UNIVERSITY OF PAVIA

FACULTY OF ENGINEERING

DEPARTMENT OF INDUSTRIAL ENGINEERING AND
INFORMATION

MASTER'S DEGREE IN COMPUTER ENGINEERING

MASTER THESIS

3D MODELING: THE BAS-RELIEFS OF THE St. Augustine's Ark

Candidate: Shibang Ta

Supervisor: Prof. Cantoni Virginio
Co-supervisor:

A.A. 2018/2019

Dissertation for the Master's Degree

3D MODELING: THE BAS-RELIEFS OF THE St. Augustine's Ark

Candidate:	Shibang Ta
Pavia Supervisor:	Prof. Cantoni Virginio
Academic Degree Applied for:	Master of Science
Major:	Computer Science
Affiliation:	School of Computer Science and Engineering
Date of Defense:	February 2020
Degree-Conferring-Institution:	University of Pavia

Table of Content

Abstract

3D models represent a physical body using a collection of points in 3D space, connected by various geometric entities such as triangles, lines, curved surfaces, etc. Being a collection of data (points and other information), 3D models can be created by hand, algorithmically (procedural modeling), or scanned. Their surfaces may be further defined with texture mapping[1].

Today, 3D models are used in a wide variety of fields. The medical industry uses detailed models of organs. The movie industry uses them as characters and objects for animated and real-life motion pictures. The video game industry uses them as assets for computer and video games. The science sector uses them as highly detailed models of chemical compounds. The architecture industry uses them to demonstrate proposed buildings and landscapes in lieu of traditional, physical architectural models. The engineering community uses them as designs of new devices, vehicles and structures as well as a host of other uses. In recent decades the earth science community has started to construct 3D geological models as a standard practice. 3D models can also be the basis for physical devices that are built with 3D printers or CNC machines.

This thesis contains an introduction to 3D modeling and some detailed explanations of blender software, as well as the production process of two 3D modeling works.

Keywords: 3D modeling, blender, method, process

Chapter 1 Introduction

1.1 3D modeling

Generally speaking, "3D modeling" refers to the use of 3D production software to build a model using 3D data through a virtual 3D space. 3D modeling can be roughly divided into NURBS and polygon meshes.

NURBS is a very good modeling method, which is supported in advanced 3D software. NURBS can better control the curve of the surface of the object than traditional mesh modeling methods, which can create more realistic and vivid shapes.

Polygon meshes are meshes with no regular topological relationship. They usually consist of polygon triangulation. Polygon meshes are internal points within a mesh area that do not have the same adjacent cells. That is, the number of meshes connected to different interior points in the meshing area is different.

NURBS has a good application for models that require precision, flexibility, and complexity, and is suitable for quantitative production applications. Polygon mesh modeling is by modifying the surface of the model, which is suitable for renderings and animations of complex scenes. Generally speaking, each has its own advantages.

1.2 Introduction:

3D is an abbreviation of three-dimensional, which is three-dimensional graphics. Displaying 3d graphics in a computer means displaying three-dimensional graphics in a plane. Unlike the real world, real three-dimensional space has real distance space. The computer just looks a lot like the real world, so the 3d graphics displayed on the computer are just like real eyes. One of the characteristics of the human eye is that it is near and far away, and it will form a three-dimensional sense. The computer screen is flat and two-dimensional. The reason why we can appreciate the real three-dimensional image is that the difference in color gray when displayed on the computer screen makes the human eye produce the visual illusion. The screen is perceived as a three-dimensional image.

Based on the relevant knowledge of color science, the convex part of the edge of a three-dimensional object generally shows a high-brightness color, while the concave part becomes dark due to being blocked by light. This recognition is widely used in the drawing of buttons and 3D lines in web pages or other applications. For example, 3d text to be drawn, that is, a high-brightness color is displayed at the original position, and a low-brightness color is used to outline its outline at the lower left or upper right position, so that the 3D text effect will be produced visually. In specific implementation, two 2D texts of different colors can be drawn at different positions using the same font. As long as the coordinates of the two texts are appropriate, 3d texts with different effects can be produced visually.

1.3　Development Prospects

The form of the Internet has always been in 2D mode, but with the continuous progress of 3D technology, in the next few years, more and more Internet applications

will be presented to users in 3D, including online video, electronic reading, online games, virtual communities, e-commerce, distance education and more. Even for the tourism industry, the 3D Internet can play a driving role. Some world famous places, sculptures and antiques will be used on the Internet to allow users to experience in 3D. The real shock of this experience is far more than the 2D environment.

Take the rapid development of e-commerce as an example. Massive products need to be displayed on the Internet. Specialized, personalized, and authentic product display is particularly important. However, due to the constraints of 3D model production costs, these requirements can only be temporarily two-dimensional. The photos are used for satisfaction, which causes the information about the products or objects passed to consumers to be incomplete, detailed, and lifelike, which reduces consumers' desire to buy and accuracy. The 3D product display technology can interactively display products in a three-dimensional manner on a web page. Consumers can view product characteristics in an all-round way and intuitively understand product information. The effect is almost the same as consumers directly facing products. Many manufacturers have adopted pseudo three-dimensional effects (sequence photo rotation) to temporarily replace the display of three-dimensional models. It can be seen that the future market desires for the technology of 3D modeling.

1.4   Software of 3D modeling

Understanding the software on the market is, I think, the first step towards a professional and 3D modeling career. This helps us to better understand our level and understand the advantages and disadvantages between software. Of course, we don't need to learn each software deeply, but it is not enough just to learn to master one of them.

- 3DS MAX

3D Studio Max, often referred to as 3ds Max or MAX, is a PC-based 3D animation rendering and production software developed by Autodesk. Its predecessor is 3D Studio series software based on DOS operating system. Before the advent of Windows NT, industrial-grade CG production was monopolized by SGI graphics workstations. The emergence of the combination of 3D Studio Max + Windows NT suddenly lowered the threshold of CG production. First, it began to use animation production in computer games and then began to participate in special effects production of film and television, such as X-Men II, the Last Samurai, etc.
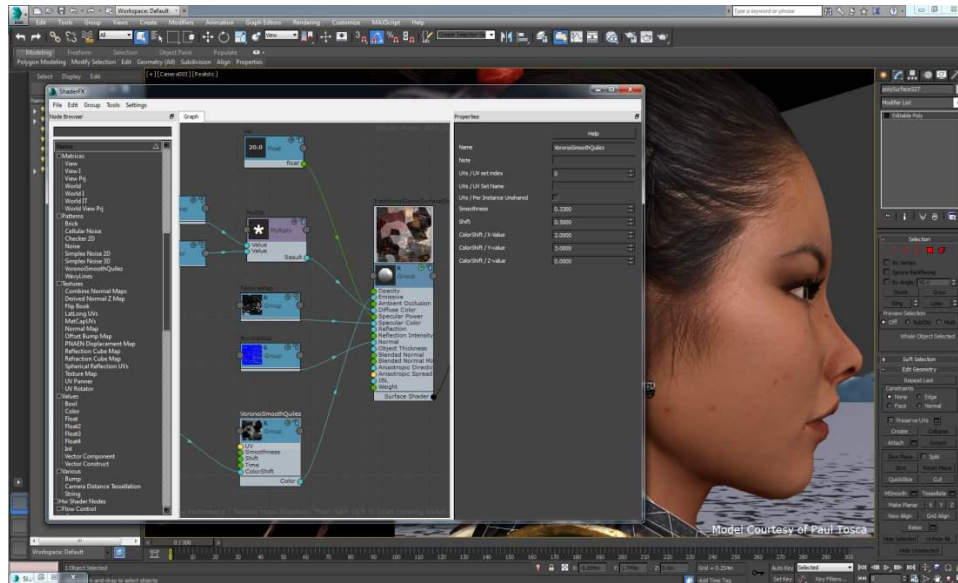
Figure1.1 Interface diagram of 3DS MAX

● Autodesk Maya

Maya is the world's top 3D animation software produced by the American company Autodesk. Its application objects are professional film and television advertising, character animation, and movie special effects, and it is a high-end production software for film grade.

Maya integrates Alias / Wavefront's most advanced animation and digital effects technology. It not only includes the functions of general 3D and visual effects production, but also combines with the most advanced modeling, digital cloth simulation, hair rendering, motion matching technology. Maya runs on Windows, MacOS X, Linux and SGI IRIX operating systems. Of the tools on the market for digital and 3D production, Maya is the solution of choice.

Maya has a high price and a well-known reputation. It is a coveted production tool for producers. Mastering Maya will greatly improve the production efficiency and quality, adjust the simulation of character animation, render the movie-like real effects, and move towards the world's top animators.

Figure1.2 Interface diagram of Maya

● Softimage XSI

SoftimageXSI is AUTODESK's flagship product for the high-end 3D film and television market. With its unique and truly non-linear animation editing, it is loved by many 3D computer art professionals. XSI pushes the computer's 3D animation capabilities to the extreme. It is the best animation tool. In addition to the new non-linear animation function, it is easier to set the traditional animation of Keyframe than before. It is a powerful tool for making movies, advertisements, 3D, architectural performance and so on.

SoftimageXSI has appeared in the world with its advanced workflow, seamless animation production and industry-leading non-linear animation editing system. SoftimageXSI is a node-based architecture, which means that all operations can be edited. Its animation synthesizer function can mix any action to achieve the effect of natural transition. Softimage XSI's lighting, materials and rendering have reached a higher level. The dozens of light spot effects provided by the system can be extended to millions of changes. The Caustic. Global Illumination and Final Gathering effects make the rendering effect unprecedented.

● Rhino

Rhino is a powerful professional 3D modeling software on the PC developed by Robert McNeel & Assoc in the United States. It can be widely used in 3D animation production, industrial manufacturing, scientific research, and mechanical design. It can easily integrate the model function parts of 3DS MAX and Softimage. It has a rock-solid performance for 3D NURBS models that require fineness, flexibility, and complexity. It can output different formats such as obj, DXF, IGES, STL, 3dm, etc., and is suitable for almost all 3D software, especially for increasing the model productivity of the entire 3D work team. So 3D designers should learn and use it.
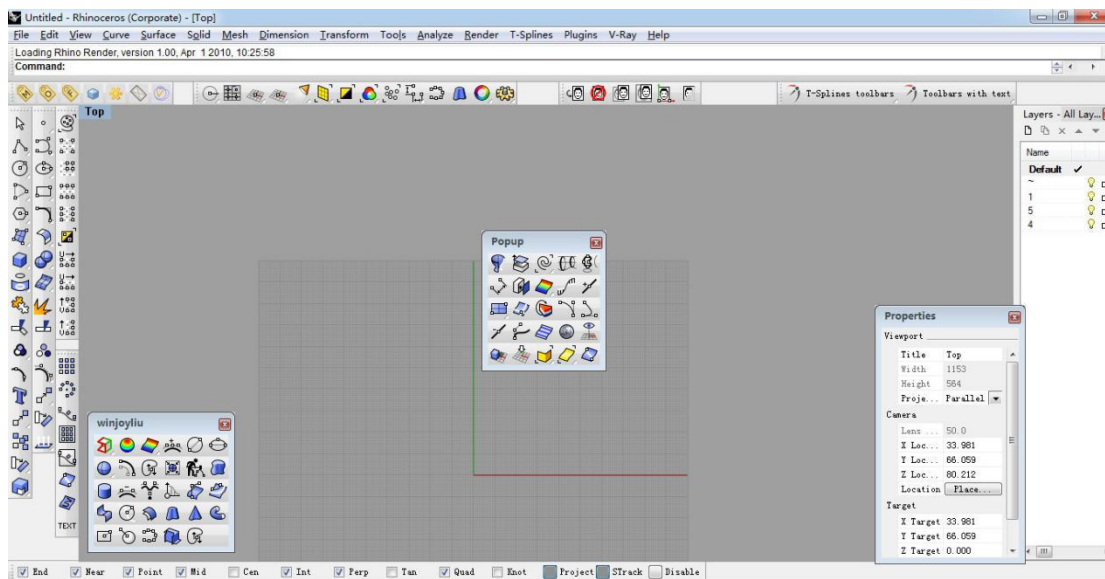

Figure1.3 Interface diagram of   Rhino

● Silo

Silo is a 3D software focused on modeling by Nevercenter, which is suitable for both biological modeling and regular object modeling. Use it to create characters or buildings for video games and movies. Silo is currently used by the world's top studios, which can be used alone or with multiple software platforms.

Silo 2.0 introduces many new workflow enhancements and tools, including powerful new features for full UV texture editing and iso-shift painting.Silo focuses on model building and shaping in the areas of 3D design, animation, video game production and communicating ideas.

Figure1.4 Interface diagram of Silo

● Zbrush

ZBrush is a digital sculpting and painting software that has revolutionized the entire 3D industry with powerful features and intuitive workflows. In a simple interface, ZBrush provides the world's most advanced tools for contemporary digital artists. The combination of functions developed with practical ideas, while stimulating the artist's creativity, ZBrush produces a user experience that feels very smooth when operating. ZBrush is capable of sculpting models up to 1 billion polygons, so the limit depends only on the artist's own imagination.

ZBrush software is the world's first 3D design tool that lets artists feel free to create creatively! Its appearance completely subverts the working mode of traditional 3D design tools in the past, freeing up the hands and thinking of artists, saying goodbye to the past awkward creation mode that relies on the mouse and parameters, and fully respects the designer's creative inspiration and traditional working habits.

