

Chapter 5. Haunted House – Modeling of the Scene

Welcome to the scary project!

In this chapter, we will model a haunted house that we will texture and render in the future chapters. You will use the modeling techniques that we have already seen in the previous chapters and learn some new techniques using some useful modifiers and time-saving tools. Moreover, you will learn how to correctly organize your scenes by grouping objects and placing elements in layers. Now that you have more experience with Blender, we aren't going to show you all the steps in detail but rather describe the key points of the process. If you have any difficulties, you can always go back to [Chapter 2](#), *Robot Toy – Modeling of an Object*, and [Chapter 4](#), *Alien Character – Creating a Proper Topology and Transferring the Sculpt Details*, in order to review some of the modeling techniques. Let's start our scene! In this chapter, the following topics will be covered:

- Modeling on scale
- Blocking the house
- Advance modeling tools
- Modeling with curves
- Organizing the scene

The final haunted house should look like the following screenshot:



Blocking the house

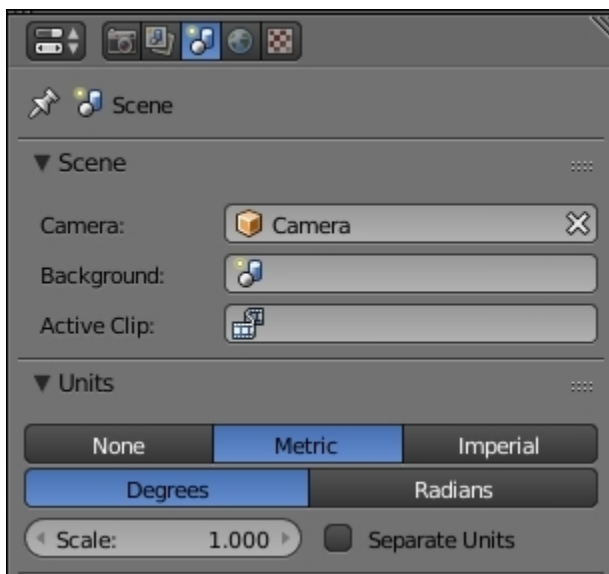
Before going into detail, we will start by testing different shapes in order to create the concept of our house. It is like a 3D sketch.

Working with a scale

In order to create our haunted house and its environment, we need to work with a real world scale. Indeed, when you are working on objects, such as buildings, where the scale matters, it is important to remember to adjust the units of measurement of Blender.

Blender uses, basically, its own unit of measure: the **Blender units** that correspond to a fictitious unit of measure. You aren't going to encounter Blender units in the real world.

There are two other unit systems of measurement in Blender that you can use: the metric system and the imperial system. We prefer the **metric system**. For this, go to the Properties panel on the right-hand side of the user interface under **Outliner** (in the default layout). In the **Scene** tab, you will find the **Units** tab. Choose **Metric** and **Degrees**.



The metric system allows us to work in kilometers (km), meters (m), centimeters (cm), millimeters (mm), and micrometers (μm). Let's choose meters in our case. For this, we set the **Scale** value to **1.000**. A value of **0.1** would make us work in centimeters.

To know the size of your 3D models, in the **Object Mode**, you can look at their size in the **Transform** tab on the right-hand side panel of the 3D viewport (*N*). This information is also given in the **Dimensions** section for the *x*, *y*, and *z* axes.

You can then display the size of the selected edges. In the **Edit Mode**, under the **Transform** tab, go to the **Mesh Display** tab, and check the **Edge Info | Length option**. If you want to measure something, Blender gives you a ruler under the **Grease Pencil** tab in the left 3D view panel (*T*). To use this, simply click on the **Rule/Protractor** button and drag it in the 3D view.

Be careful to always apply your scale and move or rotate the transformation of your objects when you manipulate them in the **Object Mode**. To do this, we open the **Apply** menu (*Ctrl + A*) and select **Rotation and scale**. It is important to avoid involuntary deformations after this.

Blocking the bases of the house

To make this house, we don't start from concept art but from an idea and a few references found on the internet.

It is very important during any creation to spend a little time documenting to confront the different possibilities of shapes and styles. We need to see what has been done previously and be informed enough to be precise in our work.

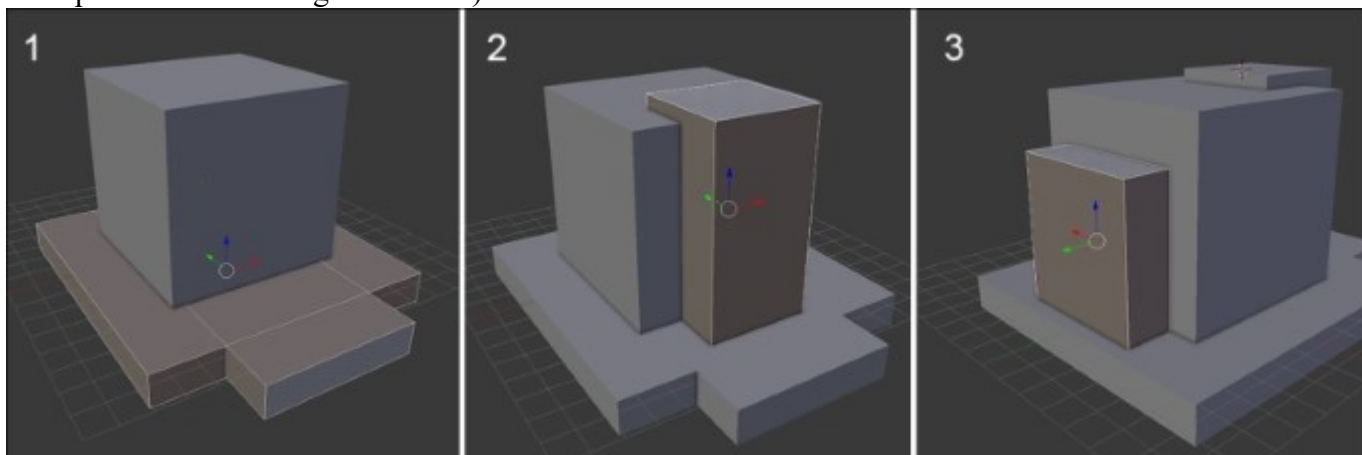
As we are not completely sure of the form as a whole, we will adopt a method that involves testing and quickly developing ideas with simple forms. This method is called **Blocking**. This is done as follows:

1. We will begin the modeling by adding a cube at the center of a new scene (*Shift + A* and select **Mesh | Cube**), which will represent the central part of the house.
2. In the **Object Mode**, we will adjust the size in the **Transform** tab to have something realistic. It is an imposing house, so we will set 7 m on the z axis, 7 m on the x axis, and 8.5 m on the y axis.
3. In the **Object Mode**, let's duplicate our cube (*Shift + D*) in order to make the terrace. So we will scale it to a height of 1.26 m, then we will place it at the base of our haunted house under the main block previously created.
4. The terrace is not completely cubic. We will add two edge loops and an extrusion to the front, which is less wide (on the x axis) than the main block.

It is necessary that this terrace is large enough to be credible, so we will create a passage of at least two meters wide. It is not necessary to be very accurate for the moment, but be aware of your measures, and remember to apply the transformations when you switch back to the **Object Mode** (refer to step 1 in the following screenshot).

5. To improve the general shape of our house a little, we will add a new cube that fits in the front of the central part of the house, centered on the x axis.

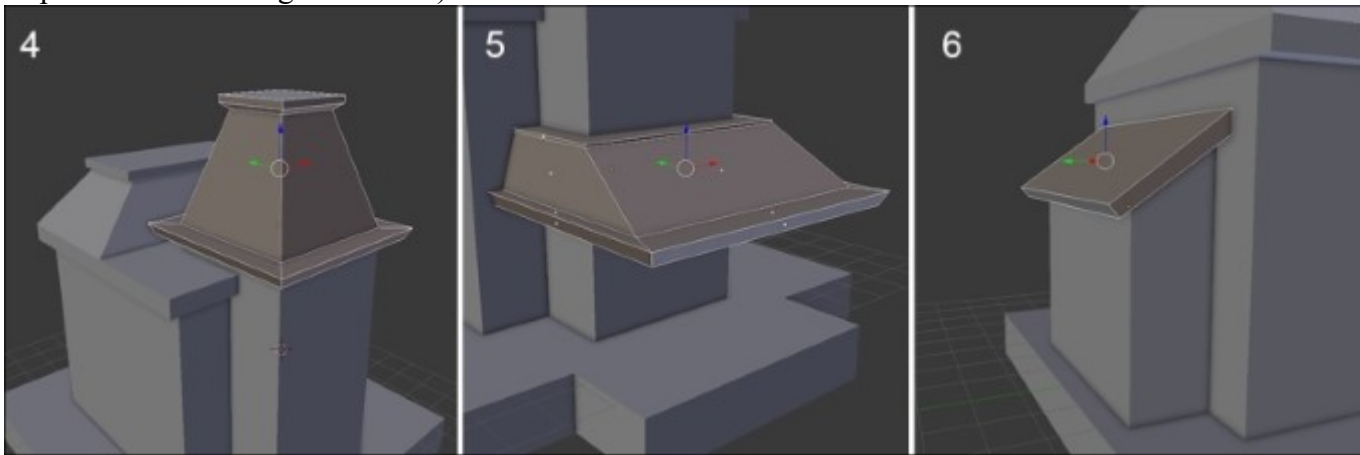
The height of this cube exceeds the height of the other cube by one third. The rest is hidden in the central block. It has a square base, and it is higher than the main block by about 45 cm (refer to step 2 of the following screenshot).



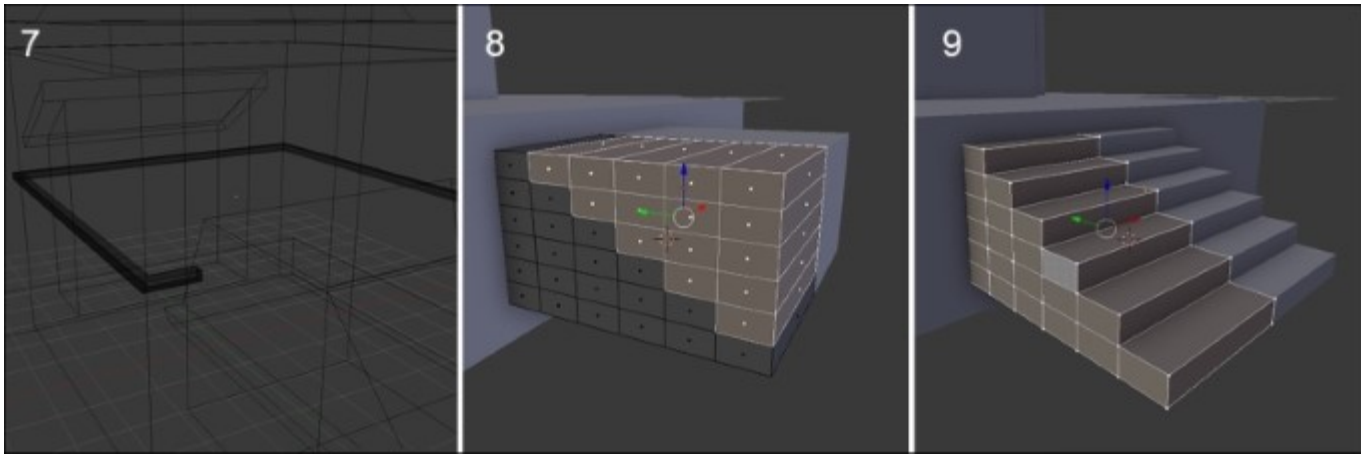
6. Likewise, on the rear part, we will duplicate our front block (*Shift + D*) and move it to the other side on the *y* axis. This block is lower than the main block. Its size is 4.7 m on the *x* axis, 1.67 m on the *y* axis, and 5.3 m on the *z* axis.

