VIRTUAL RECONSTRUCTION OF A SEVENTEENTH-CENTURY PORTUGUESE NAU

A Thesis

by

AUDREY ELIZABETH WELLS

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

August 2008

Major Subject: Visualization Sciences

VIRTUAL RECONSTRUCTION OF A SEVENTEENTH-CENTURY PORTUGUESE NAU

A Thesis

by

AUDREY ELIZABETH WELLS

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	Frederic Parke
Committee Members,	Filipe Castro
	Carol LaFayette
Head of Department,	Tim McLaughlin

August 2008

Major Subject: Visualization Sciences

ABSTRACT

Virtual Reconstruction of a Seventeenth-Century Portuguese Nau. (August 2008)

Audrey Elizabeth Wells, B.S., Mount Union College
Chair of Advisory Committee: Dr. Frederic Parke

This interdisciplinary research project combines the fields of nautical archaeology and computer visualization to create an interactive virtual reconstruction of the 1606 Portuguese vessel *Nossa Senhora dos Mártires*, also known as the Pepper Wreck. Using reconstruction information provided by Dr. Filipe Castro (Texas A&M Department of Anthropology), a detailed 3D computer model of the ship was constructed and filled with cargo to demonstrate how the ship might have been loaded on the return voyage from India. The models are realistically shaded, lighted, and placed into an appropriate virtual environment. The scene can be viewed using the real-time immersive and interactive system developed by Dr. Frederic Parke (Texas A&M Department of Visualization).

The process developed to convert the available information and data into a reconstructed 3D model is documented. This documentation allows future projects to adapt this process for other archaeological visualizations, as well as informs archaeologists about the type of data most useful for computer visualizations of this kind.

ACKNOWLEDGMENTS

Many people have contributed to the completion of this thesis and I wish to extend my sincere thanks to them.

Early in the project, Dr. Frederic Parke encouraged me to explore a variety of thesis topics, spurring me to branch out and discover something that I was truly interested in working on. He is also the impetus behind my use of the interactive and immersive system he designed, which gave me a sense of purpose while developing the models. Throughout the entire project, he provided reliable and helpful feedback.

Dr. Filipe Castro introduced me to the fascinating world of nautical archaeology and his extensive research of the Pepper Wreck. I found his enthusiasm for the Pepper Wreck infectious, and this zeal kept me going throughout the long and sometimes grueling process. His wonderful anecdotes about seafaring life were a source of inspiration for me throughout the project. I cannot thank him enough for the countless hours spent poring over even the smallest details of the project.

I also wish to thank Carol LaFayette for her encouragement and guidance. I greatly admire her artistic sensibility and value her opinion. Her enthusiasm for the project’s potential has been a source of motivation throughout the process.

Also deserving a mention are Ajay Ahuja and Jace Miller, who assisted me at various points to help me get my project working on the immersive system.

Finally, I wish to thank my fellow students and the staff of the Visualization Sciences Laboratory for their constant support and friendship. I hope I can find such comradeship wherever I may go.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

FIGURE Page

FIGURE Page

FIGURE Page

LIST OF TABLES

CHAPTER I

INTRODUCTION

> *We now live in a world where theory and practice are converging to make archaeology a study of virtual pasts where knowledge is constructed through the interactive evaluation of electronic bits and bytes.*
>
> Gary Lock [1]

Computer Modeling in Archaeology

Past human life and culture are not directly observable. Instead, we rely on archaeology, which involves the recovery and examination of remaining evidence. Computers have come to play a significant role in the study of archaeology. Use of computers can augment or replace traditional techniques for recording and organizing archaeological data. Computers can also enable new methods of analyzing and presenting archaeological research, such as through computer modeling.

The term *model* is used in many different contexts. A model can be fundamentally defined as a simplified version of something complex. Models are useful because they are often easier to work than dealing directly with a complex subject. Models vary greatly in form and intent, but all should contribute to our understanding and knowledge in some way. Abstract models, such as mathematical and statistical models, might take the form of tables, graphs, or simulations of numerical data. In contrast, physical models are tangible representations of objects or systems, such as a scale model of a building. Computer models are models that exist in digital form.

This thesis follows the style of *IEEE Transactions on Visualization and Computer Graphics.*

An archaeological reconstruction model can be used to envision how the past once was. This is useful because the subject of study is generally in partial remains. Models, and the process to create them, can have significant influence in archaeological interpretations. Gary Lock asserts, “Because the past is complex, often unknowable and unverifiable, working through models is the only way of approaching explanation and experimenting with the meaning of observed data” [1, p. 147]. Models can facilitate the analysis of specific research problems, as well as allow archaeologists to make predictions based upon the data or theory presented.

Modeling is a very powerful tool, which can allow viewers to progress from what is observable, archaeological data and theory, to concepts of what is unobservable – the past. Lock explains, “Moving from data to explanation through theory and interpretation has always been the endeavor of archaeology” [1, p. 7]. He also presents the use of computer modeling in archaeology as a *hermeneutic spiral*, or process of interpretation, in which the data model and theoretical model are derived from the archaeological record through interpretation. Digital computer models, informed by the data and theory, add an additional layer of interpretation. Based on the models, interpretive statements about the past are made. These interpretations constantly inform and reform our understanding of the past (see Fig. 1).

This project used three-dimensional (3D) computer graphics modeling to create an archaeological reconstruction of the Portuguese nau *Nossa Senhora dos Mártires*. This ship wrecked off the coast of Lisbon, Portugal in 1606. The archaeological remains of the ship have been excavated and interpreted, resulting in data (such as measurements of the hull remains) and theory (such as proposed reconstructed hull dimensions). The data and theory inform the creation of the digital model, which can then be used to make further interpretations. These new interpretations can bring about changes in the data, theory, and digital model indefinitely.

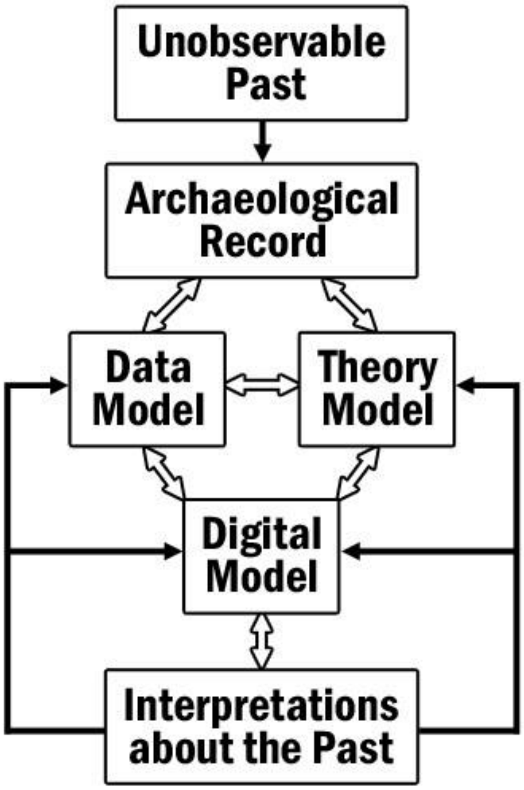

Fig. 1. The interpretive cycle using computer models in archaeology. The outlined arrows indicate interpretations made during the process. Adapted from [1, Fig. 1.1]

Introduction to Virtual Archaeology

Virtual archaeology is an application of computer visualization, usually referring to the use of 3D computer modeling to recreate archaeological sites or artifacts. Virtual archaeology can be most effective when there are limited remains existing for us to envision the past. Even when archaeological data is fragmented, which is frequently the case, virtual reconstructions may attempt to fill in the gaps through inference. These reconstructions can integrate the available information in a realistic, visual, and accessible way.

Traditional methods of publishing archaeological findings may present a fragmented understanding of the subject. Excavations generally result in many two-dimensional (2D) representations such as measured plans, cross-sections, and drawings of artifacts, which are isolated from their context. A series of such drawings and diagrams that document an archaeological site gives the viewer a very different understanding than that of the observer situated in and moving around the actual site or a reconstruction of the site. Virtual archaeology gives us the opportunity to move beyond two-dimensional representations to create a more complete and three-dimensional (3D) understanding by virtually examining archaeological sites and objects in their context. However, virtual archaeology should not necessarily replace traditional methods of presenting research, but should supplement it.

Although it is possible for archaeologists to create these visualizations themselves, it could be argued that the most successful applications of virtual archaeology are the result of collaborations between archaeologists and visualization specialists. Gary Lock affirms, "The representation of the smoky and poorly lit interior of an ancient building, the natural textures of wood and thatch, are the sorts of computer graphics research problems that fit nicely with archeological reconstructions so it is hardly surprising that the two disciplines have formed productive partnerships" [1, p. 154]. However, during collaborations of this kind, frequent communication between those with the archaeological knowledge and those with the visualization skills is vital. This will ensure that the visualization maintains archaeological validity and accuracy.

The process of constructing a 3D model can be a highly creative and iterative process of archaeological research. Models expose weaknesses in the interpretation of reconstructed spaces by forcing the archaeologist to envision an entire conjectural model and judge the plausibility of every corner, detail, or structural solution. It is easy for an archaeologist to overlook these details when looking at the big picture. This is particularly true when archaeologists deal with complex curved structures such as ship hulls, which are extremely difficult to imagine and recreate on paper. In such cases, each step of the construction of the 3D model may clarify doubts long held about the nature of a particular artifact fragment. Often times it raises entirely new questions and sends the archaeologist back to his data, documental evidence, or iconographic sources. Models, physical or virtual, are thus an important part of archaeological research.

An issue facing virtual archaeology is effectively communicating its highly interpretive nature. Archaeological reconstructions are by definition hypothetical, as the true past is unobservable. A particular reconstruction is an archaeologist's best guess as to the way things may have been. Another archaeologist may have a different, and equally valid, reconstruction of the same subject. It is difficult to convey this tenuous nature in a realistic visualization, since the usual process used to create 3D models is not good at visualizing ambiguity. Thus, when a virtual model is presented, it has the potential to be misinterpreted as fact rather than as an interpretation.

Another consideration in the design of a virtual archaeology project is the intended audience. A project may be created with educational purposes in mind, and as such might be geared towards students. A project might also be designed for public consumption, such as in a museum kiosk. Alternatively, a project might be designed for research purposes and contain specialized information for expert analysis.

Virtual archaeology should not be simply regarded as the product of archaeological research, although it is often treated that way. Virtual models are flexible representations that can inspire new interpretations. They can be modified over time, as new information is uncovered. Virtual models can even be used to present alternative interpretations within the same presentation. Perhaps most importantly, the process of creating models can also lead to the identification of discrepancies in the archaeological data and give way to new interpretations and corrections of mistaken assumptions. Models can also help substantiate assumptions when discrepancies are not found.

Virtual models should be linked to specific archaeological problems, rather than being treated as a generic reconstruction. This practice helps to keep the project more manageable for researchers and clear for viewers. An overly general virtual archaeology project has a tendency to run into problems of uncontrolled scope. It can quickly become unmanageable for researchers and difficult for viewers to understand.

Approaches to Virtual Archaeology

Computer-generated video, panoramic visualization, and immersive environments are three common approaches to presenting virtual archaeology that involves 3D computer models. These techniques offer varying degrees of viewer interactivity and immersion.

A virtual camera might be animated to fly through a 3D model of an archaeological site, which is rendered and viewed as a video. A 3D model created for this purpose can have a very high level of detail in geometry and special effects. Videos are useful in the sense that they are easy to distribute. However, videos are not interactive. The user is limited to a predetermined view. They are also not immersive, generally being viewed on a single display such as a television screen.

Panoramic visualization, such as those created with Apple QuickTime Virtual Reality (QTVR), features a fixed camera position with 360-degree panoramic images that surround the viewer. These panoramic images can be rendered from 3D environments and then mapped onto a simple cylinder, sphere, or cube surrounding the virtual camera. This pseudo 3D visualization imitates 3D environments using 2D images. The viewer can interactively aim the camera to look at different parts of the image. *Hot spots* are sometimes embedded to allow the viewer to jump instantly to other fixed camera locations. Although interactive, these panoramas are usually not immersive. They do not have the ability to move the camera smoothly through 3D space. They are also usually viewed on a single display. QTVR provides an internet browser plug-in, which makes it possible to quickly distribute and view these visualizations online [2].

Another way of presenting virtual archaeology is in the form of highly interactive and immersive environments, such as the one used for this project. These environments are often referred to as CAVEs (computer-assisted virtual environments). CAVE environments feature multiple displays that surround the viewer in some fashion to create an immersive experience. These environments are usually interactive using interaction devices that control movement through and perhaps manipulation of fully 3D virtual environments. Compared to video and panoramic visualization, CAVEs require specialized displays and equipment, but offer an experience that is more representative of reality, such as walking through an archaeological site or virtually examining a reconstructed artifact.

CHAPTER II

BACKGROUND

Precedents in Virtual Archaeology

Using computer visualization to reconstruct archaeological sites such as a shipwreck is not a new practice. Evolving technology has allowed these visualizations to improve greatly over the years. In 1994, Melissa Saul (M.S. Visualization Sciences, Texas A&M University) published her thesis focused on the reconstruction of a seventeenth-century Dutch passenger ferry [3]. Saul's research resulted in a pre-rendered flythrough animation exploring the ferry. Current graphics technology allows much more detail in the Pepper Wreck model, higher quality shading and lighting, and the ability to view the ship interactively in an immersive environment.

Additionally, this is not the first visualization addressing the subject of Portuguese naus. Alex Hazlett (Ph.D. Anthropology, Texas A&M University), used 16th century shipbuilding documents and mathematical formulas used by shipwrights to virtually reconstruct a nau timber by timber [4]. His research resulted in an annotated and illustrated construction sequence that shows the placement of every timber in the ship. My research differs because it is based specifically upon the Pepper Wreck, includes realistic shading, and is low-poly, which allows it to be used for interactive real-time viewing. Similar to the fashion in which Hazlett modeled the ship timber by timber, my reconstruction includes a proposed cargo allotment with each piece of cargo placed individually.

Beyond the field of nautical archaeology, we can also consider visualizations created within other subsets of archaeology. John Kantner of Georgia State University developed a 3D reconstruction of Chetro Ketl, a ceremonial kiva found in Chaco Canyon in northwestern New Mexico [5]. The purpose of his reconstruction was to create an educational experience of how the kiva might have once appeared. Creating a complete experience was problematic for him due to incomplete information. He states, “Because of poor preservation and incomplete reporting of the work, no information was available on the kinds of intact vessels found in the structure, the nature of the plaster covering the walls, or the roofing material that may have been used” [5, p. 51]. As a result, Kantner had to rely upon his knowledge and excavations of similar structures rather than specific data gathered at the Chetro Ketl site. He populated the building with pottery, placed pictographs on the wall, and designed a roof for the structure. To make his students aware of the hypothetical nature of the reconstruction, he provided them with documentation describing the manner in which he created the reconstruction. He explains, “Descriptions of the inferential process have actually been very useful for illustrating to students how archaeologists derive conclusions from archaeological data” [5, p. 51].

The Pepper Wreck visualization faced issues similar to Kantner’s kiva reconstruction. Since few remains of the actual ship were preserved, the 3D model of the *Nossa Senhora dos Mártires* is based on a hypothetical reconstruction. In addition, since the shipwreck occurred close to the shore and the majority of cargo had been salvaged, there is little archaeological data describing the placement of cargo. Yet, the visualization will include models of cargo, stored according to the best knowledge of the archaeologists studying this subject. This practice will help create a superior and more complete experience of such a ship. More importantly, its cargo distribution will stand as the best educated guess available for peer discussion.

Finally, there is the issue of how the interpretive nature of the visualization will be communicated to people who view it. Kantner summarizes the problem well: “The exceptional quality of 3D architectural renderings has made it substantially more difficult for viewers of visual reconstructions to determine which features were actually identified by the archaeologists and which represent more tenuous interpretations that the modelers were forced to make as part of the modeling process" [5, p. 47]. A method of conveying what is real and what is simply realistic must be found. Documentation will accompany the Pepper Wreck visualization to make clear that the model is based upon a widely published hypothetical reconstruction.

Examples of other virtual archaeology projects and the specific techniques they employed, such as photogrammetry, tomography, and augmented reality, may be found in conference proceedings [6] and [7].

Immersive Visualization System

The immersive visualization system used for this project was designed and built by Dr. Frederic Parke, at the Texas A&M Visualization Laboratory. He says, "While standard visualization techniques provide 'windows' into virtual environments, immersive visualization provides the sense of being 'within' and experiencing these environments" [8]. The system used for this project is a partial prototype of a 360-degree cylindrical display surface approximated with ten facets. Fig. 2 shows the prototype with five display panels arranged in a half-circle around the viewer that create a 180-degree, semi-immersive environment. Each panel's image is created using a separate projector and computer. All of the computers work together to create a synchronized image across all five display panels.

Content for immersive systems are not always easy to create because workflows are not yet fully developed. According to Parke, "High cost and limited access have inhibited the integration of these systems into routine workflows. Now that the cost barriers are falling, the development of much better software support and workflow integration is needed for widespread adoption" [8, p. 242]. Custom software is being developed using open-source software packages. The custom software creates the need for custom workflows, namely for users creating content with Autodesk Maya or other 3D modeling software packages. This workflow is currently in development in the form of scripts, which export and organize Maya files for the immersive system.

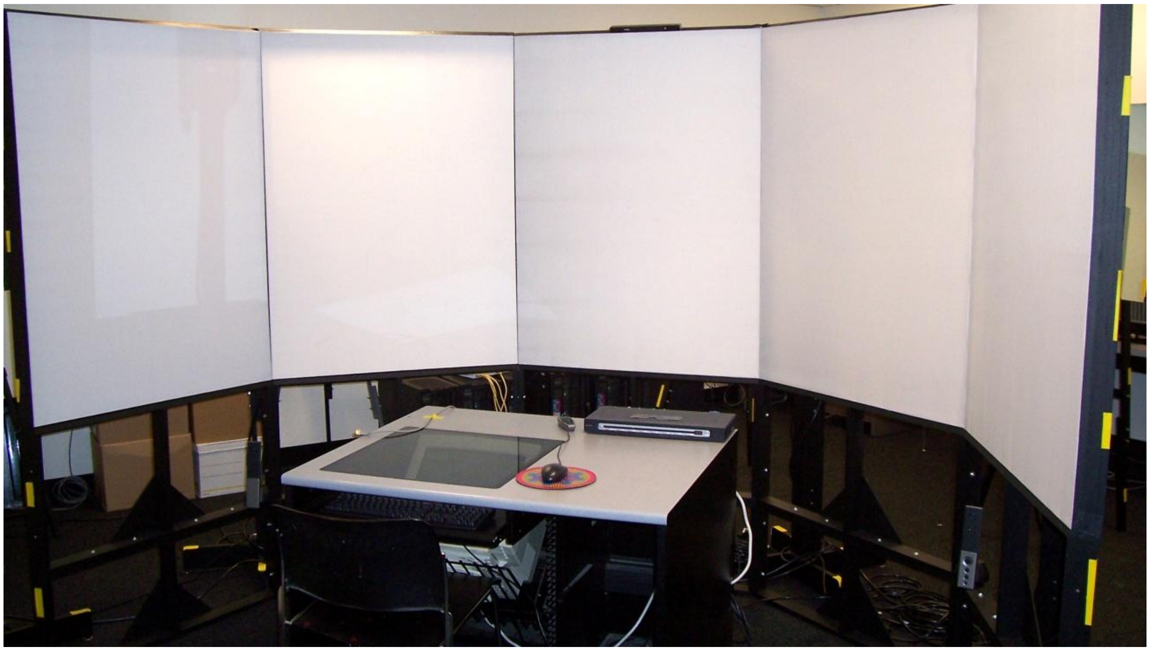

Fig. 2. Immersive system displays

Portuguese Naus and the India Route

In 1498, the celebrated Portuguese explorer Vasco da Gama was the first European to discover and establish a maritime route to India (see Fig. 3). In the 16th and early 17th centuries, Portugal (see Fig. 4) became a dominant power in the lucrative trade along the India Route. Each year, a fleet of ships sailed from Lisbon to India. Sixteen to eighteen months later, the ships returned laden with spices, fine textiles, precious gems, exotic animals, and other trade goods.

