

DDDOOLZ

A Virtual Reality Sketch Tool for Early Design

H. ACHTEN, B. DE VRIES,
*Eindhoven University of Technology. PO Box 513.
5600 MB Eindhoven. The Netherlands*

H.H.Achten@tue.nl, B.d.Vries@tue.nl

AND

J. JESSURUN
A.J.Jessurun@bwk.tue.nl

Abstract. This paper presents DDDoolz, a desktop-VR three-dimensional voxel sketchtool. DDDoolz is developed in the Design Systems Group to explore the use of Virtual Reality technology in the early design stage. The aim is to offer a sketch-like environment in VR with an unobtrusive interface. The paper presents DDDoolz, how it is used in education and with partners in architectural practice, and some future developments.

1. Introduction

Virtual Reality technology shows great potential for support of architectural design. This technology is inherently spatial by nature, it involves the architect in an immersive environment, and it enables interactive manipulation of the design. The current state of Virtual Reality (VR) however, seems limited to presenting relatively static models of more or less finished designs without much support to change elements of the design. Using VR in a design situation still requires much effort with many intermediate steps that are not relevant for the design activity itself. It is mainly used in later design stages (see for example, Achten and Roelen 1999). Design support in the early phase is not widely implemented, although there are some pioneering examples, such as Kurmann et al. (1997), Kurmann (1998), Stellingwerff (1999), Strehlke (1999), Donath and Regenbrecht (1999), and Jung et al. (1999). One of the reasons for this lack of design support seems to be the rather cumbersome interface to create and change a design in VR.

2. DDDoolz

Design tools that are implemented in Virtual Reality should profit and make use of the specific characteristics of this technology: immersion, real-time rendering, interaction, and a natural user interface. For this purpose, it is necessary to find new metaphors that map user actions meaningfully in design actions. DDDoolz has been developed to study this with the following focus points:

- Simple user interface. The interface should be as natural as point and gesture, and not feature (3D)buttons in space. The main functionality must be quite obvious and easy to learn.
- Easy creation in VR-environment. There should be no dialogues interfering with a creation act.
- Easy manipulation in VR-environment. Changing things should also be supported with as little actions through an interface as possible.
- Navigation in VR-environment. Being able to move through the design at any point should add to the spatial understanding of the design. Navigation must be easily manageable.

DDDoolz had to be available for a wide range of users, which means that it should not take high end equipment for using it. In this way, it will be relatively easy to disseminate the prototype and get feedback from students and practice.

2.1 OUTLINE OF DDDOOLZ

After starting the system a window will display a 3D environment with a red voxel in the middle. At the left side and the upper side of the window you will find a small set of buttons and sliders. The main working modes of DDDoolz are Drawing and Navigation.

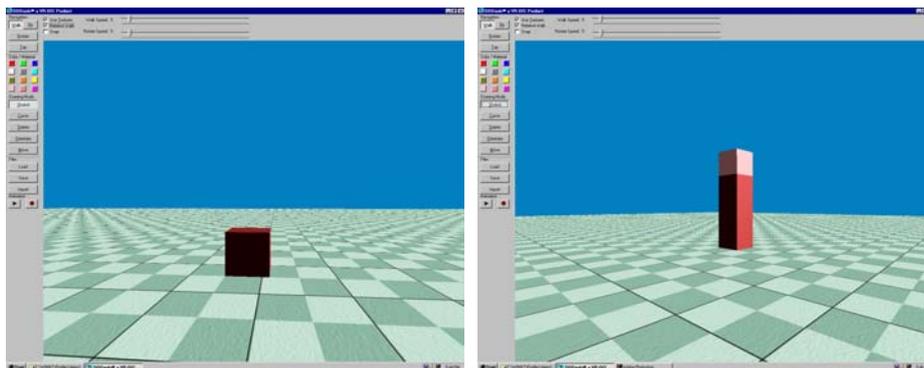


Figure 1. Left: Start screen of DDDoolz. Right: Drawing Mode.

The left bar has from top to bottom four groups: Navigation, Color/Material, Drawing Mode, and Files. Navigation consists of ‘Walk,’ ‘Fly,’ ‘Rotate,’ and ‘Top.’ The Color/Material group has a number of colour buttons. The Drawing Mode group consists of ‘Sketch,’ ‘Curve,’ ‘Delete,’ ‘Orientate,’ and ‘Move.’ The Files group consists of ‘Load,’ ‘Save,’ ‘Import,’ and ‘Animation’ (recording of a walkthrough).

