

3D Input devices integration in CAD environment

F. Bruno¹, G. Causo², M. Muzzupappa¹

¹ Università della Calabria – Dipartimento di Meccanica, Rende (CS), Italy

² Politecnico di Milano – Dipartimento di Meccanica, Milano, Italy

Abstract

Virtual Reality (VR) technologies are becoming commonly used tools in the product development process, starting from the styling review in the conceptual design phase, until to the Digital Mock-up (DMU) validation in the advanced stages of the design process. What has not yet been sufficiently investigated is the possibility to interact with the DMU directly inside the CAD environment using 3D input devices. Although few CAD systems, like CATIA V5, have an additional module to support VR devices, in most cases it is still necessary to customize the application to obtain the functionalities desired by the user.

The present paper discusses difficulties and advantages related to the integration of VR techniques and 3D input devices in CAD systems. The work has been conducted analyzing and testing the potentialities of the Unigraphics NX3 CAD environment and implementing a software plug-in that allows the user to perform such interaction tasks employing 3D input devices.

Categories and Subject Descriptors (according to ACM CCS):

I.3.7 [Computer Graphics]: Virtual Reality. J.2 [Computer Applications]: Engineering. H.5.1 [Information Interfaces and Presentation]: Artificial, augmented, and virtual realities.

1. Introduction

In a Virtual Reality (VR) environment it is possible to interact in a more natural way with the Digital Mock-Up (DMU). As a matter of fact, thanks to special devices, the user finds himself in a simulated environment in which, through the stereoscopy, he/she can perceive the depth of the scene, having at the same time the possibility of interacting with the virtual product by using his/her hands. Such interaction is possible thanks to the use of special gloves, which have sensors able to detect the bending of one's fingers. By supplying the gloves with tracking sensors able to reveal position and orientation, it is possible to reconstruct the movements of the user's hands in a virtual environment; in this way the user may carry out several types of simulations connected to the interaction with the product.

One of the main problems related to the integration of VR in the Product Development Process is the data transfer from CAD systems to VR environments. Many techniques and tools have been developed to support this crucial phase, but in some cases this process requires an effort and an amount of time that can compromise the advantage in use VR for design tasks. Moreover the generation of the VR models implies several problems:

- The tessellated model employed in the VR environment can be significantly different from the mathematical representation of the CAD.
- Some metadata (constraints, features, material properties, etc.) associated to the model are lost in the conversion.
- A double representation of the model has to be stored, managing the modifications on both sides to maintain the coherency between them.

In order to resolve the problem of data transfer from CAD to VR many researches have been conducted, most to define the techniques and to develop the technologies needed to reduce the time required for the conversion or to preserve the metadata stored in the CAD model. But in our opinion it is interesting to investigate the possibility to employ the CAD systems also in VR environment, taking advantage of the special devices that allow the user to interact with the digital model in a more natural and intuitive way. This approach has been already tried by some CAD vendors, like Dassault System for CATIA V5, that have developed an additional module to support 3D input devices, otherwise most of the CAD systems support only few devices, like the Spaceball or the Spacemouse. Anyway, also with the Immersion module

of CATIA V5, only few interaction tasks, like object manipulation, can be performed.

In the present paper some of the most common applications of VR technologies in the Product Development Process are presented as the state of the art. Then we describe the development of an application that allow users of Unigraphics NX3 CAD system to employ 3D input devices to perform such simple tasks like manipulate the model, make dynamic sections or move the parts in the assembly. The development of this application set the goal to realize difficulties and advantages related to the integration of VR techniques and 3D input devices in CAD systems. Moreover it represents a useful tool to support the collaborative design review sessions in Virtual Environment. As a matter of fact it is very common in automotive industries to employ large visualization devices, like powerwalls, to carry out the collaborative design review sessions, employing the CAD system to visualize the DMU of the car or a part of it. In these cases the people in front of the powerwall ask to a technician at the console to change the view of the model accordingly to their instructions. With our application, instead, users of the powerwall can control the CAD system quickly and easily without the need to have a technician at the console.