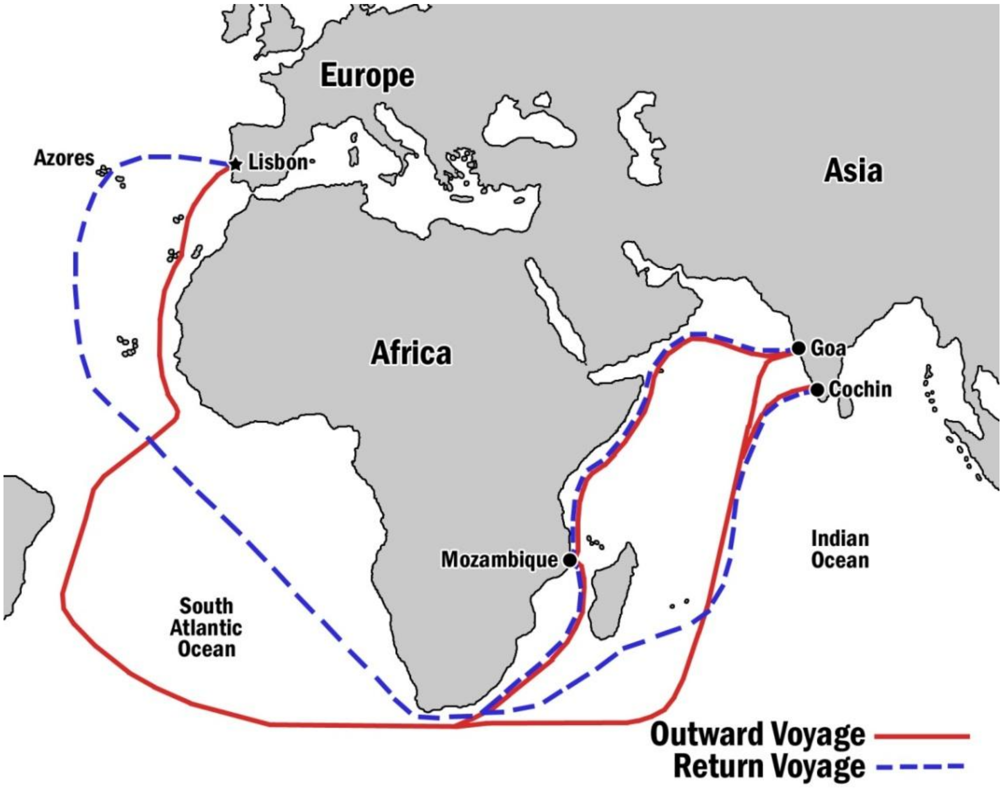

Fig. 3. Maritime route from Portugal to India. Adapted from [9]

Although several types of ships sailed to and from India during Portugal's expansion overseas, the nau is considered the "true workhorse of the Portuguese overseas fleet" [9, p. 5]. The term *nau*, which simply means *vessel*, is also referred to as *carrack* by the English, or *Indiaman* due to its business along the India Route. Portuguese naus were large with three or four decks and three or four masts, and designed for very long journeys. They accommodated several hundred passengers and crew and carried enormous quantities of cargo.

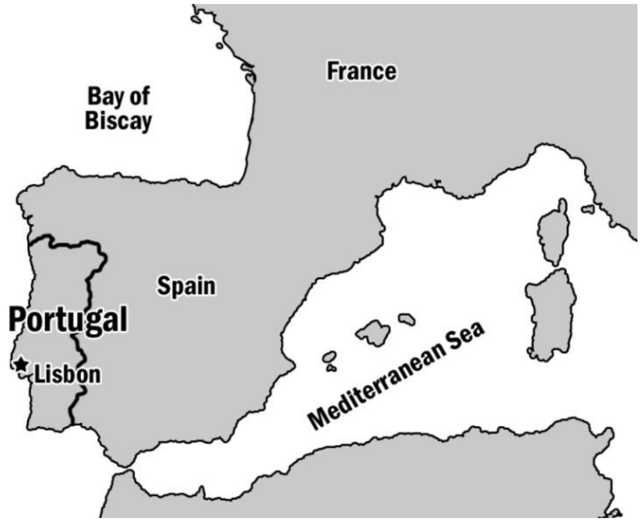

Fig. 4. Map of Portugal and the city of Lisbon

Naus carried a variety of trade goods. However, peppercorn, used to season food, was the most important spice in Portuguese trade. Special boxes were built into the holds of the ships to store the pepper. The rest of the cargo was crammed into any available space on the ship, and sometimes even hanging from ropes outside the hull.

The voyages were dangerous due to the overloading of cargo, attacks by pirates or privateers, severe weather, disease, and starvation. After the late 16th century, financial difficulties, competitors, and war threatened the Portuguese trade network. During this period, changes to the route and later departure dates – to avoid confrontation with enemies – resulted in more frequent shipwrecks due to storms [9, pp. 18-26].

The Pepper Wreck

In March of 1605, after receiving news of a Dutch war fleet in the Far East, a fleet of ten ships set sail from Lisbon to confront them. Half of the ships were to reinforce the fleet defending Portuguese positions. The rest were to take on massive loads of pepper and return to Lisbon [9, pp. 23-24]. The *Nossa Senhora dos Mártires* was one of the ships laden with pepper in Cochin, India. After a six-month voyage returning from India and a three-month stop in the Azorean Archipelago, the *Mártires* was in view of Lisbon on September 14th, 1606. A storm led the captain to seek refuge by heading towards the mouth of the Tagus River and the calmer estuary beyond. Treacherous sandbanks obstructed the entrance to the river, however. On September 15th, the *Mártires* struck a submerged rock and rapidly sank in front of the fortress of São Julião da Barra (see Fig. 5), breaking into many pieces on the rocks. The pepper boxes built into the hold ruptured, and the spilled pepper "formed a black tide that extended for leagues along the coast" [9, p. 3]. It is not surprising that this wreck has come to be known as the *Pepper Wreck.*

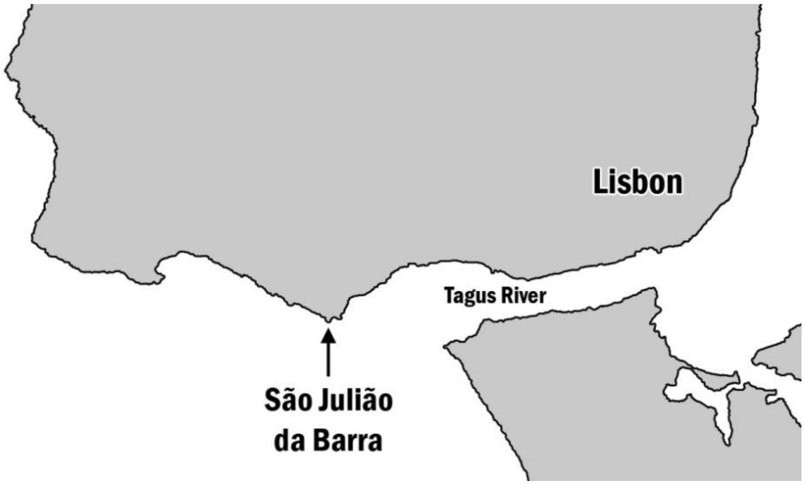

Fig. 5. Map of Lisbon, Tagus River, and the fortress of São Julião da Barra

Dr. Filipe Vieira de Castro, Professor of Nautical Archaeology and the Director of the Ship Reconstruction Laboratory at Texas A&M University, has done extensive study of the Pepper Wreck. The wreck site has been the focus of several excavations conducted between 1996 and 2000 (see Fig. 6). In 1996 and 1997, excavations were conducted on the wreck site under the direction of Dr. Castro and Mr. Francisco Alves, who was then director of the Museu Nacional de Arqueologia (National Museum of Archaeology) in Lisbon. During the summers of 1999 and 2000, the Centro Nacional de Arqueologia Náutica e Subaquática (CNANS), the national agency for nautical archaeology in Portugal, and Texas A&M University's Institute of Nautical Archaeology, sponsored two excavation seasons on this site.

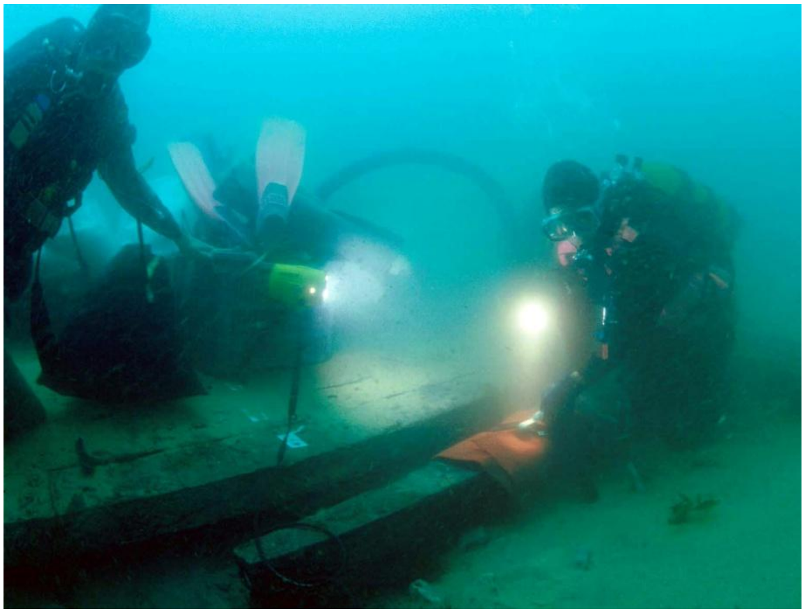

Fig. 6. Pepper Wreck hull excavation in 1999. Photo: Guilherme Garcia, Texas A&M Ship Reconstsruction Laboratory [10]

All that remains of the *Nossa Senhora dos Mártires* today is a small portion of the hull, along with many artifacts including guns, anchors, astrolabes, porcelain dishes, pewter plates, ceramic pots, and a collection of valuable trinkets. Many of these items are now on display at the National Museum of Archaeology in Lisbon.

Arguably, the most important artifact is what remains of the hull. Dr. Castro has utilized data gathered at the wreck site in conjunction with the knowledge of various ship iconography and shipbuilding texts to develop a hypothetical reconstruction of the ship. According to Castro, “Construction marks carved on the surfaces of the floor timbers allowed us to not only understand the method used by the shipwright to conceive the hull shape, but to reconstruct some of the hull dimensions with a good degree of certainty” [9, p. 5].

According to Dr. Castro, the Pepper Wreck is the only Portuguese nau of the period excavated by archaeologists. He states that very little is known about India naus, since few records exist from that period. Although there is existing artwork that contains depictions of ships like the India nau throughout the 16th century, generally an artist’s interpretation tends to be inaccurate. On the other hand, shipbuilding texts such as Manuel Fernandes' Livro *de Traças de Carpintaria* (1616) provide a starting point but are by no means complete diagrams. Castro sums up the problem well:

> Because they were built in a pre-industrial era there were no drawn plans. Because they were an inconspicuous part of the landscape, nobody bothered to paint them in detail. Finally, in spite of being complex machines, their construction relied mostly on practical knowledge and tradition and they belonged to a sphere of knowledge that did not attract many scholars [11].

Currently, there is little public dissemination of information on the India nau. A virtual computer reconstruction is a good way to document what is known about this type of ship and make it available to the public. A visualization of the Pepper Wreck ship could serve as an installation in a museum to provide both archaeologists and the public with a better understanding of the ship’s physical features, size, layout, and crew and cargo capacities.

CHAPTER III

REAL-TIME COMPUTER GRAPHICS PRIMER

Introduction

This chapter is intended to serve as a basic introduction to the real-time computer graphics (CG) techniques that I used during the implementation of this project. My hope is that this primer will aid archaeologists who wish to further their understanding of basic computer graphics as used in real-time virtual archaeology projects. This is not a complete introduction, however, as I only cover techniques that are directly related to this project. For a more comprehensive and technical explanation of computer graphics, please refer to [12].

Modeling

Polygon models are well suited to representing objects that are naturally planar, such as cubes. Individual polygons are made up of at least three edges (lines) and vertices (points) forming a single planar face (surface). A polygon mesh is a set of connected planar surfaces consisting of edges, vertices, and faces (see Fig. 7). Polygons can also be used represent curved surfaces; however, the representation is only approximate. Close viewing will still reveal linear edges in the silhouette of the model, sometimes referred to as *geometric aliasing* [12, p. 472]. Reducing the amount of geometric aliasing is a tradeoff between quality of the model and display performance (see Fig. 8). Using more polygons will result in a more accurate representation of a curved surface. However, more polygons results in more computation time to process and display the model. Using polygons to model complex curved surfaces can be tedious because many vertices have to be correctly positioned.

Parametric models are well suited to representing objects that are naturally curved, such as cylinders. *NURBS* (Non Uniform Rational Bézier Spline) modeling is a common form of parametric surface modeling in 3D graphics software. A NURBS surface is a set of curved surface patches and corresponding control vertices (see Fig. 7). Adding more control points allows better approximation to a given curve. This technique allows the modeler to create much more precise curved surfaces compared to polygonal modeling. These surfaces are usually intuitive and simple to edit because the surface can be specified with a relatively small set of control points. This is a significant advantage over polygonal modeling when working with complicated surface curvature. However, NURBS surfaces are not generally used for real-time applications and must be converted into polygon form for that purpose.

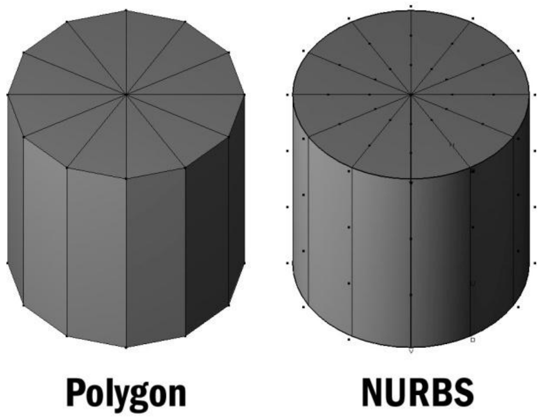

Fig. 7. A polygon model compared to a similar NURBS model

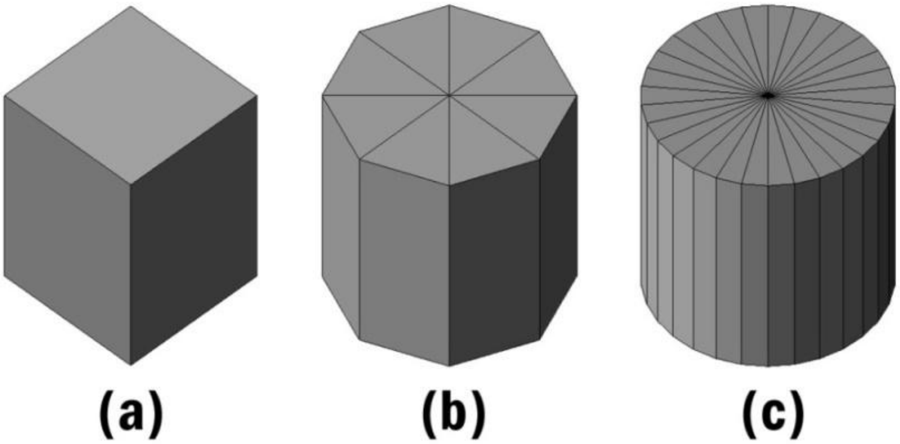

Fig. 8. Levels of geometric aliasing in representing curved surfaces. Representing the tradeoff between quality of the model and display performance: (a) low poly, high aliasing, (b) medium poly, medium aliasing, (c) high poly, low aliasing

The 3D models for this project were created using Autodesk Maya 7.0. Maya is an industry standard software package for 3D modeling, animation, effects, and rendering. Maya capabilities used by this project include polygon and NURBS surface modeling, Boolean operations on surfaces, model deformers, UV mapping, cloth simulation.

Boolean operations create geometry based on the union, difference, or intersection of two objects. *Model deformers* have a variety of functions such as bending or twisting which allow the user to modify geometry based on a few parameters. Another type of model deformer, such as a *lattice deformer*, allows geometry with many vertices to be modified more easily with a smaller set of control points. *UV mapping* is the process of preparing a 3D model so that a 2D image can be applied as a texture. *Cloth simulation* is the process of simulating cloth-like 3D geometry using dynamic forces such as wind and gravity.

Texture Mapping

Modeling small details on a surface is impractical for real-time applications. Instead, surface detail can be simulated using texture maps. *Texture mapping* is the process of applying one or more 2D images to a 3D surface to give the appearance of surface detail not present in the geometry. The primary type of texture map used for this project is called a *diffuse map* or *color map*, which affects the color of a surface. In some display systems, 3D surfaces can further be enhanced by additional types of maps, such as specular maps, bump maps, and normal maps. *Specular mapping* defines the amount of shininess and specular color of a surface. *Bump mapping* and *Normal mapping* are two methods of adding detail to shading without using more polygons. Although these maps add greatly to the realism of a model, the immersive display engine used for this project does not currently support them.

There are three primary methods of creating color maps. One method is procedural textures, which are computer generated using various algorithms such as fractal noise. The second method is to use photographs of surfaces similar to the one being modeled. The third method is to use images created using artistic techniques (either digitally or traditionally to be digitized afterwards). For this project a combination of these techniques are used. Blending textures allow, for instance, a procedurally generated or photographic base to be enhanced by layers of digital hand-painted details.

The textures for the 3D models were created using Adobe Photoshop, a powerful image creation and editing software. Photoshop has an extensive list of features and tools which assist in the process of photo editing and digital painting. The creation of textures for this project relied on the ability of Photoshop to create layered image files so that each component of a texture can be easily modified or copied for use in other textures.

A *UV map* describes how the texture map conforms to the 3D surface as if the 3D surface was unwrapped and laid flat. This UV data is stored with the geometry data. Every vertex in a polygonal surface is associated with one 2D UV coordinate per connected face. A vertex shared between faces can have different UV coordinates for each of its faces. This means adjacent faces can be cut apart and positioned on different areas of the texture map. There are several automatic tools in most 3D modeling applications that assist in the process of UV mapping, such as projections, which create a UV map by generating a "snapshot" of UVs as projected onto a plane, cylinder, or sphere. Even with these tools, manual editing of UV maps of complex surfaces is inevitable and required to fine-tune the results.

The goals when UV mapping are to minimize undesired seams, overlaps, and texture stretching. If the model is symmetric, overlaps can be used advantageously to "mirror" the texture over edges. A texture can be designed to be seamless, so that the edges match if wrapped around. When a model's UVs are scaled beyond the boundary of the unit square, the texture will tile. This is a useful technique, which allows us to repeat a texture multiple times on a surface, such as tiling a small section of hull planking to cover the entire surface. The amount of tiling is easily adjusted by scaling the UVs, and does not require modification of the texture itself.

Shading

Shading is the process of calculating the color of a 3D model's surface based on surface properties and light sources. To calculate surface shading, we need to know the direction from which the light is coming and the angle at which the light hits the surface. Although this calculation is handled by the display software, it is important for the modeler to understand because *surface normals*, unit vectors perpendicular to the surface at the vertices of a polygon, can be adjusted to create different shading effects. These normals are stored as a part of the polygonal mesh data along with vertices, edges, and faces.

Constant, or *flat* shading is a simple rendering technique in which each polygon face is given a single color value based upon a calculation using its face normal (see Fig. 9). Although this method is computationally cheap, it does not work well if the polygons are approximating a curved surface because its appearance is very faceted [12, pp. 734-735].

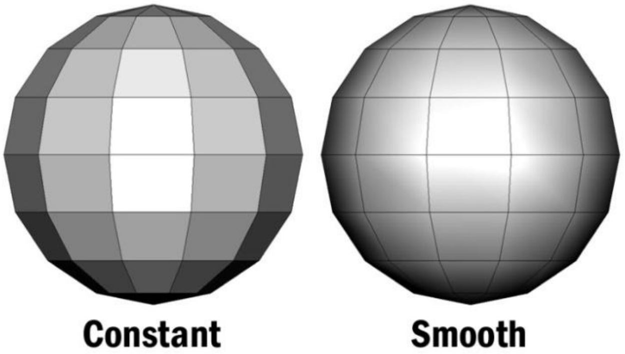

Fig. 9. Constant shading compared to smooth shading. Constant shading creates a faceted look, while Smooth shading creates a smooth appearance

Smooth shading is a rendering technique developed to simulate smooth surface shading of polygonal objects (see Fig. 9). It is also known as *Gouraud* shading after its pioneer, Henri Gouraud. The shading is calculated based on normal vectors at each vertex of a polygonal surface. The shading across each polygon face is interpolated to determine the value at each pixel [12, pp. 735-737]. This is an important technique because it allows polygonal surfaces to appear smooth, and therefore more closely represent curved surfaces (see Fig. 10). To create a smooth appearance, the vertex normals can be formed by averaging adjuacent face normals (see Fig. 10). It is also important to note that vertex normals do not need to be smoothed over the entire model, but can be arranged into *smoothing groups* in which only some vertex normals are averaged. In Fig. 10, only the vertex normals corresponding to the sides of the cylinder are smoothed together. The normals for the top and bottom faces of the cylinder are still set perpendicular to those faces, which creates a hard edge between the sides and the caps. If all the vertex normals are set perpendicularly to their corresponding faces, the surface will look very faceted as in the constant shading technique.

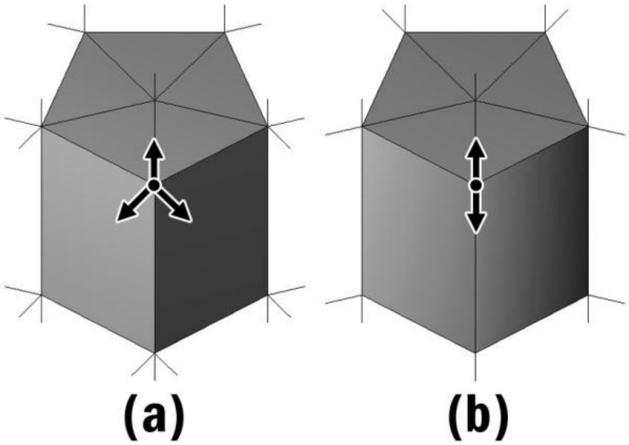

Fig. 10. Arranging vertex normals into smoothing groups. (a) All vertex normals set to face create a faceted look. (b) Vertex normals selectively smoothed across the sides of the cylinder but set to face on the top and bottom

Rendering

Rendering is the process in which 2D images are generated from 3D computer models by means of computer software. The two primary methods are rendering to still images or image sequences (which are converted to video) and rendering in real-time. Both of these methods are designed to produce and display images at a sufficiently high frame rate to create a smooth visual transition for the viewer. *Frame rate* is the measurement of the frequency at which an imaging device produces consecutive images. This takes advantage of human persistence of vision, which allows us to perceive motion instead of individual frames when they are displayed in quick succession.