In Drawing mode (‘Sketch’ or ‘Curve’) you can select one of the sides of the voxel and draw with the mouse pointer into the preferred direction. What will happen can be best described as drag and copy. Thus, the voxel will be copied while drawing along the voxel axis that is closest to the drawing direction and it will be located adjacent to the previous one. In this way a row or column of voxels can be created very fast. If the new cursor position can only be reached by drag and copy along two voxel axis directions, then the system will generate a series of voxels that form the shortest orthogonal route to the new cursor location (see Figure 1 right).

Drawing in this way seems very natural as an interface: the side of the voxel you start with implicitly defines the drawing plane in which you are working. We propose to name this general principle the Face Orientation Method (FOM). Through the Face Orientation Method, sketching direction is inferred in a very intuitive way from the actions of the user and the starting point of the pointing device. The interface is as simple as point and move, generating series of voxels. The learning curve is very low and short, and is consistent with our aim to provide tools that are as natural as possible.

Voxels are of cubic shape with 100-cm ribs, thus giving an architectural measure to the basic building block: two stacked blocks give a passage through which one can pass. This yields a very fast result in VR that can be likened to three-dimensional sketching. The dimension of the voxel forces the user to keep any model limited to a rather sober outline, thus enforcing the sketch character of the application (Figure 2, left).

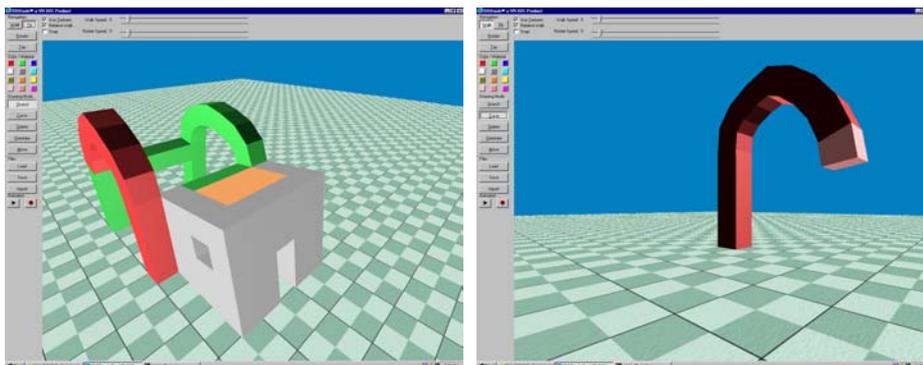


Figure 2. Left: Sketch in DDDoolz. Right: A curved line.

To escape from 'orthogonal architecture' an additional drawing style was introduced, namely drawing along curves. In curve style mode the cursor movement will be followed while copying voxels that are rotated into the drawing direction, thus forming a curved single array of voxels. Instead of drawing curves, also individual voxels can be rotated by picking on a side and rotating around the side's midpoint. Drawing can be continued started from this new position and orientation. Since the originally selected voxel is copied, setting the colour of this voxel will also determine the colour of all voxels that will subsequently be generated. Re-colouring voxels is achieved by setting and selecting a voxel with the desired colour and then moving the cursor over the voxels that should obtain the new colour (Figure 2, right).

Obviously, editing tools are available like 'Move' and 'Delete'. Moving voxels can be restricted to a step size equal to the voxel dimensions, thus staying within the (invisible) orthogonal grid. Switching off 'Snap' (top bar of the screen) will allow for positioning the voxel on any location in the 3D environment. Deleting voxels is very useful for creating holes.

In Navigation mode two options are available, namely 'Walk' and 'Fly'. In 'Walk' mode the user can navigate through the model while staying on a level of 1.5 meter (i.e. 1.5 voxel size) above the ground. Collision detection will prevent you from walking through voxels. DDDoolz is equipped with a very easy-to-use user interface for navigating. Walking forwards and backwards is triggered by pushing the left or middle mouse button. Looking around is implemented by moving the mouse into the viewing direction (left/right, up/down). 'Fly' mode is similar to 'Walk' mode with one exception, the user will now move into the viewing direction when pressing the left mouse button.

