Other wavelengths are absorbed. White light carries the complete visible range. A well-known color model is the additive RGB alpha abstraction . The basic colors red, green and blue are defined by values between 0 and 1 or 0 and 255. Alpha stands for transparency.

The object surface is affected by color definitions for diffuse reflection and specular reflection. The **diffuse** color is essentially the color of the object under white light. The diffuse reflection is independent of the viewing direction but corresponds to the light direction.

Specular color and shininess specify the specular highlight. Specular reflection is calculated in relation of the viewing angle and light source direction.

Other values influence the ambient light conditions. While the **ambient** intensity of an object specifies the amount of reflection of the ambient light , emissive color is a light source independent definition. With **emissive** color, an object is visible in the dark.

Transparency specifies the percentage of light that is transmitted through the object.
One can find those material settings in Open GL and VRML as well as in CAD render engines. A complete material definition in VRML is given in the following piece of source code:

```
 #VRML V2.0 utf8
 Shape {
  appearance Appearance {
   material DEF colorBlue Material {
    diffuseColor  0.0 0.0 1.0
    specularColor 1.0 1.0 1.0
    ambientIntensity 1.0
    shininess 0.5
    transparency 0
    emissiveColor 0.0 0.0 0.0
   }
  }
  geometry Sphere { radius 1}
 }
```

In practice, we do not need all of those definitions. A wall of red bricks will appear sufficient with a material definition taking diffuseColor only in one place of the file.

```
DEF redBricks Material{
  diffuseColor 0.638 0.357 0.255
}
```

The definition is referenced in a VRML format by the keyword USE.

```
Shape {
  appearance Appearance {
   material USE redBricks
  }
  geometry Box { size 0.24 .07 .115}
}
```

1.2 Lightning

Three major kinds of light sources are common in use:

- directional light
- point light
- spotlight

Directional light illuminates a scene with parallel rays of light from one direction. Point light distributes light from one position in all directions. Spotlight is a directed cone of light, known from theatres. A directional light source is added to a VRML-file as follows:

```
  DirectionalLight {
   ambientIntensity 1
   color 1 1 1
   direction 0.5 –0.5 0.5
   intensity 1
 }
```

Ambient intensity is the individual light's contribution to the overall ambient illumination; intensity is the amount of light of the specific light source. Direction defines the vector within the local coordinate system in which the light illuminates. The color of the light source should be defined carefully. White includes the complete visible spectrum. The number of light sources in a scene affects rendering time.

1.3 Modeling Material using Bitmaps

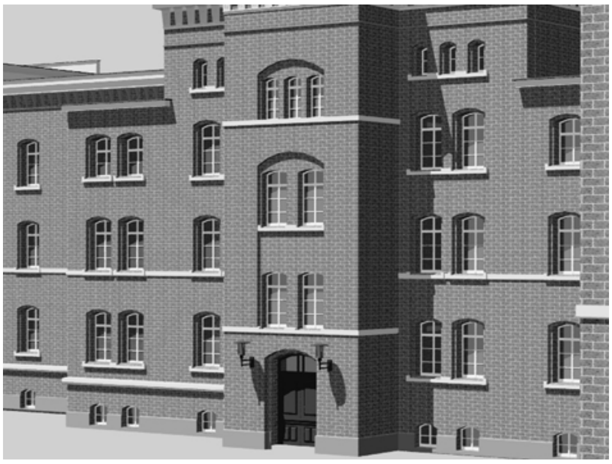


Figure 2: Texture mapping resulting in scale problems

Adding texture to an object is a step ahead to photo realistic rendering. Applied with bump mapping, textures can define a surface's shape. Bitmaps or photo textures are projected onto surfaces in different modes like:

- orthographic
- planar
- cylindrical
- spherical

Mapping a bitmap to a surface requires for every point of the 3D object surface a corresponding point in the 2D texture. The sample given in table 1 contains approx. 270 000 surfaces. In practice, it is not possible mapping textures exactly. An orthographic projection can be a sufficient approximation.