Pre-rendered computer animations are typically viewed as video at 30 frames per second (fps) or film, projected at 24 fps. Real-time systems, such as the one used for this project, have variable frame rates depending on the capabilities of the system and complexity of the image being produced.

Real cameras take advantage of *motion blur*, the streaking quality created when an object is moving within a single frame. Motion blur helps to create smoother visual transitions between frames. Motion blur can be calculated for pre-rendered video at the cost of additional render time. Real-time rendered images do not typically have motion blur unless the engine is built to simulate this feature. This means each real-time frame is a perfectly crisp image of the 3D scene, no matter how quickly the camera or visible objects are moving. This is why a real-time system with no motion blur and a frame rate of 24 fps may seem staggered, while film at the same frame rate appears continuous. A real-time system that does not have motion blur capabilities requires higher frame rates to reduce this problem. A frame rate of at least 60 fps is ideal, but lower frame rates are often acceptable.

Pre-rendered images and animations support a high level of detail in geometry, textures, and effects such as shadows, motion blur, and depth of field. However, the viewer is constrained to a pre-determined experience chosen by the animator, such as a camera flythrough of a 3D model. Pre-rendered animations can also require long render times, even hours to render a single frame.

In contrast, real-time rendering necessitates limited detail in the geometry and textures, because the model must be simplified to maintain real-time speed. Expensive effects such as reflections, translucency, and shadows usually have to be sacrificed. As computers become more powerful, even these effects are becoming possible in real-time. Real-time systems allow viewers to explore a model spontaneously, since the engine can compute images based upon viewer interaction.

CHAPTER IV

METHODOLOGY

Initial Approach

Moving from 2D data to 3D models is an involved process, which can be applied to many virtual archaeology projects (see Fig. 11). The first step is to collect the data relevant to the project. Data can have multiple forms depending on the project. The data for the Pepper Wreck visualization comes from a variety of sources including archaeological drawings, measurements, photographs, and iconographical artworks from the period. My most important source of data, however, was through frequent tête-à-têtes with Dr. Castro. I have pages full of sketches and notes generated from these conversations.

The second step is to organize the data. Data relevant to the creation of virtual models should be selected and digitized (digitally photographed or scanned into a computer) if not already in digital form. Digitizing all the data allows information to be stored in a central location, easily backed up, and easily shared amongst collaborators. The data should be carefully categorized by source or subject in a hierarchical fashion so it can be quickly referenced while working on the model. For example, I created a directory labeled Cargo that contains subdirectories such as Jars and Barrels. Further subdirectories may distinguish data based on differing sources such as archaeological data and iconographical data. This organizational structure should also extend to all other computer files related to the project, such as the geometry and texture files. Data for this project consists of over ten thousand files, so careful organization and regular backup is crucial.

The third step is to utilize the collected data to create initial virtual models. This includes creating both object geometries and surface textures. While a perfect model would include every detail present in the real world environment, virtual models are limited by the computing power available to display them, and the time available to create them. For this reason, the decision of what to model and how detailed to make the model is an important one. For example, rather than modeling each plank of the hull, it was modeled as a single surface and a texture was applied to create the appearance of many planks.

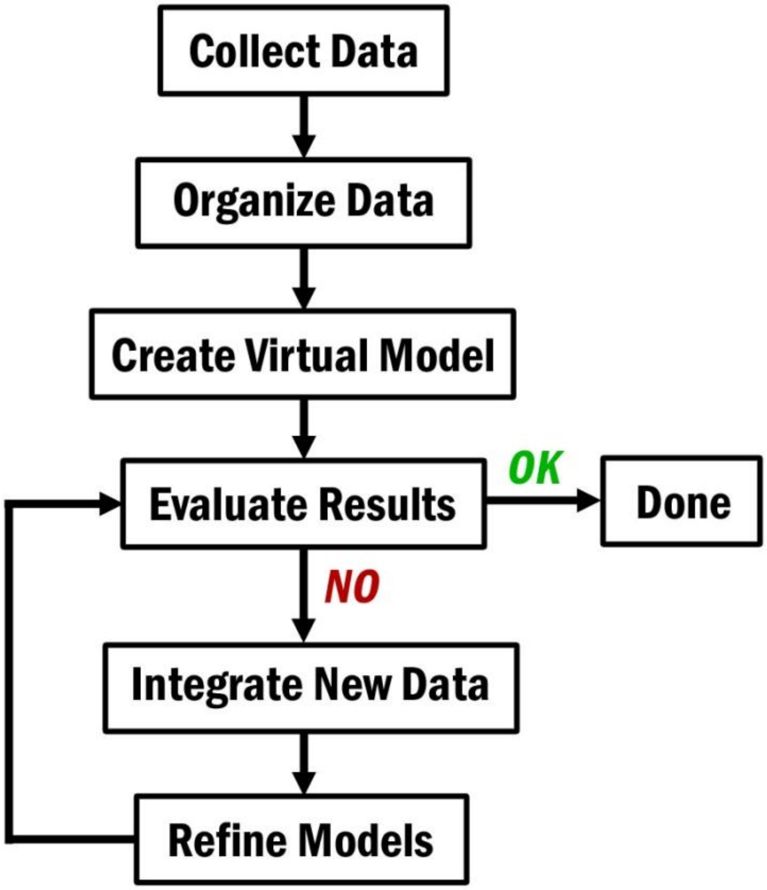

Fig. 11. General methodology for virtual archaeology modeling

Iterative Refinement

Creation of the final model is an iterative process (see Fig. 11). First, the initial model is evaluated. Second, any new data that is found or created should be integrated into the existing pool of data. Third, the model is refined by utilizing the new information. Then the model is re-evaluated and if it is now satisfactory, the model is done. If not, the model development loop iterates again.

Evaluation criteria vary depending on the project. For this project, one criterion was ensuring that the model would not overburden the real-time graphics display system by having too many polygons. Another criterion is ensuring accuracy to the archaeological data. Dr. Castro and I met frequently, sometimes as often as three times a week, to discuss the models. Dr. Castro would determine what changes or additions were required to make the models more accurate. Finally, with Dr. Castro's approval, models were considered finished.

The evaluation process would often require that new data be located or created. This new data would be integrated into the existing pool of data by digitizing and storing it in the appropriate location in the file hierarchy. For example, the original barrel model geometry was created using a simple reference drawing and measurements, and an initial surface texture was created featuring wooden staves and iron hoops. After evaluating this model, new references were discovered which suggested that the hoops would be wooden and of a different design. This new information was digitized and stored in the Barrel subdirectory of Cargo references.

In the refinement stage, new information is used to correct the model. For example, the surface texture of the barrel was modified to feature wooden hoops instead of iron ones. The geometry was modified because the wooden hoops were of different dimensions.

CHAPTER V

IMPLEMENTATION

Preparation

To work effectively and efficiently, it was important to consider several issues that affect the modeling and texturing processes. One of the most important issues was determining the level of detail present in the model. The archaeological research objective of this project was to understand the division of space inside the ship and the allocation of cargo. Models were created with this stipulation in mind.

Since the goal was not to create a complete structural model, each frame and each plank does not need to be modeled, only suggested through texture or a simplified model. The models were designed to have a low number of polygons to run more efficiently on the interactive system. Keeping the polygon count as low as possible while striving for the desired shape and realism is a delicate balance we attempted to strike. Most objects do not need to be detailed in their form, since much detail can be added through texture maps.

One other basic issue to consider was geometric scale. All of the 3D models were designed to conform to the same scale. This way, when a piece of cargo was created in a separate file, it did not require any scale adjustment when imported into the primary scene file and placed on the ship.

Many objects on the ship are identical copies, each placed in different locations. This can be used to our advantage to reduce the load on the interactive system. The system supports object instancing, which essentially allows one object file to contain all the geometry data, while any number of other files point to the first file for the geometry data and only contain unique information for their translation, rotation, and scale attributes. It is important to determine ahead of time which objects would be good candidates for instancing, for objects must be created as instances within the modeling software as well.

The interactive system's performance will be better when a minimal number of texture maps are needed. To achieve this, texture images must be spatially optimized (see Fig. 12), taking advantage of the full area of a texture map. It also means reusing the same textures on more than one model, if possible. With careful layout of object UVs, separate objects can share the same texture map by using separate areas of the map. This is helpful because objects that share the same texture can be combined when completed, reducing the total number of objects and textures required for import onto the interactive system. In addition, the pixel dimensions of the texture map should be considered. Textures that are not prominently seen or are mapped onto small objects in the scene do not need to be large. For example, a 128 × 128 or 256 × 256 map should be sufficient for a small item. A larger texture might be desirable for something seen much larger and closer to the camera, such as 512 × 512.

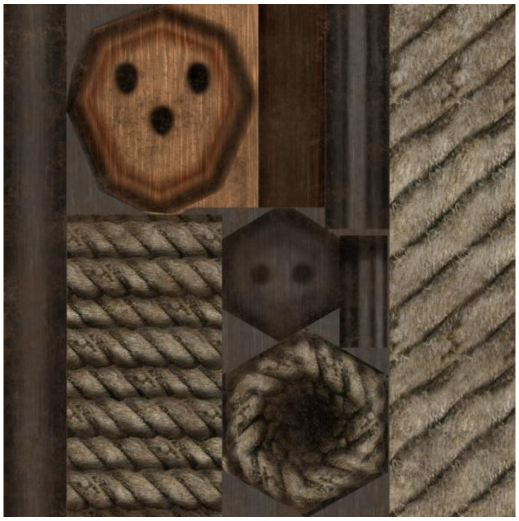

Fig. 12. Example of a spatially optimized texture map

Objects that are adjacent to each other in the 3D scene can lead to visible seams, gaps, or visible intersections. For example, the decks must completely penetrate the hull to avoid gaps and seams. However, the penetration means that the deck would appear to stick out from the other side of the hull. To hide these irregularities, I use a technique I call *edge tucking* (see Fig. 13). In the case of a ship hull, instead of one, there are two separate surfaces – the outer surface, and the inner surface. These separate surfaces are necessary to simulate the thickness of the hull. They also provide a convenient unseen space in which the edges of other geometry, such as the deck, can be hidden.

This technique is also useful for placing objects, such as barrels, in the scene. The ship's decks are slightly curved, which means it is difficult to set a barrel with a flat base squarely on the deck. A corner might poke through to be visible from underneath, or it might appear to be floating above the deck. With two layers of deck, we can both simulate deck thickness and solve these problems. The barrel can slightly penetrate the upper layer without being visible from below.

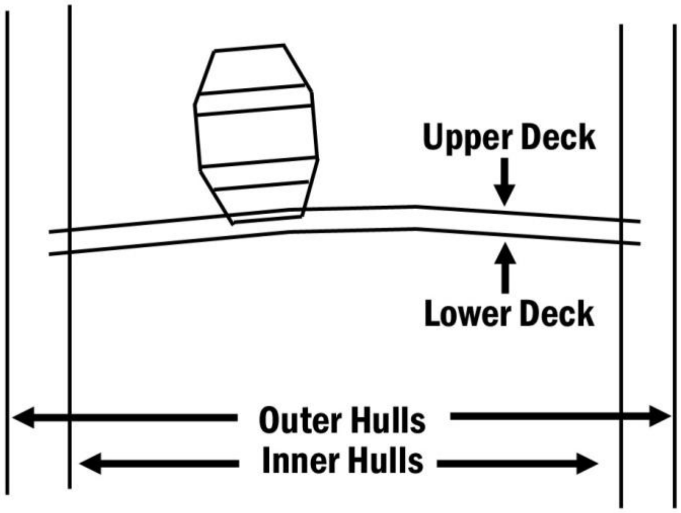

Fig. 13. Illustration of edge tucking

Tucking allows us to reduce the total number of polygons. Without tucking, I would only be able to hide the seams between two objects by exactly matching the vertices of each object to every other object it is in contact with. For example, the vertices of the deck would have to be matched to vertices on the hull. I would have to add unnecessary geometry by running edges down the entire surface just to match a deck to the hull. It is much more efficient in terms of polygon count and time invested to simply let the edges of the wall be hidden in between the two hull surfaces.

Ship Hull

Dr. Castro provided several drawings from various views representing his hypothetical reconstruction of the *Mártires* hull. These drawings include important details about hull dimensions and shape, and have been the basis for the virtual reconstruction. Fig. 14 shows a reconstruction drawing of the ship from the side view.

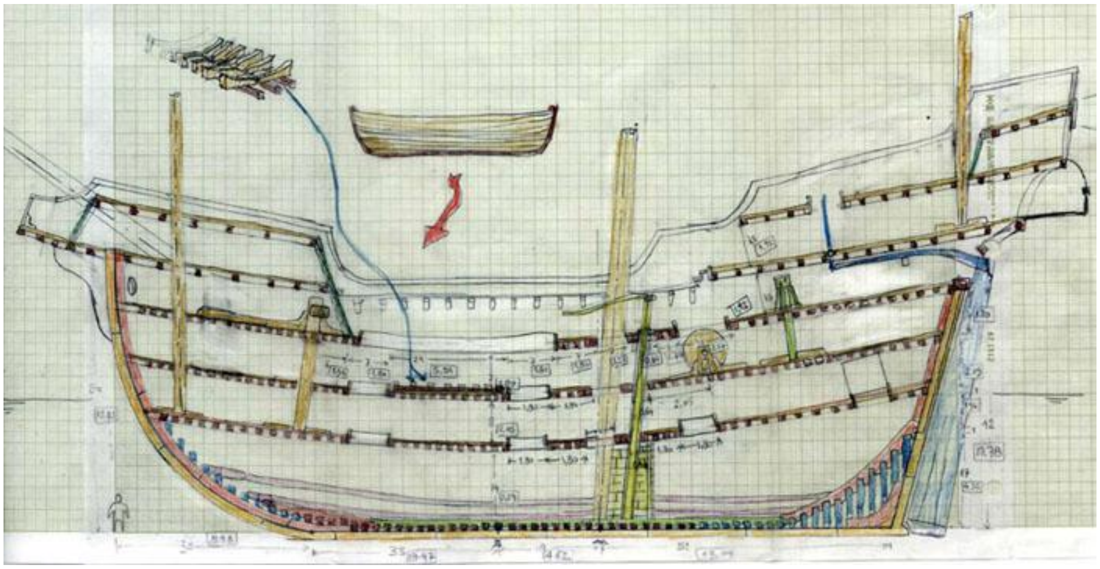

Fig. 14. Side view reconstruction drawing

It is a common practice to use orthographic image planes to aid in the creation of complex 3D models. Three or fewer image planes are usually sufficient. However, when constructing an object with complex curvature, several image planes from the same orthographic viewport can used in the construction. To create the ship's hull, six front view image planes were used. These were matched to the side view plane. (see Fig. 15). Using multiple image planes allowed a greater degree of accuracy in representing the curvature of the hull down the entire length of the ship.

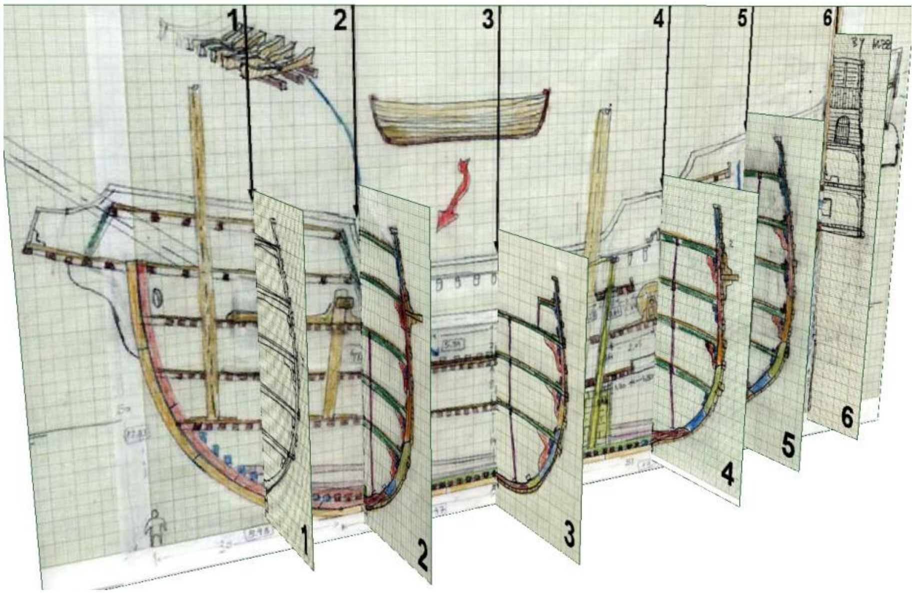

Fig. 15. Side and multiple front image planes combined

Although the hull surface must ultimately be polygonal geometry for the immersive system, the curvature of the hull in polygonal form is extremely hard to edit without introducing obvious flaws in the surface. Editing NURBS control points to adjust the overall surface shape is intuitive and easier than trying to modify individual vertices in a polygonal surface. Thus, it made sense to use NURBS surfaces as a springboard to create the polygonal ship hull.

First, multiple NURBS curves were created based on the front view image planes (see Fig. 16). Each curve has an identical number of control points to create a consistently lined NURBS surface, which means the curves extend beyond where the hull would normally end. This excess was trimmed in the polygonal stage. The placement of curve control points mimicked the curvature of the hull planking, which curves upwards towards the bow and stern. This aided the texturing process later, making it easier for the texture to curve with the geometry.

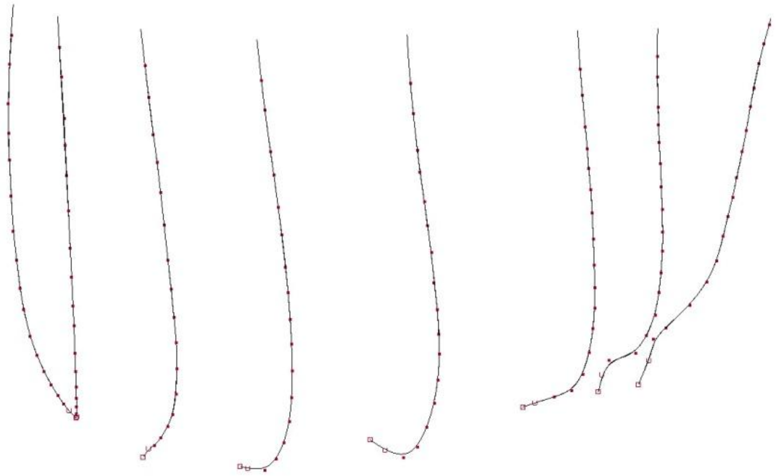

Fig. 16. Hull curves and control vertices

Next, these curves were lofted together. *Lofting* is the process of creating the NURBS surface over the curves, and can be likened to stretching a piece of fabric over curved beams (see Fig. 17). Maya has the capability of maintaining the correspondence between the original curves and the new surface. This means that moving the control points on the curves will automatically be updated in the NURBS surface. Some adjustment of these control points was needed, as the drawings used to create the curves did uncover some discrepancies in the curvature of the hull. This is only natural since they were hand-drawn, and the 3D image planes were placed by eye.

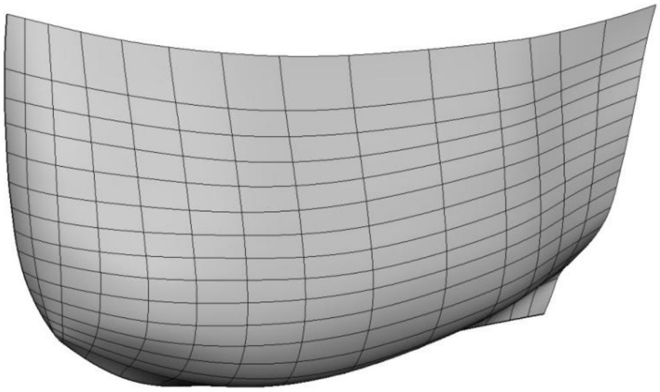

Fig. 17. Rough NURBS hull

After editing of the NURBS surface was completed, based upon Dr. Castro’s feedback, it was converted into polygon form. After this point, any changes in the curvature would be difficult, so finalizing the surface was important. Trimming edges were added along the side of the hull to match the silhouette of the ship, and the excess geometry was discarded. Last, the triangular section on the bow and the stern were created and added. All of this resulted in half of the completed hull. Mirroring the geometry created the complete outer hull (see Fig. 18).