2. Previous Work

The idea to employ VR technologies in design has been investigated from over a decade. The first approaches, such as 3-Draw [SRS91] and JDCAD [LG94], were based on normal desktop VR configurations equipped with a pair of shutter glasses for stereoscopic vision and a magnetic tracking device for interacting with the system. JDCAD supports creating 3D primitives and performing Boolean operations. In 3-Draw the user sketches a set of curves and uses them to define the skeleton of a surface, as in a wireframe representation, but there is no possibility of creating a surface with the curves drawn. One of the first modelling applications using immersive Virtual Reality technology was 3DM [BDHO92]. It uses head mounted displays to visualize surfaces modelled by triangle nets.

COVIRDS [DG97], Conceptual Virtual Design System, tries to combine various VR-technologies, such as speech input, gesture recognition as 3D input and output during the conceptual phase and to develop new interaction techniques for creating and modifying free-form surfaces. Modifications can be performed with the so called virtual tools: with a cone tool, for example, it is possible to create a hole in a surface.

In ARCADE [SSD00] and in SpaceDesign [FDMS02] the 3D free-form surface modelling is implemented in different ways, e.g. Coons patches from just one 3D outline stroke, skinned surfaces with immediate visual feedback during creation, symmetric free-

form surfaces by using the pad as a mirror plane, subtractive sweeping, etc..

The VEIS system [BMPR02], [BMLR02] employs a desktop VR configuration to allow users to sketch and model free-form curves and surfaces using 3D input devices. The sketching tools are based on the concepts of extrude, revolve and skin, the modification tool is based on the over-sketching technique [BMLR03].

Other interesting applications concerning the integration of VR technologies in the Product Development Process concern the area of serviceability analyses.

Gomes de Sa and Zachmann [GZ99] have investigated the steps needed to apply VR for virtual prototyping (VP) to verify assembly and maintenance processes. They present some interaction paradigms and functionality that a VR system must implement in order to be suitable for Virtual Assembly. Authors assert that VR/VP does have the potential to reduce the number of Physical Mock-Ups and improve overall product quality, especially in those part of the business process chain where humans play an important role.

Jayaram et al. [JCL97] present a feasibility study for using VR in design for assembly, the design of a Virtual Assembly environment and preliminary results from the use of this environment. Their system uses a HMD, an electromagnetic tracking system, the Cyberglove and the Tactools feedback system that provide a sensation of touching at the fingertips. During the assembly process the user can store the path that was created or reject it and reassemble the part. Collision detection methods warn the user of interference and tolerance problems.

Bao et al. [BJC*02] present a framework and prototype system of an immersive virtual product design (IVPD) which enables users to navigate and interact with the display system by using 3D peripherals. A virtual assembly application of this framework is also presented, it allows the users to identify interferences and design conflicts, check the movement of parts, verify the fitness of interrelated parts and almost eliminate physical prototypes. It is also possible to visualize and interact with every part of a complex design, grasp, move, and rearrange virtual objects.

Ye et al. [YBBD99] compare non-immersive and immersive environments for assembly planning. They present the benefits of VR environments in supporting assembly planning.

3. VR task in CAD environment

All the researches reported in the previous section concern VR applications in which some design tasks are performed in a virtual environment in order to get advantage from the immersive visualization and/or the

natural interaction with 3D input devices. From the developer point of view, this kind of applications can be implemented employing one or more Software Development Kit (SDK) devoted to 3D Graphics (OpenGL, OpenSG, etc.), VR (World Toolkit, VR Juggler, etc.) and 3D Input devices support (Trackd, Open Tracker, etc.). These SDKs allow the developers to reduce the time needed to implement the VR related functionalities so they have to focus their attention on the application specific tasks. In the case of a VR application addressed to design or virtual assembly, the developers need also some specific libraries like a CAD Kernel (Parasolid, Acis, etc.) or a physic engine (Havoc, ODE, etc.).

However, in our opinion, it is possible to follow a different approach to implement a VR application in the area of industrial design. Most of the CAD environments have a SDK that allow developers to implement external or internal program that have access to most of the functionalities of the CAD itself. Such SDKs are often employed in the area of CAD automation to provide new tools to the CAD users or to automate repetitive design tasks. In the present research we have investigated the possibility to employ the SDK of Unigraphics NX3 to develop a VR application directly inside a CAD environment. To reach this goal the first step is the integration of the 3D input devices to allow the user to naturally interact with the 3D models.