A texture is scaled and repeated until the object is completely covered. The procedure is called tiling. For materials like wood or marble procedural solid texture mapping can be applied. Figures 2 and 3 display a building textured with a bitmap of bricks. Scale problems occur and can result in arte facts like moirés. In the large-scale image, details can be detected, that do not correspond to the reality due to planar mapping procedure.

2 Renderings

2.1 Photo Realistic Renderings

Photo realistic rendering is a topic in Computer Graphics since 1960. Today we are sometimes talking about ultra realistic renderings. Very high sophisticated techniques were introduced to generate images displaying material, shadows, environment, light sources, water surfaces, landscape etc. Very high quality computer images are used in a wide range of applications today.

Despite this success photo realistic rendering has some disadvantages. The image itself displays a high degree of information that does not allow focusing on particular objects. Outlining and communicating visual ideas is limited. The amount of rendering time required for one frame makes VR applications impossible.

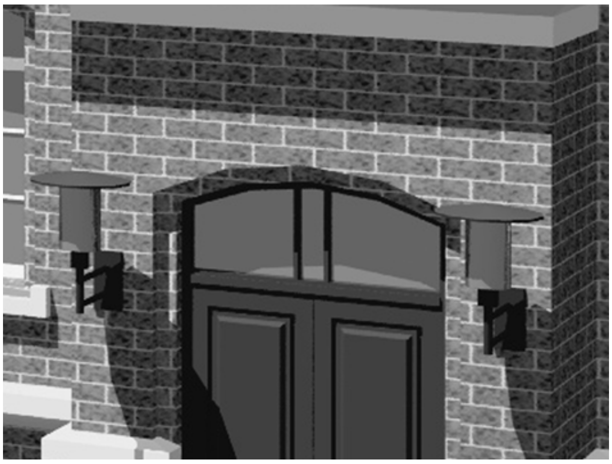


Figure 3: Texture mapping displaying unreal artifacts in large scales

Rendering is the presentation of a computer model, converting the mathematical description to a readable form, a drawing or an image. Wire frame drawings and hidden line drawings were the first rendering techniques used with pen plotters. With the development of raster graphic devices, the vector to raster conversion and shading algorithms became popular. Flat shading and smooth shading designation as local rendering models. Flat shading displays a constant color for every point of a mesh. The intensity is calculated under consideration of the angle between the surface and the light. Smooth shading interpolates the intensity along the surface. Techniques of smooth shading are admits under the names Phong-Shading and Blinn-Shading for example. The calculation only treats single objects and does not consider the complete scene. Global and much more sophisticated methods are **ray tracing** and **radiosity**. Ray tracing specifies an image surface and calculates the light equation for every point in the image by tracing, the projection ray back to the scene, taking into consideration all light sources and objects. Radiosity is a technique treating the light as energy and therefore calculates the light very exactly. Radiosity is very time consuming but delivers very good results particular for indoor scenes.

2.2 Non-Photo Realistic Rendering NPR

With non-photo realistic rendering one can produce stylized images with artistic look for enhanced graphic and visual appeal. NPR is a research topic since 1990. It is essential for outlining features particular in architectural drawings. The major kinds of NPR drawings are cartoon types and sketch drawings. Some typical examples are shown in the following figures 4, 5 and 6.

Sketch rendering is influenced by the line type settings. The fill mode of surfaces can be the object color, a pencil hatching (figure 6) or watercolor (figure 5). Calculating watercolor is very time consuming.

