The Aesthetics of Science Fiction Spaceship Design

by

Kate Kinnear

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2010

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

In this thesis, we present a detailed analysis of the conventions that appear in fictional spaceship design, including a discussion of their origins, their uses in emulating certain traits, and reasons these conventions might be followed or ignored. We uncover these conventions by examining and comparing popular spaceship designs from the past sixty years, which we present in a detailed survey.

We also examine an aesthetic interpretation of information theory, which can be used to describe the balance of uniformity amidst variety, and discuss specific strategies for incorporating these principles into the creation of spaceship surface details.

Procedural modeling describes a set of techniques used to allow computers to generate digital content such as 3D digital models automatically. However, procedural modeling to date has focused on very specific areas: natural scenery such as trees and terrain, or cityscapes such as road maps and buildings. While these types of models are important and useful, they focus on a specific subset of the procedural modeling problem. Though procedural generation can be an invaluable tool for providing viable and dynamic content, it is troubling that so few types of objects have been studied in this area. Using the aesthetic and spaceship principles we define, we have developed a prototype system to procedurally generate the surface details of a large scale spaceship. Given a surface representing the frame of a spaceship, we apply geometry automatically in a coherent manner to achieve the appearance of a spaceship by emulating important traits.

Acknowledgements

I would like to express my appreciation to everyone who supported me and assisted me with this work over the past two years. I would like to extend a special thanks to my supervisor, Craig Kaplan, for his support, guidance, and enthusiasm, and for helping me find a project I could be passionate about.

I would like to thank Todd Boyce for contributing his spaceship expertise and insight during the early stages of this work. I would like to thank my readers, Steve Mann and Dan Berry, for their comments on this thesis.

I would also like to thank the members of the Computer Graphics Lab for making my graduate school experience enjoyable and memorable.

On a personal note, I would like to thank my parents and family for their unending support. Finally, I couldn't have done this without Lee to keep me sane and calm. He has always stood by me, and endured the endless discussions about spaceships. I hope to one day know as much about computers as he does.

Dedication

To my dad, for teaching me a love of mathematics, computers, video games, and command line interfaces at an early age. Thank you for your guidance, enthusiasm, and patience.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Fascination with the sky and space as the realm of the gods began centuries ago with classical mythology and scripture [88]. Now, space offers visions of the future; advanced technology, alien races, and traveling to distant planets have captured the imagination of generations. Science fiction, of which spaceships and space travel is a central theme, has a long history in film and television, and more recently in video games.

Science fiction artists that work in a 2D medium, such as painters John Berkey and Vincent Di Fate, are able to create designs that are limited only by the imagination. Because of the fantastic nature of the genre, the introduction of computer graphics has been a great asset. Digital content creation allows greater flexibility, so that science fiction designs can be transformed into 3D models without the limitations of physical restrictions.

With computers and graphics cards becoming increasingly powerful, viewers have come to expect an ever improving level of photorealism from digital media. Where once a designer might simulate detail with a busy texture, it is now plausible to create painstaking levels of detail entirely from geometry. However, this emphasis on detail has led to a bottleneck in the creation of digital content. While computers have undoubtedly facilitated the creation and manipulation of content in these media, the creation of content of the calibre now expected by viewers still requires a comparable number of hours of artists' work.

Procedural techniques allow some of this load to be shifted from the artist to the computer by automating portions of the content creation process. Currently, these techniques are not intended to replace the artist, merely to lessen the workload and free the artist from repetitive tasks.

Procedural generation can be an invaluable tool for providing affordable and dynamic content, but few types of objects have been studied in this area. This research provides a starting point for defining methods for describing futuristic looking, technological objects for procedural modeling, by providing an analysis of spaceships as a case study.

1.1 Problem Definition

This research focuses on examining aesthetic principles as they relate to the beauty and visual interest of spaceships, especially surface details, and determining how these principles can be applied in practice to procedurally model spaceships.

Because of the incredible variety of spaceships, we make a few important distinctions to allow us to focus on a specific subset of spaceships. First, we are not interested in real-life spaceship designs, except for how they have influenced science fiction designs. Second, we are interested in creating spaceships with a high level of visual interest. These spaceship models are meant to be seen close up, and hence will be very detailed. These details will come primarily from geometry.

In certain cases, specifically our analysis of surface details in Section 4.9 and our implementation in Chapter 5, we restrict our focus to examine just mechanical spaceships, as opposed to organic spaceships. Differences between mechanical and organic spaceships are discussed in Section 4.2, and the reasons for this restriction are discussed in the introduction of Chapter 4.

1.2 Research Questions

To procedurally model a class of objects, not only must we first formalize the characteristics that define this class of objects, but we must also determine how these characteristics combine to create an aesthetically pleasing design.

Science fiction models differ from other commonly procedurally generated models because they do not have a built-in sense of plausibility. Objects such as terrain, trees, and buildings all have an innate sense of correctness that arises from familiarity and an adherence to physical constraints. Fictional technology is interesting because it allows for limitless design possibilities, but it also presents a difficult challenge: how does a designer imbue a fictional design with a sense of reality? What artist James Gurney refers to as “imaginative realism” [42], this style differs greatly from the design of a real technological or mechanical device; a real object looks plausible by virtue of the fact that it actually exists, but what *is* mechanical and what *looks* mechanical are not always equivalent. Since these spaceships do not yet exist, they must borrow plausibility from external sources.

While spaceships’ limitless design space allows for much flexibility, it also creates the problem of deciding how to properly constrain this space to create a coherent set of rules. While any random configuration of shapes could potentially be labeled a spaceship, we wish to define a set of rules that describe discernible characteristics of spaceships. We are interested in features of spaceships that allow them to clearly be identified as spaceships.

Spaceships are also an interesting collection of objects because they are purposefully difficult to generalize. Spaceships are often as much characters in film and television as the actors they carry. They are designed to be especially recognizable; that is, they are designed to be easily differentiable from other spaceships so that they stand out to the viewer. Because of this concentration on uniqueness, it is difficult to generalize the common features of spaceships. To circumvent this difficulty, we focus on identifying high level conventions that appear across spaceship designs, as well as conditions for ignoring these conventions.

This thesis attempts to formalize the characteristics and aesthetics of spaceships by examining the following questions:

- What makes spaceships different from other types of objects?

- What conventions are used in spaceship designs and why?
- How does a design convey a sense of plausible, working technology?
- How do we separate form from function in a fictional design?
- What makes a design appear futuristic?
- How can we quantify the attributes that make a spaceship design aesthetically pleasing?

1.3 Important Terms

The following terms are used in this work to describe the components of a spaceship. Also, the terms “spaceship”, “starship”, or simply “ship” are used interchangeably in this thesis and are considered equivalent. However, in the *Star Trek* universe the terms “spaceship” and “starship” have distinct meanings, and are discussed in the survey in Section A.3.

Nurnies: Geometry added to the surface of a model to make it visually interesting. Nurnies are also referred to as greebles.

Panels: Divisions on the surface of the ship. Nurnies are often contained within panels.

Surface Details: A term that encompasses both panels and nurnies.

Functional component: A division of the frame that serves a definitive purpose. This purpose should be visually identifiable.

Frame: The basic geometry of the ship, excluding any surface details.

There may not always be a clear division between different features of the ship; we cannot always determine if a feature is a surface detail or a part of the frame. However, these terms provide us with a good starting point, and we can worry about exceptions as they arise.

1.4 Contributions

This research makes the following contributions to the area of procedural modeling.

- **Aesthetics and spaceships.** By examining aesthetic principles and their relevance to spaceship design, we provide a method for discussing and analyzing the appeal of a given spaceship, focusing on the arrangement of surface details.
- **Spaceship analysis.** We formalize the characteristics that are important to spaceship design and define how these characteristics can be emulated. These characteristics include the type of design (mechanical or organic), symmetry, scale, direction, expectations, technology, personality, and plausibility.

- **Spaceship Survey.** We present a detailed survey of popular and influential spaceship designs, in which we analyze the factors that influence their design with respect to the characteristics defined in the spaceship analysis.
- **Surface detail prototype.** Given the aesthetic principles identified, we determine which procedural modeling techniques are best suited to generating spaceship surface details. We implement a prototype system to demonstrate the effectiveness of these techniques.

1.5 Motivation

The most natural starting point for motivating this problem is to examine how spaceships were designed and created before the use of computers and how they are designed as computer models. This section examines both of these methods to demonstrate how computers have affected the creation of spaceships.

1.5.1 Pre-Computer Spaceship Design

Before computers, designers had no way to create a 3D virtual representation of a spaceship. Instead, they crafted an actual physical model. Concept art for the spaceship would be created by an artist or designer, and this sketch would then be given to a team of model builders, who would be responsible for crafting the final model. Models were usually built from a combination of wood, metal, and plastic.

There are a few obvious disadvantages to creating physical models. This process is time consuming, as well as expensive, as materials and labour are needed to build the model. Various versions of the same model are often built for different purposes; models are needed in different scales depending on their proximity to the camera and other models in the scene, and specialized models may be needed for special effects or pyrotechnics.

Revisions to the design after construction begins are costly at best and impossible at worst. While surface detail alterations may be as simple as removing one piece and replacing it with another, any changes to the frame of the spaceship could require starting from scratch.

In some cases, the design of the model may be shaped by necessity. For example, the design of the *Star Wars* Millennium Falcon required that heat vents be added to the body to prevent the interior engine lights from melting the model [72]. While these vents, shown in Figure 1.1, are an interesting part of the final ship and have been incorporated well into its design, the fact still remains that the designers were forced to revise their original vision for the ship to deal with physical restrictions. Other physical restrictions include ensuring that the model is light enough to suspend or carry, ensuring that the model's colors are suitable for compositing techniques, and ensuring that the model's materials are capable of producing desired special effects, such as pyrotechnics.

Despite these problems, there are a few distinct advantages to building physical models. Though the construction is a long and tedious process, it is also *known* to be a long process, allowing the construction of the model to be a part of a continuous creative process that involves

Figure 1.1: The *Star Wars* Millennium Falcon. Note the six circular heat vents on the top rear of the ship, which were added to the design to prevent the engine lights from melting the model.

the artistic staff [72]. The construction of each individual piece of a model can be overseen by designers to ensure that the result is faithful to their original vision. Ships are designed with interconnectedness strongly in mind, to appear as if the pieces together form a plausible, working whole. Surface details are placed strategically to imply the scale and direction of the ship [72]. Working directly with physical materials also required craftsmanship and ingenuity, as model builders often had to make due with the materials available to them.

Physical model builders come from diverse backgrounds, bringing a variety of knowledge and creativity to their craft. Physical model building encompasses a variety of tasks, and draws in professionals from a variety of disciplines, including industrial design, fine arts, engineering, architecture, construction, chemistry, electrical engineering, audio engineering, photography, theater design, costume design, makeup, shipbuilding, illustration, and puppetry. Modelers were trained to perform as many tasks as possible, resulting in a team where each individual has a strong set of heterogeneous skills [72]. This diversity results in a greater wealth of artistic and technical experience.

The team dynamics of model building were also different. Physical model builders for the *Star Wars* films always worked in teams, ensuring that different areas of the model were worked on in rotation to prevent any section from being dominated by one person's style [72]. While it was usually impossible for a single person to create a physical model, computers can allow fewer people to work more independently, and prevent multiple people from working on the same model at the same time.

Physical model builders had another excellent tool at their disposal: kitbashing. Kitbashing is a common modeling technique that involves constructing a model by scavenging parts from commercial model kits [72]. As shown in Figure 1.2, details on the *Star Wars* Millennium Falcon were pieced together from various tank models. Kitbashing allows designers to start with a library of interesting mechanical shapes at their disposal. Where designs are unclear or filler is needed to add visual interest to sections of the ship, it is easy to mix and match pieces from the kitbashing library to meet those needs. Kitbashing is also an important technique for brainstorming and finding inspiration.

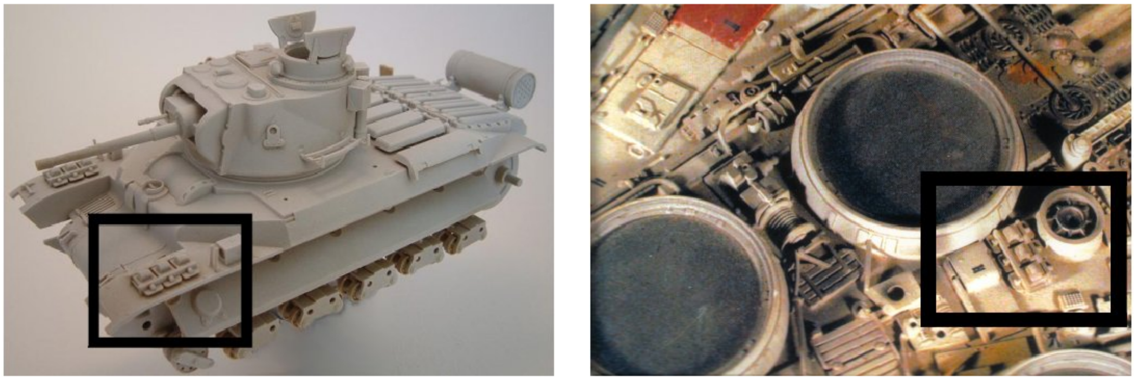


Figure 1.2: Pieces from a Tamiya Matilda tank kit (left) are used to detail the *Star Wars* Millennium Falcon (right).

While being restricted by physical limitations was previously mentioned as a disadvantage of this method, it is also a distinct advantage for ensuring the plausibility of the final model. Physical models were preferred in the production of the *Star Wars* prequels, even though computer models were available, because the designers were afraid that these digital models, having no restrictions, might be “too removed from the physical boundaries of reality” [72]. As these spaceships belong to the realm of science fiction, creating a sense of plausibility is one of the most important design aspects. Since plausibility is a hard thing to define, never mind enforce, and computer models are limited only by imagination (and processing power), it is difficult to ensure that the resulting model will look like it could have been built. Physical models, having actually been built from physical materials, are granted this degree of plausibility by default.

1.5.2 Computer-based Spaceship Design

As computers became faster, cheaper, and easier to use through the 1990s, computer generated imagery soon became the logical choice for film and television special effects. Computer models offer a wide variety of advantages over traditional physical models. The only tool needed is a computer, eliminating the costs of materials and specialty equipment for cutting or shaping these materials. Creation of a computer model is faster, and changes can easily be made. Only a single version of the model is required, as use of the model for any sort of special effects is not destructive to the original model. Models can also easily be scaled as needed for the scene.

However, these advantages did not come without a cost. Where a sketch was once given to a team of model builders, it is now given to a 3D modeler (or possibly a team of modelers, depending on the scale of the project). For physical models, a design was given to a creative team who spent a lot of time and effort building a physical model, during which time the original artist could easily see the evolution and creation of the concept sketch. However, the creation of a computer model is expected to take much less time, so the modeler can rely on computer techniques to quickly fake detail.

An artist’s sketch is not a CAD drawing. It does not describe specific measurements, nor does it contain every detail that should be modelled. It would be impractical, not to mention difficult, for the artist to draw the concept of a model in a large enough scale to make every individual detail clear to the modeling team. Even if this were possible, often the artist does not know exactly what details should be included in the final model for all areas of the design. Artists use color and texture to suggest detail, but a 3D modeler has no such abstractions and must make concrete decisions about details.

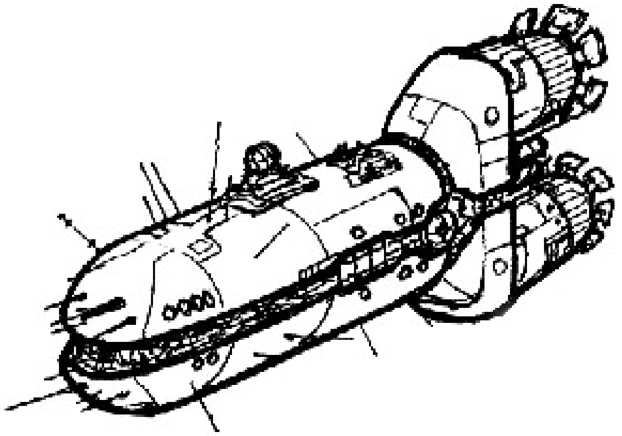


Figure 1.3: An example concept sketch of a spaceship, courtesy of Todd Boyce [18]. Note that few specific details are drawn, though the artist conveys where detail is necessary.

Unfortunately, the 3D modeler is not invested in filling in these details. Computer modeling techniques are tailored to operations that computers excel at: repetition and randomness. These undefined details can easily be filled with simple random polygons or busy textures that create the appearance of complexity and mask the lack of interest. Even assuming that the modeler has the artistic ingenuity to properly add in these details, time is a constraint [18]. Physical model builders did not suffer from these difficulties for a few reasons. First, while time was clearly still a constraint for them, physical modelers did not have short cuts that could be used to create the appearance of detail. Where a computer modeler can cut and paste geometry to fill in gaps, physical modelers must still add each individual detail by hand. Second, physical modelers had a team of people alternatively working on different sections of the model. This dynamic team setting divided the responsibility, so that no single person was pressured to finish a model alone. Finally, physical modelers did not have to start from scratch, as kitbashed parts provided easy filler.

Though we have focused on traditional kitbashing, virtual kitbashing is not unheard of in 3D modeling. However, it is not as natural as traditional kitbashing, as computer models are not usually available as a set of parts, but a single whole. Parts may be scrounged from existing models, but require the extra step of identifying appropriate pieces and separating them from the model. Copyright issues also prevent modelers from taking full advantage of this technique.

As of yet, no software tool approaches the expressivity of working with physical materials. 3D modeling packages attempt to provide convenient tools, but these tools stem from functions that are traditionally supported by computers, and they are not as natural as shaping a model from physical materials. A physical modeler does not need to determine how to correctly align a shape on the model's surface or create natural-looking weathering, as these properties arise naturally from the medium.

While the introduction of computer modeling has allowed the film and television industry to create things that would be impossible to achieve with previous modeling methods, this transition has also taken away from the creative process. We believe that computers, specifically procedural modeling, can provide us with techniques to enhance the abilities of artists and model builders in the creation of spaceships. We wish to recapture some of the benefits of physical modeling practices.

1.5.3 A Comparison of Physical and Computer Spaceship Models

The television show *Battlestar Galactica* provides us with an excellent opportunity for comparison between physical and computer models of spaceships. *Battlestar Galactica* aired originally in 1978, before computers were used for effects in television. The spaceships used in this version of the show were physical models. However, *Battlestar Galactica* was recently remade in 2004, and this new version uses computer models for the featured spaceships. Consider Figures 1.4 and 1.5, showing the 1978 and the 2004 versions of the titular spaceship.

Figure 1.4: An image of the titular spaceship from the 1978 *Battlestar Galactica* television show. This spaceship is a physical model.

Figure 1.5: An image of the titular spaceship from the 2004 *Battlestar Galactica* television show. This spaceship is a computer model.

These two ships exhibit distinct styles. The 1974 Battlestar Galactica is obviously kitbashed. It features intricate mechanical details and a high level of visual interest. The details are essentially symmetric; all major surface details are symmetric, though smaller nurnies may differ. The ship has a strong mechanical appearance, in keeping with the *Star Wars* style of ship. The panels are complex shapes.

The 2004 Battlestar Galactica gives an entirely different impression. It is sleek; its panels and nurnies are simple and repetitive. There are no small mechanical parts visible on the surface of the ship, giving it a futuristic feel.

While a given person may have a personal preference for one ship over the other, these designs are meant to evoke different feelings from the viewer. It is not possible to say that one is definitively better than the other, as this depends on subjective criteria. However, it is easier to see what features of the 1978 Battlestar Galactica design make it look like a spaceship. The surface of the 1978 Battlestar Galactica is pieced together from small parts that are inherently mechanical themselves. These parts are laid out strategically to make the ship as a whole appear feasible, but even a haphazard arrangement of mechanical parts can achieve the appearance, though not as effectively, of a spaceship. Essentially, it is easier to make an arbitrary shape or frame look like a spaceship by using the kitbashing method to add surface details that are inherently mechanical in appearance. *Star Trek*'s geometrically simple Borg ships, discussed in Section A.3.7, are the canonical example of how a kitbashed style can give a random shape the appearance of a spaceship.

The 2004 Battlestar Galactica, while obviously a spaceship, does not have any specific defining surface details that can be identified as making it look like a spaceship. The effect as a whole is still present, but it is unclear what combination of features are contributing to this. Because the goal of this research is to take a frame and have the computer automatically apply geometry to make it look like a spaceship, the kitbashing method is valuable. While it does not guarantee a cohesive design, it does facilitate the desired mechanical appearance.

While computers have greatly facilitated the creation of spaceship models, some of the style and creativity of physical model building have been lost during the streamlining of this process. As time becomes more of a constraint, computer-based techniques leverage the computer's aptitude for repetitive tasks to quickly add detail. While physical modeling techniques such as kitbashing exist to ease the addition of interesting detail, there is a clear opportunity to improve computer-based techniques to recapture the practices used by physical model builders.

1.6 Organization

The remainder of this thesis is organized as follows: Chapter 2 provides an introduction to common procedural modeling techniques, and surveys prior research in procedural modeling of relevance to this thesis.

Chapter 3 focuses on the aesthetics of spaceship surface details. It first provides an introduction to the relevant aesthetic principles, and then describes the application of these principles to spaceship surface details.

Chapter 4 presents specifics of spaceship design conventions, and discusses general conventions that govern spaceship design, as well as characteristics of the spaceship's frame and surface details individually.

Chapter 5 describes the implementation details of the prototype program for generating spaceship surface details.

Finally, Chapter 6 concludes this thesis with a discussion of results, possible directions for future work, and some final conclusions.

Chapter 2

Background

Procedural techniques are algorithms that describe the behavior or characteristics of computer-generated content [34]. Procedural modeling refers specifically to procedural techniques that are used to generate 3D computer models. These techniques are useful for a variety of reasons. First, they allow for condensed storage; the details of every individual model do not have to be specified explicitly, as each one can be generated as needed from the procedural algorithm. Second, since each model does not need to be specified individually, procedurally generated models require much less effort from model builders. Procedural techniques employ either parametrization, randomness, or both to create diverse content from a single algorithm. While it is true that there will be some initial overhead to create the procedural algorithm, once completed, this algorithm can be used to create a near infinite amount of unique content [34]. Common techniques in procedural modeling include fractals, L-Systems, and shape grammars.

Fractals are used to create visual complexity in many computer graphics applications, both for textures and geometry. The complexity of a fractal arises from self-similarity; that is, the geometry of a fractal appears the same at different scales. Fractal shapes are common in nature, appearing in snowflakes, plants, mountains, and other natural formations, and can be used to describe shapes that are difficult to define with traditional geometry. For example, a tree has a self-similar structure, in that a branch of the tree has the same structure as the entire tree. This self-similarity means that complexity can be generated by taking a given base shape and repeating it over and over at a variety of scales [34]. Though fractals are a simple way to add visual complexity, technology generally does not exhibit this property of self-similarity, making fractals ill suited to spaceship generation.

Lindenmayer Systems, commonly known as L-Systems, were first introduced in 1968 by Lindenmayer for the study of plant development, and were later adapted for use in computer graphics [76]. L-Systems are a type of formal grammar that define string rewriting rules that are applied in parallel to simulate natural growth. L-Systems are primarily used for the generation of plants and trees, as the growth of the string closely relates to this type of biological development [34]. Though well suited to natural processes, L-Systems do not lend themselves to the structured organization of spaceships.

A *shape grammar* is a set of production rules used to generate geometric shapes. Production rules define how an existing shape can be transformed or replaced by a new shape (or a set of new shapes). Rules may also be parametrized to provide more control over shape production. Shape grammars were first introduced by Stiny in 1972 [90]. A restricted type of shape grammar, a *split grammar* allows only substitution rules in which the volume of the new set of shapes is completely contained within the volume of the original shape. Split grammars were first introduced by Wonka et al. in their 2003 paper [97]. Shape and split grammars are commonly used in procedural generation of architecture, as they are suited to the regularity of buildings. Shape grammars are interesting because they allow for the creation of a set of related shapes that, though different, are bound by the same structural laws. We discuss the application of shape grammars to spaceship generation in Chapter 5.

2.1 Previous Work

The vast majority of work done to date in procedural modeling focuses on three subjects: terrain, vegetation (especially trees), and cityscapes (especially buildings and road plans). While these types of models are common, and hence their procedural generation would provide much benefit to the artist, it is troubling that virtually no other types of models have been researched in procedural modeling.

Terrain generation is usually achieved via a height map. A height map is a 2D image that stores elevation data; these variations in elevation are what constitute the terrain. Height maps can be generated using a variety of techniques, such as midpoint displacement algorithms or Perlin noise [87].

While a variety of models have been created to describe the behavior and structure of plants, we are interested in methods that are used to create representations of plants for computer graphics. Some of the earliest 3D procedural plants were based on fractals and generated with recursive functions, such as Oppenheimer's fractal trees [70]. Structured particle systems have also been used to generate trees and plants. In this type of system, the particles are not independent, but are connected to form a 3D object [79].

The most popular technique for the procedural generation of plants is L-Systems. L-Systems have progressed greatly since their introduction, and now include parametric L-Systems for modifying the characteristics of the resulting model, context-sensitive L-Systems for representing interactions between different parts of the plant, differential L-Systems for plant growth animations, environment-sensitive L-Systems for adaptation to the surrounding environment, and open L-Systems for modeling the exchange of information between the plant and the environment [29, 76].

Other interesting directions in procedural modeling of plants include procedural generation of topiary with environment-sensitive L-Systems [75], image-based generation of plant models [77], generation of trees at different levels of detail while preserving their overall structure [62], and automatic distribution of plants based on ecosystem [44].

Plant generation software is popular for a variety of applications, and both commercial and freeware options exist. A brief comparison of various plant generation software can be found at the *Virtual Terrain Project* website [31].

For a detailed reference of computer generated vegetation techniques, see Deussen's *Digital Design of Nature: Computer Generated Plants and Organics* [29].

Another popular model for procedural generation is cityscapes. Procedural cityscapes can be broken down into two main categories: buildings and road plan generation. Recent trends in road plan generation focus on increasing user control of results over complete automation. Techniques have been proposed using templates of common road layouts [91], L-Systems that grow a network of roads based on geographical and sociostatistical image maps [71], a combination of Voronoi diagrams and L-systems [40], an agent-based approach that models urban development based on zoning [58], an interactive visual interface that applies L-Systems to user-defined nodes [52], and, most recently, a visual interface that uses tensor fields to guide road generation and allows intuitive editing of the final layout [20].

Current procedural building generation is achieved almost exclusively with some type of shape grammar, though L-Systems were used in an early paper to generate simple office buildings [71]. Shape grammars were first used for procedural architecture in computer graphics in Wonka et al.'s 2003 work [97]. In a 2006 follow-up paper, the authors introduce the shape grammar *CGA Shape*, which uses volumetric models to allow for greater variety in building shape and context-sensitive shape rules to create consistent façades [68]. This grammar was used to successfully create a wide variety of visually appealing results.

Other directions of interest have also been studied in this area. Recent work has been done to automatically generate shape grammar rules from an image of a building façade [69]. Determining shape grammar rules, as opposed to just generating a static 3D model, is useful because it allows the user to create new buildings in the style of the image building. While previous work has focused solely on visual appeal, a 2009 paper introduces the addition of structural feasibility in building generation by analysing forces to determine a stable configuration of parameters that adheres to the user's production rules [95]. A 2008 paper by Lipp et al. introduces a visual editing paradigm for the existing shape grammar *CGA Shape*, allowing the entire set of grammar rules to be created without text editing [61]. The visual editor also introduces a new method to allow local modifications of buildings without the complexity that usually results from these types of modifications.

For a survey paper related to current work in procedural modeling in these areas, a 2009 paper by Smelik et al. [87] provides a brief overview of current procedural techniques used to generate terrain, vegetation, road networks, and cityscapes.

Objects such as plants and architecture are fundamentally different from spaceships in a few important ways. First, there are explicit rules that already existed to describe the behavior and appearance of these objects, prior to their examination for procedural modeling. Plants are governed by strict biological processes. Buildings must adhere to certain structural restrictions, and most follow a well-defined set of architectural standards. Conversely, fictional spaceships are not technically bound by any laws, though, as we demonstrate in this research, convention can be used to determine certain standards. Second, the rules for these objects encompass a large percentage of different instances. In other words, these objects are fairly uniform and have consistent similarities. The design of spaceships is much less limited, allowing for a greater diversity of design.

Despite their obvious uses for games, television, and film, spaceships represent a specific type of model. However, there has been a recent trend in this area towards the procedural modeling of highly specialized types of models. This is discussed further in Section 2.1.1.

While procedural techniques are often used to create new models, they can also be used to add visual complexity to an existing model. Visual complexity can be added to computer models in two ways: textures and geometry. Traditionally, texture mapping has been used in computer graphics as a fast, cheap method to add visual complexity to a 3D model. Texture mapping applies a pattern of color to the surface of a model to create detail without the addition of extra polygons.

More recently, procedural texture generation has been developed to allow unique textures to be generated on the fly, without explicit storage of a texture map. Many procedural texture generation techniques rely heavily on well-behaved noise and turbulence functions to simulate randomness found in nature, but other approaches exist. Worley textures use a Voronoi diagram to create a cellular texture commonly seen with stones and scales. Texture synthesis uses a small sample image to generate a large, non-repetitive texture [34]. Procedural texture generation is becoming more and more sophisticated with the increasing generality supported by graphics processing unit (GPU) programming models.

The addition of busy textures is the easiest way to add visual complexity to a model, but this approach is not appropriate for all applications, especially those such as film and television where the models are meant to be examined closely by the viewer and require a high level of detail. While texture synthesis originated from the desire to add detail in an economical manner, the increased speed and storage capacity of current computers has made this less of a concern. Texture mapping is also not perfect, and can suffer from artifacts.

Combining both textures and geometry, Zhou et al. describe a geometric texture synthesis method that synthesizes geometry on the surface of a model from a swatch [99]. The swatch is repeated and stitched together seamlessly. While this paper represents a trend in returning to geometry to add detail to a model, as opposed to textures, this method is not directly applicable to spaceships. As we discuss in Section 4.9, spaceship surface details require a strategic arrangement derived from both semantics and aesthetics.

Geometry can also be applied to an arbitrary model through the use of plug-ins available for most major 3D modeling software packages. However, these plug-ins are simplistic, often using only a single type of geometric primitive, and do not take into account any semantic information about the model, such as size and direction, making these methods ineffective for spaceship generation. These plug-ins are discussed in more detail in Section 2.1.2.

Finally, this thesis emphasizes the influence of perception and aesthetics in spaceship design. By tailoring the selection and placement of geometry to leverage these principles, we can better capture the essence of spaceship aesthetics and design, and determine what makes an interesting spaceship. Human perception in the field of computer graphics has experienced a recent resurgence in interest, and is discussed further in Section 2.1.3.

2.1.1 Specialized Procedural Modeling

A recent trend in procedural modeling focuses on modeling a specific type of constructed object; this implies procedural modeling techniques must be tailored for specific types of objects. While the forms of some objects, such as plants, are easy to generalize, to procedurally model other types of objects, they must be studied individually to formalize their characteristics.

Specialized procedural modeling gained attention with the 2008 release of *Spore*, an open-ended simulation game developed by Maxis. *Spore* consists of extensive procedurally generated content, most notably its Creature Creator [22]. However, this type of procedural generation is strictly parametrized and does not incorporate any randomness. In the case of the Creature Creator, the player is given what is essentially a toolbox of creature parts, consisting of various sensory organs and limbs, that can be added and modified in certain configurations. This type of modular procedural modeling defines a set of possible creatures, limited by the size of the given toolbox, and the player is able to select a specific creature from this set.

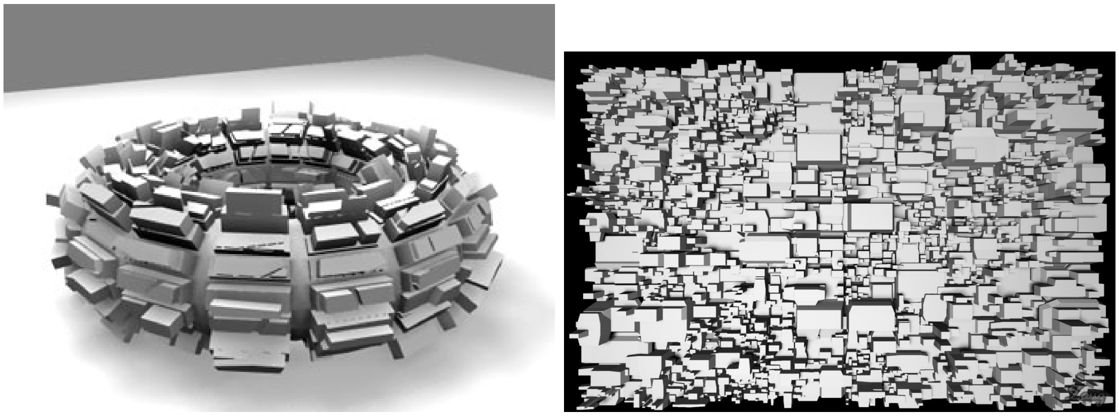
Recent papers in specialized procedural modeling include Teoh's 2009 paper, which describes methods for procedurally generating ancient East Asian buildings, including traditional Chinese, Japanese, and Korean architecture [92]. The author formalizes the characteristics of these buildings, including the similarities and differences between the various types. Modification of parameters allow the user to specify either an exact building, or just a type of building, so that other parameters can be randomized to produce a variety of buildings within the same family [92]. Hart's 2009 work describes a cell-division algorithm, similar to L-Systems, that procedurally grows natural forms to model underwater sea life, such as coral [45]. Other examples of specialized procedural modeling include medieval castles, ocean shorelines, and cracks and fractures [25, 56, 64].

2.1.2 Greeble Plug-ins

Plug-ins to automatically cover a model in greebles exist for most major 3D modeling software packages. Greeble plug-ins are available for 3D Studio Max (Greeble Plugin), Blender (Discombobulator), Realsoft3D (Real GreebleZ!), trueSpace (Greeble maker), and Lightwave (Greeble script) [48, 83, 2, 6, 54].

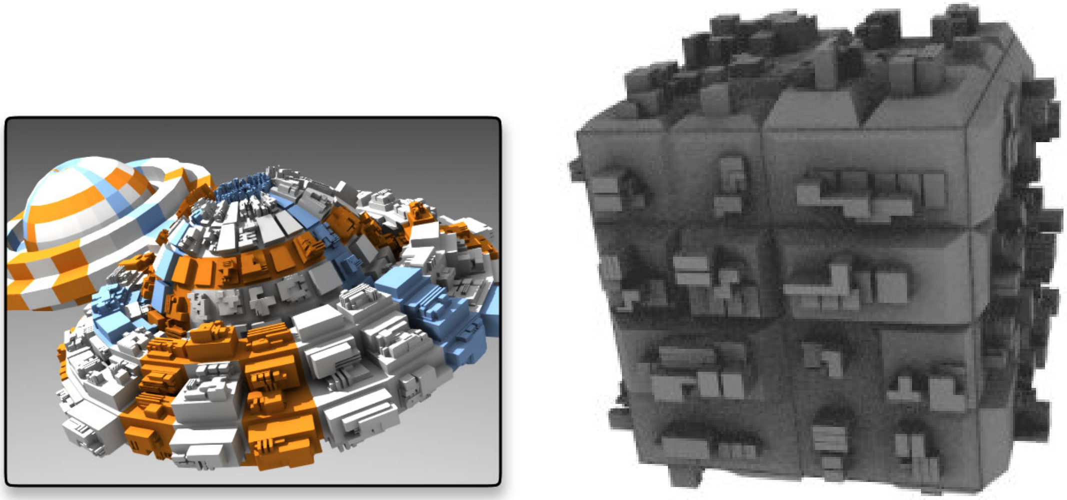
Though most of these plug-ins do not give details of their algorithm, it is clear from studying their output that they follow a fairly standard formula for adding detail to a model. All of these plug-ins add only cuboids (rectangular prisms) and combinations of cuboids for surface details. Some, such as Blender's Discombobulator, allow the cuboids to be tapered. Modifications are made on a per-face basis, so the results are entirely dependent on the triangles and quadrilaterals that make up the original model. Some of these plug-ins, such as Blender's Discombobulator, handle only quadrilateral faces. Faces from the model are selected based on some criteria, which may be based on shape (triangle or quadrilateral), or simply a random selection. Selected faces are then either broken down randomly or, in the case of trueSpace's Greeble maker, by the user. Cuboids are then added to the surface in a random configuration or chosen from a set of predefined (though parametrizable) configurations. This process is sometimes repeated recursively. Example output from each plug-in is shown in Figure 2.1.

While these plug-ins are an easy and efficient way to add detail to a model, they are useful only in certain instances. They excel at adding detail and complexity to objects that are not expected to be closely scrutinized by the viewer. However, on a model that is expected to be closely inspected, it is obvious that the details consist of a random, meaningless configuration of cuboids. Another weakness of this method is its lack of connectivity; faces are processed individually with no regard to the faces around them, which creates an obvious lack of continuity. This is especially evident for curved surfaces, such as the processed torus in Figure 2.1(a).



(a) A torus processed with trueSpace's greeble plug-in [6].

(b) Example output from Realsoft3D's Real Greeblez plug-in [2].



(c) A saucer model processed with Lightwave's greeble plug-in [54].

(d) A cube processed with the Blender Discombobulator plug-in [83].

(e) An image created using 3D Studio Max's Greebles plug-in [48].

Figure 2.1: Example images created from various greeble plug-ins.

2.1.3 Graphics and Perception

While computer graphics and perception have always gone hand-in-hand, researchers in computer graphics, especially non-photorealistic rendering (NPR), are becoming increasingly interested in the role of human perception. Since photorealistic rendering has now reached an astounding level of realism, the problem is no longer whether the computer can render a believable looking scene. Instead, the focus must turn to the viewer's interpretation of the scene; is the viewer being presented with the desired information so that he or she can effectively interpret what is relevant? Conveying the meaning of a scene is often more important than conveying it realistically [65]. Aesthetics is also a concern; in many cases, what is pleasing to the viewer is not always what is physically accurate.

Much work has been done recently in the field of NPR to analyze the general principles of visual perception to determine what features of an object facilitate understanding and recognition, so that these features can be altered or emphasized to ensure that the intent of the scene is clear. Much of this research focuses on automatically identifying lines and contours that are important for depicting shape [27, 26, 35]. Cole et al. recently studied the placement of lines and their effectiveness in depicting shape for both computer-generated and hand-drawn line drawings [24, 23].