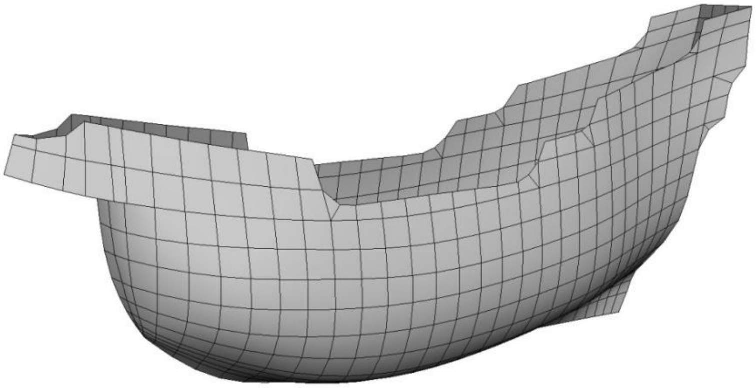

Fig. 18. Polygon hull

The hull was fairly simple to texture by ensuring during modeling phase that the lines of the geometry curved in the direction of the planking. After creating a basic UV map through planar projection, it was merely a matter of straightening out the curved lines as shown in Fig. 19. Finally, applying a tileable plank texture (see Fig. 20) caused the planks to curve in the direction of the geometry. Since the texture can tile seamlessly, the UVs can be scaled to adjust the frequency of tiling.

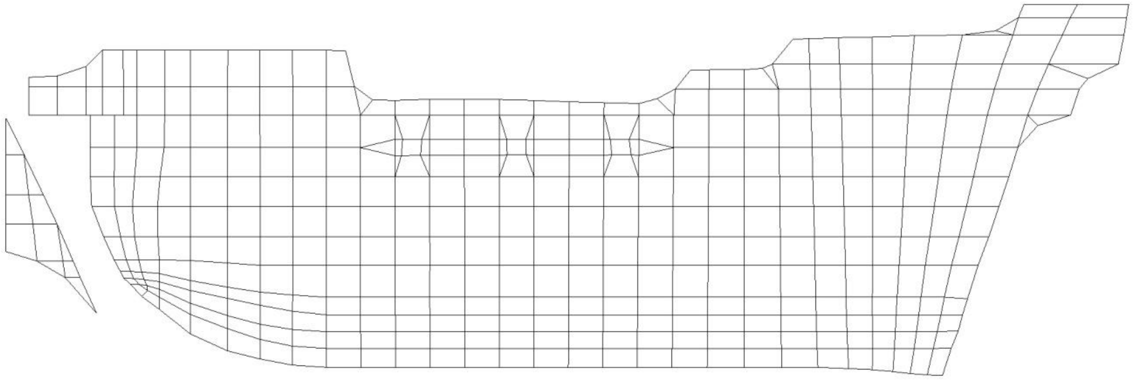

Fig. 19. Hull UV map

Once the hull was finalized, other details could be added. To create thickness to the hull, a copy of the hull was modified to fit inside the outer with a thickness of 25 cm. A railing was constructed all the way around the top edge, covering the opening between the inner and outer layers of hull. Gun ports were cut into the side of the hull. A doorway leading to the stern balcony was cut. Wales (horizontal reinforcement timbers running along the side of the ship) were extruded from the outer side of the hull. Frames (vertical timbers forming a skeleton along the inside of the ship) were extruded from the inner side of the hull.

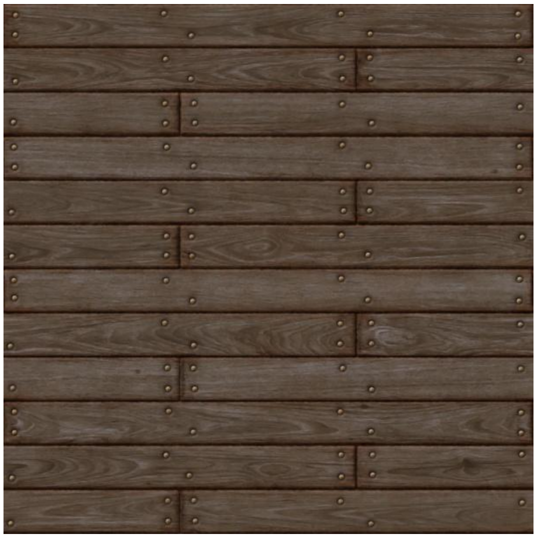

Fig. 20. Tileable hull texture map

We decided to reduce the number of frames in the model to approximately half of the original ones. The distance between two consecutive frames recorded by the archaeologists was around 47 cm. We opted for 1 m to simplify the number of polygons in the model. For this same reason, we opted to make the frames in one single “rib” instead of the archaeologically recorded frames, which would have been assembled from multiple pieces (floor timbers and futtocks).

Ship Decks

The decks were modeled from flat polygonal planes as seen in Fig. 21. Starting with the flat plane, the surface was UV mapped using *planar projection*. Similar to the hull, the UVs can be scaled to adjust the tiling frequency of the applied seamless texture. From the side view, the vertices of the plane were translated to match the reconstruction drawing provided by Dr. Castro (Fig. 14). A *bend deformer*, which bends geometry based upon variables such as curvature, was used to give curvature to the decks so that they bowed upwards slightly in the center as seen from the front view. Next, a *boolean operation* with the hull geometry was used to cut away the excess and fit the deck with the hull. The geometry was cleaned up by removing unneeded edges and leftover vertices. Additionally, holes were cut into the surface wherever hatches were placed.

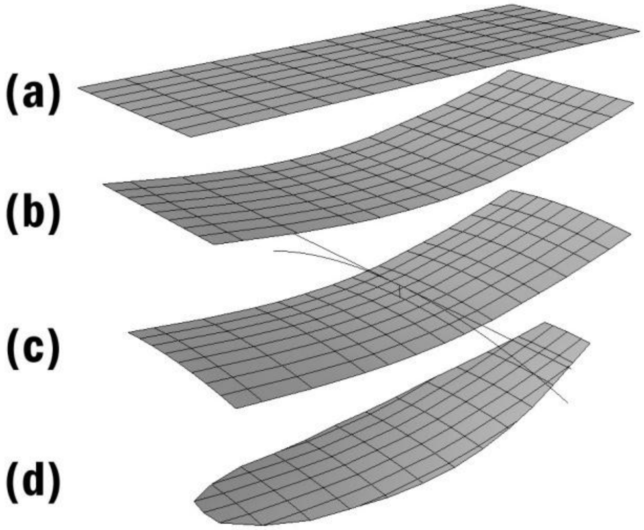

Fig. 21. Basic deck modeling process

To simulate deck thickness, each deck was extruded to a thickness of 15 cm. Again, in order to simplify the model, neither deck beams nor carlings were added to the pavements. In actuality, the pavements (planking) would have been approximately 4 cm thick, with the supporting beams and carlings no more than 18 cm thick. Dr. Castro felt 15 cm was a close enough approximation. The hollow space between upper and lower deck layers were also very useful to prevent other geometry, such as cargo, being visible from the next deck down. The outer edges of the decks were hidden in between the inner and outer hull surfaces so that there were no gaps inside and they were not visible outside the ship.

Masts and Yards

Rigging is the general nautical term for the apparatus of masts, yards, sails, and the lines necessary to support and operate them. The vertical masts and horizontal yards are designed to support the sails, and are labeled in Fig. 22 below.

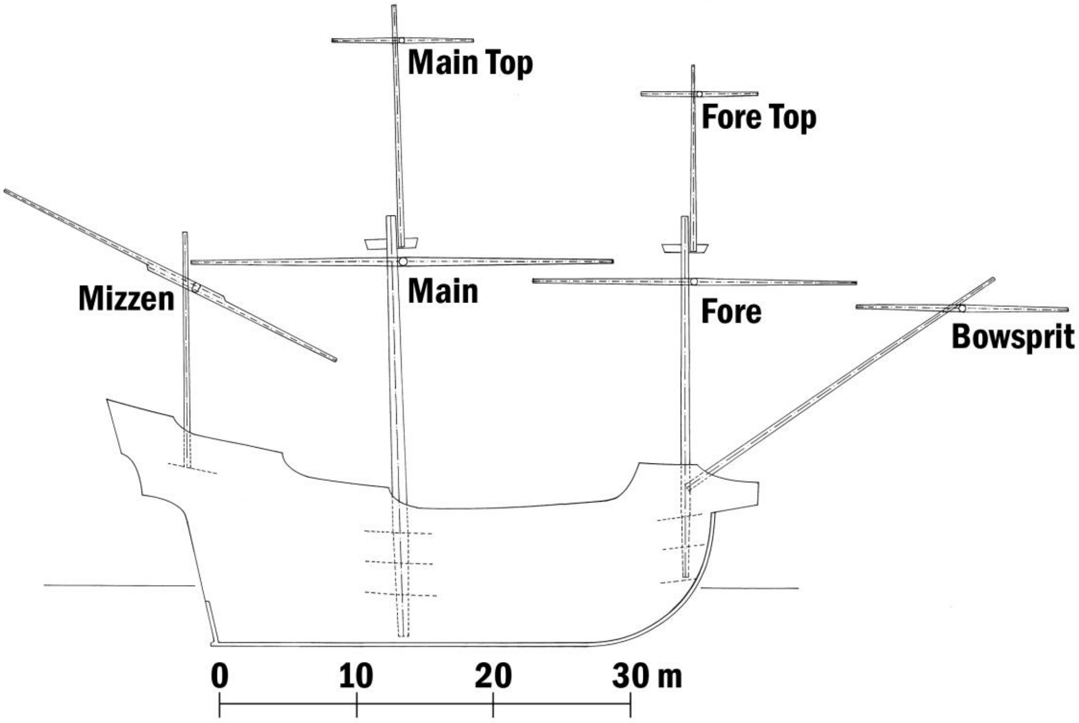

Fig. 22. Masts and yards labeled

The dimensions of the masts and yards for the ship are based on Dr. Castro's calculations as shown in Tables 1 and 2. They were modeled as cylinder primitives modified to match the measurements so that they tapered at the ends. The UV maps were created using cylindrical projections. A tileable wood texture was applied to all of the masts and yards.

In addition to the masts and yards, crow's nests and mast tops were modeled. The tops anchor the shrouds to the tops of the masts, and were modeled simply as four interlocked beams mounted on the top of each mast. Crow's nests were additionally placed at the top of the main and fore masts. Measurements of the crow's nests are found in Table 3.

Table 1. Measurements of masts [13]

Mast	*Length*	*Max Diameter*	*Min Diameter*
Main	31.68 m	116 cm	83 cm
Main Top	18.48 m	44 cm	22 cm
Fore	27.28 m	77 cm	51 cm
Fore Top	14.08 m	39 cm	13 cm
Mizzen	17.60 m	44 cm	29 cm
Bowsprit	28.16 m	51 cm	26 cm

Table 2. Measurements of yards [13]

Yard	*Length*	*Max Diameter*	*Min Diameter*
Main	31.68 m	52 cm	26 cm
Main Top	10.56 m	29 cm	15 cm
Fore	24.64 m	44 cm	22 cm
Fore Top	8.80 m	26 cm	13 cm
Mizzen	28.16 m	29 cm	15 cm
Bowsprit	15.84 m	33 cm	17 cm

Table 3. Measurements of crow's nests [13]

Top	*Height*	*Rail Diameter*	*Base Diameter*
Main	0.77 m	4.10 m	3.59 m
Fore	0.64 m	3.59 m	3.08 m

Standard Rigging

Standard rigging are the lines that do not change position, primarily consisting of shrouds, ratlines, forestays, and backstays. The primary purpose of these components is to hold the masts securely in place. The *shrouds* are thick vertical ropes tied near the top of the mast and attached to the hull via the channels. *Channels* are planks that protrude from the hull, supported by *fenders*. *Ratlines* are thinner horizontal ropes, which are tied between the shrouds to form a ladder for the crew to climb and work aloft. Backstays are similar to shrouds but do not have ratlines tied between them. Mainstays are very thick cables tied from the top of a mast to another mast.

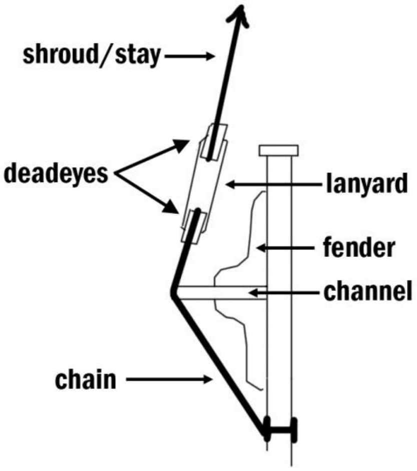

Fig. 23. Illustration of a deadeye

A *deadeye*, illustrated in Fig. 23, is a rounded wooden block with holes through it. They are typically used in pairs with a rope called a *lanyard* run back and forth between them. The top deadeye is tied to the bottom of a shroud or stay. The bottom deadeye is attached to a rope or iron loop, which is then passed through or wrapped around the channel and affixed to the side of the hull, if it is part of a shroud or backstay. If it is part of a mainstay, it is tied around a mast. A pair of deadeyes and a lanyard function as a set of pulleys to adjust tension in the shrouds or stays.

The deadeye models (see Fig. 24) were created based on rough sketches and measurements by Dr. Castro, photographs and drawings of similarly rigged ships, and period iconography. According to Dr. Castro's conjectural sketches, the fore mast has eight shrouds and two backstays attached to it. The top fore mast has six shrouds attached to it, which pass through the mast *top* (platform) and are tied off near the top of the lower shrouds. The main mast has ten shrouds and two backstays attached to it. The top main mast has eight shrouds. The mizzen mast has four backstays. There are four mainstays, tied at the tops of the fore and main mast and extending downwards to be tied off on the next mast towards the bow.

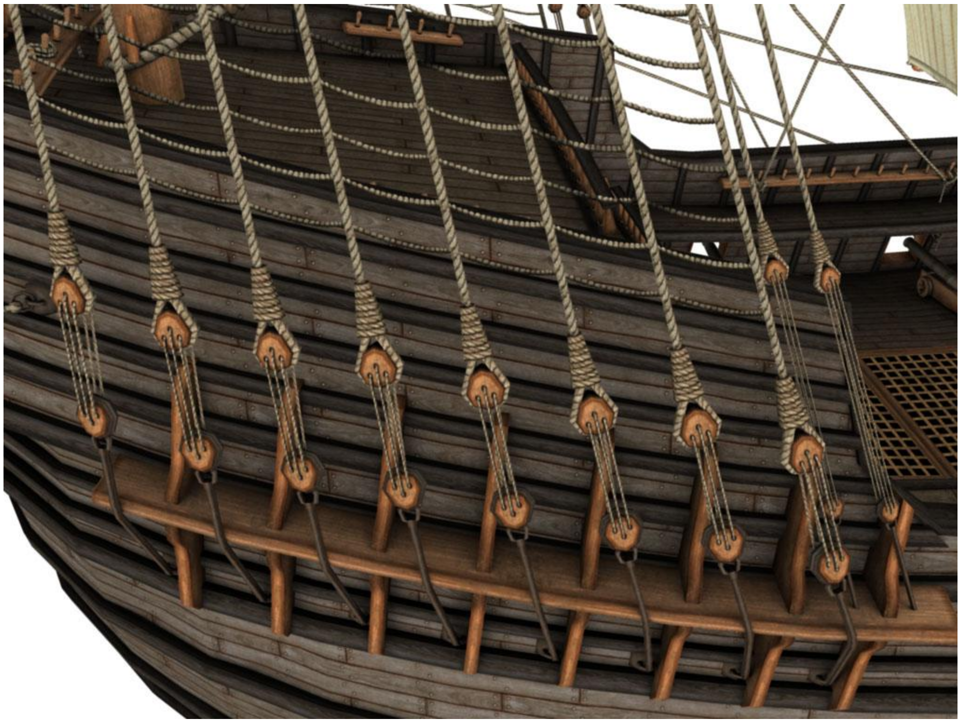

Fig. 24. Deadeyes fastening shrouds and backstays

The shrouds, backstays, and mainstays were created from simple 5-sided cylinders with a rope texture applied, and placed in the configuration described above. To help reduce the amount of rope geometry, the ratlines were textured onto a simple polygon plane with vertical divisions corresponding to the number of shrouds. An alpha channel in the texture map creates transparency between the ropes. Using a *lattice deformer*, I was able to easily modify the position of the top and bottom rows of vertices of the ratlines without needing to adjust every vertex in the ratline geometry. Using the lattice control points, the ratlines were carefully fitted so that the vertical edges were hidden inside the shroud ropes.

Because of the large number of deadeyes required to fasten the shrouds, backstays, and mainstays, I decided that the deadeyes should be instanced geometry. This not only made it easier to adjust the model and UV map, but also improves the performance on the interactive system. Each component of the deadeye was modeled and UV mapped separately. Three variations of the model were made for each type (shroud, backstay, and mainstay). All of the pieces were then combined into the complete deadeye models. A texture was created which all of the deadeyes shared. Instances of shroud deadeyes were placed at the bottom of each shroud rope, and rotated to match the slope of the ropes. In a similar fashion, instances of the backstay deadeyes (smaller blocks with longer lanyards) and mainstay deadeyes (larger blocks with ropes one each end) were placed at the bottom of each backstay and mainstay. The tops of the shrouds were tied around the top of the masts as seen in Fig. 25.

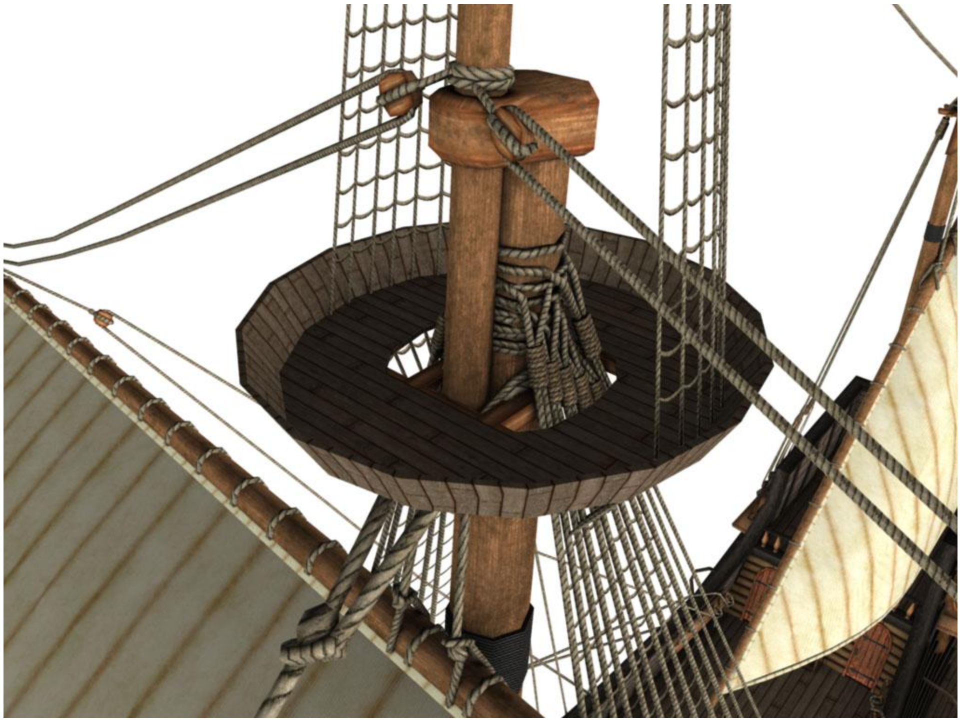

Fig. 25. Shrouds and mainstay tied off at top of the main mast

Sails and Banners

When modeling dynamic objects such as cloth, it is difficult to achieve natural looking results using traditional modeling techniques. Instead, software can simulate real world physics. Using this technique, models can be creating through simulation without the need for tedious geometry editing. Cloth simulation is one type of dynamic simulation that Maya provides and was used to create the shape of the sails and banners on the ship. Maya lets you set up the conditions and constraints for the simulation, and then automatically determines how to animate the objects in the scene. Cloth simulation is performed by the *cloth solver*, which uses information on the cloth geometry, the objects the cloth interacts with, and any forces or constraints applied to the cloth. In Maya, forces such as gravity and air are called *fields*. An air field, defined primarily by its magnitude and direction, was used to create the wind that affected the sails and banners.

First, cloth geometry needs to be created. NURBS curves were created to define the edges of the sails, according to the dimensions provided by Dr. Castro shown in Fig. 26. Maya automates the process of converting these curves into a *garment*, a type of polygonal mesh that can be used in a Maya cloth simulation. These meshes are made up of triangles sized and arranged in a randomized fashion. This randomization helps to prevent artificial stress lines that would appear in the cloth if regularly tessellated. Irregularity in the geometry promotes more natural folding and stretching of the cloth. In the two examples shown in Fig. 27, the irregular cloth garment creates more natural simulation results.