4. 3D Input Devices integration in Unigraphics NX3

The tracking system employed is the Ascension Flock of Birds (FOB), a six degree-of-freedom measuring device that tracks the position and orientation of three sensors by a transmitter. Each sensor is capable of making from 20 to 144 measurements per second of its position and orientation when the sensor is located within ± 8 feet of its transmitter. The FOB determines position and orientation by transmitting a pulsed DC magnetic field that is simultaneously measured by all sensors in the Flock. From the magnetic field characteristics measured, each sensor independently computes its position and orientation. One sensor is used to track the head position, so the system can update the visualization point of view according to the user's head movements. Another sensor tracks the position and the orientation of an input device (3D Joystick) supplied with some buttons that can be customized to perform various tasks such as navigation, picking, sectioning, etc.

The 3D Joystick has been built using a commercial joystick opportunely modified. The handle of the joystick has been divided from the base, linking the two parts with a long cable. One of the FOB sensor has been placed inside the handle, so the user can move the 3D joystick in the space to control a virtual hand or to navigate in the scene.



Figure 1: The 3D Joystick

As display device we have adopted a retro-projected screen 1.7 m wide and 1.2 m high, so the user can easily interact with the 3D joystick having also the tracking of his point of view. Since Unigraphics NX3 does not support the OpenGL stereo under Windows OS we have not tested the application with the stereoscopic vision, but perhaps it will be possible with the next release of the software.

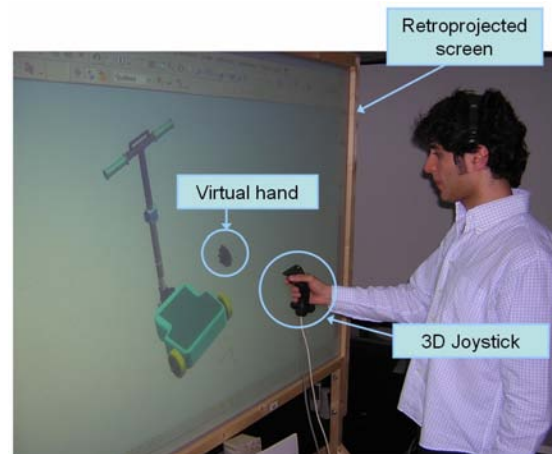


Figure 2: Hardware set-up

Since we need to develop a platform independent application, able to run on different operating systems, with several input devices, the Virtual Reality Peripheral Network (VRPN) library has been employed to implement the device drivers. VRPN provides a device-independent and network-transparent interface to virtual reality peripherals. It is a public domain, open-source software widespread in academia, industry and research labs. The set of supported devices is continuously updated and new drivers can be easily implemented. The

implementation of a VRPN interface in Unigraphics NX3 allows us to quickly change the hardware configuration and also to support input devices connected to other computers present on the network.

The adoption of the VRPN library allows us to employ a PC with Linux OS to manage the FOB while the CAD runs on a PC with Windows OS. The FOB data are transmitted via LAN as explained in figure 3.

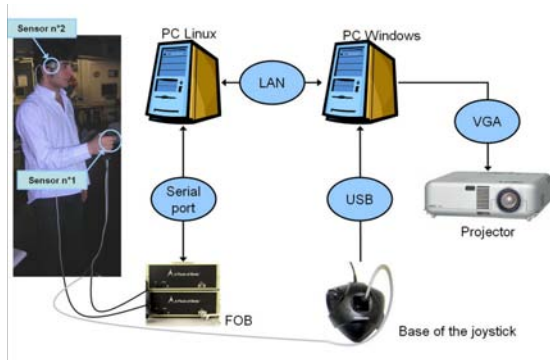


Figure 3: *Layout of the hardware set-up*

5. Reach-ability and accessibility analyses

The verification of the reach-ability of the components involved in the serviceability test is a crucial aspect for designers, especially when the analyses regard the most common service-ability tasks like the substitution of a lamp or the oil filter in the engine bay of a car. These kinds of analyses are efficiently performed in the virtual environment because the use of the 3D input devices makes easier and quicker the manipulation of the virtual hand, compared to the software based on virtual manikins. Besides, the collision detection allows the user to interactively search a correct and collision-free position for the virtual hand. The need to check if there is enough space to perform the maintenance task and if the various locations are easily accessible requires that the size of the virtual hand has to be standardized using appropriate percentiles.