Now we have our basic volumes. We can now make the roof that is composed of several parts; one for each block. This is done as follows:

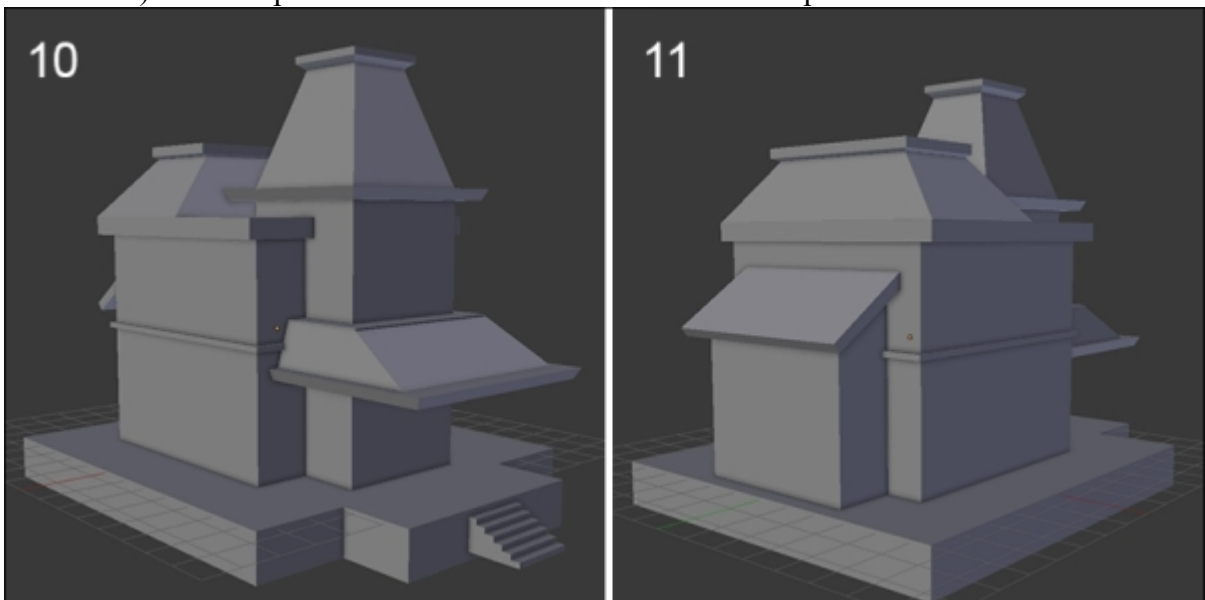
1. We will begin with the roof of the front block by adding a new cube (press *Shift + A*, and select **Mesh | Cube**). We need to adjust its size to be larger on the *x* and *y* axis.
2. We will move down the top face to flatten it. It is the bottom part of this section of the roof.
3. Then we will do an inset (*I*) and an extrude (*E*) to the top. We will adjust the scale of the top, and then we will add two extrusions (*E*) to make the top thicker and finish the shape (refer to step 4 of the following screenshot).



4. We will duplicate this part of the roof (*Shift + D*), and we will place it on the rear half of the central block. We will scale it on the *x* axis (press *S + X*) in order to have the same width as the central block. This is also lower than the roof of the front part, so we will also scale it on the *z* axis (*S + Z*) (refer to step 4 of the following screenshot).
5. In the same way, we will make another roof that covers the front portion of the terrace. It will be supported by pillars. We will again duplicate our roof that is cut in half, and we will adjust it according to the dimensions of the front of the terrace. We will remove the top part to form a small balcony (refer to step 5 of the preceding screenshot).
6. For the roof of the rear block, we will slightly change the style with a simple tilted platform. We will change the rear block to bevel it. We will duplicate the top face (*Shift + D*) and make a new object with it (press *P* and select **Selection**). We will need to make an extrusion on the *z* axis to add a thickness, then we will adjust the size and the position of the wireframe in the **Shading Mode** (refer to step 6 in the preceding screenshot).



7. We will now mark a boundary of two floors with a concrete ledge. For this, we will need to add a new cube (press *Shift* + *A* and select **Mesh** | **Cube**) scaled on the *y* and *x* axes (*S* + *Shift* + *Z*) to be around the main block. We will give it a height of 15 cm and make it exceed the block by about 20 cm (refer to step 7 in the preceding screenshot). We will do an inset (*I*) on the top and bottom faces, then we will delete the nonvisible faces.
8. Let's form the stairs. We will add another cube, then we will resize it to be 84 cm on the *z* axis and 1.5 m on the *y* axis. We will need to divide it into six equal parts horizontally and vertically. In order to gain time, we will add a **Mirror modifier** (refer to step 8 in the preceding screenshot). We will remove the unwanted faces, and then extrude the contour of the top towards the symmetry axis to create the missing faces (refer to step 9 in the preceding screenshot). We will place our stairs in the middle of the front part of the terrace.



These few simple 3D models done in a short time gives us an idea of what our haunted house will look like with further modeling.

Refining the blocking

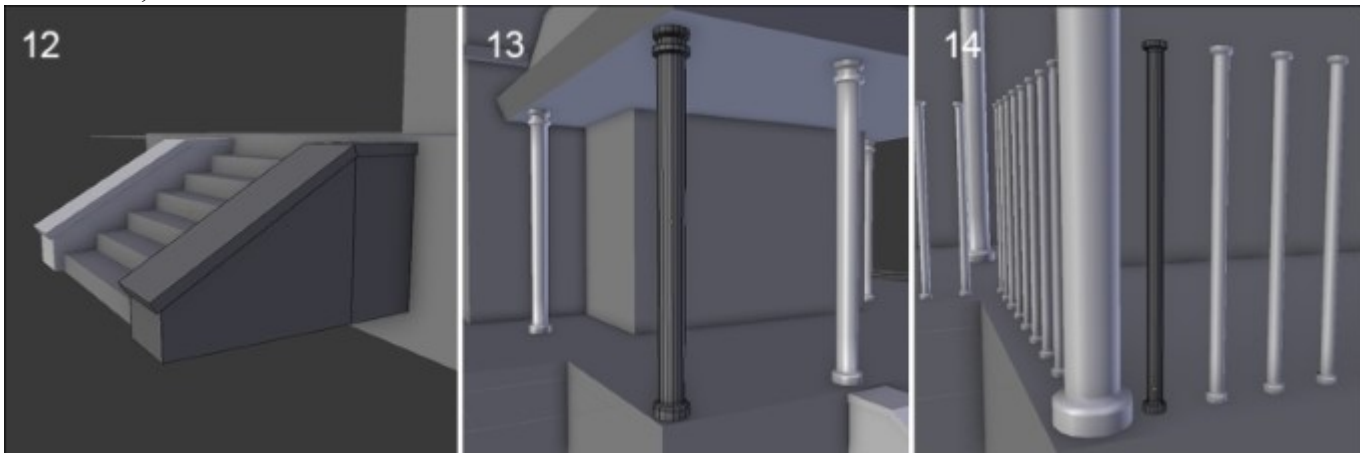
Now that we have the foundations of our model, we will go into the details by adding more defined objects.

Adding instantiated objects

If we analyze the majority of the houses, we can see that they are mainly composed of repetitive shapes such as windows and doors.

So we will use the techniques that allow us to duplicate objects by instance. This means that if we change the geometry of the source object, all the duplicates will change too. As you may have understood, this is really useful in order to save time: for instance, with UVs. Now, perform the following set of steps:

1. Let's start with the low wall around the steps. We will add a cube that we will orient with the slope of the stairs.
2. We will add an edge loop in order to break the slope. Then we will add a thinner piece that recovers the slope. To do this, we will extrude the top faces and scale them appropriately on the same level, and we can redo an extrusion.



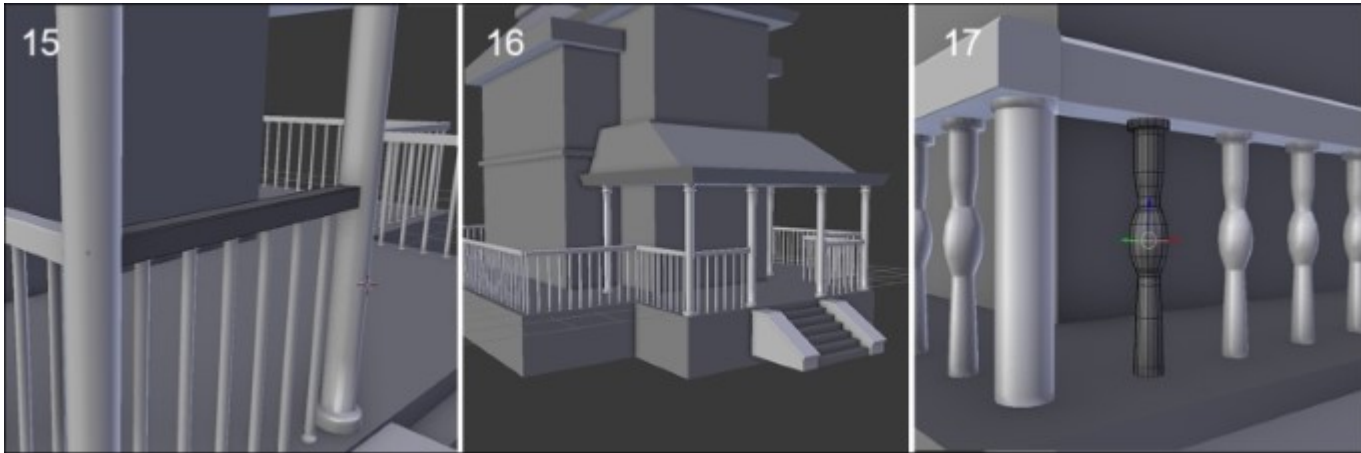
3. In order to mirror the other side of the low wall, we will center the pivot point of the stair object at its center (*Ctrl + Alt + Shift + C* and select **Origin to Geometry**). Now we can safely add a mirror modifier with the stairs as the mirror object (refer to step 12 of the preceding screenshot).
4. Next, we will do the columns that will support the roof that covers the front of the terrace. We will need a 16-face cylinder that is 18 cm wide with a height of 2.8 m. In the last tool options in the left 3D view panel, we will choose the **Nothing** option under **Cap Fill Type**. We will then position it on the left-hand side of the stairs, and we will duplicate it as instances with the **Duplicate Linked** (*Alt + D*) function.
5. In the **Object Mode**, we will place our columns at the four corners of the terrace roof. In order to add some details to the columns, we will add some loop cuts (*Ctrl + R*) on the top of the roof and extrude the face loops along the normals (*E + Alt + S*) (refer to step 13 of the preceding screenshot).

Note

Duplicate Linked

This tool allows you to duplicate your 3D model as instances. This means that when you do a modification in the **Edit Mode** on the source object, the transformations are applied to the other duplicated objects in real time. The UVs are also instantiated. However, when you manipulate the object in the **Object Mode**, the changes are not reflected in the other instances.

In order to break the instantiation link, we can use the **Make Single User** menu (**Call Menu** (press *U*) | **Object and Data**).

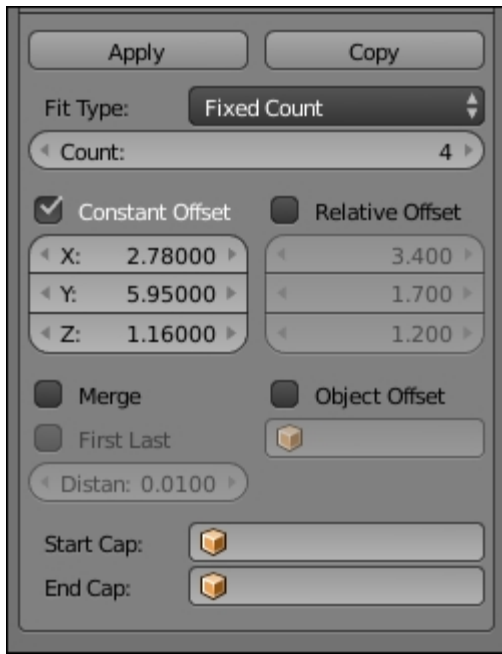


6. We will now work on the bars that delimit the terrace. We will take a new cylinder, this time thinner with a radius of 5 cm and a height of 1.2 m. We will again remove the caps that are pointless here.
7. In order to duplicate our 3D object, we are going to use the Array modifier. We will use a relative offset of 3.100 on the x axis. We will take advantage of the replication of the array in order to improve the shape of the bars a little bit with some loop cuts and extrusions (refer to step 14 of the previous screenshot).
8. Since the bars are along a straight line, we will duplicate them with a normal duplicate (*Shift + D*) to place them on each side of the terrace. We will also need to adjust the number of bars to match the surface of the terrace.