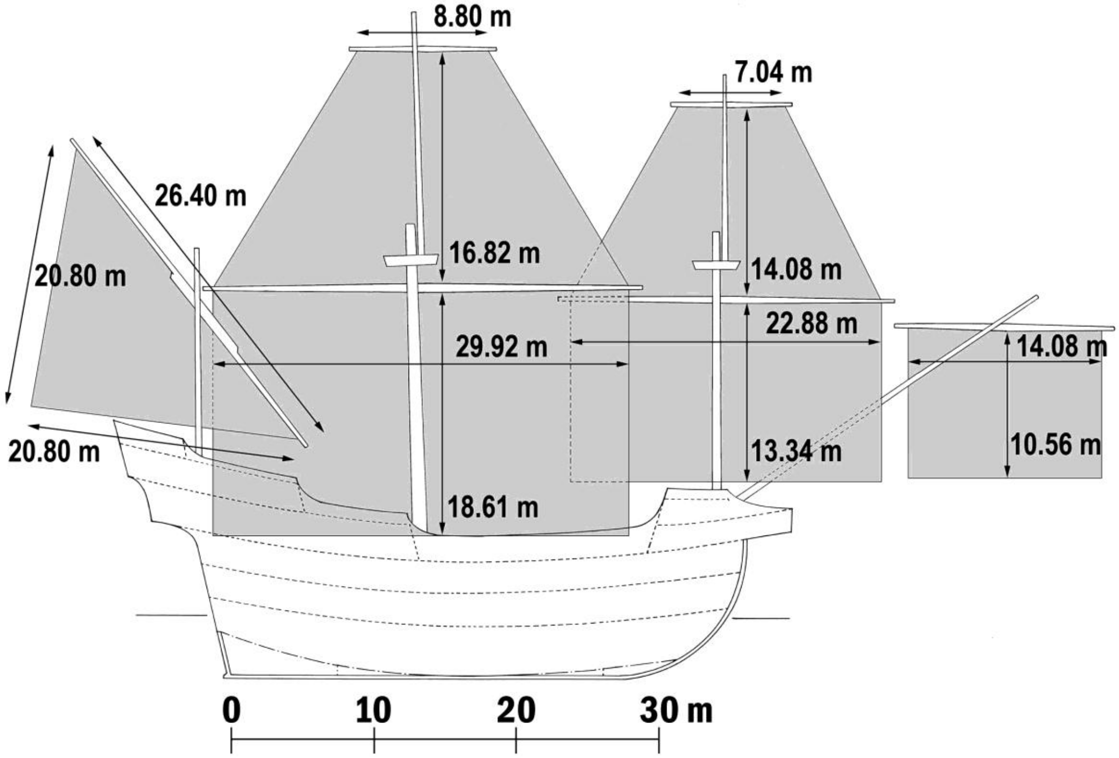

Fig. 26. Measurements of sails

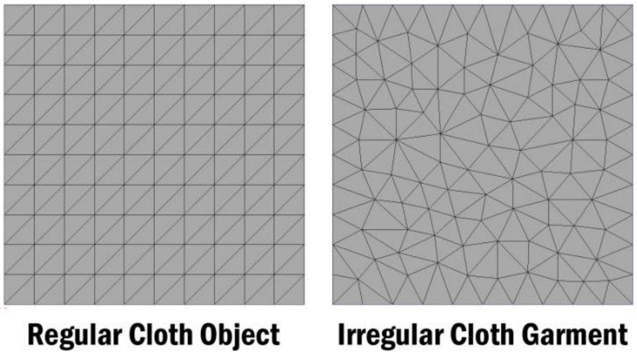

Fig. 27. Regular versus irregular tessellation of cloth geometry

The garments then had to be affixed to the yards. One way of manipulating the cloth during a simulation is to use a *transform constraint*. A transform constraint attaches vertices of the cloth to another object. If the object moves, the points on the cloth that are constrained to it also move. The top row of sail vertices were constrained to *locators,* objects that mark points in 3D space. During simulation, these locators were moved to match the curvature of the yards so that there were no gaps where the sails attach to the yards. The corners of the sails, where ropes will be tied, were also constrained to locators. During simulation, these locators could be moved to adjust the fullness of the sails (see Fig. 28).

The triangular lateen sail (see Fig. 29) was a special case because it is placed underneath the stays on the mizzen mast. To prevent the sail from inflating and going through the stays, a special object called a *collision object* was created. A polygon cube was modified to fit around the backstay geometry and added to the cloth simulation as a collision object. During simulation, the sail is prevented from intersecting the collision object, and instead billows around it.

Cloth simulation would also be an excellent way to create animating sails and banners. However, the immersive display engine does not have support for animation at this time.

When Dr. Castro was satisfied with the position of the sails, the cloth simulation data was frozen, and the sails became static models. Geometry edging was added to give the sails some depth. Additional rope geometry, the running rigging, was then added to create a more complete looking rig.

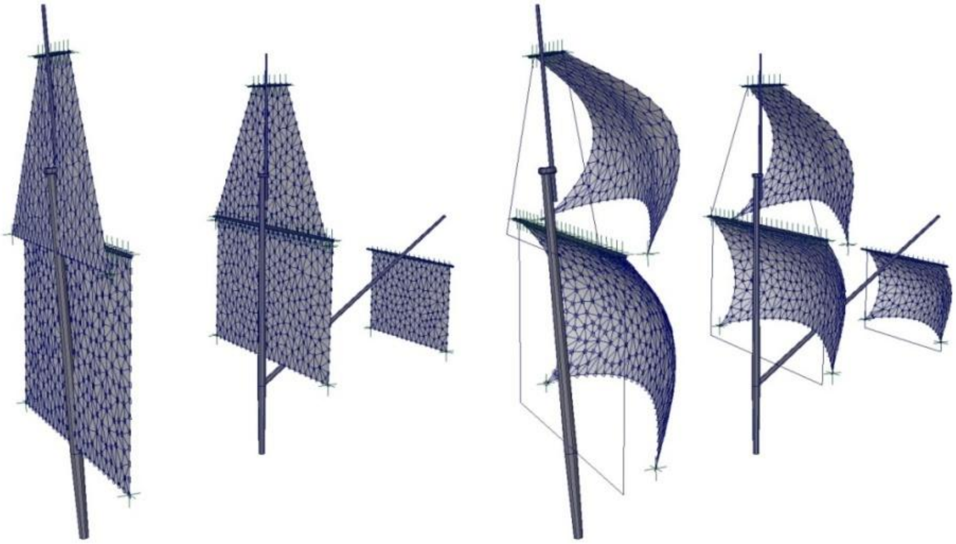

Fig. 28. Sails before and after cloth simulation

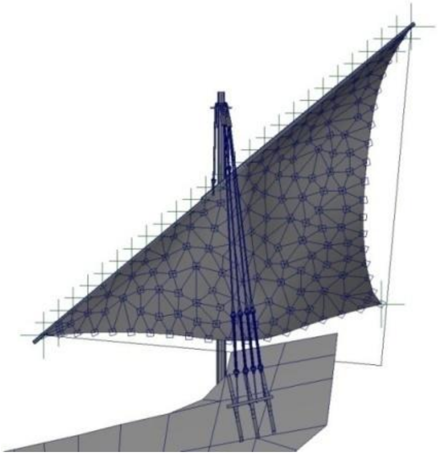

Fig. 29. Lateen sail billowing around backstays after simulation

Running Rigging

Running rigging is used to adjust sails. The arrangement of running rigging on the model was based upon sketches, photographs of similarly rigged vessels, and verbal communication with Dr. Castro. The arrangement of ropes is incredibly complex in a large sailing ship, so this project relies on a simplified rig to keep the polygon count and time invested lower. Lines were constructed to fasten the loose ends of the sails, which are tied off either to a nearby yard or to the outside of the ship with a metal ring, then run through a block at the sail tip. The running lines are made *fast* (unmoving) by *belaying* them to (wrapping them around) a cleat or a *belaying-pin* located in a *pin-rail*. These pin-rails and belaying pins were also constructed and mounted on the inside walls of the ship. The completed rigging is shown in Fig. 30.

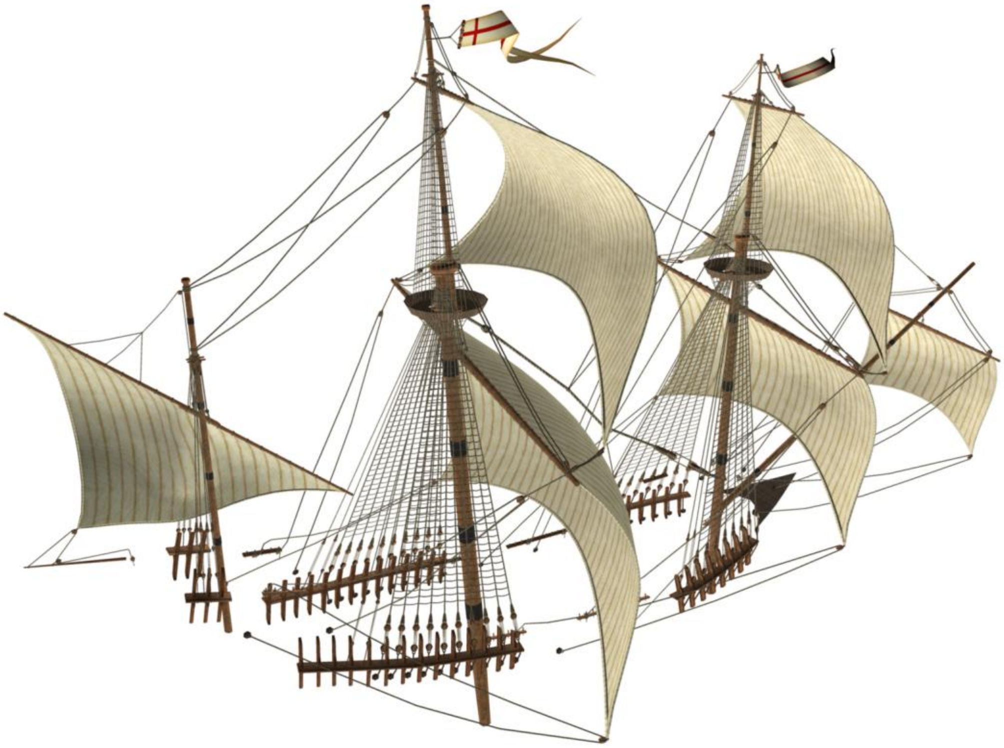

Fig. 30. Completed rigging, sails, and banners

Ship Equipment

Several unique models were created for various pieces of equipment that the ship would have had, including the ship's boat and berth (Fig. 31), windlass (Fig. 32), capstan (Fig. 33), bitts (Fig. 34), and pumps (Fig. 35).

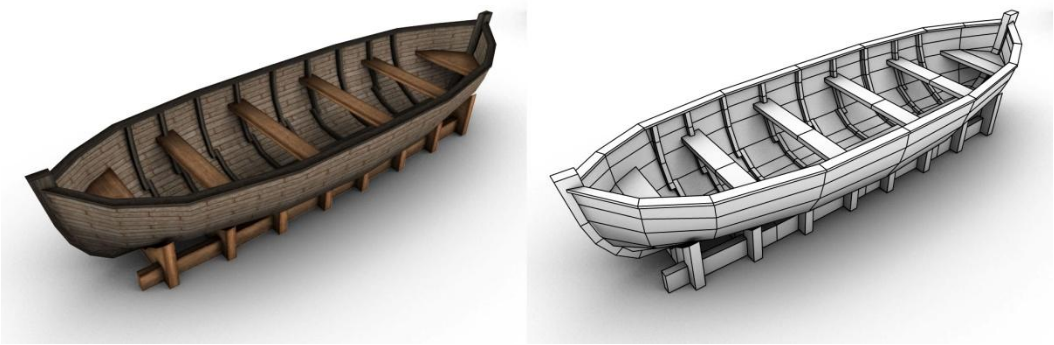

Fig. 31. Ship’s boat and berth, located central gun deck

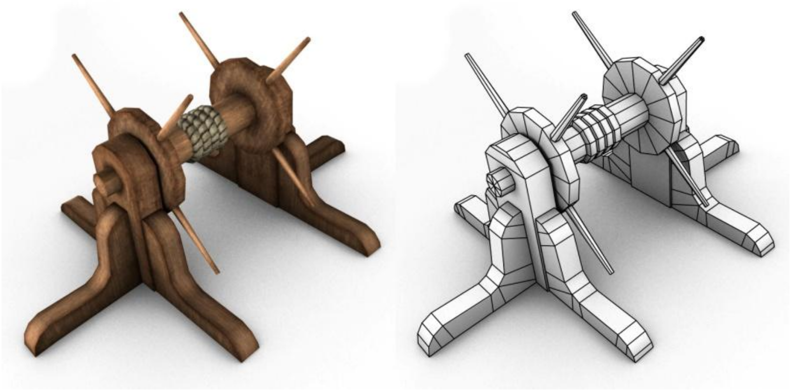

Fig. 32. Windlass, located aft gun deck

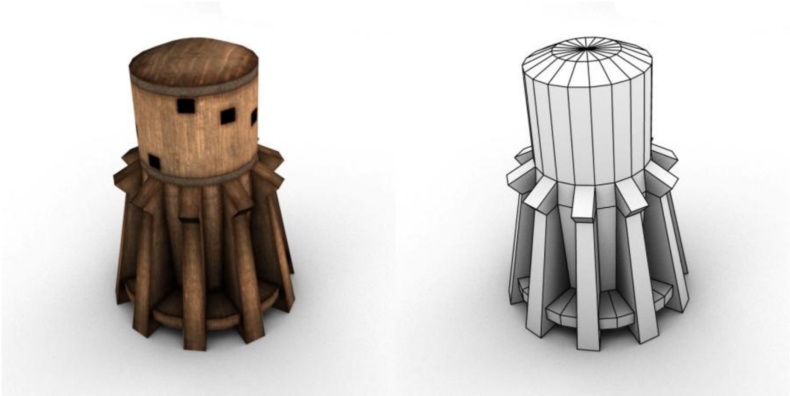

Fig. 33. Capstan, located aft main deck

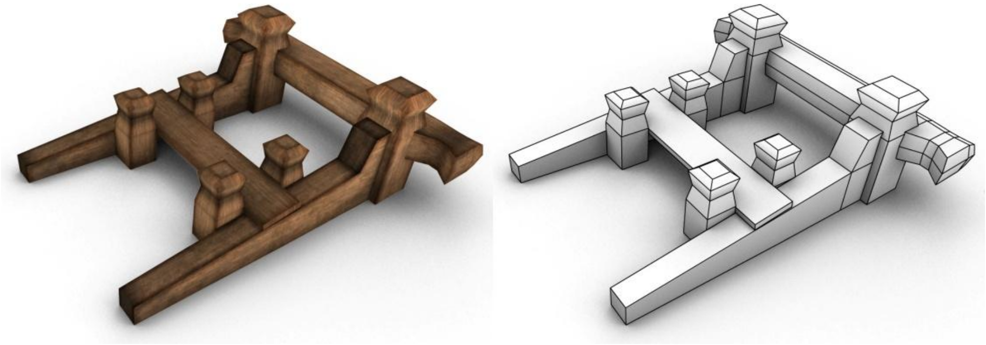

Fig. 34. Large bitts, located fore main deck. A smaller variation is also near the stern

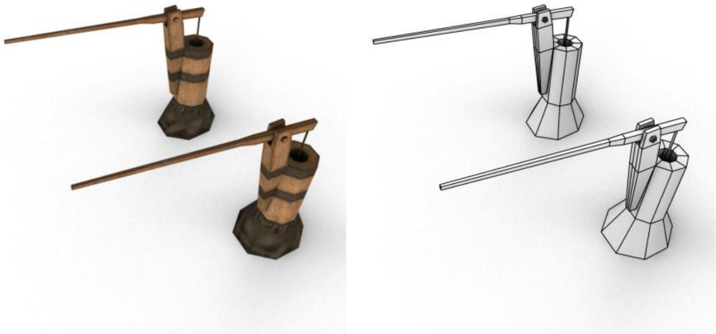

Fig. 35. Pumps, located on main deck, abaft the main mast

We decided to make eight cannons (see Fig. 36), three on each side of the vessel on the main deck, and two at the stern on the gun deck. The cannon models were placed as instanced geometry for the benefit to performance.

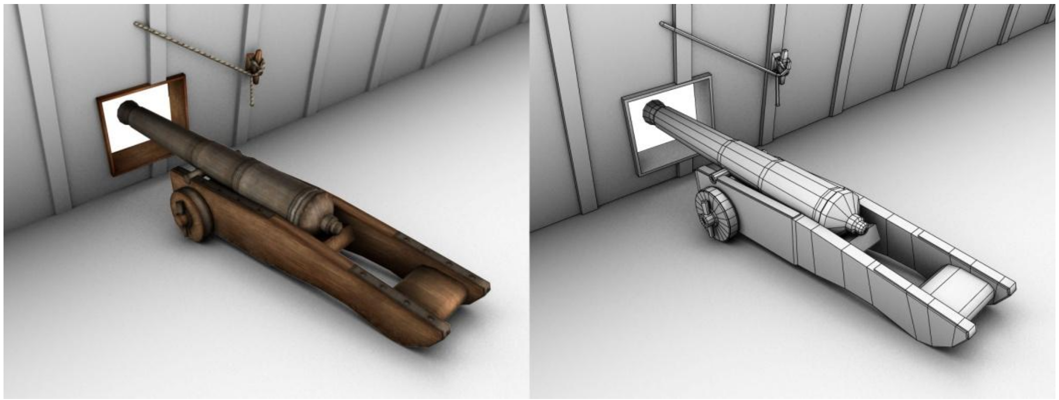

Fig. 36. Cannon model

All of the models were designed according to Dr. Castro's specifications. Photographs and drawings of similar equipment on other ships were also very helpful.

Cargo

The *Nossa Senhora dos Mártires* would have carried a variety of cargo, including crew provisions, personal provisions, personal belongings, and commercial goods. The crew and soldiers generally ate the daily rations supplied by the ship complemented with whatever they had brought in their boxes. The ship’s provisions for the long journey occupied an important part of the cargo space. Table 4 contains Dr. Castro’s estimations of the type, quantity, and ration amount of food available. The most important information for this visualization is the quantity of storage on the ship, since that determines the number of cargo models needed.

Table 4. Estimated types and quantities of ship's provisions [14]

Type	*Quantity*	*Rations (per person)*	*Storage*
Hardtack	1,074 *quintais*	989 grams, per day	**1,289 *sacas***
Wine	115 *pipas*	1.006 liters, per day	**115 *pipas***
Meat	1,086 *arrobas*	250 g, per day	**35 *pipas***
Hakes	150 dozens	5 fish, whole trip	**18 *pipas***
Olive Oil	315 *quartilhos*	0.435 l, whole trip	**1 *pipa***
Vinegar	13 *pipas*	0.796 l, per week	**13 *pipas***
Water	313 *pipas*	2.738 l, per day	**313 *pipas***
Salt	25 *moios*	0.3 l, per day	**20 *tonéis***
Sardines	130 *arrobas*	5.387 kg, whole trip	**13 *quartos***
Chick-peas	14 *alqueires*	0.5 l, whole trip	**1 *pipa***
Almonds	10 *alqueires*	0.36 l, whole trip	**1 *quarto***
Plums	10 *alqueires*	0.36 l, whole trip	**1 *quarto***
Lentils	10 *alqueires*	0.36 l, whole trip	**1 *quarto***
Mustard	2 *alqueires*	0.07 l, whole trip	**1 large jar**
Garlics	724 *cabos*	2 braids, whole trip	**20 baskets**
Onions	724 *cabos*	2 braids, whole trip	**70 baskets**
Sugar	8 *arrobas*	325 g, whole trip	**8 large jars**
Honey	8 *arrobas*	325 g, whole trip	**48 jars**

The total number of ship's provisions from Table 4 includes 1289 sacks, 20 *tonel*-size barrels, 496 *pipa*-size barrels, 16 *quarto*-size barrels, 90 baskets, 9 large jars, and 48 small jars. In addition to the ship's provisions, the wealthy passengers brought an enormous amount of provisions for themselves and their servants. These personal provisions would have included an estimated 20 *tonel*-size barrels, 100 *pipa*-size barrels, 40 *quarto*-size barrels, 150 baskets, and 100 small jars. Images of the sack, barrel, basket, and jar models can be seen in Figs. 37 to 40.