Figure1.5 Interface diagram of ZBrush

● Blender

Blender is an open source cross-platform all-round 3D animation production software, providing a series of animation short film production solutions from modeling, animation, materials, rendering, to audio processing, video editing and so on[2].

Blender has a variety of user interfaces that are easy to use in different tasks, with built-in advanced video solutions such as green screen keying, camera back tracking, mask processing, and post node synthesis. There are also built-in cartoon strokes (FreeStyle) and GPU-based Cycles renderer. It uses Python as a built-in script and supports multiple third-party renderers.

Blender is designed for media workers and artists around the world. It can be used for 3D visualization, and can also create broadcast and movie-quality video. In addition, the built-in real-time 3D game engine allows independent playback of 3D interactive content may.

Figure1.5 Interface diagram of blender

Chapter 2 Principles of 3D modeling

2.1 Basic principles of 3D

2.11 Coordinate system

3D refers to the representation of all shapes in 3D space, and its position can be calculated using a coordinate system.
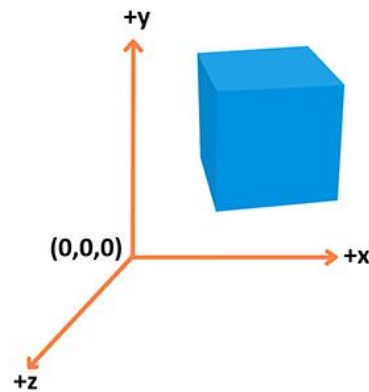
Figure 2.1 coordinate-system

This figure uses a right-handed coordinate system—the x-axis is right, and the y-axis is up and the z-axis is pointing off the screen.

2.12 Object

Use vertices to create different types of objects. A vertex is a point that has a coordinate position in a 3D coordinate system and some additional information that can define it. Each point contains these attributes:

- Location: Used to identify (x, y, z) in 3D space..
- Color: Contains RGBA (R, G, and B are red, green, blue, and alpha channels, respectively. The alpha channel controls transparency — all channel values range from 0.0 to 1.0).
- Normal line: Describes the vertex orientation.
- Texture: A vertex is used to decorate a 2D picture of the model surface. It is one of the alternatives to simple colors. You can use this information to build the geometry. Here is an example of a cube:
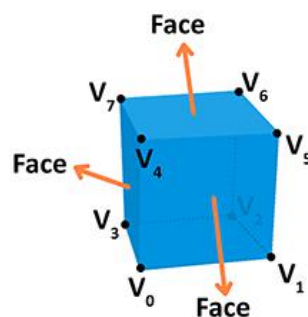
Figure 2.2 Object attributes

A face of a given shape is a plane between vertices.For example, a cube has 8

different vertices and 6 different faces, and each face consists of 4 vertices. A normal defines the orientation of the face. At the same time, connecting these points can create the sides of the cube.This geometry is made up of points and faces.The material uses a texture map, here a pure blue or a picture. The object's geometry consists of vertices and faces, while the material consists of textures. If we connect the geometry and the material we will get a mesh.

2.13 Rendering process

The rendering process is a process of outputting the previously prepared model to the screen. The 3D rendering process will accept the raw data describing the 3D object using vertices as input, calculate its fragments, and then render as pixels and output to the screen.
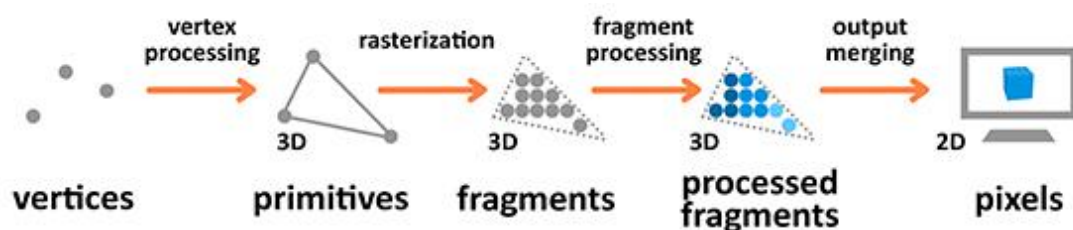
Figure 2.3 Rendering process

- Primitives: Input in the rendering process. Generated by vertices, it may be triangles, points or lines.
- Fragment: A 3D projection of a pixel has the same properties as a pixel.
- Pixels: The points arranged in the 2D grid on the screen, including RGBA.

Vertex and fragment processing is programmable, and we can write our own shaders to control the output.

2.14 Vertex processing

Vertex processing is to combine independent vertex information into raw data and set its coordinates in 3D space for easy display recognition. It's like when you take a picture: you have to place an item, set the camera parameters, and then take a picture.
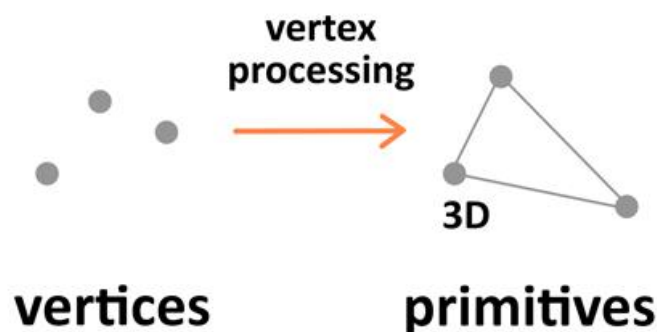
Figure 2.4 Rendering process

This process is divided into four steps: the first step is to prepare the objects in world coordinates, also known as model conversion. Then the view conversion, this step only concerns the position and the orientation of the camera in 3D space. The camera has three parameters (position, Direction and orientation), these three parameters must be defined in the newly created scene.
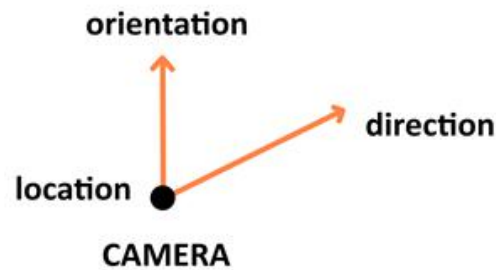

Figure 2.5 Camera settings

Projective transformation, also called perspective transformation, this step defines the camera settings. During this process it is set which are visible in the camera, and the configuration contains the field of view, aspect ratio and optional near and far cropping parameters.
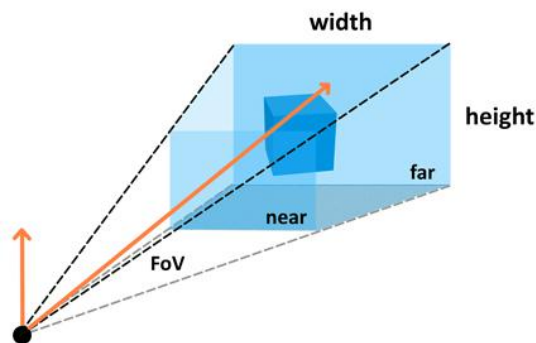
Figure 2.6 Projection transformation

The last step is the viewport transformation, this step will output everything to the next step in the rendering process.

2.15 Rasterized

Rasterization converts raw data into a series of fragments.
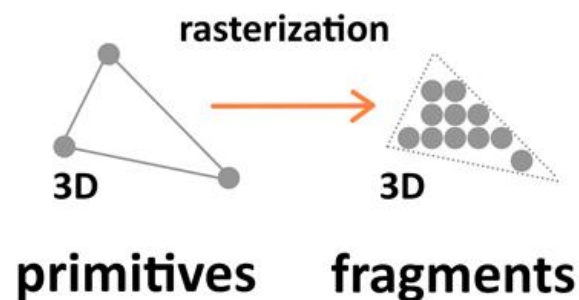
Figure 2.7 Rasterization

Those fragments correspond to the pixel grid, so in the end they will be printed directly onto the pixels by the 2D screen.

2.16 Fragment synthesis

Fragment synthesis focuses on texture and lighting。It calculates the final color based on the given parameters.
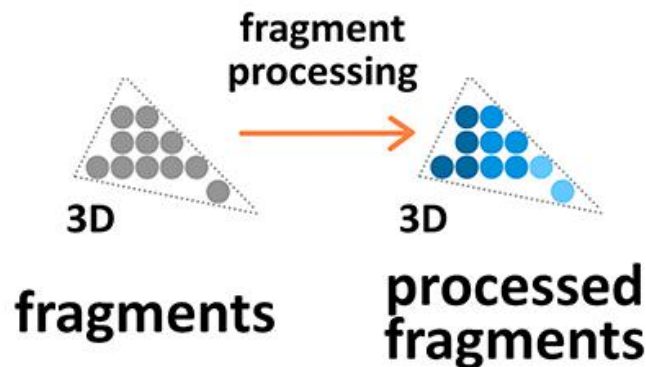
Figure 2.8 Fragment processing

The texture is used in 3D space to make the model look more realistic 2D pictures. Texture is a combination of individual texture elements called texels, the same principle as pixel combination. If necessary, adding textures to the model during the fragment processing stage of the rendering process allows us to use packaging and filtering for adaptation.

Texture packaging allows you to reuse 2D pictures on 3D models. Texture filtering is when the original resolution of the texture map and the resolution of the fragment to be rendered are different, the texture is scaled according to the situation.

The colors we see on the screen are the lighting and model colors, and the final result after the materials interact. Lights are absorbed and reflected. The standard Phong lighting model implemented in WebGL has four lighting parameters:

- Diffuse reflection: Distant direct light, just like the sun.
- Highlight: Point sources, like incandescent lamps in a room.
- Ambient color: Constant light that affects all models in the scene.
- Self-luminous: Light emitted by the model itself.

2.17 Output merging

During the output operation phase, all fragments of raw data from 3D space will be transformed into a 2D pixel grid and then printed to screen pixels.
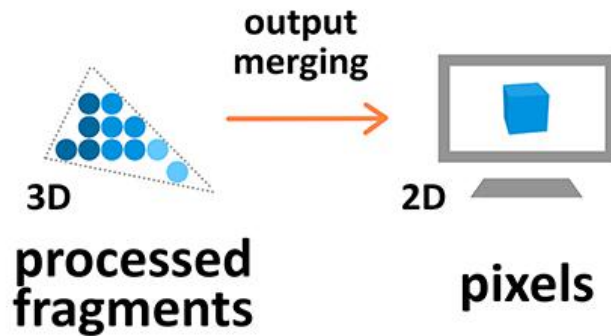
Figure 2.9 Output merging

2.2 Modeling methods

Modeling refers to the process of using a sketch drawn in two-dimensional space as a basic object to form an object in three-dimensional space. Modeling methods can be divided into two parts: basic modeling and advanced modeling. These modeling approaches complement each other.

2.21 Basic modeling

- Modeling with basic and extended geometry. Basic geometry is like some basic building modules on the construction site, such as bricks and tiles, which can quickly set up some simple scenes and have a wide range of uses. Although extended geometry is used less frequently, sometimes it can save a lot of time when properly applied.
- 2D with a modifier to 3D modeling. In 3d max, users can use the Shapes command panel to create two-dimensional graphics such as lines, rectangles, ellipses, circles, and polygons. Users can modify the 2D graphics through the commands in the editor to create the required 3D model. In the process of converting two-dimensional graphics into three-dimensional models, the modifiers often used are Extrude, Lathe, and Loft.
- Compound geometry modeling. Composite geometry is a very efficient modeling method. It is a combination of multiple shapes. It can use two or more three-dimensional or two-dimensional geometry to create another three-dimensional object. Among them are Loft and Boolean.

2.22 Advanced modeling

- Polygon modeling

Polygon modeling is a method of creating more complex 3D models by arranging modified points, lines, and faces. This method occupies a small system capacity and is easy to operate. Polygon modeling usually starts with a box or other simple geometry. Through continuous refinement and smoothing, you can finally create the desired model.

- Surface/Patch modeling

Patch is a surface modeling technique that is a type of modeling between polygon

meshes and NURBS. A patch is essentially a polygonal surface with a spline curve. You can modify the surface effect of the patch by modifying the spline curve, and finally make a model.

Although the principle of patch modeling is simple, it has high requirements for the sense of three-dimensional space. In order to improve the modeling efficiency and appropriately reduce the requirements for modelers, Surface Tools came into being.

You only need to build a three-dimensional frame of the spline curve in three-dimensional space according to certain rules, and then add a Surface modifier to generate a three-dimensional surface model of the Patch class.

● NURBS modeling

NURBS modeling is a very widely used modeling method at present, based on the control points to adjust the curvature of the surface and automatically calculate the smooth surface accuracy. Its advantage is that it has fewer control points and is easy to adjust and shape in space. It is especially suitable for describing the appearance of complex organic curved objects, complex biological surfaces and streamlined industrial products, such as cars, animals, etc. It is not suitable for creating regular mechanical or architectural models.

NURBS modeling method: First create several NURBS curves, and then connect these curves to form the desired surface object. Or use the NURBS creation tool to modify some simple NURBS surfaces to obtain more complex surface objects.

Chapter 3 Blender

3.1 Overview and application areas of blender

Blender is a free and open-source 3D authoring suite. Blender can create a variety of 3D visual content, such as still images, 3D animations, visual effects footage, and video editing. Blender is perfect for independent developers as well as small and medium-sized studios. Its highly unified workflow and very reliable development process can benefit everyone. Blender is a cross-platform application tool that can run under Linux, macOS, and MS-Windows systems. Compared to other 3D modeling tools, Blender requires less memory and drivers. Its interface uses OpenGL to provide a consistent user experience across all supported hardware and platforms[5].