Reach-ability and accessibility analyses are the first tasks we have considered to perform with our application. For this purpose, a component corresponding to the hand of the operator must be inserted in the scene before the user starts to work with the application. A hand model representing the 95% of the population, the so-called 95° percentile, has been chosen. The geometrical figure of the virtual hand has thus been generated from the software Jack 4.0 of the UGS, an application of ergonomic analyses capable of creating models based on statistical findings which considers anthropometric measurements of the human body dimension in a standard static position. Anyway the user can easily change the size and the shape of the hand to perform the same

analysis with several models of the hand, representative of a precise percentage of the human population, obtaining more general and objective results.

The virtual hand is directly controlled by the 3D joystick, so the user can easily move it in the scene, having the virtual hand always in the same position of his real hand. To obtain a precise matching between the real and the virtual hand positions the tracking of the user point of view is essential. As a matter of fact when the user moves his head in front of the screen the position of the 3D scene is automatically updated accordingly with the user head movements, while the virtual hand remains in the same position of the real hand.

5. DMU analysis

The collaborative working is a widely used practice in many design tasks. Automotive industries often employs large visualization devices, like powerwalls, to carry out the collaborative design review sessions, employing the CAD system to visualize the DMU of the car or a part of it. In these cases the people in front of the powerwall ask to a technician at the console to change the view of the model accordingly to their instructions. The integration of the 3D input devices in the CAD software allows users of the powerwall to control the CAD system quickly and easily without the need to have a technician at the console.

To support the DMU analysis our software has three different functionalities. The first one allows the user to move the whole scene with the 3D joystick. At first we implemented a navigation tool that allows the user to manipulate the DMU with the metaphor of the object in the hand: when the user presses a button of the 3D joystick the DMU follows every movements of the 3D joystick. This technique is not very easy to understand for the users and does not allow a precise control. Then another navigation tool has been implemented. It allows to control separately the zoom, the panning and the rotation of the camera.

Another functionality that has been implemented to support the DMU analysis is the dynamic section. The user can activate a section plane parallel to the current view that can be translated with the 3D joystick to see the inner part of the assembly.



Figure 4: *The dynamic section tool*

The last functionality is a tool that allows the user to pick and drag the various parts of the assembly with the virtual hand. When the virtual hand, which the user controls with the 3D joystick, is in collision with an object this is coloured in blue and, if more than one object are in collision, the user can switch among them moving the stick placed on the top of the 3D joystick (*pov*). When the object that the user wants to move has been selected, he/she can drag it by pressing the trigger of the joystick. The dragged object is coloured in red. This tool allows engineers to move some components in order to better analyze the DMU or to plan a possible layout of the assembly.

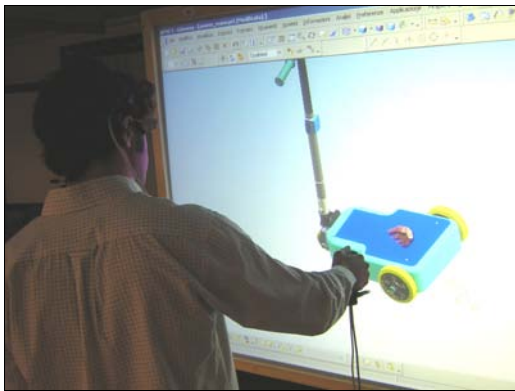


Figure 5: *The user selects a part with the virtual hand*

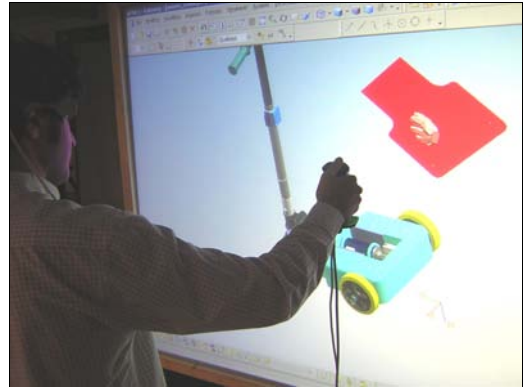


Figure 6: *The user drags a part with the virtual hand*

6. Implementation

The application described in the previous sections has been implemented as a dynamic linked library which runs within Unigraphics NX3. The application interacts with any assembly in the working session.

