

Automatic Generation of 3D Environments Using Floor Plans Drawings

Leandro R. Mattioli

Federal Center of Technological Education of Minas Gerais (CEFET-MG / Brazil)

William Gambardella, Alexandre Cardoso, Edgard Lamounier Jr., Gerson Lima

Virtual and Augmented Reality Group

Faculty of Electrical Engineering, Federal University of Uberlândia (UFU)

Paulo do Prado, José Newton

Electric Company of the State of Minas Gerais (CEMIG)

ABSTRACT

The construction process of three-dimensional scenes for games and simulators based upon virtual reality environments is composed by many tasks such as the 3-D modeling of each object and their positioning in terms of rigid body transformations. The precise layout of these objects can be previously defined by an engineer or architect, and is made, generally, by means of schematic drawings, created with the help of CAD software. However, game development engines usually don't accept these drawings as direct inputs that could be translated into objects' positions. Moreover, the level of abstraction offered by these drawings is relatively poor, using only an ordinary set of lines and circles as opposed to using groups of composite objects (blocks). As a result, companies and developers take a meaningful amount of time to deliver final versions of 3D environments. This work presents a generic framework capable of recognizing patterns in CAD drawings and automatically arranging virtual objects in scene editors, based on the provided schematics.

Great effort has been made to improve development time of three-dimensional environments by automating some of the required tasks. For instance, batch positioning 3-D models based on fuzzy inputs is possible thanks to systems such as the one proposed in [1], which receives relative positioning information between objects (e.g.: NORTHEAST) and evaluates to relative coordinates (e.g.: $\Delta x = 20$; $\Delta y = 15$), in such fashion that the scene as a whole looks realistic. Similar works were developed in [2] and [3]. While these constraint-based object placement approaches might be appropriated when modeling a bedroom or an office, they lack the precision required for dealing with virtual environments that try to clone real world environments. In such systems, modeling can be aided by devices such as 3D scanners, by photo sets that completely describes the environment or by using floor plans. The use of *ad hoc* devices might be expensive and result in meshes with an exaggerated number of polygons, thus restraining rendering performance.

Referring to the second approach (photo sets), it may as well require complex image processing algorithms that will eventually introduce distance errors due to perspective projection. Finally, if floor plans are available and each required virtual object has already been individually modeled, those can be used as inputs for an automatic scene generation algorithm, capable of placing the objects in the 3-D world by mapping position and rotation (see Figure 1).

This method has an immediate limitation: instances belonging to the same class of virtual objects but with different offsets from the ground level cannot be easily detected depending on how the floor plan has been drawn. For small scenes or environments with just a few occurrences of this problem, a simple yet effective workaround consists of ignoring the *z* dimension (or *y* dimension for Y-up coordinate systems) while running the automation script and then adjusting its values once the scene has been created in the engine's editor.

A more severe issue is noticed when dealing with old floor plans, either digitized from paper blueprints or drawn in obsolete CAD software. These drawings suffer from a quite low level of abstraction, where the document is directly dismembered as a set of lines, circles, arcs and text, instead of representing entities as composite objects. In this case, a pattern recognizer routine is needed before feeding the drawings into the automatic arranging system.

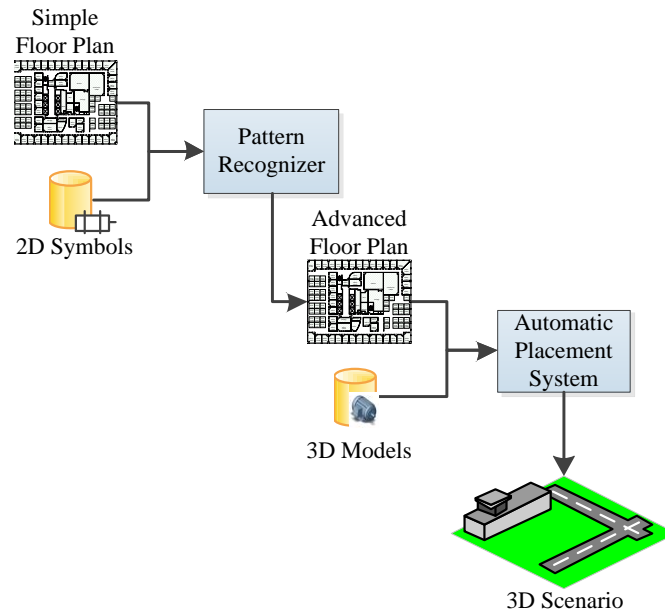


Figure 1: Automated VR creation process pipeline

This research introduces an iterative algorithm for recognizing symbols in drawings such as (but not limited to) floor plans and blueprints and their conversion to blocks and other representational structures, as well as scripts for mapping the extracted 2D entities (CAD side) to 3D entities (VR engine side).

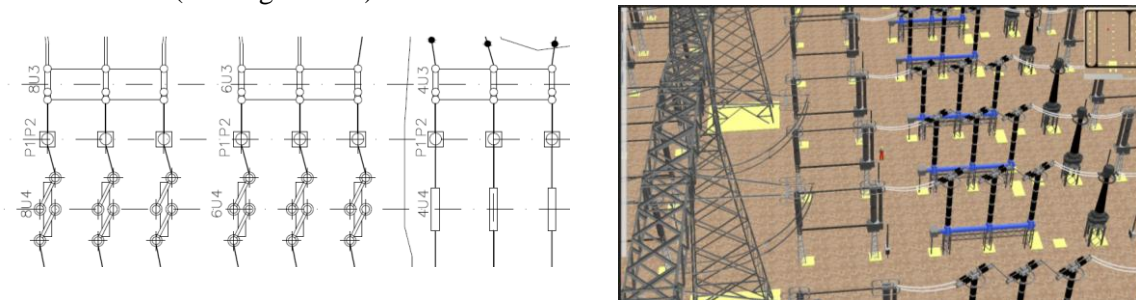


Figure 2: Jaguara power substation (floor plan and virtual environment)

The approach here described has been successfully applied in the generation of three power electric substations virtual reality environments, as shown in Figure 2. It has proved itself to be time-saving and cost-effective. Some small fine-tuning was required solely for the purpose of correcting inconsistencies introduced by older documents and the non-standardised symbology they contained.

Keywords: Architectural design, visualization, 3D Navigation, power substations

Acknowledgement: This work was supported by CEMIG's Project GT411.

References

- [1] J. Luet al. A new framework for automatic 3D scene construction from text description. In Progress in Informatics and Computing (PIC), 2010 IEEE International Conference on , vol.2, no., pp.964,968, 10-12, December 2010.
- [2] B. Coyne and R.Sproat. WordsEye: An Automatic Text-to-Scene Conversion System. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH '01), pages 487–496,2001.
- [3] K.Xu, J. Stewart and E. Fiume. Constraint-Based Automatic Placement for Scene Composition. In Graphics Interface, pages 25–34, 2002.