To support the creation of 3D shapes while keeping a good overview of the created model, the 'Rotate' option was introduced in the Navigation group. Switching on 'Rotate' will make the model rotate around its current midpoint, which is updated as you draw. If the user will 'Fly' to a position that gives a good overview of the model, this will prevent from repositioning your viewpoint while drawing in most cases. While the model is rotating you can continue drawing. Another feature of DDDoolz is the possibility to record a Walk or Fly session in order to create an animation or a walk through of the model.

DDDoolz has an import filter for 3Dstudio files, allowing for importing any kind of geometry (e.g. furniture) into the 3D DDDoolz environment. In AutoCAD a macro was developed to support the import of a model created in DDDoolz. This feature is extremely useful for performing a Union operation, thus creating a model with a drastically reduced number of faces. This new model could again be exported from AutoCAD and imported into DDDoolz. Obviously editing in DDDoolz with respect to this model is then limited to move and delete.

DDDoolz as it stands now, is related to VoxDesign as is developed in 1995 at the Bauhaus Universität in Weimar (Donath and Regenbrecht 1999), which featured a fully immersive VR system in which $2.5 \times 2.5 \times 2.5 \text{ cm}^3$ voxels could be ‘sprayed’ in space. A crucial difference of DDDoolz lies in the 1 m^3 voxel size which truly restricts the user to a sketch-like approach. The FOM also provides additional implicit structuring so that actions yield structures and spaces that can have a ready architectural interpretation.

2.2 IMPLEMENTATION

DDDoolz is implemented using WorldUp (by Sense8). WorldUp is a VR application development environment consisting of a rendering engine, a basic set of geometrical primitives and a Basic like scripting language. Geometries in DDDoolz can be created by either instantiating one of the basic primitives (such as the Block primitive used for the voxel), or importing geometries from other applications such as 3Dstudio. Geometries are part of a scene that will be rendered using the shape and material properties of each geometry. To each geometry a script can be attached that will be executed in every rendering cycle. The number of rendering cycles per second that determines the responsiveness of your application is usually indicated in the frame rate. Geometries are organised in a tree structure, which determines the rendering order and the order of execution of the associated scripts.

Navigation through the environment is achieved by moving an (invisible) viewpoint with a specific viewing direction through space. The viewpoint is usually attached to your mouse as the default navigation tool.

BasicScript is a ‘normal’ programming language extended with a set of functions that operate upon the WorldUp objects, such as the geometries and the viewpoint. Since BasicScript is executed within the WorldUP environment, it is hardware platform independent.

When DDDoolz is in drawing mode, the main creation script is executed. DDDoolz will detect which voxel side is pointed at and which voxel axis is closest to the drawing direction. As soon as the cursor crosses the voxel’s corner a new voxel will be created. The voxels properties are copied from the previous one. The origin of the new voxel is calculated by adding the drawing axis vector to the origin vector of the original voxel.

3. DDDoolz in First-Year Education

DDDoolz is used in education for a number of purposes. The first-year CAAD course at the Eindhoven Department of Building and Architecture runs in the first eight lecture-weeks of the year. The course is aimed to provide a basic understanding of CAAD and the use of AutoCAD together with a first

introduction to design, design processes, methods, and design thinking. Since the users are starting students, the aspiration level is set to basic skills. The course has a two-fold structure, with two hours of CAAD and design theory, and four hours reserved for exercises a week. Each week has a different exercise that builds on the theory and results taught up to that point. Students get their theoretical and exercise material via the Internet. They also upload and view peer results on the Internet through a system developed in the group (van Leeuwen et al. 1998). The main architectural theme of the exercises is located around housing, since that is something first-year students can easily relate to.

In order to broaden the scope of the lectures that are built on AutoCAD, DDDoolz is provided as a design software in the last exercise. The task is to design with DDDoolz an exhibition pavilion for some artwork that they either make themselves in AutoCAD or that they have downloaded from some model repository on the Internet. As a presentation format, they have to generate some images of the design and the objects, and present it as a kind of poster that advertises for the exhibition. This should stimulate them not only to use DDDoolz as a modelling instrument to build a pavilion, but also to actively engage the model in 3D and find challenging or telling perspectives of the design.

3.1 STUDENTS RESULTS

DDDoolz was made suitable for the standard notebooks that students at Eindhoven University of Technology have at their disposal. Technical difficulties did occur however, which delayed a portion of the submissions. Furthermore, since this was the first ‘public trial’, we found that the exercise task was formulated in a too vague way to give students enough clues for self-evaluating the designs. Many of them used DDDoolz to explore the shape-creating facilities and experimented with most of the functionality in an explorative way, without getting to the design task itself. At the end, we found a limited number of students were able to submit a finished design (see Figure 3).