By understanding how we can manipulate these visual cues, we can better convey the intended information. NPR provides more freedom to manipulate these cues, as styles of lines and colors for the same scene can greatly vary if photorealism is not the goal. However, procedural modeling allows these principles to be integrated automatically into a photorealistic scene, through the manipulation of the layout, orientation, and geometry of the objects themselves, instead of their rendering style.

In this work, we analyse these principles of perception as they relate to both aesthetic perception in general and the aesthetics of technology in specific. For more information on the role of perception in computer graphics, we refer the reader to Bartz's 2008 survey paper [14]. This survey discusses the state of the art in perception trends that are applied to rendering, animation, virtual reality, and visualization.

Chapter 3

Aesthetics

> Delight lies somewhere between
> boredom and confusion.
>
> E.H. Gombrich

While we are interested in the characteristics that define spaceships, the goal of this research is to generate pleasing or interesting spaceships. But how do we define the characteristics of an interesting spaceship? More generally, what factors determine whether an individual finds a given design aesthetically pleasing? This is a fundamental question in the analysis of art, and one that is difficult to answer, especially in an objective way.

Spaceships, as a form of art, are special; there are no clearly defined laws that govern their design. We first need to step back and examine more general questions about man-made art, specifically "What makes a design interesting?"

Obviously, an individual's perception of beauty is subjective. There are no precise indicators for determining if something is beautiful. Similarly, there is no one object or person that can be definitively defined as being beautiful to everyone. The perception of beauty is influenced by an individual's expectations, history, and knowledge.

While beauty is important in determining whether or not we find a design pleasing, there are other factors that influence our overall feeling towards an object. Though an individual may find a certain color beautiful, a painting consisting of a solid block of this color, while beautiful, would be boring. Conversely, a painting consisting of random blocks of color, which may be beautiful individually, also does not convey any meaningful information to the viewer. Creating an interesting painting lies somewhere between these two extremes of order and chaos. Eighteenth century philosopher Francis Hutcheson called this phenomenon "Uniformity amidst Variety" [49]. Art historian E.H. Gombrich called it "graded complication" [41]. Computer scientists call it information theory.

We first introduce the principle of uniformity amidst variety, and then show how aesthetic interpretations of information theory support this principle.

3.1 Uniformity amidst Variety

The goal of effective and interesting art is to convey as much meaningful information to the viewer as possible, without confusing or overwhelming them. Uniformity amidst variety describes a harmony of order and complexity; complexity in a design provides the viewer with new areas of interest to discover, while order allows the viewer to perceptually organize the space. If these two features are not in balance, the design suffers. An excess of order is too easily understood and becomes boring. An excess of complexity overwhelms the viewer and becomes confusing.

Information theory provides a method of describing the amount of information being conveyed to the viewer, as well as a metaphor for illustrating the characteristics of human perception. We first review the basic concepts of information theory so that we can later make use of this terminology to discuss aesthetics.

Information theory describes a measurement of the amount of information contained in a message. Shannon's classical definition of information theory considers a simple transmission: a transmitter sends a message, encoded as a signal, to a receiver over some channel. The simplest way to define a message is as a sequence of symbols chosen from a set of symbols known to both the transmitter and the receiver [85]. The message can take many forms; for our purposes, if the message is an image, it would consist of a sequence of color values representing the pixels of the image. The order of the symbols is important to the meaning of the message, as pixels are arranged in a certain way to create an image [67].

It is important to recognize the distinction between *data* and *information.* A message may contain a lot of data, but little information. For example, consider a 400 x 400 pixel image. This image consists of a large quantity of data, 160000 pixels worth. However, if every pixel in the image is the same color, the image contains almost no information. Hence, we cannot determine the quantity of information in a message based solely on its size [67].

In essence, information theory determines the quantity of information in a message based on uncertainty. Messages that contain information already known to the receiver, or information that can easily be predicted by the receiver, contain little information. The value of the message, the amount of information it contains, is based on the *originality* of the message [67]. A message is most useful when the receiver did not previously have the contained information.

The ultimate goal is to send a message with the highest possible density of information by making the best use of our channel. Shannon's work described how we can determine the maximum possible level of data communication for a given channel, depending on the level of entropy or uncertainty of the information and the noise level of the channel [85]. To reach this maximum, it may be necessary to encode the message to increase its information density. Encoding techniques are useful only when both the transmitter and receiver are informed; the receiver must have knowledge of the encoding method used so that the message can be properly decoded. Encoding techniques are often adapted to make use of the properties of a specific channel [67]. An example of a encoding technique is text messaging shorthand. As text messaging on a mobile device allows for limited data entry, it is in the transmitter's best interests to compress the essentials of the message as much as possible to increase the speed of communication. This has led to the use of certain shorthand conventions, such as spelling words with number equivalents (e.g., replacing "one" with "1", whereby "anyone" is shortened to "any1").

While we can make use of these techniques to transmit our message in the densest manner possible, this is not always practical. While a perfect channel would convey the transmitter's exact message to the receiver, channels often have some level of noise. Logically, noise is distinguished from the signal by intent; noise is any unintended alterations, deletions, or additions to the transmitted signal [67].

Since the original goal was to increase the density or uncertainty of the information in the message, this implies that a high percentage of the signal is crucial. Since the message is designed to be highly unpredictable by the receiver, the introduction of noise compromises the receiver's ability to reconstruct parts of the signal that may be lost or altered. The transmitter attempts to counter the effects of noise by introducing redundancies into the signal. Redundancy increases the robustness of the message by providing the receiver with extra information from which the intended message can be restored. While the intentional addition of redundancy may seem contrary to the original goal, it is crucial to conveying the intended message with accuracy. The amount of redundancy required to reliably transmit the message will depend on the level of noise in the channel [85].

Shannon's classical definition of information theory provides us with an appropriate metaphor for human perception, which we use to examine uniformity amidst variety in greater detail in the following section. However, the information content of a single message is better explained using Kolmogorov complexity. Shannon information theory relies on knowledge of the probability distribution of all possible messages; the information content of a given message is defined by this distribution. Kolmogorov complexity can describe the information content of a single message, without knowledge of the entire set of messages, making it much more practical for our application. The Kolmogorov complexity of a message is the length of its shortest possible representation, given a previously defined description method [60]. A message with a low Kolmogorov complexity has a high level of repetition and a low information content. A message with a high Kolmogorov complexity has a high level of randomness and a high information content.

When we describe the information content of a single design, we are referring to its Kolmogorov complexity. However, we still refer to Shannon's information theory to discuss perception. We do not make precise measurements of information content, but we can speak about the relative amounts of information when comparing messages.

3.1.1 Information theory and perception

To apply information theory to perception, the dynamics of our original transmission must change slightly. Our message is now an artistic design, which may consist of a 3D object, such as a sculpture or model, a 2D object, such as an image or painting, or a 1D sequence, such as a musical composition. For the purpose of this thesis, we will consider the message as a 3D model. However, since a 3D model is viewed in 2D on a computer screen, the two are comparable. The transmitter can be considered the artist, which in our case is the computer that is generating the model. The final modification is the most important: the receiver is now assumed to be an arbitrary person [67]. This change has certain ramifications on the quantity of information that can be transmitted.

Where we were originally limited by the amount of noise introduced in the channel, we are now limited by the capacity of the receiver. The human mind can only process so much information at an instance; once a certain threshold is reached, the mind becomes overwhelmed by the flood of information and it becomes meaningless noise [67]. This limit is perceptible by considering reading and text scanning. While we are capable of viewing an entire page of a book at once, our focus is limited. To read and comprehend the page, we must scan through it progressively. Note how when the eye focuses on a given word or sentence, comprehension of the surrounding text is limited. While the words around the central focus are visible, they become meaningless and fade into the background [67].

Because of this limitation in the receiver's ability to process information, redundancies must again be introduced into the signal. However, the levels of redundancy and uncertainty that are pleasant to a given person will vary within certain limits around an average. Since different people have different tolerances, some are better able to understand complexities and thus prefer slightly more uncertainty, while others respond better to simplicity and thus prefer more redundancy [49]. The goal of effective art is to achieve the level of information content that appeals to the greatest number of people [15].

This limitation of human perception also requires that encoding techniques be chosen with care. As previously stated, encoding techniques can be used to increase the information content of a message; however, they rely on the assumption that the transmitter and receiver share some common mechanism for encoding and decoding the message. An encoded message, no matter how high its information content, is useless to the receiver if it cannot be decoded to an intelligible format [67].

The question remains, what constitutes an encoding technique with respect to art? E.H. Gombrich describes a phenomenon he calls “perceptual generalizations” [41]. A perceptual generalization is an abstraction of an object; it is a representation that is not meant to distract the viewer or attract attention [41]. An example of a perceptual generalization would be when an artist draws an open book in a scene, but fills in the text of the book with squiggles instead of words, as seen in the book depicted in Figure 3.1. While this may be used to represent that the text is simply out of focus or not visible in the scene, often its purpose is to add realism without distracting the viewer. Instead of filling the pages with actual text, which is not important to the message, a simpler representation is used to convey the intention of text. Cartoonists often make use of such perceptual generalizations, which are referred to as “indications” [96]. A cartoonist may draw a brick wall by filling in only small clusters of bricks instead of detailing every individual stone. The goal is to inform the viewer that this is a brick wall without distracting from the focus of the scene. Perceptual generalizations allow detail to be added to a message (in this case, an image) in a condensed form, acting as a type of encoding to increase the information content.

Perceptual generalizations can be thought of as a special case of symbolism. A symbol is a representation of an idea or object. Symbols are useful for conveying emotions or ideas through associations, as well as for representing objects in a more abstract form. Symbolism in general is a useful encoding technique, though symbols are usually culturally dependent and require some assumptions about the viewer's background.

3.1.2 Limitations

While information theory provides a starting point in the effort to define a quantitative measure of art, there are limits to the measurements it offers. Even in the best case scenario, informa-

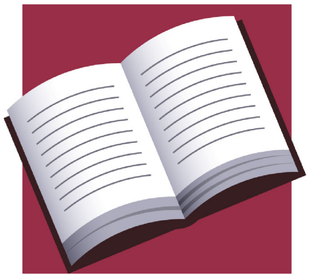


Figure 3.1: An example of a perceptual generalization of an open book. Note that the text is represented as simple lines, instead of actual words.

tion theory can only hope to provide indicators for the innate, universal aesthetics of art. Any preferences resulting from learned or cultural tendencies would vary greatly between individuals. As stated above, even universal principles must be tuned slightly to the individual; as Berlyne states, "people's evaluations may be governed by the same variables, but the values of these variables that are optimal for some individuals will not be optimal for others. The curves that can be drawn to represent how evaluations vary with a particular variable may well be distinct but have the same general shape" [15]. Information theory provides us with a method of describing the content of art and guidelines for adjusting the order and complexity of a design, but past attempts to devise formulas for calculating the appeal of art using these principles have not been successful.

Mathematician G.D. Birkhoff proposed a formula for measuring the aesthetic value of an object consisting of the ratio of order and complexity [17]. This formula has been shown experimentally to be inconsistent, the product of order and complexity being in closer agreement with current data [15]. Regardless, the application of such a formula to more than the simplest of objects is troublesome, and was not covered by Birkhoff [15]. While the general idea of the formula is sound, and varying degrees of order and complexity can account for some part of aesthetic value, calculating their values for an arbitrary piece of art is much too rigid.

3.2 Uniformity amidst Variety in design

Since our goal is to present the viewer with an optimal level of stimulation, we must determine which elements of a design increase or decrease information content. There exist techniques to both increase the perceived amount of information presented, and to reduce or order the perceived amount of information presented. In the following sections, we describe methods specific to visual perception for increasing and decreasing the amount of information conveyed, as well as methods to balance effectively the two.

3.2.1 What makes a design visually restful?

The ability to organize and prioritize information is necessary for interacting with an environment that provides a constant bombardment of sensory input. Perception stems from the ability to choose which sensory inputs are relevant to deal effectively with this environment [67]. A *restful* design element is one that facilitates perceptual analysis and organization [15]. Restfulness arises from familiarity. Familiarity, in this context, refers to objects or configurations that are predictable, and hence easily recognizable [15]. While too much familiarity is boring, a certain amount is necessary for the understanding of complex designs. By recognizing patterns in the world around us, we are able to make sense of what would otherwise be overwhelmingly complex. The appreciation of art is more than just passive discovery, but a detailed selection process similar to that which evolved in the brain to extract important information from the environment [78].

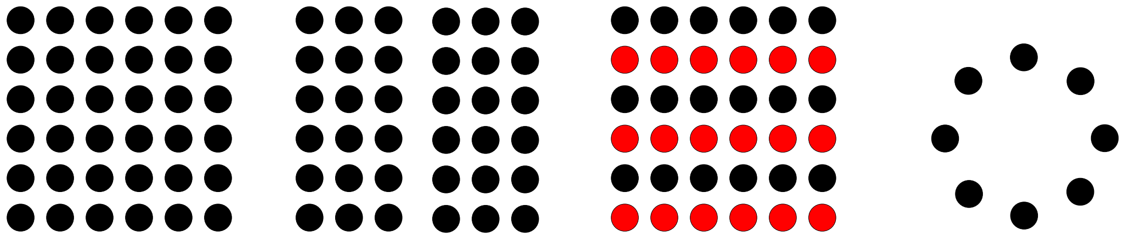
Repetition is the most common expression of familiarity. The same or similar objects are shown in some kind of regular sequence. Repetition allows the viewer to understand all of the information in an image by examining only a small fraction of it [41]. For example, a painting that consists of the same image tiled over and over in a regular grid can be summed up by examining just one of the repeated images. After this single image has been examined, there is nothing left to see; the rest of the painting can be predicted with certainty. The brain is able to simplify a design by identifying the unique elements of its pattern and the manner in which they are repeated.

Symmetry is a special form of repetition. A symmetry of a 3D object is a transformation in 3D space that maps the object to itself. For spaceship design, we are interested in reflections through a plane, such as bilateral symmetry, or rotation around a central axis. Not only is symmetry restful, but the axis of symmetry creates a focal point to guide the eye [41]. This is especially true of bilateral symmetry, where there is a single plane of reflection, and hence a single area of focus. This focal area reduces cognitive effort by allowing the viewer to concentrate on a single non-redundant copy of the object, and creates a restful design because it directs the viewer to areas of interest more readily [63].

However, symmetry's influence on our perception of a design is more complex that just allowing the eye to rest. There are many factors that combine to explain why the human brain prefers symmetry. The brain is finely tuned to detect symmetries. Symmetry detection is an important coping mechanism: it allows us to organize and simplify the huge amount of data we gather from the world around us, allowing us to make sense of the world [15]. Symmetry detection is also an important survival mechanism; many objects that are biologically significant, such as predators, prey, and mates, are symmetric [78]. Symmetry is especially important for mate selection. Experiments have shown that both humans and animals prefer mates that are more symmetric, as asymmetries can be an indication of health problems. Hence, a preference for symmetry has evolutionary advantages [78].

Grouping is another method of conveying familiarity. Grouping allows individual elements to be perceived as a single unit. Grouping elements into larger sets decreases the total number of elements that must be organized; details of individual elements can be ignored in favor of the whole [15]. This is especially useful when the viewer is presented with a large number of objects. Gestalt psychologists have outlined several rules for determining the way in which the brain will compose a set of elements into groups. These include elements that are linked by similarity (elements sharing certain visual characteristics), proximity (elements appearing close together in time or space), and closure (elements perceived to form a complete pattern despite missing or incomplete information) [53]. Examples of these types of groupings are shown in Figure 3.2.

Finally, arrangements that have repetition, pattern, and order imply meaning. If we consider all possible configurations of a set of objects, few of these configurations will exhibit a regular



(a) A set of equally spaced circles in a regular grid.

(b) Proximity: The circles now appear to form two separate grids.

(c) Similarity: Circles of the same color now appear to be grouped in horizontal lines.

(d) Closure: The figure as a whole is perceived as a circle, even though the parts are not connected.

Figure 3.2: Examples of Gestalt principles of grouping

pattern. Since it is far less likely for a regular pattern to appear by chance, we assume that these features have meaning or value [67]. Repetition increases the chance that the viewer will notice an area of interest, not simply by increasing its frequency, but by increasing its perceived significance.

3.2.2 What makes a design visually challenging?

A *challenging* design element is one that inhibits or impedes perceptual analysis and organization [15]. These challenges arise from the unpredictability of the design. Note that unpredictability is reliant on the viewer's expectations of the design; any conflict between the design and the viewer's expectations will increase the effort required to make sense of the design [15].

One way to increase unpredictability is to increase the complexity. The complexity of a design increases with the number of independent elements it contains [15]. As the number of independent elements increases, so does the number of possible combinations and configurations of these elements. For example, consider a connected, undirected graph. If we assume that each edge must connect distinct nodes, then a graph with two nodes has only a single possible configuration; the two nodes are connected by an edge. However, as the number of nodes increases, more complex and diverse graphs are possible.

Complexity of a design can be increased with embellishments. These unnecessary additions are used to disguise the underlying uniformity of the design, making it more challenging to detect [15]. Embellishments are the visual equivalent of noise; they camouflage intended information, making it more difficult to organize. Where patterns can be condensed into a set of unique elements and rules for repetition, embellishments add raw information to the design that cannot be simplified by the brain. Embellishments can add interest to a design that would be too easy to understand otherwise.

Novelty can also be used to increase the amount of information in a design. Novelty consists of unexpected objects or arrangements of objects. Novelty does not necessarily require that the object or arrangement have never been seen by the viewer; it may simply be novel with respect to other objects or arrangements that are in close proximity spatially or temporally [15].

Randomness creates both complexity and novelty. In the selection of a random event, all outcomes have an equal probability of occurring and are thus all equally unexpected. This violation of expectations can create both novelty and surprise [15]. Randomness is also used to create a contrast or background from which we can distinguish order [67].

A design becomes confusing when the viewer cannot distinguish any order or pattern to its elements. This type of design is characterized by complete randomness; a lot of information is presented to the viewer, but there is no inherent meaning or obvious relation between the parts. Because there is no logic or continuity in the arrangement of the design, it is impossible for the viewer to form expectations about it. The design also lacks clear divisions, so the eye is forced to try and process the entire design all at once. There is also no focal point to draw the eye. This level of randomness causes confusion. Just as arrangements that have order imply meaning, those that have randomness do not. With too much information that is seen to be unrelated, the mind tends to filter it out and dismiss it as noise [41].

When used appropriately, what Gombrich called a “calculated dose of randomness” can make a design more interesting by providing new elements for the viewer to discover [41]. As discussed by Berlyne and Boudewijns in their 1971 study, an image’s level of interest increases as its complexity increases, but after a certain threshold of complexity, interest declines [16]. This is consistent with the idea that the mind has an upper limit to the amount of information that can be processed at an instance [67].

3.2.3 What makes a design visually interesting?

The goal of an interesting design is to combine challenging and restful elements effectively to approximate the optimal amount of information that the viewer can comprehend. However, this combination is more than just achieving an information quota; it is possible to increase the amount of challenging information that can be presented through the proper use of perceptual organization techniques. Order and chaos must interact harmoniously in the design. Aside from being tools to increase or decrease the information content of a design, they must also be used to complement and contrast one another.

Assuming uniformity and variety are independent, the best possible design would consist of a set amount of random information, at whatever level the viewer would be capable of taking in at one time. From here, any increase in randomness could be offset by a similar increase in order, maintaining the original information level. However, this simple method will not result in an optimal design. As demonstrated by the principles of curiosity and resolution (which is discussed later in this section), the interest of a design can also be increased by the relations between uniformity and variety, not just their individual proportions [15]. The ineffectiveness of this simple method is also demonstrated with an example in Section 3.3.2.

A challenging design increases interest not only via complexity, but via curiosity as well; the viewer is instilled with the desire to discover the underlying organization of the design. If the design does have an underlying unity which is then resolved, the viewer is further rewarded and the design is perceived to be even more interesting. In this way, the order and complexity of a design combine to increase aesthetic appeal [15]. While the organization of a design can be pleasing in and of itself, a less obvious organization can create greater pleasure through the resolution of a challenge [49]. This principle is also referred to as "perceptual problem solving" [78]. As Platt described it, "What is beautiful is a pattern that contains uncertainty and surprise and yet resolves them into the regularity of a larger pattern" [73].

Conversely, a relaxing design can create pleasure in and of itself, but this pleasure can be increased with the use of subtle complexities or discrepancies. A design with clear groupings or pattern can appeal to the viewer's sense of order. However, if upon closer inspection the order can be revealed to contain hidden or subtle variations, pleasure is derived from the viewer's feelings of surprise and discovery.

Order and complexity can be unified to create variety that adheres to a unifying principle. This method creates interest by forming elements "which are not felt to be boringly obvious but which we can still understand as the application of underlying laws" [41]. In other words, uniformity amidst variety can manifest as design elements that possess both similarities and differences [16]. A simple example of this principle is demonstrated in Figure 3.3.

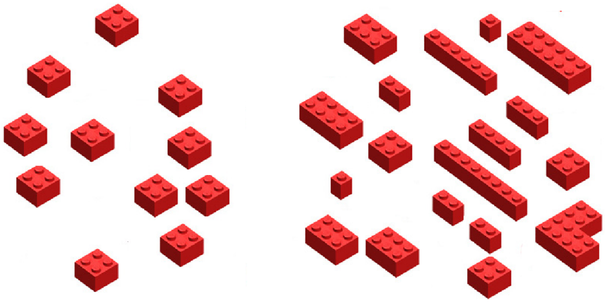


Figure 3.3: This figure demonstrates uniformity amidst variety that adheres to an underlying law. While the blocks on the left are clearly repetitions of the same piece, the set on the right features blocks of various sizes and shapes that are united by an underlying rule.

Uniformity and variety may interact in other ways as well. Randomness is often used to create the background from which order or pattern may emerge, and this contrast can strengthen the form of the pattern [67].

These configurations show how uniformity and variety can be used to complement each other, especially when used *within* one another. Unity that emerges from randomness, such as distinct elements that are derived from a common family, and chaos that appears within order, such as a pattern that appears symmetric but exhibits subtle variations, are subtle, yet effective, ways to increase the visual interest of a design. This recursive nature of uniformity and variety is often demonstrated in nature. Plants, for example, exhibit this pattern in their structure, from the placement of their leaves down to the organization of their molecules [49].

As with all aesthetic rules, uniformity amidst variety alone does not guarantee aesthetic pleasure, but it can be used as a tool to increase the likelihood that a design will be regarded as pleasing [15]. While we have previously stated that the optimal balance of uniformity amidst variety will vary for the individual, the human mind is adapted to perceiving this property. Often the mind will enhance given features of an object, either similarities between differing objects or differences between similar objects, to increase the perceived level of balance between uniformity and variety [47].

3.3 Graded Complication

The previous section described the principles of uniformity amidst variety, including techniques for modifying and balancing these factors in a design. Here we discuss how these techniques can be applied in practice to spaceship design, and present examples of effective designs that can be seen as adhering to uniformity amidst variety, whether consciously, or through an artist's intuition for good design.

It is important to clarify that this application of uniformity amidst variety to spaceships applies only to the creation of surface details. The frame of the spaceship, while an important part of the design, does not incorporate randomness and thus is not a candidate for these techniques. The geometry of the frame is usually chosen with a specific purpose or meaning in mind.

Of the different types of surface details, panels and nurnies constitute most of a spaceship's aesthetics, as we have the most freedom in placing them; other types of surface details have meaning that dictates how they are placed. Because they are more important to the aesthetics of the design, we will focus on the arrangement of these two types of surface detail.

As discussed in Section 3.2.3, aesthetic value comes from more than just varying levels of order and chaos. An object is greater than just the sum of its parts; the way in which these parts are combined to form a whole is also important [49]. We first discuss how uniformity amidst variety can be applied to various levels of spaceship design, then demonstrate this effect by comparing two versions of the Death Star trench: the physical model from the *Star Wars* films and a computer reconstruction created by a fan.

3.3.1 Graded Complication and Spaceships

Uniformity and variety can be combined in a recursive, hierarchical structure to promote harmony. This nesting of uniformity and variety is what Gombrich calls "graded complication" [41]. Graded complication organizes repetition and randomness into hierarchies, such that the design contains different levels of pattern. These hierarchies are created using three techniques: *framing* "delimits the field", *filling* "organizes the resulting space", and *linking* relates the framing grid and the filling motif to create a coherent structure [41]. Linking refers to the fact that we may fill a frame[1] based on properties of the frame or properties of surrounding frames. Framing and filling are nested at various scales to create the final network of pattern [41]. Graded complication is often seen in ornamental art.

---

[1] For clarity, we use "spaceship frame" in this section when referring to the basic geometry of the ship to distinguish from this type of hierarchical structure frame.

There are natural hierarchical levels of organization that appear in spaceship design. At the highest level, the spaceship has a natural order imposed by its frame. If we consider only spaceship frames that are bilaterally symmetric, then this symmetry imposes a high level order on the design. The complexity of the spaceship frame then dictates the complexity of the surface details; geometrically simple spaceship frames warrant complex surface details, and complex spaceship frames warrant simple surface details.

The next hierarchical level is the spaceship's panels. Panels are both a filling motif and a frame. First, the panels are the filling motif of the single frame created by the spaceship frame. Panels link to this frame by arranging and orienting themselves based on the geometry and functional components of the spaceship. They can also be used to highlight certain portions of the ship, such as the windows. Panels also act as frames to contain groups of nurnies or, in some cases, nested panels. The nurnies conform to the shape of the panel in some way, usually by filling it completely or partially. Nurnies often relate to one another within a panel, but can also relate to nurnies in complementary or symmetric panels.

Often, sets of nurnies will be perceptually grouped because they share common features. We refer to these nurnies as "related nurnies". Similarities between nurnies may arise from exact repetition of a shape, shapes that are variations on a common theme, or distinct shapes that are physically connected in some manner. These types of similarities are ordered in terms of strength of grouping, with exact repetition being the strongest type of similarity. In general, nurnies that are closer in proximity benefit from weaker, less obvious types of similarity, whereas stronger types of similarity can be perceived even when the nurnies are not close together.

The regularity of the panels will vary with the density of the nurnies. In locations where the nurnies are tightly packed, panels are less obvious and more likely to be regular in shape and layout. In locations where nurnies are scarce, such as wings, the panels can compensate for this lack of detail by including more complex shapes and patterns. The *Star Trek* Klingon Bird-of-Prey, shown in Figure A.15, features this increased complexity in panel arrangement to compensate for its lack of nurnies, which are left off to emphasize its sleek, streamlined appearance.

Finally, individual nurnies can themselves act as a frame from which smaller nurnies are grouped. This framing can continue at increasingly smaller scales to create an illusion of infinite detail.

Since perception groups from the general to the particular, we are more likely to notice high level patterns and symmetries, and hence we are more likely to notice discrepancies at this level. For this reason, obvious surface details such as panels and surface details that are large relative to the size of the ship's frame are more likely to be symmetric, while this is not as necessary for small details. Large details are seen as more likely to have a specific purpose, while small details might differ depending on circumstances, such as availability of parts, repairs, or slight internal variations. This effect is seen in the conning tower of the *Star Wars* Star Destroyer, shown in Figure 3.4. In this way, graded complication creates a composition of patterns that become increasingly complex.

Aside from the fact that they draw the eye more easily, higher levels of the hierarchy are more likely to feature order and pattern for other reasons. Most importantly, the order of lower levels of the hierarchy is dependent on order at higher levels; if the spaceship frame were a completely random shape, it would be difficult to create a pattern of panels on its surface. Similarly, random panels make it difficult to create ordered nurnies because nurnies are contained within the panels. High level order creates the framework that facilitates order at lower levels of the hierarchy.

Figure 3.4: The conning tower of the *Star Wars* Star Destroyer. Note how large details, such as the sphere-like structures on top, are exactly symmetric, while smaller details on the face of the tower may vary.

3.3.2 The Death Star Trench: A comparative example

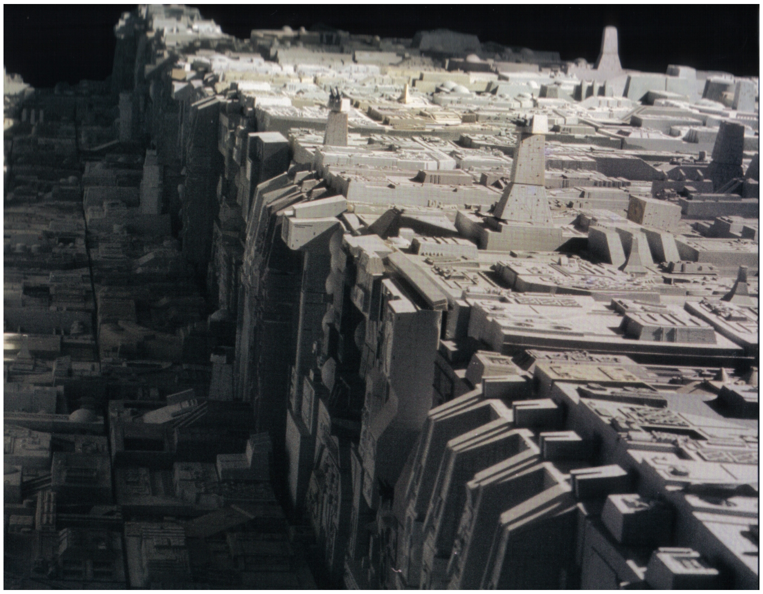


Figure 3.5: The real *Star Wars* Death Star trench. This is a physical kitbashed model.

The *Star Wars* Death Star trench, shown in Figure 3.5, is an excellent example of a large scale spaceship with interesting surface details. However, it is easier to demonstrate what features of the Death Star trench are appealing when we contrast it with the trench recreation from Figure 3.6. While the Death Star trench is a physical model made from kitbashed parts over a period of months, this second trench was created by tiling the surface with fewer than a dozen unique

Figure 3.6: A recreation of the Death Star trench created by repeating the same set of square tiles over and over [3].

blocks [72]. We use the comparison of these two images as a basis for arguing for the benefits of the correct balance of uniformity and variety.

Let us examine the trench recreation. First, this trench does have both repetition and randomness. It is formed by repeating a small set of square tiles with random placement and orientations. However, there is no underlying pattern or organization. In other words, there is no way for the viewer to perceptually organize the elements of the trench recreation. The design is repetitive without symmetry, and so there is nothing to draw the eye. Because the tiles are oriented and placed randomly, there is no sense that they relate to one another in any way, which makes it hard to imagine their purpose. The tiles are in no way connected, except where similar shapes happen to align by chance. With no underlying order, the design appears haphazard, and cannot be perceptually grouped.

Despite the randomness of the tile placement, the design is far too easy to understand. The viewer can tell at a glance how this trench was constructed. It is clear that the panels are arranged in a simple, regular grid, and because there are so few tiles, the design offers nothing new to discover; once all the tiles have been seen, there are no unexpected elements of the design.

In contrast with the real trench, the elements of the trench recreation all have similar scale; there is no variety. Also, the color is nearly uniform. With no wear and tear or variation in the surface texture, this trench appears even less plausible.

The trench recreation clearly features too much repetition. However, even if we were to increase the randomness, perhaps by increasing the number of unique tiles, there is still no way for the viewer to perceptually organize the elements. Repetition and randomness must be combined recursively to create an interesting design.

Now let us examine the Death Star trench from Figure 3.5. The most noticeable difference from the trench recreation is the planned placement of certain elements. For example, the columns along the vertical sides of the trench are placed to form a pattern, giving these elements a sense of purpose that is obviously lacking in the trench recreation's tiles. Though this is not visible in the image, these columns are placed symmetrically on both sides of the trench, creating a focal point at its center. Repetition appears elsewhere as well; notice the towers that appear randomly across the surface. Some small surface details are also repeated, both exactly and as variations of a theme. This repetition is recognizable, yet subtle. Symmetry is used to tie the individual parts of the trench together as a whole, and to create a sense of purpose. The vertical columns appear to line the trench for a reason, perhaps for structural support.

Random elements are then used to fill in the space between these pattern elements. The panels (i.e., the groupings of the nurnies) are not laid out in an exact grid. There are certain regular grid lines that stand out on the surface, such as along the floor of the trench, but these do not appear everywhere and are obscured in some places. Also, these nurnies feature varying levels of repetition and originality. Certain pieces exhibit similarities, or are almost identical save for a few slight variations. Note that the small details on the trench's vertical columns are slightly different for each column. In this way, pieces that appear to be the same at first glance exhibit slight differences upon close inspection, and details that appear random at first actually are variations of a common design.

The surface details of the Death Star trench also appear to relate to surrounding pieces. Because the grid is less regular, even elements that are not expressly connected appear related because of how they fit together tightly on the surface.

The Death Star trench's attention to detail also enhances its level of interest. The colors of the elements are varied, and signs of wear create a sense of realism. Details appear at a variety of scales, suggesting that there are further patterns to be discovered. Care must be taken to avoid extreme precision, especially when dealing with computer generated art. Precise details create an unsettling, inhuman effect, where subtle variations or imperfections create a hand-made, realistic appearance [41]. While these random imperfections appear naturally in physical models, they must be purposefully added to computer models.

This comparison demonstrates how uniformity and variety must be combined in specific ways to increase aesthetic appeal. The recursive nesting of uniformity and variety is crucial in creating an interesting design.

3.4 Summary

This chapter examined information theory's application to aesthetics, specifically the harmony of uniformity and variety. It then detailed methods of varying the levels of uniformity and variety, as well as methods for effectively combining the two into an interesting design. It also described how graded complication can be applied to the creation of spaceship surface details to create a nested hierarchy of uniformity and variety that appeals to these principles of information theory. The effectiveness of these techniques were demonstrated by comparing two versions of the *Star Wars* Death Star trench, one which implements these techniques, and one which does not.

Chapter 4

Spaceship Analysis

As discussed in Section 3.1.1, certain encoding techniques can be used to increase the interest of an object. These perceptual generalizations allow us to provide information to the viewer in an abstract way to increase detail without drawing attention or distracting. This leads to the question, what types of perceptual generalizations can be applied to spaceship design? By identifying the characteristics, shapes, and patterns that best represent spaceships as a whole, we can use this information to create a representation of a spaceship more efficiently. In other words, we create an object with a higher density of spaceship information.

While creating a generally interesting design is important, we must still consider what specific principles contribute to the design of spaceships. There is more to spaceship design than a combination of repeated and random pieces. Among other things, it must appear futuristic, technological, and plausible. First, this chapter discusses these general principles of spaceship design in detail to determine what features most effectively express the qualities of spaceships.

Sections 4.8 and 4.9 discuss design principles that apply to the frame and surface details individually. While most of this chapter describes conventions that are relevant to both mechanical and organic spaceships, these sections are applicable only to mechanical ships.

This restriction is imposed for several reasons. First, mechanical spaceships are a much more common design, meaning that more examples exist to be studied. Because this research focuses on finding patterns in existing designs, this is an important consideration. Second, mechanical spaceships are more traditional and familiar in their design than the exotic organic ships. Mechanical ships are designed so that their key functional components, the cockpit, wings, and propulsion systems, are easily identifiable. Because organic ships are intended to look alien, it is often impossible to determine the purpose of any given part of the ship. This functional component information is useful because it allows us to better decide where to place surface details on the ship to provide these visual clues. Finally, mechanical spaceships almost always feature some amount of surface details, while organic spaceships only rarely feature surface details.

4.1 Expectations

While certain principles can be used to predict aspects of the general population's reaction to a certain object, our perceptions are still heavily influenced by our individual experiences. Learned associations between different objects, or objects and ideas, can alter our aesthetic perceptions. These associations create expectations for what we believe a class of objects, in this case spaceships, should look like. We could take any shape and call it a spaceship, but for survival purposes, the human brain prefers to categorize and recognize things based on previous patterns and habits. This categorization facilitates communication, discovery, and recognition [41].

These expectations are especially important for the design of fictional objects, as we have little other concrete information on which to base a fantastic design. Realism has little to do with the overall effectiveness of this type of design, and in certain cases, realism can detract from a design where the viewer has been conditioned to expect otherwise. For example, ancient Greek architecture has traditionally been portrayed in film as plain white marble, as evidenced by their monochromatic ruins. Though it has long been confirmed that their buildings were actually painted in bright, vibrant colors, they are still portrayed as white because this color scheme has come to be expected by the audience [37].

In relation to aesthetics, the ideal levels of uniformity and variety for a given object depends on the viewer's expectation for that object. If expected to be random, the object will appear more beautiful if it exhibits a higher level of randomness, and similarly if the object is expected to be highly ordered [49]. However, our aesthetic response to a design relies both on its intrinsic effect on our perceptual systems, and its extrinsic connections to our memories and experiences. These responses may interact, and can be seen to lie on a continuum according to level of cognitive abstraction. We now present further examples of associations that may affect a viewer's perception of a spaceship's design.

As spaceship design has now had more than sixty years to evolve, our expectations are most influenced by our knowledge of popular spaceships. Our expectations of what a spaceship should look like are influenced by these past experiences. This influence implies that a thorough examination of spaceship design requires more than just aesthetic principles. An in-depth survey of existing designs is necessary to determine what principles are associated with good design. This survey is presented in Appendix A.

Another example of this phenomenon with respect to spaceships is their association with traditional aircraft, such as airplanes. Airplanes are designed to be aerodynamic, as these streamlined shapes allow the plane to move faster by limiting wind resistance. While spaceships have no need to be aerodynamic (ignoring the case where the spaceship may also be capable of flying in atmosphere), we still associate these sharp, streamlined shapes with speed. To quote John Berkey, "A triangle looks more like speed than a rectangle" [36]. Real-world objects that we know are fast, such as planes, tend to be aerodynamic. All types of vehicles are designed to be aerodynamic: triangles instead of rectangles. But instead of consciously recognizing that objects shaped like this are aerodynamic, and hence fast, an unconscious association is made between these shapes and speed. There is nothing in our everyday experience of the world to prepare us for the lack of drag in space. Thus, when seeking archetypes for space-faring vehicles, we naturally turn to known, streamlined, earthbound forms.

While it is possible that our expectations may be influenced by knowledge of how technological or mechanical devices work in reality, in general this limitation is not an issue. Effective spaceship design, and science fiction in general, relies on the viewer's suspension of disbelief. While the design must appear to be bound by some laws, as discussed in Section 4.7, these laws are not necessarily the same as those that govern the design of our current technology. In fact, it is more likely that the viewer expects these laws to be different. For example, current spacecrafts are powered by thruster propulsion systems; a force expelled in one direction moves the ship in the opposite direction. However, we are not necessarily bothered by spaceships that feature no visible propulsion system.