Note

The Array modifier

This modifier allows you to duplicate your 3D models with a customizable offset. You only need to choose the number of repeated objects that you need with the **Count** parameter and the distance of the offset (constant or relative) on any axis. You can also automatically merge your duplicated polygons with the **Merge** option.

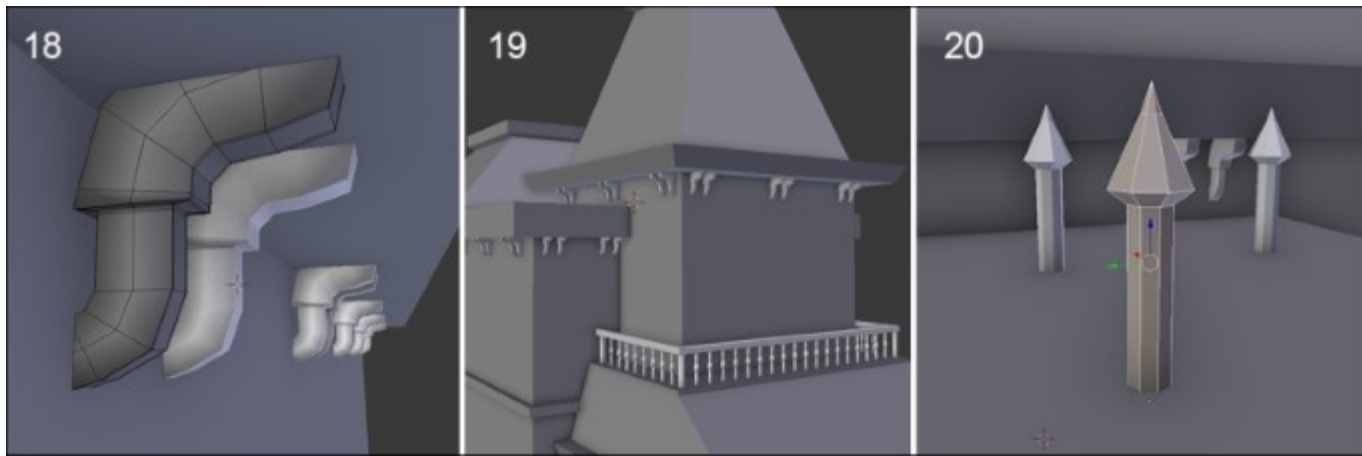


If you want to modify the geometry of a mesh, the array takes the object volume into account, so be careful. A transformation in the **Edit Mode** can change the offset.

9. We will complete this with ramps. The ramp is a simple cube scaled on the x axis to make it longer. We will duplicate the ramp as an instance ($Alt + D$) wherever needed, but remember that you need to duplicate with $Shift + D$ if you want to do some changes in the geometry (refer to steps **15** and **16** in the previous screenshot).

The Duplicate Linked tool is very useful, but it is not very flexible when we only want to do a transform on certain objects.

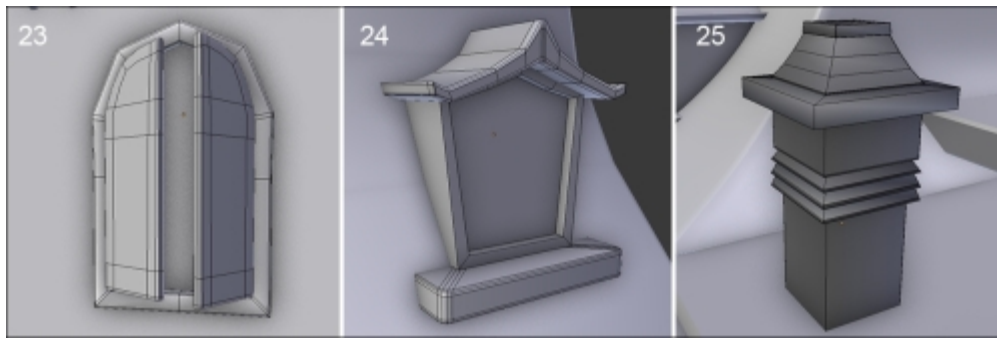
1. Let's repeat the same technique to make a balcony railing on the top of the roof that covers the front of the terrace. We are going to change the shape of bars a little (refer to step **17** of the preceding screenshot).
2. We will also use an **Array modifier** to make the wall brackets that will support the different parts of the roof. We will use one Array modifier to make a pair and another to duplicate it with a good offset (refer to steps **18** and **19**).



3. The same thing is done for the pikes on the roof that give a threatening look (refer to step 20 in the preceding screenshot).



4. The walls are a bit flat at the moment, so we are going to model a bay window with a particular shape (refer to step 21 in the preceding screenshot). We will start with a new cube (press *Shift* + *A* and select **Mesh | Cube**), and we will resize it on the *x* axis to form the base.
5. We duplicate it to make the top part and add an inset for the frames of the windows. When this is done, we will duplicate it as a new instance (*Alt* + *D*), and we will place it on both sides of the main block of the house.



It is also time to make a few conventional old-style windows. We will start with the windows on the front of the house (refer to step **22** in the previous screenshot). Let's start again from a cube:

1. We will delete the left side to use a **Mirror** modifier. From the front view, we will make the shape of the window with a few edge loops (*Ctrl + R*) and add an inset. From this model we will can make the frame and extract the shutters (refer to step **23** in the preceding screenshot).
2. We will create another window model for the roof. We will start its base with a flattened (*S + Z*) and beveled (*B*) cube. The central part is an extruded edge with an inset, and there is a little roof with a curved slope (refer to step **24** in the preceding screenshot).
3. Let's add a fireplace (refer to step **25** in the preceding screenshot). This is also fairly simple to model, starting from a cube that is extruded and scaled. It uses the same basic modeling techniques that we have previously covered.

Reworking the blocking objects

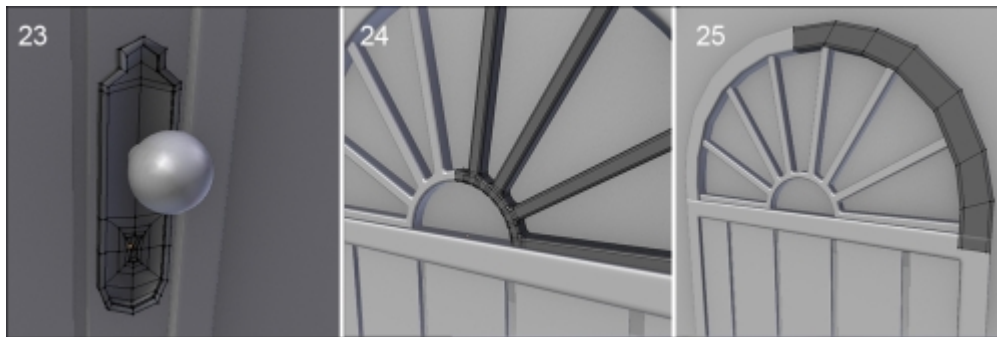
Until now, we had quickly added a series of 3D objects to form a fairly rudimentary house and have a general idea of the entire model of the house. Now we will finish the modeling and review the topology of our models in order to make them more presentable.

There is still a final important object to be made: the front door. It is composed of several 3D objects. There is the frame, door, lock, handle, and there is the top frame with a decorative pattern. The frame and the door are quite similar to the windows. The door will be created as follows:

1. We will use a mirror modifier each time. The lock is quite simple, just an extruded cube with a few **insets** (*I*) (refer to step **23** in the preceding screenshot). For the decorative pattern, we will use a circle in order to get a better round shape.

The rays are extruded and moved by hand. Don't forget to add a few edge loops when you add the **Subsurface** modifier (refer to steps **24** and **25** in the preceding screenshot).

2. We will place the address number of the house. For this, we will use the **Text tool** (*Shift + A* and select **Text**). We will use the basic **Bfont** font of Blender, but you can download and use any font instead. We will use the **Extrude: 2.3 cm** and **Depth: 1.2 cm** parameters to make a small bevel. We will make two small nails to hold them.



3. We will now review every asset and remove the nonvisible polygons that are useless in our case. Instantiating many of our objects has greatly facilitated the work.



4. We will also often add a **Subdivision Surface** modifier when it's needed.

But sometimes, this is not the case, as a simple bevel will do the job as well. As always, you need to maintain your geometry with loop cuts or bevels. In both cases, we want to activate the **Smooth Shading** (in the left panel of the 3D viewport).

5. We will also detach some parts of the geometry from other objects. To do this, we will go in the **Edit Mode** and we will select the faces that we want to detach by pressing *P* and selecting **Selection**.

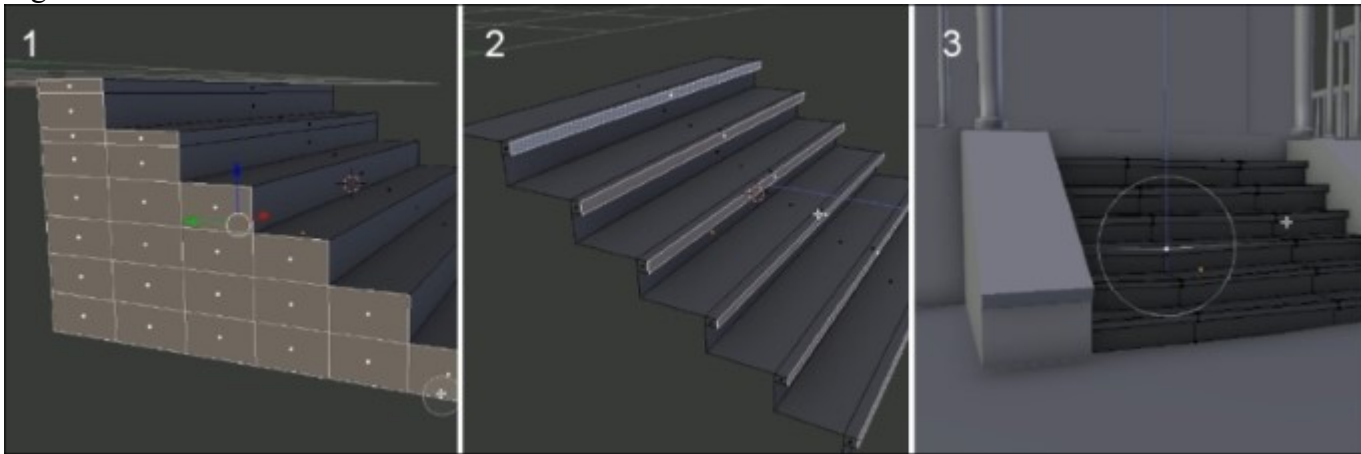
This will allow us to rework certain objects and add the **Subdivision Surface** modifier with ease. For instance, for the roof, we will detach the center part and add tight edges near the four corners with the **Bevel** tool. We can also increase the windows' resolution by adding and placing some edge loops to form a nice round shape in the corners.

All the improvements that we've done here are, in our case, pretty easy because we don't have to get a realistic result. We are taking a more "cartoony" path, and in the next chapter, we will paint textures by hand in order to add more details.