Blender is a fully integrated 3D authoring suite, providing a large number of basic tools, including modeling, rendering, animation & rigging, video editing, visual effects, compositing, mapping, and many types of simulation.

Cross-platform, GUI using OpenGL can show consistent display effects on all major platforms (and can customize the interface through Python scripts). High-quality 3D architecture brings a fast and efficient workflow.

The project works of this thesis are based on blender software. So this chapter will introduce the general operation of some commonly used blender software.

3.2 Some functions of Blender

3.21.User Interface

- When starting Blender, the splash screen appears in the center of the window. It contains options to create new projects or open recently opened blend-files[7].
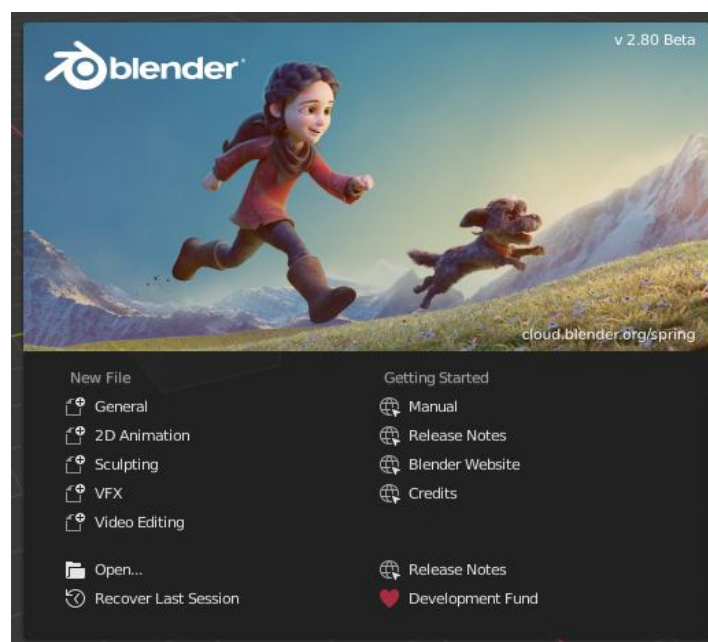
Figure3.1Blender Splash Screen

To close the splash screen and start a new project, click anywhere outside the splash screen (but inside the Blender Window) or press Esc. The splash screen will disappear

revealing the default screen.

Information Region: The upper part of the splash screen contains the splash image with a lot of key information overlaid.
Title: Besides the Blender icon and text, it shows the Blender version.
Date: At the top-right corner, you can see the date of when Blender was compiled.
Hash:The Git Hash. This can be useful to give to support personnel, when diagnosing a problem.
Branch: Optional branch name.

Interactive Region: The interactive region is the bottom half of the splash screen.
New File: Start a new project based on a template.
Recent Files: Your most recently opened blend-files. This gives quick and easy access to your recent projects.
Open: Allows opening an existing blend-file.
Recover Last Session:Blender will try to recover the last session based on temporary files. See Recovering Data.
Links: Links to the official web site. The same links can be found in the Help Menu of the Topbar.

3.22 Window System Introduction

After starting Blender and closing the Splash Screen the Blender window should look something similar to the image below; as Blender's user interface is consistent across all platforms.
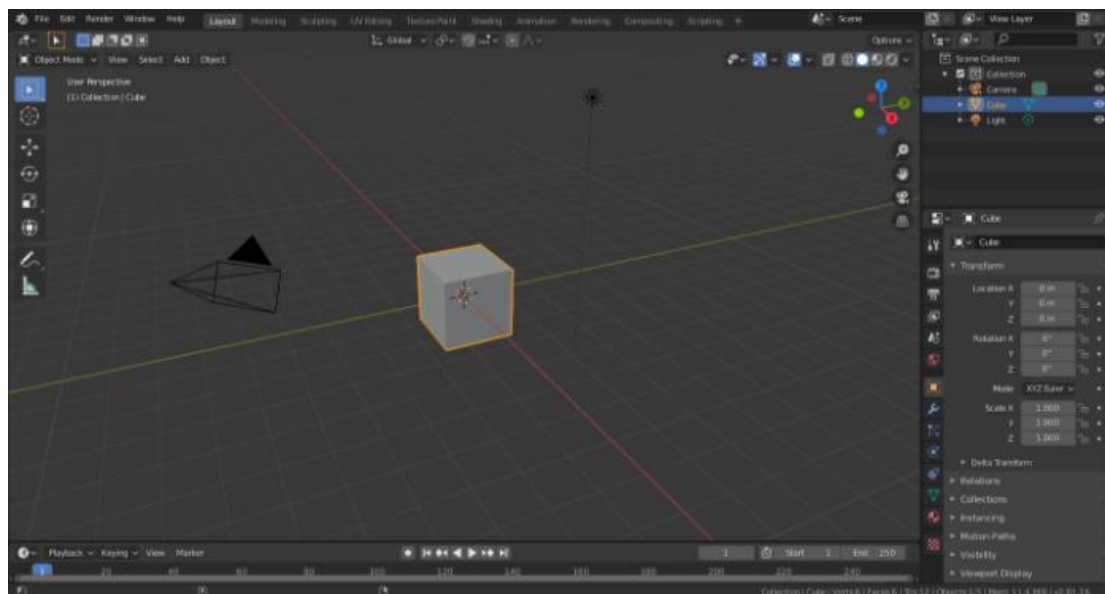
Figure3.2 The default startup Blender window.

Blender's interface is separated into three main parts:
- Topbar at the very top.
- Areas in the middle.
- Status Bar at the bottom.
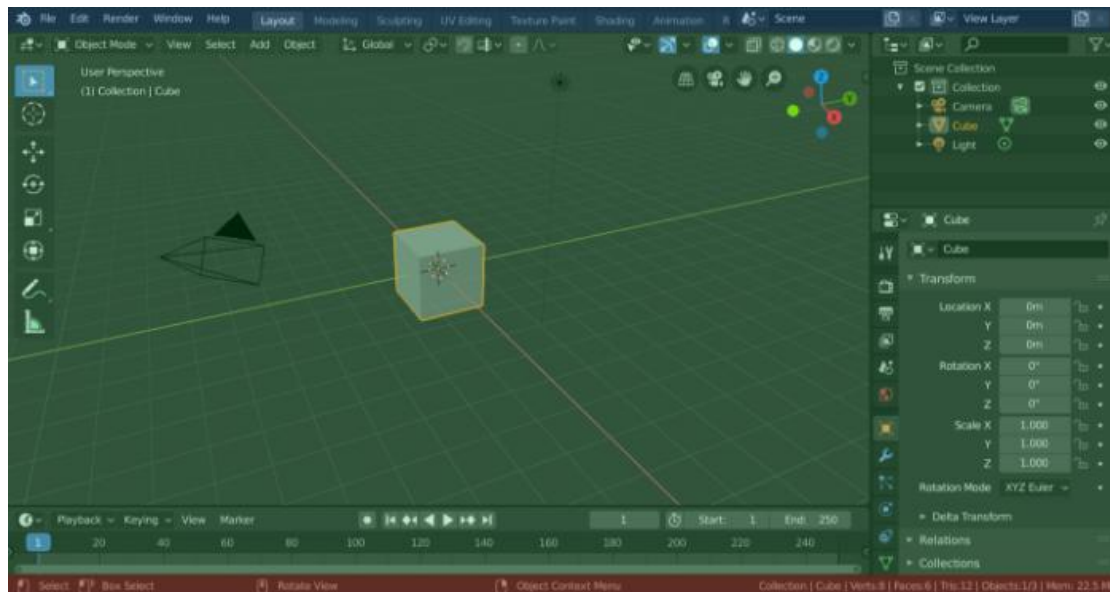
Figure3.3 Blender's default Screen Layout. Topbar (blue), Areas (green) and Status Bar (red).

3.23 Workspaces

● Workspaces

Workspaces are essentially predefined window layouts. Blender's flexibility with Areas lets you create customized workspaces for different tasks such as modeling, animating, and scripting. It is often useful to quickly switch between different workspaces within the same file.

Figure3.4 Workspaces are located at the Topbar.

● Controls

Tabs: Click on the tabs titles to switch between the workspaces. To cycle between workspaces use Ctrl-PageUp and Ctrl-PageDown. Double click to rename the workspace.

Add +: Click on the Add button to add a new workspace.

Context menu RMB: The context menu contains options to duplicate, delete and reorder workspaces.

● Default Workspaces

Blender's default startup shows the "Layout" workspace in the main area. This workspace is a general workspace to preview your scene and objects and contains the following Editors:

➢ 3D View on top left.
➢ Outliner on top right.

➢ Properties editor on bottom right.
➢ Timeline on bottom left.
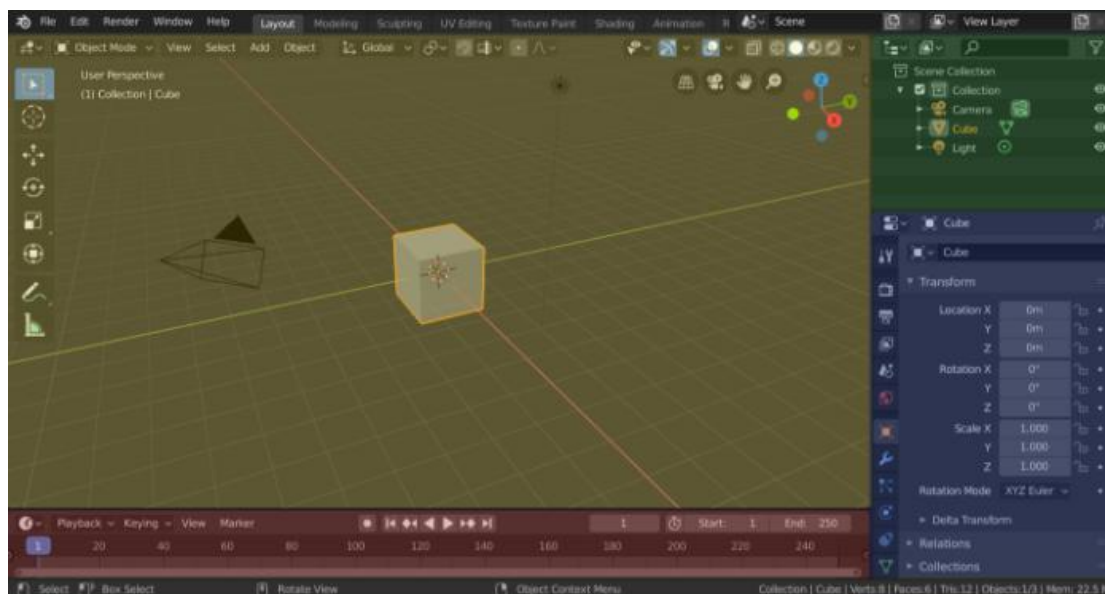
Figure3.5 Blender's Layout Workspace with four editors.
3D View (yellow), Outliner (green), Properties editor (blue) and Timeline (red).

Blender also has several other workspaces added by default:

Modeling: For modification of geometry by modeling tools.
Sculpting: For modification of meshes by sculpting tools.
UV Editing: Mapping of image texture coordinates to 3D surfaces.
Texture Paint: Tools for coloring image textures in the 3D View.
Shading: Tools for specifying material properties for rendering.
Animation: Tools for making properties of objects dependent on time.
Rendering: For viewing and analyzing rendering results.
Compositing: Combining and post-processing of images and rendering information.
Scripting: Programming workspace for writing scripts.

3.24 Status Bar

● The Status Bar is located at the very bottom of the Blender window and is used to present information. The Status Bar can be hidden by dragging it down or from the Window Menu.

Figure3.6 Status Bar.

● Keymap Information
   The region to the left of the Status Bar displays information about what mouse buttons and keys do for each editor, mode and tool, and it updates as you hold modifier keys.

● Status Messages

16

The region in the middle of the Status Bar displays information about currently running tasks or tools.

● Running Tasks

A progress bar and a cancel button are shown while doing longer computational tasks such as rendering or baking. Hovering over them shows a time estimate.

● Report Message

Label for an operator to display results or warnings. It disappears after a short time. Clicking on the icon on the left side of the report message will open the Info Editor which shows the full message.

3.25 Areas

● Areas.

The Blender window is divided up into a number of rectangles called Areas. Areas reserve screen space for Editors, such as the 3D Viewport, or the Outliner. In general an Editor provides a way to view and modify your work through a specific part of Blender. All hotkeys you press will affect the contents of the Editor in the Area the mouse pointer is located. Area boundaries are indicated by rounded (beveled) corners.

Areas can be customized to match specific tasks called Workspaces, which can then be named and saved for later use.
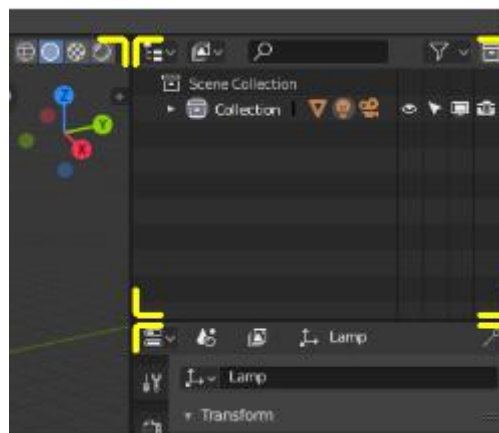
Figure3.7 Area boundaries are indicated by rounded corners (yellow highlights).