While expectations undoubtedly play an important role in any type of design, spaceship design is less constrained than most. More established types of designs, such as cars, are much more sensitive to change; it seems that "the entire world knows what a cool car looks like" [82]. Furthermore, deviating from the target viewer's expectations does not necessarily mean that the design will not be enjoyed. An unexpected design is crucial for making a spaceship appear exotic and alien. *Star Trek*'s Borg Cube, shown in Figure 4.1, is an iconic example of this: the designers chose a frame that in no way resembled a traditional aircraft or vehicle to deliberately violate established conventions for spaceship design. The Cube's blatant lack of directionality, order, and recognizable functional components resulted in a unique design.

Figure 4.1: *Star Trek*'s Borg Cube violates all previous spaceship expectations to create a truly alien design.

4.2 Organic and Mechanical Spaceships

At the highest level, there are two fundamental classifications of spaceship: organic and mechanical. While some hybrids do exist, they are rare, and most spaceships tend to fall clearly into one category or the other. There are various hybrid or techno-organic races that appear in science fiction, most notably the Borg from *Star Trek*, however their spaceships tend to be mechanical in appearance, not hybrids. Spaceships that are hybrids tend to have frames that are influenced by organic forms, but surface details that are mechanical. Note that this discussion of hybrid spaceships refers only to a ship's appearance; spaceships that appear mechanical but are actually organic, such as the Cylon raiders from the 2004 *Battlestar Galactica* seen in Figure A.7, or vice versa, are not considered as hybrids here.

Mechanical ships are the much more common design, and tend to be influenced by traditional technology. Their designs are more likely to be influenced by real-life aircraft, and their components, as well as the intended functions of these components, are likely to be recognizable. This information is useful from a design perspective because it means that this type of ship is relying on visual cues, which can hopefully be identified, to convey information about its function. Mechanical ships are comprised of panels and can be covered in a rich variety of mechanically inspired surface details. Mechanical ships favor traditional thruster based propulsion systems, though more exotic propulsion systems are not uncommon.

Mechanical spaceships are an interesting class of objects in that, unlike actual mechanical or electronic devices, the design is intended to make them *appear* mechanical. However, real devices will either look mechanical inherently, as form often follows function, or they will not; since the real device *is* mechanical, it is not necessary to intentionally make it appear mechanical. Since these spaceships are fictional, the aim of the design is to create the appearance of a plausible, functioning mechanical device. Despite the fact that these machines are fictional devices, this imposes certain constraints on the design, as we have certain knowledge and expectations of how a mechanical device should look. An example of a mechanical spaceship is shown in Figure 4.2.

Organic ships are used to emphasize the strangeness of their creators and are always alien in nature. Because of this, it is often impossible to determine the function of any given part of the ship, as their technology is intended to appear foreign and advanced. Organic ships, like organic creatures, tend to be smooth with few to no surface details, including panels. Their frames take a wider variety of shapes than mechanical spaceships, though they are often influenced by fish or plant shapes. They usually have no visible means of propulsion. Because organic ships are so different from any technology we experience in the real-world, they are not limited by the connections to reality that mechanical spaceships are subject to, allowing for a greater diversity. While this class of spaceship emulates the appearance of a living organism, it is not mistakable for such an organism, creating a feeling of disconcertion that likely stems from the uncanny valley hypothesis. The alien qualities of these spaceships may also originate from the concept of spaceships as objects that were consciously design. The idea of designing an organism is so foreign to us that we assume an organic vehicle must be alien. While plausibility is still an issue for this style of ship, its exotic appearance allows for more flexibility than mechanical designs. An example of an organic spaceship is shown in Figure 4.3.

Most of this chapter can be applied equally to both mechanical and organic ships. However, since organic ships tend to have few surface details, sections dealing primarily with surface details will consequently focus on mechanical ships, and an explicit statement will be made to this effect. The reader can assume that the following analyses apply equally to both types of spaceship unless otherwise stated.

4.3 Symmetry

When life first formed on Earth, it exhibited highly symmetric body plans. As these simple organisms evolved to better interact with their anisotropic environment, certain asymmetries formed in reaction to the forces or motivators that were present [94]. The first motivator was gravity; as the

Figure 4.2: An example of a mechanical style spaceship, an X-Wing fighter from *Star Wars*.

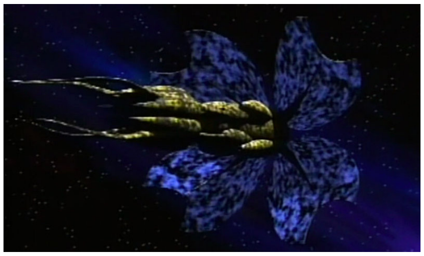


Figure 4.3: An example of an organic style spaceship, a Vorlon transport from *Babylon 5*.

lower half of the body needed to interact differently with the environment, namely the ground, it required certain specializations. Top-bottom asymmetry developed to allow for improved mobility. The second motivator was locomotion; front-back asymmetry developed to provide a sense of directionality to enable rapid motion, both to catch a prey and escape a predator. Finally, only bilateral symmetry remains. Because there is no force causing any preference or difference in interaction between an organism's left and right halves, this symmetry remains. This type of asymmetrization can be found in varying degrees in different types of plant and animal life [39].

In Earth's atmosphere, vehicles are subject to the same forces that evolved these specific asymmetries. Gravity requires that the top and bottom of a vehicle provide different functions. Aerodynamic concerns lead to asymmetries between the front and back of a vehicle. However, conditions in space are not the same as on Earth. With no gravity and no need for aerodynamics, there are no forces requiring spaceships to mimic these asymmetries.

However, functional need is not the issue. While the spaceship itself may interact with the environment exactly the same way in any direction, the pilot of the ship cannot. If the ship is small enough that it is controlled directly from the perspective of the pilot, the movement of the ship is subject to the same asymmetries as the pilot. We are assuming that the pilot is human and thus abides by these evolutionary asymmetries, hence the ship must abide by them as well. It is not useful for the ship to be able to fly in any direction because this functionality is not compatible with the capabilities of the pilot. Since the pilot is unable to see and maneuver equally well in any direction, the ship must turn first before moving. Hence, the spaceship exhibits front-back asymmetry to accommodate locomotion.

The top-bottom asymmetry is less straightforward. While gravity is absent in space, and thus the function of the ship is not limited with respect to these directions, top-bottom asymmetry is required to properly orient the pilot in space by providing an up direction. This asymmetry also allows the viewer to infer the orientation of the spaceship.

For larger ships, these top-bottom and front-back asymmetries are not as crucial. Because of the scale of the ship, its movement is not directly associated with the perspective of the pilot. It is not necessary for its asymmetries to be complementary with those of the pilot. The pilot is not controlling the ship by looking out the windows; it is sometimes difficult to tell where in the ship the pilot would be located. Hence, the ship has more freedom of movement.

It is important to note that, in the case of alien crafts, these asymmetries are not always reliant on the form of the actual pilot, but on the traditional human form. Though alien pilots may not be subject to the same asymmetries, their spaceships generally are.

These asymmetries are also important for aesthetic purposes. From the perspective of the viewer, the orientation of the spaceships is important to understanding the scene. Asymmetries provide a frame of reference; they allow us to properly orient the ship in space by distinguishing between the front and back of the ship, as well as the top and bottom. However, science fiction scenes are often depicted with multiple ships oriented the same way and lying in the same plane, though this arrangement is unlikely to occur by chance. *Star Trek* is notorious for this bias [80].

Another consideration is our predisposition towards certain types of vehicle asymmetries. It is not hard to understand why we would design spaceships to have the same types of asymmetries as all other vehicles we design. Bilateral symmetry appears in nearly all of our vehicles.

Close inspection will show that perfectly symmetric spaceships are rare. Certain features are more likely to be symmetric than others. The frame of the ship, for example, is almost always perfectly symmetric, panels and other large surface details are usually symmetric, but small details are less likely to be perfectly symmetric. This relates to the factors that make a ship interesting, as discussed in Chapter 3. Symmetries may also be violated to convey damage or age, as missing or broken surface details are not likely to occur symmetrically.

4.4 Scale and Direction

Spaceships are most often depicted in the vast emptiness of space, with no outside frame of reference that can be used to infer scale and direction. Therefore, it is important that the ship's surface details convey these characteristics. The absence of these visual cues, or worse, the presence of conflicting cues, can easily cause the viewer to become disoriented.

The sizes of recognizable features, most commonly windows, are inversely proportional to the size of the ship as a whole [72]. Often large ships will have no visible windows at all, instead representing them as points of light. The size of the ship is seen to be larger as the density of lights increases, and smaller as the density of lights decreases. While windows convey the strongest sense of scale, this principle of proportions can be applied to all other surface details as well. Small ships tend to have a low density of large surface details, while large ships tend to have a high density of small surface details.

Other objects with discernible scale, such as ladders, may also be used to convey the scale of the spaceship. This is useful for medium sized ships, and ships that are designed without windows.

The conventions for conveying a sense of direction are not quite as clear. For small ships, the placement of the windows is often enough to clearly show the direction of the ship. However, for large ships with many or no windows, this placement is not helpful. Instead, an obvious placement of the propulsion system may be used. This is especially useful for ships that are unconventionally shaped. If the propulsion system is exotic or not visible, it is usually safe to assume that the most aerodynamic end of the ship is the front.

In certain special cases, such as with space stations, no visual cues to imply direction are present. Space stations can be thought of as stationary spaceships; therefore, it is best for them to present as few directional cues as possible to prevent confusion. To accomplish this, space stations often exhibit radial symmetry (or a suggestion of radial symmetry), and a single top-bottom asymmetry. This lack of directional cues is the only real difference between space stations and spaceships. For example, by shaping the *Star Wars* Death Stars as spheres, they completely lack directionality and remind the viewer of a stationary planet. The Death Star is shown in Figure 4.4. The titular space station from *Babylon 5*, shown in Figure 4.5, does not exhibit radial symmetry, making it easier to confuse it for a spaceship.

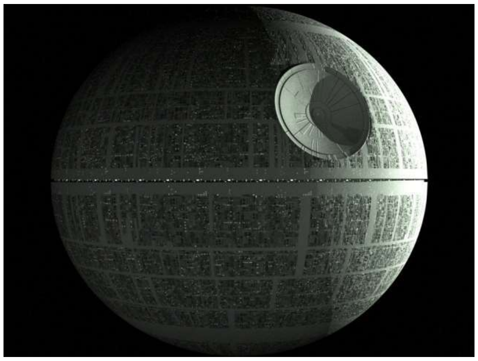


Figure 4.4: The *Star Wars* Death Star's lack of directionality mimics that of a planet.

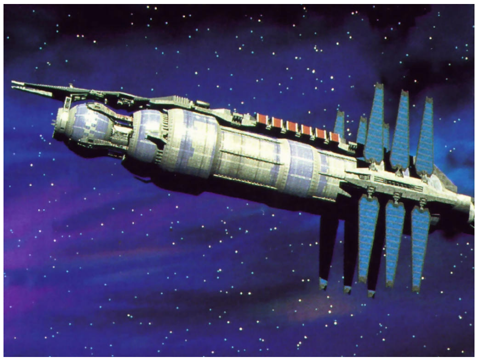


Figure 4.5: *Babylon 5*'s titular space station has only bilateral symmetry, making it easier to mistake it for a spaceship.

Saucer shaped spaceships also intentionally lack directional cues. As saucers are almost always portrayed as alien spaceships, this lack of directionality (and the lack of an obvious propulsion system) give saucers an advanced, alien appearance. Also, they are often shown tilted to create an appearance of motion to avoid mistaking them for a space station.

4.5 Technology

For mechanical spaceships, it is important to create the appearance of a working, technological device. Creating an effective spaceship design requires the use of certain forms and patterns to symbolize these features. Because the designer is not creating an actual mechanical device, attempting to design a spaceship in this way would not be effective. Form following function does not work when the designer is inventing the function. Specifically, we want to create a perceptual generalization of technology. The spaceship should appear mechanical as a whole, without drawing attention to the individual components.

There are two general approaches to spaceship design with respect to technology. The first method simulates advancement through complexity, and is colloquially known among science fiction fans as the "used future" design method. By packing the spaceship with a variety of devices and parts, it appears complicated and thus advanced. The second method simulates advancement through mystery, which, by analogy, we call the "new future" design method. By making the spaceship as sleek and bare as possible, it appears mysterious and beyond the comprehension of our current technology. Examples of new and used future designs are shown in Figures 4.6 and 4.7.

Figure 4.6: The *Alien* franchise follows the "used future" design methodology. Shown here is the U.S.S. Sulaco from 1986's *Aliens*.

Figure 4.7: The "new future" spaceship from the 1986 film *Flight of the Navigator* is sleek, shiny, and devoid of surface details.

While a well-chosen frame can provide a natural semblance of advancement, the technological appearance of a spaceship is largely determined by its surface details. Since we are interested in the creation of surface details, we are more interested in the "used future" design. The goal of surface details is to take an arbitrary shape, the frame, and give it the appearance of a mechanical whole, without distracting from the overall design.

There are specific details that can go a long way towards making a design look mechanical or technological. Mechanical primitives such as gears, grates, fans, coils, bolts, and vents are generic mechanical objects that can be used for this purpose. Pieces that create connectivity, such as pipes and wires, are also useful for this, but are more important for creating cohesiveness and connectivity. Arrangements of surface details are discussed in detail in Section 4.9.

4.6 Personality

Designers often seek to convey certain characteristics or evoke certain emotions with their designs, allowing the spaceship to speak for itself about the personality of the crew, race, or fleet it belongs to. Personality can be used to add depth to a design by connecting the spaceship to a group or idea outside of itself.

For film and television, conveying personality in the design of a ship is especially important. As the audience often has only seconds to digest a scene before it changes, spaceship design is used strategically to convey information relevant to the plot, such as the owner, alignment, and type of the ship [21]. These personality traits may be necessary to understanding the spaceship's role in the narrative.

Representation of personality traits is due to convention. Features of a spaceship that evoke different traits are highly dependent on learned associations from past experiences. These associations are demonstrated in color choice: in Western culture, black is most often associated with evil and white is most often associated with good. Therefore, spaceships that are darker in color are more likely to be assumed to be evil. These color associations tend to be culture-specific. For example, Americans often associate the colors red, white, and blue, the colors of their flag, with patriotism and honor.

Two striking examples of spaceships that convey strong personality traits, the Shadow vessels from *Babylon 5* and the USS Enterprise from *Star Trek*, are shown in Figures 4.8 and 4.9.

Figure 4.8: The Shadow vessels from *Babylon 5*, with their pointy, spider-like frames were designed to evoke feelings of fear, power, and stealth. Their signature smooth, black surface associates them with darkness and evil.

Figure 4.9: The USS Enterprise from *Star Trek* features the red, white, and blue color scheme common to all Federation starships. In the *Star Trek* universe, the Federation represents a force for liberty, peace, and justice in the galaxy.

Certain personality traits are more important than others. Imagine a space traveller approaching an unknown spaceship. What characteristics of this ship would it be most beneficial for the traveller to be able to deduce? For survival purpose, it is likely most important to determine whether the ship is hostile or friendly. The unknown ship's origins may be crucial in determining the traveller's actions as well. More generally, the traveller may wish to determine whether the ship is human or alien. Finally, the primary function of the unknown ship may also be relevant. A civilian transport poses less of a threat than a military vessel. These three categories are discussed in the following sections.

4.6.1 Good versus Evil

The terms "good" and "evil" are used loosely in relation to spaceships. Usually "evil" translates to aggressive, and "good" is everyone else. Good and evil may also be a matter of perspective; the protagonist may feature "good" design characteristics despite his actual moral alignment, because the viewer is intended to identify with him.

As stated above, the obvious color choices for good and evil are white and black, respectively. However, this color choice is relative; dark colors of any hue can be seen as evil when contrasted with a relatively light color.

Not surprisingly, spaceships that feature a large number of weapons tend to be seen as more aggressive. However, sharper, jagged shaped spaceships are also seen as being more aggressive. In a 1957 study by Provins et al., subjects were shown a pictorial representation of a radar featuring two groups of aircraft represented by a cluster of circles and a cluster of triangles [74]. When asked to determine which group of aircraft were the aggressors, nearly all subjects selected the triangles. In a similar study by Halper et al. in 2003, subjects were shown a simple line drawing of a house and a group of trees and asked to select the safest location. Cultural motivators guide most subjects to select the house as safer in this original image, but when the image is altered so that the house is rendered in jagged, threatening lines, most subjects select the trees as the safest location [33]. This study is interesting because it shows that the sharpness of an object can denote a threat even for objects that are traditionally associated with safety. Both of these studies demonstrate that sharp, pointed shapes are commonly associated with aggression. However, this does not imply that softer, rounded shapes are associated with safety and civility, merely that they are neutral.

4.6.2 Human versus Alien

There are a few common features that can be used to determine if a spaceship belongs to a human or alien race. We use the terms “human” and “alien” loosely here, as “human” is best defined as the race or people which the viewer most identifies with.

Note that the features presented here are relative, not absolute. For example, alien ships tend to feature more advanced technology, which means they tend to either have few or many surface details, relative to the ships established as human. Alien ships appear foreign because of their contrast with the human ships. Important features are summarized in a flow chart in Figure 4.10.

The strongest indicator of race is whether the spaceship is organic, since human ships are never organic. Alien ships may still be mechanical, so further examinations are required to determine race if the spaceship is mechanical.

Alien spaceships usually feature a wider variety of colors, often used to visually differentiate distinct alien races, while humans tend to favor traditional metallic colors, such as grey and beige for the main body of the ship.

Alien spaceship frames also tend to be more free-form, while humans tend to gravitate toward traditional aircraft-inspired shapes.

With respect to surface details, alien spaceships favor designs that appear more technologically advanced. As discussed in Section 4.5, this leads to the two extremes of sparse or dense surface details. Alien ships also feature less uniformity in the layout of their surface details, while humans are more likely to conform to bilateral symmetry. Panel shape can also be an indicator of race. Humans almost always use quadrilateral panels, likely because this shape of panel is most commonly used in real-world construction. Any other panel shape is a strong indicator that the spaceship is alien.

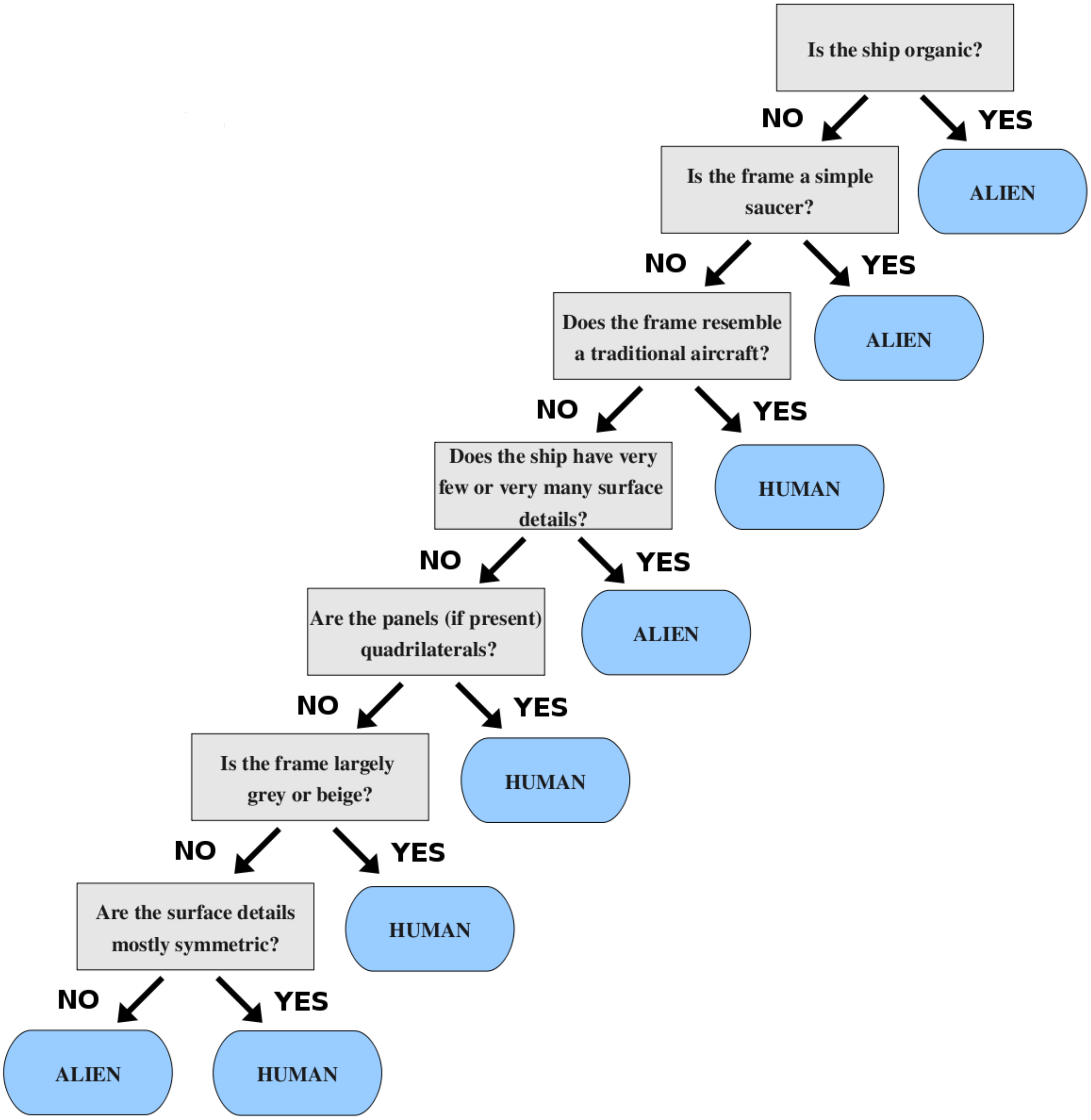


Figure 4.10: A flowchart to estimate the origins of a given spaceship.

4.6.3 Intended Function

The primary function of the ship can be determined by identifying which functional components are the focal point of the ship. Making one functional component more prominent than the others implies to the viewer that this part of the ship is most important to its function. We discuss functional components in more detail in Section 4.8.1.

A prominent propulsion system implies speed, large or elaborate wings imply maneuverability, many weapons imply militaristic affiliation or aggressive tendencies, and a large cockpit or main hull usually means the vessel is primarily a transport or cargo vessel. The *Star Trek* Enterprise, shown in Figure 4.9 is an excellent example of this principle; its prominent propulsion system and lack of weapons are indicators of its peaceful, exploratory function.

While the characteristics of a ship are generally introduced within the context of the story in which it appears, these visual cues allow the characteristics of the ship to be determined without this explicit mention. For example, the viewer only knows that the *Star Wars* Millennium Falcon is fast because it is mentioned in the story; in fact, it is emphasized in the story that the ship is intentionally designed to be more than meets the eye, as there are no visual cues to make it appear fast. However, this ship is a major plot device in the films. In general, the designer wishes the viewer to be able to interpret these characteristics automatically, without having to introduce each and every ship.

4.7 Plausibility

Plausibility is the most important factor in spaceship design. Though the resources to create computer models were available, designers of the *Star Wars* prequels often chose to create physical models instead; they were concerned that computer models would be “too far removed from the physical boundaries of reality” [72]. The idea was that a model that can be physically constructed will automatically have a certain level of plausibility, because of the physical properties and limitations of the materials used to build it [72]. While science fiction calls for a certain suspension of disbelief, viewers are becoming increasingly technologically savvy and are not fooled by illogical or impossible mechanical designs. However, it can be difficult to define what makes a fictional vehicle appear plausible.

Spaceships, like all technology, are engineered and designed, which means that their parts should appear to have reasons for being the way they are. The viewer takes what Dennett calls the “design stance”, the belief that a constructed object should behave as it was designed to behave [28]. Plausibility, in terms of spaceship design, is achieved by conveying this sense of conscious design. Essentially, the ship needs to appear to be bound by some set of laws [41]. While these laws may have little to do with any actual laws of physics or current design conventions, the design of the ship must be consistent.

Conscious design is best conveyed with order. Regularity is so unlikely to occur randomly, that when it appears, we naturally assume that there must have been some intentional design that caused it [49]. However, irregularity does not necessarily imply a lack of design; it may simply be that the design is too complex for us to understand, or that the irregularity is itself the design [49].

Repeated elements arranged in a pattern create the impression that they were placed there for a reason. This is especially true of symmetry. Rorschach ink blots are an excellent example of using symmetry to invite meaning. Rorschach ink blot tests are psychological tests in which the patient is presented with a set of ink blot images to interpret. These interpretations are then used to make inferences about the patient’s personality. The blot is created by dabbing ink onto a page and then folding the inked page in half to create a symmetric image. Where an image that has not been folded looks like just an ink blot, symmetry causes the blot to lose its haphazard look and causes the viewer to assume the existence of meaning [41].

Connectivity between elements is important to creating this sense of conscious design. Elements of the ship, at all levels of granularity, must be combined together in a meaningful way. This refers to both how pieces connect together to make larger elements, but also to how different configurations of similar elements create diversity while implying the existence of underlying laws.

Plausibility also involves integrating the new and the fantastic with the familiar. Familiar technology may refer to real technology as well as to well-known science fiction designs. This integration leverages the plausibility of the familiar design to lend credibility to the new additions. This is why new spaceship designs are often reminiscent of older designs. By associating a new design with an established design, even unconsciously, the new design can leverage the plausibility of its predecessor. Similarly, spaceships that are reminiscent of a familiar object, especially a vehicle or aircraft, also gain this sense of familiarity.

Other minor details can also contribute greatly to the plausibility of a design. For example, perfection can appear unnatural; rules need to occasionally be broken. This can be achieved by introducing signs of wear, broken or missing parts, or slight asymmetries.

4.8 Frame

The frame of the spaceship is its basic shape, devoid of all surface details. The frame does not necessarily hold up as a spaceship on its own; surface details are usually required to create the appearance of a spaceship. Only frames that mimic traditional aircraft are capable of getting away without surface details. This means that frames are able to take on more arbitrary shapes, since it is the surface details that are responsible for the spaceship look.

While good configurations of surface details can be defined in terms of aesthetics, good frame configurations are much more difficult to define. However, there are a few observations we can make about frame shapes.

First, it is clear that good frames can be created by imitating a known object. Many spaceship frame designs are influenced by a real-world object. Vehicles are the most common influence, including a variety of planes, cars, and boats. Animals are also a common influence. Insects appear most frequently, but birds, fish, and reptiles are also seen, with a clear preference for predatory animals. Plants, including trees, flowers, and seeds, projectile weapons such as guns, bullets, darts, and missiles, and tools such as shields and wheels are also common. The chosen object usually relates to the purpose or personality of the spaceship in some way. While this type of frame shape is clearly influenced by the selected object, it is not an exact depiction of the object. Some distortions or modifications are typically applied to form a spaceship inspired by a real-world the object.

There are also certain popular shapes that appear in a variety of contexts. The saucer or disc is one of these shapes, appearing in dozens of spaceship designs in an assortment of configurations. However, there are no clear patterns for how these popular shapes should interact and combine with other shapes. It is not clear how to combine arbitrary shapes into a frame that does not imitate a known object.

It is important for the spaceship frame to have a strong silhouette. Science fiction artist and film designer Doug Chiang's first key to spaceship design is creating a strong silhouette [21]. Silhouettes are important in design for many reasons: understanding, recognition, and ease of recollection, just to name a few. A strong silhouette is one that is easily recognizable; in other words, it is unique in some way such that it stands out to the viewer. The use of strong silhouettes to direct the viewer's attention to points of interest is a well-established technique in animation, called *staging* [57]. Unfortunately, it is not clear how to define the features of a strong silhouette, never mind how to create one.

The scale of the spaceship also dictates certain features of the geometry of its frame. As noted in Section 4.4, small spaceships tend to have a low density of large surface details, while large ships tend to have a high density of small surface details. Aside from indicating the scale of the ship, there is another explanation for this convention. Small spaceships are more likely to look like traditional aircraft, and therefore need less help from the surface details to look like a spaceship. The focus of these ships is the cockpit windows, and too many small surface details would overwhelm the design and distract from this. Small spaceships are also usually intended to be faster, so the absence of surface details contributes to their sleek appearance.

Large spaceships, however, are more likely to have an arbitrary frame shape, which may not resemble an aircraft in any way. They also tend to have simpler shapes, so large spaceships compensate by having more surface details to make them appear more spaceship-like. For example, many of the large spaceships in *Star Wars*, such as the wedge-shaped Star Destroyers and the spherical Death Star, would not inherently look like spaceships if completely devoid of surface details. The films' small scale spaceships, such as the X-Wing fighters, begin with a frame that is already reminiscent of a vehicle of some kind.

Certain sections of the frame may also be affected by their function or purpose. The following section discusses these functional components and their impact on the spaceship frame.

4.8.1 Functional Components

A *functional component* is a section of the spaceship's frame that serves a definitive (though possibly fictional) purpose. To be considered a functional component, the purpose of the section should be visually identifiable.

As with other aspects of spaceship design, the definition of functional components must avoid focusing on specifics of fictional technologies. These details are distracting, and irrelevant to the appearance of the ship. Whether or not a spaceship can travel faster than light is not important to its appearance; there are no inherent visual cues that can be added to a ship to automatically signify to the viewer that this ship is capable of faster than light travel. The design of the ship is not dependent on the specifics of fictional technology. However, these features may *indirectly* affect the design of the core functional components. For example, the designer may choose to emphasize the spaceship's propulsion system to signify its speed, though knowledge of the specifics of its capabilities are not necessary. This distinction allows the design of a spaceship to be based directly on the visible features of functional components, instead of specifying a fictional, ill-defined requirement and attempting to associate it with the ship's outward appearance. As another example, instead of specifying that our spaceship must be capable of atmospheric flight, we would instead specify that the spaceship must have wings.

The set of functional components we define are as follows: cockpit, wings, weapons, and propulsion. These were selected for the following reasons. First, these sections of the ship, when present, are immediately identifiable. This means that there are certain standards or conventions that set these sections apart. Second, surface details behave noticeably different depending on location. This means that different methods must be applied to place surface details on different functional components. Third, these components are independent of technological specifications, but allow almost all desired features of the ship to be specified indirectly. Similarly, the type of spaceship can also be implicitly defined by altering this set of functional components, so that it is unnecessary for the designer to directly specify which type of ship they wish to create. These features are discussed as each functional component is examined individually below.

Not all of these functional components will be present on all spaceships; their inclusion is dependent on the ship's size and purpose. Also, it is expected that there may be sections of the ship that cannot be clearly categorized as belonging to one of these functional components. Generic sections of the ship may exist to connect separate functional components without belonging to one or the other, to increase the mass of the ship without altering its existing components, or to serve some indistinct purpose, such as a cargo hold or passenger quarters.

The following examines each of the four functional components in detail.

Cockpit

The cockpit is the control center of the spaceship, and may also be referred to as the bridge for large scale spaceships. If the ship is flown manually, this is where the pilot would be located. The cockpit is defined by the presence of windows, which often exist whether or not this view is necessary for the piloting of the ship.

The cockpit is crucial for expressing the size and direction of the ship. For small and medium sized spaceship, the cockpit windows should be clearly visible, as their size is the most important visual cue for the ship's scale and direction. Large ships may or may not have an identifiable cockpit, and often rely on other visual cues to relay scale and direction.

Particularly alien ships that purposefully omit scale or direction information will not have an identifiable cockpit. This is often the case for saucer-like spaceships.

Wings

Realistically, wings are only needed for flight in a planet's atmosphere. However, wings, like aerodynamics, are strongly associated with flight and speed. So while the presence of wings may imply that a spaceship is capable of flying in atmosphere, this is not necessarily the case. More likely, the presence of wings implies speed and maneuverability.

Wings are most common on small scale ships, though they sometimes appear on medium scale ships. Large scale ships almost never have wings. This follows from the wings' association with speed and mobility, as small ships are much more likely to be fast and maneuverable than a large, bulky ship.

Wing surface details are generally restricted to panels. Small nurnies appear occasionally, but usually in the context of exposed parts from a missing panel. This lack of extra geometry is to preserve the wing's smooth, aerodynamic appearance, emulating the appearance of airplane wings.

Weapons

While many spaceships do have weapons, most are retractable and not an important feature of the design. Visible weapons generally indicate aggression and hostility, and thus may inaccurately color the viewer's perception of a ship. For example, *Lexx*'s titular spaceship, despite being a super weapon, does not feature any visible weapons systems, as its primary role in the show is not based on its weapons capabilities. The Lexx is shown in Figure A.94.

The presence of weapons systems is also dependent on the type of ship. Military ships almost always have prominent weapons systems, but civilian ships usually do not.

Propulsion

The spaceship's propulsion system is any system used to displace the ship. This includes both traditional and exotic propulsion systems. Traditional propulsion systems mimic those of real aircraft and spacecraft: a reaction creates a force that expels some propellant in the opposite direction of flight, and the equal and opposite force propels the craft forward. This reaction is contained within a set of cylindrical thrusters at the rear of the craft. Thrusters tend to appear in groups of odd numbers, one and three being the most common, though there is usually an even number of groups. More traditionally shaped spaceships, such as *Battlestar Galactica*'s Viper and Raptor, tend to feature traditional propulsion systems.

Exotic propulsion systems encapsulate all other types of system. Any propulsion system that is visibly recognizable as such, yet does not resemble a traditional propulsion system is categorized as exotic. The *Star Wars* Millennium Falcon's strip of light is an example of an exotic propulsion system.

The propulsion system serves as a directional cue for large spaceships and other ships with no visible cockpit. Most spaceships have a visible propulsion system, though certain exotic ships, such as the *Star Wars* TIE fighters, do not. A prominent propulsion system tends to indicate that the spaceship is intended for transportation or exploration, though it may also be an indication of raw power or speed.

4.9 Surface Details

While the frame is a crucial part of the design, it is often the surface details that contribute most to the spaceship aesthetic. Surface details are responsible for providing many important visual cues, as well as maintaining the balance between uniformity and variety.

Certain spaceship surface details perform more specific functions than just creating visual interest. The following sections discuss the behavior of these surface details, including panels and semantic details.

4.9.1 Panels

A *panel* is a division on the surface of a spaceship. Panels serve to simulate that the surface of the spaceship is created from individual components. In certain cases where the surface of the spaceship is densely packed with surface details, there are no visible panels. In this case, panels refer to the clusters or groupings of surface details.

The panels' layout is almost always bilaterally symmetric, except where panels are broken, missing, or faded to signify age or damage. Panel layout can be varied, but tends to follow the edges of the frame geometry. Panels are also used to frame focal points of the ship or draw attention to certain details. Modified grid and radial patterns are most common.

Panel layout is comparable across different scales of spaceship, except for large scale spaceships, such as the *Star Wars* Death Star or the *Star Trek* Borg Cube. For this extreme scale of spaceship, panels are not visible at all except during close-up views of the spaceship's surface. Because of this, panels of large scale ships tend to be either regular, almost grid-like, or completely random. Large scale spaceships usually feature rectangular panels.

Panel shapes are nearly always simple convex polygons. By far the most common panel shape is quadrilateral, with a special preference for isoceles trapezoids. Sometimes triangles will appear in conjunction with these shapes when quadrilaterals alone cannot fit the geometry of the frame nicely. Non-convex polygons, polygons with greater than four sides, or curved shapes can appear, but usually only on alien ship designs. Human ships favor traditional quadrilaterals. Note that while panels tend to have simple polygonal shapes, these polygons may not lie perfectly in the plane if the surface of the spaceship is slightly curved.

Nurnies are usually contained within their respective panels, though not always. There may be surface details that are placed independently of the panels or that serve to connect geometry between panels. For the most part, nurnies are confined to a given panel and attempt to fill the panel completely, fill a part of the panel that approximates its shape, or add a single piece or small cluster of geometry. These behaviors may be coordinated between nearby panels to form patterns or clusters of surface details.

Sometimes, complex panels will be used in place of nurnies to add visual interest to the surface. This is especially common in places where nurnies are not appropriate, such as the wings. The *Star Trek* Klingon Bird of Prey, shown in Figure A.15, features detailed panel shapes on its wings that simulate a bird's feathers. Panels may also be colored in a decorative pattern or grid. An example of this type of coloring can be seen on the *Star Wars* X-Wing and Y-Wing fighters, which use colored panels to denote squadron. These panels often follow the edges of the geometry of the ship.

Panels may also feature decorative line patterns instead of nurnies. These line patterns will approximate certain aspects of the panel's shape while introducing random deviations and are usually seen on quadrilateral panels. We see a simple application of this pattern on the rectangular panels of the *Star Wars* Death Star turrets shown in Figure 4.11.

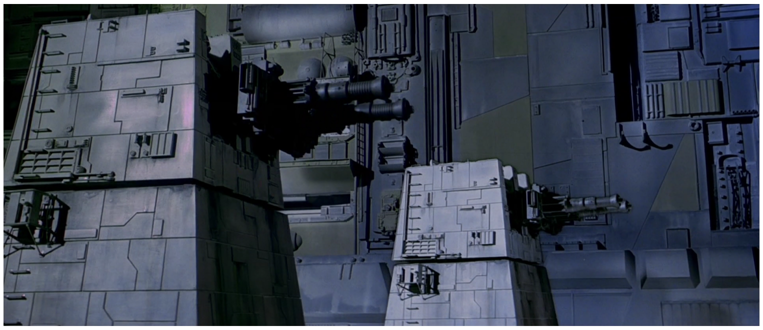


Figure 4.11: The two *Star Wars* Death Star turrets show simple rectangular panels that feature decorative line patterns. Note the vertical and horizontal line segments that are connected across the faces of the panels.

4.9.2 Semantic Details

Some surface details are objects with obvious functions. We refer to these surface details as *semantic details* because their placement is related to their function. These surface details need to be placed intelligently with respect to the scale and direction of the ship, as well as in relation to each other. These details should be placed on the spaceship first, and then other nurnies can be added to enhance them. Examples of semantic details include windows, doors, antennas, lights, hatches, decals, ladders, and pipes. Each of these types of detail have separate conventions that dictate how they should be placed.

Unfortunately, due to the scope of this work, it is not possible to include detailed definitions for the behavior of each of these semantic details. To demonstrate the process as it might be applied to other types of semantic details, we include here an in-depth report of the characteristics of windows and pipes. Windows and pipes are easily the more common semantic details, appearing in almost all spaceship designs. Windows are important for conveying the scale and direction of the ship, and pipes contribute to the overall mechanical design by providing connectivity. They also feature logical behavior and have a concise set of rules for describing their layout. Windows and pipes are described in the following two sections.