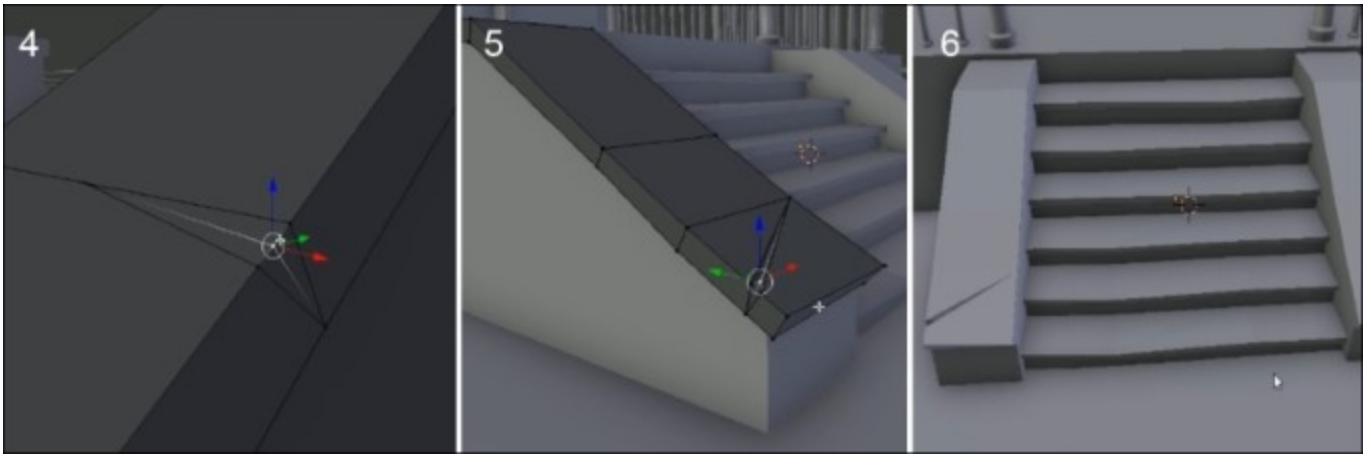
Breaking and ageing the elements

It's now time to refine some elements in order to give them an old/destroyed look. This will mainly come from the texturing work that we will do later, but we can still work on the geometry a little bit:

1. We will begin by working on the stairs. We will delete the side faces that are invisible. (Refer to step 1 in the following screenshot.)
2. Let's add loop cuts below the angle of each stair. This will allow us to extrude a face in the front direction of the top of each stair. (Refer to step 2 of the following screenshot.)
3. We will then add two edge loops at the center to add more resolution.
4. Now with **Proportional Editing** turned on (*O*), we can select some vertices on the stairs and move them to break the shape a little bit. (Refer to step 3 of the following screenshot.)
5. We will now enter the stair ramps object and delete the one on the left. We will have to delete the invisible faces, those that are in the ground and in the house.
6. After this, we can move the origin back to its center (*Ctrl + Alt + Shift + C*) and duplicate the object (in the **Object Mode**) with *Shift + D* in order to move it back to the left-hand side.
7. We can then add definition to the object on the right-hand side with the Loop Cut tool.
8. At the top of the ramp, we will use the **Knife** tool and cut around a loop cut that we had made before. We will then push this hole inwards. (Refer to step 4 in the following screenshot.)
9. We can then repeat this process elsewhere on the other object. (Refer to step 5 in the following screenshot.)
10. The stairs are now finished. (Refer to step 6 in the following screenshot.)
11. We can also add a cut in the back wall of the house and move the vertices a little bit to add some sag.



We now encourage you to go over each object and slightly move the geometry, with the **Proportional Editing** tool turned on (*O*), in a random manner in order to break each object's silhouette. You can also rotate the objects, such as the curtains. This is a house that is not new, plus it's probably abandoned!



Simulate a stack of wooden planks with physics

We are now going to add little bit more details by adding a stack simulation of wooden planks in the front of the house. But instead of placing each plank by hand, we will take advantage of the physics engine of Blender that will do the job for us. In order to keep our stack simple and manageable, we will use the instancing principle that we've seen before. Let's start with the modeling of the plank:

1. The plank will be very easy to model by starting with a cube primitive.
2. We will then rescale this cube to make it thinner and larger. Note that if you do this in the **Object Mode**, you will have to apply the scale with *Ctrl + A*.
3. In order that the plank catches the light on its edges better, we will add a small bevel to it. There are two ways of adding bevels in Blender. The first one is the one that we've already used before with *Ctrl + B*. The other method is by adding a **Bevel** modifier. That's what we've done here. Note that if you want, you can apply the modifier too.
4. We will now duplicate the planks by instancing them (*Alt + D*). You need to place them one on top of the other. In order to add a little bit of randomness, we will rotate them slightly in an unordered way.

Note

About the Bevel modifier

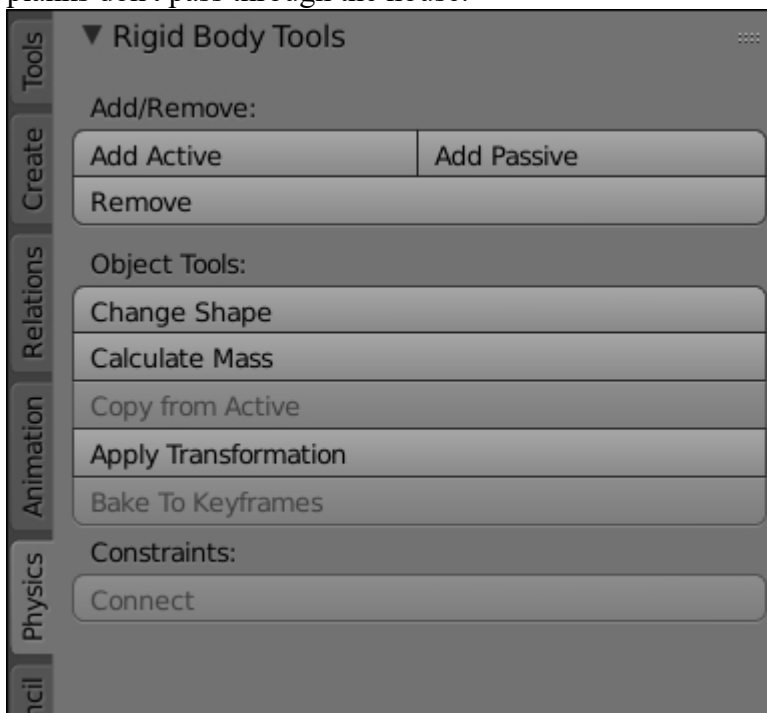
The **Bevel** modifier is nice because it is applied on the whole object, so we don't have to manage a lot of geometry while we are in the **Edit Mode**. We can adjust the **Width** slider to tighten or enlarge the effect of the bevel. The **Segments** option allows us to choose the number of cuts the bevel will be made of. The **Profile** of the bevel corresponds to the direction of the bevel; if it's negative, it will go inwards.

Creation of the simulation of a stack of planks

We will now create our simulation. In any rigid body simulation, the objects have some properties that define them. For instance, you can set their mass, velocity, or simply let gravity act on them as the sole force. In any decent physics engine, you can have static and dynamic objects. A static object is an object that, as its name implies, can't move at all but will be considered in the simulation when collisions occur.

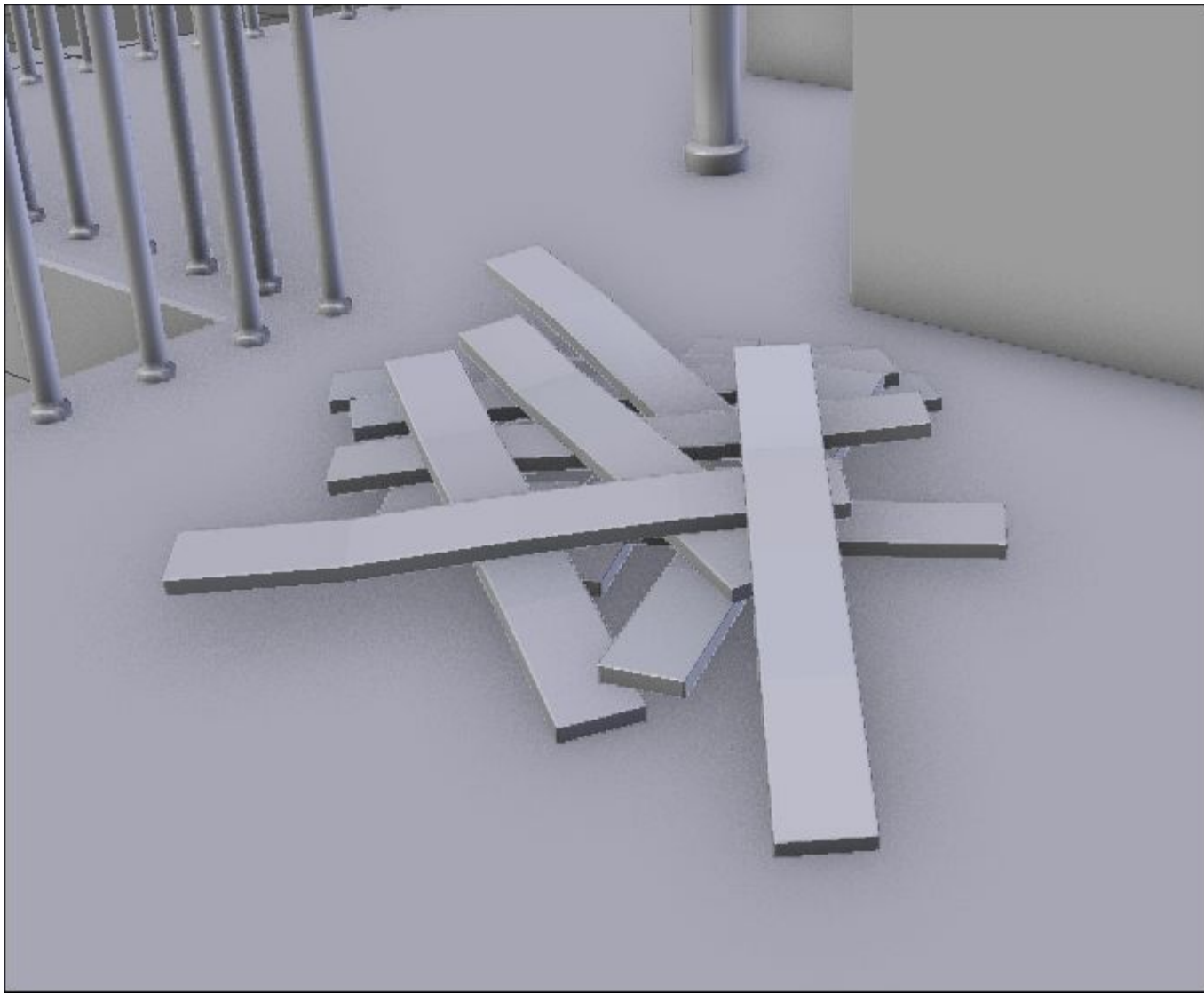
A dynamic object is an object that can receive forces. In Blender, static objects are defined as **Passive** and dynamic objects as **Active**.

1. We will select all our planks, and in the **Physics** tab, in the left 3D view panel (*T*), we will press the **Add Active** button. They will have a green outline.
2. Now we will set the ground object as passive by pressing the **Add Passive** button so that the planks don't pass through the house.



The Physics tab

3. In order to simulate our stack, we will launch the animation. To do this, we will use the *Alt + A* shortcut or press the **Play** button of the **Timeline** editor. Note that if the simulation doesn't seem to launch, you can replace the Timeline bar on the first key frame.



The final stack of wooden planks

4. As you may have seen, there is an orange line on the Timeline that tells us that a simulation has been cached, but as soon as you go backwards in time, the simulation will be removed. So after the simulation has been completed, we will have to apply the placement of each of the planks. In order to do this, we will select them and click on **Apply Transformation** in the **Physics** tab. We can now safely replace the Timeline bar at the first frame, and our stack will rest still.

Modeling the environment (8 pages)

Now that we've finished the house modeling, we will improve our scene with an environment composed of a cliff, a barrier, a cart, and some rocks.