● Resizing

You can resize areas by dragging their borders with LMB. Move your mouse cursor over the border between two areas, so that the cursor changes to a double-headed arrow, and then click and drag.

Figure3.8   Move the mouse to change the size.

- Splitting.

    Splitting an area will create a new area. Placing the mouse cursor in an area corner will change the cursor to a cross (+) to indicate that pressing down LMB will activate splitting or joining operator. Dragging from area corner inward will split the area. You define the split direction by dragging either horizontally or vertically.
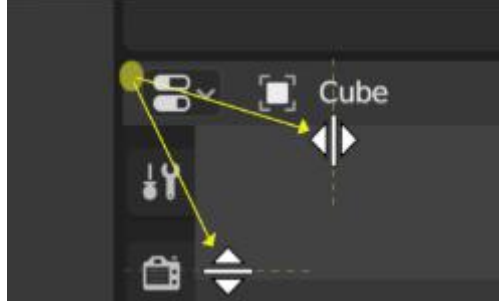
Figure3.9　Move the mouse to split.

- Joining

    Dragging from an area corner outward will join two areas. The area that will be closed gets a dark overlaid arrow. You can select which area will be closed by moving the mouse over areas. Release the LMB to complete the join. If you press Esc or RMB before releasing the mouse, the operation will be aborted.
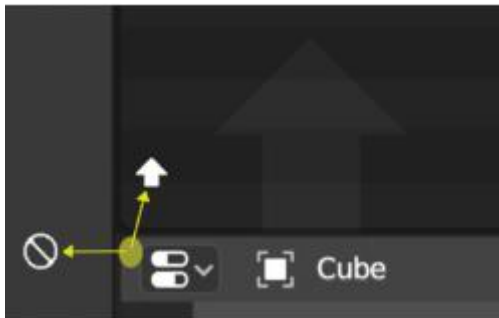
Figure3.10　Move the mouse to Joining.

3.26 Regions

    Every Editor in Blender is divided into Regions. Regions can have smaller structuring elements like tabs and panels with buttons, controls and widgets placed within them.
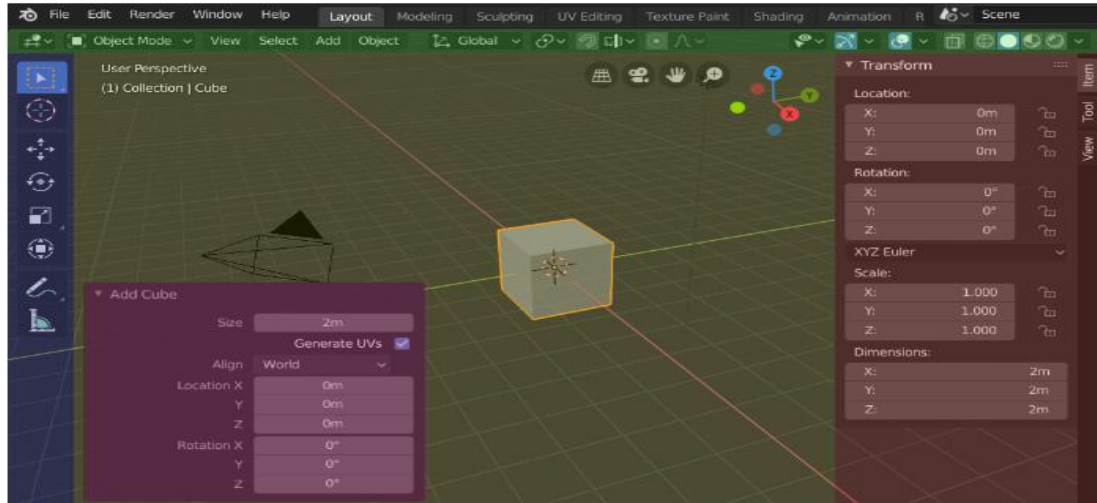
Figure3.11 Header (green), Main region (yellow), Toolbar (blue), Sidebar (red) and Adjust Last Operation panel (pink).

● Main Region.

At least one region is always visible. It is called the Main region and is the most prominent part of the editor.

Each editor has a specific purpose, so the main region and the availability of additional regions are different between editors. See specific documentation about each editor in the Editors chapter.

● Header

A header is a small horizontal strip, which sits either at the top or bottom of an area. All editors have a header acting as a container for menus and commonly used tools. Menus and buttons will change with the editor type and the selected object and mode.

Figure3.12 The Header of the 3D Viewport.

● Toolbar

The Toolbar (on the left side of the editor area) contains a set of interactive tools. T toggles the visibility of Toolbar.

3.27 Tabs & Panels

● Tabs

Tabs are used to control overlapping sections in the user interface. Contents of only one Tab is visible at a time. Tabs are listed in Tab header, which can be vertical or horizontal.

● Switching/Cycling

Vertical tabs can be switched with Ctrl-Wheel from anywhere in the region, and horizontal tabs with mouse cursor over tab headings.

You can also cycle through tabs with Ctrl-Tab and Shift-Ctrl-Tab, or press down

LMB and move mouse over tab header icons.
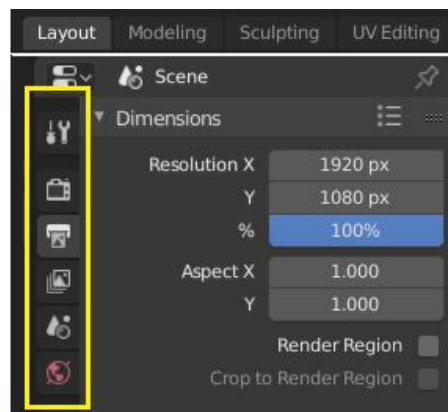
Figure3.13 Top: Horizontal Tab header in the Topbar. Bottom: Vertical Tab header shows tab icons in the Properties Editor.

● Panels

The smallest organizational unit in the user interface is a panel. Panel header is always visible, and it shows the title for the panel. Panels may also include sub-panels.

● Collapsing and Expanding.

A panel can either be expanded to show its contents, or collapsed to hide its contents. An expanded panel is indicated by a down-arrow (▼) in the panel header, while a collapsed panel is shown with a right-arrow (►).

Figure3.14 Panels in Properties editor.

3.28 Editors

Blender provides a number of different editors for displaying and modifying different aspects of data.The Editor Type selector, the first button at the left side of a header, allows you to change the editor in that area. Every area in Blender may contain any type of editor and it is also possible to open the same type multiple times.
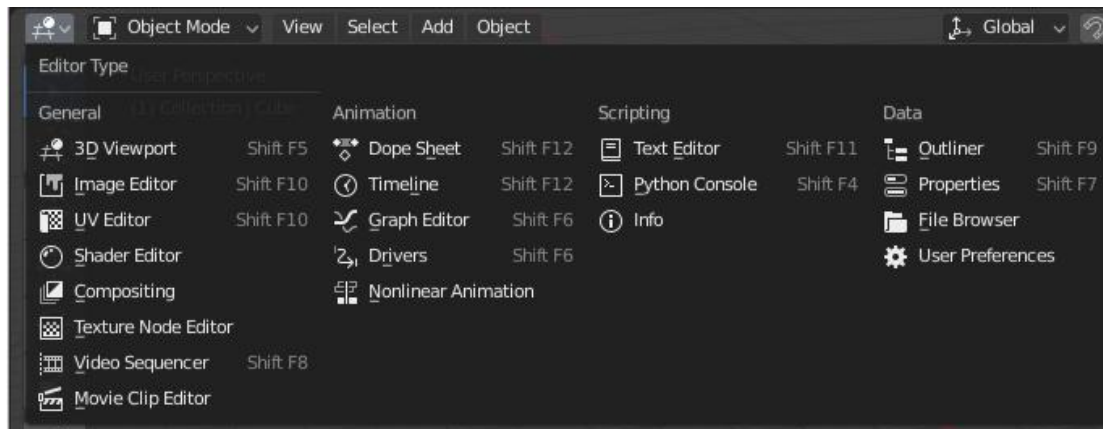
Figure3.15 The Editor Type selector.

- Object Modes.

Modes are an object-oriented feature, which means that the available modes vary depending on the selected active object's type–most of them only enable the default Object Mode (like cameras, lights, etc.).

Each mode is designed to edit an aspect of the selected object. See Tab. Blender's Modes below for details.
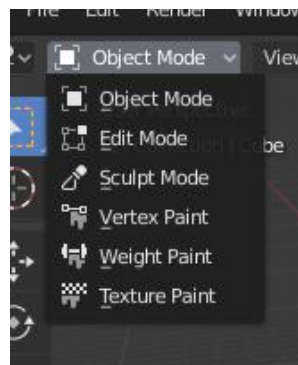
Figure3.16 The Mode select menu.

Modes can affect many things in Blender:
- They can modify the panels and/or controls available in some Properties editor tabs.
- They can modify the behavior of the whole editor, like e.g. the UV Editor and 3D View.
- They can modify the available header tools (menus and/or menu entries, as well as other controls…). For example, in the 3D Viewport, the Object menu in Object Mode changes to a Mesh menu in Edit Mode (with an active mesh object!), and a Paint menu in Vertex Paint Mode…
- They can modify the available shortcuts.

Icon	Name	Details
	Object Mode	The default mode, available for all object types, as it is dedicated to *Object* data-block editing (e.g. position, rotation, size).
	Edit Mode	A mode available for all renderable object types, as it is dedicated to their "shape" *Object Data* data-block editing (e.g. vertices/edges/faces for meshes, control points for curves/surfaces, strokes/points for Grease Pencil, etc.).
	Sculpt Mode	A mesh-only mode, that enables Blender's mesh 3D-sculpting tool.
	Vertex Paint Mode	A mesh-only mode, that allows you to set your mesh's vertices colors (i.e. to "paint" them).
	Weight Paint Mode	A mesh-only mode, dedicated to vertex group weighting.
	Texture Paint Mode	A mesh-only mode, that allows you to paint your mesh's texture directly on the model, in the 3D Views.
	Particle Edit Mode	A mesh-only mode, dedicated to particle systems, useful with editable systems (hair).
	Pose Mode	An armature only mode, dedicated to armature posing.
	Draw Mode	A Grease Pencil only mode, dedicated to create Grease Pencil strokes.

Figure3.17 Object Mode List

Chapter 4 Modeling

4.1 Modeling of character and building

The paper's modeling project includes architecture and characters. This chapter will introduce the modeling process of the project. The building is modeled with a "little castle" as an example. First, we will open a new project interface, and then proceed. As shown, this is the initial interface.
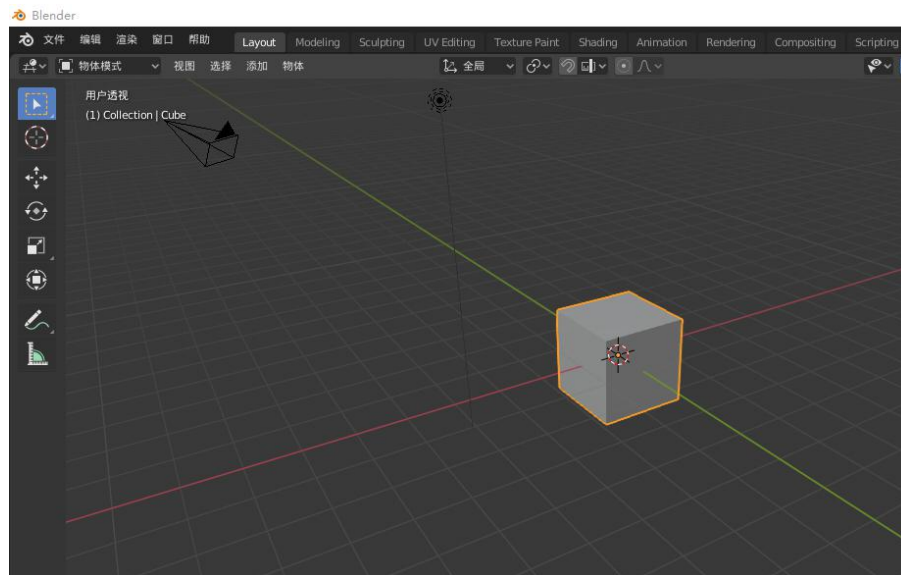
Figure4.1 Initial Interface

We will add the basic object elements we need in "Object Mode". In this mode, we can perform basic operations such as "moving" and "rotating" the object's spatial position to achieve our needs. Then we started to do the first modeling.

Then we will enter "Edit Mode" to manipulate the shape of the object element.