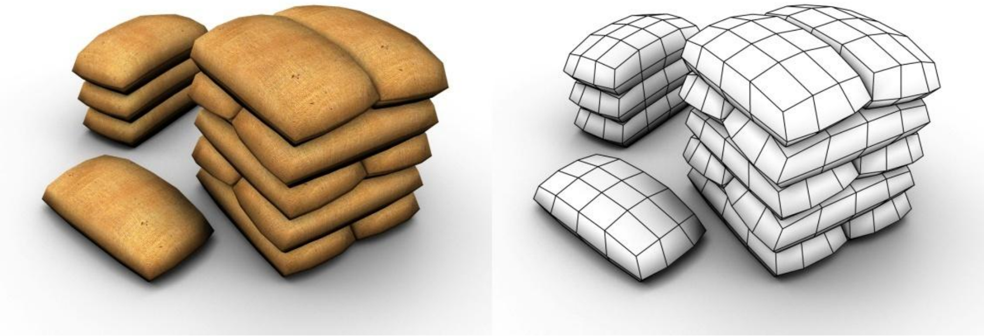

Fig. 37. Models of sacks

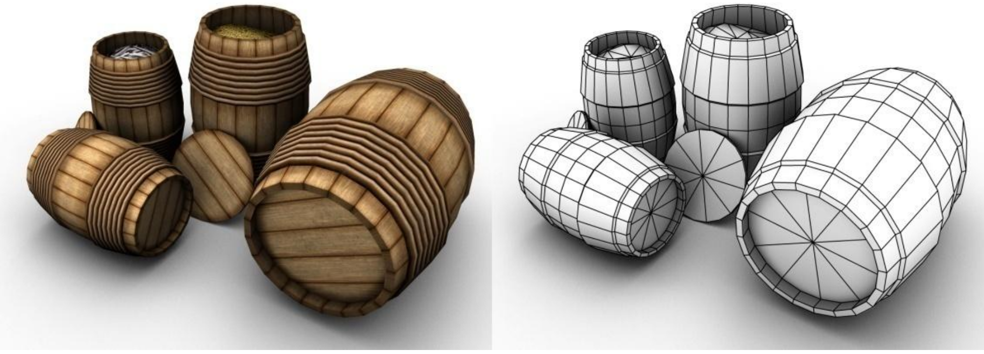

Fig. 38. Models of barrels (tonel, pipa, and quarto sizes)

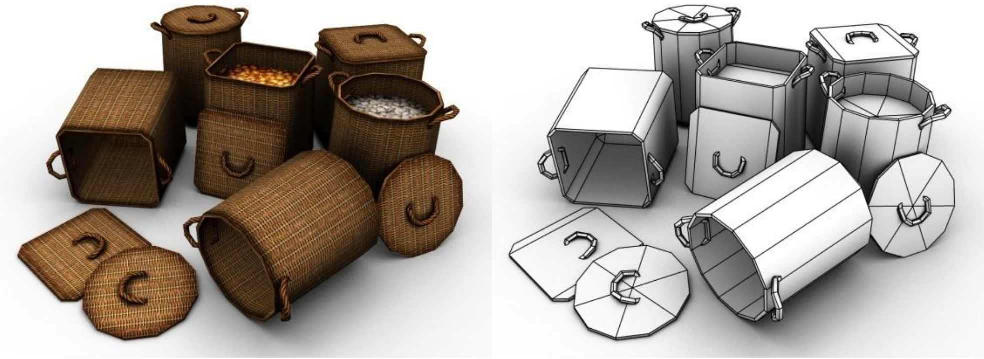

Fig. 39. Models of baskets (square and round variations)

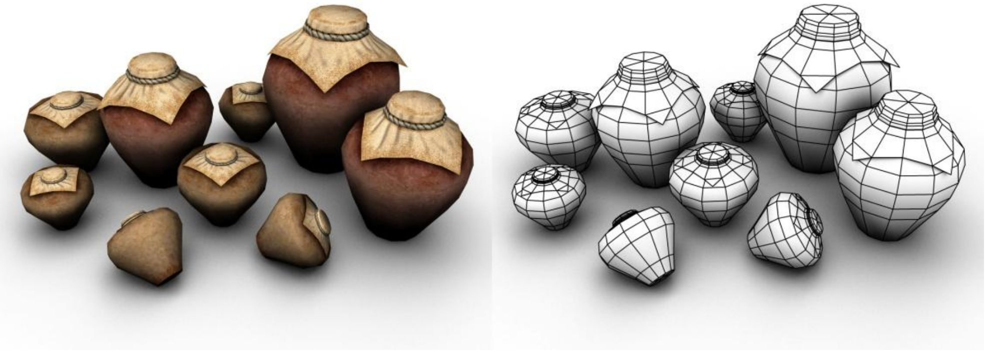

Fig. 40. Models of jars (large and small variations)

The personal provisions would have also included live animals (approximately 5 sheep, 2 pigs, 350 chickens and ducks, and 100 rabbits), but these were not included in the visualization due to time constraints.

Other personal belongings would have included the crew's boxes, which Dr. Castro estimates at 253 boxes (106 single owned, 147 shared). The box model is shown in Fig. 41.

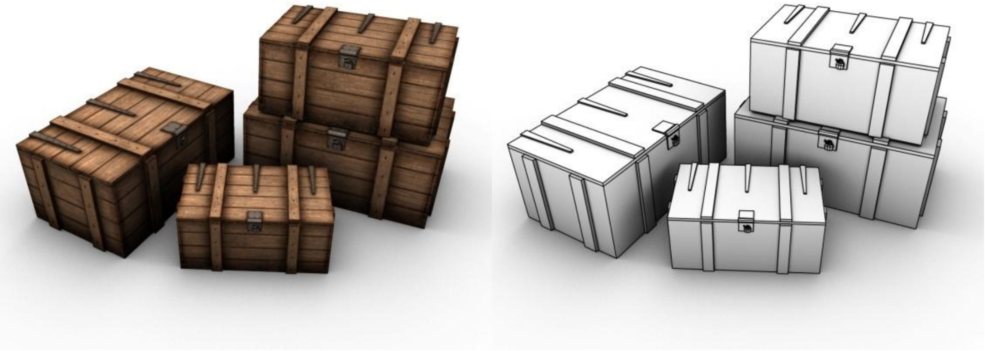

Fig. 41. Models of boxes

All of the models were created using the basic measurements provided by Dr. Castro in Table 5, descriptions, and when available, iconography. Each model was UV mapped and textured. Although the barrels and sacks are relatively standardized sizes, the sizes of the baskets, jars, bales, and boxes would have varied. This variation was simulated through some randomization of model scale while placing these items on the ship.

Table 5. Approximate dimensions of cargo

Type	*Width*	*Height*	*Length*	*Diameter*
Sack	0.60 m	0.40 m	1.20 m	-
Tonel barrel	-	1.54 m	-	1.02 m
Pipa barrel	-	1.23 m	-	0.82 m
Quarto barrel	-	0.98 m	-	0.64 m
Basket	-	0.80 m	-	0.60 m
Small Jar	-	0.35 m	-	0.35 m
Large Jar	-	1.00 m	-	0.80 m
Bales	0.70 m	0.50 m	1.50 m	-
Boxes	0.66 m	0.50 m	1.10 m	-

The primary commercial cargo consisted of peppercorn, which was stored in the hold of the ship. Special boxes were built just above the ballast and orlop deck, and were closed and caulked to withstand the long voyage. The boxes reach the full height of the hold, with small hatches cut into the deck above each box, through which the pepper was loaded. The models for the pepper boxes were based on sketches, measurements, and conversations with Dr. Castro. The resulting model featured twelve compartments, six on each side of the ship, 18 meters in length, which allow the storage of approximately 200 tons of peppercorns. A narrow hallway separates the sides to provide access to additional storage areas at the bow and stern. A surface textured with peppercorns was modeled to fit the shape of the surrounding deck and hull. As a result, removing the side of the hull and other decks reveals the compartments full of peppercorn . Additional commercial cargo was created using bales. The bale model is shown in Fig. 42. A total of 46 bales were placed on the ship.

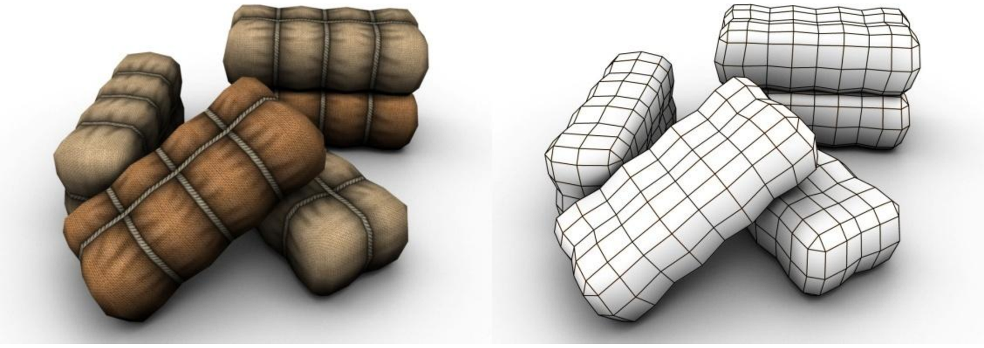

Fig. 42. Models of bales (light and dark variations)

Interior Distribution

There is very little information indicating how the interior space was divided and how people and cargo would have been distributed throughout the ship. According to Dr. Castro, the only Portuguese naus preserved well enough to have information on the position of interior walls were destroyed by treasure hunters. He also notes the rarity of documents from the period. One of the few documents is a 1607 diagram (see Fig. 43) showing the distribution of the cargo and crew in a two-decked nau. Although rudimentary, this diagram provided a starting point for the interior distribution of space in the virtual model.

However, since the *Mártires* had more than two decks, placing the rest of the walls is a conjecture. Likewise, the distribution of the actual cargo is almost entirely speculation. Under Dr. Castro's direction, walls were built and space allocated for people and cargo.

According to Dr. Castro, the upper decks were primarily occupied with people and animals, while the lower decks were primarily filled with cargo.   These decks are labeled in Fig. 44.

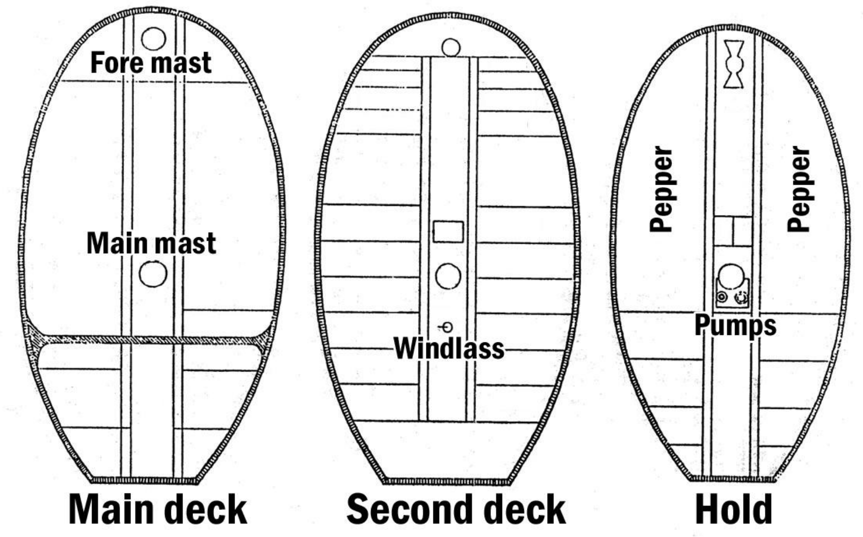

Fig. 43. Diagram showing nau distribution. Figueiredo Falcão (1607) [9, p. 20]

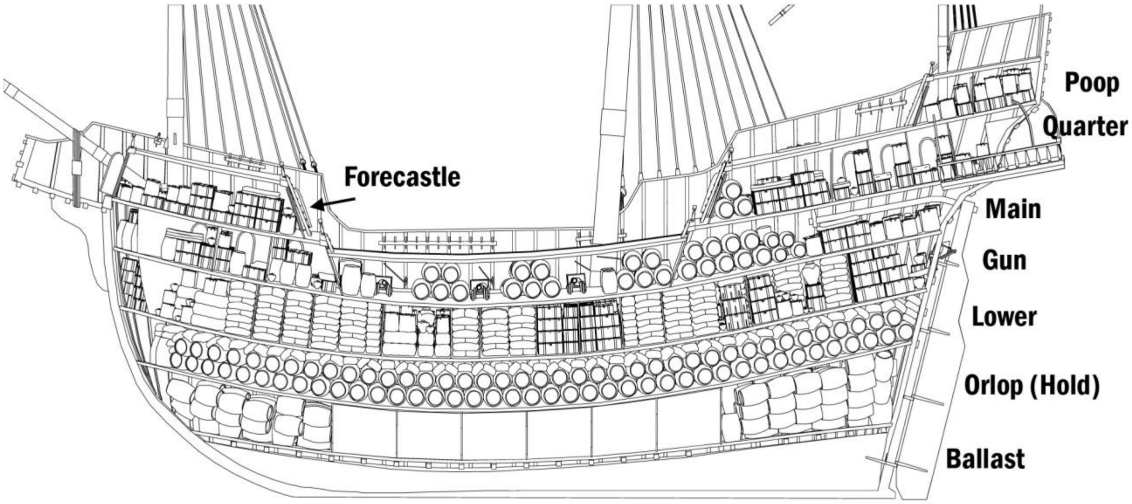

Fig. 44. Decks labeled and filled with cargo (side view)

In the hold, on the *orlop deck*, are the huge pepper boxes. The remaining space towards the bow and stern was filled with 80 tonel-size and 112 pipa-size barrels (see Fig. 45).

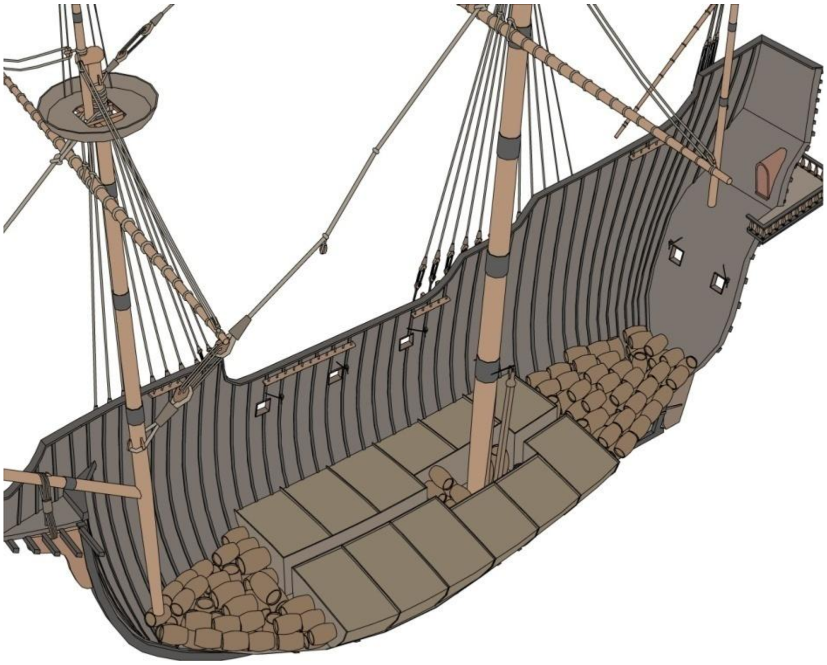

Fig. 45. Orlop deck (hold) filled with pepper boxes and barrels

The *lower deck*, above the hold, was left undivided by walls and filled with 414 pipa-size barrels. 360 sacks of hardtack fill the remaining space on top of the barrels. 35 small jars are also scattered around amongst the barrels (See Fig. 46).

The next deck, the *gun deck*, was divided similarly to the 'second deck' illustration in Fig. 44. On each side of the ship are multiple rooms, filled primarily with 929 sacks, 46 bales, 85 baskets, 8 large pots, 75 small pots, and 129 boxes (see Fig. 47). Two rooms on the starboard side of the gun deck were reserved for a kitchen and firewood storage to be added later.

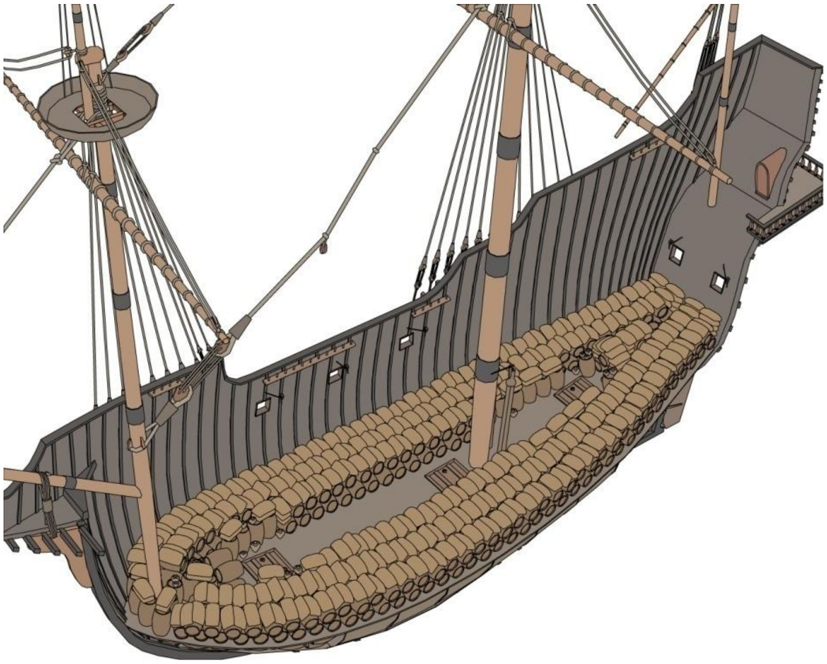

Fig. 46. Lower deck filled with cargo

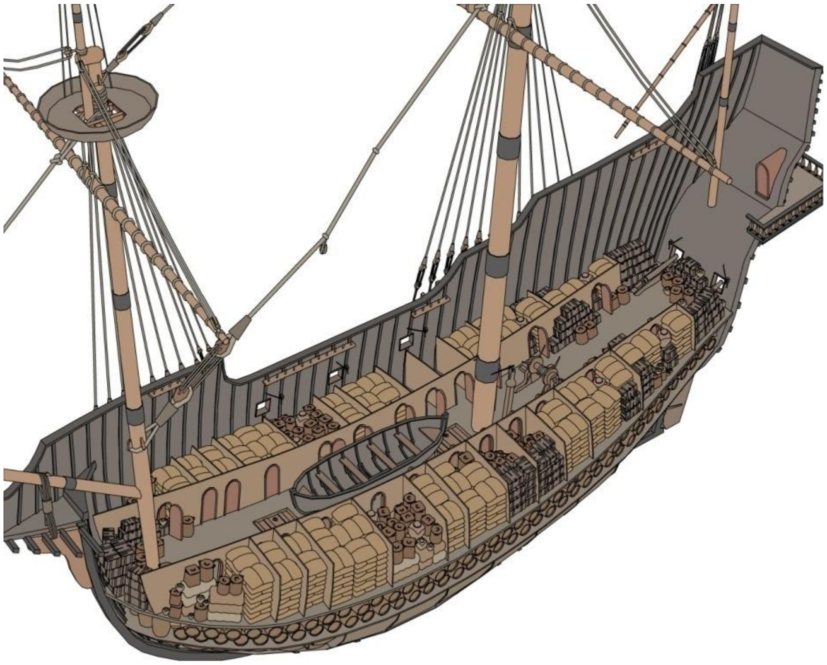

Fig. 47. Gun deck filled with cargo

Above the gun deck is the *main deck* (see Fig. 48). The bow area is roofed by the forecastle, and was divided into small rooms. The stern area is roofed by the quarterdeck and was left open for the operation of the rudder. The central area is exposed to air, sometimes called the *weather deck*. On this level I included 64 pipa-size barrels, 56 quarto-size barrels, 20 small jars, 1 large jar, 46 boxes, and 27 baskets. I left this level somewhat open for occupation by people, and included 22 mattresses tucked away or laid out for use.

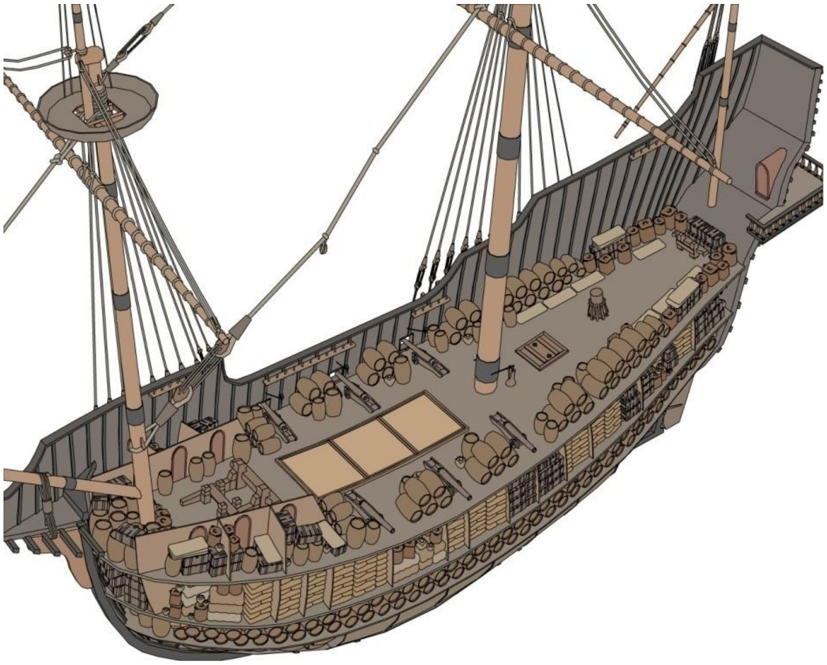

Fig. 48. Main deck filled with cargo

As shown in Fig. 49, above the main deck is the *forecastle* (bow end) and the *quarterdeck* (stern end). In the forecastle, I placed 3 small jars, 25 boxes, 50 baskets, and 4 mattresses. In the quarterdeck, I placed 6 pipa-size barrels, 45 boxes, 42 baskets, and 13 mattresses. The quarterdeck is partially open for operation of the whipstaff, but the remaining area is divided into a hallway and rooms leading to the balcony.

