Contents

List of Figures

Chapter 1

DGTk User Manual

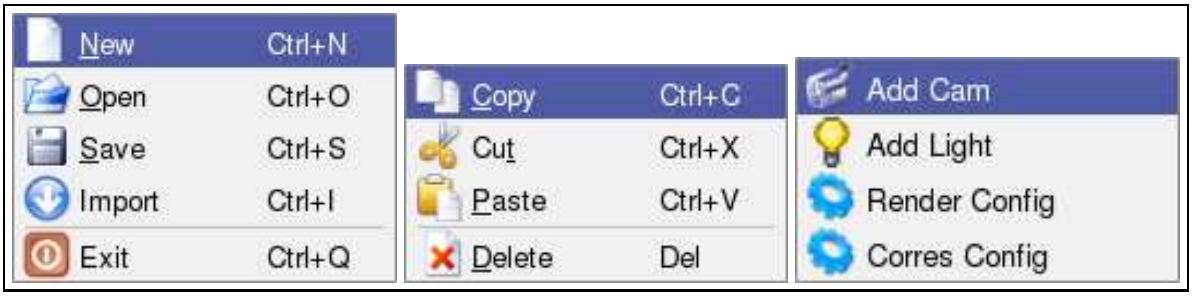
DGTk is a vison aware tool which can generate depth maps, object maps, point correspondences, etc which are very valuable for CV and IBR researchers. We have designed the tool to be very flexible, versatile and easy to extend. The reasons for developing a separate tool for data generation rather than extending already available tools include, (1) Lack of availability of information about internal representation of data in such tools, (2) Very complex and difficult to learn interfaces, (3) Require unintuitive extensions to tools due to their internal architectures. (4) Lack of good documentation for extending such tools. In this chapter we intend to introduce various features of our tool to the users.

1.1 Main Window

The GUI of our tool is similar to that of a standard 3D authoring tool. The main window that appears once the application launches, contains a menu bar and a tool bar. The menu has the standard elements like $File, Edit$ and $Tools$. These three menus provide all the functionality that is needed for the user in terms of data generation. This is another major advantage of using our tool. The user does not have to spend hours trying to figure out which item performs what function. The menu items contain the standard images depicting the operation to be performed by that particular menu item (Figure 1.1).

1.1.1 File Menu

The standard operations like opening an already saved scene, creating a new scene, saving the current scene, importing 3D models in known formats like 3DS, AC3D and MD2 and finally an item to quit the tool. While designing our tool we have designed a simple ASCII file format for saving the scene



(a) (b) (c)

Figure 1.1 (a) File menu, (b) Edit menu and (c) Tools menu. Each operation has it's own standard icon beside the operation.

description. These files generally have an extension ".scene". The user can generate any data he/she requires by opening these scene files.

1.1.2 Edit Menu

The major use of our tool would be for creating complex synthetic scenes which mimic those in the real world. The user may want to place hundreds and hundreds of trees in a scene which are actually a single 3D model file. To make such operations easy, we provide simple editing operations (cut, copy and paste). The user can cut, copy or paste any object that has been imported into the scene. Even the cameras can be deleted or copied. This is particularly useful for creating complex outdoor scenes.

1.1.3 Tools Menu

The tools menu is the most important for the users. Users can add a camera to the scene using $Tools->addCam$ menu item (The menu item with a camera beside it). Adding light is done using $Tools->addLight$ menu item (The item with a bulb icon next to it). $Tools->RenderConfig$ is the menu item that is useful for setting the data to be generated. This menu item brings up a dialog box which looks like Figure 1.1.3. The user can generate different outputs by checking against the required outputs. For example, if the user wants to generate depth maps for the scene setup in the tool, the user should open the render configuration dialog by using $Tools->renderConfig$, then in the controls tab of this dialog box the user has to check against the depth maps option and click on the $Start$ button. This will generate the depth map for the scene. Any number of formats can be generated at a time.