Windows

Windows are the most common surface detail to appear on spaceships. Windows are important for spaceship design because they allow the viewer to locate the cockpit and provide a clear indicator of scale and direction.

Windows are rarely curved surfaces, and are represented instead as an approximation using flat panels of glass. Originally this configuration was preferred because of the difficulty of constructing a curved piece of glass for a physical model, but it is still the norm.

Window placement is almost always symmetric. For small to medium scale spaceships, this translates to a single, central cockpit window or group of windows. For large scale spaceships, windows are usually represented as horizontal rows of lights along the front and sides of the ship, symbolizing the floors or decks of the ship. These ships may feature any number of rows of lights, and usually do not feature a row for every floor of the ship, just a few to demonstrate the scale. Also, these rows of lights are randomly interspersed, still creating the sense of a complete row of windows, but with certain windows closed or unoccupied. The *Star Wars* Trade Federation Battleship, shown in Figure 4.12, is a huge space station that features these rows of windows. Note that it features various densities of rows of lights, though the scale is consistent. Lights within the rows are also randomly interspersed.

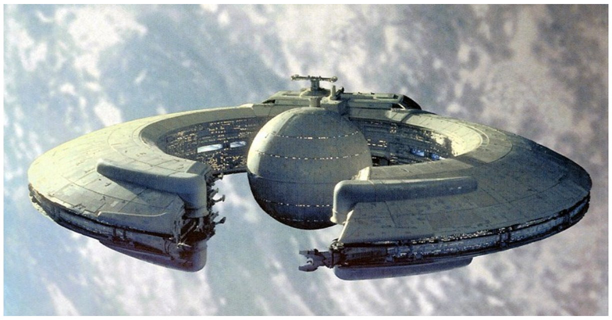


Figure 4.12: The *Star Wars* Trade Federation Battleship is an immense battle station, as evidenced by the tiny rows of light that represent its windows.

Pipes

Pipes are a common feature of mechanical style spaceships. When used properly, they work well to add complexity to the surface of the ship while providing a sense of intelligent design by connecting different components of the ship. Pipes tend to appear in conjunction with other surface details to create a sense of connectivity.

There are some common conventions for pipe layout on spaceships, most of which arise from how pipes behave in reality. While it is impossible to say how pipes might function in an unknown technology, these conventions convey plausibility by tying the current design to both past spaceship designs and knowledge of logical pipe behavior.

Pipes tend to be placed in areas with a high density of nurnies. This is especially important for the ends of the pipes; they should begin and end at areas with a high density of nurnies, but may be used to connect two disjoint areas with a low density of nurnies. Logically, pipes are used to connect areas of interest.

Pipes tend to have duplicate paths. Pipes will either merge with an existing path of pipes or mimic a nearby path while remaining separate. Pipes are designed this way in reality for various reasons: First, by joining nearby pipes that are carrying the same fluid, less overall piping is needed to distribute the fluid. By merging nearby pipes that carry different liquids, creating a group of pipes that follow a similar path, it is easier to both insulate the pipes and segregate them if needed [59]. This principle is shown in Figure 4.13.

Groups of pipes tend to connect or meet at a single central location. They appear to branch away from a single source because pipes are used to distribute liquid from a central location or store to other positions in the area [59]. This principle is shown in Figure 4.14.

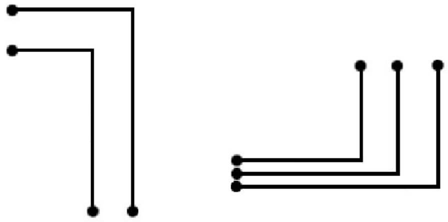


Figure 4.13: Pipes tend to mimic or merge with other paths of pipes.

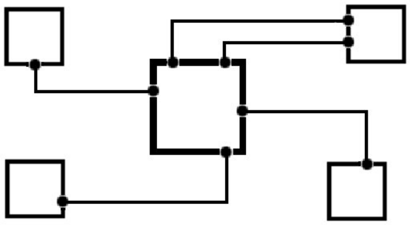


Figure 4.14: Pipes tend to meet at a common source.

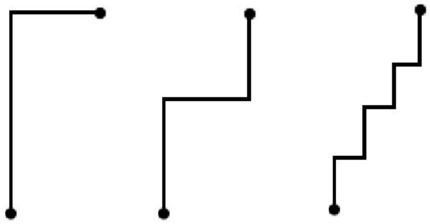


Figure 4.15: Pipes avoid bending whenever possible. These configurations are ordered by preference from left to right. The left configuration is ideal, the middle configuration is nearly ideal, and the right configuration is avoided.

Figure 4.16: Pipes may deviate slightly from the optimal path.

Pipes prefer to travel in a single direction for as long as possible. Straight pipes allow for better flow of the fluid contained within them, as bends in the pipes will increase friction and slow the fluid [59]. Examples of preferable pipe layouts are shown in Figure 4.15.

Pipes may deviate slightly from the optimal path for no obvious reason. Adding slight imperfections to a pipe's path adds an element of realism. While there may be no visible reason for the deviation, a pipe may be forced to avoid a certain area because of extreme temperatures or high voltages [59]. This principle is demonstrated in Figure 4.16.

4.10 Summary

This chapter presented high level principles that combine to give an object the appearance of a spaceship. We first emphasized the role of the viewer's expectations in determining desirable features of spaceship design. We then differentiated between the two fundamental types of spaceship, organic and mechanical, and discussed the significance of symmetry in spaceship design. We also discussed how properties of the frame and surface details are used to create the appearance of scale and direction, advanced technology, personality, and plausibility.

The principles described in this chapter were derived from the survey of popular spaceship designs described in Appendix A. For an analysis of these principles for each spaceship design surveyed, see Appendix B.

Next, we discussed design conventions for spaceship frame and surface details individually. We presented observations about important features of the frame, such as a strong silhouette, popular shapes, and common real world influences. However, it is not clear how to combine these features to generate a spaceship frame automatically. We also discussed our selected functional components, the cockpit, wings, weapons, and propulsion, and defining characteristics for each. Finally, we analysed the behavior of three categories of surface details: panels, windows, and pipes.

Chapter 5

Implementation

This chapter discusses the details of our prototype program to procedurally generate spaceship surface details. This prototype was developed in response to the aesthetic and spaceship-specific principles discovered from our survey and discussed in Chapters 3 and 4. We selected procedural modeling techniques that were judged to be most appropriate for spaceship generation.

The implementation was done in C++ using OpenGL. The GLUI library[1] was used for the user interface and the GLM library[2] was used for importing and exporting OBJ files.

Because of the open-ended nature of this work, there are various techniques that we explored via trial and error to determine their applicability to procedural modeling of spaceships. To narrow the scope of this project, we ended up selecting a specific subset of spaceship features for the implementation of this thesis. However, for completeness, we later discuss areas that we examined that were not implemented, either because they were determined to be inappropriate, or because it was not clear how they could be implemented. These areas are discussed in Chapter 6.

5.1 Problem Definition

To narrow the scope of this work to a manageable focus, we selected a subset of the spaceship generation problem for implementation. First, we assume that a description of the frame is given by the user. The input method for the frame is described in Section 5.2. We do not attempt to procedurally generate the frame for a few reasons: first, the designer is more likely to desire artistic control over the frame, as it is the largest and most iconic portion of the spaceship. Second, it is not clear how to generate automatically a pleasing spaceship frame, since it is difficult to separate the aesthetics of the frame from the influence of the surface details. Surface details can create the appearance of a spaceship even when the frame is a simple geometric primitive, which suggests that they are more important to spaceship creation. We discuss a few possible future considerations for frame generation in Section 6.4.1. Finally, since the primary focus of this research is the aesthetics of surface details, we decided to make this the focus of the implementation. Deliberately restricting our focus to a simple frame allows us to scrutinize

---

[1] http://glui.sourceforge.net/

[2] http://glm.g-truc.net/

the resulting surface details without being distracted by the complexity or aesthetics of the frame. This restriction also allows us to better compare our surface detail generation method with existing methods, such as the greeble plug-ins mentioned in Section 2.1.2, and make a fairer judgement of the success of our technique.

This implementation focuses solely on mechanical spaceship designs. The inherent differences between these two types of designs forced us to choose one or the other, and since mechanical spaceships are much more common, we opted for this type of design. We also assume that the frame is bilaterally symmetric, both because this type of symmetry is important for aesthetics and because the majority of spaceships feature this type of symmetry.

We also decided to focus solely on surface details for large scale spaceships. As stated above, we are primarily interested in the aesthetics of surface details, and large scale spaceships generally feature dense nurnies with few semantic details, while small scale spaceships feature mostly semantic details and few nurnies. Large scale ships also do not require as much consideration of scale and direction cues.

While the polygon count of our output is kept within a reasonable range, the models we generate may be sufficiently complex to cause performance problems in real-time applications. More work will be required to reduce this complexity, or mitigate it using level-of-detail techniques, so that these models may be incorporated in real-time applications.

The primary goal of this implementation was to demonstrate the effectiveness of procedural modeling techniques that we had determined to be best suited to spaceship generation, given the aesthetic principles that were discussed in Chapter 3. We also wish to demonstrate that simple improvements that abide by the characteristics of spaceships can have a large, positive impact on results. Finally, we wish to show that incorporating kitbashing into model creation can be used to easily create a technological design.

5.2 Frame Implementation

Our frame implementation supports a set of rectangles of any sizes and orientations. For our tests, we use a simple rectangular model that simulates the Death Star trench.

The frame is represented as a graph, such that the vertices of the graph (which we refer to as “nodes”) are the rectangles, and an edge between two nodes implies that these two rectangles are connected. Each node of the graph has its own translation, scale, and rotation values. This information for each node is specified in a formatted text file that forms the input to the program.

Instead of asking the user to define the ship’s axis of symmetry, up direction, and forward direction, we assign these to be the x-axis, the positive z-direction, and the positive x-direction, respectively. The user must keep these constants in mind when specifying the locations and orientations of the functional components, as we use these values in determining symmetry. If the frame does not adhere to these constants, then functional components are processed individually and symmetry is not incorporated into the design.

5.3 Surface Details

Surface details are created in three stages: panels, nurnies, and pipes. Panels are created first, as they are used later both to contain nurnies and to guide the creation of pipes. Panels are created in two stages: grid panels, shown in Figure 5.13(a) and subpanels, shown in Figure 5.13(b). Subpanels are created recursively and stored at each level of recursion to facilitate the creation of special panels, which is discussed in Section 5.3.2. Next, nurnies are recursively added to the final panels depending on the desired level of detail. A base primitive is assigned to each panel, starting with cuboids. Wedges are added next to complement the cuboids. Remainging panels are assigned either another primitive or a model from the kitbashing library. Shape grammar rules are then applied to each panel, using the base primitive as a starting point. Finally, pipes are added along the edges of the panels. The steps of the panel and geometry creation processes are shown in Figure 5.13.

Since surface details rely to some degree on symmetry, nodes are processed differently depending on whether they are self-symmetric (i.e., they are bisected by our predefined axis of bilateral symmetry) or paired with a mirror image on the opposite side of the symmetry axis. Internally, nodes that are self-symmetric are split in two along the axis of symmetry and then processed as two separate nodes, so that they can be treated the same as paired symmetric nodes.

5.3.1 Panels

Since the frame consists of rectangles, we create rectangular panels. Panel layout provides the underlying order for the entire frame; since geometry is generally contained within panels, the arrangement of geometry is constrained by the layout of panels. Because of this, panels must exhibit some level of symmetry, following the observation that larger surface details are more likely to be symmetric than smaller surface details.

We create panels by recursively splitting each node of the frame. First, we split the original node's rectangle into a regular grid of panels, which we refer to as "grid panels". This step is demonstrated in Figure 5.13(a). This grid allows us to both enforce a high level sense of order that can later be refined, and to facilitate the placement of special symmetric panels, discussed in Section 5.3.2. The user specifies the dimensions of this grid.

Starting from this grid, each panel is split recursively into a set of smaller subpanels. All panels are stored in a tree structure such that each level of the tree represents a single level of recursion. The root of the tree is the original frame node rectangle, the first level of the tree contains the grid panels, and further recursions are stored such that children are subpanels of their parent. Not all panels are split at each level of the recursion. This leaves some panels with larger, coarser pieces of geometry and other panels with smaller, finer geometry. An example of a panel tree with two levels of splits is shown in Figure 5.1.

We now describe the process of splitting a single panel into subpanels. A single split is created by first selecting the location of the split, by generating a single random point within the panel. To ensure that the subpanels' dimensions are larger than a predefined minimum, we randomly select a split point from a subset of the panel, demonstrated in Figure 5.2. Using this point, we must split the rectangular panel into a set of smaller rectangles. We can achieve this by splitting

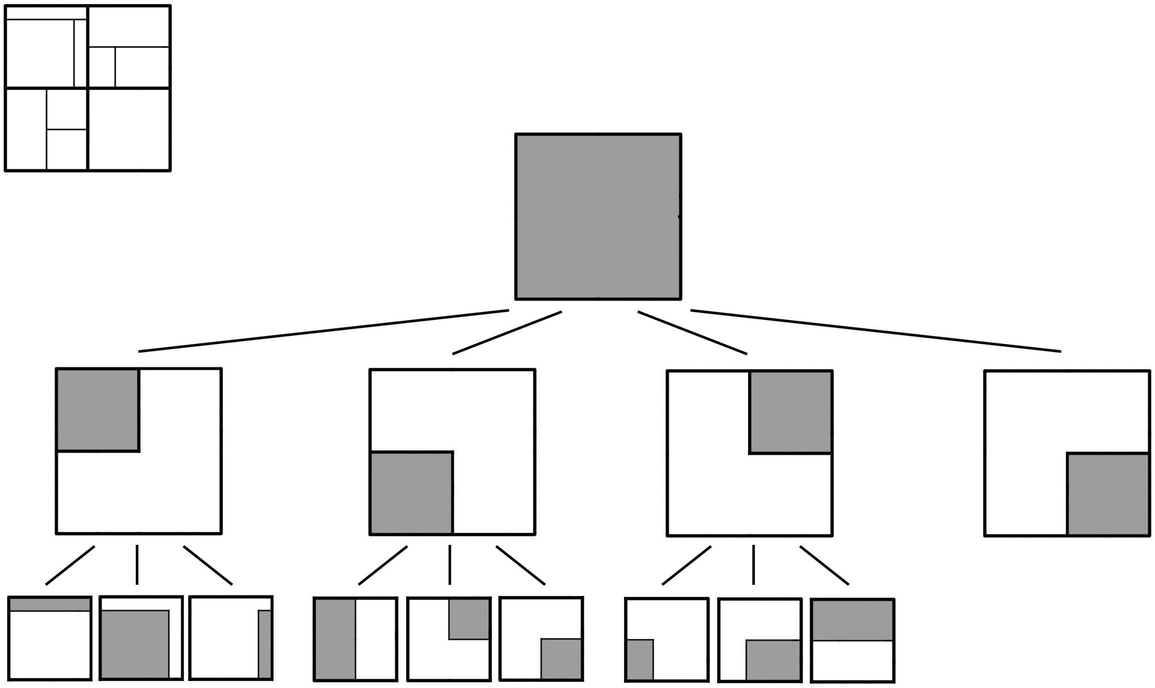


Figure 5.1: A panel tree with two levels of splits. The complete set of panels generated by this tree is displayed in the top left. Note that not every panel is split at each level of the tree.

in two, three, or four directions, along the x- and y-axes. We have chosen to split each point in three directions to encourage variety in the final shape while ensuring that all subpanels will be rectangular. An example showing the four possible split configurations for a single point is shown in Figure 5.3.

While splitting in two or four directions also ensures that all subpanels are rectangular, they also result in a more homogeneous pattern of subpanels, which is undesirable for our intended use of uniformity amidst variety. Assuming that no equalities arise by chance, splitting a panel into three will result in only two subpanels that have one equal dimension (out of a possible six dimensions). If the panel is split in two, two of four subpanel dimensions will be equal; if split in four, each subpanel's dimensions will be equal to another's. Therefore, splitting a panel into three subpanels results in the most heterogeneous set of subpanels. We demonstrate these three types of split in Figure 5.4.

To split a panel into more than three subpanels, we iterate this process. We define the desired number of splits per panel to be N. The original panel is split into three subpanels using the method described above, by randomly selecting a split point. For the following $N-1$ iterations, we select one of the M largest subpanels, based on area, to split, where M is a percentage defined by the user. If the subpanel cannot be split without violating the minimum rectangle size, i.e., both of its dimensions are less than $2min$, it is skipped, and the next largest subpanel, if exists, is selected. If the subpanel can be split in a single direction while maintaining this minimum, then a split point is selected and the subpanel is split into two panels instead of three. This special

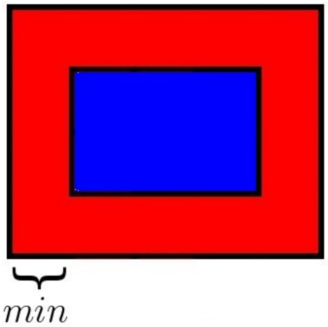


Figure 5.2: In this diagram, the blue center of the rectangle denotes the area from which a split point will be selected. The exterior (red) is not used in the selection process to ensure that new rectangles are larger than a predefined size, specified by *min*.

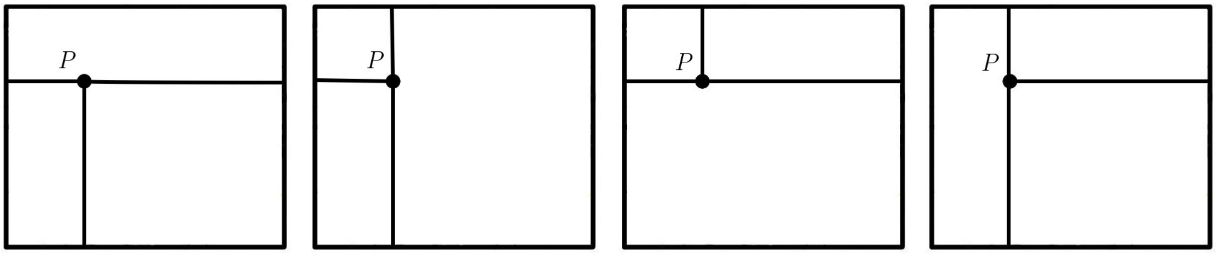
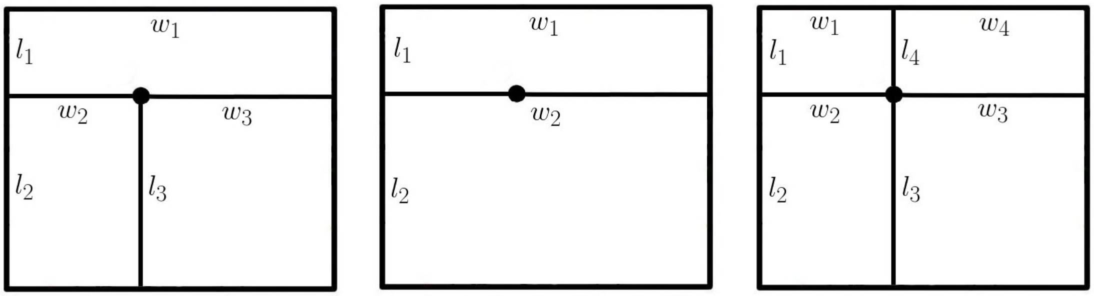


Figure 5.3: This diagram shows the four possible configurations that could be generated from a single split point, P, using our three direction split.



(a) When a panel is split into three subpanels, a minimum of two out of six subpanel dimensions must be equal. In this example, $l_2 = l_3$.

(b) When a panel is split into two subpanels, a minimum of two out of four subpanel dimensions must be equal. In this example, $w_1 = w_2$.

(c) When a panel is split into four subpanels, all subpanel dimensions will be equal to another's. In this example, $w_1 = w_2$, $w_3 = w_4$, $l_1 = l_4$, and $l_2 = l_3$.

Figure 5.4: Splitting panels into two, three, or four subpanels results in varying levels of heterogeneity.

case is demonstrated in Figure 5.5. A simple example result for a panel with four desired splits is shown in Figure 5.6.

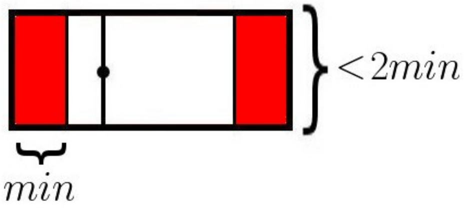


Figure 5.5: This diagram shows the special case of a panel that is split into only two subpanels. This panel has one dimension that is less than $2min$ and thus would produce subpanels that violate the minimum rectangle size if split. Therefore, we select a point and split only along the other dimension.

When creating panels, we process each frame node in conjunction with its symmetric partner. Both nodes are split identically in the first step, so they have the same set of grid panels. This is to allow pattern panels, discussed in Section 5.3.2, to be selected such that they are exactly symmetric. Then, for each further level of recursion, there is a user-specified probability that a given panel will be split symmetrically for both frames. For each future recursion, this probability is halved, so that higher level panels are more likely to be symmetric than lower level panels. Obviously, if we decide that two high level panels will not be split symmetrically, they are not considered in future recursions for the symmetric split.

5.3.2 Special Panels

There are certain locations on the surface of the spaceship that have special behaviors. We refer to these locations as pattern and edge panels. An example of a frame with these panels highlighted is shown in Figure 5.11.

An *edge panel* is a panel that lies on the edge connecting two adjacent frame nodes. An edge panel is found by checking for intersection between this frame node and all nodes that it connects to. Edge panels are examined to determine the angle between the two rectangles of the frame nodes, based on their respective up directions. If the angle is greater than 180°, the edge panels are assigned cubes to create an adjoining edge. If the angle is less than 180°, only the edge panels of one of the frame nodes are assigned cubes; the other edge panels are left blank to prevent the geometry from overlapping.

A *pattern panel* is a panel that has been set aside to be used for geometry that creates a regular pattern that spans the surface of the ship, creating unity across the entire design and providing a degree of bilateral symmetry. Before further splitting steps, grid panels are selected and reserved for pattern elements. We select pattern panels at this step because grid panels are the only panels that are guaranteed to be symmetric for two frame nodes. As discussed in the previous section, further splitting may be done symmetrically with a certain probability, but subpanel symmetry is not certain.

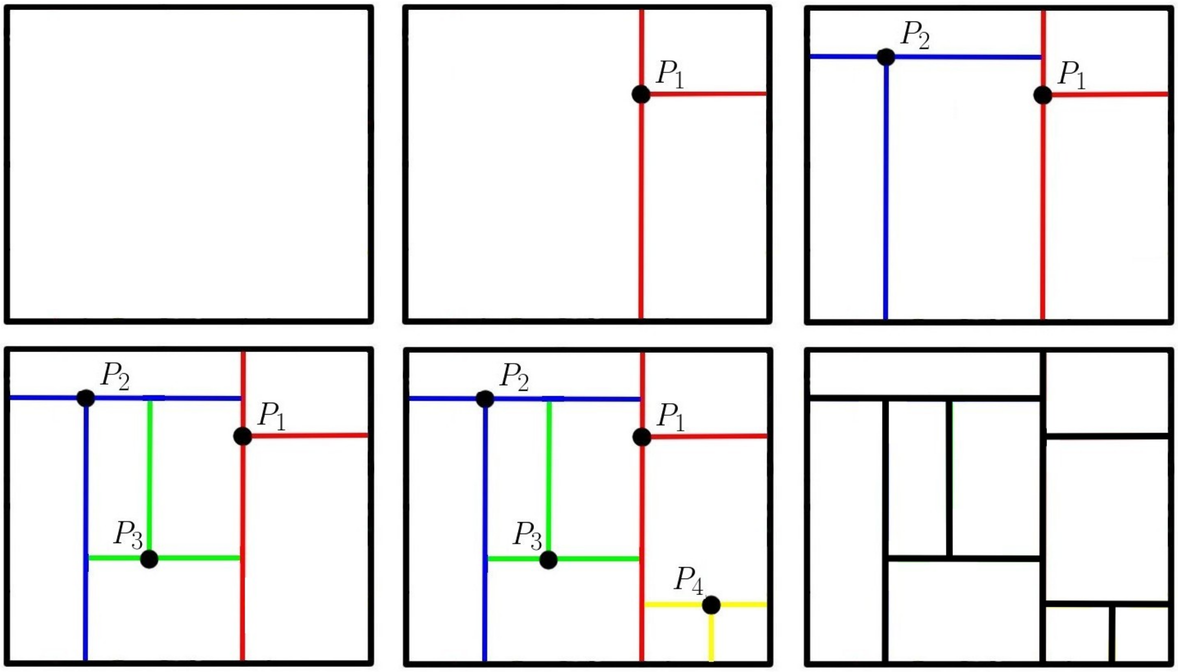


Figure 5.6: This diagram demonstrates the process of panel splitting for four desired splits. For simplicity, we assume that the largest subpanel is always selected to be split. *Top Left*: The original rectangular panel. *Top Middle*: The first split point is selected from the original panel. We use this point, P_1, to split this panel into 3 subpanels. *Top Right*: From these three subpanels, the largest is selected and a second split point, P_2, is generated. We split this subpanel into three new subpanels. *Bottom Left*: The largest subpanel is selected again, and split into three subpanels. *Bottom Middle*: This step is performed a fourth and final time. *Bottom Right*: The final set of nine panels created from four splits.

A pattern panels is determined with the following process. First, we randomly select an integer within a user defined range, usually between one and four. We use this value to select a random n-omino, a connected set of n panels taken from the grid panel divisions of the frame. This n-omino will be repeated in a pattern across the surface of this frame node and its symmetric counterpart.

Since our pattern occurs in a grid, we must decide the axis of repetition (along one of the frame node's two principle axes, parallel to the edges of the node's rectangle) and the number of instances that the pattern will consist of. First, the axis is selected randomly, assuming that the grid size along both axes is large enough to accommodate at least a single copy of our n-omino, otherwise we select whichever is large enough. If neither axis is large enough, no pattern panels are created. Next, we select the number of instances of the n-omino that will occur, assuming an equal spacing between each repetition. We first calculate the maximum number of instances that can fit along our selected axis by dividing the width of the n-omino by the size of the grid, then select a random integer between one and this value. Figure 5.7 shows an example frame node with a regular layout of pattern tiles.

These steps allow us to create only a regular pattern of tiles. To introduce some randomness to the pattern, we allow a given n-omino to be perturbed along either axis with some probability. This probability is determined by the user, and is represented as a a single floating point value between zero and one. A value of one results in total regularity; the pattern is repeated in an exact grid at regular intervals, such as in Figure 5.7. The closer the value gets to zero, the more likely it is that a given instance of the pattern will feature a random deviation, or be skipped entirely. An example of a pattern with these perturbations is shown in Figure 5.8.

Since pattern panels are symmetric across the entire frame, they need be generated only for half of the nodes of the frame, and can then be translated to the appropriate grid panels for the symmetric nodes.

The geometry assigned to the pattern tiles is created using a tile-based system reminiscent of Wang tiles. We use a set of six tile models that, when rotations are included, permit any n-omino to be filled deterministically with geometry. The models are designed to line up perfectly with the others in the set based on the adjoining number of sides and the location in the pattern. These tiles allow the pattern to appear as a single, whole piece of geometry. The tile models were created by hand and are shown in Figure 5.9. An example tile configuration is shown in Figure 5.10. The four sides of a tile may be one of two types: an alignment side (shown in red in Figure 5.9) or an exterior side (shown in black in Figure 5.9). An alignment side must be adjacent to another tile, and an exterior side must not.

5.3.3 Nurnies

In our implementation, all nurnies are contained within panels. Nurnies can be either geometric primitives or polygonal models and are represented as instances of objects. Because of this, we need only store a single instance of the geometry of each primitive and model, so the representation within the program is concise. Only when the geometry is rendered or written to a file are we required to evaluate each instance of nurnie.

First, we assign a base primitive to each panel. This base primitive is later used to determine a set of secondary primitives. Possible base primitives include empty (no primitive), cuboid

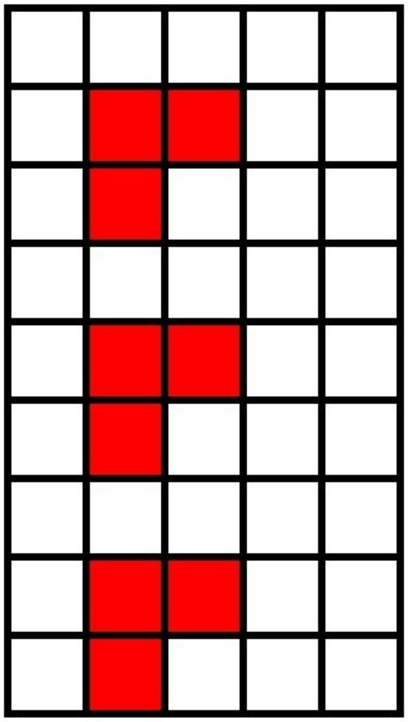


Figure 5.7: An example placement of pattern tiles, highlighted in red. This grid shows a regular placement of a 3-omino. Three repetitions were selected for the pattern, of a possible maximum of four.

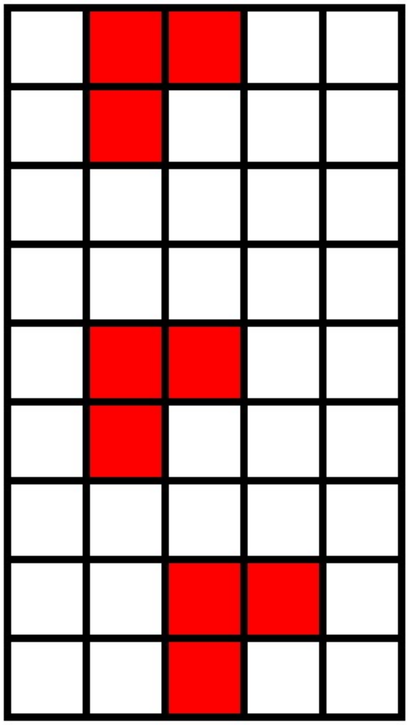


Figure 5.8: An example placement of pattern tiles, highlighted in red. Similar to Figure 5.7, this grid shows a pattern of 3-ominoes. Three repetitions were selected for the pattern, of a possible maximum of four. However, two of the three instances have been slightly perturbed from their original layout.

(i.e., rectangular prism), cylinder, wedge, pyramid, cone, and model. Models are selected from the kitbashing library, discussed later in this section. These pieces are not added completely randomly, but in a specific order to force newly added pieces to complement their neighbours and lend stylistic unity to the finished design.

Cuboids are added to the surface first, since they are the most flexible primitive in terms of later processing, and they work best with the most number of primitives because of their flat top surface. Cuboids are added to random panels and are scaled to fit the length and width of the panel exactly. Cuboid height is selected randomly from a user-defined range except when a neighbouring panel already contains a cuboid; in this case, the new cuboid is scaled to match the height of its neighbour. The percentage of panels that receive cuboids in this step is specified by the user, but we found that thirty percent worked best for this value. The cuboid creation step is shown in Figure 5.13(c).

Next, we add what is called complementary geometry. Complementary geometry consists of primitives that can be made to line up with a cuboid, so that they appear connected. In this case,

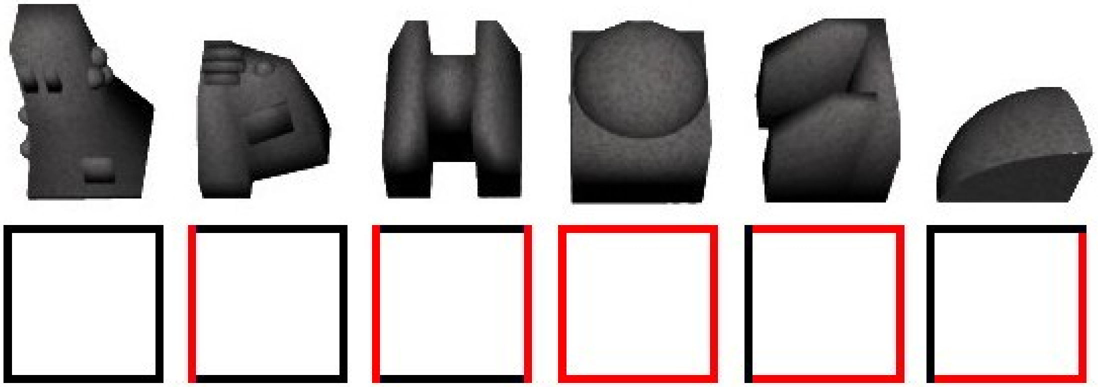


Figure 5.9: The six tiles used for pattern panels. The square below each tile represents how it connects with other tiles in the set. Alignment sides (red) represent the edges of the tile that must align with other tiles, and exterior sides (black) must not. Note that these tiles can be rotated to create any needed pattern configuration.

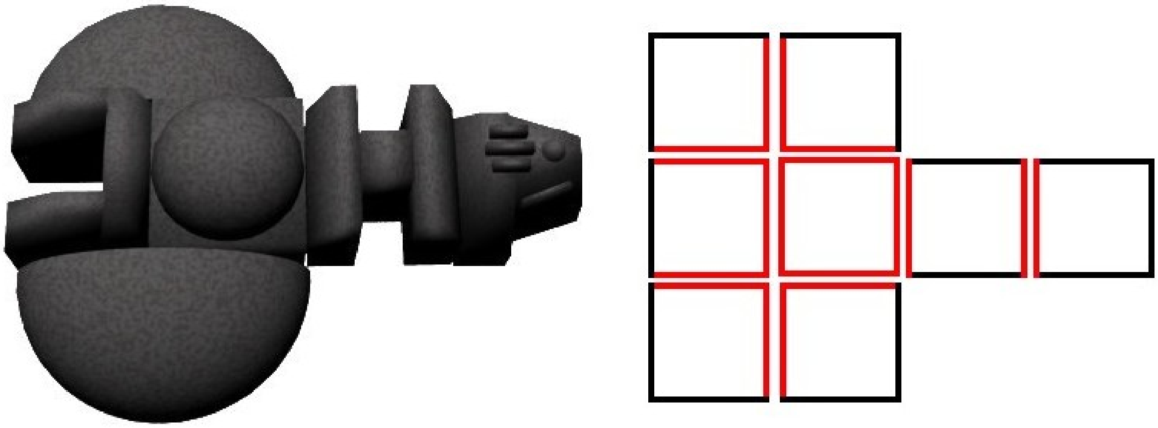


Figure 5.10: An example tile pattern, including the panel configuration that would create this pattern. Note that the pattern panels will not necessarily be square; they may also be rectangular.

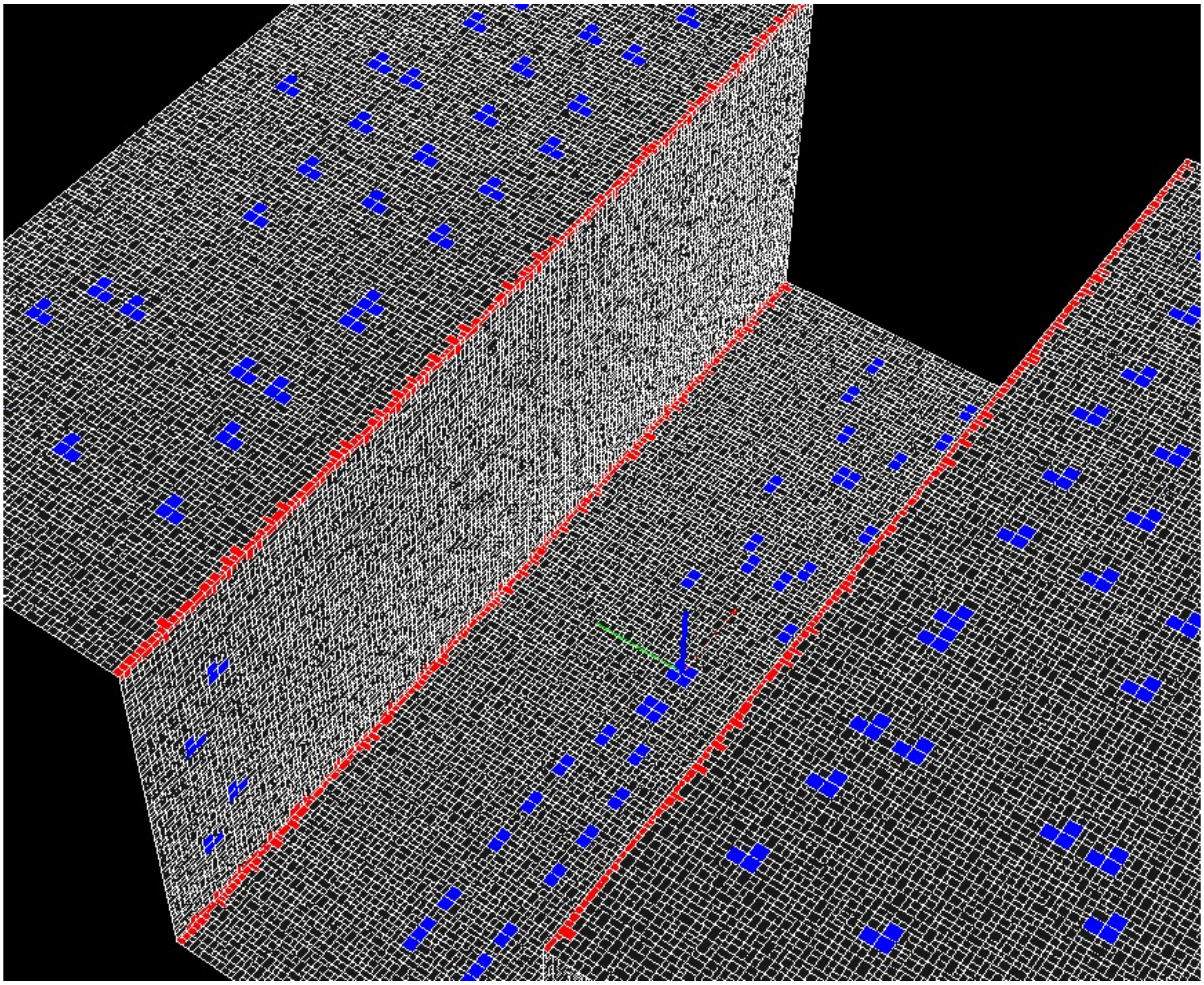


Figure 5.11: A screenshot of the program showing the panels of a generated trench, with pattern panels and edge panels highlighted in blue and red, respectively.

our complementary geometry primitives are wedges. The incorporation of wedges is intended as a prototype for a broader range of inter-panel connectivity that could be implemented in a similar manner. For each panel, we determine if it is adjacent to a cuboid such that the entire face of the wedge can align with the face of the neighbouring cuboid. If so, we assign a wedge to align itself to this cuboid with a certain probability. If this assignment fails, this panel is left empty until the next step. The complementary geometry creation step is shown in Figure 5.13(d).