（1）We can change the shape of the object by "stretching".
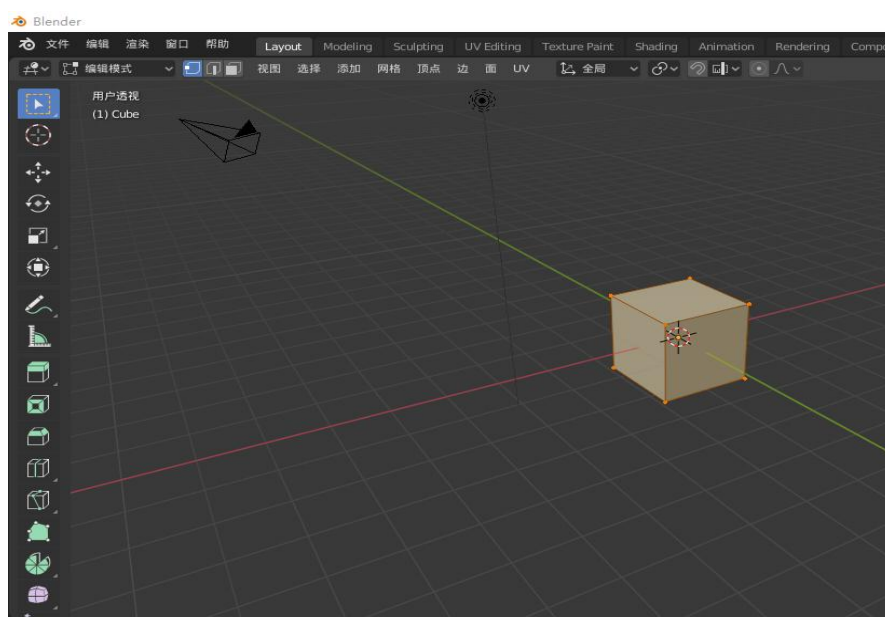
Figure4.2 Edit Mode
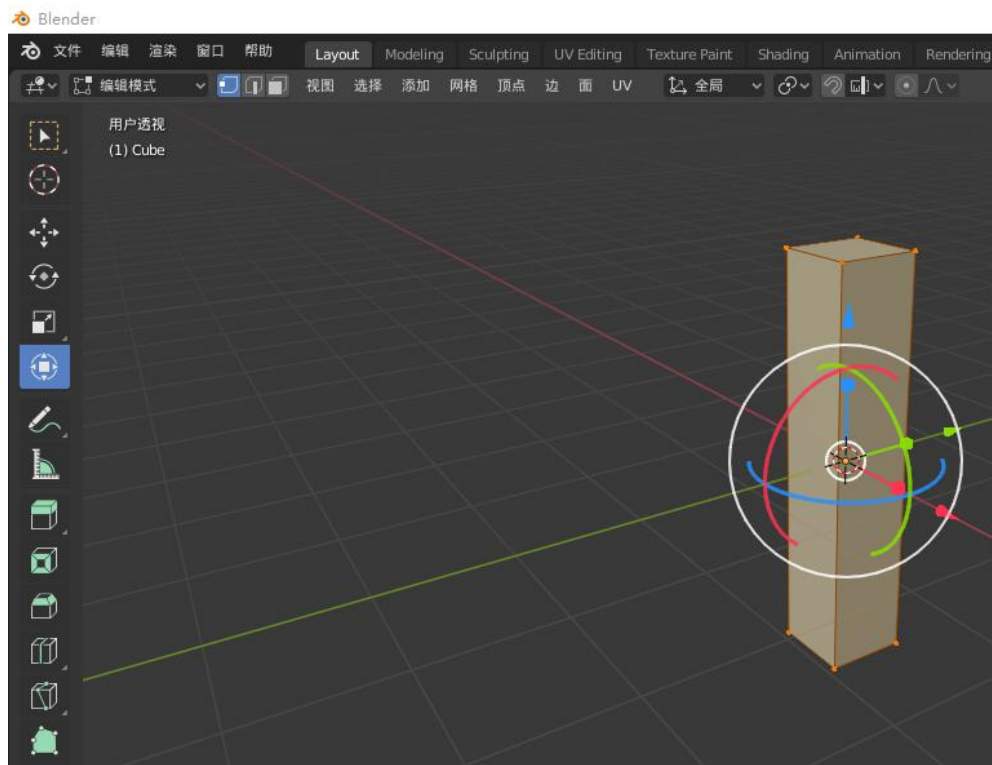
Figure4.3 Change shape

（2）After the size is completed, we add two new object elements. Also, modify the size we need. Then move to our proper location and combine.
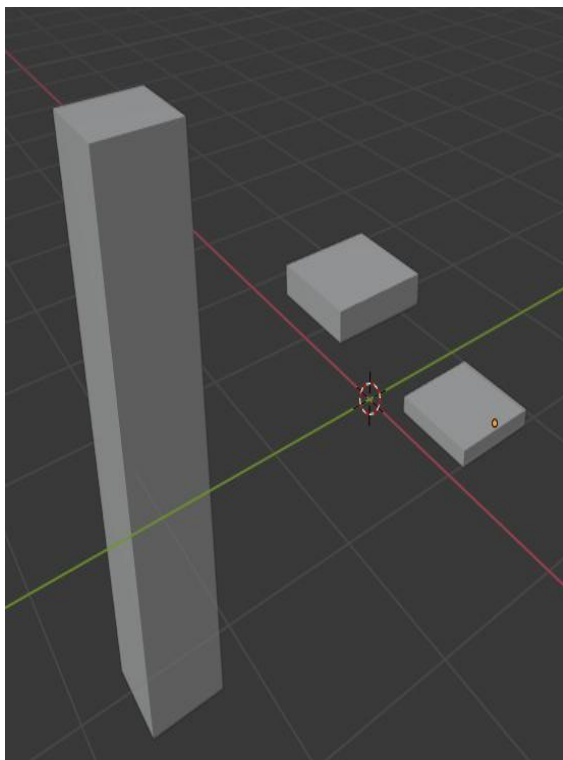
Figure4.4 Add Object
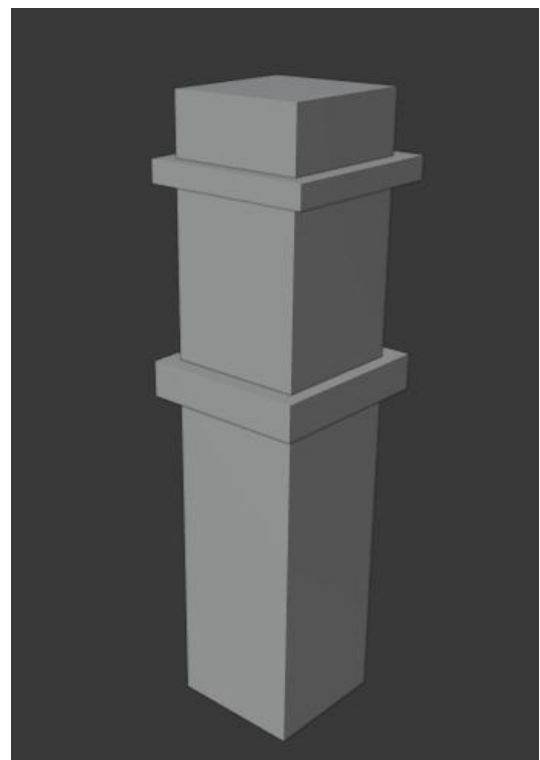
Figure4.5 Object Combination

（3）In this step, we will use the "modifier" to "cut" the object to change the shape. Similarly, we need to add two new object elements and move the new object to a certain position.
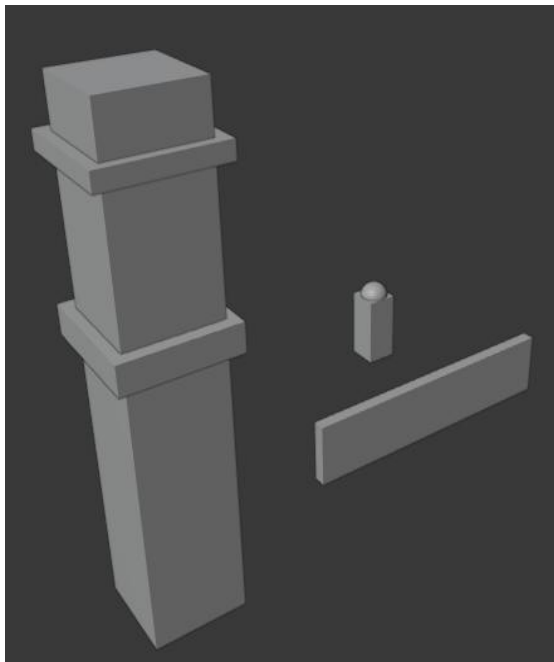
Figure4.6 Add Object
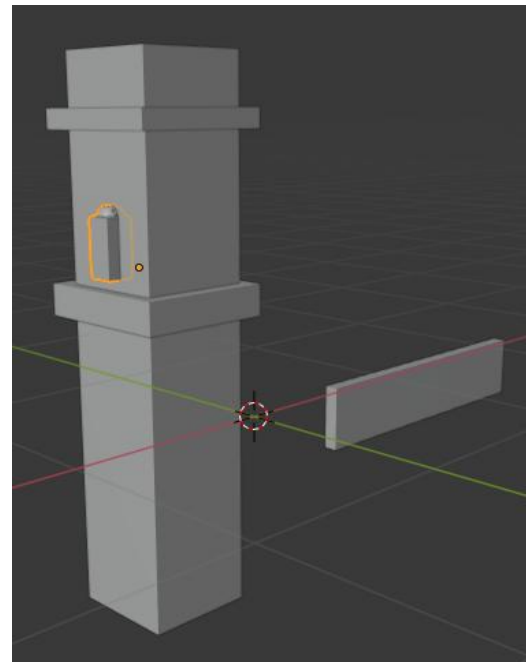
Figure4.7 Object Combination

Then use the "Boolean modifier", select the corresponding object, set the parameters, and remove the parts we don't need.
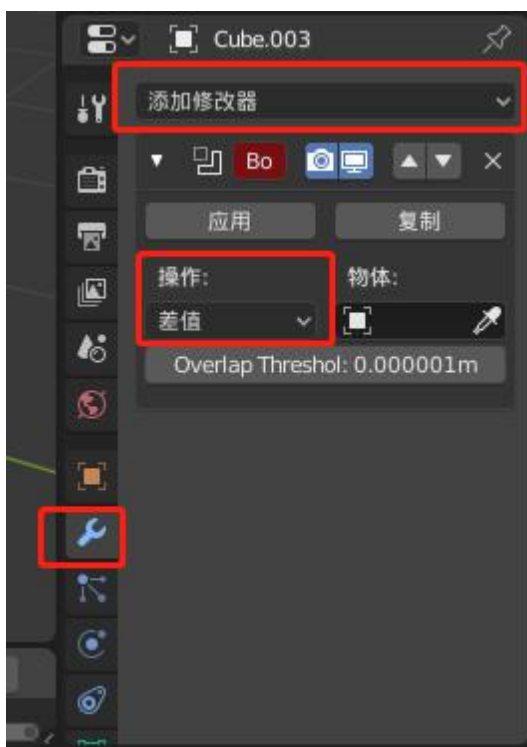
Figure4.8 Set Modifier
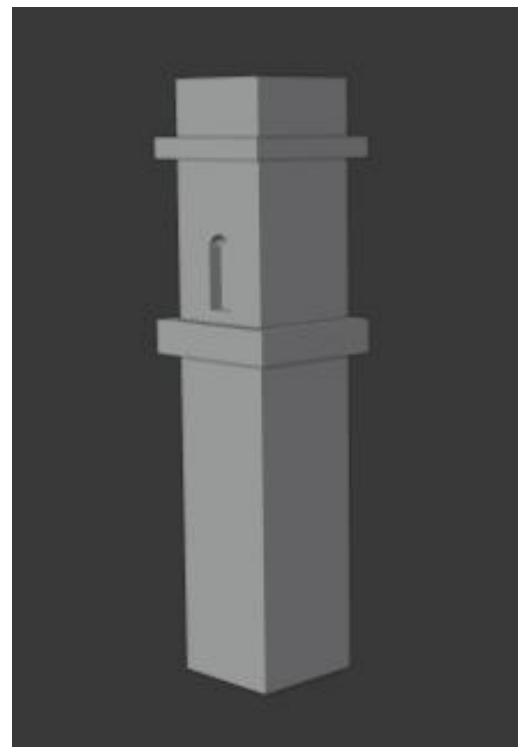
Figure4.9 Overall Outline

Do the same for another object, and the basic shape is ready. Finally, make some minor adjustments and modifications to the corners and positions, so the first "small castle" modeling has been completed.
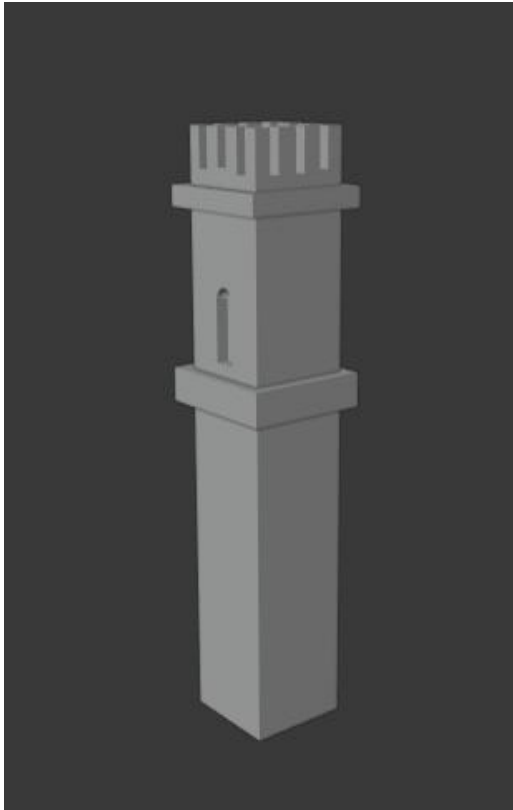
Figure4.10 Overall Outline
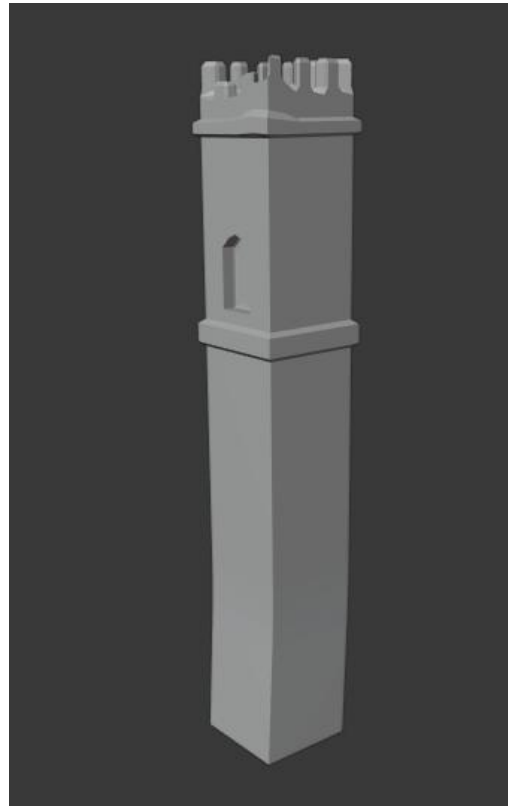
Figure4.11 Finish

