Christian ILIESCU-FEIDYEU, Gabriel George GEAMBAŞU

INTERACTIVE SIMULATIONS FOR DEMOS, EXHIBITIONS AND AS A TESTING PLATFORM FOR DESIGNERS

Abstract: *An interactive simulation is made by combining 3D graphics with the capabilities offered by the simulator in order to obtain valuable results to the design process or for communicating with potential clients through promotion. We can significantly reduce production costs, test products in multiple new scenarios in a short timeframe or to reduce the usage time of dedicated simulation software by using Autodesk Inventor along with Blender 3D. Although the models' fidelity decreases as they're converted for use in simulations, it is enough for solving potential problems that arise during modelling as well as for implementing commercial presentations.*

Key words: *Simulation, Modelling, Computer graphics, Realtime, CAD, Cost reduction.*

INTRODUCTION

The emergence of high performance hardware has enabled the creation of both realtime realistic simulations, as well as on demand. This process involves several steps out of which 3D design is essential. The accuracy of the 3D model can significantly affect the simulation results together with the clients' impression during presentations. It is also desirable to minimize costs in order to afford to improve the products over a longer period of time.

Therefore, we have implemented a realtime simulation with minimum software costs by using two 3D modelling programs without compromising the accuracy of the results or the graphical quality of the presentation. Although the accuracy of results is nowhere near that of those provided by software dedicated to simulations, by using this approach we can significantly lower production costs as the set of additional tests is reduced.

By being able to run 60 physical simulations per second or more, even by changing Blender3D's source code, we can test situations in which most known physical parameters can vary, from the objects' inertial factors (they can be configured separately for either rotation or movement), the friction coefficients of surfaces and gravitational acceleration to deformation factors, masses of objects or even limiting their degrees of freedom [1].

2. APPROACH

We have used Autodesk Inventor for students to model the car. It will be remade in Blender3D for usage in the interactive simulation, while the GIMP software will be used for editing the used screenshots as well as for creating textures. Because of Inventor's vast selection of tools, we have been able to accurately and promptly model the car's body and wheels.

Blender3D is a free software with the help of which we can create advanced, realistic and realtime simulations that provide new ways of perfecting the design or quickly removing deficiencies [2].

Fig. 1 Orthographic view projections.

3. MODELING THE CAR BODY IN INVENTOR AND MERGING OF COMPONENTS

We have used a series of available tools and operators by making blueprints from orthographical projections of the car in Blender3D [3]. Usually, these blueprints would be taken from the technical sketches of the vehicle, but the model for this project has been initially made in Blender3D without using Inventor. The modelling process will be described subsequently.

In order to create the blueprints we have taken screenshots of the front, side, top and back views of the car by using the orthographic view (Figure 1). We have removed Blender's interface and kept the model by using the GIMP's Fuzzy Select tool. The next step is using the projections in Autodesk Inventor as in Figure 2.

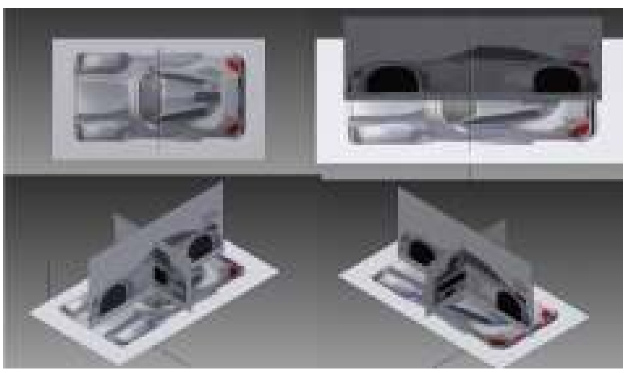

Fig. 2 Use of projections in Inventor.

In order to do this we will add 2D sketches on the XY, XZ and YZ planes. The screenshots will be added inside them while carefully locking their position and dimensions by using fixed elements made in the sketch. Next, by starting from a reference image we have repositioned and resized the other sketches two by two until the projections were accurately coinciding.

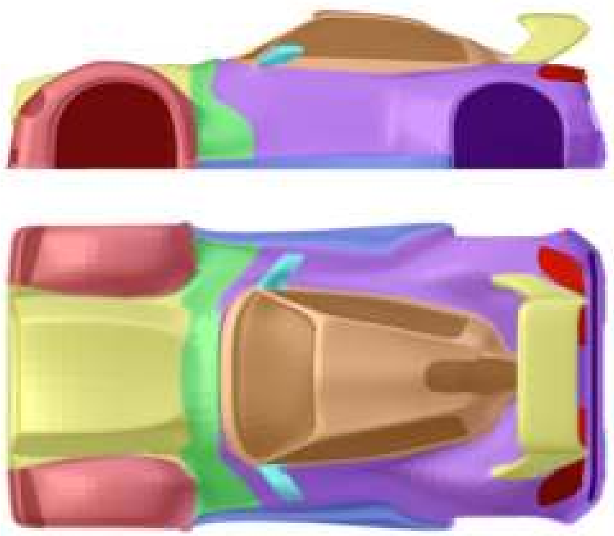

Fig. 3 Car body partitioning.