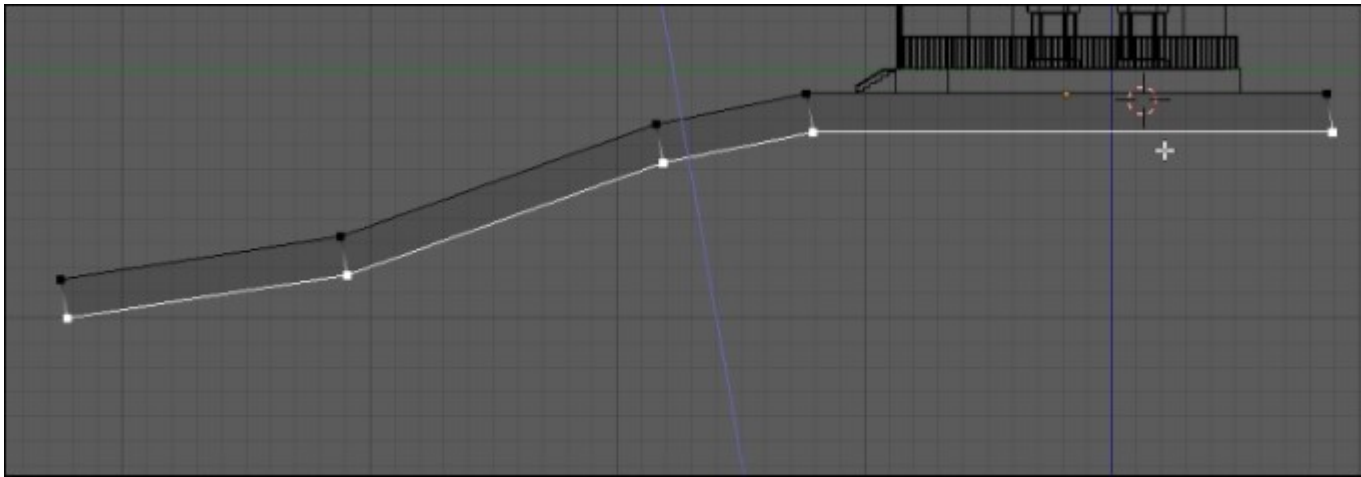
Modeling the cliff

Let's now model the cliff:

1. We will start by modeling its ground part. In order to do this, we add a plane and scale it.
2. We will then move the ground, so the house is placed above it.
3. We will use the scale tool in order to make the ground wider.
4. In the side view, we will enter the **Edit Mode** and activate the wireframe (Z). We will select the two vertices in the front of the plane, and by pressing *Ctrl* and the LMB, we will extrude the cliff profile to the left.
5. We will then reshape the geometry from a top view by moving the vertices with the Grab (*G*) tool. The cliff should be narrower where the house is.
6. The whole geometry will then be extruded down to form the height of the cliff.

Note

Note that you may have the normals of the face pointing inwards causing lighting mistakes. In order to recalculate them, use the *Ctrl + N* shortcut.



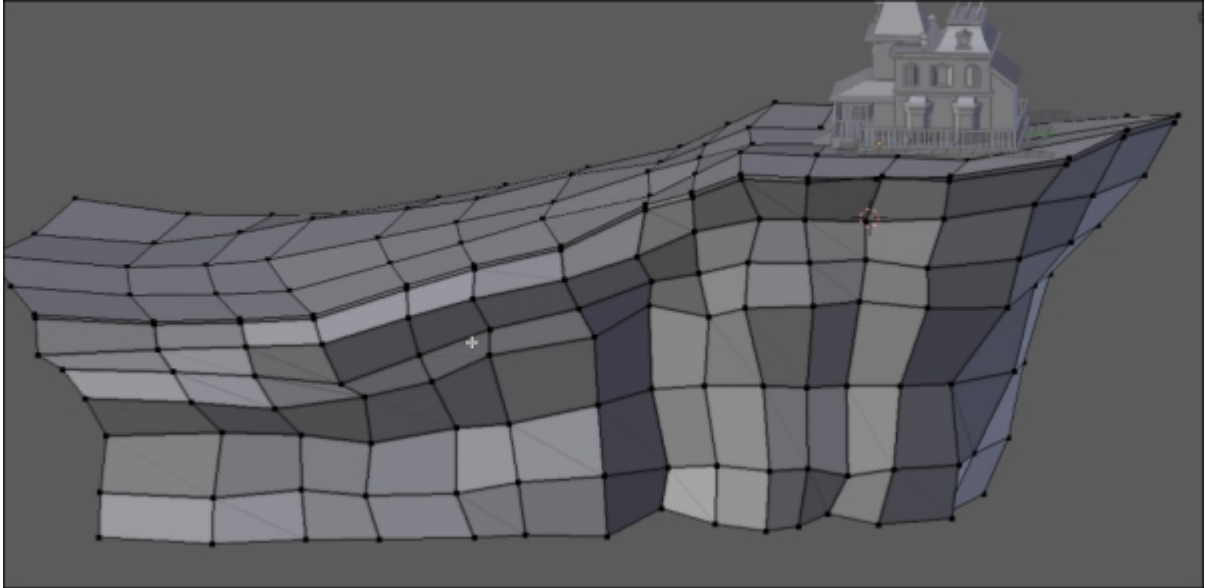
Starting the cliff modeling

7. At this point, we can add some new horizontal loop cuts (*Ctrl + R*) in order to add definition to our model.

Now our goal is to give the impression of a rock with the few polygons that we have. To get the shape, we will go in the top view, and with the wireframe turned on, we will select the edges of the contour of

the cliff and move them to form ridges and a valley. This will give an angular look. Remember that all the details will come with the texturing process in the next chapter, so we only have to give the overall shape here.

1. We can improve the model with some randomness. To do this, we go in the **Tool** tab of the left 3D view panel, and under the **Deform** section of the **Mesh Tools** subpanel, we click on **Randomize**. The effect of the tool can be tweaked with the **Amount** slider in the last tool option panel. This tool will simply push all the individual selected components in a random direction.



The final cliff

2. Lastly, we can arrange the global shape of our object with **Proportional Editing**. For instance, we can scale the tip of the cliff with it.

Modeling a tree with curves

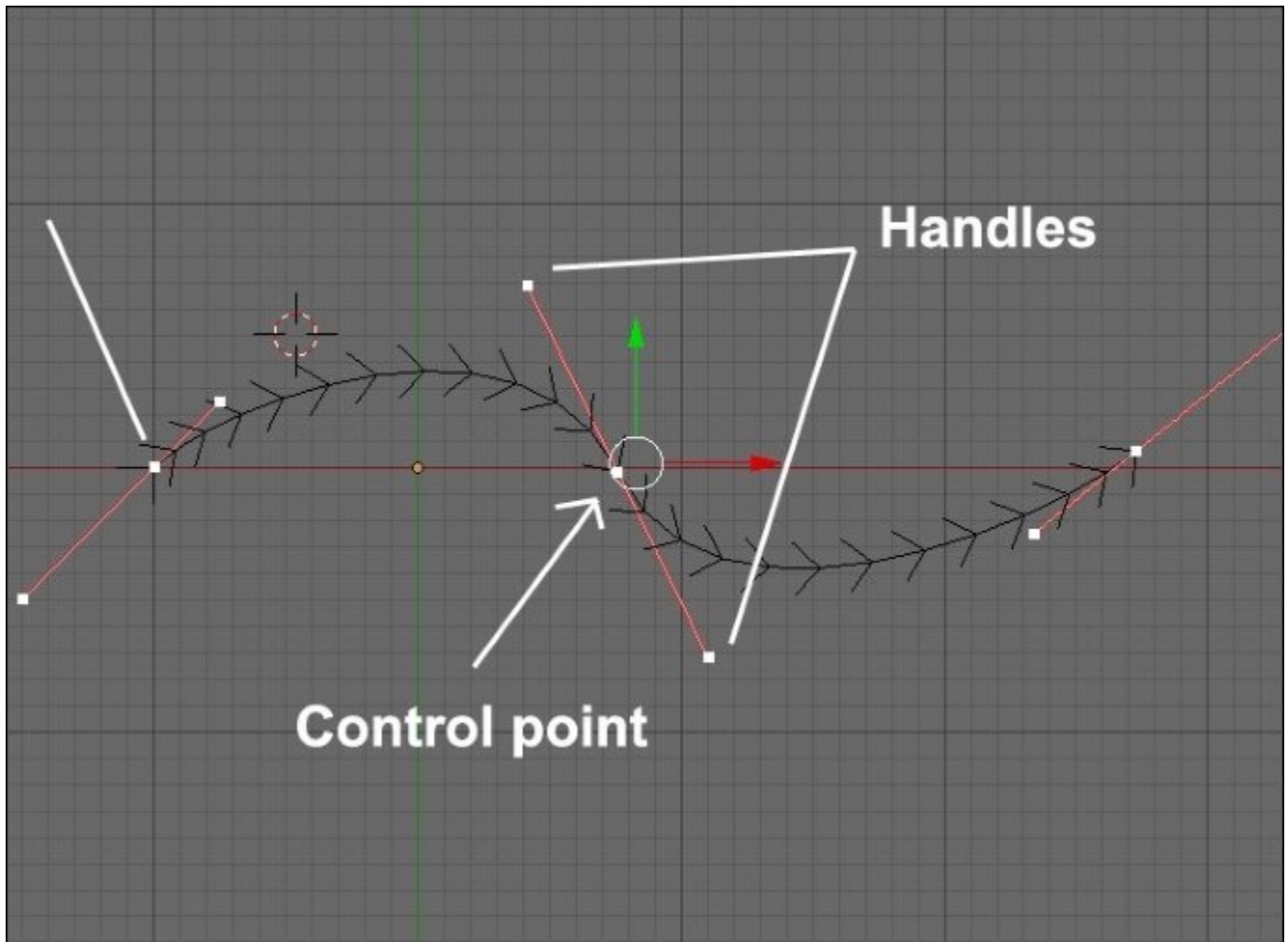
We are now going to learn a new way of modeling certain objects with curves. Curves are entities that can be useful to model ornaments, shoelaces, ropes, tree branches, and so on. In our case, we will model an entire tree with them.

Note

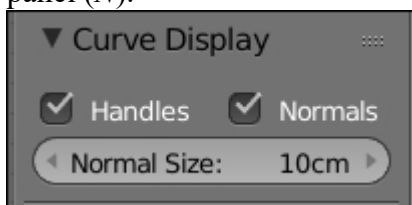
A word about curves

A curve is defined by control points that are connected together. Each point has a handle type. This can be changed by first selecting the control point that you want and by pressing the *V* key. By choosing the **Vector** type, the point will be angular. We are choosing **Automatic** or **Aligned**, so you will have Bezier handles that define the smoothness of the point. Handles always work in pairs. You can break their link by selecting the handle that you want to disconnect and by pressing *V* and select **Free**. In the beginning,

you will find that curves are hard to manipulate, but with a little bit of practice, it will become as easy as pie. But, of course, there is lot more to discover about them!



1. We will add a new curve of the Bezier type (press *Shift + A* and select **Curve | Bezier**). Then we will enter in the **Edit Mode**.
2. We will then turn off the handles and the normals of the curve. To do this, we will uncheck the **Handles** and **Normals** check boxes under the **Curve Display** subpanel in the right 3D view panel (*N*).