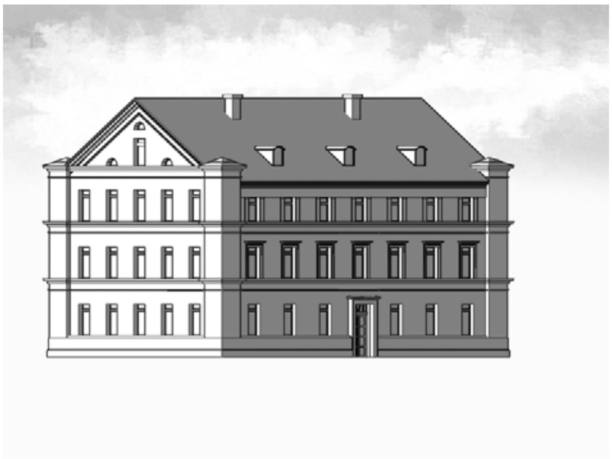


Figure 4: NPR black and white cartoon type

Ambient light color definition in cartoon rendering sets the darkest spot of an object in the scene in the rendered image. Colors of areas in low light are blended between object color and light color. Adjustment between the color mixes is the key to the desired result. Further design possibilities lie in the choice of the background that can be paper structure, bitmap or color. Mixed renderings as hinted in figure 7 may point out special characteristics.

The NPRs shown here are generated with Penguin 1.0 from McNeel.

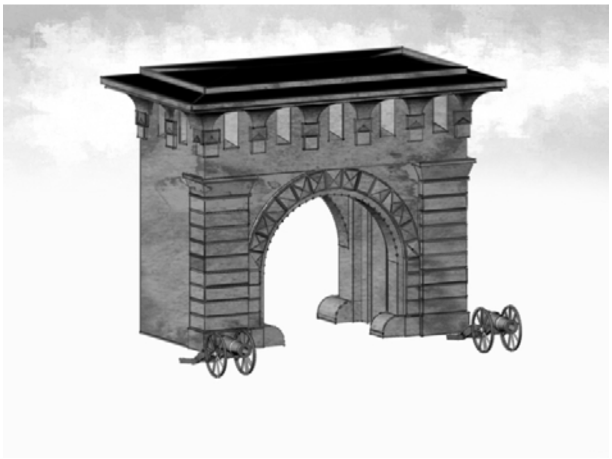


Figure 5: NPR sketch with watercolor filling

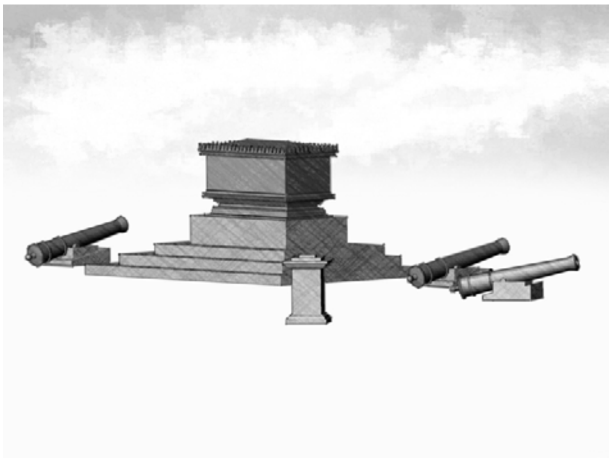


Figure 6: NPR sketch with pen type filling

2.3 Architectural Rendering

Between photo-realism and non-photo realistic rendering, we settle architectural rendering. It is not photo realistic but it corresponds to a certain extend with physical model construction. The material definition is based on a minimum of color settings. We use limited numbers of diffuse color settings. In case of metal and glass we add some specularColor. The colors used, can be gray scaled or defined according to the building's material like sandstone, bricks stucco , glass or metal etc.

The 3D perception is a combination of perspective and shadow (lightning). Placing scale-giving elements like people or trees could advice to particular ideas. Billboards and mapping on texture cards is appropriate.