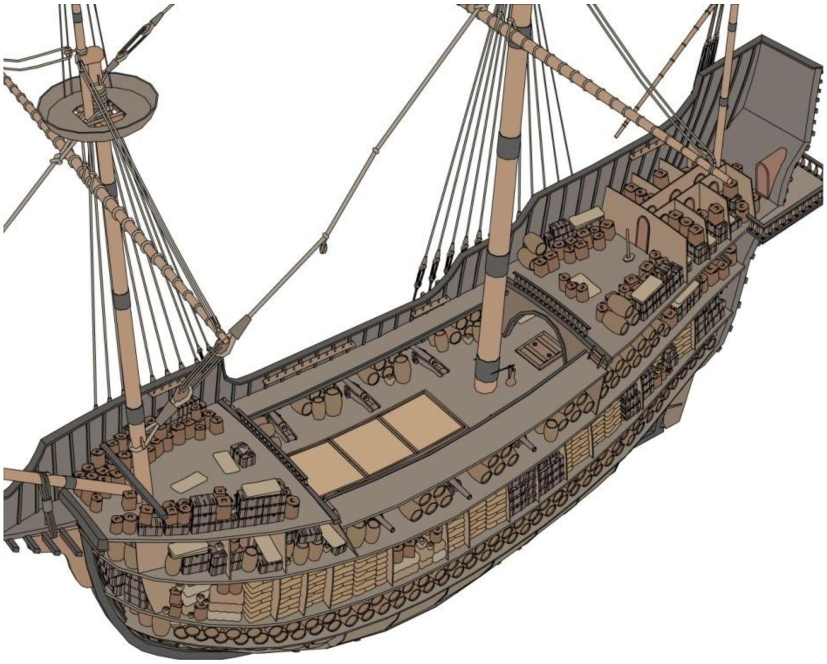

Fig. 49. Forecastle and quarter deck filled with cargo

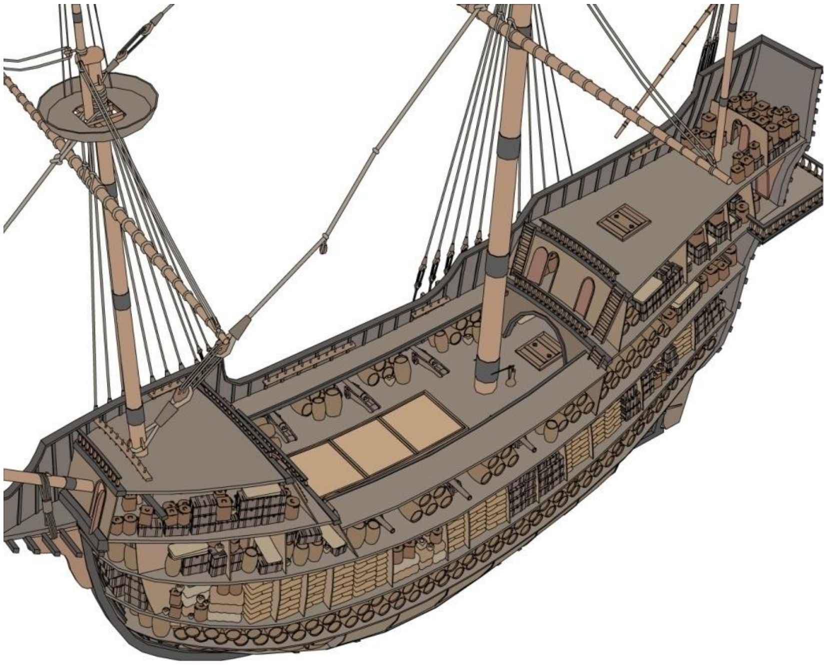

Fig. 50. Poop deck filled with cargo

Above the quarterdeck is a small area called the *poop deck*, in which I placed 8 boxes and 36 baskets (see Fig. 50). Fig. 51 shows the full hull (with the portside of the ship replaced) and the roof of the poop deck.

The final count matches Dr. Castro’s projected cargo counts, with 1,289 sacks, 40 tonel-size barrels, 596 pipa-size barrels, 56 quarto-size barrels, 240 baskets, 9 large jars, 148 small jars, and 243 boxes. On top of his data, I have added 46 bales and 39 mattresses.

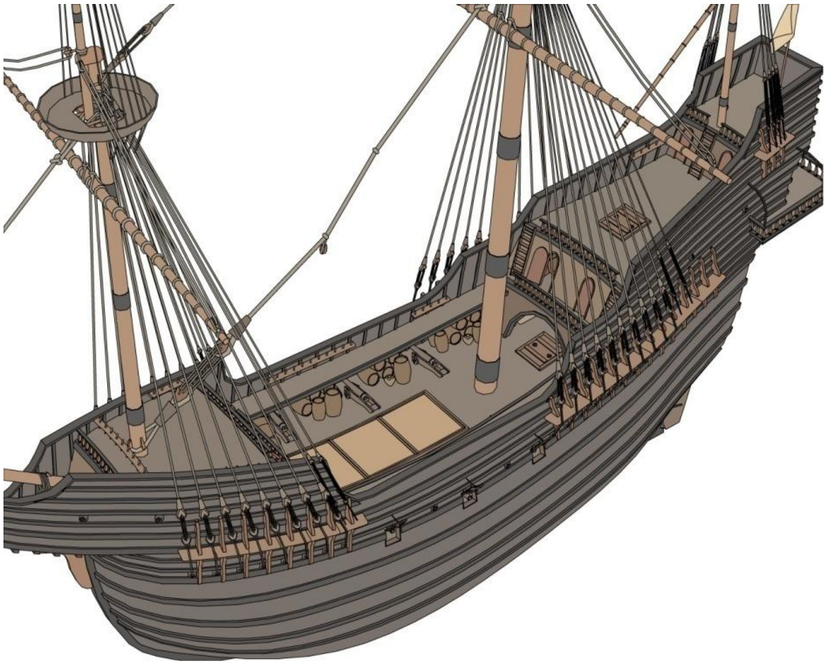

Fig. 51. Complete with portside of ship

Environment

To complete the visualization of the Pepper Wreck, the model of the ship was placed in a plausible environment. Placing the ship in open water reduces the need for detailed landscape or city modeling, but having a meaningful terrain and a landmark in view makes the environment interesting. We decided to place the ship near the wreck site, which is a few miles outside Lisbon, Portgual. The site is at the mouth of the Tagus River, in view of the fort São Julião da Barra.

The Shuttle Radar Topography Mission (SRTM) is an international project led by the National Geospatial-Intelligence Agency (NGA) and the National Aeronautics and Space Administration (NASA). During a mission in February 2000, a radar system flew onboard the Space Shuttle Endeavour. The system obtained elevation data over 80% of the Earth's surface, creating the most complete high-resolution digital topographic database to date [15]. This topographic data is freely available for download via FTP on the SRTM website. Each data file covers a one-degree-of-latitude by one-degree-of-longitude block of Earth's surface and contains raw elevation data measured in meters above sea level. The Pepper Wreck is located within latitudes 38 to 39 North and longitudes -9 to -10 West. I downloaded this data from the SRTM website.

3DEM is a freely available software product of Visualization Software LLC by Richard Horne [16]. This software has the ability to merge multiple DEMs (Digital Elevation Models) to provide detailed overhead maps, 3D renders, and flythrough animations of large surface areas. The elevation data for the area surrounding the Pepper Wreck was imported into 3DEM and converted into a grayscale height map for use in Maya (see Fig. 52). The black portions of the image indicate the lowest terrain at sea level. The white portions indicate the highest points, which in this case is 508 meters above sea level.

3DEM can also use satellite imagery as overlays on elevation data. According to the software documentation, the best images to use as overlays are the orthorectified Landsat 7 Enhanced Thematic Mapper (ETM) data. Landsat 7 ETM images consist of grayscale images recorded in different color bands. Bands 3, 2, and 1 are a reasonable match to red, green, and blue visible light. Images of these three bands covering the Pepper Wreck elevation data were downloaded from Global Land Cover Facility [17]. The grayscale images were combined into one color image as RGB channels in Photoshop and adjusted for color balance and contrast. Metadata included with the images provides geographical information such as latitude and longitude of the image corners. 3DEM is able to match the overlay to the elevation data through manual referencing of points placed at the image corners with the geographical information. The resulting image was exported into a texture map for use in Maya (see Fig. 52).

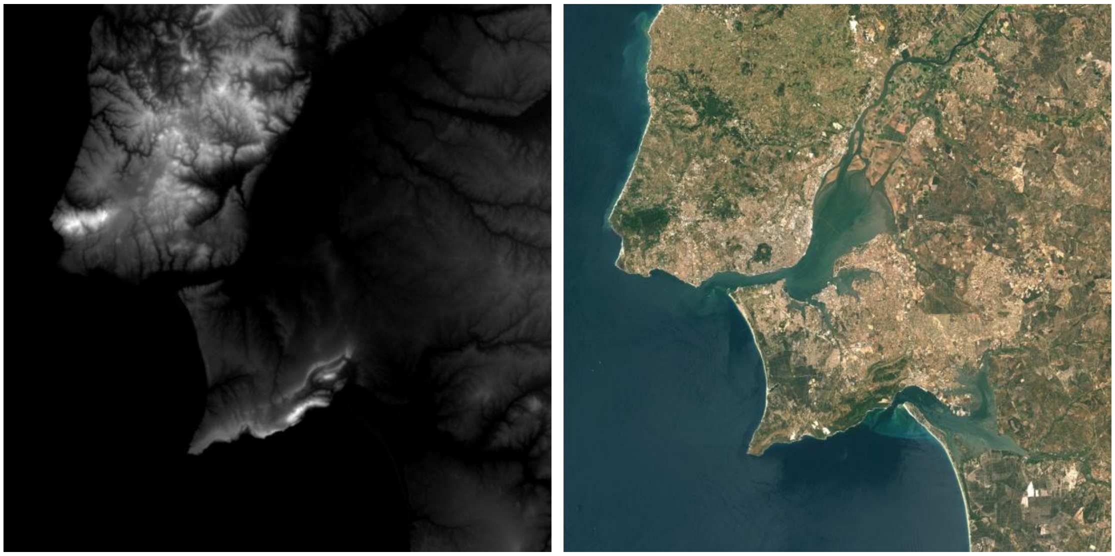

Fig. 52. Terrain height map (generated from SRTM data) and color map (generated from Landsat data)

In Maya, the height map was applied as displacement to a flat plane. The filtering settings of the height map controls the smoothness of the terrain by blurring the image, which is useful when working with lower polygon counts. The alpha gain of the height map controls the amount of displacement. To create the correct terrain scale, the alpha gain is calculated by a simple equation: *scene distance / desired distance = alpha gain /*

maximum elevation. Scene distance is the measurement between any two points of the model in Maya. I was able to determine desired real world distance by using measurements of the terrain in 3DEM. Then, using the maximum elevation (508 m), I was able to determine what value to use for alpha gain to create the correct model displacement. The tessellation division factor of the plane controls the quality of the displacement, with higher divisions resulting in more polygons and a higher degree of accuracy to the height map. Maya also offers *feature-based displacement*, using a threshold value, to determine where to cut off data to make flat planes. This helps to simplify the terrain model where the change in elevation is not noticeable, and put more geometry where the hills are. Finally, the automatic process to convert displacement to polygons was used to generate a 3D model of the landscape.

The geometry generated by Maya from the displacement map was cleaned up, reducing the polygon count where possible. The boundaries of the geometry were modified to be a circular shape to help create the appearance of a flat horizon from the ship. The final geometry is shown from an overhead view in Fig. 53, and from ship level in Fig. 54.

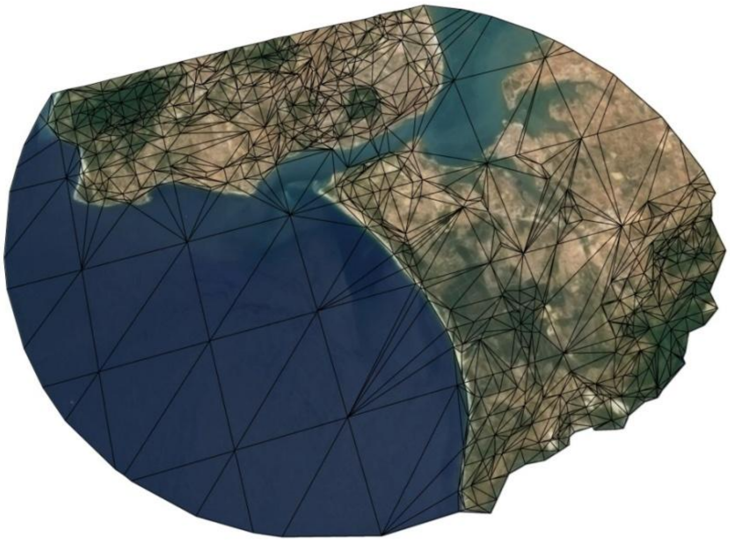

Fig. 53. Terrain model with satellite image applied as a texture

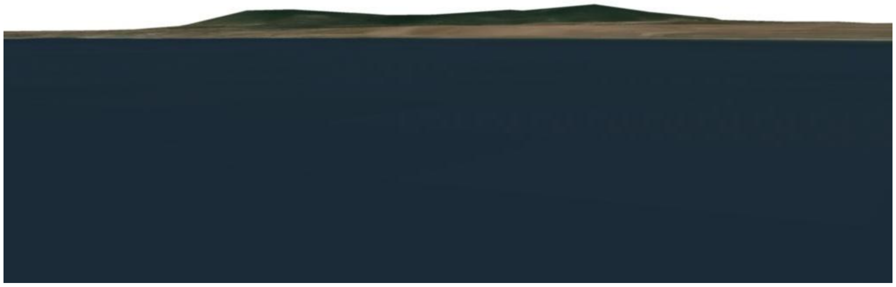

Fig. 54. View of the terrain from ship level

On the interactive system, a background such as a sky is created by using a *skybox*. A skybox is a cube that is rendered surrounding all other geometry in the scene. Each side of the cube is textured with an image looking in the six axial directions, resulting in a seamless backdrop which appears infinitely far away. Fig. 55 shows the six textures which I rendered using freely available scenery rendering software, Planeside Software's Terragen [18]. I used a simple script to automate the process of orienting the camera and rendering each image.

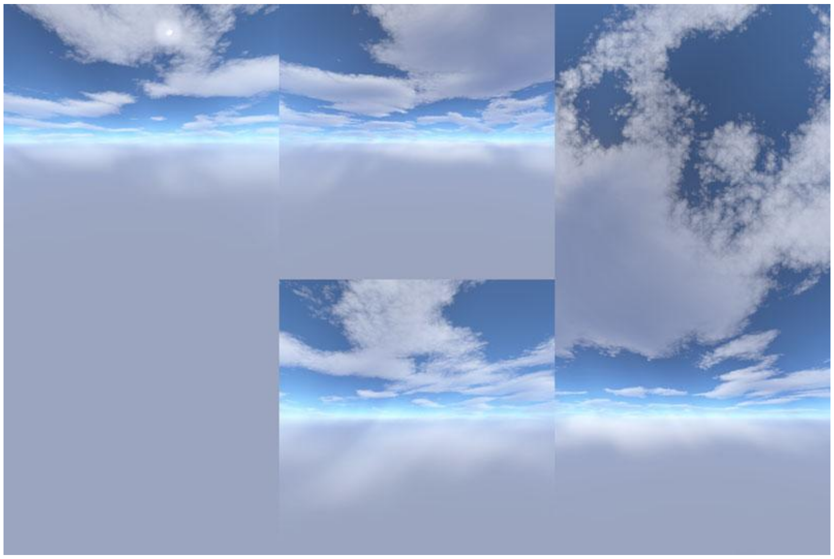

Fig. 55. Sky box texture created using Terragen

Interactive Installation

To view the Virtual Nau on the immersive display, the entire scene was exported from Maya into a particular format and organizational structure understood by the immersive system software. Once the scene was loading correctly using a single display, the project files were propagated to the other computers. The scene can then be loaded and projected onto the facets of the spatially immersive display. Then the scene can be navigated interactively using a keyboard or mouse. This entire process was made significantly easier using scripts provided to me by Dr. Parke, with the exception of the first script which I wrote myself.

I wrote the first script to prepare the Maya scene for export. Throughout the process of testing my project on the display engine, I was required to move back and forth between the working version of my scene, in which all of the objects retained deformers and objects were separated, and the export-ready version where all deformers were removed and objects sharing textures were combined. I the script to automate the process to prepare my scene for a second export script described below. The script deletes the history (active deformers, for example) of objects and deletes hidden geometry, such as reference models. Then the script loops through each of the materials in the scene and combines any objects that are sharing the same textures. The combined objects are renamed appropriately based on the name of the texture. This script does not apply to objects which are instanced, however, which must be exported in a hierarchal fashion using the export script.

The second script carries out the actual export and creates the file format and structure required by the immersive system software. This script exports each polygon object from the Maya file into the object (.obj) file format. The object format is a specific file format describing a polygonal object which includes vertex positions, vertex normals, UV data, and texture information. The script also exports the scene’s lights into a light

(.lit) file, which contains the position, rotation, color, and other settings for each light. The script finally creates a scene (.sce) file, which points to all of the objects and lights that need to be loaded by the engine.

After export, I modified the scene file to set up the scene for viewing. A variety of settings were needed including camera orientation, camera field of view, camera clipping planes, skybox texture, viewer movement, and viewer rotation speed. In addition, material properties of the objects were specified in this file. I set up and tested the project using a single monitor display. Fig. 56 and Fig. 57 are screen captures from project while running on a single display.

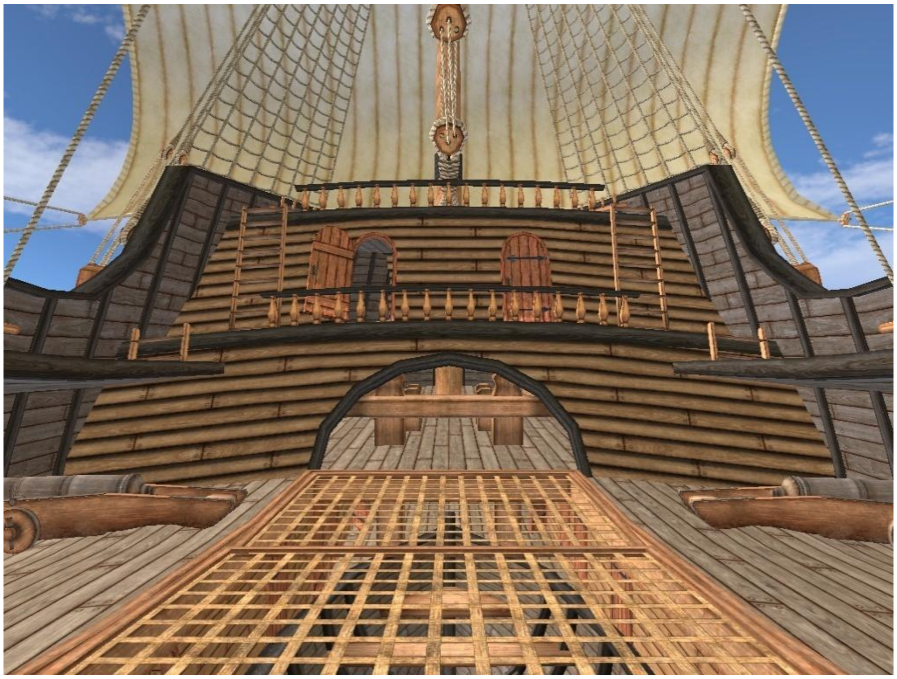

Fig. 56. Screen capture 1 of project running on Guppy 3D (single display)

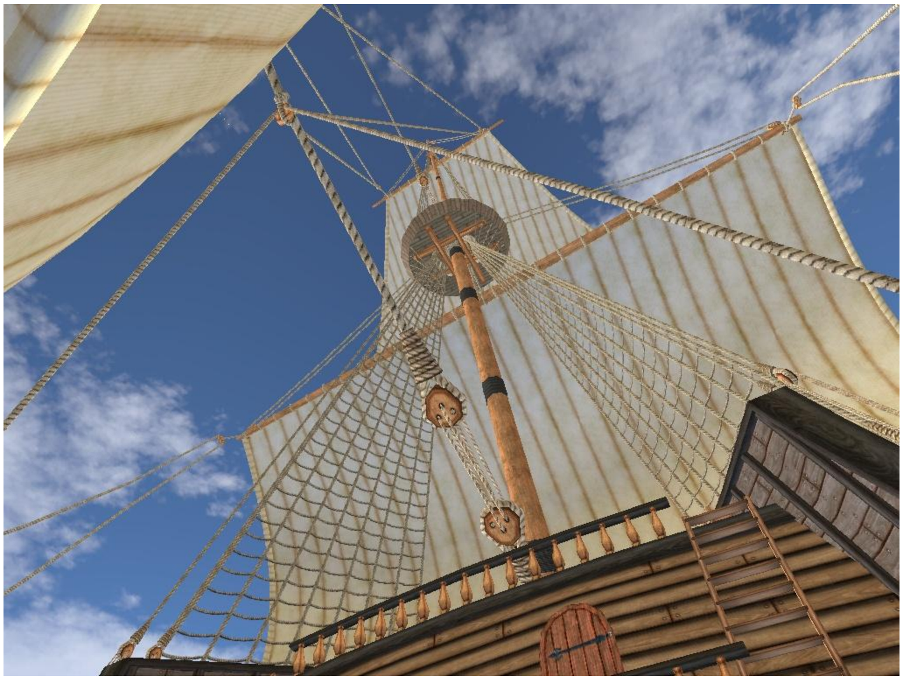

Fig. 57. Screen capture 2 of project running on Guppy 3D (single display)

The export of the Virtual Nau from Maya onto the immersive display engine was a trial and error process. Quite often, issues came up. Some of the problems were due to modeling errors with the 3D geometry, which were fixed by cleaning up the geometry. Some of the problems arose due to limitations in the immersive system itself. There are two versions of the immersive display software, and initially my project was viewed using the older version, called *3D Engine*. I encountered several limitations using this version, such as lack of support for alpha masks and double-sided surfaces. These issues were serious, because my project made specific use of textures with transparency (ratlines) and flat surfaces (sails) to reduce polygon counts. The second version of the software is called Guppy 3D. Using this version, textures with alpha masks and double-sided surfaces displayed correctly.