The user can change the tile dimensions by specifying the dimensions in the text boxes provided in the interface. These are then used to render the final image. The default values are set to 640 and 480 which

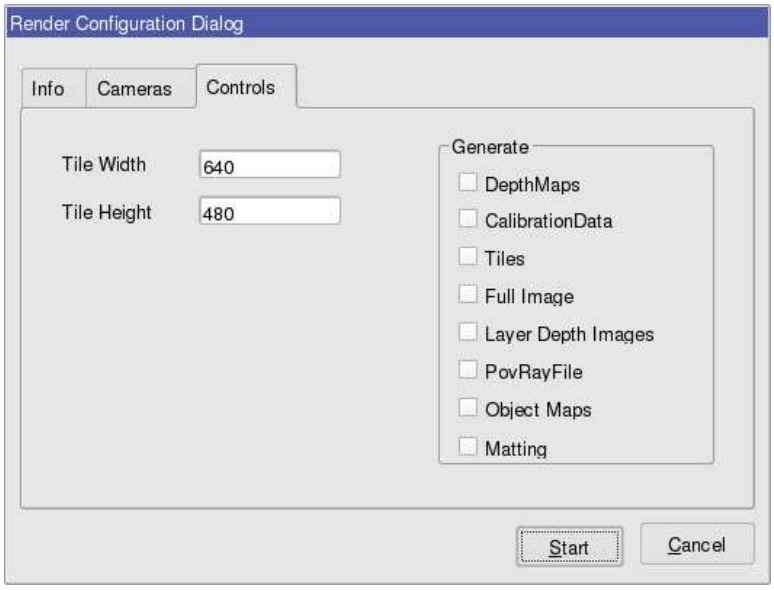


Figure 1.2 The render configuration panel. This is where the user selects what outputs are to be generated by the tool.

are minimum values supported by most of the graphics cards. The resolution of the image to be rendered can be changed by dragging the scroll bars presented in the render configuration dialog box. The user is provided with vital information like the worst case resolution that would be achieved by rendering the image with current dimensions.

The $Tools->addLight$ brings up a popup window which looks like Figure 1.3. The user can specify the exact values required in the text boxes provided in the UI. Once the values are filled, the user has to click on the $SetValues$ button. The user can enable or disable any light by selecting the light from the combobox and checking/un-checking the check box provided. All three color properties of the light Ambient, Specular and the Diffuse can also be specified in this dialog box. Once a particular light is enabled, the user will be able to see a colored sphere in the four views of the tool. This is the graphical representation of the light and can be moved and placed at any desired location just like any other object imported into the tool.

The camera dialog (Figure 1.4) can be accessed by using $Tools->AddCam$ item in the menu. The camera center can be specified in the 3D world cordinates using the text boxes in the UI. Simillarly the look at and orientation of the camera can also be specified. The field of view, aspect ratio, near plane

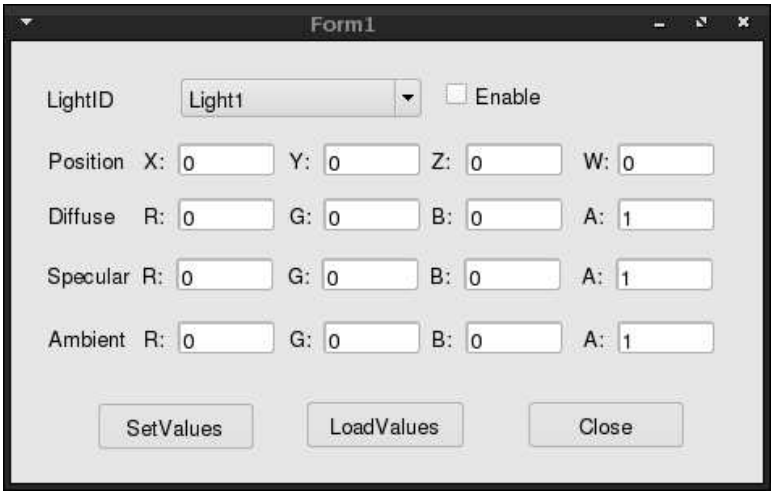


Figure 1.3 The configuration dialogs for placing lights at specific locations

and far plane are also specified using the same UI. The user can directly preview the camera's view by clicking the $Preview$ button. Clicking on the Add button adds the camera with specified parameters to the current scene.