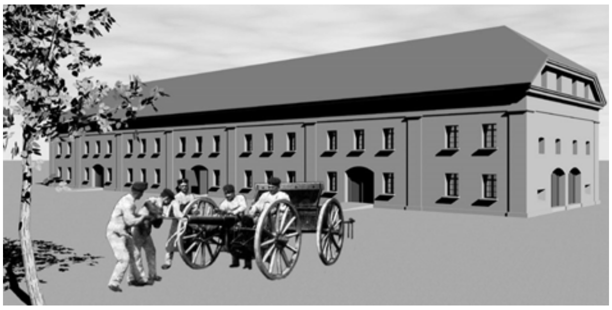


Figure 7: Billboards added to NPR building presentation

Modeling in a CAD environment and converting to VR scenes needs a strategy for structuring the VR model and some little software helpers. Modeling is usually performed in an AutoCAD environment. For rendering we use AccuRender. Material definition is performed only with solid colors. The converter translates colors one by one. Some helpers are written to transform AutoCAD blocks to VRML instances using PROTO or and the DEF / USE Functions. Street lanterns, trees can be coded as prototypes in separate files. In the original scene, those objects are referenced as EXTERNPROTO and can be called, like functions or methods in programming languages. Arguments are passed to the parameters for alterations

For DTM generation Surfer comes into the game. Surfers DTM surface should be transformed into elevationGrid nodes for performance purposes.

3 PRESENTATION OF URBAN AREAS

3.1 Level of Detail

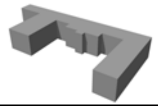
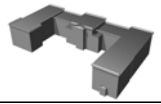
Publication of huge models requires implementation of the level-of-detail paradigm. For architectural modeling, we define the LODs as follows:

LOD 0 only block structure of buildings, elevation estimated

LOD 1 building structure including shapes of roofs etc.
LOD 2 wall openings like windows and doors
LOD 3 façade elements

LOD 4 indoor structures
LOD 5 photo textures

One of the buildings of the urban area we look at, demonstrate the amount of data and rendering time used applying the LOD structure.

	LOD 0	LOD 1
		
Surfaces	68	1510
Memory	16 kByte	16kByte
Time	24 sec	26 sec
CAD size	62 kByte	1.501 kByte
VR size	5 kByte	227 kByte

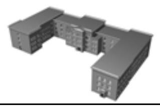
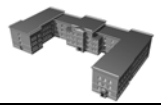
	LOD 2	LOD 3
		
Surfaces	41 750	273 080
Memory	22 574 kByte	56 617 kByte
Time	45 sec	78 sec
CAD size	19 073 kByte	67 623 kByte
VR size	2 759 kByte	15 025 kByte

Table 1: LOD compared to memory size and rendering time

The most detailed model with LOD 3 needs 6 times more VR memory as the LOD 2 model. LOD 3 is not usefully for Web-applications. LOD 2 reaches the acceptable size. In VR, the concept of LOD is the alteration of the model in relation to the viewers distance. A more detailed file is loaded if the visitor the scene enters a region in space. In VRML, it is performed with the proximity sensor. We modify this concept by passing the decision to the user.

3.2 A VRML Component System

The application used for demonstration purposes here, is an urban area with changes and developments over several epochs. We have several buildings with different level of details and we have separate epochs for each object. All files are located on a Web server in a well-defined directory structure.

A HTML Web page with GUI elements in a Form-Tag enables the visitor to select objects, the degree of detail and the epoch. The structure is comparable with a cube; the visitor is able to select multiple elements along the axes of the cube.