The main functions of the plug-in are essentially two. The first one makes the whole process start, and this is done by the program only once. The second function, instead, includes all the functionalities of the application which are cyclically called up according to the type of button pressed and to the position of the hand in respect to the other components.

Two VRPN servers are employed to read data from the FOB and the 3D joystick. The reading of the data from VRPN can take place through a function which defines the configuration strings in order to start the two VRPN clients that read data from 3D the joystick and from the FOB. It is interesting to see how simple it is to interact with VRPN server. "Tracker0@192.168.0.2" and "UsbJoy@localhost" actually tell that the FOB (Tracker0) is found on a specified IP address, while the joystick (UsbJoy) is connected locally.

Once the client starts to read data from VRPN server, these must be acquired and formatted before they can be used by all the functions in the program. For this reason a structure, called *vrpn_data*, was created and defined as follows:

```
typedef struct {
    double pos[3];           // position of the hand
    float RotMat[3][3];     // rotation matrix
    int button[4];          // selectable buttons
    float pov;              // value of pov stick
} vrpn_data;
```

As one can read from the comments written on the code, the data are: the position and the rotation matrix

coming from the tracking system, and the states of the buttons set in the joystick.

An important aspect, which was immediately considered on implementation, regards the position of the virtual hand which has to be always coherent with the position of the real hand. To let this happen, it is necessary to avoid that any changes in the point of view done by the head tracking or the navigation tool affect the visualization of the virtual hand. Another function was then implemented to link the view transformation matrix to the data coming from the VRPN client. The result is that the position of the hand of the operator will be always coherent with the hand visualized on the monitor.

Once the input data have been read and formatted, the *UF_ASSEM_reposition_instance* function of Unigraphics NX3 Open API is used to point out a new origin and a new reference system to the specified instance. The instance is the way, used by Unigraphics NX3, to identify singularly an object inside the assembly where it is possible to find more components having the same name. By using this function cyclically, it has been possible to accurately simulate the movement of the hand in the space. In order to improve the aspect of the moving component, the *UF_VIEW_update_view function* has also been used to update visualization.

At each movement of the hand it is necessary to check if there is a collision with any components of the assembly present on the scene. The result of a collision analysis is saved in a summary, an NX Open API structure, and then, on the basis of the number of interferences found, there are three types of operation to be done. If the number of interferences is greater than zero, then the component colour, which is present in the list and saved in summary, turns blue and its original colour is subsequently saved in a variable to allow it to be restored in the future. If the hand collides with more than one part then, by simply pressing button 3 of the joystick, it is possible to select cyclically all the objects present in the list. On the contrary, if no collisions arise and an object had been previously selected, the function restores the original colour to the component.

In order to move the selected component it is necessary to connect the movement of the object with the one of the hand. This has been possible by modifying the hierarchical position of the object in the structure of the assembly. The *link_obj* function has been created and it will start by pressing button 1 on the joystick: the selected component becomes red and it is the child of a new assembly whose "parent" is the hand. The object is therefore removed from the early assembly and the function of collision analysis is inhibited. In this operation it is also necessary to carry out some transformation on the reference systems to keep

components still in the 3D space when a request to change the position in the hierarchy of assembled components arises. The initial hierarchical position and the first reference system must be saved to be correctly restored at the end of any operation regarding movement.

5. Conclusions

At the present, the application functionalities are useful to freely interact with Unigraphics NX3 during the design review sessions that, in the automotive companies, are often carried out using powerwalls. During these meetings, the engineers usually stand in front of the screen while an operator at the control console commands the CAD in order to modify the vision, to bring out sections or to move component parts. With this tool, instead, it is possible to control the CAD with 3D input devices, without the necessity to have any operator at the console. Besides, it is no longer necessary to translate the model in a different file format to be read from the VR system. This reduces considerably the time needed in the conversion of a CAD model so that the operator may use the CAD system also for the design review session at the powerwall, without the need of a specific VR software.