Before commencing with the modelling, we have analysed the model and concluded that it had to be split into multiple components. The third iteration can be seen in Figure 3 after we have experimentally tried and failed in modelling by fewer components.

They were implemented by mainly using the Loft tool as in Figure 4. With it we were able to create bridges by using 3D sketches that define the exterior bounds guided by rails made with 3D sketches as well. We were therefore able to model 3D surfaces that will be converted to Solids through either Sculpt or Thicken.

Fig. 4 Using Loft to create the purple part.

Loft had to be used several times for some components. The resulting surfaces were merged with the Stitch command and in some cases where the conversion to Solid was impossible we had to also use Patch. For the cuts we have used Extrude Cut as well as Split with surfaces and planes.

The engine's grilles were made by extruding 2D sketches and chamfering them along with the Rectangular Pattern tool and Mirror at the end. For the ones above the exhaust we have used a similar process starting from a 2D sketch made from circles places with Rectangular Pattern that was used to cut into the grilles by using Extrude Cut and finishing with Fillet.

The wheels were modelled by Revolving 2D sketches and using the Extrude and Fillet tools.

The merging consists in selecting the resulting Solid objects and using the Combine tool. Fillet with a variable radius is used afterwards on the components' areas of merging.

In our chosen approach we had decided to model half of the car. In order to finish the model, we have used Mirror together with Fillet or made manual corrections to the imperfect merging points.

4. CREATING THE ASSEMBLY

The vehicle is presented by assembling the modelled components. Two more parts are needed to do this: one made of two perpendicular cylinders for anchoring the front wheels and to enable steering, while the second part is a cylinder. We will also have to add a cylinder to the car body in order to anchor the new components. We have inserted the wheel's part four times, the car body was inserted once, and the two new parts were each inserted twice. The assembly is done after merging the parts with coincident and concentric constraints (Figure 5).

Fig. 5 Final car assembly.

5. REMAKING THE MODEL FOR THE SIMULATOR

Traditionally, this is achieved by using blueprints with the vehicle's projections and afterwards tracing its margins with polygons, edges and vertices. A smoothing operator is used in order to increase fidelity that simultaneously increases the number of polygons. Using the operator requires the reinforcement of sharp edges of the object; otherwise the initial shape is partially lost.

6. MODELING THE CAR

The car was initially made in Blender3D without the aid of blueprints since the design is brand new. The concept combines the curves from cars at the Le Mans circuit, along with the design of sports rally cars and the protrusive shape of the hoods from the American auto industry.

In order to create the general shape we have added several cylinders with the proportions of a car's wheel and afterwards positioned them in a manner similar to that of a vehicle.

Then we have traced the sections using vertices and edges, modelling the car by components in a manner similar to the partition from Figure 3. These sections were extruded and repositioned giving form to the car's body. We have used sections similar to the car's shapes as modelling progressed and added bridges as needed with polygons between the resulting components (Figure 6).

Fig. 6 Blender3D modelling progress.

Afterwards we have used a smoothing tool: Subsurf, which divides the mesh into multiple polygons while partially interpolating the shape of the already existent ones. During the subsurf operator we have reinforced the protruding margins in order to keep the desired shape by cluttering the areas with polygons, edges and vertices. In the end we applied the operator to obtain the final shape (Figure 7).

We have used textured planes with normal maps in order to create the engine and the back protection grilles. These components haven't been modelled in 3D since they will not influence our chosen simulation. The maps were created in Blender using the Red, Green and Blue colour channels in textures of the grilles' 3D models: the intensity of the colours varies depending on how the texture wraps on the 3D Cartesian coordinates of the model's surface.

Most of the textures have an alpha channel as well in order to specify transparency, for example for the trees and the protection grilles. Blender can use the images' brightness instead, but dark textures are harder to use since black specifies transparency.

The wheels were made using the Spin operator on the section through the wheel around its axis of rotation. We have used the Extrude tool for the cuts and reinforced their shape; the Subsurf operator is applied in the end.

Fig. 7 Final car model in Blender3D.

7. MAKING TEXTURES FOR THE SCENE

For this step we have taken photos of the most noticeable elements from the scene, such as grass, the sky, the sun and the asphalt. By using the Make Seamless command from GIMP on specified areas of grass and asphalt we are able to create tillable textures that give the impression of continuity. Afterwards we analysed the pictures and created the necessary colour gradients for the simulation. The sun was created using a gradient changing from the inside to the outside of the circle and finished by applying Gaussian Blur to give the impression of vagueness.

8. CREATING THE SIMULATION SCENE

We had to use a series of tools and techniques in order to achieve a pleasant and ideal ambient for tests. We chose to model a driver test site; its shape was implemented by dividing a plane into multiple rectangles using Subdivide and enabling the Proportional Editing mode to lower the asphalt surface. This mode allows us to have a smooth flow between the asphalt and the hill as it also affects the polygons near the selection depending on how far they are. Multiple smoothing functions can be chosen, such as Smooth, Linear, Random or Sphere.