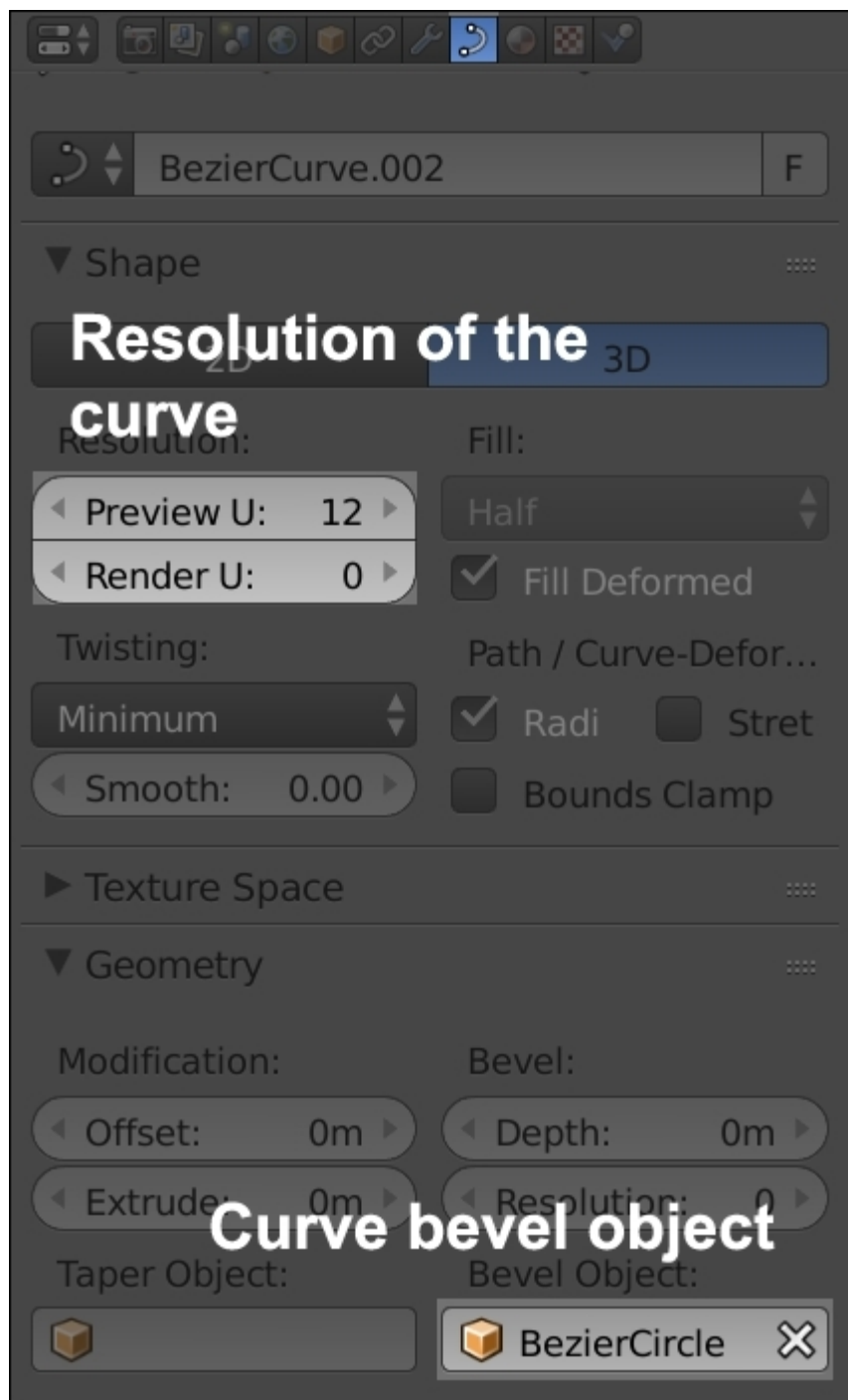


The Curve display options in the N panel.

3. When you start working on a shape that uses curves, it's easier to set all the control points as **Vector** (press *V* and select **Vector**).
4. Now we will leave the **Edit Mode** and add a new curve object of the bezier circle type (press *Shift + A* and select **Curve | Circle**). This circle will define the volume of our bezier curves.
5. We will select the bezier curve, and in the Properties editor, we will click on the object data icon. Under the **Shape** subpanel, we will then select the circle as the **Bevel Object**. As you can see, the curve is now getting a volume defined by the circle. If you change the circle, its scale, for instance, will change the curve volume.
6. We will then close the holes on each side of the curve by checking the **Fill caps** option under the **Bevel Object** one.
7. We can now modify our curve to form the trunk of the tree. To do this, we can simply vertically rotate the two points that we already have and extrude the tip several times with *E* or press *Ctrl* and LMB. You can also subdivide the curve by selecting at least two consecutive points and pressing *W* and select **Subdivide**.
8. Now we can change the radius of each point by changing its value in the **Transform** subpanel in the left 3D view panel. This is how we will taper the trunk.
9. We can now add new branches to our tree by simply adding a new point by pressing *Ctrl* and LMB without anything else selected. This point could then be extruded.

We can also change the radius of the branch. If you want to quickly select a branch, you can either select one of its points and press *Ctrl + L*, or select one of its points and press *Ctrl* and the *+* numpad key several times.

10. We will now add more branches by duplicating the first one that we've created. The process simply involves the duplication and placement of a lot of branches that differ in size and radius.
11. Next, we can convert our curves to a mesh object. In order to do this, we will have to first decrease the subdivision of our branches and our circle profile object by going to their **Object data** tab in the Properties editor and by changing the **Preview U** (for the viewport) and **Render U** (for the final render) sliders. We can now select our curve object and press *Alt + C* and select **Mesh from Curve/Meta/Surf/Text** to convert our curve object into polygonal geometry.



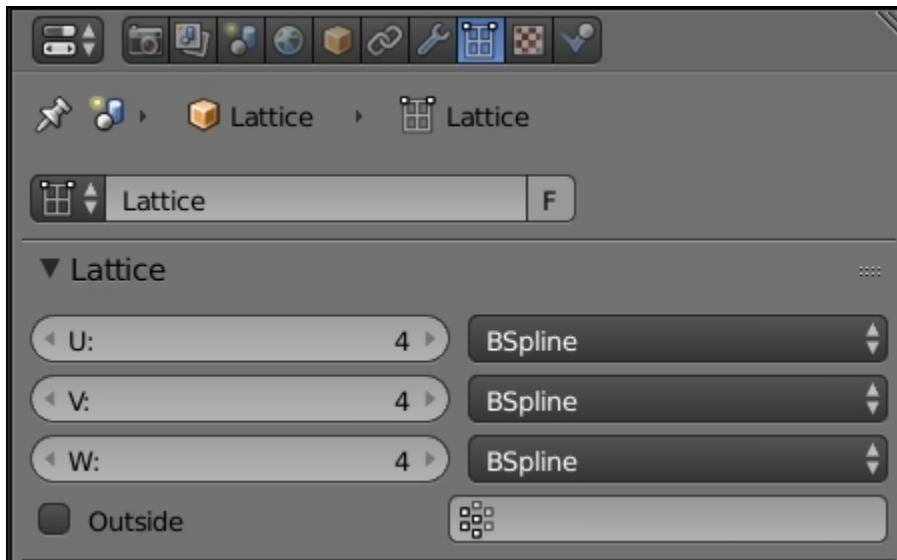
The curves option in the Properties editor

We will now show you how to deform the tree with a special type of object called a lattice. This will serve us to change the overall shape of it.

Note

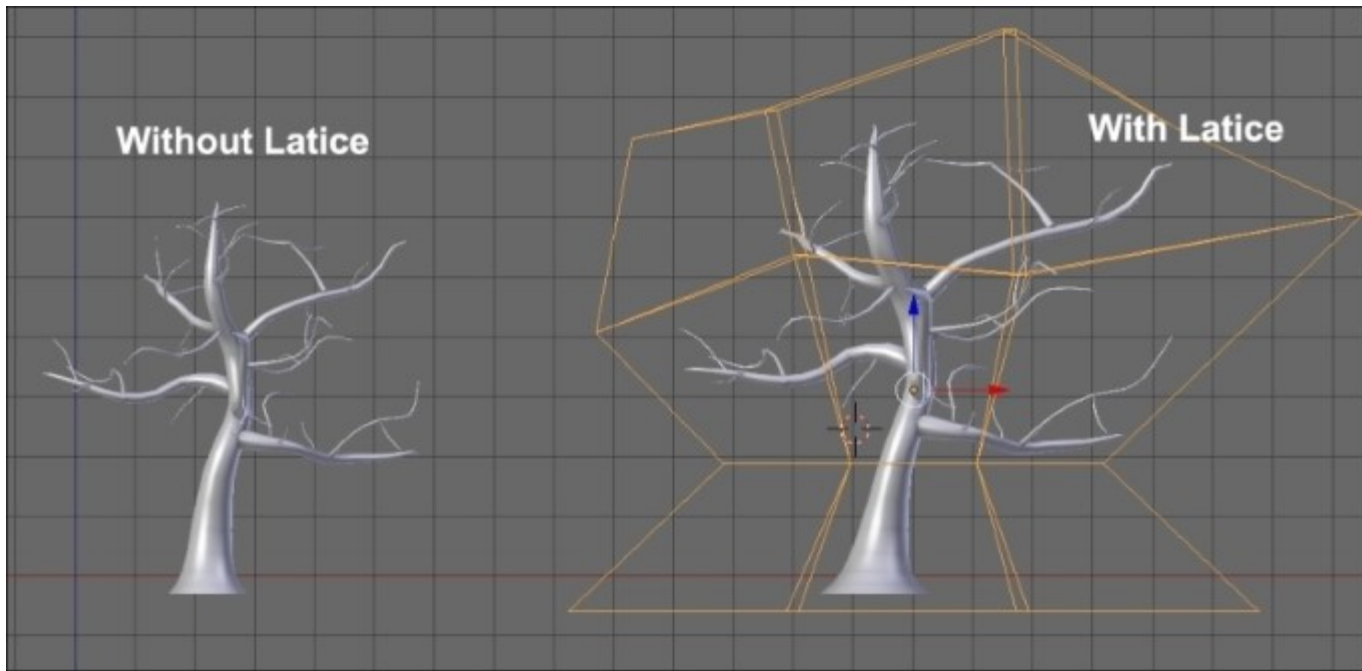
What exactly is a lattice?

A lattice is a very useful object that is bound to another object. It is a cube that can be subdivided along three axes: U, V, and W. This needs to encapsulate the object to which it is bound. Then, when we move its control points, it performs a sort of a projection that allows us to deform the bound mesh.



*The **Lattice** options in the Properties editor.*

1. We will add a new lattice by pressing *Shift + A* as usual.
2. Now that we have a new lattice, we can place it around our tree. No geometry should be outside the lattice or it will cause problems.
3. We will now increase the subdivisions of the lattice in U, V, and W to 4. This will give us enough control points in order to tweak our tree.
4. Now it's time to bind our lattice to our tree by simply selecting our tree and adding a new **Lattice** modifier. We will now have to specify to Blender the lattice object that we want to bind to our mesh in the **Object** parameter. In our case, we only have one lattice so it's easy to find it in the list, but if you have many lattices, don't forget to name them.
5. We can now enter **Lattice Edit Mode** and move the control points as if they were vertices.