4.2 Results display

Because there are a lot of items for modeling, the basic modeling operation steps are similar. Therefore, instead of enumerating the processes one by one, the final results are directly displayed. As these picture shows.
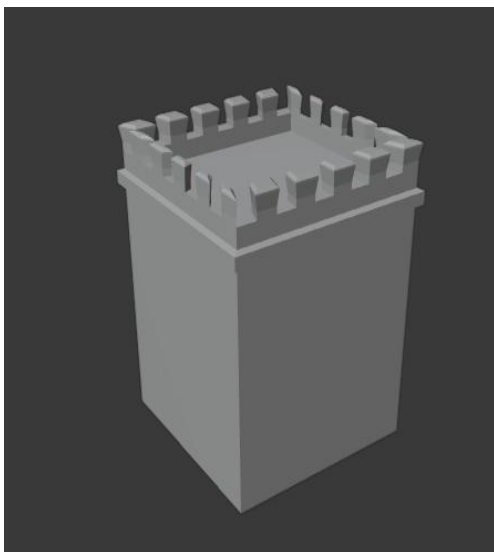
Figure4.12
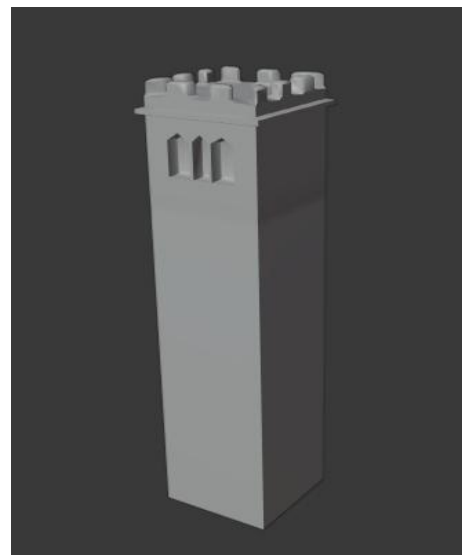
Figure4.13
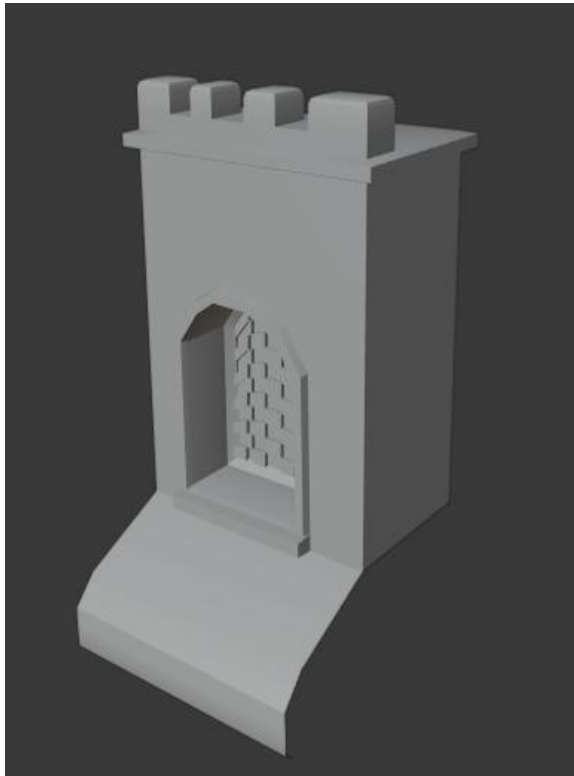
Figure4.14
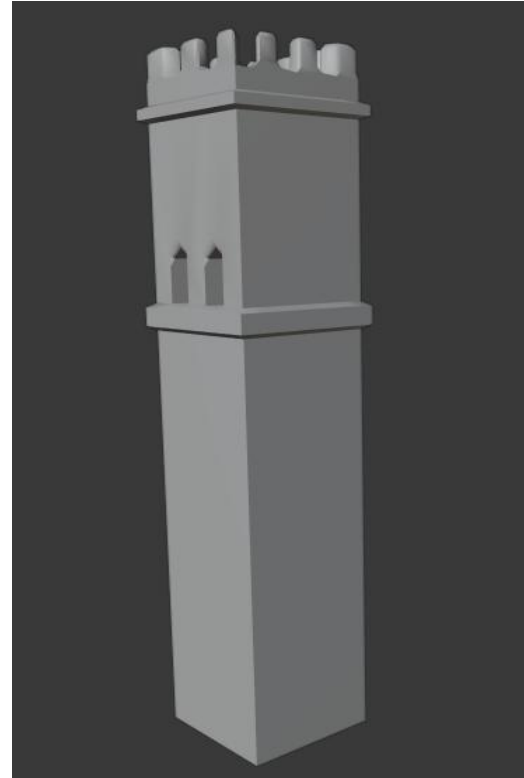
Figure4.15

Figure4.16

Figure4.17

Figure4.18

Figure4.19

Figure4.20

Figure4.20
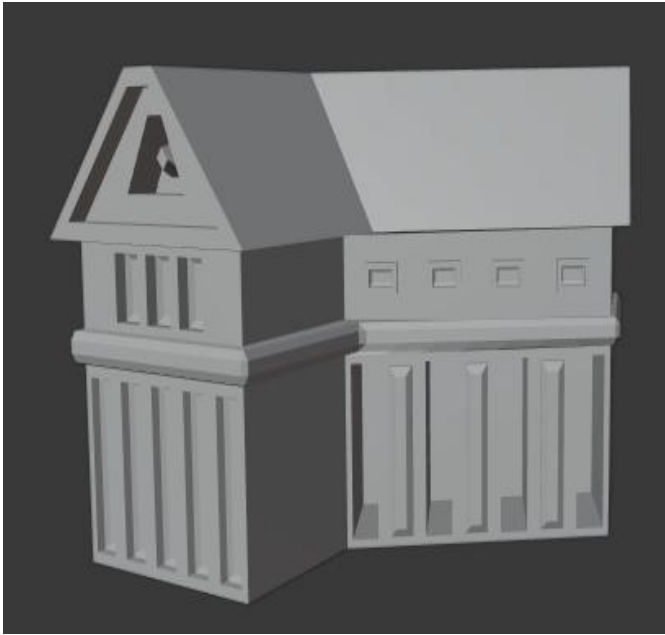
Figure4.21

Figure4.22

Figure4.23

Figure4.24
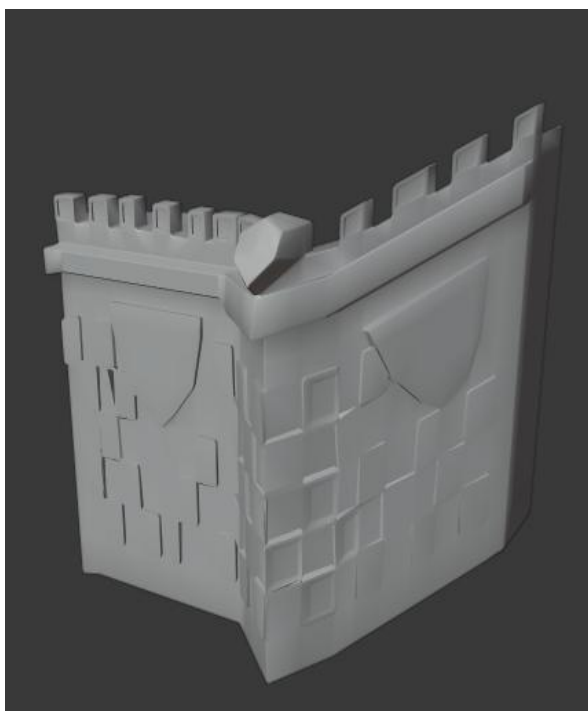
Figure4.25

Figure4.26
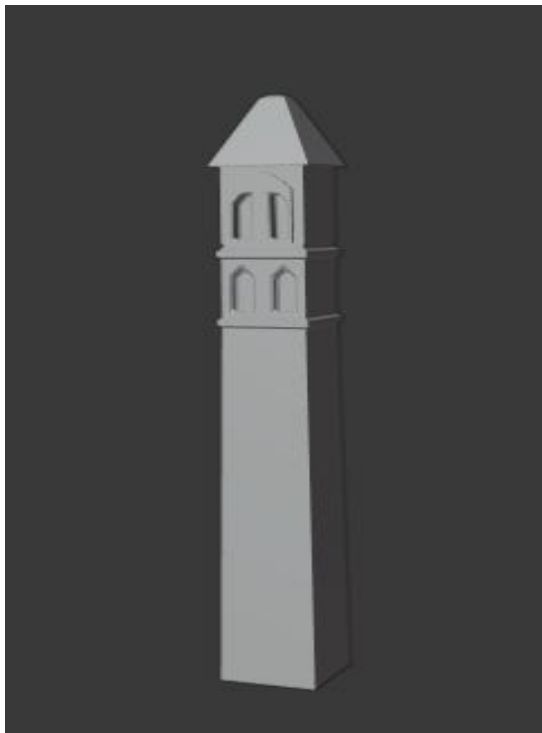

Figure4.27

Figure4.28

Figure4.29

Figure4.30

Figure 4.31

4.3 Carving of character

We all know that character modeling is indispensable and very important in all modeling. Next, the process and method of character sculpting will be briefly introduced.

（1）First, we will enter the Blender's "sculpt mode" and create a basic object element we need.
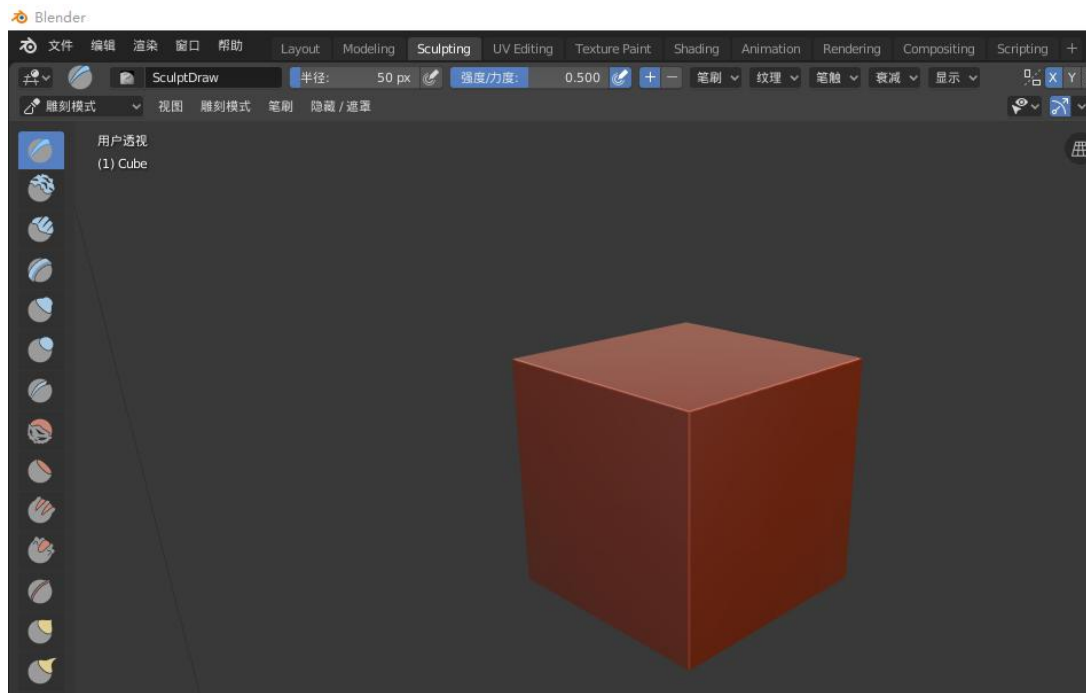
Figure 4.32 Sculpt Mode