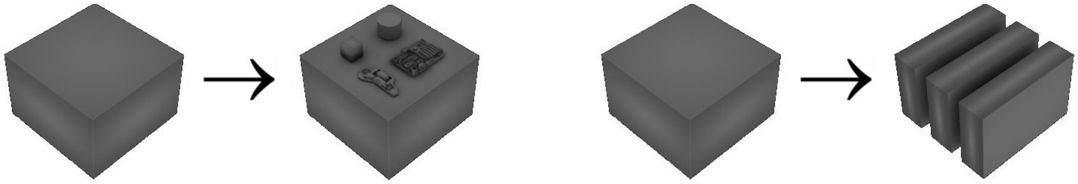
In the final step, we assign a base primitive to every other panel that does not already have one. Each primitive is assigned a probability that determines how likely it is to be selected. If a model is chosen, we select one uniformly at random from the kitbashing library. The base primitive is assigned a height from the user-defined range. This geometry creation step is shown in Figure 5.13(e).

Once each panel has been assigned its base primitive, we then use that primitive to create the panel's secondary geometry. We do this by applying shape grammar rules recursively to the base primitive until the desired level of detail is reached. The secondary geometry creation step is shown in Figure 5.13(f).

In relation to aesthetics, we often mention that plausible, coherent geometry appears to adhere to a set of underlying laws. Shape grammar rules embody these laws and allow us to apply the same rule to different instances to create geometry that appears related without obvious repetition. They also allow the same rule to be applied at a variety of scales to create detail at different levels of granularity.

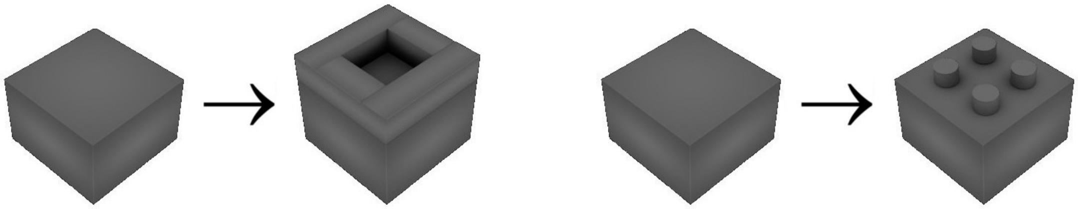
For this implementation, our shape grammar rules are hard coded. Because geometry in our system is represented as primitives, it is easy to perform shape recognition and substitutions. Some simple shape grammar rules we implement include the following. In general, we found that it was best to keep these rules relatively simple, since they may be applied in conjunction with one another. Examples of these shape grammar rules can be seen in Figure 5.12.

- The top surface of a cuboid is treated as a new panel so it can be split and further geometry can be added (Figure 5.12(a)).
- A cuboid is split into a group of smaller cuboids (either by length, width, or to create an open box shape) so they can later be processed individually (Figure 5.12(c)).
- Cylinders are added to the surface of a cuboid in a regular pattern (Figure 5.12(d)).
- Geometry representing screws or bolts are added to the outer boundaries of a cuboid's top face (Figure 5.12(e)).
- A division is added to the surface of a cuboid. Two random points are selected from opposite sides of the top face of the cuboid. We connect these points with three line segments, two that are straight and parallel, and one angled between them. The lengths of the three segments and the orientation of the middle segment are determined randomly (Figure 5.12(f)).
- A cylinder is replaced with a stack of smaller, tiered cylinders (Figure 5.12(g)).



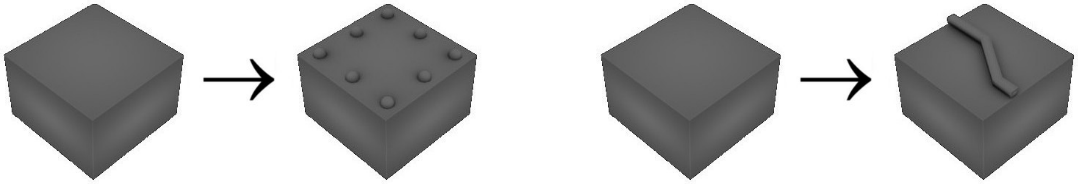
(a) The surface of a cuboid is treated as a new panel, so it can be split and further geometry can be added. Models and primitives were added to this cuboid's surface.

(b) A cuboid is split into a group of smaller cuboids, so they can later be processed individually. In this case, the original cuboid has been split into three.



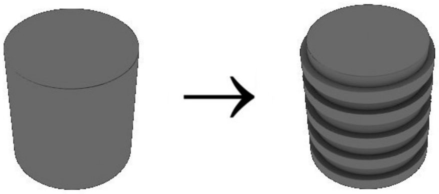
(c) A set of four cuboids are added to the surface of the original cuboid to create an enclosure.

(d) Cylinders are added to the surface of a cuboid in a regular pattern.



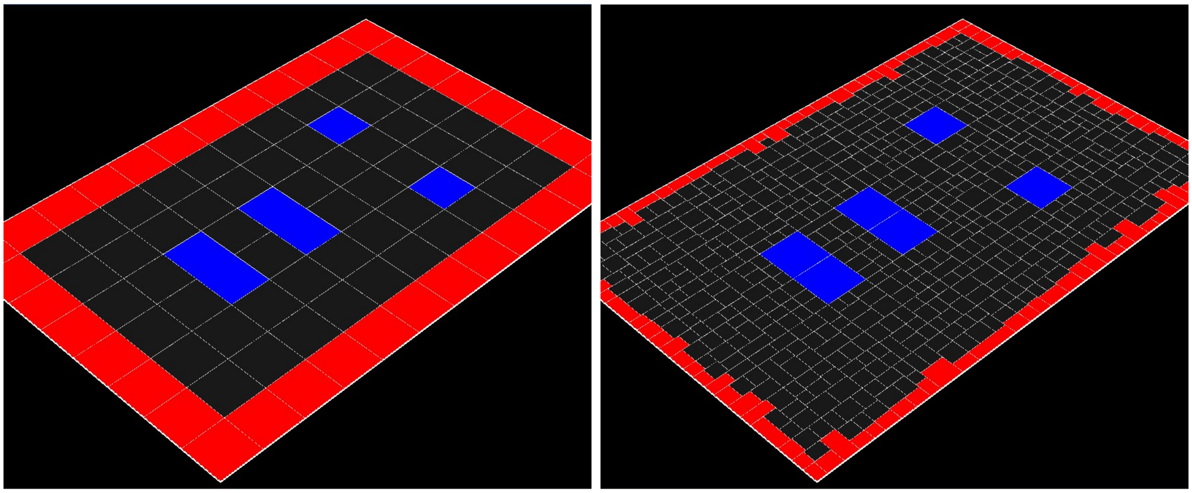
(e) Geometry representing screws or bolts are added to the edges of a cuboid.

(f) A division is added to the surface of a cuboid.



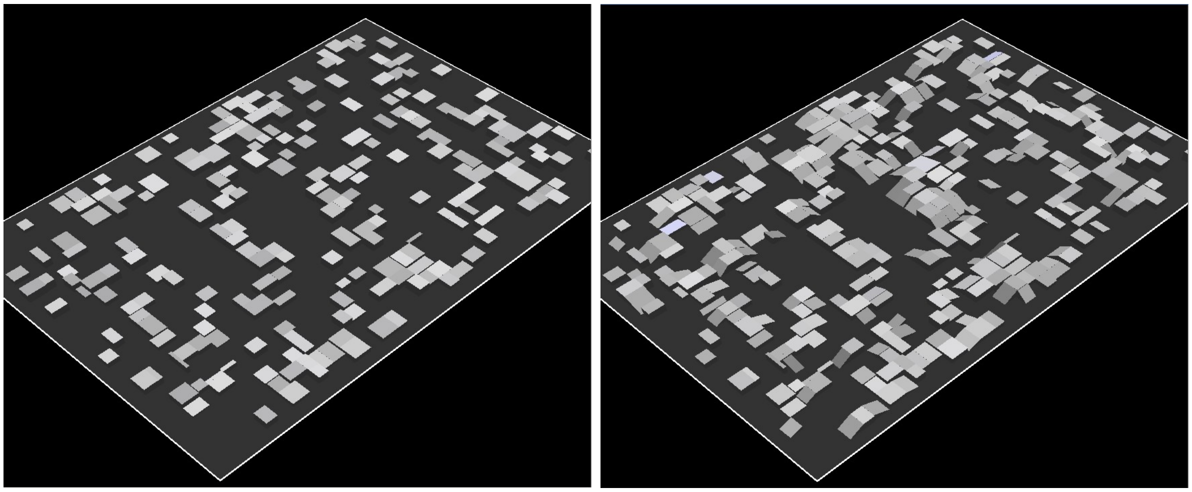
(g) A cylinder is replaced with a stack of smaller, tiered cylinders.

Figure 5.12: Renderings of before and after images of shape grammar rules.



(a) The initial frame rectangle, divided into grid panels. Edge panels are shown in red, and pattern panels are shown in blue.

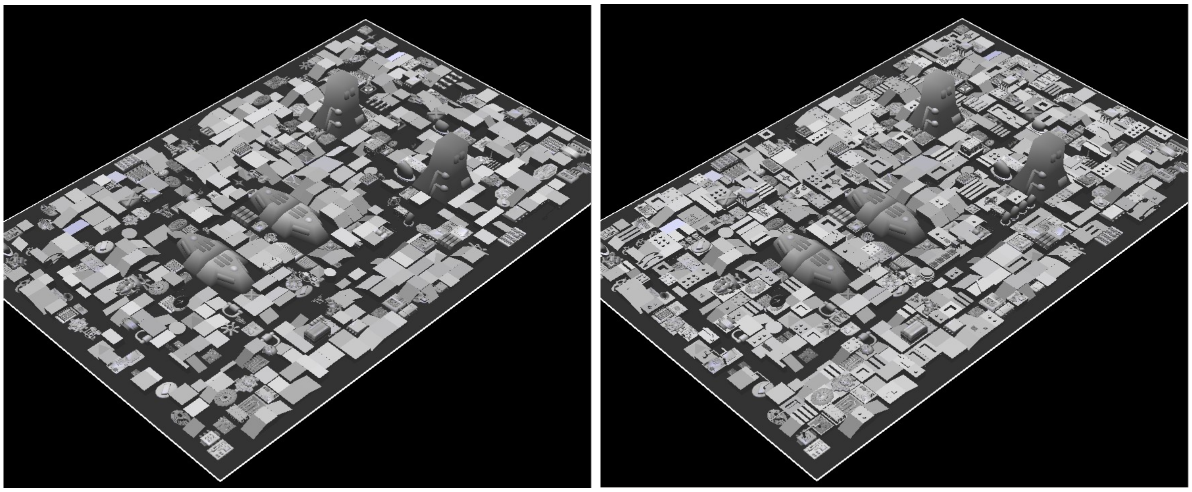
(b) Step 1: Grid panels are divided into subpanels. For simplicity, we performed only a single level of subdivision for each grid panel.



(c) Step 2: Cuboids are added to random panels, and adjust their height to match neighbours.

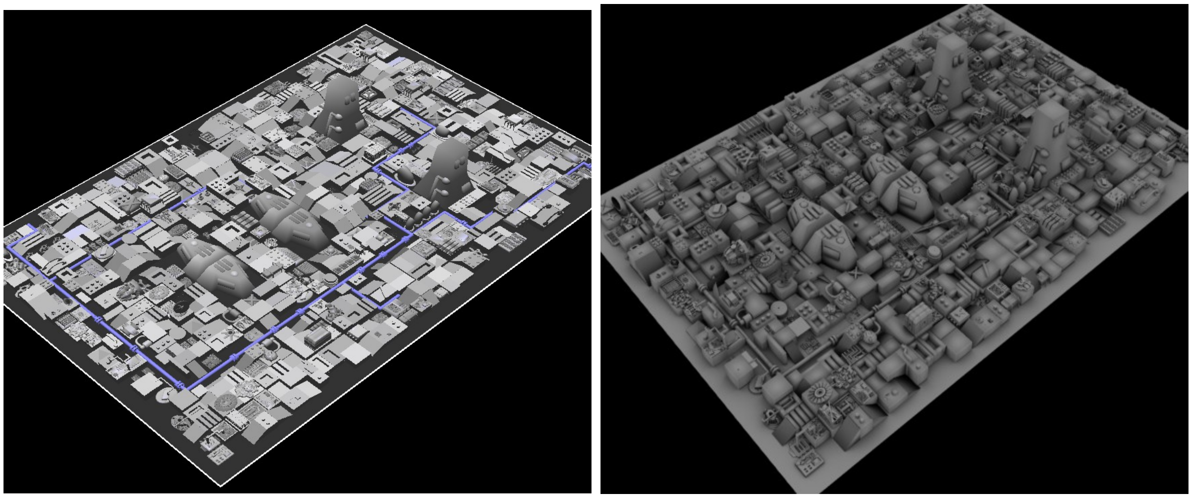
(d) Step 3: Wedges are added beside cuboids and are scaled and rotated to match their neighbours.

Figure 5.13: The major steps of the geometry creation process.



(e) Step 4: The remaining empty panels are assigned random base primitives and pattern panels are assigned tile models.

(f) Step 5: Secondary geometry is added to the base primitives.



(g) Step 6: Pipes are added to the surface.

(h) The final surface.

Figure 5.13: Continued. The major steps of the geometry creation process.

Kitbashing

For the set of kitbashed parts, we used the Houdini Apprentice [86] software package to take apart existing 3D models obtained from online libraries [12, 8]. Other model packs were obtained courtesy of Casper Del Blanco, Mark Kuykendall, and Carl Fredsberg [11].

The collection of parts for the kitbashing library was rather subjective, as parts were evaluated and selected solely at our discretion. However, we found that tanks are easily the best type of model for scavenging parts, as they tend to be covered in an assortment of complex weapons and mechanical-looking parts. Other useful types of models are four-wheelers, trains, helicopters, mech robots, internal car parts, weapons, internal computer parts, electronics, and spaceships. Car and plane models are generally not useful unless they include interior parts, as their exteriors are generally smooth with few mechanical-looking pieces. The kitbashing library contains 70 models, and is shown in Figure 5.14.

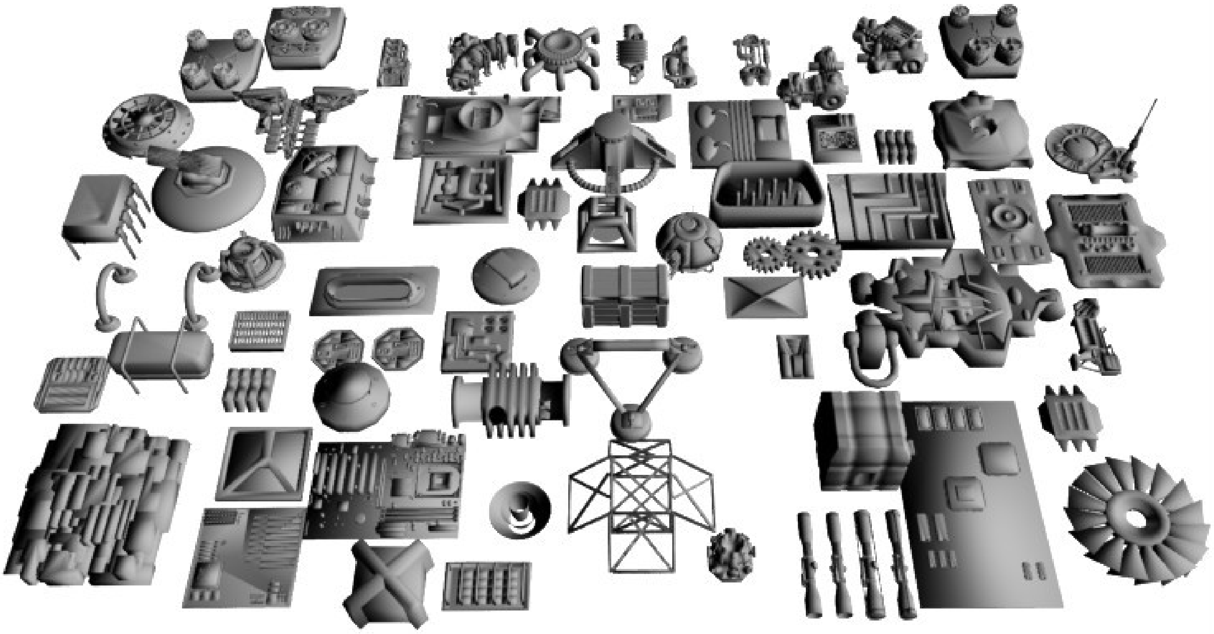


Figure 5.14: The set of models used for our kitbashing library.

For general kitbashing, it is best to avoid shapes that are recognizable as having a specific purpose, as they can look out of place in the wrong context. We focus on shapes with simple geometry, as complex shapes can look unnatural when used in conjunction with simple primitives and are more prone to distortion when scaled to different proportions. It is also important to select pieces that have a flat bottom and have a roughly square footprint.

Models placed within a panel are scaled to fit the size of the panel, except when the proportions of the panel exceed a ratio of 1:2. In this case, multiple copies of the model are placed within the panel to best fit the size of the panel while introducing the least amount of distortion to the model. Not only does this minimize distortion, but it creates clusters of repeated models.

Another important consideration was the creation of connections between models. Kitbashing relies on creating a collage of connected elements. In traditional kitbashing, this collage is created by placing elements strategically so that they appear to connect or line up. However, with a set of arbitrarily shaped models, determining features of models to logically place them to appear connected would be difficult. Hence, we determined that it would make sense to instead add geometry to connect nearby models. The ideal way to connect nearby models would be to identify pieces on each model that could function as connectors and extend these pieces to create a connection. However, since it is not clear how to determine what types of geometry would constitute a possible connector on an arbitrary model, we opted instead to connect models by identifying a flat area on the top of each model (determined by querying the values of the normals) and connecting these areas with pipes. The pipes are created by selecting a height that is greater than the taller of the two models (to avoid intersecting geometry) and creating a path at this height to the edge that connects the two panels. An example of model connector geometry can be seen in Figure 5.15. Since we require kitbashed parts to be extracted and added manually to the library, it may be practical in the future to also require that the user manually identify connector points on the model.

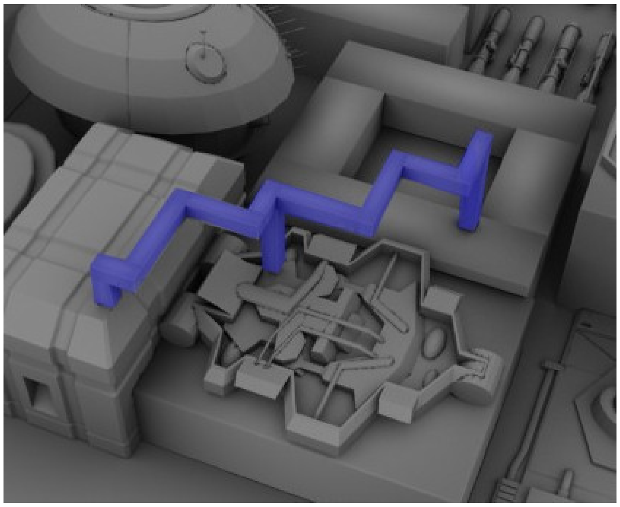


Figure 5.15: An image of the model connection geometry joining three adjacent models. The added connector geometry is highlighted in blue.

5.3.4 Pipes

The final type of surface detail generated is the pipes. Symmetry is not enforced for pipe layout, and pipes are created independently for each functional component. The user is able to specify a range for the number of desired pipe paths, but this is the only control provided for their generation.

Pipes and panel geometry must share the same space, and so we face the decision of which one to place first, later adjusting the second to avoid interference. In Section 4.9.2, we advised adding semantic details (such as pipes) first, as their placement is usually unrelated to the layout of nurnies, but since pipes are used to connect nurnies, it is equally valid to create them after and constrain them to behave according to the layout of the nurnies. The benefit of adding them second is that, by forcing them to avoid interfering with connected nurnies, we naturally introduce slight deviations from the optimal path. Unexplained deviations from the optimal path is one of our pipe behavior principles desribed in Section 4.9.2, and helps create more interesting pipe paths. Were the pipes to be added first, these irregularities would have to be added artificially.

Pipes can only travel along the edges of the panels, not through panels. The pipes' layout is generated automatically by first creating a doubly-connected edge list, where the vertices represent panel edges, the edges represent connections between these panel edges, and the faces represent the panels.

Recall that panels are created by recursively splitting rectangles, and that panels are stored in a tree structure such that each level of the tree represents a single level of recursion.

To start, we construct a naive version of the list such that for each panel that is a leaf node of the panel tree (i.e., each panel that contains no further subpanels) we add a vertex to the list for each of the four sides of its rectangle and an edge between vertices that are connected. Because panels that have the same parent are aware if they are neighbours, we can ensure that redundant vertices are not added for edges that overlap, and ensure that vertices are properly connected in the list. However, while each doubly-connected edge list is complete for this lowest level of subpanels, we now have a set of disconnected edge lists that need to be merged. We must consider the panel edges that lie along the edges of the parent panels, as these panel edges will overlap with panel edges of adjacent parent panels. A diagram showing these overlapping edges is presented in Figure 5.16.

To merge these lists, we must the consolidate the vertices that represent panel edges that are not unique by computing the overlay of two planar subdivisions, where each planar subdivision is contained within a rectangle. For our simplified version of the problem, these subdivisions only meet along one common boundary, and the vertices that lie on this boundary are known. We merge the two lists together by inserting new vertices or updating existing vertices where two panel edges overlap. Using Figure 5.16 as an example, we see that vertices a_3 and b_1 represent two panel edges that overlap; a_3 is a subset of b_1. If we consider two panel edges at a time, there are three types of overlap that can occur: one panel edge is a subset of the other and they share an endpoint, one panel edge is a subset of the other and they do not share an endpoint, and the two panel edges overlap without one being a subset of the other. These three types of overlaps are demonstrated in Figure 5.17. These cases are considered separately because they result in a different number of vertices being created in the list, and because they require different methods of transferring the edges of the two original vertices to the newly created vertices.

This process is repeated until all doubly-connected edge lists are merged for this level of the panel tree. If there are more levels in the tree of panels (other than the tree's root), then this process is repeated until all vertices have been merged into a single connected list with unique vertices.

Once the list is complete, we must select the paths that the pipes will follow. Since pipes tend to branch out from a common source, we create the pipe paths by randomly selecting a source point and a set of destination points. The number of destination points is randomly generated from a user defined interval. We then create the pipe paths by connecting each destination point to the source point.

The path of each pipe is determined by calculating the minimum weight path with Dijkstra's

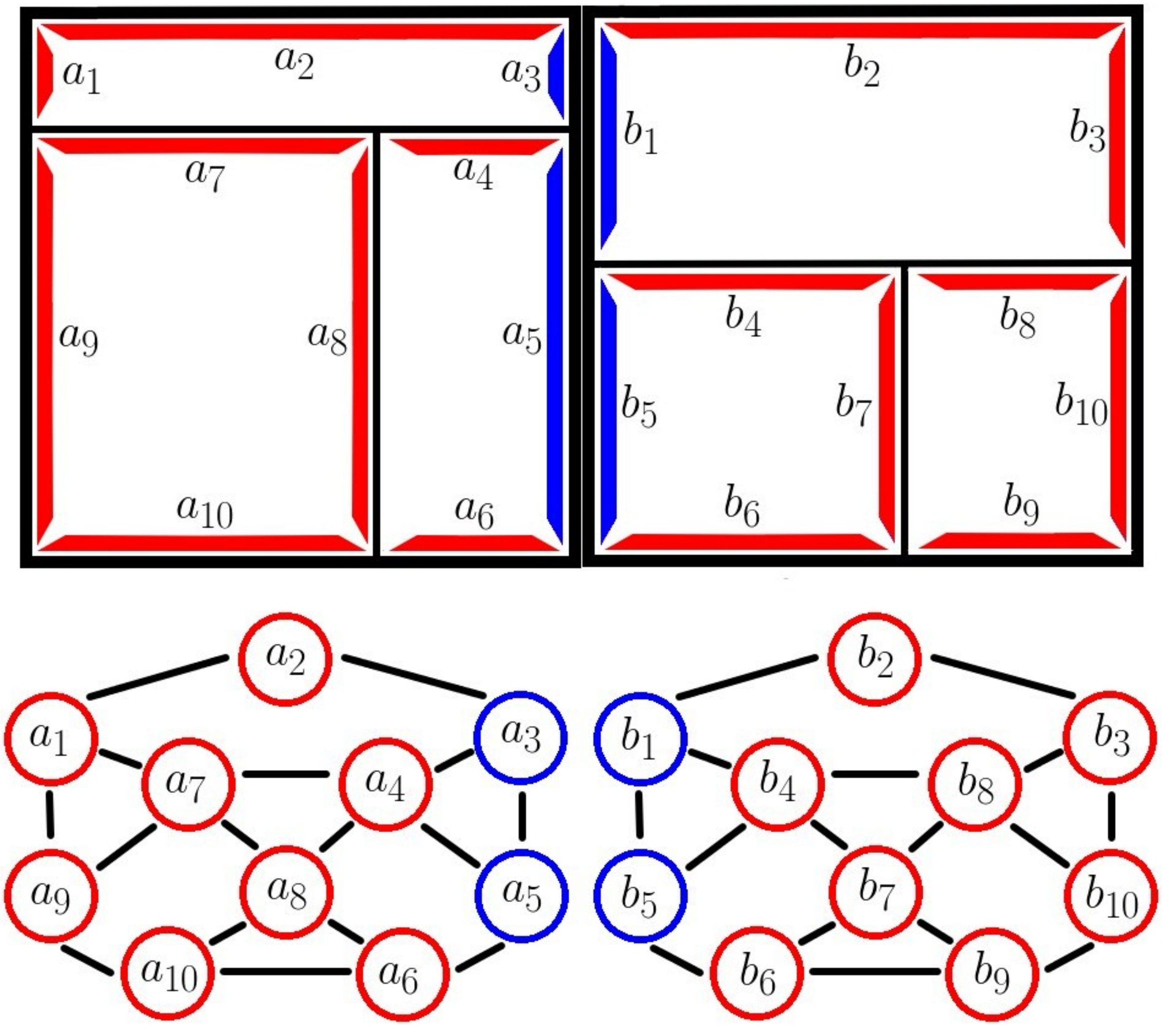
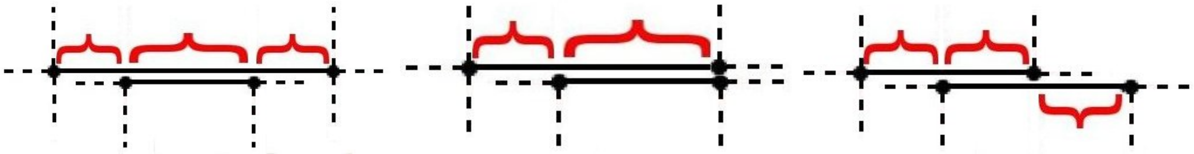


Figure 5.16: This diagram shows an example of a simple frame node split with two levels of recursion, once into a regular grid of two parents, and then again so that each parent has three children. The set of colored edges shown are the vertices of the naive lists, shown below. We need to merge these lists together by consolidating the blue vertices. Vertices are labeled in no particular order.



(a) Case 1: One edge is a subset of the other, resulting in three vertices.

(b) Case 2: The two edges share an endpoint, resulting in two vertices.

(c) Case 3: The two edges overlap, resulting in three vertices.

Figure 5.17: The three cases for merging edges of adjacent panels. The two solid lines are the original edges, and the dotted lines are the locations of possible adjacent edges of the original lines. The red brace indicates the newly formed edges.

algorithm. Following from the rules described in Section 4.9.2, the weight of a path is affected by the following factors. First, each vertex is assigned an initial weight equal to its length. When selecting the next vertex to travel to, an additional weight is added if this vertex is perpendicular to the current vertex. This weight forces pipes to prefer the path with the fewest number of turns. An effective weight of infinity is added for vertices that separate two connected pieces of geometry so that pipes will avoid separating connected geometry unless no other path exists. Similarly, vertices that are adjacent to pattern or edge panels also receive this weight penalty, so that the pipes do not interfere with their geometry.

When a complete path is created, the vertices used in this path are all given a slight reduction in weight in the original list. This provides incentive for future pipes to merge with existing paths.

When all the paths are created, there are two final steps. First, we create the geometry for each path of pipes, including connectors between the sections of pipe and randomly placed details. These details are added with a certain probability to sections of pipe that are long enough to accommodate them. Second, we make room for this geometry to fit along the panel edges by scaling down and translating the geometry of the panels that are adjacent to this edge. This is easily accomplished by storing pointers to the two panels that are adjacent to each edge.

5.4 Output

Our program allows the user to write all geometry to an OBJ file. We use this feature to render results in a 3D modeling program. The following images are screenshots from our program or renderings of our program's geometry that was first written to a file, then rendered using the Houdini software package. The program uses default parameter values (listed in Appendix C) unless otherwise noted.

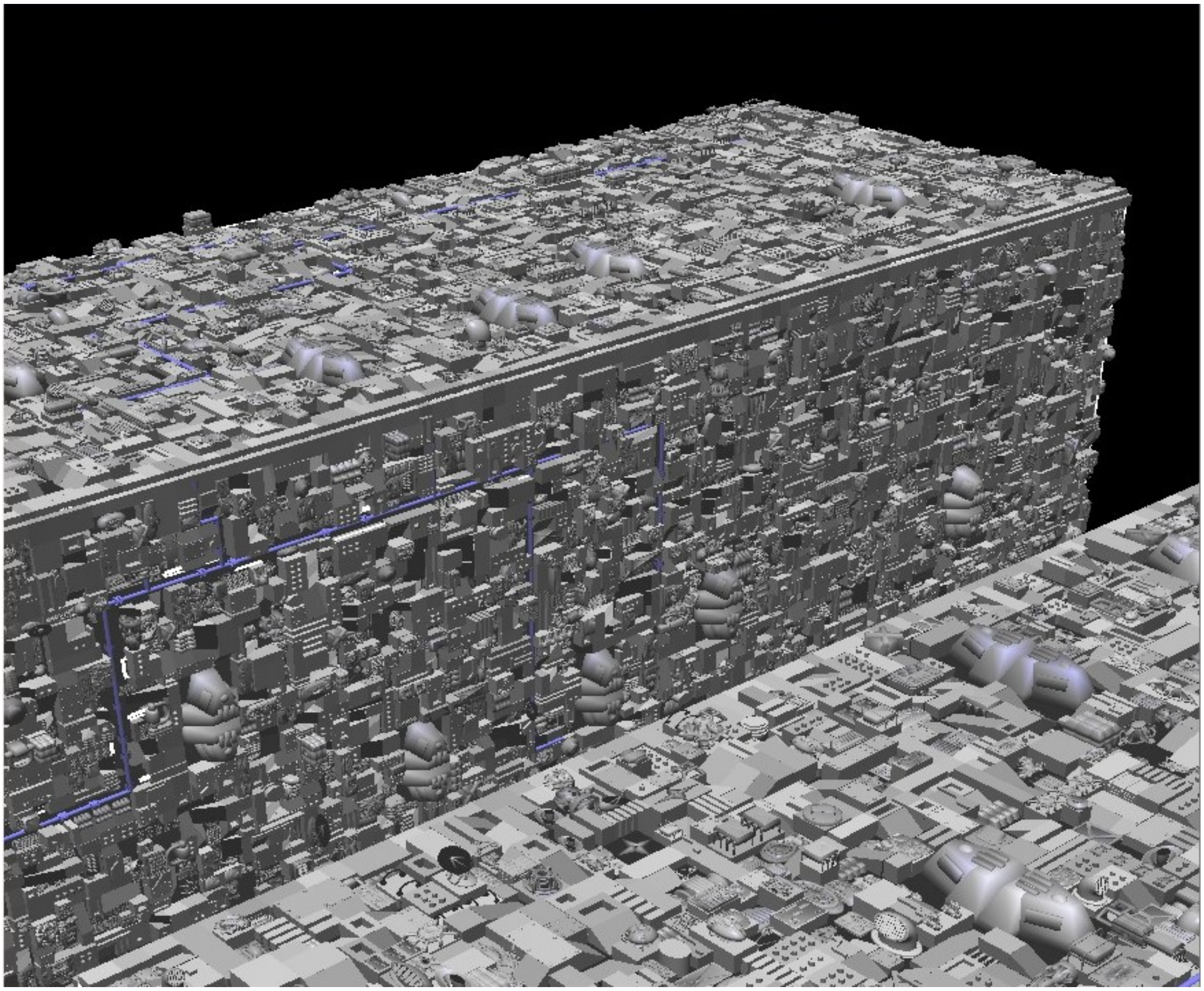
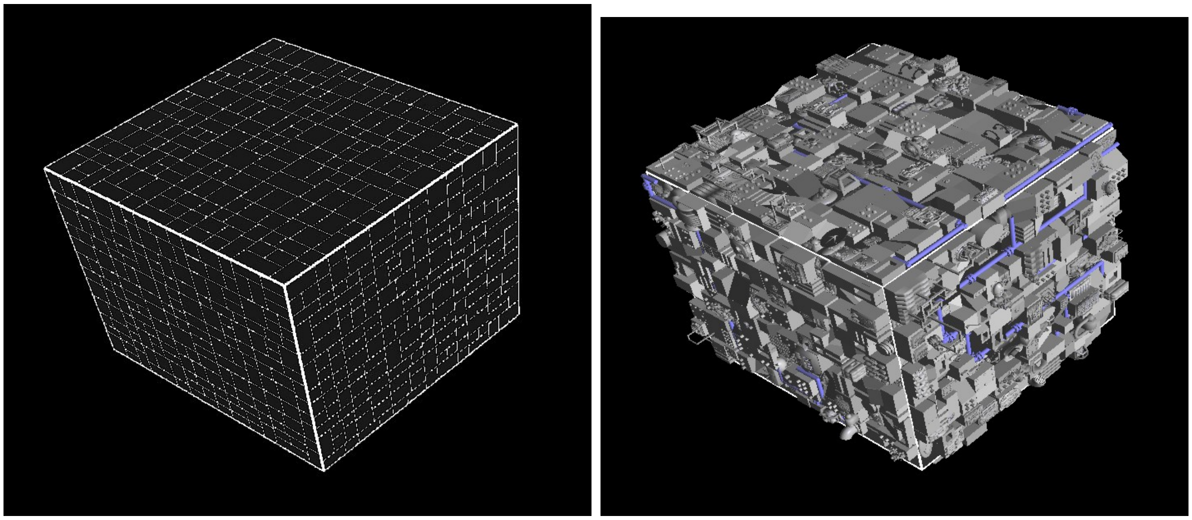
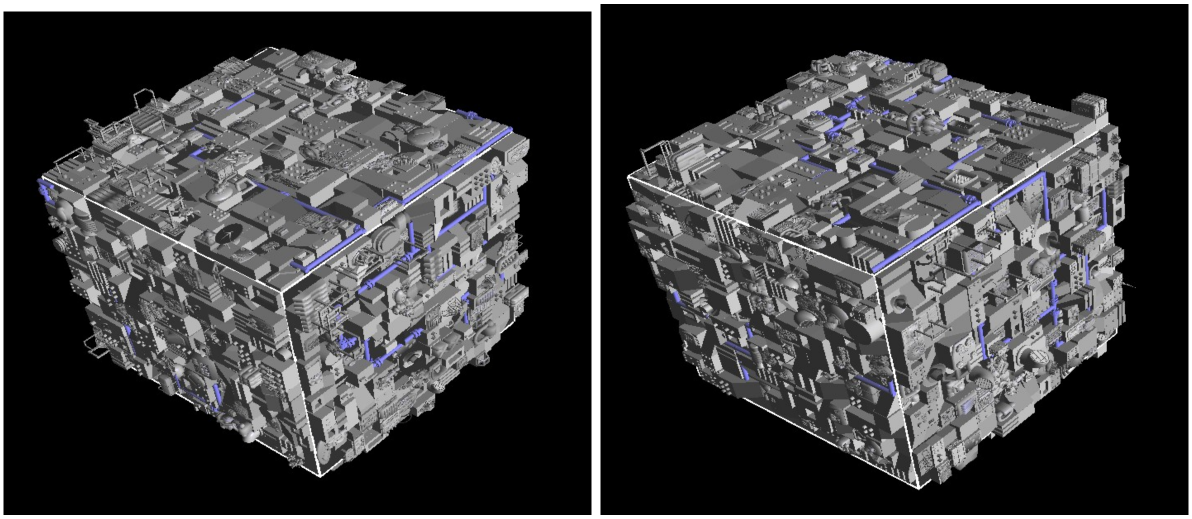


Figure 5.18: A screenshot of a trench rendered in the program. Pipes are highlighted in blue for visibility.



(a) The original cube frame, split into panels.

(b) The frame with geometry applied to the panels from Figure 5.19(a).



(c) The frame with the same panles and base geometry as Figure 5.19(b), but with different secondary geometry.

(d) The frame with the same panels as Figure 5.19(a), but different base geometry (and consequently different secondary geometry).

Figure 5.19: A cube frame with the same set of panels processed with different geomtry.