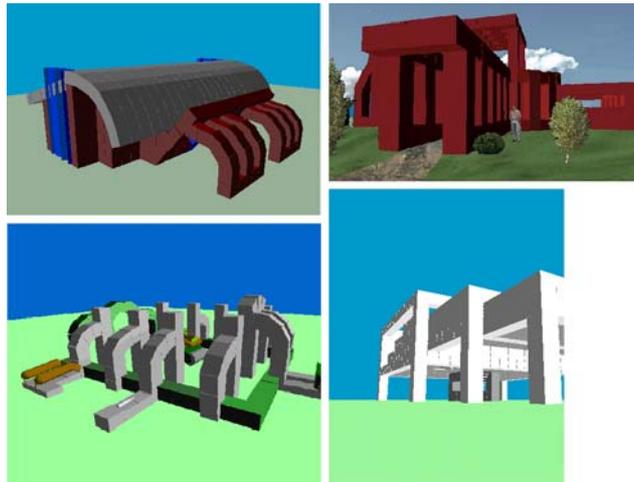


Figure 3. Examples of student work with DDDoolz.

In general, the interface seems to support a fast learning curve. Most students were able to use DDDoolz in the rather short time span that was available for the last exercise and use all of DDDoolz' features, including the less obvious ones such as importing 3DS models.

Next assignments with DDDoolz need to be more clear about the design goal, state some ways of self-assessing the acquired quality of the design, and focus a bit more on the possible added value of such a way of designing relative to the conventional CAD approach as is implemented in AutoCAD.

4. DDDoolz in Architectural Design Practice

An important aspect of the research programme is to see whether the developed concepts and prototypes apply to architects in practice. For this purpose, cooperation with an established architectural firm was initiated, Wiegerinck Architecten Arnhem. DDDoolz is made available for them and one of the staff is working with it. The feedback from the practical experiences then gives valuable information on the practicality of DDDoolz and potential further development.

The purpose of DDDoolz as a sketch tool was quickly acknowledged and the architect identified the added value of direct projection of the work in VR. A work cycle was established where DDDoolz was used complementary to AutoCAD or 3Dstudio. A conceptual model was loaded in DDDoolz and then new ideas were sketched around it, while the walkthrough option provided visual feedback of the influence of the sketches. It was generally agreed that this application had strong benefits over traditional 3D CAAD representations (static renderings).

Also, it was noted that the potential of DDDoolz was not adequately addressed in the prototype. Many additional features were suggested that could increase its strength and efficiency in the early design stages. A number of suggestions considered expansion of the user-interface and navigation in a model. A further developmental step in DDDoolz could be the transition from the sketch by voxels to more complicated shapes that are used in a next developmental cycle of the design process (Figure 4, right).

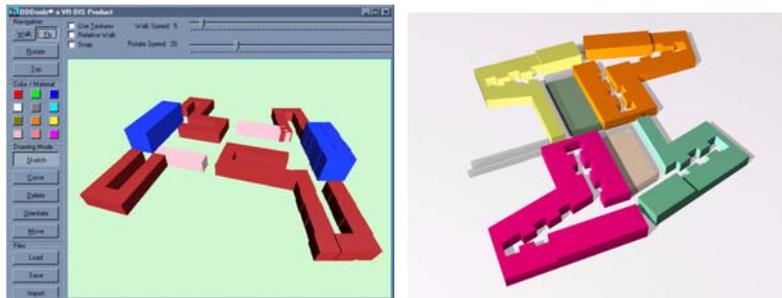


Figure 4. Left: Architect's use of DDDoolz. Right: next development step in 3Dstudio

5. DDDoolz in Research of Design Systems

The first versions of DDDoolz did not feature the icon section in the left part of the screen. One of the reasons for this is that we consider this interface type less suited for a VR application. The interface as it is present in DDDoolz now has been incorporated in the last versions to make the software usable for the students.

As a research tool, DDDoolz serves as a platform for experimentation with various input devices. Mouse control, keyboard input, mouse and keyboard, voice control, flock-of-birds 6D mouse, and pen have been implemented and tested. The various combinations all bear their influence on how the system is used, which is one of the considerations in our research programme. Another link we are experimenting with is a Feature Model of the design and offering a VR-based interface of manipulation of the information model as well as the design in DDDoolz.