A substantial problem found in the implementation of the plug-in regards how the NX Open API allows developers to control the software. Unigraphics NX3 plug-ins, in fact, are intended to run one task at a time and only at the end of the task it is possible to restart to interact with the CAD Graphical User Interface (GUI). Call-back functions can be used only to customize the GUI. Plug-ins, therefore, inhibits the normal operation of the CAD system, and the user is actually forced to work only with the tools implemented in the plug-in.

From these results we can gather that, in spite of the limitations related to the API, all the potentialities of the application realized so far can be developed and extended significantly. For instance, new supports for the data gloves could be developed to enable the operator to move the fingers of the virtual hand and interact more freely with the objects on the scene. Mating conditions could be introduced to simulate the movement of kinematic mechanism or, however, to avoid the penetration of the objects for more realistic interactions.

Naturally real progresses are still ahead when it will be possible to overcome the actual API limits of the CAD systems. Therefore, if the attention given by the CAD vendors increase towards substantial development of their products, certainly there will be a software in the near future which will have all those features able to help the customer to realize his job in the best way.

References

- [BDHO92] Butterworth J., Davidson A., Hensch S., Olano T. M.: 3DM: A three dimensional modeler using a head-mounted display. *Proc. Symposium on Interactive 3D Graphics*, Cambridge - Massachusetts, 1992
- [BJG*02] Bao J.S., Jin Y., Gu M.Q., Yan J.Q., Ma D.Z.: Immersive virtual product development. *Journal of Materials Processing Technology*. 129:592–596, 2002.
- [BLMR02] Bruno F., Luchi M. L., Muzzupappa M., Rizzuti S.: A Virtual Reality desktop configuration for free-form surface sketching. *Proc. XIV INGEGRAF*, Santander - Spain, 2002
- [BLMR03] Bruno F., Luchi M. L., Muzzupappa M., Rizzuti S.: The over-sketching technique for free-hand shape modelling in Virtual Reality. *Proc. Virtual Concept 2003*, Biarritz, France, 2003
- [BMPR02] Bruno F., Muzzupappa M., Postorino P., Rizzuti S.: VEIS: Virtual Environment for Interactive Sketching. *Proc. 1st Eurographics Italian Chapter*, Milano - Italy, 2002
- [DG97] Dani T.H., Gadh R.: COVIRDS A Conceptual Virtual Design System. *Computer Aided Design*, 29(8):555-563, 1997
- [FDMS02] Fiorentino M., De Amicis R., Monno G., Stork A.: Spacedesign: A Mixed Reality Workspace for Aesthetic Industrial Design. *Proc. IEEE and ACM International Symposium on Mixed and Augmented Reality*, Darmstadt - Germany, September 30 - Oct. 1, 2002
- [GZ99] Gomes de Sá A., Zachmann G.: Virtual reality as a tool for verification of assembly and maintenance processes. *Computer & Graphics* 23:389-403, 1999.
- [JCL97] Jayaram S., Connacher H. I., Lyons K. W.: Virtual assembly using virtual reality techniques. *Computer Aided Design* 29(8):575-584, 1997.
- [LG94] Liang D. Green M.: JDCAD. A highly interactive 3D modelling system. *Computers and Graphics*, 18(4):499-506, 1994
- [SRS91] Sachs E., Roberts A., Stoops D.: 3-Draw. A tool for designing 3D shapes. *IEEE Computer Graphics and Applications*, 11(6):18-25, 1991
- [SSD00] Stork A., Schimpke O., De Amicis R.: Sketching Freeforms in semi-immersive virtual environments. *Proc. ASME 2000*, Baltimore - Maryland, September 10-13, 2000.
- [YBBD99] Ye N., Banerjee P., Banerjee A., Dech F.: A Comparative Study of Assembly Planning in Traditional and Virtual Environments, *IEEE Trans. on systems, man, and cybernetics*, Part C, Vol. 29, 1999, pp. 546–555.