Once I was satisfied with how the project was running on a single display, it was ready to be tested on the immersive display. The project files first had to be copied to each display computer. This was done automatically using a propagation script. Figs. 59, 60, and 61 are photographs of the project running using the faceted immersive display.

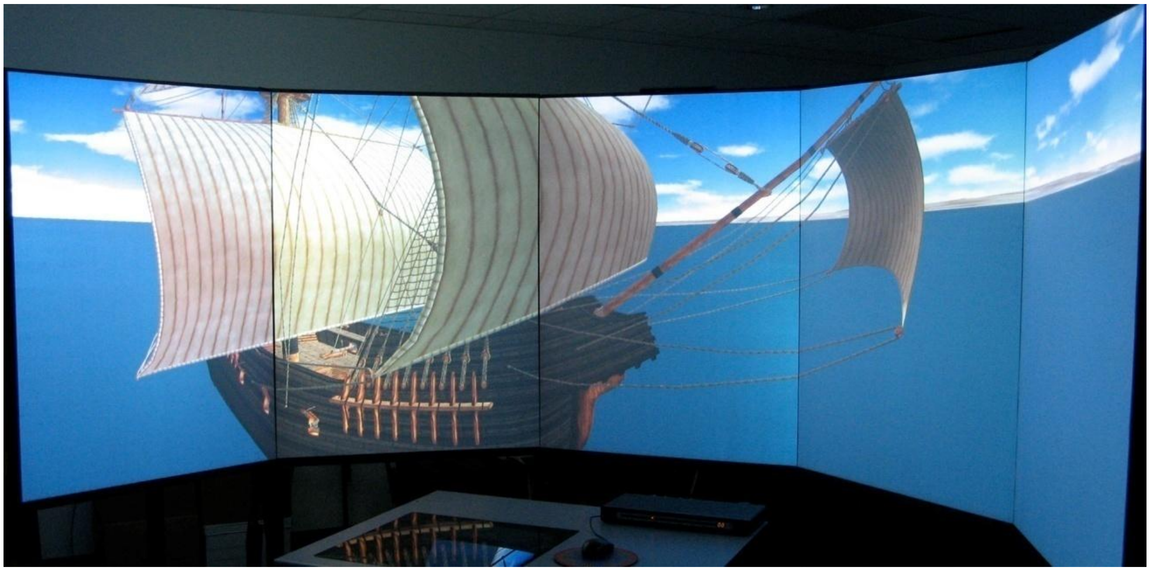

Fig. 58. Photograph 1 of project running on Guppy 3D (immersive display)

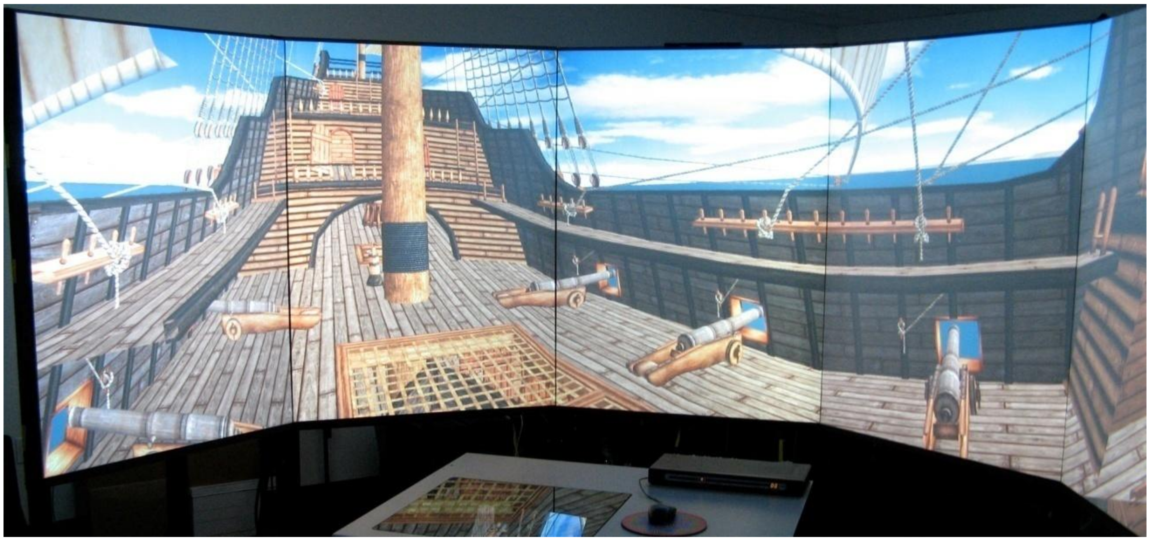

Fig. 59. Photograph 2 of project running on Guppy 3D (immersive display)

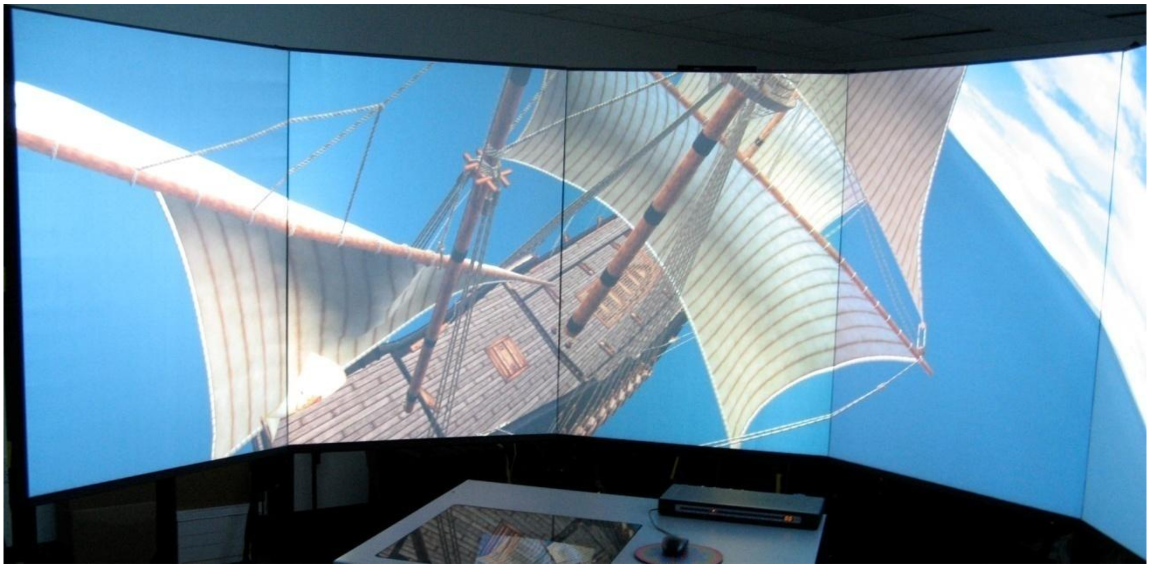

Fig. 60. Photograph 3 of project running on Guppy 3D (immersive display)

People

Due to the seemingly never-ending nature of this project and the complexity of solving the cargo arrangement, it was inevitable that things I had originally planned to do were not completed at the time this thesis was presented. The biggest of these uncompleted "sub-projects" was the addition of detailed, textured, and posed people models to the ship. It is extremely helpful to have humans as a part of the model to give the viewer a sense of scale. In lieu of the more detailed people, I ended up using a few simple human scale models on ship as scale reference. However, some preliminary research, modeling, and texturing work was conducted on this front, I simply ran out of time to finish this task. Some of that preliminary work is detailed here, along with descriptions of what I hope to complete in the future.

There would between 400 and 450 people models aboard the ship, including men, women, and children. However, I estimate only a handful of unique models will need to be developed, enough to distinguish between very basic gender, age, and class groups.

The women had dresses, the priests had robes, the crew and soldiers wore pantaloons and shoulder puffs of various sizes. Crew members of rank or import will be textured to appear more sophisticated in their best clothes. These people, numbering about fifteen, would include the captain, clerk, chaplain, master, pilot, second pilot, boatswain, boatswain's mate, cooper, purser, bailiff, barber, constable, and caulkers. Lesser crew members would be textured in more plain outfits. These people would include roughly 50 seamen, 20 gunners, 50 ship's boys, and 4 pages. All of the models of people can be reused with a variation in size, pose, and texture for maximum efficiency.

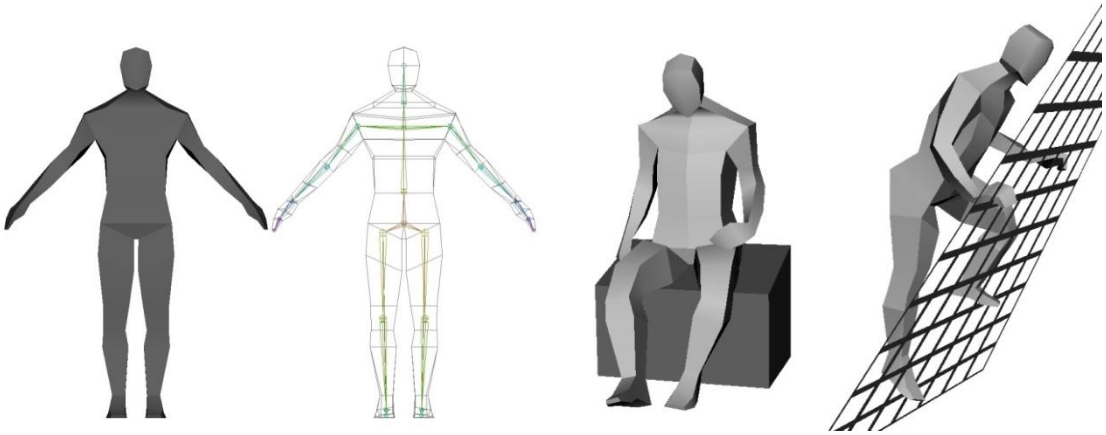

Fig. 61. Preliminary figure rig and pose tests

I have already developed a simple and generic rig of bones and joints for posing the models (see Fig. 61). Joints were placed to create basic rotations in the arms, legs, hands, feet, torso, and neck. This rig uses a technique known as inverse-kinematics for the legs and arms. *Inverse kinematics* (IK) calculates the required articulation of a series of joints, such that only the final joint, such as the wrist or ankle, must be directly controlled. The elbow and knee will follow and move appropriately, within limits. This is in contrast to *forward kinematics* (FK), which requires the manual movement of each joint in the arms and legs. The rig is basic enough that it could be adapted to work with any of the people models.

Significantly more textures than people models will be necessary as the people on board should look varied from one another, while clothed appropriately for their period. This is where iconography becomes very useful, especially the Japanese Namban painted screens (1590 - 1630). Namban screens such as the example shown in Fig. 62, provided me with information about the shape, color, and patterns of clothing. More information about Namban art can be found in [19] and [20].

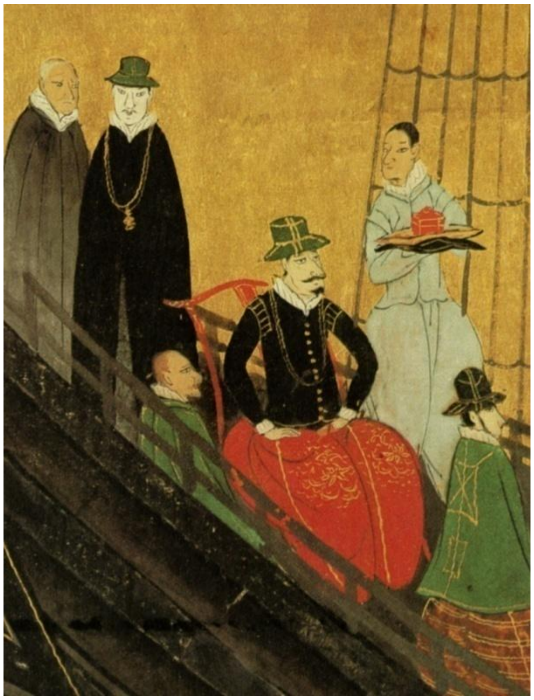

Fig. 62. Detail from a Namban painted screen (circa 1590 - 1630)

Using the Namban example shown in Fig. 62 as reference, I modified the model for clothing shape and created a texture. This model and corresponding texture are shown in

Fig. 63. I planned to create a series of textures that have variations in color and pattern based upon the iconography.

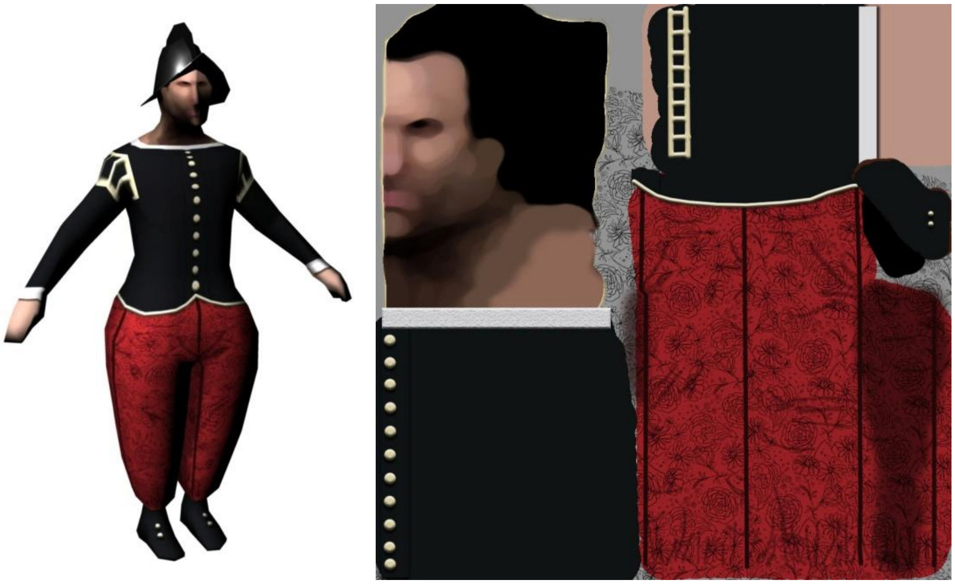

Fig. 63. Preliminary Namban figure model and texture

CHAPTER VI

CONCLUSION

Future Work

The interpretive development cycle discussed in Chapter I and the iterative project methodology discussed in Chapter IV make this project a constantly evolving one. I believe it will never be truly “finished” because archaeological research is always uncovering new information, which could be applied to the model.

Future work on this model could include modifications to demonstrate alternative reconstructions or hypotheses. The model could also be used to form answers to alternative research questions, such as the appropriation of space by people. By expanding on my preliminary modeling of people for the ship, we could attempt to learn how 400 to 450 people would fill the remaining space after it has been loaded with cargo.

Technology makes archaeology more accessible to the public than ever before, which is why I am especially excited about the possibilities of this visualization. In fact, the nature of this model lends itself to many potential applications ranging from entertainment to education – it is not limited to use on the immersive system.

This model could be used for entertainment purposes, such as a computer game, or for creative purposes using artistic rendering techniques. I hope to port the ship model onto a game engine and develop an interactive exploration experience. This software could be published on CD or DVD, or made available for download on the Internet.

From an educational standpoint, this model could be displayed in an interactive or video museum kiosk as part of a larger exhibit about the Pepper Wreck. The model is detailed enough to be rendered to video and used as a segment in a documentary. Presentations can be published on the Internet. It could be the focus of an educational website enhanced with QTVR, images, audio, videos, and other information to help users understand the history, cultural significance, and archaeological evidence behind the Pepper Wreck project.

Evaluation

Although thousands of books have been written about the European maritime expansion of the 16th century, still not much is known about the ships on which that expansion was carried out, especially the Portuguese nau. Therefore, one of the most important outcomes of this project was the construction of a plausible virtual model of a Portuguese India nau. This model of the *Nossa Senhora dos Mártires* is the first complete reconstruction of such a ship based on archaeological data, and represents a significant advance to the understanding of this vessel type, both at a scholarly level and, hopefully, at a popular level as well.

I believe this model adds a new dimension to the understanding and interpretation of the India nau, and paves the way for future interpretations to be discussed and created. Further analysis will help us refine this study and produce a more reliable virtual hypothesis of a late sixteenth century India nau.

REFERENCES

[1] G. Lock, *Using Computers in Archaeology: Towards Virtual Pasts*. London: Routledge, 2003, p. 13.

[2] Apple Inc. QuickTime VR. [Online]. Available: http://www.apple.com/quicktime/technologies/qtvr/ [Accessed: May 2008].

[3] M. Saul, "The Lelystad Karveel: reconstruction of a seventeenth-century Dutch passenger ferry," M.S. thesis, Dept. Architecture, Texas A&M University, College Station, TX, 1994.

[4] A. Hazlett, "The *Nau* of the *Livro Nautico*: reconstructing a sixteenth century Indiaman from texts," Ph.D. dissertation, Dept. Anthropology, Texas A&M University, College Station, TX, 2007.

[5] J. Kantner, "Realism vs. reality: creating virtual reconstructions of prehistoric architecture," *Virtual Reality in Archaeology*, J. Barceló, M. Forte, and D. Sanders, Eds. Oxford: Archaeopress, 2000, pp. 47-52.

[6] *Virtual Reality in Archaeology*, Proc. Computer Applications and Quantitative Methods in Archaeology, Barcelona, March 1998, J. Barceló, M. Forte, and D. Sanders, Eds. Oxford: Archaeopress, 2000.

[7] *Virtual Archaeology:* Proc. VAST Euroconference, Arezzo, November 2000, F. Niccolucci, Ed. Oxford: Archaeopress, 2002.

[8] F. Parke, "Lower cost spatially immersive visualization for human environments," *Landscape and Urban Planning*, vol. 73, no. 2-3, pp. 234-235, 2005.

[9] F. Castro, *The Pepper Wreck: A Portuguese Indiaman at the Mouth of the Tagus River*, College Station: Texas A&M University Press, 2005, p. 19.

[10] Ship Reconstruction Laboratory, Texas A&M University. [Online]. Available: http://nautarch.tamu.edu/shiplab/ [Accessed: May 2008].

[11] F. Castro and N. Fonseca, “Sailing the Pepper Wreck: a proposed methodology for understanding an early 17th century Portuguese Indiaman,” *The International Journal of Nautical Archaeology*, vol. 35, no. 1, pp. 97-103, 2006.

[12] J. Foley, A. van Dam, S. Feiner, and J. Hughes, *Computer Graphics: Principles and Practice*, New York: Addison-Wesley Publishing, 1996.

[13] F. Castro, “Rigging the Pepper Wreck. Part I – masts and yards,” *The International Journal of Nautical Archaeology*, vol. 34, no. 1, p. 120, 2005.

[14] F. Castro, N. Fonseca, A. Wells, “Boarding the Pepper Wreck,” *Historical Archaeology*, submitted.

[15] NASA Jet Propulsion Laboratory, California Institute of Technology. “Shuttle Radar Topography Mission.” [Online]. Available: http://www2.jpl.nasa.gov/srtm [Accessed: May 2008].

[16] R. Horne, Visualization Software LLC. Home of 3DEM Software for Terrain Visualization and Flyby Animation. [Online]. Available: http://www.visualizationsoftware.com/3dem.html [Accessed: May 2008].

[17] Global Land Cover Facility, University of Maryland. [Online]. Available: http://www.landcover.org/index.shtml [Accessed: May 2008].

[18] Planetside Software. Terragen: Photorealistic Scenery Rendering Software. [Online] Available: http://www.planetside.co.uk/terragen/ [Accessed: May 2008].

[19] Museu Nacional de Arte Antiga. *Biombos Namban: Namban Screens*, 4th ed. Lisbon: Instituto Português de Museus, 1993.

[20] Okamoto, Y. *The Namban Art of Japan*, 1st ed. New York: Weatherhill, 1972.

VITA

Name: Audrey Elizabeth Wells

Address: Visualization Laboratory
Texas A&M University
C418 Langford Center
3137 TAMU
College Station, TX 77843-3137

Email Address: wellsae@viz.tamu.edu

Education: B.S., Computer Science, Mount Union College, 2004
M.S., Visualization Sciences, Texas A&M University, 2008