（2）Set the parameters of the "modifier" to change the shape of the object, from a cube to a sphere[10].
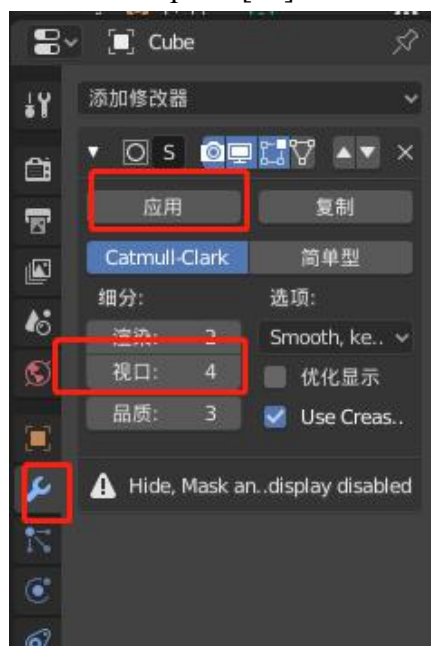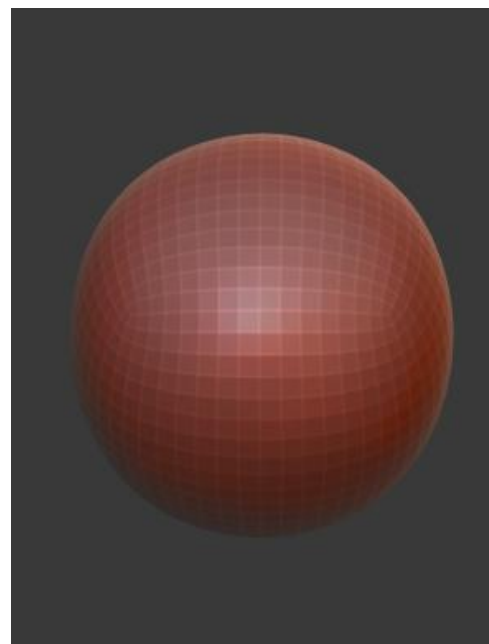

Figure 4.33 Modifier

Figure 4.34 Sphere

（3）Then, we will use various "brush" tools to sculpt the object, sculpting step by step into the "head" of the character we want. You can also set the parameters we need in the "modifier".
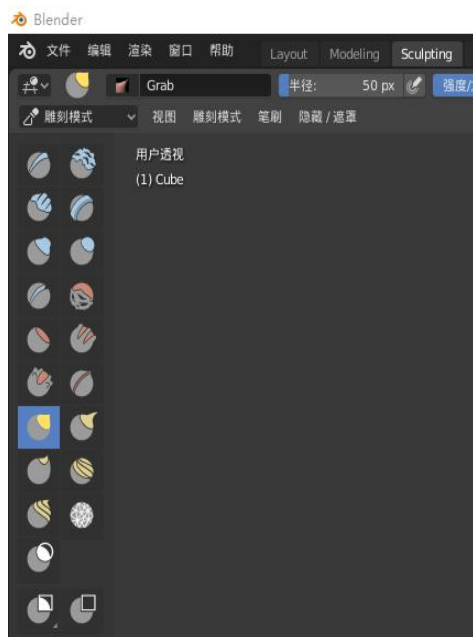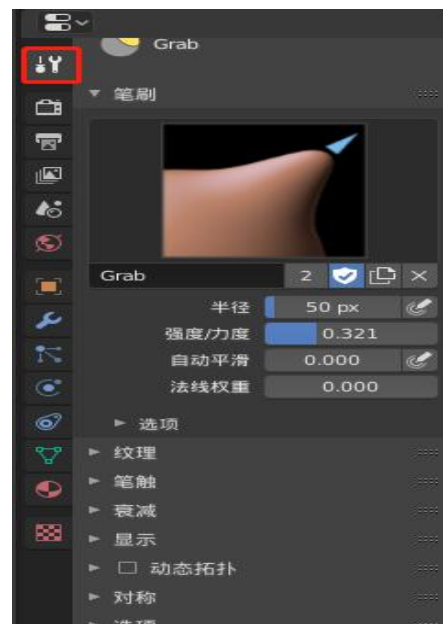

Figure 4.35 brush tools

Figure 4.36 Modifier

Select the correct "brush" and start sculpting. Usually, for convenience, we sometimes need to open multiple window views. In this way, we can sculpt more intuitively and from multiple perspectives. Of course, the requirements vary according to the needs, and the time it takes varies.
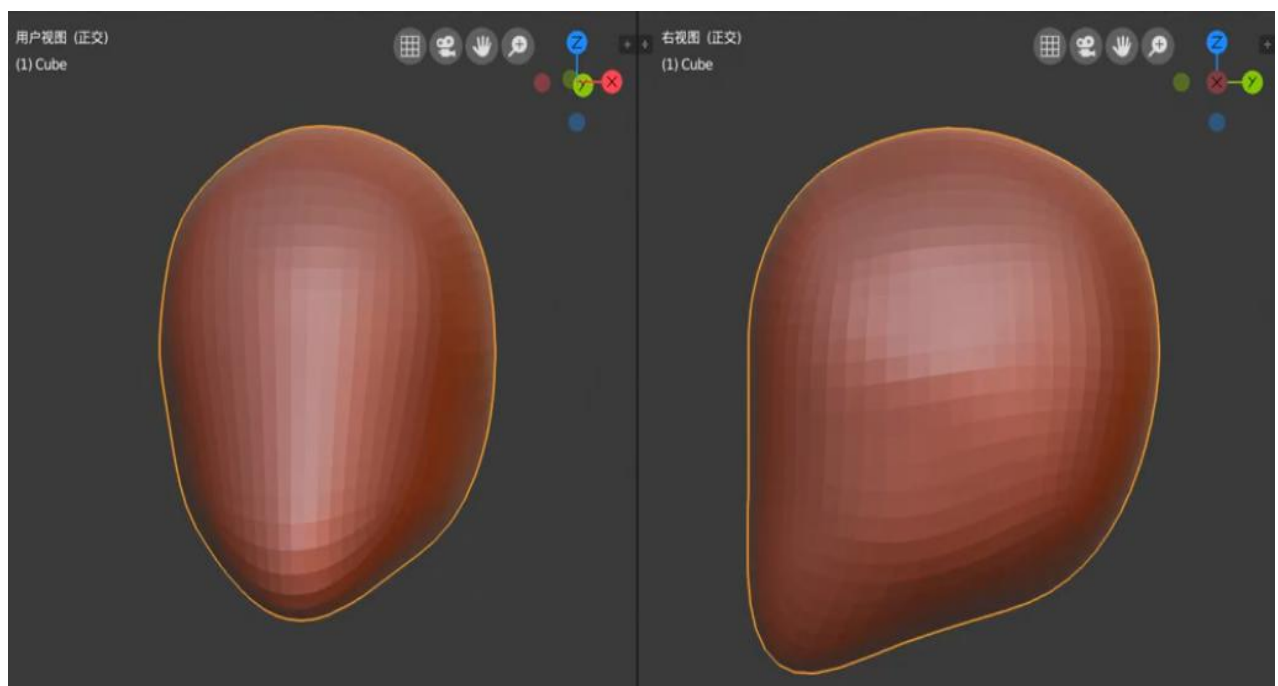

Figure 4.37 Contour

（4）Through the "brush" function, we perform "stretch" and "squeeze" operations on objects to sculpt the appearance shape we need. Use the "modifier" to add the extra objects we need, and then sculpt until we are done. Therefore, the head of the character is basically completed.
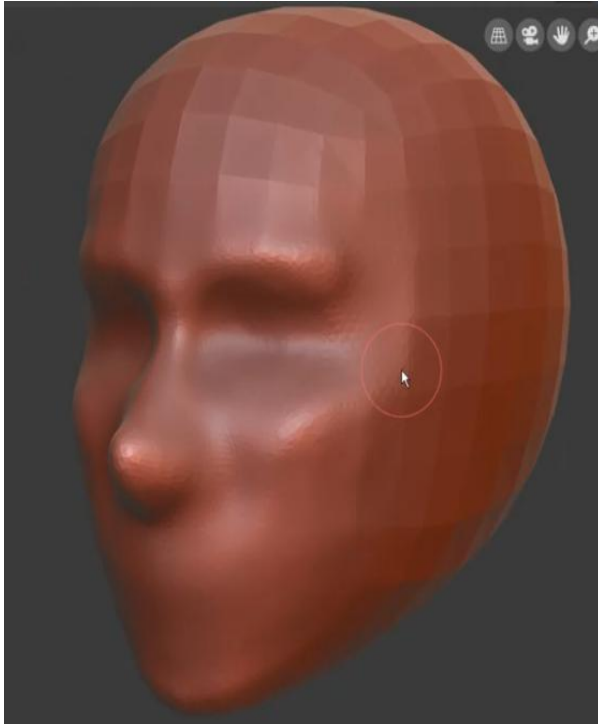

Figure 4.38 Shape The Head   Figure 4.39 Finish

（5）Body shaping. The operation steps are basically similar, we add basic object elements, sculpt the basic outline, and then sculpt the details step by step.
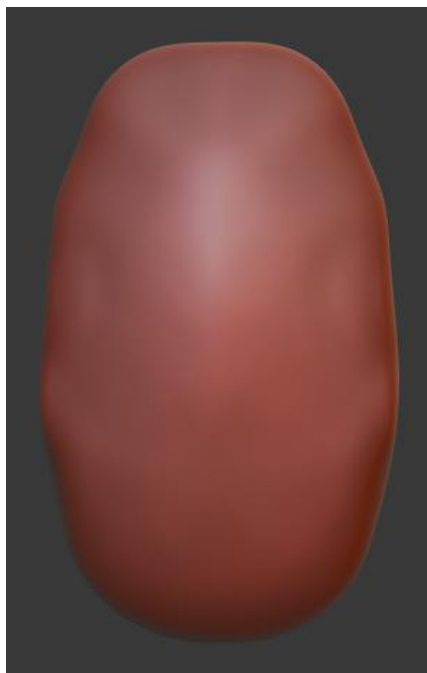

Figure 4.40 Shape The Body   Figure 4.41 Finish

4.4 Combination of characters

Similarly, there are a large number of characters in the project, and we will show you the results directly. As shown in the figure, this is the result of one of the character combinations.
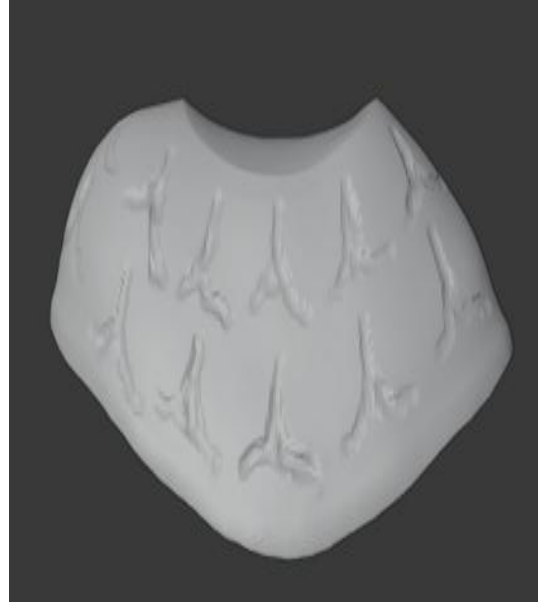

Figure 4.40 Head

Figure 4.41 Shape The Body

Figure 4.42 Body

Figure 4.43 Finish

4.5 Full results and finished drawings of the first project work.

Figure 4.44

Figure 4.45

Figure 4.46

Figure 4.47

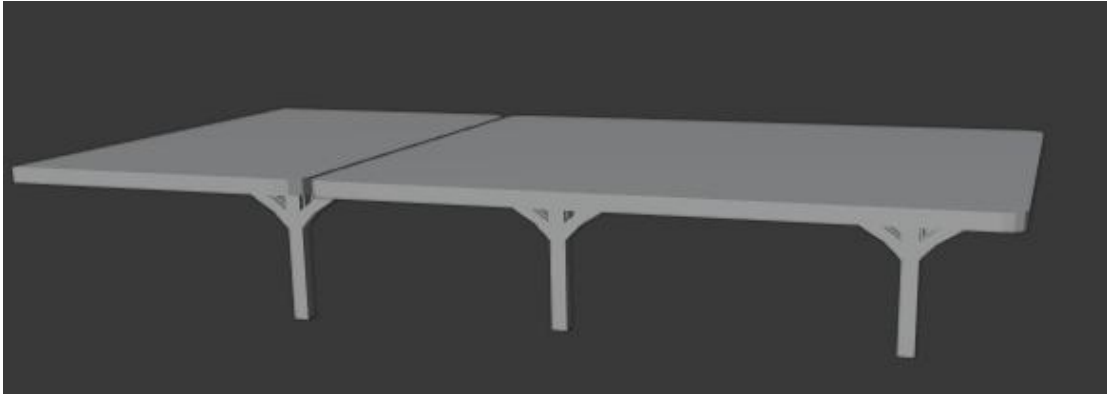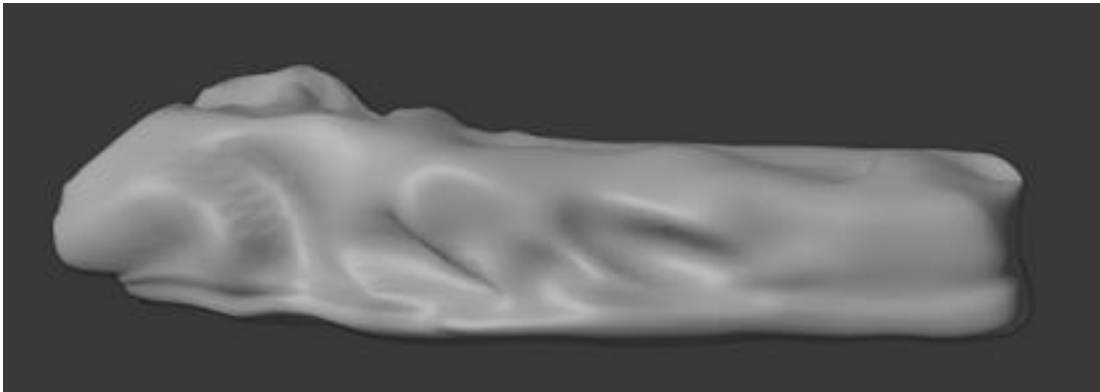Figure 4.48