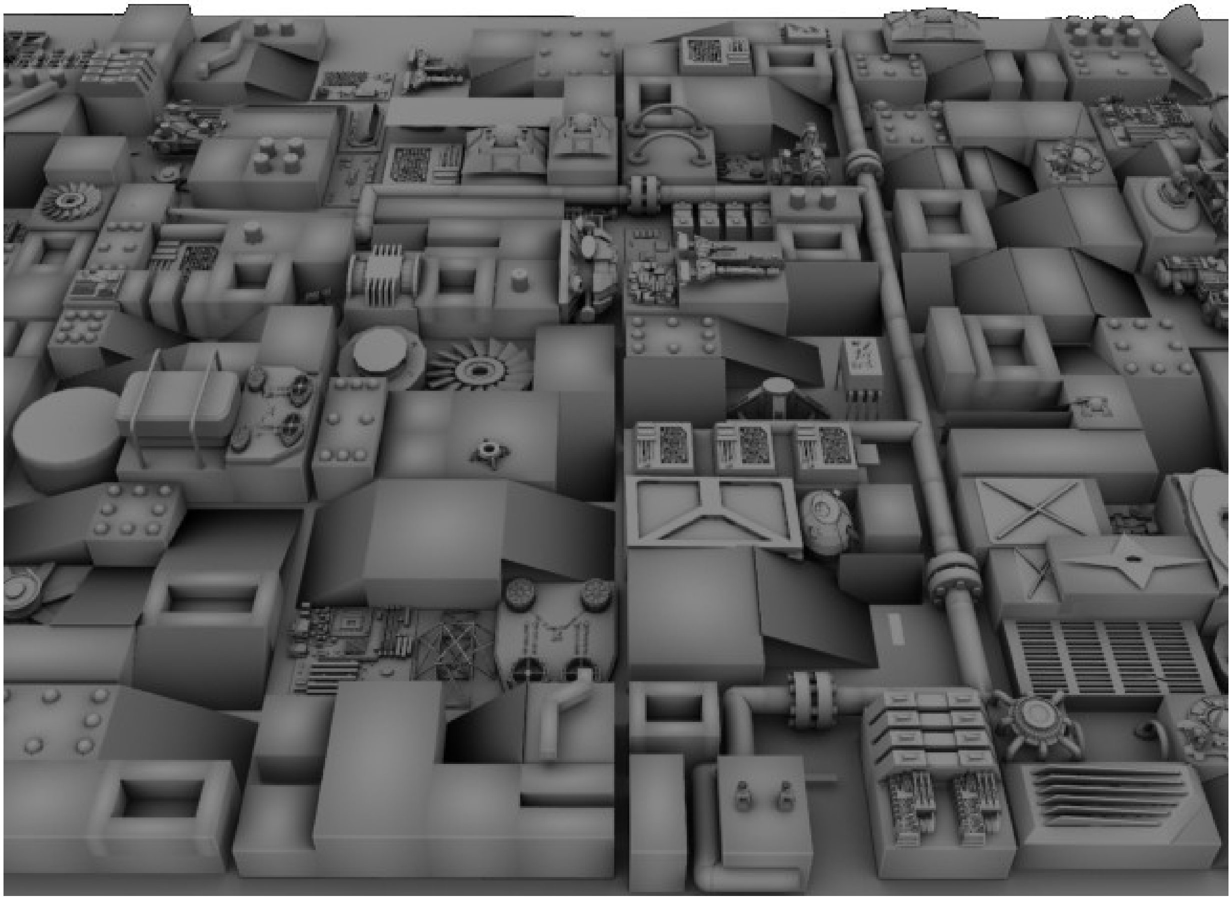


Figure 5.20: A close-up of the surface.

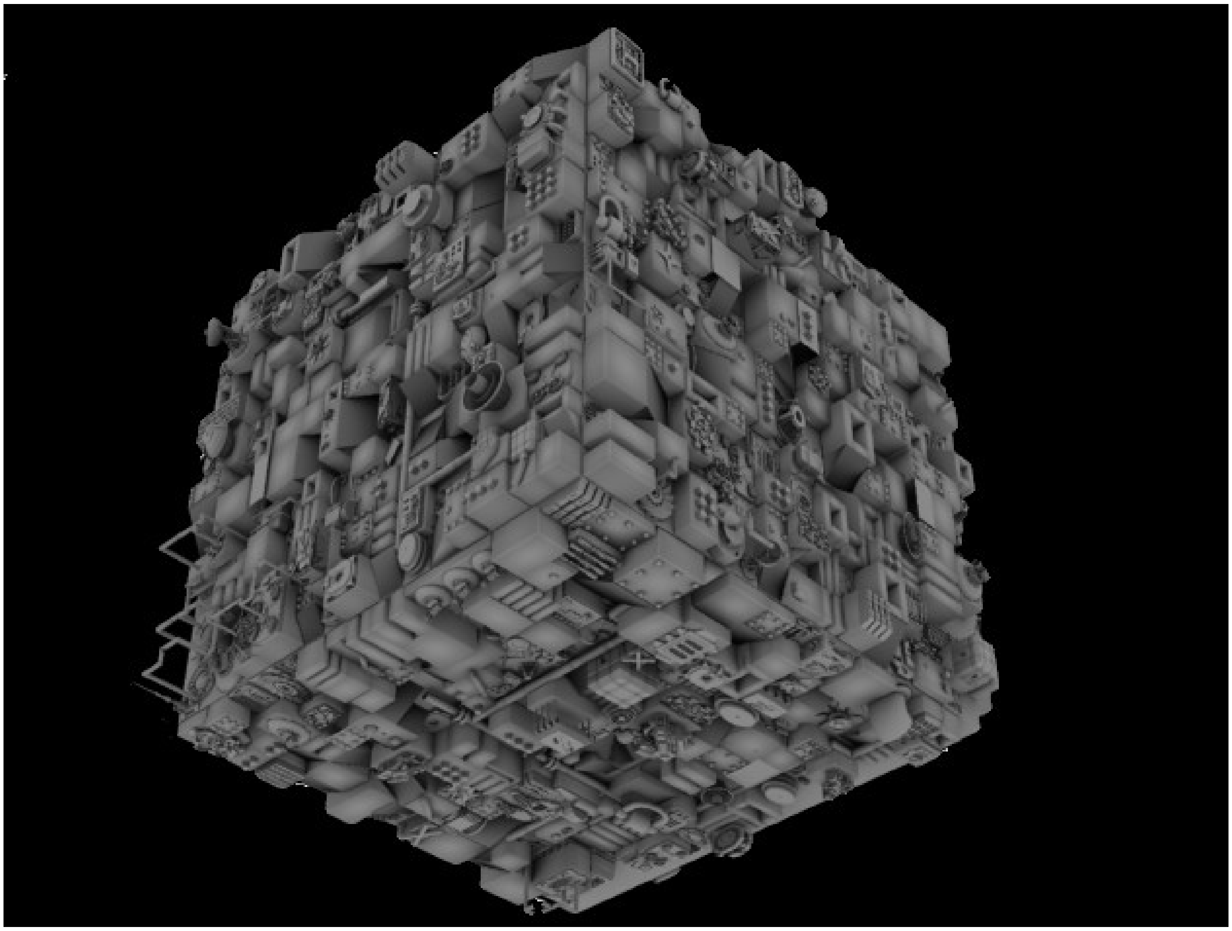


Figure 5.21: A cube frame.

Chapter 6

Conclusions

This thesis presents an analysis of spaceship design based on aesthetic principles and design conventions, which lead to a prototype implementation that followed from this analysis to procedurally generate surface details. This chapter discusses decisions concerning spaceship analysis, evaluates the results of our implementation, and suggests improvements, directions for future work, and some final conclusions.

6.1 Discussion

Much of the work in this thesis consists of analyzing and summarizing conventions that appear in popular spaceship designs, and many of the rules we define are based on patterns that appear in these conventions. This thesis also describes the most logical origin of these conventions, as well as reasons for following them, and cases where it might be appropriate to disregard them. However, all of this work is based on the assumption that these conventions evolve and become established because they are a characteristic of good design. We feel that it is reasonable to assume that, in most cases, design practices are repeated because they are popular or favored. Because there is really no other baseline for determining favorable characteristics of spaceship design, we are forced to make this assumption. While in certain cases conventions may arise for other reasons, such as familiarity, convenience, or habit, we believe that these cases are the exception to the rule.

The discussion of aesthetics frequently references the idea that aesthetic pleasure can be derived from the harmony of uniformity and variety. Although it also includes various strategies for increasing both uniformity and variety, as well as for uniting the two, there is no formulaic method for calculating the perceived interest of a design, nor is there a value for the level of interest that is most appealing. As previously discussed in Section 3.1.2, there is no formula that can be applied to determine the appeal of a design. The appropriate levels of uniformity and variety must still be discerned by the judgement of the designer. Therefore, the levels of uniformity and variety demonstrated in this thesis were selected at our discretion.

Another important contribution of this work is to separate the properties relevant to the outward appearance of the spaceship from the meaningless distractions that so often appear in relation to fictional designs. These distractions indirectly and unclearly affect the appearance of the spaceship, and include explanations of fictional technology, interior arrangements of the spaceship (unless this is also a goal of the design), and classifications of spaceship. We describe our spaceship in terms of functional components with a visually distinguishable purpose, which avoids relying on series-specific information, hypothetical questions, and ambiguous interpretations. However, our method requires that the user have some understanding of the associations between these functional components and desired characteristics of the design. For example, it may be more natural for a user to describe the desired design as a "transport vessel"; however, this phrase is ambiguous, as the characteristics of a transport vessel are not well defined. Instead, a transport vessel could be created with our system by emphasizing the functional components that are likely to be most prominent in this type of spaceship, such as the propulsion system and cockpit, but this selection requires the user to reason about which functional components convey the correct properties for the type of spaceship desired.

We emphasize that complementary geometry between neighbouring panels, such as the aligned wedge and cube, have a large impact on quality, but this leads to the question, why not just add a primitive that is a cube and a wedge attached? While these types of compound primitives may simulate connectivity, they also either limit the possible configurations that we could achieve, or greatly increase the complexity of the number and types of primitives considered. This question stems from the more general question, what types of shapes should we use as elementary building blocks from which we create the surface details? In this implementation, we chose to combine both simple primitives and kitbashed models. If using only kitbashed models, the number of building blocks required is greatly increased to mask the repetition. It is also difficult to create natural connectivity automatically. Using only primitives would be possible, but it would require the creation of complex shape grammar rules to emulate shapes that could just as easily be replaced with static, kitbashed models.

6.2 Results

Our prototype program illustrates how the principles of aesthetics we described might be implemented. We present a method of generating surface details that demonstrates how kitbashing can be incorporated into such a system, the effectiveness of shape grammars in achieving variable yet similar shapes, the appropriateness of a recursive solution for creating a hierarchy of uniformity amidst variety, and incorporation of neighbouring geometry to create connectivity between elements.

If we compare our results to the output of the greeble plug-ins presented in Section 2.1.2, our method makes a few obvious improvements. The most significant change is our addition of kitbashed models. While it is possible to create an interesting surface with cubes alone, this approach would require sophisticated rules and arrangements, which are clearly not present in these plug-ins. Our kitbashed models give our surface a technological appearance that is lacking from the plug-ins. However, we do use some geometric primitives, which we use to connect neighbouring geometry, creating a cohesiveness that, at best, appears only by chance from the greeble plug-ins. We also enforce varying levels of symmetry in our design, allowing our output to appear as a stand-alone design, instead of adding all geometry independently. We also apply shape grammar rules to create geometry that is similar, yet distinct, instead of simply repeating the same elements.

Our results follow the principles we outlined as being important for interesting spaceship design: technological primitives (kitbashing), variety that adheres to an underlying law (shape grammars), graded complication (recursion), and knowledge of neighbouring panels (connectivity). In general, previous methods do not incorporate any of these principles. While a few of the greeble plug-ins incorporate recursion, they use it to create various scales of geometry without including the hierarchy of uniformity and variety.

While we can qualitatively compare the properties and results of these methods, we do not have a formal method of evaluation. We discuss possible future methods of evaluation and other directions for future work in the Section 6.4.

While our implementation represents an improvement over previous methods, many refinements are still needed to make the resulting models suitable for production work.

6.3 Improvements

Our implementation focused on a specific subset of the spaceship generation problem: large scale surfaces created from rectangles. Ideally, we would like to have included the complete set of scales, functional components, and primitives. Unfortunately, time constraints forced us to select a specific subset of these features to include in the implementation. We also chose to focus on carefully and thoroughly categorizing the conventions and aesthetics of spaceships, instead of dedicating a lot of time to the prototype.

We also placed little emphasis on the design of the interface. Usability was considered only in terms of viewing and testing of results. Input parameters were taken from text files, which would be inappropriate for an end user. Our program allows only OBJ files for models, but it would be useful to allow other types of files to be accepted.

Procedural modeling techniques always face the trade off between usability and control, and we chose to err on the side of usability to make our prototype as automatic as possible. However, there are certain cases where a greater level of user control would be appropriate. For example, the user ought to be able to add their own shape grammar rules to the system, to further customize their results.

6.4 Future Work

While this thesis is an attempt to make a starting point for the problem of procedural spaceship generation, there are many other directions to be explored in this area. Since this work focuses almost exclusively on spaceship surface detail, there is much work is left to be done to procedurally generate spaceship frames. We discuss possible alternatives to frame generation and specification in Section 6.4.1. First, we discuss a few possible extensions for the procedural generation of surface details.

As this research focuses solely on mechanical ships, an obvious extension would be to examine the design conventions for organic ships, including the types of surface details that appear and how they relate to and interact with one another.

While we have taken the time to manually mine 3D models for parts that are suitable for kitbashing, the process would benefit from the automation of this step. Models could be segmented and searched for suitable parts that could be used for kitbashing [55]. This would be useful for large spaceships that require a variety of parts to cover the surface.

Our implementation joins two adjacent kitbashed pieces by finding two flat areas on the models and joining them by adding connector geometry. While this is the simplest way to connect the two pieces, it would make more sense to first segment the models and search for already existing geometry that would serve as a logical connector. This connector could then be deformed and extended to connect to the other model, either by attaching it to a flat area or another compatible connector. It would be best to segment and search both models first, to identify whether one, both, or neither has a piece of geometry that would make a good candidate for a connector. Another possible alternative would be to create connectors from a library of kitbashed connector geometry.

The main difficulty of automated kitbashing is determining how to combine nearby models so that they appear related or connected. Instead, it would be easier to simply provide an interface to facilitate manual kitbashing by the user. Geometry could be placed individually, or the system could generate patterns to allow the user to place multiple instances of the same piece. The interface could be programmed to snap the pieces to the lines of the panels and properly position them on the surface to aid the user in aligning geometry on the model. Similarly, panels could also be created or edited interactively to allow the user to customize the generated layout before adding geometry.

Our implementation uses a tile-based system to create pattern geometry with the appearance of a single, connected model. While we created the minimum number of models needed to cover any pattern configuration, it would make sense to provide multiple variations of these models to allow for a greater diversity in pattern geometry.

Another interesting extension would be the creation of steampunk style models. Steampunk is an sub-genre of science fiction that embraces steam powered machinery and the aesthetics of Victorian era England. This style would require replacing sections of the original model with mechanical parts instead of simply covering it, turning this into a packing problem [38]. However, the parts still need to be combined in a mechanically logical way to create a model that looks plausible. Steampunk lends itself to a growth pattern; each piece that is added can decide what other pieces could logically be attached to it in certain configurations with a certain probability.

It would also be possible to use these methods to define a style of spaceship that could be saved and applied to different frames. Often fleets of spaceships consist of groups of different types of ships that have similar visual characteristics. By recording and analyzing the surface detail placement stage, a distinct style of ship could be created. This style could then be applied in varying combinations to other types of ships, simplifying the creation of a fleet of ships.

Formal evaluation of our designs is also an issue. Currently, it is unclear how to evaluate whether a generated spaceship, or really any spaceship design, is good or not. While there are certainly existing spaceship designs that are widely accepted as being good, it is not necessarily correct to evaluate new spaceships based on a resemblance to these old favorites. As the purpose of this research is to create new designs, and not to capture the style of any particular existing ship, it would be worthwhile to find a new way to evaluate the output. However, qualitative or quantitative evaluation of the visual appeal of any type of object or art is difficult, and is still an open problem in non-photorealistic rendering. While the field of computational aesthetics has been formed in part to address these problems, there exists no standard means of evaluating the artistic value of an object. An evaluation of the final designs by a panel of subjects may be one possible course of action. However, this type of subjective evaluation presents difficulties in extracting concrete, meaningful conclusions. It would be of greater interest to further examine how basic principles of aesthetics, such as symmetry, affect the design of spaceships, and technology in general.

6.4.1 Frame

While our implementation assumed that the frame was given, we also examined a few promising methods for automating the creation of the frame, such as a polygonal model or a user-defined 2D silhouette. For each method, we also discuss possible representations for the functional components.

Silhouette

As discussed in Section 4.8, it is clear that the spaceship's silhouette is important to the visual impact of the frame. Hence, it may be desirable to allow the user to specify the 2D silhouette of the frame, and then use this input to generate a 3D shape. This method would allow the user to easily define the general shape of the spaceship frame, and would transfer the responsibility of creating a strong silhouette to the user. However, it still permits the system room for interpretation in generating the final 3D model.

There are existing programs for creating 3D shapes from a user-input 2D silhouette. Takeo Igarashi's *Teddy* is one example of sketch-based modeling, however this system is only capable of creating rotund models, as the 3D object is created by inflating the 2D silhouette [50]. More recent work has introduced the ability to match silhouettes to predefined templates that describe key features of certain types of common objects [98]. Because of the complexity and diversity of spaceship frames, neither method is suitable.

Generating a 3D spaceship frame from a 2D silhouette presents a few obvious difficulties. First, which view of the spaceship should be used for the silhouette? Spaceships are generally seen from the side view, meaning this should be the most recognizable silhouette. However, the side view alone makes it impossible to specify common feature that protrude from the side of the frame, such as wings. The top view may provide the most shape information for the majority of frames, but not all, and is not the natural angle from which the viewer observes the ship.

The natural solution would be to have the user provide both the side and top silhouettes, but this introduces significant complexities when trying to consolidate the two views. First, combining the two views may simply not be possible. Even if we were able to assume that two given silhouettes were possible to consolidate, it would be difficult to ensure that the geometry of the resulting model would be connected.

Input considerations aside, it is not clear how a 2D shape should translate into 3D, especially for an object as complex as a spaceship. While certain objects can be represented faithfully in 2D, spaceships feature such variance and irregularity that a lot of information is lost when this third dimension is removed. Given a simple 2D shape, it is difficult to decide how to form a complex 3D shape.

Finally, since the 2D silhouette may translate to an incredibly complex 3D object, it would likely not be possible to denote the functional components of the spaceship by annotating the silhouette. The user would likely have to annotate the resulting 3D model.

Polygonal Model

Since many spaceship frames are derived from or inspired by certain types of real world objects, it would be logical to allow the user to provide a polygonal model as input and use this model as a basis for the spaceship frame. This method is also convenient because it allows the user to take advantage of the vast number of polygonal models that already exist.

This method would require that the faces of the model be processed in some way to make them better suited for the panels of the spaceship. While we may not use these faces as the final panels, they must still be cleaned up to provide an appropriate base from which panels could be created. This would likely involve alignment of the faces with the axes of the ship (defined by the axis of symmetry, up direction, and forward direction provided by the user), adjustment of size to avoid panels that are too small or too large, consistency of shape that favors quadrilaterals, and enforcement of bilateral symmetry. The final model should be a coarse representation of the original input.

This method would also require that the user somehow annotate the geometry of the model to delimit the desired functional components. This step should occur after the faces of the model have been processed.

This is the most promising method for providing the highest level of automation for frame generation, as many spaceship frames are clearly based on real world objects, such as vehicles and animals, but the success of this method would rely heavily on the original input model.

The main disadvantage of this method is that it does not naturally provide the program with knowledge of the general shape of the frame, as polygonal models provide only local information about shape. Therefore, to split this set of polygons into panels as is, we can make only local decisions without an idea of what these decisions will look like as a whole. To gain high level information about the shape of the frame, we would first have to segment the model, and then fit primitives to its surface [13]. This process would likely require some user input, but would allow the program to have an idea of the frame's overall shape.

Also, there is no easy way for the user to specify which portions of the model belong to which functional component. It would be best to provide an interactive paint interface to allow the user to color code the geometry of the model to represent a given functional component.

Constructive Solid Geometry

While a polygonal mesh is the most versatile and convenient form of frame representation, it provides no natural high level information about the shape and configuration of the spaceship frame. Constructive solid geometry (CSG) would be appropriate for this problem. CSG is a representation of solids that uses Boolean operations to combine primitives. Common primitives include cubes, spheres, cylinders, cones, and pyramids. Common Boolean operations include union, intersection, and difference.

By representing the frame as a combination of CSG primitives, we gain high level information about the shape of the frame. CSG also provides a natural way to specify the frame's functional components; each primitive is defined as belonging to one type of functional component. Furthermore, it allows the user to semantically break up the frame of the ship such that parts of the surface that should be processed together are part of the same primitive.

Not only does this simplified representation of the frame allow for more meaningful processing, it also allows for a simplified representation of the frame. While we represent the frame as a set of CSG operations, we do not actually want to compute the results of these operations. Computing the exact surface would leave us again with a polygonal mesh, destroying the high level information that was the purpose of using CSG. Instead, we process the frame as a set of primitives, calculating the exact surface only as needed for rendering the frame.

6.5 Summary

This thesis presents a detailed analysis of the conventions that appear in spaceship design, including a discussion of their origins, their uses in emulating certain traits, and reasons these conventions might be followed or ignored. We uncovered these conventions by examining and comparing popular spaceship designs from the past sixty years, which we present in a detailed survey.

The thesis also examines the aesthetic principles of information theory, which describe the balance of uniformity amidst variety, and discusses specific strategies for incorporating these principles into the creation of spaceship surface details.

Finally, it presents a prototype implementation for automatically generating surface details of large scale spaceships that follow the conventions and aesthetic guidelines that were previously discussed.

This research is a first step towards automating the generation of technological models, specifically spaceships, and will hopefully lead to research in procedural modeling branching out to less traditional content. Procedural design of spaceships can find wide applicability in movies, television, games, and other forms of entertainment. Besides these practical applications, they serve as an excellent domain in which to probe the deep underlying relationship between computers and aesthetics.

Appendix A

Spaceship Survey

This chapter presents a detailed survey of spaceship designs. While we cannot hope to mention every science fiction artist, movie, television show, or video game, this section attempts to categorize and analyze the strengths and weaknesses of some of the major science fiction influences. These features are described in terms of the aesthetic and spaceship-specific conventions that were described in Chapters 3 and 4. This section focuses on specific designs that use these conventions to their advantage, as well as notable deviations from them.

It is important to note that, when dealing with a class of objects that relies heavily on expectations, namely ship designs that have come before and been successful, similarities between designs are common. Discovering these similarities and why they are popular is part of the purpose of this work. However, fans sometimes see these reproductions as copying or stealing another's design, when this is simply not the case. Similarities between ships are common, not just because they are a way of expressing admiration for another designer's work, but because they provide credibility to a new design by borrowing from the credibility of an already established ship.

Before beginning, we would like to establish a few limitations that have been imposed to make this survey manageable. For many sections, a ship is examined that serves as a representative of a certain class. For example, the TIE series of starships from *Star Wars* contains many different ships. However, because all of these ships follow the same basic design, a single representative, the TIE fighter, has been singled out for detailed examination in Section A.5.2. Where applicable, a ship is chosen that exemplifies a certain style. Variations that offer design anomalies that warrant individual examination or comparison may also be considered.

In general, this survey ignores any details of a spaceship other than its outward appearance. While science fiction universes often offer detailed specifications for the technologies behind their spaceships, these details, aside from the most basic of classifications, have no bearing on the visual design of the ship. A spaceship's appearance is always conceived of first; rationale to explain its inner workings are then added later [81, 89]. Separate from this, focusing on fictional technologies as a rationale for design is counter-productive. While some fictional technologies may have detailed specifications, especially those that have been used as a basis for role playing games, these specifications are never complete, and are often tailored to a certain purpose (such as combat). Without concrete rules for how these technologies behave in all situations, the result is an ever increasing series of hypothetical questions. These questions distract from the fact that the nuances of a fictional technology have nothing to do with good spaceship design.

For example, while the TIE series of spacecrafts in *Star Wars* is now recognized as an acronym for Twin Ion Engines, George Lucas originally named these ships TIE fighters because of their resemblance to bow ties. The rationalization and technological explanations, including the Twin Ion Engines acronym, were added later [4]. Technical descriptions and rationalizations are rarely a top concern for film and television designers; having the right look, as well as staying within budget, are their priorities [66].

There is also no preference between hard science fiction and space opera designs. Hard science fiction refers to designs that attempt to represent scientific knowledge and theory accurately, while space opera refers to fantastic science fiction. Films such as *2001: A Space Odyssey* have been praised for their scientific rigour; however, aside from the perceived effort and knowledge of the film's designers, this attempt at realism has little bearing on the aesthetic value of a spaceship's design. First, if we assume that the viewer has no knowledge of the origins of the spaceship's design, it is unlikely that these accuracies would be noticed. Second, since the film's creators did not design and build an actual working spaceship, it is unlikely that every aspect of the ship, i.e., every surface detail, was placed with a specific function in mind. The rigor of hard science fiction, while laudable, serves only to limit the space of possible designs. While it is clear that the knowledge and application of certain physical restrictions may increase the plausibility of a design, we merely wish to discourage the judging of a design based on factors that are not visibly manifested in the design. This survey focuses on the inherent pleasure of a design, not pleasure that may arise from knowledge of the design process.

As a final note, this survey focuses primarily on spaceship designs from movies, television, and video games. As the goal of this research is to generate 3D models of spaceships intended for these applications, it follows that our study should focus on spaceships from these types of media, as opposed to 2D media such as cartoons, comics, and manga. Spaceships in this survey are organized roughly in chronological order, with the exception of the designs presented in Sections A.8 and A.9.

A.1 Early Spaceships

With the launch of the Soviet Union's Sputnik satellite in 1957 came the dawn of the Space Age. With the prospect of space exploration suddenly realistic, the world's interest in science fiction was renewed. Science fiction took a decidedly propagandist turn, as space travel was heralded as mankind's inevitable destiny to extend out into the galaxy. Space travel became romanticized; the saucer, which had already earned itself a questionable reputation in alien encounters, became the symbol of all that is foreign and alien, while the rocket became a lofty symbol of mankind's evolution and power [88].

The rocket has clear human affiliations. It mirrors our current technology and spacecraft design (both in the 1950s and now), making it familiar. In the 1940s and 1950s, this familiarity was crucial for believability. With the introduction of the German V2 rocket, the first large enough to leave the atmosphere in a test in 1943, many film and television designers modeled their spaceships after the V2 to borrow from this success. In addition to this, designing a spaceship to resemble the V2 as much as possible allowed special effects teams to use stock footage from the V2's actual launch test, instead of having to create their own launch (and landing) sequences. This footage was taken advantage of in a variety of films and television shows in the 1950s, including the *Tom Corbett, Space Cadet* series in 1952, the film *Rocketship X-M* in 1950, the film *Abbot and Costello go to Mars* in 1953, and the 1959 film *Missile to Mars*, among others [43]. However, the rocket design is rarely used now, and is associated with retro science fiction. Current trends in science fiction tend toward the prophetic, so a design that reflects our current technological level is at odds with a design meant to appear sophisticated and advanced.

The rocket's pointed shaft and distinct windows give it a clear sense of scale and direction. The rocket is usually devoid of any surface details, with the exception of the occasional use of regular panels. It uses a traditional propulsion system that may contain a variable number of exhaust cylinders. Wings are sometimes added to the shaft of the rocket, usually at the bottom, in a radially symmetric layout. Examples of 1950s rockets are shown in Figure A.1.

The saucer is the rocket's alien counterpart, emphasized by its obvious deviation from the laws of science adhered to by the rocket design. The term "flying saucer" was popularized in the late 1940s, first appearing in a report made by Kenneth Arnold in which he describes a group of flying objects near Mount Rainier, Washington State, in June 1947 [88]. The term was soon adopted by the public and inevitably made its way into science fiction, where it soon became a staple of the genre. Flying saucers are more objectively referred to as unidentified flying objects (UFOs), a fitting term as their exotic design was nothing like the current technology of flight. The saucer has acquired many negative associations over the years and is commonly associated with alien abduction theories [88]. Its frame is a simple disc, though this disc may feature various layers or domes. The disc is often ringed in rows of lights, but this is usually the only type of surface detail present. The saucer is radially symmetric with no visible propulsion system, and thus no clear sense of direction. Examples of 1950s saucers are shown in Figure A.2.

The traditional rocket and saucer are now seldom used in modern science fiction, though their designs are iconic of the traditional human versus alien dilemma. However, adaptations of these design are still used, and without the human/alien bias. Modifications of the saucer appears in many popular human spacecrafts, most notably *Star Trek*'s USS Enterprise and the *Star Wars* Millennium Falcon.

(a) Cover art for the October 1952 edition of Italian magazine *Romanzi di Urania.*

(b) Cover art for the February 1952 edition of *Galaxy Science Fiction* magazine.

Figure A.1: Examples of 1950s rockets.

(a) A movie poster for the 1957 film *Invasion of the Saucer-Men.*

(b) Cover art for Gray Barker's 1956 book *They Knew Too Much About Flying Saucers.*

Figure A.2: Examples of 1950s flying saucers.

A.2 Science Fiction Artists

Science fiction art, and the artists behind it, tends to be overlooked, especially when compared with other media such as film and television. However, science fiction art often appears behind the scenes as concept art, posters, and advertising. Though often passed over by all but the most hardcore fans, a wide variety of science fiction novels have featured art from the genre's most celebrated artists.

Science fiction art, especially art that is not created as a commissioned work, provides an interesting perspective on the design of spaceships. Compared to other types of media, art is static; it tells a story in a single frame. An artist's view of technology and space travel, without the usual input from model builders, engineers, and directors, is freed from the constraints of budget and physics that these conditions inevitably create. Frame and silhouette are given a greater emphasis, and since the spaceship is only seen from a single angle, this angle can be chosen strategically. Details are not specified on the surface of the ship, but are implied using color and texture.

This section examines a few of science fiction's most famous artists, who have contributed to everything from *Star Wars*, to Orson Scott Card's *Ender's Game*, to H.G. Well's *The War of the Worlds* [36, 30, 1].

A.2.1 John Berkey

> There are hazards in knowing too much about engineering or technology. It can limit the imagination.
>
> John Berkey

John Berkey is one of science fiction's most respected artists. After attending the Minneapolis School of Art in 1950, he spent his early years painting a variety of calendar illustrations. Known for his various works of American Vistas, Berkey was soon recognized for his various series of aircraft, cars, and ship paintings for the US Army and Navy. From modern vehicles, he then turned to science fiction, attracted, like most artists, by the aesthetic freedom the genre brings. Over the course of his career, Berkey created a variety of mainstream movie poster designs, including artwork for *Star Wars* as well as numerous sci-fi novel covers [36].

Though often described as an electronics buff, Berkey tries to keep knowledge from interfering with imagination by avoiding the use of current technology as a basis for his art. In fact, his attempts at depicting hard science fiction were less than successful. As technological limitations are based on the knowledge of the day, his works of realistic science fiction appear dated when compared to his usual work [36].

Berkey's avoidance of photo-realism is different from the style of the typical science fiction illustrator. Science fiction paintings are not encumbered by the need for detail, definition and realism that his illustrative art is known for, allowing him a more experimental outlet. One of the first science fiction artists to break out of the mold of the rocket, his spaceships feature few sharp edges, showing a definite preference for organic, free-form shapes. These forms are almost always adapted from familiar shapes, creating a bond between the real and the fantastic that makes Berkey's designs so accessible. Fish, birds, dogs, eggs, surfboards, sail boats, and acorns can all be recognized in the exotic forms of his ships [36]. Despite this variety, Berkey's ships always feature a recognizable cockpit to give the viewer a definite sense of scale and direction.

Berkey prefers strong, massive ships, which he refers to as "behemoth space yachts" [36]. This type of spaceship allows him to better convey the intensity of space. His approach to spaceship design was simple and logical, recognizing the primacy of the illusions of flight, speed, and size.

Berkey's impressionistic style features a bright, colorful palette not often seen in this type of art. It is through a clever use of color that Berkey adds the appearance of detail and definition that would not otherwise be possible with such loose brushstrokes. He favors balance and symmetry in his designs, but not to the point of interference [36].

Examples of John Berkey's work are shown in Figure A.3.

(a) A painting of one of John Berkey's space yachts

(b) An advertisement painted by John Berkey for *Star Wars* [9]

Figure A.3: Example works by John Berkey

A.2.2 John Harris

John Harris began painting at an early age, eventually studying painting at Exeter, but did not begin working seriously as a painter until the late 1970s. He specialized in science fiction art, with the majority of his work appearing as book covers and advertisements, including the artwork that appears on the covers of Orson Scott Card's *Ender* series. He has also produced paintings for travel books and general fiction. His primary medium is oil painting, though he has experimented with other methods and techniques.

Harris describes his muse as "the sense of scale, the atmosphere of being in an unknowable and unlimited space" [1]. His spaceships are gritty and rugged with a pieced together appearance. They tend to be large scale and clunky, not smooth or sleek like the traditional spacecrafts of the time.

Examples of John Harris's work are shown in Figure A.4.

(a) Cover art for Jack McDermott's *A Seeker* [1]

(b) John Harris cover art for Barrington J. Bayley's *Annihilation Factor* [1]

Figure A.4: Example works by John Harris

A.2.3 Vincent Di Fate

In contrast with John Berkey, whose painting talents extends to everything from the American countryside to aeronautics, Vincent Di Fate is a science fiction artist; in fact, the subject of each and every one of his more than 3000 illustrations has been science fiction, fantasy, or aerospace [30].

Vincent Di Fate is a celebrated science fiction artist who has worked in the field for over thirty years. He began working as an illustrator for *Analog* magazine as early as 1965 [30], graduating from the New York-based art school The Phoenix in 1967. Di Fate found science fiction art appealing because it allowed him to create designs that were freed from the constraints inherent to the design of familiar objects. Indeed, he was often called upon to “rearrange the universe to suit the whims of a client” [30].

The influences of the 1950s can be clearly seen in Di Fate’s work, featuring various incarnations of rockets and saucers. Di Fate prefers the “less is more” design philosophy, as fewer details leaves more to the imagination of the viewer. He attributes the success of his style to his ability to integrate the familiar with the fantastic, by excelling at “hiding the seams between what is real and what is imagined” [30].

Examples of Vincent DiFate’s work are shown in Figure A.5.

(a) Vincent Di Fate cover art for Donald Kingsbury’s *To Bring in the Steel* [30]

(b) Vincent Di Fate cover art for the Science Fiction Book Club’s anthology *The Good Stuff* [30]

Figure A.5: Example works by Vincent Di Fate

A.3 Star Trek

The *Star Trek* franchise, created by Gene Roddenberry, began in 1966 with the start of *The Original Series* television show’s three year run. The *Star Trek* universe proceeded to spawn four more television shows: *The Next Generation*, *Deep Space Nine*, *Voyager*, and *Enterprise* beginning in 1987, 1993, 1995, and 2001 respectively. Eleven feature films were also created, with the first six featuring the cast of *The Original Series*, the seventh featuring a combined cast of *The Original Series* and *The Next Generation*, the eighth through tenth featuring the cast of *The Next Generation*, and the eleventh featuring a re-imagining of a younger version of the crew from *The Original Series*. Since most of the films are feature-length versions of the television series and follow its design principles, this section focuses on the various incarnations of ships that appear in the five live-action series, as well as how the ships of the more prominent races have evolved over time.

Star Trek takes place in an optimistic and relatively stable future. The plot generally revolves around the crew of a single ship (or space station, in the case of *Deep Space Nine*), belonging to the United Federation of Planets, or simply the Federation. The Federation is an alliance of various races, including humans, dedicated to liberty, justice, and cooperation in the galaxy. Starfleet is the military branch of the Federation, though its ships are usually used for exploratory missions, with the occasional defense or peace-keeping assignment [10].

The *Star Trek* universe is clean, ordered, and civilized, and embodies hope for the future. Introduced at a time of social, political, and racial turmoil, the secret to *Star Trek*'s continued success is thought to stem from its positive outlook of the future and the prosperity of the human race [80].

By the start of *The Original Series*, the rocket spaceship design was already becoming dated, so *Star Trek* spaceships began the deviation away from these traditional shapes, with the Enterprise being a combination of both saucer (the hull) and rocket (the nacelles). Color television had only recently been invented and was still a novelty, so the series took advantage of this by making all of the sets, props, and costumes as colorful as possible to make them seem futuristic [80].

The design philosophy for *Star Trek* can be broken down into three main factors: cost, practicality and consistency. Television shows often suffer from restricted budgets and tight deadlines, so designs and effects are limited by what can be done within these constraints. Alien ships in *The Original Series* were often reused in various episodes, sometimes with different scaling, to avoid having to design and build new ships each week. Federation starships appear alike, not only to maintain consistency, but because they were sometimes created from models or parts of previous ships to avoid starting from scratch. Curved hulls on spaceships were avoided because they were difficult and expensive to build [80].

The practicality of a design refers to its functional details. In other words, a design must appear futuristic, but not so advanced as to be unidentifiable; the intended purpose of its parts must be clear. *Star Trek* was one of the first to depict science fiction as utilitarian and avoided adding unnecessary embellishments. The smooth design of the Federation starships stems from the designer's idea that the mechanics of a spaceship should be completely contained to avoid the danger of a crew member having to travel outside to make a repair [80].

Consistency is important for creating continuity in the show's universe; new designs must appear to fit in with what has come before [80]. For example, Federation starships should be recognizable as such, and alien spaceships should be distinguishable based on race, not only to inform viewers of their origins, but because it is likely that spaceships built using a single type of technology will follow similar design patterns. *Star Trek* ships emphasize the striking profile of the frame, rather than the complexity of the surface details.

Uncommon to most science fiction shows, *Star Trek* featured spaceship designs that were common across species, such as the Cardassian and Klingon cargo transport vessels, and the Klingon and Romulan Bird-of-Prey. Though this duplication likely resulted from model reuse, it may also suggest that it is not uncommon for a species to trade, salvage, or steal from another. However, this reuse deviates from the common practice of ensuring that the affiliation of a given spaceship is immediately identifiable. Identical Klingon and Cardassian transports are shown in Figure A.6.

(a) A *Star Trek* Klingon transport (b) A *Star Trek* Cardassian transport

Figure A.6: Two identical transport vessels, affiliated with different species from *Star Trek*

Though "spaceship" and "starship" are often used interchangeably in science fiction, these terms have distinct meanings in the *Star Trek* universe. "Starship" refers to a vessel that is capable of faster than light travel and is equipped with warp nacelles. "Spaceship" refers to a vessel that is not capable of faster than light travel and does not feature warp nacelles. Using these definitions, almost all Federation vessels are categorized as starships.

The following sections detail the spaceship design of the most prominent species and organizations featured in *Star Trek*, specifically focusing on those that were featured throughout all of the different series.

A.3.1 Federation Starships

The original Federation starship is a Constitution class ship; it defined the style of the Federation Starfleet, and all other designs are variations of this base. This consistency in spaceship design brings credibility to new designs by creating the appearance of a common technological basis. Federation starships all have the following features: a *primary hull*, usually in the shape of a disc or sphere, an *engineering hull* attached to the primary hull, and some number of *warp nacelles*, placed symmetrically, for propulsion. These three features are shown in Figure A.7. We refer to the section connecting the primary hull to the engineering hull as the *neck*, and the sections connecting the warp nacelles to the engineering hull as the *nacelle pylons* [84].