The street lights were modelled by starting with two extruded cylinders and using Proportional Editing for the shaping. The bumps for the suspension test are modelled by extruding sections made through them in a manner similar to achieving a cylinder.

The sky was created by using a flattened hemisphere and a projection from sideways (from the view perpendicular to the vertical axis) in order to wrap the texture with the gradients made in GIMP. GIMP was also used to extract the trees from the photos, which were afterwards used to texture vertical planes, giving the impression of a forest without sacrificing performance when running the simulation (Figure 8).

Fig. 8 Final simulation scene.

9. SIMULATION PROGRAMMING AND THE PRODUCT'S PROMOTING

Blender3D supports both logic brick programming, which is based on conditions, links and actions, as well as functions written in the Python language [4].

Blender has a module specifically designed for vehicle creation which allows for their rapid setup. The module has functions for adding wheels, positioning them at the desired location, as well as configurable physical parameters such as suspension height, the wheels' friction coefficient and their tendency to roll. Although the realism of the simulation doesn't comply with international safety standards, it can help with obtaining fast results that are relevant to subsequent, more advanced simulations. By using the module together with the joystick for control, we have been able to implement the car's movement together with the wheels' rolling and traction as well as the adjustable suspension height. In order to adjust the headlights and taillights' intensity we have used oriented lamp objects of the Spotlight type at their location, together with textures made in GIMP to emphasize their blinding effect. The adjustment is made using a knob on the joystick which is seen by Blender as a third axis of control. We have animated the lights' intensity together with the size of the textures around the car's lights; the frame of the animation changes depending on the knob's position, giving the impression of lighting with varying intensity (Figure 9).

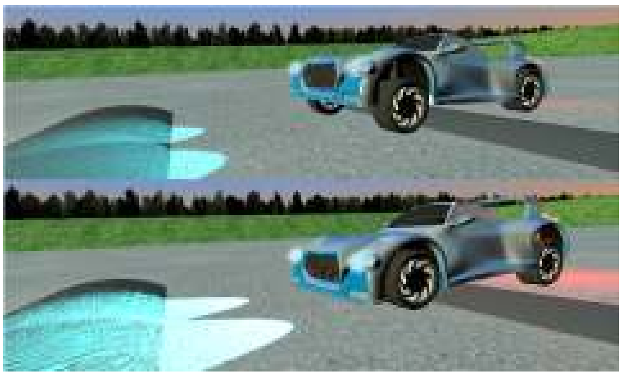

Fig. 9 Testing of lights and adjustable suspension.

Changing the view of the car is achieved by using Child-Parent constraints between the car's body and various cameras placed around it depending on what test needs to be observed. In order to reliably see the wheels' maximum steering angle based on the suspensions' height we had to use a logic brick that enabled us to toggle the visibility of objects such as the ground.

The product's qualities and capabilities can be demonstrated by using a visually appealing scene. This can be easily achieved either by exporting the simulation to an executable for distribution and sharing or by using an internet browser extension known as Burster Plugin – it basically contains Blender3D and can be used for integrating the scene into an online site. As an example, auto manufacturers could increase the number of potential clients by providing easy and fast access to a virtual tour that can impel them to buy a certain car.

10. CONCLUSIONS

In conclusion, industrial designers may use free software such as Blender3D to test their models without buying expensive alternatives. It can be used to quickly find design issues, significantly reducing the number of additional tests required to implement a product conforming to international standards of quality and safety. Also, unlike the other available alternatives, Blender3D has its source code available for free to be used for any purpose, even commercially, thus allowing users to implement any desired functionality that isn't available. Promoting can be easily done without requiring advanced programming knowledge, while the graphics pose no problem to the designers since they already have the required abilities. Therefore, Blender3D and Autodesk Inventor together with realtime interactive simulations prove their worth by offering both simulations on demand, as well as new results of great usefulness.

REFERENCES

[1] Mullen, T. (2012). *Mastering Blender, 2nd Edition*, Wiley/Sybex, ISBN 978-1-4571-4549-0.

[2] Hess, R., Roosendaal, T. (2007). *The Essential Blender,* No Starch Press, ISBN 978-1-59327-166-4.

[3] Felinto, D., Pan, M. (2013). *Game Development with Blender,* Cengage Learning PTR, ISBN 978-1-43545-662-4.

[4] Lutz, M. (2013). *Learning Python, 5th Edition,* O'Reilly Media, ISBN 978-1-44935-573-9.

Authors:

Master student Christian ILIESCU-FEIDYEU, Politehnica University of Bucharest, Faculty of Aerospace Engineering, Department of Engineering Graphics and Industrial Design, E-mail: christian.if39@gmail.com

Assist. Prof. Gabriel George GEAMBAŞU, Politehnica University of Bucharest, Faculty of Aerospace Engineering, Department of Engineering Graphics and Industrial Design, Romania, Email: geambasu_gabriel@yahoo.com