Figure 4.49

Figure 4.50

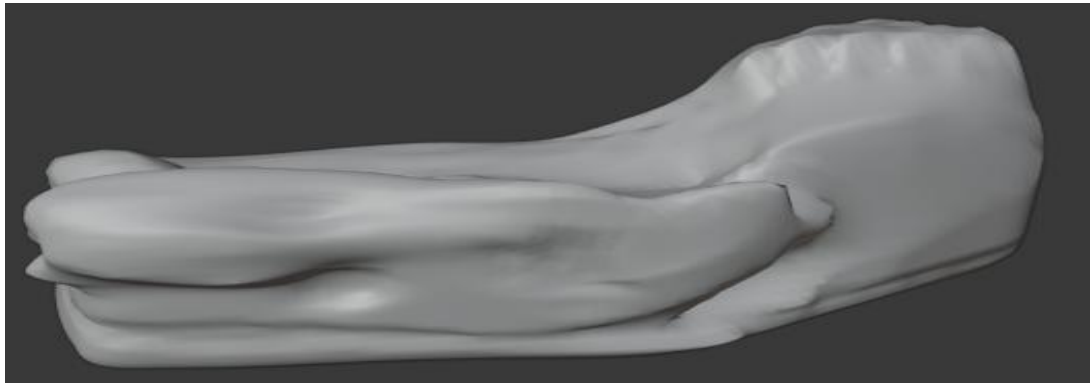Figure 4.51

Figure 4.52

Figure 4.53

Figure 4.54
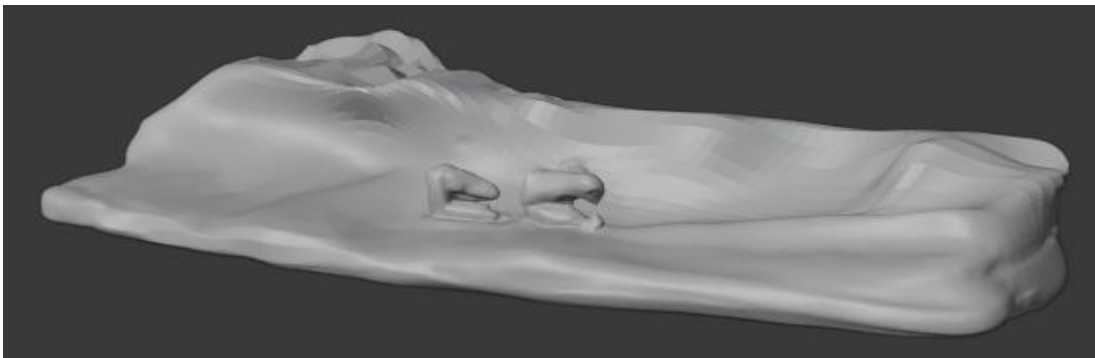
Figure 4.55 All Combinations

4.6 Full results and finished drawings of the second project work.

Figure 4.56

Figure 4.57

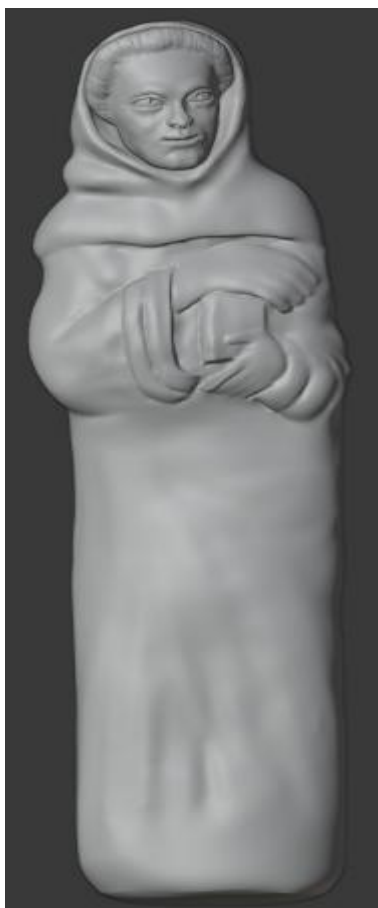Figure 4.58

Figure 4.59

Figure 4.60

Figure 4.61
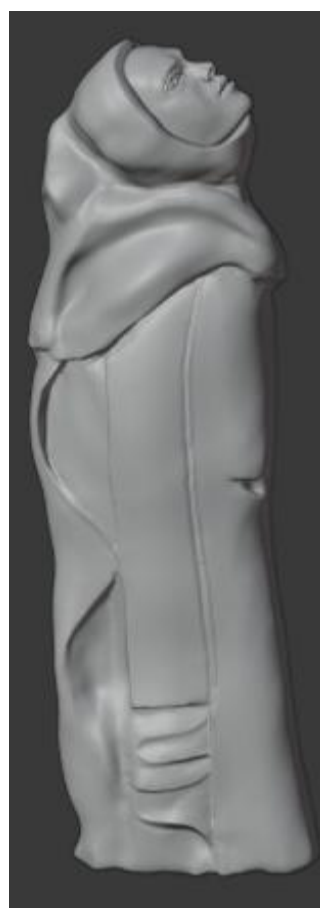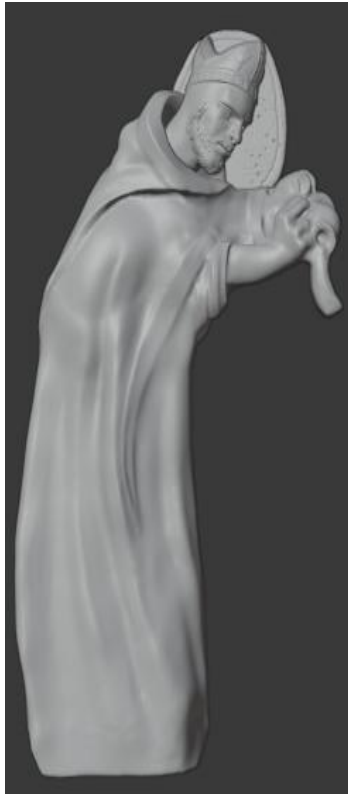

Figure 4.62

Figure 4.63

Figure 4.64

Figure 4.65   Figure 4.66   Figure 4.67
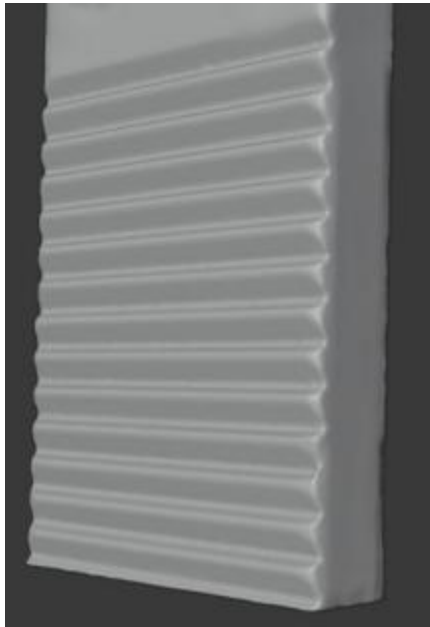

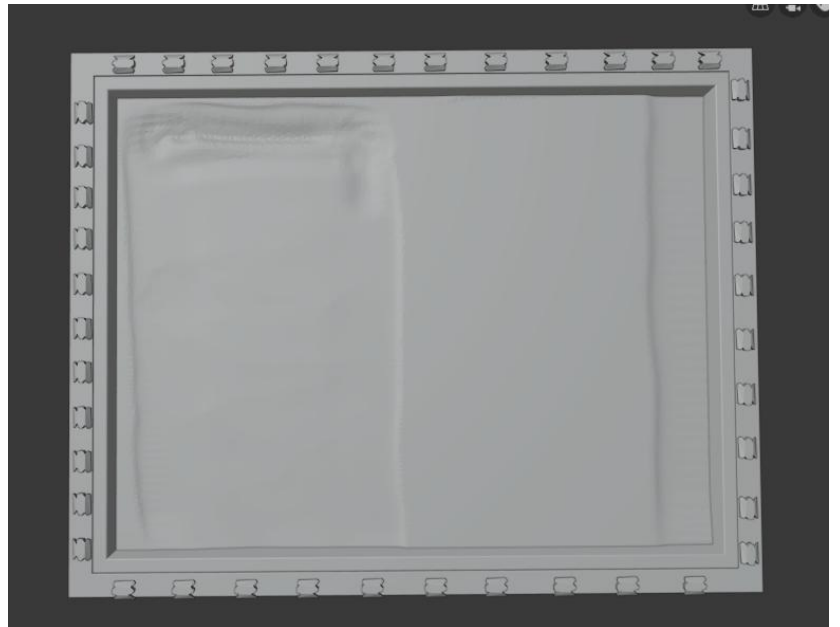Figure 4.68   Figure 4.69

Figure 4.70 All Combinations

Chapter 5 Rendering

5.1 Introduction of Rendering

Rendering is the process of turning a 3D scene into a 2D image. Blender includes three render engines with different strengths:

- Eevee is a physically based realtime renderer.
- Cycles is a physically based path tracer.
- Workbench is designed for layout, modeling and previews.

More renderers from third-party developers are available as add-ons. Each renderer has its own render settings to control render quality and performance. What the render looks like is defined by cameras, lights and materials. These are shared between Eevee and Cycles, however some features are only supported in one or the other. Renders can be split up into layers and passes, which can then be composited together for creative control, or to combine with real footage. Freestyle can be used to add non-photorealistic line rendering.

Blender supports interactive 3D viewport rendering for all render engines, for quick iteration on lighting and shading. Once this is done, the final quality image or animation can be rendered and output to disk.

5.2 Cycles

Cycles is Blender's physically-based path tracer for production rendering. It is designed to provide physically based results out-of-the-box, with artistic control and flexible shading nodes for production needs.

5.3 Gpu rendering

GPU rendering makes it possible to use your graphics card for rendering, instead of the CPU. This can speed up rendering because modern GPUs are designed to do quite a lot of number crunching. On the other hand, they also have some limitations in rendering complex scenes, due to more limited memory, and issues with interactivity when using the same graphics card for display and rendering[6].

5.4 Rendering result graph

Because the personal computer performance is not good, the best rendering effects cannot be used, otherwise the computer will stop working immediately. Only some parameters can be reduced to get the rendering result.

Figure 5.1 Rendering Result

Figure 5.2 Rendering Result

CONCLUSION

The first part of this thesis introduces some of the basics of 3D modeling and applications in many fields. 3D modeling can be applied in the fields of architecture, film and television, games, arts, industrial manufacturing, online video, e-reading, virtual communities, e-commerce, distance education and other fields. Its appearance can help people understand and use objects better and more clearly and intuitively. With the advent of the 5G era, 3D modeling technology continues to advance, and its functions will become more powerful.

The second part focuses on the basic use of blender in 3D modeling software, including:

(1) The specific production process steps and skills used in the two 3D modeling projects.

(2) All project results are finally displayed.

Now that 3D modeling has been applied to many fields, 3D modeling software has also appeared a lot. Of course, each software has its advantages and disadvantages, as well as its unique features. By completing this project, I learned a lot of knowledge and strengthened my practical ability. As a beginner, I learn about Blender software while completing project tasks. Although, by comparison, my methods and techniques seem awkward, I have been liking it since I got into the Blender software. It has taken me into another world, and I will continue to learn more about 3D modeling and hope to contribute my strength to society.

ACKNOWLEDGEMENT

First of all, I would like to thank my professor Cantoni Virginio. He is a professor that all our classmates like because he is really a very good person and a very good professor. In the classroom, he taught me a lot of knowledge. In the homework project, because my English is not very good, I did not do well in many places, but he never blamed me, but gave me a lot of encouragement and taught me some learning methods. In the thesis project, under his guidance, my project is getting better and better. He is not only my professor, but also my life mentor.

Thanks to all my dear classmates. I am an introverted person. Whenever I encounter difficulties or feel bad, I will not easily tell others. But they always help me and encourage me like a brother or sister. They can make me happy again and build confidence to solve difficulties. At the same time, they taught me how to learn better. They are my very important classmates and friends. Without them, there would be no me now. Thank you very much.

REFERENCES

[1] https://en.wikipedia.org/wiki/3D_modeling.

[2] https://en.wikipedia.org/wiki/Blender_(software).

[3] "Reference/Release Notes/2.80/UI - Blender Developer Wiki". Blender Developer Wiki. 26 June 2019. Retrieved 11 July 2019.

[4] "System Requirements". Blender.org. Retrieved March 29, 2016.

[5] "Introduction — Blender Reference Manual". www.blender.org. Retrieved 2015-10-18.

[6] "GPU Rendering — Blender Reference Manual". www.blender.org. Retrieved 2015-10-18.

[7] "Features — Blender Reference Manual". www.blender.org. Retrieved 2016-03-24.

[8] "Integrator — Blender Reference Manual". www.blender.org. Archived from the original on 2015-10-09. Retrieved 2015-10-18.

[9] "External Renderers". Archived from the original on 2018-03-09. Retrieved 2018-06-14.

[10] "Modifiers". blender.org. Retrieved 2019-01-14.

[11] "New Particle options and Guides". Blender.org. Retrieved 2012-07-06.

[12] "Blender History". Blender.org. Retrieved March 29, 2018.