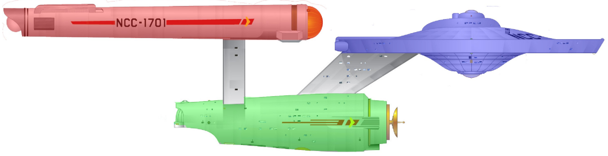
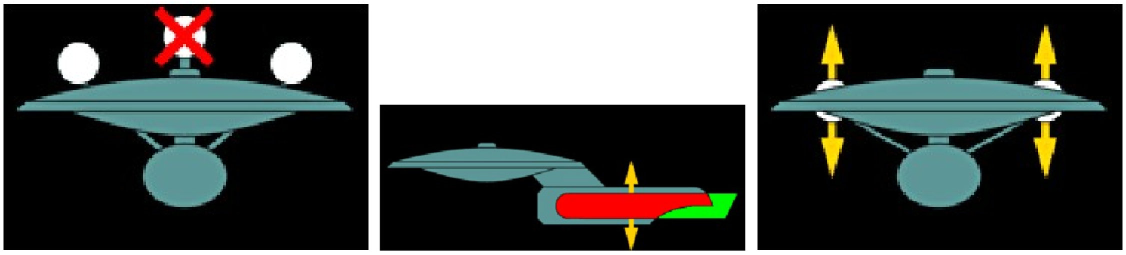


Figure A.7: A side view of the *Star Trek* Enterprise, annotated to show the three major sections: the primary hull in blue, the engineering hull in green, and the warp nacelles in red.

Beyond the inclusion of these basic components, Federation starship design also follows specific rules regarding the placement of the warp nacelles, shown in Figure A.8. Though these rules are often credited to Gene Roddenberry, they were not all established by the show's creator [93]. Other design patterns, though not explicitly specified, are also followed. Federation starships are primarily grey, featuring red and blue accents, their panels are rectangular and placed in a regular pattern that follows the lines of the ship, and the only surface details are lights (representing windows) that are arranged in coordination with the panels, with the exception of the occasional raised panel or additional decal. There are more than fifty classes of spaceship in the Federation, most of which follow these principles. There are a few classes with an odd number of nacelles, such as the Niagara class with three and the Freedom class with one, but this is the only design rule to be broken.

Ships of the Starfleet embody the values of the Federation; they are sleek, shiny, and clean, keeping in line with the Federation's optimistic role in the future.



(a) *Star Trek* Rule 1: Warp nacelles must be in pairs.

(b) *Star Trek* Rule 2: Warp nacelles must have at least fifty percent line-of-sight on each other across the hull.

(c) *Star Trek* Rule 3: All warp nacelles must be fully visible from the front.

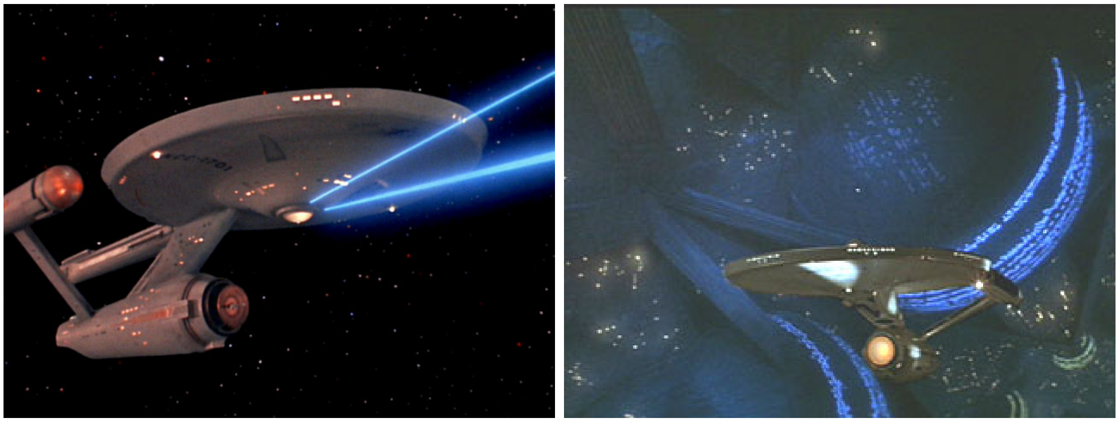
Figure A.8: *Star Trek*'s Federation Starship design rules. Images used with permission of the author [84].

A.3.2 The Enterprise

The Enterprise is the original Constitution class starship, and the most iconic and recognizable ship of the *Star Trek* universe. The Starship Enterprise has had various incarnations over the years, but since *The Original Series*, it has been heavily revised only twice: once for *Star Trek: The Motion Picture*, and again for *Star Trek: The Next Generation* [80].

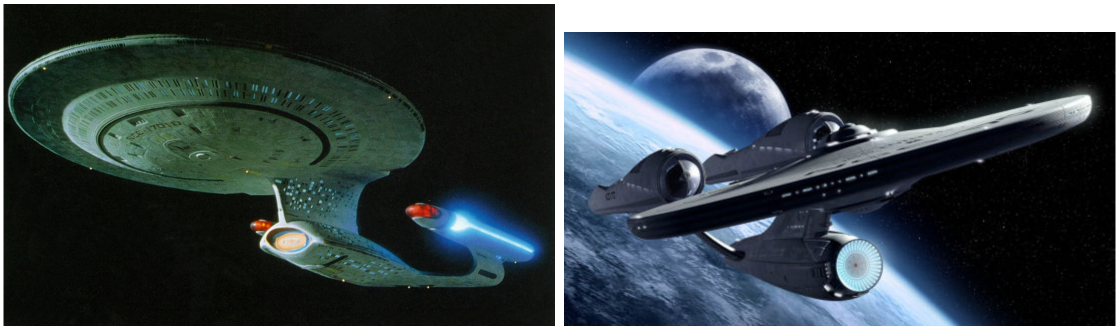
The original Enterprise was designed by Matt Jeffries, who took a practical approach to spaceship design. Gene Roddenberry was adamant about avoiding the already dated rocket, so Jeffries' only direction was to avoid this shape. The basic design came from the intended speed of the ship; a fast ship would have to have a large, powerful propulsion system. Jeffries decided that the propulsion system should thus be removed from the main body of the ship, as engines of such power would likely be volatile. Though the primary hull was originally designed as a sphere to purposefully avoid the flying saucer connection, it ended up being flattened out into a saucer after a few revisions. Jeffries' practicality was also responsible for the smooth surface of the Enterprise [51].

The Enterprise is sleek and majestic, with a striking profile. The ship has no visible weapons, and its pronounced propulsion system mirrors its exploratory purpose. Various incarnations of the Enterprise are presented in Figure A.9.



(a) The original Enterprise as depicted in *The Original Series*. This Enterprise is comprised of mostly simple geometry. The neck is a rough parallelepiped, and the nacelle pylons are rectangular cuboids. The warp nacelles appear perfectly cylindrical.

(b) The revised Enterprise that appeared in *Star Trek: The Motion Picture*. This Enterprise is slightly more complex than its predecessor. Nacelle pylons are now more triangular, and connect to the engineering hull much closer to the neck, which is nearly twice as thick as before. Warp nacelles are slightly smaller than the originals [84].

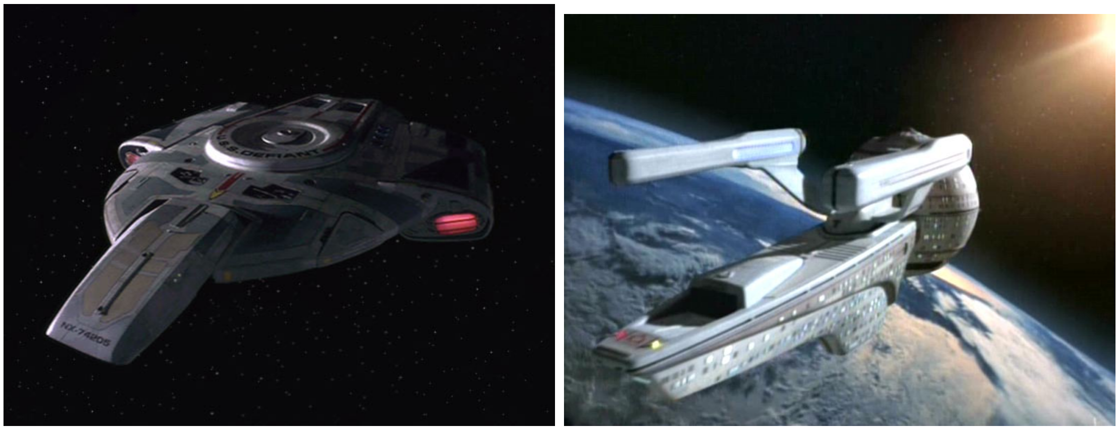


(c) The completely new Enterprise that appeared in *Star Trek: The Next Generation*. The primary hull is now flat and smooth, with an increased density of windows and surface details. The connection between the primary hull and engineering hull is much wider than previous incarnations, and is slightly curved. Nacelle pylons also feature an exaggerated curve, and attach to the bottom of the engineering hull instead of the top. The warp nacelles are now elliptic cylinders, not the usual circular cylinders.

(d) The revised Enterprise that appeared in the 2009 *Star Trek* prequel film. This Enterprise is reminiscent of the original, returning to largely simple geometry. The most noticeable change is the shape of the warp nacelles, which are sloped at the back.

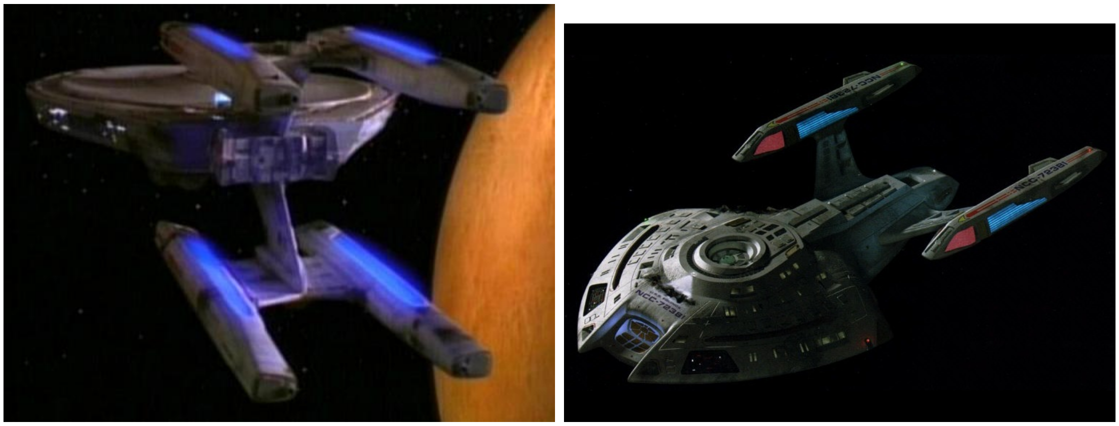
Figure A.9: The evolution of the *Star Trek* Enterprise.

A.3.3 Other Federation Starships



(a) The Defiant class is one of the few Federation starships designed specifically for combat, with no other use in mind [10]. Likely because of this, it is unlike any other Starfleet designs.

(b) *Star Trek*'s Olympic class starship. This class of starships were used as medical ships.



(c) *Star Trek*'s Constellation class starship, featuring four warp nacelles

(d) *Star Trek*'s Nova class starship

Figure A.10: Other *Star Trek* starships belonging to the Federation.

A.3.4 Vulcans

Vulcans have appeared in all of the *Star Trek* series, and appear as main characters in three, *The Original Series*, *Voyager*, and *Enterprise.* Vulcans were the first species to make contact with humans, and were the primary contributors to the formation of the Federation. Vulcans are best known for their dedication to logic and the suppression of emotions, particularly aggression. Cool and controlled, their honesty is legendary among the other races. Many Vulcans have some form of telepathy [10].

Despite their importance in *Star Trek*, major Vulcan spaceships were not introduced until the franchise's final show, *Star Trek: Enterprise.* Because so few Vulcan spaceships had been seen previously, these new designs were based largely on Vulcan architecture and clothing [32]. As Vulcans are a generally peaceful race, few of their starships function primarily as warships and do not feature any obvious weapons systems. Their ships favour rounded contours and are decidedly not aggressive in appearance. They are usually reddish in color. The ring component common to larger Vulcan starships is the warp drive, the technology that permits faster-than-light travel. Note that this ring is not physically connected to the main body of the ship.

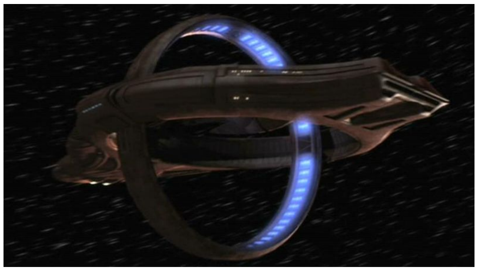


Figure A.11: A Vulcan D'Kyr class starship

Figure A.12: A Vulcan Suurok class starship

A.3.5 Romulans

Romulans also appear in all of the *Star Trek* series, generally depicted as villains or antagonists. Though they share a common ancestry, Romulans and Vulcans are distinctly opposite; Romulans are passionate and calculating where the Vulcans are calm and rational. Romulans are a moral people, believing that the world can be divided cleanly into right and wrong. They believe that their race is destined to conquer the galaxy and their favored war tactic is patience, preferring to wait and observe an enemy before striking to test and evaluate them [10].

The Romulans are modeled as a futuristic Roman Empire, and include many references to ancient Rome. Their home planets, Romulus and Remus, are named for the mythological brothers who founded the city of Rome. Romulan ships are generally green and grey and are modelled after large birds of prey, which is also used as the symbol of the Romulan Empire. Their most common ship types include the Warbird, shown in Figure A.14, and the Bird-of-Prey, shown in Figure A.13, both of which have an avian appearance. Romulan ships are typically green and their appearance is aggressive, including hooked beaks and pointed wings.

Figure A.13: A Romulan Bird-of-Prey

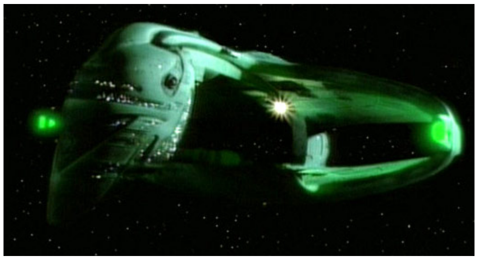


Figure A.14: A Romulan D'deridex class Warbird

A.3.6 Klingons

The Klingons are a warrior race, driven by the pursuit of honor. They are hostile and aggressive by nature. Klingons appear in all of the *Star Trek* series, though their appearance was completely revised after *The Original Series* [80]. Originally the Klingons were primarily antagonists, but in later series they became associated with the Federation, with *The Next Generation* featuring a Klingon crew member as a main character [10].

Where the Federation's Starfleet employs primarily exploration vessels, the Klingon Empire uses warships almost exclusively. Though the D7-class battlecruiser was the first Klingon ship featured in *Star Trek*, the Klingon Bird-of-Prey, shown in Figure A.15, is the most common. First appearing in *Star Trek III: The Search for Spock*, it was originally intended to be a Romulan ship that had been stolen by the Klingon commander. However, this association was later written out, but the ship kept the Romulan name and design connections, based on the Romulan Bird-of-Prey shown in Figure A.13 [80]. Like the Romulans, Klingon ships are generally avian in appearance, with pointed wings featuring triangular feather-like panels. The propulsion systems for Klingon ships are mounted on the tips of the wings, an uncommon location.

Figure A.15: The Klingon Bird-of-Prey

Figure A.16: A Klingon Vor'cha class heavy cruiser

A.3.7 The Borg

The Borg were first introduced in *The Next Generation* and soon became the series' primary villains. The Borg are a cybernetic race of humanoids, made up of other species that they have assimilated. The Borg seek new technology, which they acquire through the assimilation of other races into their collective consciousness. The Borg have no concept of the individual or the self, only an awareness of the hive [10].

The Borg are the most alien of the races encountered in *Star Trek*; they are incredibly powerful, with their hive mind allowing for almost instantaneous adaptation, and pose a particularly sinister threat. The Borg cannot be reasoned with, exemplified by their "resistance is futile" motto [10].

All Borg ships are simple geometric shapes. They are dark grey, sometimes lit by an interior green glow. They are huge in size, with the surface packed full of layered configurations of pipes, as well as other small technological details. These surface details were actually created from the plastic flashing left over from an assembled model kit [46].

The Borg's first and most iconic spaceship was the Borg Cube, shown in Figure A.17. The ship's complete disregard for all design conventions made it revolutionary, enhancing the alien qualities of the Borg. It lacks size and direction cues, any recognizable surface details such as windows, and identifiable functional components. This homogeneity of the Borg ships, and lack of identifiable features, mirrors the Borg hive's lack of individuality. Though other Borg spaceships would follow this pattern, such as the Borg Sphere, it was the Borg Cube that pioneered this style, leaving subsequent designs feeling slightly less authentic.

The Borg Renegade spaceship is also of interest, and is shown in Figure A.18. Crewed by a group of rogue Borg drones, this ship's frame was irregular and asymmetric, a strong contrast to the Borg's usually symmetric designs. Its appearance reflected the chaos and individuality of its crew, who had been separated from the Borg collective and infected with a sense of self [10].

Figure A.17: The Borg Cube

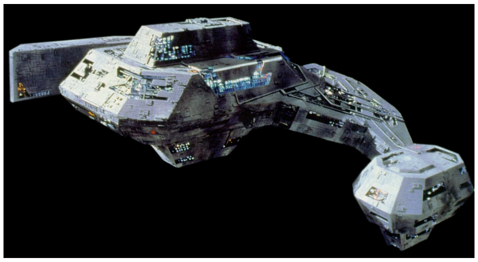


Figure A.18: The Borg Renegade spaceship

A.3.8 Other *Star Trek* Ships

Figure A.19: A Galor class starship, the most common type of Cardassian ship.

Figure A.20: The Breen Warship, designed by John Eaves.

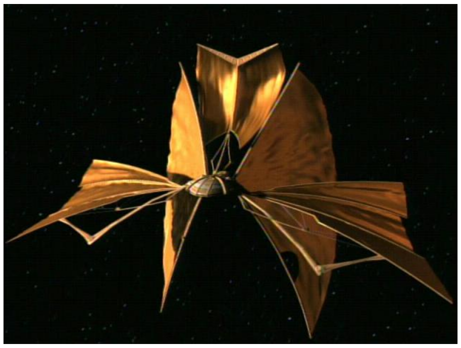


Figure A.21: The Bajoran lightship, powered by its solar sails, is clearly modeled after some kind of sailboat.

Figure A.22: The Deep Space Nine space station was originally a Cardassian mining station that was later used by Starfleet. The design of the station was originally based on a gyroscope shape, but was later inspired by the nucleus of an atom [10]. The station's spherical shape provides the appropriate lack of directionality and its missing panels and exposed parts reveal the grittier atmosphere of the show.

A.4 2001: A Space Odyssey

The film *2001: A Space Odyssey*, released in 1968, ushered in a new age of realism for science fiction spaceship design. The previous favorite, the classic rocket, had been the ultimate in realism, depicting spaceship design as a stylized version of our current technology. *2001* introduced audiences to a futuristic realism by depicting spaceship designs as we might conceivably build them in the future. *2001* also introduced a new age for surface details. Where smooth, shiny surfaces were once the norm, this film presented spaceships with mechanical nurnies and began the trend for more realistic surface details. Though *2001* is often credited as the pioneer of kitbashed spaceships, this is incorrect, as some science fiction films and shows in the 1950s employed kitbashing for their models [43]. However, this film did popularize the method, breaking out of the mold of shiny, sleek spaceships, and leading the way for the *Star Wars* style of pieced together surface details.

Written as a collaboration between director Stanley Kubrick and renowned hard science fiction author Arthur C. Clarke, *2001* is characterized by an unparalleled level of scientific research and realism. Frederick Ordway, the renowned aerospace historian, was recruited as scientific adviser for the film, but he was only the beginning; a variety of industries and companies were enlisted to help in the design of everything from spaceships to communications. This included, among others, NASA, General Electric's Missile and Space Division, Grumman Aircraft, IBM, and Bell Labs [43].

The main spaceships in the film include the Orion III Pan Am shuttle, modeled roughly after the Sänger antipodal bomber, the wheel-like Space Station V, an updated version of Willy Ley's design that appeared in Collier's magazine in 1952, the almost spherical Aries IB shuttle, clearly based on the Apollo Lunar Module that was in development at the time, and the spaceship Discovery, the setting of most of the film's action. Surface details for these ships (aside from the Orion III, which was mostly smooth to emulate commercial airliners of the day) were kitbashed [43]. A major source of the appeal of these designs is consistency, produced by the purposeful creation of technologies that appear to have a common background.

However, *2001* is still a film, and films must sometimes sacrifice realism for aesthetic appeal. For example, the original design of the Discovery was reminiscent of a dragonfly, with the engine surrounded by triangular radiator panels needed to dissipate the extra heat generated by the propulsion systems. However, these radiator panels were soon removed from the design, as they did not line up with the intended vision for the spaceship [43]. Other elements were changed or added to increase dramatic impact as well.

While the film's dedication to scientific rigour is laudable, when creating a work of fiction, science can only take you so far. Realism is useful up to a point because it lends a natural plausibility to the design. However, beyond this point, this strict adherence to realism becomes superfluous, as so few people in the general population have enough knowledge of aeronautics to recognize a truly functional design. Also, adherence to current scientific and technological laws greatly limits the design space, and unnecessarily inhibits the creativity and aesthetics of a design.

Where science is put first, aesthetics can suffer. *2001*'s ships were criticized by *Star Wars* model builders as lacking interconnectedness, and that the surface details "often don't really relate to each other" [72]. However, it is unlikely that this perceived lack of connectivity was due to the prioritization of scientific rigour; since ship designer Harry Lange and art director Tony Masters were not designing and building actual spaceships, it is unlikely that individual surface details were chosen and placed with a specific purpose in mind. Furthermore, after studying the surface details of these ships, we have found no foundation for these accusations, aside from the usual criticisms that inherently arise between members of the same field.

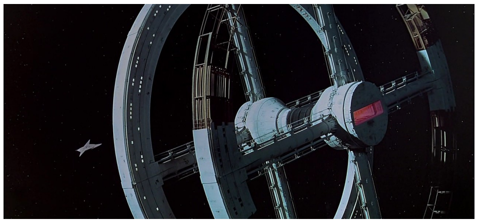


Figure A.23: The wheel shaped Space Station V from *2001: A Space Odyssey*

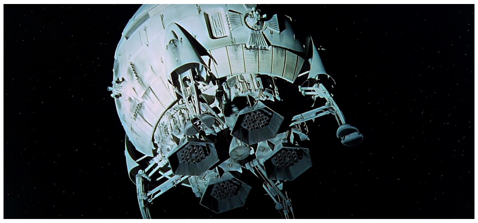


Figure A.24: The Aries IB shuttle from *2001: A Space Odyssey*

Figure A.25: The Orion III shuttle from *2001: A Space Odyssey*

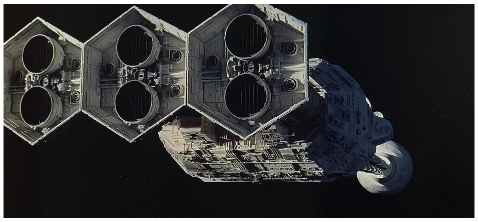


Figure A.26: The Discovery I spacecraft from *2001: A Space Odyssey*

Figure A.27: Screenshots of spaceships from Stanley Kubrick's *2001: A Space Odyssey*.

A.5 Star Wars

The *Star Wars* films introduced such a variety of detailed models, of such a grand scale, that the face of space opera, and science fiction as a whole, would be changed forever. The first *Star Wars* film, Episode IV: A New Hope, was released in 1977. Episodes V and VI followed at three year intervals, appearing in 1980 and 1983, respectively. The first three episodes, referred to as prequels, were also released at three year intervals, in 1999, 2002, and 2005.

Star Wars ushered in a new era for spaceship design; where *2001: A Space Odyssey* had popularized the kitbashed model look, *Star Wars* set a whole new standard, introducing a new vision of the future. Where previous film and television shows depict an optimistic role for futuristic technology, in the *Star Wars* universe, technology is depicted in a more cynical light. This "used future" is a place where real people live and struggle with day to day life; technology has become ubiquitous, and is available to the civilized and uncivilized alike [4]. Spaceships are treated like cars, in that they reflect personal tastes, they require maintenance, and they run on hard work, not magic.

Star Wars spaceships are distinguished by their heavy use of kitbashing and high level of detail. While the original episodes did not have the luxury of computer modeling to represent their spaceships, even the spaceships from the prequels were first built as physical, kitbashed models. Though they were later digitized and represented as computer models, designers felt that first building them as physical models made their ships more plausible by firmly planting the design within some limitations of physics [72]. The *Star Wars* style favors dull, worn surfaces, though this was a requirement, not a choice, for the original films, as blue screen compositing did not work well with shiny surfaces [72].

The original films feature two major forces: the Galactic Empire and the Rebel Alliance. The Galactic Empire, firmly in place by Episode IV, is an evil totalitarian system and is a clear symbol of oppression in the films. This is conveyed in their spaceship designs in various ways. All ships of the Imperial Navy (the Empire's starfleet) are the same standard grey color, providing a degree of uniformity to the fleet. They tend to be well maintained, sleek, and generally clean, especially when compared to the crude, grimy, mismatched assortment of ships that make up the opposing Rebel Alliance. Their large spaceships tend to take size to the extreme; the Death Star was supposedly the size of a small moon, and the Super Star Destroyers reach nineteen kilometres in length.

The Rebel Alliance are the resistance force dedicated to removing the Galactic Empire from power and restoring the former glory of the galaxy. Their ships come in a wide variety of designs and colors and often have mismatched panels and exposed parts. They are always dirty and scratched, and look like they are barely holding together. This style is suited to a band of rebels, as they do not have the resources to afford to waste a single scrap of metal. Many Rebel ships feature smooth, organic-inspired shapes to differentiate themselves from the harsh, sharp forms favored by the Empire.

The following sections examine some of the famous classes of spaceship featured in *Star Wars*.

A.5.1 Millennium Falcon

Star Wars's famous smuggler ship, the Millennium Falcon is the epitome of the souped-up junker. The original design for this ship was discarded for being too similar to the Eagle from *Space: 1999* and later became the Blockade Runner, so the new Falcon, as told by George Lucas, was based on his favorite food, a hamburger with an olive on the side [4]. The Falcon is a variation on the traditional saucer shape, but with the familiarizing addition of scale and direction cues, notably the pincer shaped front and recognizable details such as cockpit windows (though located in an unconventional place) and satellite dish. Note that these details are the most prominent features that are asymmetric. In this case, the asymmetry makes the Falcon appear more rugged and pieced together. The propulsion system is exotic and is represented by a glowing strip at the rear of the ship, with no traditional thrusters in sight. Luckily, the forward direction is clearly shown. An image of the Millennium Falcon and a close-up of its surface details are shown in Figures A.28 and A.29, respectively.

Figure A.28: The Millennium Falcon

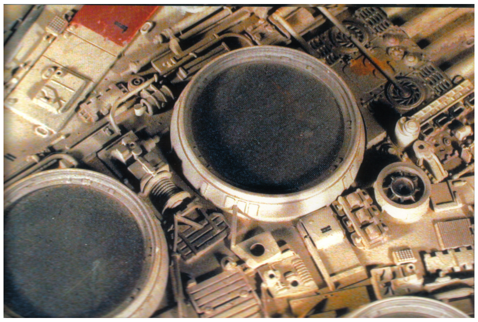


Figure A.29: A close-up of the top of the Millennium Falcon.

A.5.2 TIE series

The TIE fighter is the standard single-person fighter of the Galactic Empire. Though TIE fighters are built for maneuverability and speed, they have no visible means of propulsion, despite the Twin Ion Engines acronym. They are powered by the solar panels that cover their hexagonal wings, though the term "wing" is used loosely here. Scale and direction are indicated by the large round cockpit window on the front of the spherical cockpit. For a fighter ship, the weapons systems are surprisingly unobtrusive, represented by two small cylinders beneath the window. The combination of these features results in a high-tech appearance, though not especially aggressive for an Imperial fighter. In this case, most of the threatening aspect of the ship is added by the sound of its engines, which of course is not a part of the design of the ship itself, and the fact that they always appear in large numbers.

TIE fighters come in a variety of designs, usually with some alterations to the wing style. Darth Vader's personal TIE fighter, altered to make his ship easily recognizable from the others on screen, has wings that are bent inward towards the cockpit. Other variations include the TIE interceptor, which has forward pointed wings, giving it a sharp and aggressive look that was lacking in the original fighter, and the TIE bomber, whose design is an updated version of Darth Vader's TIE fighter that includes an additional spherical attachment to the cockpit. Other variations that feature three equally distanced wings also exist. Notably, the design of Darth Maul's Sith infiltrator is a descendant of the TIE fighter as well, but features the spherical cockpit elongated to a point [72].

The TIE series of ships were named for their resemblance to bow ties, not the other way around; the shape of the TIE fighter's frame was not inspired by the shape of a bow tie [4]. An image of a TIE fighter and a close-up of the surface details of Darth Vader's TIE fighter are shown in Figures A.30 and A.31, respectively.

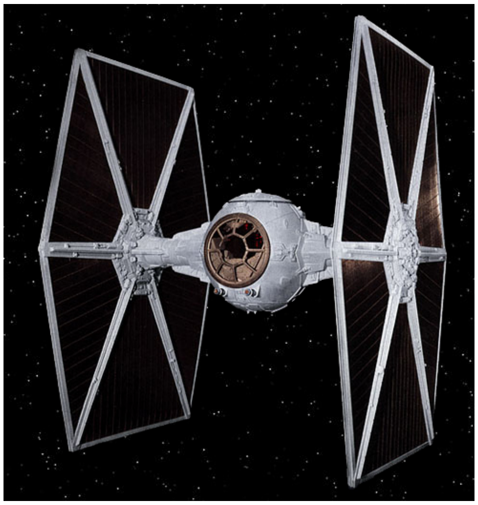


Figure A.30: An Imperial TIE Fighter

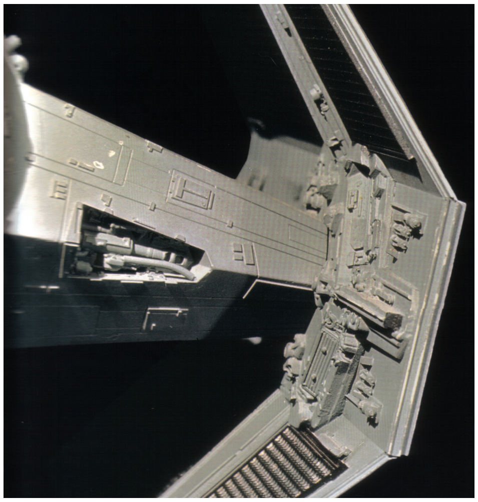


Figure A.31: A close-up of the wing of Darth Vader's specialized TIE fighter

A.5.3 X-Wing Fighter

The X-Wing single person fighter is the Rebel Alliance's response to the TIE fighter. The X-Wing is an innovative update of the traditional fighter plane design that incorporates folding, X-shaped wings that lend the ship its name. The X-Wing has a clear sense of direction, emphasized by the four long, pointed weapons mounted on the four corners of the X. These ships are rugged, and often show burns and cracks from previous expeditions. Other than this, the X-Wing has few surface details aside from its rectangular panels. The X-Wing has a traditional propulsion system, with four thrusters arranged in a square at the rear of the ship. The X-Wing fighter is shown in Figures A.32 and A.33.

A.5.4 Death Star

The Death Star was the ultimate weapon of the Galactic Empire. A space station immense in size, described as comparable to a small moon, the Death Star boasted enough firepower to

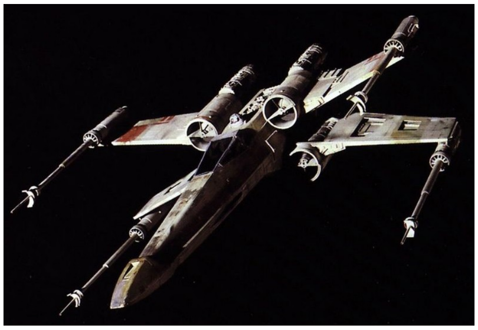


Figure A.32: A Rebel Alliance X-Wing fighter with wings open in attack position

Figure A.33: A close-up of two X-Wing fighters

destroy an entire planet and is truly a tribute to the manpower, resources, and arrogance of the Empire.

The frame of the Death Star is a simple sphere with a round concavity. This complete lack of directionality is fitting for a space station and makes the comparison between the Death Star and the enormity of a moon or planet that much easier.

While the frame of the Death Star is effective, it is still simple, and it is the surface details of the Death Star and its trench that are most interesting. An ambitious kitbashing project, the Death Star trench took months of work to create. The surface details of the trench are described in more detail in Section 3.3.2.

The design of the Death Star II is also worth noting, if only for its interesting relation to the plot of the films. While spaceships (and stations) have been portrayed as under construction before (notably the wheel shaped Space Station V from *2001: A Space Odyssey*), the Death Star II was merely designed to *appear* unfinished, though still menacing, to fool the Rebel Alliance into thinking that it was not yet operational. The Death Star II features roughly the same shape and size as the original, but leaves large sections of exposed structure.

Various images of the Death Star and its surface details are shown in Figures A.34 through A.37.

A.5.5 Imperial Star Destroyers

The Imperial Star Destroyers are the command ships of the Galactic Empire, known for their immense size and incredible firepower. The Star Destroyer's triangular shape gives it an exaggerated sense of direction, and the multitude of pinpoint lights (which represent windows and create a modification of a city's lights) and tiny surface details express its imposing size. This sense of scale is the most important aspect of its design, creating an illusion of power and oppression.

This spaceship has a naval design, reminiscent of the destroyer that lends it its name, especially with the high conning tower. The surface does not have true panels per say, though a variety of divisions lightly decorate the surface, reminiscent of a handmade line drawing (which were actually etched in by hand with an X-Acto knife). Kitbashed surface details are densely packed and confined to vertical faces of the ship. The Star Destroyers are the stark, standard grey of all

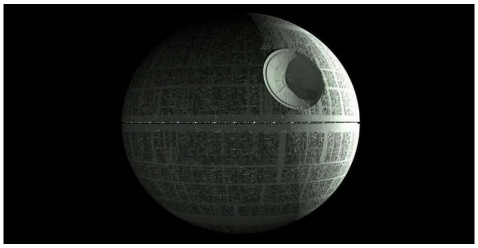


Figure A.34: A full shot of the Death Star

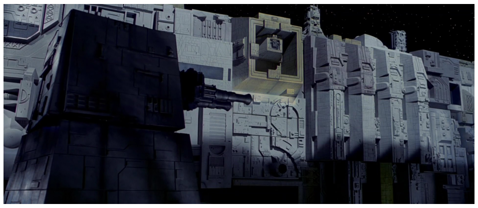


Figure A.35: Inside the Death Star trench

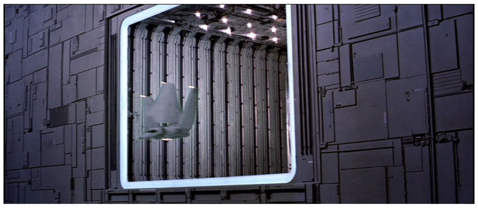


Figure A.36: A docking bay opening on the side of the Death Star

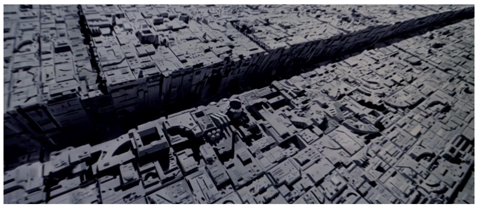


Figure A.37: An overhead shot of the Death Star trench

Empire vessels [72]. An image of a Star Destroyer and a close-up of a conning tower are shown in Figures A.38 and A.39, respectively.

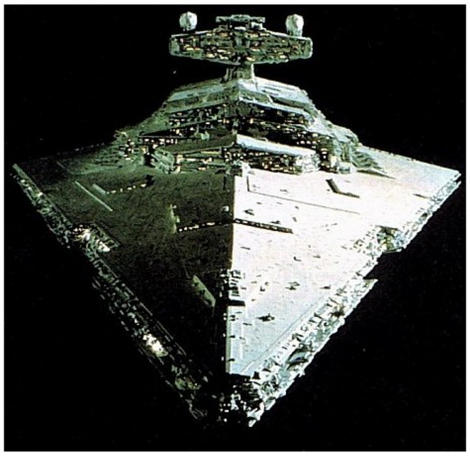


Figure A.38: An Empirial Star Destroyer

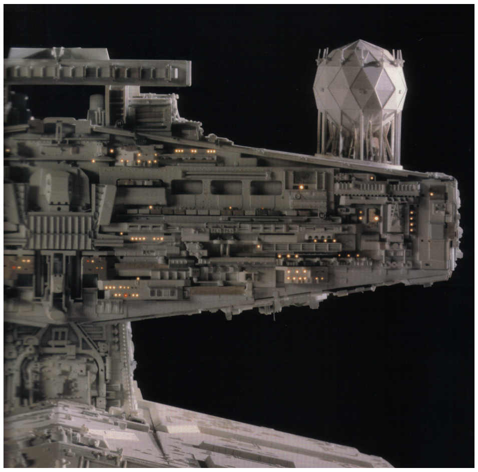


Figure A.39: A close-up of the kitbashed details of the Star Destroyer's conning tower

A.5.6 Other Star Wars Ships

Other *Star Wars* spaceships that did not play a central role in the films or whose design origins are less detailed are featured in Figures A.40 through A.49.

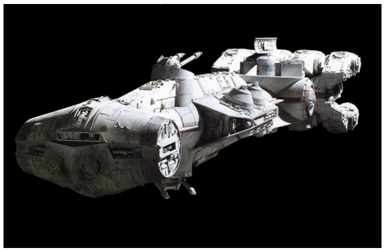


Figure A.40: The Rebel Blockade Runner featured in Episode IV. This design was originally intended to serve as the Millennium Falcon and is vaguely rocket shaped. The focal point of the ship is clearly the numerous thrusters; the designers make it clear that this spaceship can really move [72]. The hammer-shaped nose also contributes to the ship's sense of direction.