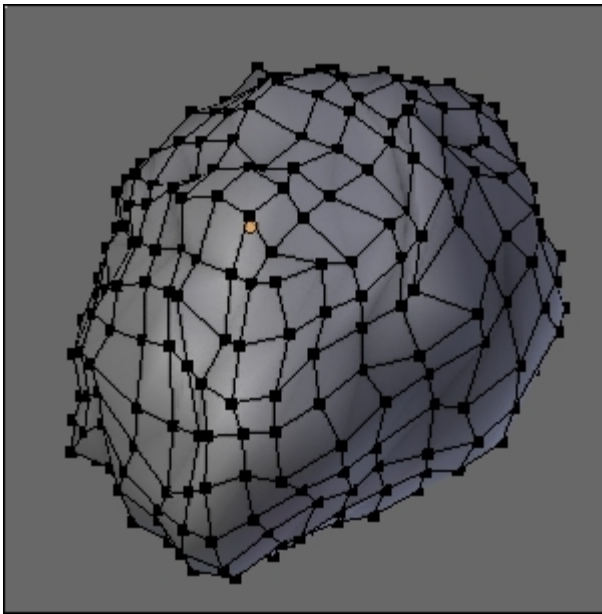


The final tree with its lattice

Enhancing the scene with a barrier, rocks, and a cart

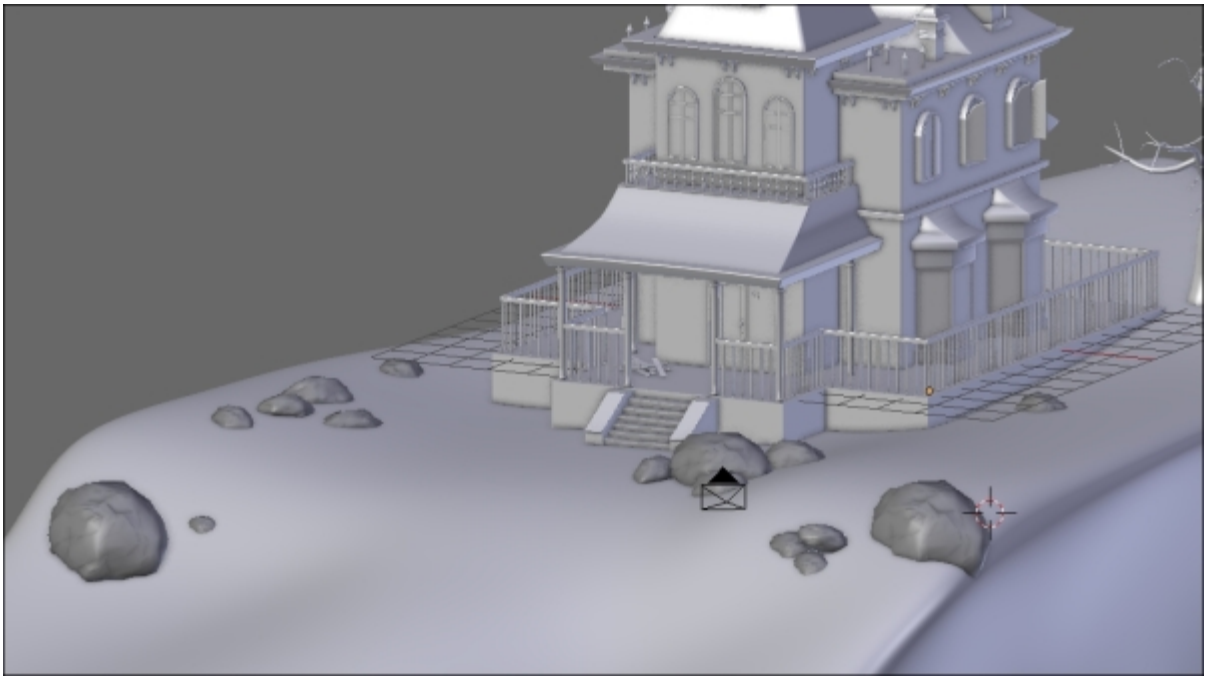
We will now enhance our scene by adding some cool assets using the techniques that we've seen in this chapter:

1. Let's start with the easiest asset, the rock. We will start this element with a cube.
2. Now in **Edit Mode**, we will use the **Subdivide smooth** option (press *W* and select **Subdivide smooth**).
3. In the last tool subpanel of the left 3D view panel, we can tweak the **Fractal** slider in order to add some randomness to the subdivision.
4. We can now repeat steps **2** and **3** several times.



A single rock

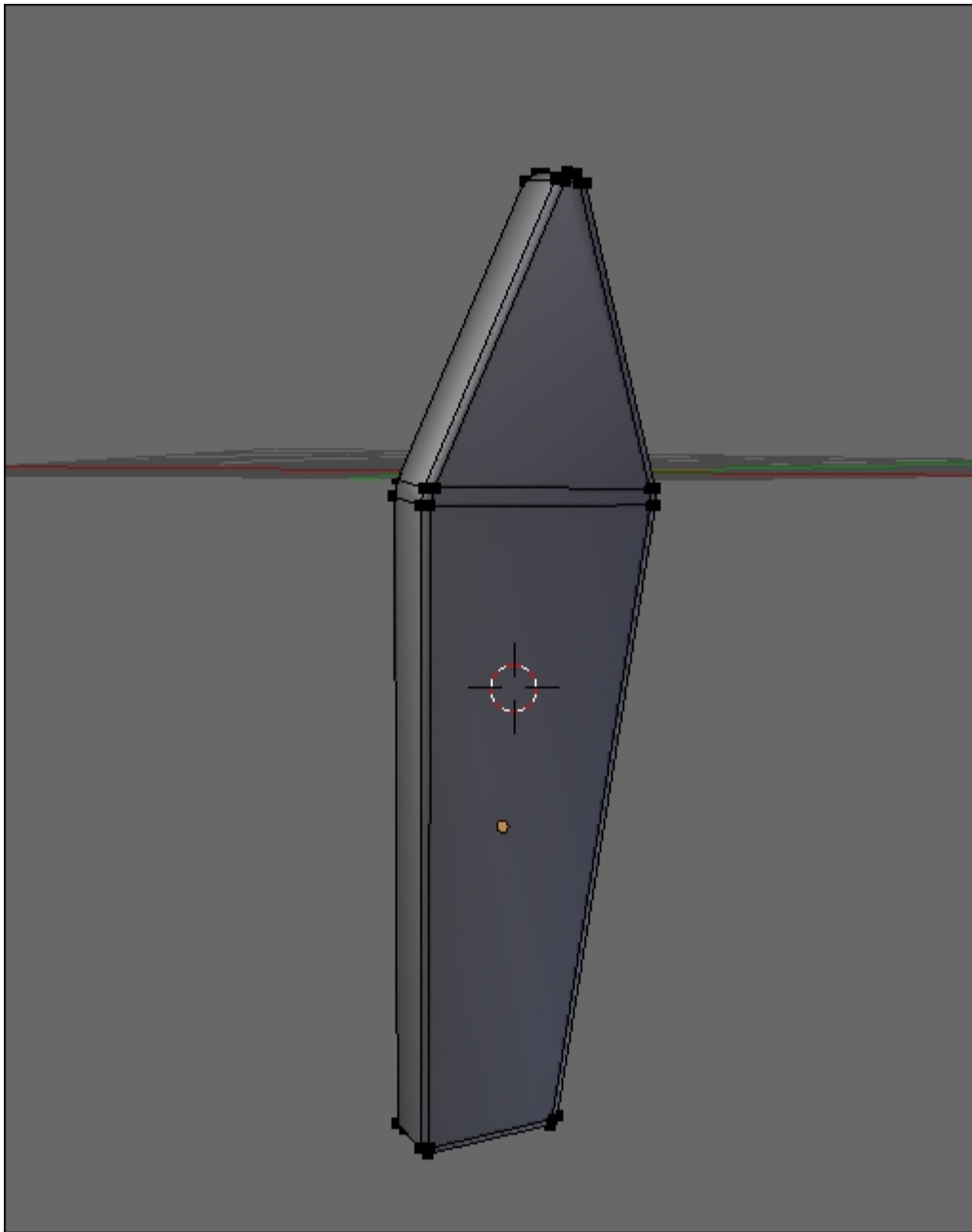
5. Now that we have a pretty spherical rock, we will use a **Lattice** modifier to shape it more like a rock. In our case, we will set our lattice with three subdivisions in U, V, and W, and after we've done the job, we will apply the modifier on the rock.
6. We can now use the instance duplication (*Alt + D*) tool in order to populate our terrain with rocks. This is a little bit of cheating, but we don't have to do any other objects in order to give the impression that there are many different rocks. Indeed, we can simply rotate and scale the rocks in a random manner! In this case, the Free Rotation tool (press *R* key twice) is quite useful.



The rocks are now placed all around the house.

We will now create a barrier in front of the house with a very nice modifier.

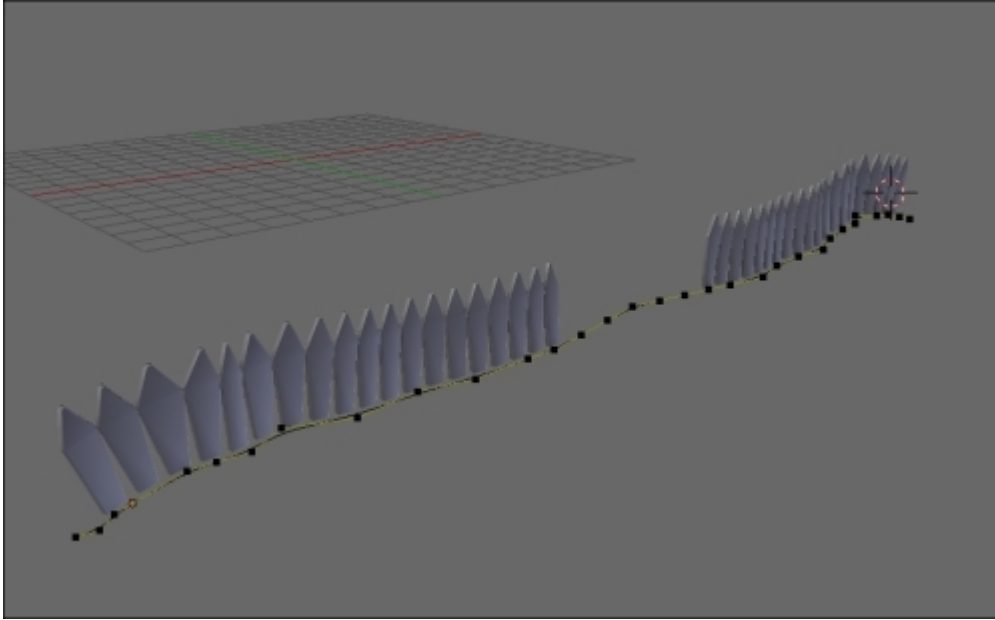
1. We will first add a new cube that we will scale on the x and y axes to make it thinner.
2. Next, we can select the top face in the **Edit Mode** and move it to change the height of the plank.
3. With the top face selected, we will extrude the top part of the plank and scale it on X .
4. We can then select all the objects (A) and use the **Bevel** tool ($Ctrl + B$) to smooth their edges.



The barrier plank

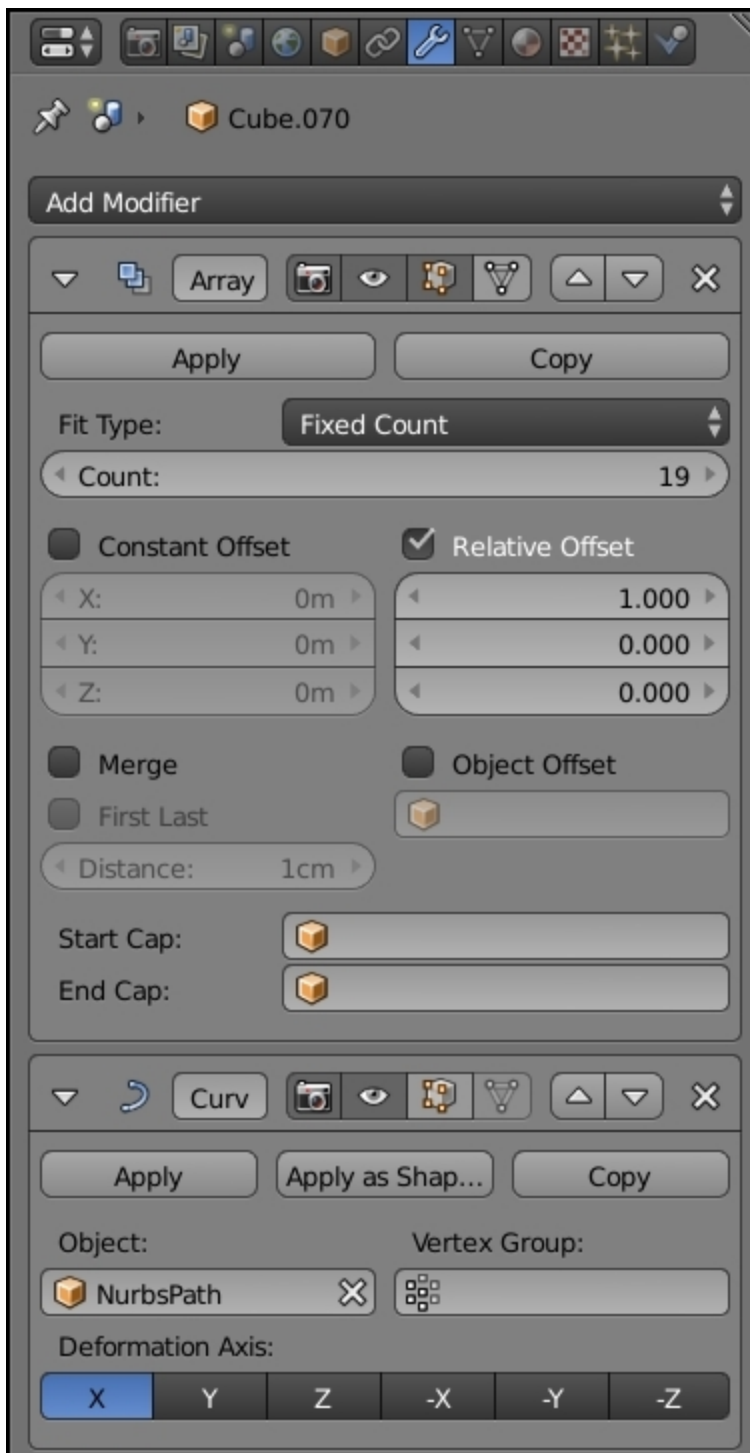
5. We can now place it on the far left of the cliff in front of the house.
6. The next thing is to use the **Array** modifier in order to do a line of planks to the middle of the house.
7. We can duplicate this barrier piece on the other side.
8. In order to break the boring alignment of these objects a little bit, we will use a path curve (*Shift + A* and select **Curve | Path**) that will guide the planks.