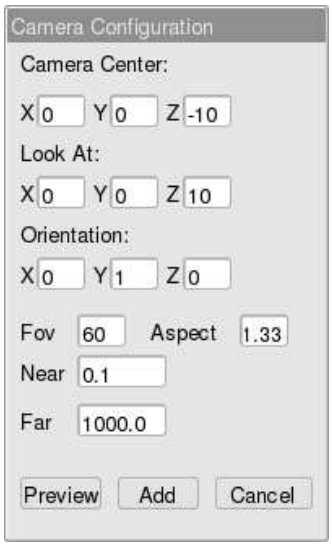


Figure 1.4 The GUI for specifying the exact values for the cameras

1.1.4 Tool Bar

The tool bar in the main window uses the same icons to represent the operations that are presented in the menus. The only operations missing in the tool box are the render configuration and correspondence. The user can access these dialogs only through the menus.

1.2 Creating a Static Scene

For creating a new scene in DGTk, first invoke $File- > New$. This clears all the objects in the scene and resets all the four views. Now the user can start importing new objects into the scene using $File- > import$. Currently the tool allows the user to import AC3D, MD2, 3DS and some simple povray models. If there were no errors while loading the model into the tool, the object appears in all the four views of the tool. The user can now click and drag the object to place it at a desired location. The user can identify which object is currently selected with the help of a green bounding box which appears around it after selection. Once all the objects required in the scene are imported to the tool, the user must add at least one camera in the scene to be able to generate data. There may be more than one cameras in a scene. The user can preview the scene from the camera's point of view using the camera's context menu which pops up when the user right clicks on it. A final check to see if every required object has been added would be helpful before making any further progress. The user must now right click on the first frame in the time line widget and select $MakeKeyFrame$ item. This creates the first key frame of our scene. For creating a static scene the user should now left click on the third frame in the time line widget, then right click and make it a key frame. If the user desires, he/she can save it using $File- > Save$. Now that the scene is setup, the user can launch the render configuration dialog using $Tools- > RenderConfig$ and start rendering the required representations of the scene.

1.3 Creating a Dynamic Scene

Once the first key frame is created after a scene is set (i.e, the first frame in the time line widget is made as a key frame), The user can create a dynamic scene quite easily. We shall show how to create a dynamic scene which has simple linear motion of one or more objects. Now that the first key frame is set, left click on a frame some ten frames away from the first frame. Move the objects in the scene by clicking and dragging them to a new position. Right click on the same frame again and make it a

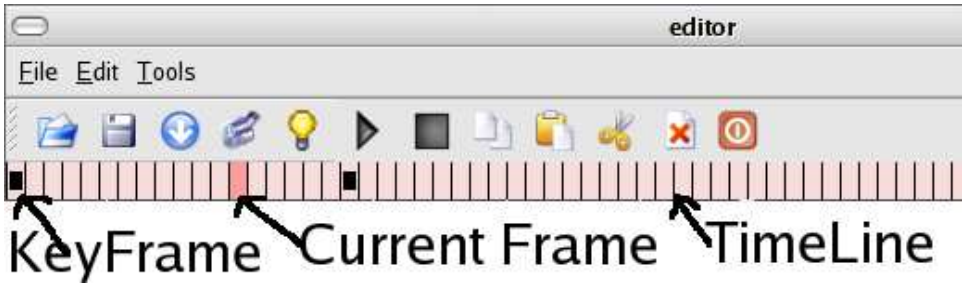


Figure 1.5 The time line that is to be used for creating dynamic scenes.

key frame. This results in a scene which is nine frames long and has moving objects. The user can also create dynamic scenes with rotating objects. For this, instead of dragging the objects, right click on the objects, select the $rotate$ item in the context menu and move the move a little to see the rotation of the object. Once the object reaches the desired orientation, make the second key frame. This process can be used to create as many key frames as required. The important point to note while creating a scene is to import all the required objects before making the first key frame.