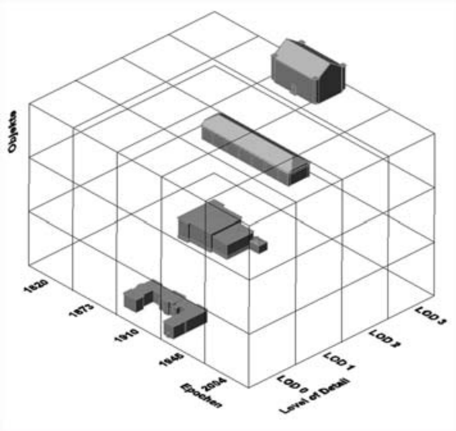
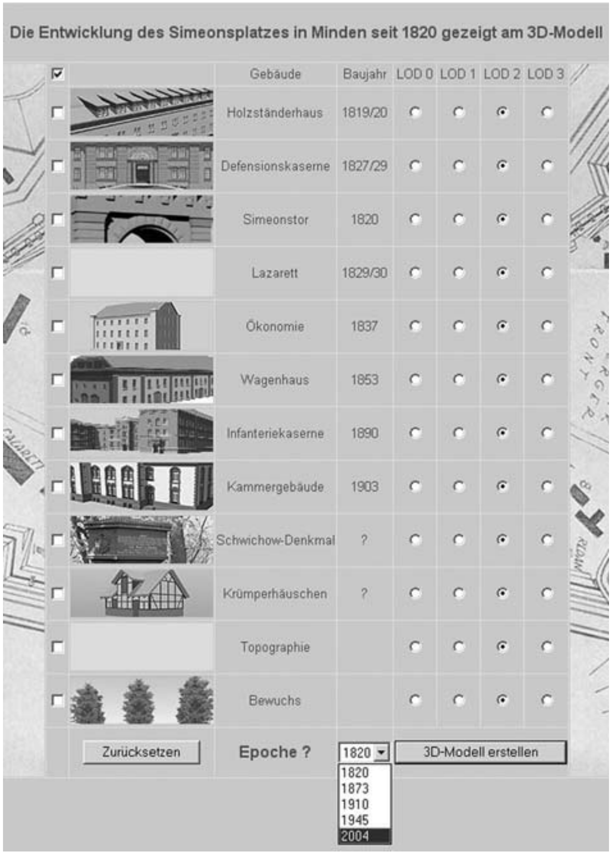


Figure 8: Selection of objects from a cube

Clicking the submit button of the form passes the selected values to a PHP script. This script writes a VRML-File with Inline-nodes, those nodes carry only a reference to a URL . During loading a VRML file, all Inline nodes are collected and displayed in the Browser. The interface of the beta version is shown in Figure 9.

The author believes that this kind of interface will encourage museum visitors to interact with the installation we are preparing. The beta version of this PHP script can is accessible By visiting the Web **www.programmierpraktikum.de/d/php** The web site is unfortunately in German language only. The set of objects is not complete there, since modeling of the area is still ongoing. One can imagine that there are some more extensions to the PHP Script on the drawing board.



Die Entwicklung des Simeonsplatzes in Minden seit 1820 gezeigt am 3D-Modell

☑		Gebäude	Baujahr	LOD 0	LOD 1	LOD 2	LOD 3
□		Holzständerhaus	1819/20	○	○	●	○
□		Defensionskaserne	1827/29	○	○	●	○
□		Simeonstor	1820	○	○	●	○
□		Lazarett	1829/30	○	○	●	○
□		Ökonomie	1837	○	○	●	○
□		Wagenhaus	1853	○	○	●	○
□		Infanteriekaserne	1890	○	○	●	○
□		Kammergebäude	1903	○	○	●	○
□		Schwichow-Denkmal	?	○	○	●	○
□		Krümperhäuschen	?	○	○	●	○
□		Topographie		○	○	●	○
□		Bewuchs		○	○	●	○
	Zurücksetzen	Epoche ?	1820	3D-Modell erstellen			

Figure 9: VRML configuration interface

References

Carey, R. , Bell, G. , 1997. *The Annotated VRML 2.0 Reference Manual*. Addison-Wesley Developers Press

Pomaska, G., 2002. *Internetpräsentation von Bauprojekten*. Bauwerk Verlag Berlin

Pomaska, G., 2003. *Implementation of Web3D Tools for Creating Interactive Walkthrouh Environments from Builduing Documentations.* ISPRS International Workshop on Vision Techniques, Ancona, Italy

Pomaska, G., 2004. *Grundkurs Web-Progranmmierung* . GWV-Fachverlage, Wiesbaden

Access to the VRML model : *www.divide-by-zero.com*
VRML Configurator: *www.programmierpraktikum.de*