At some point, a design in a voxel format needs to be changed into a parameterised geometrical format. This requires some shape recognition ability that will transform sets of voxels into other geometries. Furthermore, when such more complex shapes are introduced, interaction and meaning of design acts need to be redefined and reinterpreted by the system (Wijk et al. 1999).

6. Conclusion

DDDoolz is currently in its first stage of development. It provides a nice platform to teach students some innovative directions of CAAD, and gives the research feedback on the use of VR in the early stages of design. Together with practising architects we hope to expand the functionality of DDDoolz towards a more comprehensive design aid system. The use in education will need to be further developed, in particular with respect to a more integrated and clear design task. DDDoolz can be downloaded from:

<http://www.ds.arch.tue.nl/Research/DDDoolz/>

Information on the first year course can be found on:

<http://www.ds.arch.tue.nl/Education/Courses/7M063/>

Acknowledgments

The main part of DDDoolz was developed in a sabbatical leave of Bauke de Vries at the ETH Zürich in the summer of 1999, in the CAAD group of Maia Engeli. The notion of the Face Orientation Method is based on Kai Strehlke's work (1999). DDDoolz is further developed at the Eindhoven University of Technology with the aid of Joran Jessurun and under continual discussion with the research group of Design Systems. The program is used in first-year education by Bauke de Vries and Henri Achten, and in architectural practice by ir. B. v.d. Moosdijk of Wiegerinck Architecten, Arnhem.

References

- Achten, H. and Turksma, A.: 1999, Virtual Reality in early design: the Design Studio experiences, in J. Verbeke et al. (eds), *AVOCAAD Proceedings of the 2nd AVOCAAD Conference*.
- Achten, H. and Roelen, W.: 1999, Virtual Reality in the Design Studio: the Eindhoven Perspective, in A. Brown and M. Knight and P. Berridge (eds), *Architectural Computing From Turing to 2000*, pp.
- Donath, D. and Regenbrecht, H.: 1999, Using immersive Virtual Reality systems for spatial design in architecture, in J. Verbeke et al. (eds), *AVOCAAD Proceedings of the 2nd AVOCAAD Conference*.
- Gross, M.D. and Yi-Luen Do, E.: 1996, Ambiguous intentions: a paper-like interface for creative design. UIST '96, Seattle, Washington, pp. 183-192.
- Jung, Th. and Do, E. Yi-Luen and Gross, M.D.: 1999, Immersive redlining and annotation of 3D design models on the web, in G. Augenbroe and C. Eastman (eds), *Computers in Building: Proceedings of the CAAD Futures '99 Conference*, Kluwer Academic Publishers, Dordrecht, pp. 81-98.
- Kurmann, D. and Elte, N. and Engeli, M.: 1997, Real-time modeling with architectural space, in R. Junge (ed): *CAAD Futures 1997*, Kluwer Academic Publishers, Dordrecht, pp. 809-819.
- Kurmann, D.: 1998, Sculptor - how to design space? in T. Sasada et al. (eds), *CAADRIA '98 Proceedings of The Third Conference on Computer Aided Architectural Design Research in Asia*, pp. 317-325.

- Leeuwen, J.P. van and Dubbelman, T. and Achten, H.H.: 1998, ICT as a means of education, in Porada, M. and Boutros, N. (eds): *16th eCAADe Conference Proceedings*.
- Stellingwerff, M.: 1999, SketchBoX, in A. Brown and M. Knight and P. Berridge (eds), *Architectural Computing From Turing to 2000*, pp. 491-497.
- Strehlke, K.: 1999, An environment for collaborative three-dimensional modelling over the internet, in J. Verbeke et al. (eds), *AVOCAAD Proceedings of the 2nd AVOCAAD Conference*.
- Wijk, J.J. van and Vries, B. de and Overveld, C. van: 1999, Towards an understanding 3D VR architectural design system, in A. Nijholt and O. Donk and B. van Dijk (eds), *Interactions in Virtual Worlds*, pp. 219-224.
- Yi-Luen Do, E. and Gross, M.D.: 1997, Tools for visual and spatial analysis of CAD models - implementing computer tools as a means to thinking about architecture, in Junge, R. (ed.) *CAAD Futures 1997*, Kluwer Academic Publishers, Dordrecht, pp. 189-202.