9. The path object looks like a curve. We can select each control point, extrude them, and subdivide them. We will extend the path object from the far left to the far right of the cliff and subdivide it several times (*W* and select **Subdivide**).
10. We can now break the path by moving each control point in a random way.
11. On the barriers, we will add **Curve** modifiers and select our path as the object. In our case, the deformation axis is *x*. Be careful that curve modifier is placed above the array of the stack. Indeed, we want to deform only one plank. As you can see, we now have our barrier following the path.



The final barriers with their curve

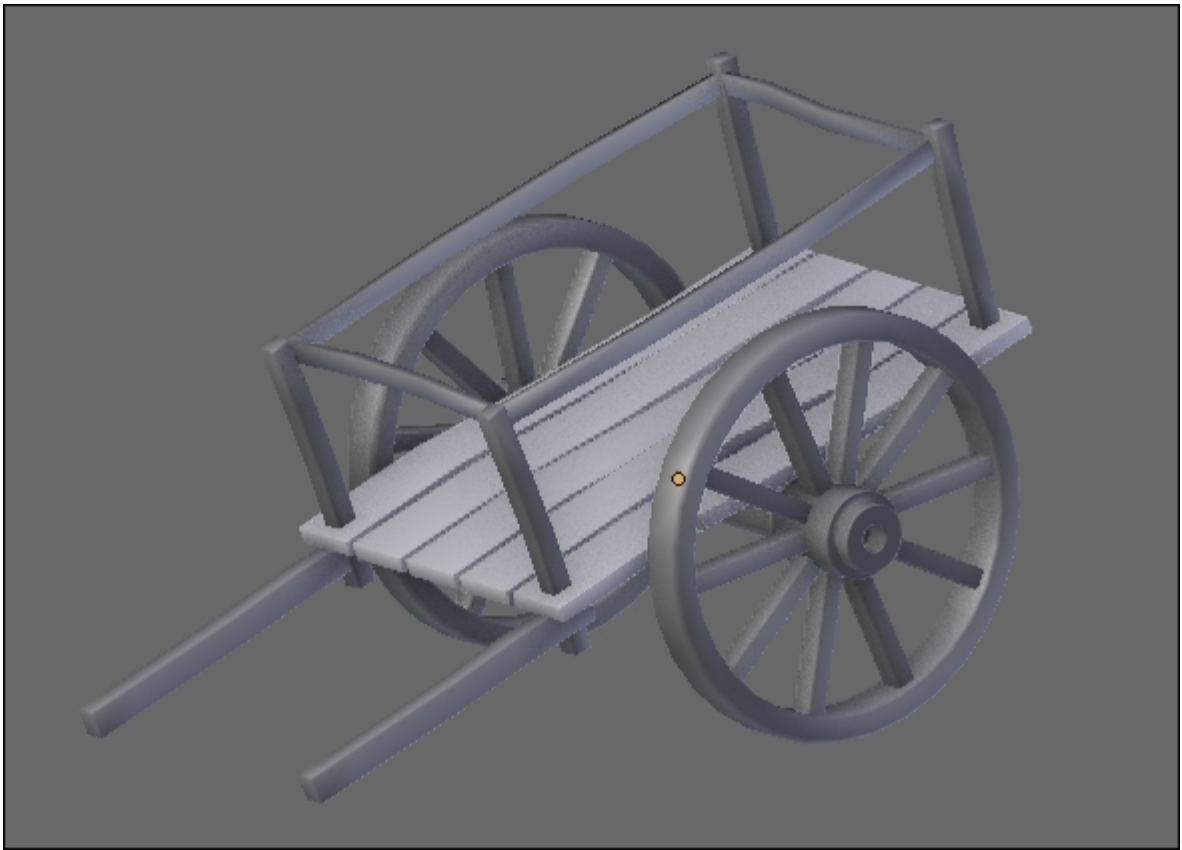
The following screenshot shows the barrier modifiers:



The barrier modifiers

It's now time to model the cart, which is a more complex object, but still simple to do with all the tools you've learned until now:

1. Let's start by duplicating one of the planks that we used in our physics simulation. This time we want to modify it without breaking the ones that are part of our simulation, so we press the *U* key and select **Object & Data** to make it a single user.
2. We can now use an **Array** modifier to make the bottom part of our cart. We can specify a little margin between each of them.
3. Now we will apply the modifier and tweak each plank so that they don't look the same.
4. We will now add a small cube and change its height. We can break its silhouette a little bit with **Loop Cut** (*Ctrl + R*) that we will move, and finally, add a bevel.
5. Now we can duplicate this object (*Shift + D*) at each corner of the cart.
6. We can now add a small plane that we will extrude by following the outline of the cart. We will add a **Solidify** modifier and apply it.
7. We can now add a new cube that we will shape into handles. As always, it's a matter of loop cuts, bevel, and placement.
8. Now, the last thing we need is two big wheels. To do this, we will add a circle, and in the **Edit Mode**, we will extrude it along the normals (press *E + Alt + S*). We can also extrude the wheel to give it a thickness. Remember to check your normals' orientations and clean them with *Ctrl + N* if needed.
9. Now we will add some radius. To do so, we first select one of the inner faces of the wheel, duplicate it and extrude it. We can select the whole radius and press *P* for **Separate selection**. We will also need to change its pivot point at the center of the wheel (press *Ctrl + Alt + Shift + C* and select **At center**).
10. We then duplicate the wheel with *Alt + D*, and without validating the action, we will press *R* to rotate it. This is because now we can press *Ctrl + R* to repeat the duplication and rotation process. What a time saver!
11. In order to cap the wheel, we will add a cylinder that will be extruded several times.
12. Finally, we can duplicate the wheel on the other side with *Alt + D* and add a cylinder that will connect both wheels. Remember to delete the unneeded faces on each side.



The final cart

As you can see, the items that we created are quite simple, so we encourage you to add more of them, such as a dead trunk or a scarecrow. Be creative, you have all the tools needed to do that yourself.

Organizing the scene

This is maybe the largest scene you have ever made, in terms of details and number of objects. So we will take some time to learn how to organize ourselves in order to have a more manageable scene.

Grouping objects

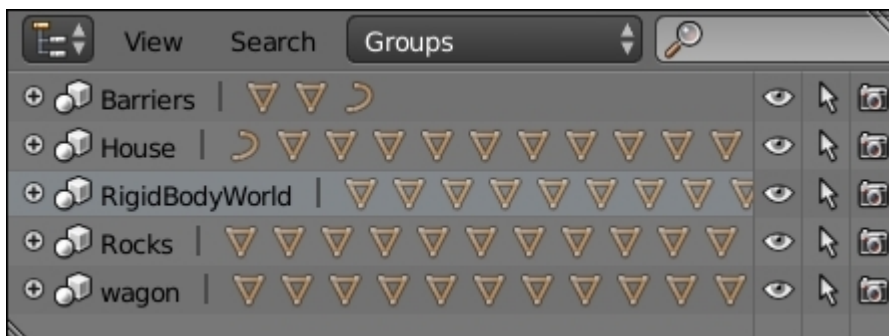
One interesting thing that Blender has to offer is the grouping tool. Let's see how it is working:

1. We will group objects that have a logical relationship between them. So we first select every part of the house, such as the foundations, the walls, the curtains, and the fireplace, and we will place them in a group by pressing *Ctrl + G*.
2. If we look at the last tool options in the left 3D view panel, we can enter a name for our newly created group.
3. Now we can select barriers and create a new group with them.
4. We'll leave the cliff alone, so we don't have to put it in a group.
5. The rocks will also be part of a group.
6. We will also group all the cart pieces.

Note

Using groups

Groups are very useful as they allow you to organize a scene better. You can see all the groups that are in a scene in the outline by changing the **All Scenes** drop-down menu to **Groups**. Here you can select them, disable their render, and hide them. You can also select one object that is part of a group in 3D view and press *Shift + G* and select **Group** to select all the other objects of the same group. You can add or remove an object from a specific group in the **Object** tab of the Properties editor under the **Groups** subpanel.



Working with layers

Another thing that can help your scene organization are the layers. They are located in the 3D View header. You can move objects on specific layers with the *M* key. A popup will show you all the layers.

You can move an object on multiple layers by pressing shift and selecting the layers that you want. The layers that have objects are marked with a little circle. You can display multiple layers at a time by pressing the *Shift* key and clicking on them in the header bar. Note that you can also see the layers in **Scene** tab of the Properties editor. In our case, the house is on layer one and the other elements on layer two.



Layer one is selected, but we can see that there are objects on layer two

Summary

You have completed the first part of the Haunted House project! You have learned some tricks to save time, such as how to instantiate objects or how to use the array modifier. You have also learned another modeling method with curves. You will clearly see the purpose of duplicating objects by instance in the next chapter where we will UV unwrap the objects. But don't think that it's over with the modeling process here. We can later change our objects after the texturing part in order to break the likeness between each instantiated object. Let's texture our scene!

Chapter 6. Haunted House – Putting Colors on It

This chapter will be devoted to the texture creation pipeline. We will explain new ways to you for the UV unwrapping processes, such as the **Project From View** or the **Smart UV** method. Then we will cover the basis of the powerful Texture Paint tool in order to create hand-painted textures for our house and environment. You will also learn about the tiling method in order to save time. In order to enhance the final set, we will show you how to create transparent textures, such as grass or grunge to age the house. In this chapter, you will not learn how each object is unwrapped and painted step by step, but you will gain a thorough understanding of the process. Finally, you will learn how to produce a test render with Blender Internal. This will be a good transition to the next chapter. So let's dive into the texture creation process with Blender!

In this chapter, we will cover the following topics:

- Learning the different ways to project UVs on an object
- Painting your model with the Texture Paint tool
- Creating hand-painted and tile-able textures
- Baking diffused textures on proper UVs
- Making transparent textures
- Creating a draft render in Blender Internal

Unwrapping UVs

We will now cover the UV unwrapping process that will later allow us to add textures to our objects.

Using Project From View

In order to quickly unwrap the UVs of the walls of the house, the shape and size of the wall still being accurate, we can use the **Project From View** method.

For this to work, we need to align the walls on the axis of the world coordinates (X, Y, Z). Indeed, we are going to make use of the angle of the 3D View of Blender to project the UVs. So, for each wall, we are going to rotate our view so that the wall that we are going to project is flat. This method only works for objects that are flat and aligned, so don't use it for organic shapes:

1. We will enter the Orthographic mode (5) and then enter the view that corresponds to the face that we want to unwrap. For instance, in order to unwrap the UV of the front wall we go into the front view (1).
2. In the **Edit Mode** and in the **Face Mode**, we will select all the polygons of the 3D model that we want to unwrap and be careful with the beveled parts.
3. In the **UV Mapping** menu (U), we will select the **Project From View** option.

We will obtain a perfect UV Island that is well proportioned according to the mesh. Usually, we don't need to tweak UVs with this method, except when we are willing to scale them. To check the size of our UVs, we will use a UV grid texture.

4. We will create a new editor by selecting **Split Area**, then we will go to **UV/Image Editor**.