Figure A.41: First appearing in Episode VI, The Mon Calamari Star Cruisers were referred to by the modelers as the "pickle ships", and for obvious reasons: the rounded frame is covered with a variety of wart-like bumps that constitute the only surface details. One of the few *Star Wars* spaceships that was organic, and hence not kitbashed, the Mon Calamari Cruiser was designed to be the antithesis of the Imperial Star Destroyer [72].

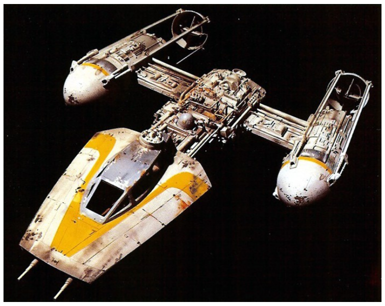


Figure A.42: The Y-Wing fighters, named for their obvious shape, are the work horses of the Rebel Alliance fleet. Only the front cockpit is sleek and smooth, the rest of the ship is a dense jumble of kitbashing. This design was meant to create the appearance of a stripped hot rod, full of exposed parts.

Figure A.43: The Rebel Transport, while not strictly an organic spaceship, features a soft, rounded shape and looks strikingly like a large worm. The surface of the ship looks similar to a shell, which is reinforced by the unfinished front lip that looks like a soft underbelly. The transport has nothing to indicate its scale, though it is described as being ninety metres in length [4].

Figure A.44: Bounty Hunter Boba Fett's Slave I personal spaceship looks much like its owner, with a color scheme almost identical to his armor. Though often compared to a street lamp, the frame of the ship was originally based on a radar dish. The scale of the ship is evident from its large cockpit window, but the direction of the ship is problematic. The spaceship changes orientation when it lifts off, and the thrusters that originally pointed downwards are rotated ninety degrees to propel the ship forward [72]. While this makes for an interesting take-off scene, it conflicts with the natural assumption that the flat side of the ship should face down.

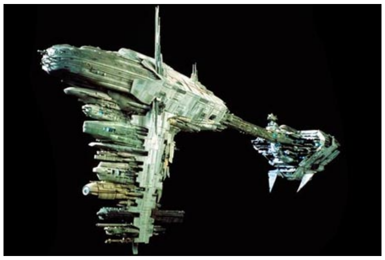


Figure A.45: The design of the Rebel Medical Frigate was inspired by an outboard motor, and its jutted front, almost like a cliff face, is created from kitbashed hulls of various ships and submarines [72]. The rear of the ship, consisting of a blocky, triangular end connected by a long, thin shaft, appears to be inspired by *2001*'s Discovery.

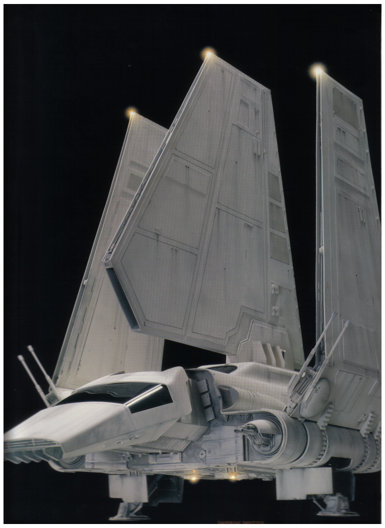


Figure A.46: The Imperial shuttle has a sleek and graceful design, especially for an Imperial spaceship. The outer wings unfold during flight, giving the shuttle the overall impression of an origami swan [72].

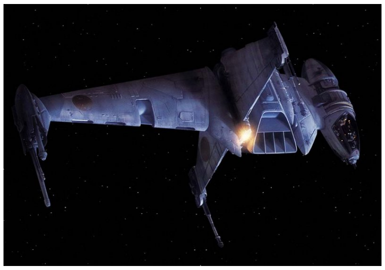


Figure A.47: The B-Wing fighter features a non-traditional T-shaped frame. Like the Slave I, the B-Wing fighter can rotate, making its directional information confusing. The side placement of the cockpit was inspired by that of the Millennium Falcon, which is similarly placed [72].

Figure A.48: The Trade Federation Battleship, which appeared in the prequels, was inspired by the Star Destroyers of the original films. The ship's enormous size and nearly radially symmetric frame give it the appearance of a space station rather than a battleship. The single break in symmetry in the outer ring adds to rather than detracts from this, as it creates the appearance of an entrance. The design was intended to be an updated saucer, and features the traditional *Star Wars* kitbashing and city of lights effect.

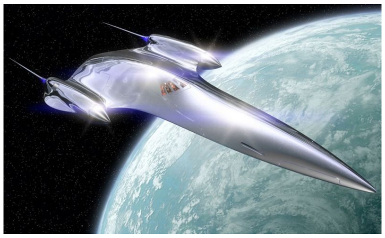


Figure A.49: The Naboo Royal Starship is featured in the prequels, and its smooth, shiny surface embodies a more peaceful and civilized time. Its elegance and craftsmanship makes it clear that this ship is not owned by any ordinary person. Aside from irregularly shaped panels that split the surface of the ship and the small windows that indicate its scale, this design is devoid of any surface details.

A.6 Babylon 5

The TV series *Babylon 5* first aired in 1994 and continued for five seasons. The series revolves around the daily life of the crew and guests aboard Babylon 5, the Earth Alliance space station. The station's purpose is diplomatic, allowing members of different races to meet and negotiate in neutral territory.

Babylon 5 is the earliest example of a science fiction show with spaceships created entirely using computer models. Because all this, *Babylon 5* demonstrates the significant change that the departure from physical models afforded. Computer models allowed for an incredible diversity and flexibility in the design of the frame, but almost entirely replaced surface details with busy textures. One reason behind this replacement was the extreme time constraints placed on the designers and modelers, but technological limitations also played a part [66].

Each of the primary alien races had a distinct style of ship, based on their defining characteristics [66]. This was important to ensure that different ships were instantly recognizable by the viewer.

Babylon 5 ships tend to have a poorly defined sense of scale, especially since ships were scaled up and down as needed for dramatic effect, docking, and comparisons relative to other ships. Canon size comparison charts were created for the show during Season 4 in an attempt to foster continuity [66]. Ships were designed based on aesthetic appeal, convenience, and storyline, not functionality [66].

A.6.1 Narn

The Narns are a proud people, among the strongest and toughest of the *Babylon 5* races. They value honor and strength above all else, though they are considered one of the most primitive, even barbaric, races. Narns appear reptilian and have a reddish orange complexion. These characteristics are reflected in the designs of their ships, which feature predominantly reddish orange and brown patterns and simple, tribal shapes [66]. Examples of Narn spaceships are shown in Figure A.50.

A.6.2 Earth Alliance

Earth Alliance is the government of Earth and its colonies. Earth Alliance ships are civilian ships, used for exploration, transport, trade, and settlement; Babylon 5 itself is an Earth Alliance station. Because of their wide assortment of uses, Earth Alliance ship designs are varied. Examples of Earth Alliance spaceships are shown in Figure A.51.

A.6.3 Earth Force

Earth Force is Earth Alliance's military force. Earth Force spaceships tend to be boxy and functional, with designs based on naval or aircraft vessels. They also follow a predominantly grey color scheme. Examples of Earth Force spaceships are shown in Figure A.52. The Earth Force Starfury fighters, shown in Figure A.52(a), were influenced by the Gunstar from the film *The Last Starfighter*, shown in Figure A.93 [66].

(a) Narn G'Quan Class Heavy Cruiser from *Babylon 5*

(b) Narn Cruiser from *Babylon 5*

Figure A.50: Spaceships of *Babylon 5*'s Narn Regime

(a) Earth Alliance's Babylon 5 station from the show of the same name.

(b) Earth Alliance Asimov class luxury liner from *Babylon 5*

Figure A.51: Spaceships of *Babylon 5*'s Earth Alliance

(a) Earth Force Starfury Fighter from *Babylon 5*

(b) An Earth Force Omega Class Destroyer, the Agamemnon, from *Babylon 5*. This design was modeled after a steam engine [66]. The centrifuge section of the Agamemnon was modeled after the similar section of *2010: The Year We Make Contact*'s Leonov.

Figure A.52: Spaceships of *Babylon 5*'s Earth Force

A.6.4 Minbari

The Minbari are an old race compared to all but the Vorlons and Shadows, and are thus far more technologically advanced. Minbari ships have an organic, fish-inspired design with surface details similar to bones, reminiscent of the bone structure headpieces that all Minbari have [66]. Examples of Minbari spaceships are shown in Figure A.53.

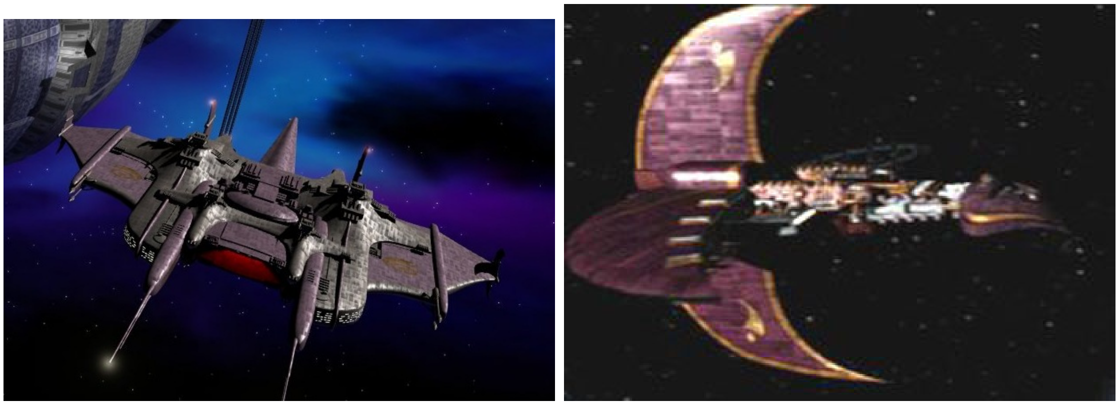
(a) Minbari Sharlin Class Warcruiser from *Babylon 5*. This was the most common type of Minbari ship portrayed in the show.

(b) Minbari Tinashi Destroyer from *Babylon 5*

Figure A.53: Spaceships of *Babylon 5*'s Minbari Federation

A.6.5 Centauri

The Centauri Republic, despite its name, is an imperial culture. The Centauri are a proud people that rely heavily on position and status. Their spaceships, like all things Centauri, tend to be ornate, and it is no surprise that their ships are mostly purple, the color most associated with royalty. Centauri ships tend to feature a crescent-shaped design [66]. Examples of Centauri spaceships are shown in Figure A.54.



(a) Centauri Primus Heavy Cruiser from *Babylon 5*

(b) Centauri Vorchan Destroyer from *Babylon 5*, influenced by the Klingon Bird-of-Prey ship from *Star Trek*

Figure A.54: Spaceships of *Babylon 5*'s Centauri Republic

A.6.6 Shadows

Perhaps the most memorable of the *Babylon 5* races, the Shadows are an old and mysterious race. Their philosophy is a variation of "survival of the fittest", believing that conflict, which they strove to create, leads to adaptation and strength. Their methods of spreading this philosophy included manipulation and stealth. Their ships are invariably a glossy black that is barely visible against the blackness of space, and clearly inspired by a spider or insect.

Shadow ships, and the shadows themselves, were meant to be so far advanced as to be incomprehensible to humans. Thus, their ships have no visibly recognizable systems; there are no obvious propulsion systems, weapons systems, cockpit, or traditional wings. In fact, the Shadow vessels are completely devoid of any surface details, other than a glossy, slightly oily texture. Examples of Shadow spaceships are shown in Figure A.55.

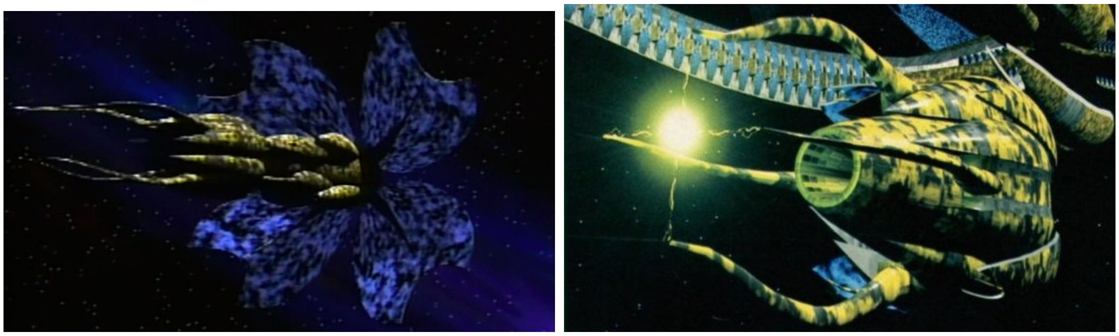
A.6.7 Vorlon

Similar in age to the Shadows and just as secretive, the Vorlons are a formidable force. Where the Shadows enforce "survival of the fittest", the Vorlons seek to protect and guide the younger

(a) Shadow Battlecrab from *Babylon 5* (b) Shadow Fighter from *Babylon 5*

Figure A.55: Spaceships of *Babylon 5*'s Shadows

races, though their cryptic and manipulative methods can be just as suspect. Vorlon spaceships are organic and, to some degree, sentient. Physically, Vorlons appear as winged beings of light, so their technology was meant to appear almost god-like to the younger races. The design appears to be somewhere between a blossoming flower and a squid, and has no visible propulsion system, weapons, wings, or cockpit. Designer Ron Thorton revealed that the Vorlon transport design was actually based on a bulb of garlic, which can be seen in the body of the ship. The coloring of the ships appears similar to camouflage [66]. Examples of Vorlon Spaceships are shown in Figure A.56.



(a) Vorlon Transport from *Babylon 5* (b) Vorlon Fighter from *Babylon 5*

Figure A.56: Spaceships of *Babylon 5*'s Vorlon Empire

A.6.8 Other Babylon 5 Ships

A few other *Babylon 5* spaceships that do not fall into one of the main categories are shown in Figures A.57 through A.60.

Figure A.57: A Drakh Marauder Heavy Fighter from *Babylon 5*. The design was inspired by an insect stinger, and was purposely given a distinct sense of direction [66].

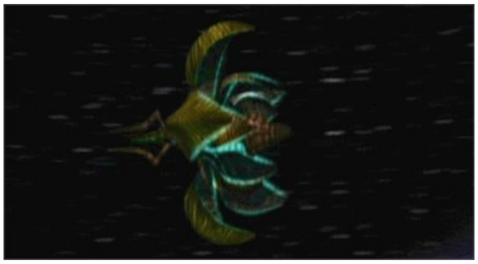


Figure A.58: A Drazi Sunhawk from *Babylon 5*. The design is an obvious nod to *Blake's 7*'s Liberator, which designer Ron Thornton also worked on in the show's final year [66].

Figure A.59: The Interstellar Alliance's White Star from *Babylon 5*. The White Star is a combination of Minbari and Vorlon technology, and the design is influenced by the pelican [66].

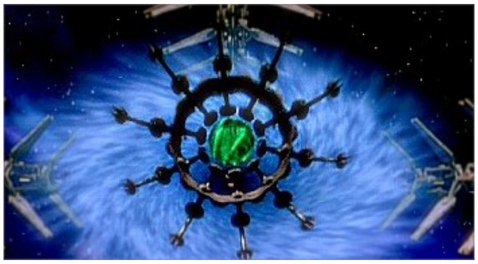


Figure A.60: A Brakiri Dikati Transport from *Babylon 5*. The design evolved from incorporating a sphere that would not be mistaken for *Star Wars*' Death Star.

A.7 Battlestar Galactica

The original *Battlestar Galactica* television series first aired in 1978, but fans were treated to a remake in 2004. *Battlestar Galactica* and its remake provide an interesting opportunity to compare the styles of physical, kitbashed models, seen in the original, to the entirely digital models seen in the new version.

Released just a year after the first *Star Wars* film, the original *Battlestar Galactica*'s ship designs appear to be heavily influenced by the "used future" style. Not surprisingly, the two shared the same Visual Effects Supervisor, John Dykstra [72, 5]. The models are heavily kitbashed, especially the Galactica itself.

As seen in *Babylon 5*, the introduction of computer models leads to spaceship frames that are uninhibited; curved surfaces that were difficult to create as physical models are predominant. As spaceships central to the plot of the show have clearly evolved from their original counterparts, this freedom is instead expressed in the civilian ships, which feature complex, rounded surfaces. However, the models in general tend to have fewer surface details than their originals, though this is balanced by new and interesting frame shapes.

Though the plots of the original and the re-imagining differ slightly, the spaceships from both shows consist of three groups: human military vessels, made up of the last surviving human warship Battlestar Galactica and its support ships, human civilian vessels consisting of whatever ships and survivors managed to escape from their homeworld, and the Cylon vessels, the spaceships of a race of robotic (or human-machine hybrids) warriors that were at war with the remaining humans.

The human military spaceships are traditional in design and in keeping with the military theme. The Vipers are shaped and colored like traditional fighter jets.

Cylon ships are shinier and sleeker than the utilitarian design of the human ships, especially in the remake. The new Cylon Raiders have clearly been updated to look sharper and more aggressive then the originals.

Images of both the original and remake versions of the Battlestar Galactica, Colonial Viper, and Cylon Raider are shown in Figures A.61 through A.66. Two civilian ships from the remake are shown in Figures A.67 and A.68.

Figure A.61: The 1978 version of Battlestar Galactica

Figure A.62: The 2004 version of Battlestar Galactica

Figure A.63: The 1978 version of the Colonial Viper

Figure A.64: The 2004 version of the Colonial Viper

Figure A.65: The 1978 version of the Cylon Raider

Figure A.66: The 2004 version of the Cylon Raider

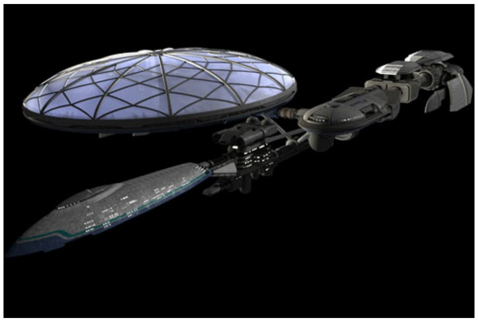


Figure A.67: A civilian ship from the 2004 *Battlestar Galactica*

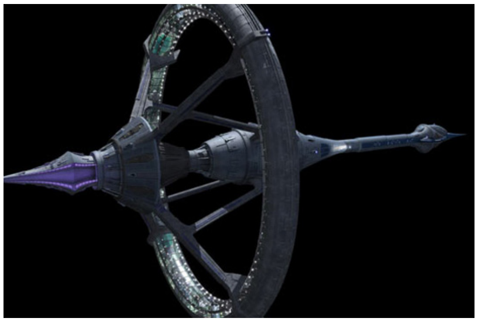


Figure A.68: A civilian ship from the 2004 *Battlestar Galactica*

A.8 Video Games

This section covers spaceship designs featured in video games, excluding video games based on film or television shows, as these spaceships are already covered. Since this survey focuses on spaceships with a high level of visual detail, old games with low quality graphics are not included.

A.8.1 Eve Online

Eve Online is a Massively Multiplayer Online Role Playing Game (MMORPG) that was first released in 2003. The ships of *Eve Online* function as the player avatars, representing their presence in the game world. While ships can be customized in terms of skills and performance, their appearance is not yet customizable. The *Eve Online* world consists of five major races of humans: the Amarr Empire, the Gallente Federation, the Caldari State, the Minmatar Republic, and the Jovian Empire. Only the first four are available for player characters. Each race has a unique approach to spaceship design that reflects their background and values. Though each empire has its own preferences and patterns for spaceship design, the same basic types of spaceship exist for each to encourage players to distribute themselves evenly among the empires and to avoid giving one an unfair advantage in the game [19].

Once enslaved by the Amarr Empire, the Minmatar are a tough and determined people, and now prize independence over all else. The Minmatar are an extreme example of the "used future" design principle, with many of their ships appearing pieced together from old, reused parts. Their spaceship frames are often asymmetric, and their weapons systems are crude [19].

The Amarr Empire are a highly religious people, and view themselves as the most powerful race in EVE, controlling forty percent of the inhabited worlds. The Amarr Empire's spaceship frames are borderline organic, favoring rounded, bulbous shapes. Their ships tend to be sleek, and shiny, using a gold color scheme that exemplifies their status [19].

The Caldari State is a capitalist society, controlled by large corporations that keep their economy strong. They are an aggressive, militaristic society and are feared and admired by the other races. The Caldari prefer function over form, and their ships tend to be utilitarian and boxy, favoring more traditional naval and aircraft forms [19].

The Gallente Federation is the only democracy of all the empires, valuing freedom and progression. The Gallenteans produce much of the world's entertainment, including the most luxurious ships. Similar to the Amarr Empire, Gallentean ships tend to be smooth and rounded, but have less uniformity in their design [19].

Images of Battlecruiser, Frigate, and Titan class spaceships for each of the four player races are shown in Figures A.69 through A.80.

Figure A.69: An Amarr Battlecruiser from *EVE Online*

Figure A.70: A Gallente Battlecruiser from *EVE Online*

Figure A.71: A Minmatar Battlecruiser from *EVE Online*

Figure A.72: A Caldari Battlecruiser from *EVE Online*

Figure A.73: An Amarr Frigate from *EVE Online*

Figure A.74: A Gallente Frigate from *EVE Online*

Figure A.75: A Minmatar Frigate from *EVE Online*

Figure A.76: A Caldari Frigate from *EVE Online*

Figure A.77: An Amarr Titan from *EVE Online*

Figure A.78: A Gallente Titan from *EVE Online*

Figure A.79: A Minmatar Titan from *EVE Online*

Figure A.80: A Caldari Titan from *EVE Online*

A.8.2 Homeworld 2

Homeworld 2 is a real time strategy game, released in 2003. The two most prominent races in the game are the Hiigarans, controlled by the player, and the Vaygr, the opponent.

The Hiigarans seek to protect their new-found homeworld from the new threat posed by the Vaygr. Hiigaran ships are easily identified by their blue and gray color scheme. Two Hiigaran ship designs are shown in Figures A.81 and A.82.

The Vaygr are a nomadic, warrior race that scour the galaxy in search of new races, which they seek to overpower to amass new technology. The Vaygr ships are generally fast, light, and agile, consistent with the race's wandering tendencies. Their ships all feature a black and white striped color scheme that usually incorporates a splash of red. Two Vaygr ship designs are shown in Figures A.83 and A.84.

Figure A.81: The Hiigaran Mothership

Figure A.82: A Hiigarian carrier surrounded by a few interceptors.

Figure A.83: A Vaygr Minelayer Corvette

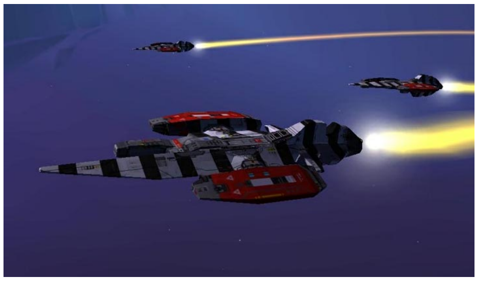


Figure A.84: A Vaygr Bomber

A.8.3 Sins of a Solar Empire

Sins of a Solar Empire is a real time strategy game released in 2008. Screenshots from the game are shown in Figure A.85 and Figure A.86.

Figure A.85: Screenshot from *Sins of a Solar Empire*

Figure A.86: Screenshot from *Sins of a Solar Empire*

A.8.4 Halo

Halo is a series of first-person shooters that began with the release of *Halo: Combat Evolved* in 2001, and now includes three other games, as well as a real-time strategy spin-off, *Halo Wars*. The Halo universe consists of two major players: The Covenant Empire and the United Nations Space Command (UNSC). The Covenant Empire are the alien enemy that has waged war against humanity, which is in turn protected by the UNSC Marine Corps. The Covenant are made up of a variety of alien species, but their ships tend to follow a common design. All Covenant ships are purple and are often shaped like weapons. They feature rounded, bulbous forms, but are actually mechanical. Two Covenant ship designs are shown in Figure A.87.

UNSC ships tend to be modeled after traditional vehicle shapes, as is the common theme in traditional human-versus-alien scenarios. Their large spaceships are naval inspired, while small craft are based on traditional fighter planes, though often slightly modified to resemble the animal they were named for. Two UNSC ship designs are shown in Figure A.88.

(a) *Halo*'s Covenant Destroyer

(b) *Halo*'s Covenant Assault Carrier

Figure A.87: Spaceships of *Halo*'s Covenant Empire

(a) *Halo*'s UNSC Phoenix Class Spirit of Fire

(b) *Halo*'s UNSC Longsword Interceptor has a manta ray profile when seen from below

Figure A.88: Spaceships of *Halo*'s United Nations Space Command

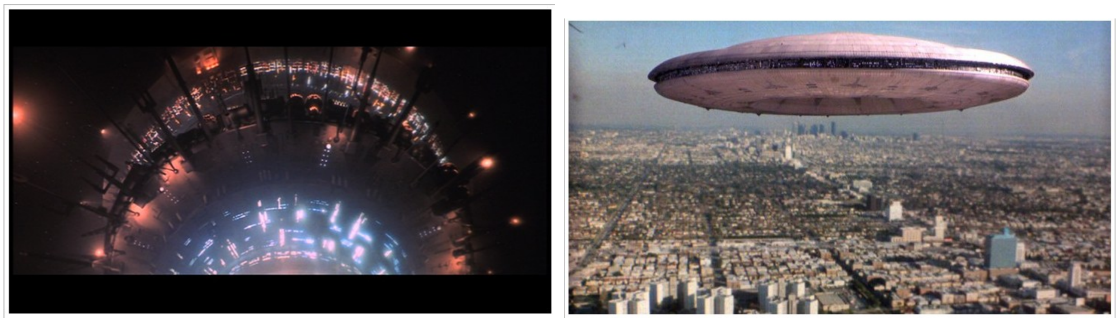
A.9 Others

A.9.1 Reincarnations of the Saucer

Films and television shows that follow traditional alien invasion or alien encounter story lines tend to favour an updated version of the classic saucer design. Some new versions of the saucer are shown in Figure A.89.

(a) Saucer from the 1996 film *Independence Day*

(b) A Vree saucer from the television show *Babylon 5*



(c) Saucer from the 1977 film *Close Encounters of the Third Kind*

(d) Saucer from the 1983 miniseries *V: The Final Battle*. The 2009 remake also features a similar saucer design.

Figure A.89: Various updated versions of the classic saucer

A.9.2 Alien

The first *Alien* film was released in 1979. Though the film is famous for its alien designs by renowned artist H.R. Giger, it is the mining ship, the Nostromo, that is of interest for this work.

The "used future" style is evident in the Nostromo's design; the ship is industrial and rugged, with a high density of surface details. The Nostromo, shown in Figure A.90, was built as a physical model with kitbashed details. This similarity in styles is not surprising considering the film's Visual Effects Supervisor, Brian Johnson, also held the same position for *Star Wars Episode V* [72, 7].

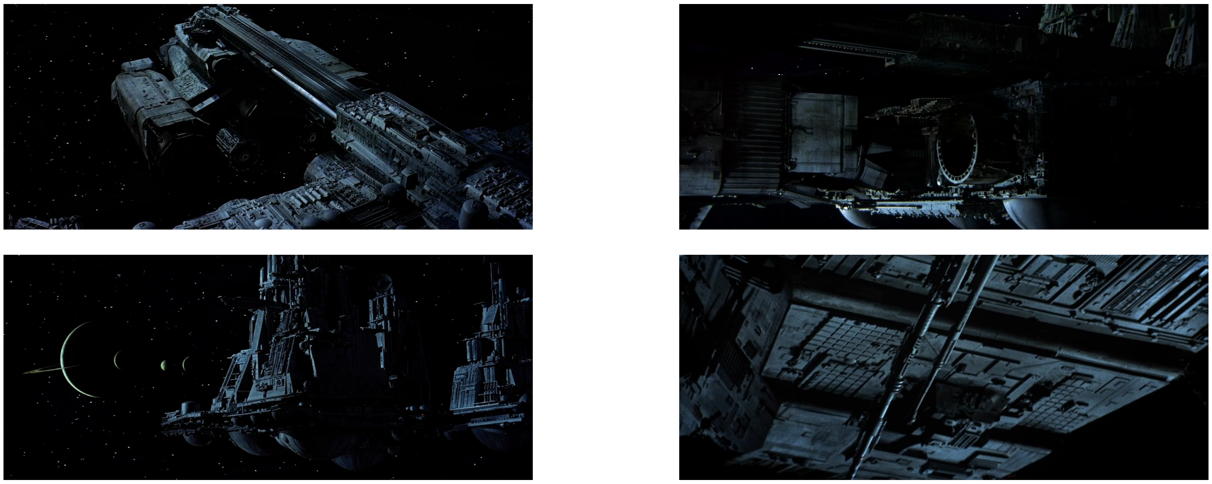


Figure A.90: Screenshots of the ship Nostromo from the 1979 film *Alien*

A.9.3 Red Dwarf

BBC's science fiction comedy *Red Dwarf* first aired in 1988. The spaceship designs featured in the show are in keeping with its campy humor; they are brightly colored and non-threatening, with an inclination towards the obvious. For example, the Starbug shown in Figure A.91(c) looks just like a large green insect. Other *Red Dwarf* spaceships are shown in Figure A.91.

(a) The titual spaceship from the 1988 BBC series *Red Dwarf*.

(b) A close-up of some of the surface details of the Red Dwarf.

(c) *Red Dwarf*'s Justice World spaceship. Many of the spaceships for this show were designed with a literal interpretation, befitting of a comedy.

(d) *Red Dwarf*'s Starbug, clearly modeled after some sort of ant or other small insect, has a bright, happy, and decidedly non-threatening design.

Figure A.91: Spaceships from the 1988 BBC science fiction comedy show *Red Dwarf*

A.10 Film: Honorable Mention

Figure A.92: The titular spaceship from the 1997 film *Event Horizon*. The fittingly Gothic design of the ship follows from the film's "ghost story in space" plot.

Figure A.93: The Gunstar spaceship from the 1984 film *The Last Starfighter*. This was one of the first films to use largely computer models for its spaceships, including the Gunstar.

A.11 Television: Honorable Mention

Figure A.94: Despite the fact that it is modeled after a wingless dragonfly, the titular spacecraft from the 1997 television show *Lexx* is actually a powerful and specialized weapon, designed to destroy planets. The organic Lexx ship is interesting because it is actually a living, sentient organism.

Figure A.95: The aptly named Firefly class spaceship Serenity, featured in the 2002 television show *Firefly*. The frame features a distinctive head-shaped front, wing-like structures, and a bulbous, lighted abdomen. The story's space western theme is reflected in the rugged design of the ship: panels are oddly shaped and often mismatched, and the exterior is worn and used.

Figure A.96: *Futurama*'s Planet Express has a simple, retro design. Note that the ship, like many of the 1999 show's characters, has a distinctive overbite.

Appendix B

Summary of Spaceship Attributes

We have summarized the features of each spaceship discussed in Appendix A in the following table. We include the name of the spaceship (including the film or show it first appeared in), its figure in Appendix A, the model representation (computer or physical), the spaceship type (organic or mechanical), the scale (single-person, medium, or large), the symmetry of its frame, the technological representation (new or used future), race (human or alien), any confirmed or speculative inspirations for the design, the visible functional components (in order of prominence), and the general shape of the panels.

Unless explicitly stated, we are referring to spaceships from the original version of films and television shows, as opposed to remastered versions that may feature new digital models. This is relevant for many older entries, such as Star Trek, Star Wars, and Red Dwarf.

Multiple entries may be given for a field if its value is unclear. The *Scale* field is the most likely to feature multiple entries, as a lack of visual cues can result in ambiguity. Similarly, "Unknown" may be assigned to this field if the size of a ship cannot be determined visually. Often the same model may be used to represent multiple scales, especially in the case of computer models, making it impossible to assign a single scale.

For fields whose values are selected from a small set, we make use of abbreviations, shown in Table B.1. These fields include *Model*, *Type*, *Scale*, *Symmetry*, *Technology*, and *Race*.

Field	**Abbreviations**		
Model	Physical (Phys)	Computer (Comp)	
Type	Mechanical (Mech)	Organic (Orgn)	
Scale	Single-person (S)	Medium (M)	Large (L)
Symmetry	Bilateral (B)	Radial (R)	Assymetric (A)
Technology	New future (New)	Used future (Used)	
Race	Human (H)	Alien (A)	
Any	Unknown (UNK)		

Table B.1: Field abbreviations used in Table B.2

Table B.2: Summary of Spaceship Attributes

Name	Figure	Model	Type	Scale	Symmetry	Tech	Race	Inspiration	Funct. Comp.	Panel Shape
Rocket (Various sources, 1950s)	A.1	Phys	Mech	M	Radial	New	H	Current technology	Cockpit, propulsion (traditional), wings	Quadrilateral
Saucer (Various sources, 1950s)	A.2	Phys	Mech	M, L	Radial	New	A	Sightings	None	Variable
USS Enterprise (*Star Trek*, 1966 -2009)	A.9(a)	Phys	Mech	L	Bilateral	New	H	Hope, liberty, peace	Propulsion (exotic), cockpit	Quadrilateral
Federation Defiant Class starship (*Star Trek: Deep Space 9*, 1993)	A.10(a)	Phys	Mech	M, L	Bilateral	New	H	Unique, advanced	Propulsion (exotic)	Complex
Federation Olympic Class starship (*Star Trek: The Next Generation*, 1987)	A.10(b)	Phys	Mech	L	Bilateral	New	H	Previous Federation starships	Propulsion (exotic), cockpit	Roughly quadrilateral
Federation Constellation Class starship (*Star Trek: The Next Generation*, 1987)	A.10(c)	Phys	Mech	L	Bilateral	New	H	Previous Federation starships	Propulsion (exotic), cockpit	Roughly quadrilateral
Federation Nova Class starship (*Star Trek: Voyager*, 1995)	A.10(d)	Comp	Mech	L	Bilateral	New	H	Previous Federation starships	Propulsion (exotic), cockpit	Roughly quadrilateral
Vulcan D'Kyr Class starship (*Star Trek: Enterprise*, 2001)	A.11	Comp	Mech	L	Bilateral	New	A	Vulcan architecture and clothing	None	Complex, curved
Vulcan Suurok Class starship (*Star Trek: Enterprise*, 2001)	A.12	Comp	Mech	L	Bilateral	New	A	Vulcan architecture and clothing	None	Complex
Romulan Bird-of-Prey (*Star Trek: Enterprise*, 2001)	A.13	Comp	Mech	M	Bilateral	New	A	Bird	Wings, propulsion (exotic)	Complex
Romulan D'deridex Class Warbird (*Star Trek: The Next Generation*, 1987)	A.14	Phys	Mech	L	Bilateral	New	A	Bird	Wings, cockpit, propulsion (exotic)	Complex
Klingon Bird-of-Prey (*Star Trek III: The Search for Spock*, 1984)	A.15	Phys	Mech	M	Bilateral	Used	A	Bird, hawk	Wings, cockpit, propulsion (exotic)	Complex

Continued on next page. . .

Table B.2 – Continued

Name	Figure	Model	Type	Scale	Symmetry	Tech	Race	Inspiration	Funct. Comp.	Panel Shape
Klingon Vor'cha class heavy cruiser (*Star Trek: The Next Generation*, 1987)	A.16	Phys	Mech	M	Bilateral	Used	A	Bird	Wings, cockpit, propulsion (exotic)	Complex
The Borg Cube (*Star Trek: The Next Generation*, 1987)	A.17	Phys	Mech	L	Radial	Used	A	Cube	None	None
The Borg Renegade spaceship (*Star Trek: The Next Generation*, 1987)	A.18	Phys	Mech	M	Asymmetric	Used	A	UNK	None	Complex
Galor Class starship (*Star Trek: The Next Generation*, 1987)	A.19	Phys	Mech	L	Bilateral	Used	A	Ankh (unconfirmed)	Wings, propulsion (exotic)	Quadrilateral
Breen Warship (*Star Trek: Deep Space 9*, 1993)	A.20	Comp	Mech	UNK	Asymmetric	Used	A	UNK	None	Complex
Bajoran lightship (*Star Trek: Deep Space 9*, 1993)	A.21	Comp	Mech	M	Bilateral	New	A	Sailboat	Wings, cockpit	Quadrilateral
Deep Space 9 station (*Star Trek: Deep Space 9*, 1993)	A.26	Phys	Mech	L	Radial	Used	A	Gyroscope, nucleus of an atom	None	Complex
Space Station V (*2001: A Space Odyssey*, 1968)	A.23	Phys	Mech	L	Radial	Used	H	Wheel	None	Quadrilateral
Aries IB shuttle (*2001: A Space Odyssey*, 1968)	A.24	Phys	Mech	M	Radial	Used	H	Apollo Lunar Module	Cockpit, propulsion (traditional)	Quadrilateral
Orion III shuttle (*2001: A Space Odyssey*, 1968)	A.25	Phys	Mech	M	Bilateral	New	H	Sänger antipodal bomber	Propulsion (traditional), cockpit, wings	Quadrilateral
Discovery I spacecraft (*2001: A Space Odyssey*, 1968)	A.22	Phys	Mech	L	Bilateral	Used	H	Technological predictions	Propulsion (traditional), cockpit	Quadrilateral
Millennium Falcon (*Star Wars*, 1977)	A.28	Phys	Mech	M	Asymmetric	Used	H	Hamburger and olive	Propulsion (traditional), cockpit	Roughly quadrilateral
TIE Fighter (*Star Wars*, 1977)	A.30	Phys	Mech	S	Bilateral	Used	H	UNK	Wings, cockpit	Complex

Continued on next page...

Table B.2 – Continued

Name	Figure	Model	Type	Scale	Symmetry	Tech	Race	Inspiration	Funct. Comp.	Panel Shape
X-Wing Fighter (*Star Wars*, 1977)	A.32	Phys	Mech	S	Bilateral	Used	H	Aircraft	Wings, weapons, propulsion (traditional), cockpit	Roughly quadrilateral
Death Star (*Star Wars*, 1977)	A.34	Phys	Mech	L	Spherical	Used	H	UNK	None	Quadrilateral
Empirial Star Destroyer (*Star Wars*, 1977)	A.38	Phys	Mech	L	Bilateral	Used	H	Naval destroyer	Propulsion (traditional)	None
Rebel Blockade Runner (*Star Wars*, 1977)	A.40	Phys	Mech	M	Bilateral	Used	H	Hammer	Propulsion (traditional), cockpit, weapons	Roughly quadrilateral
Mon Calamari Star Cruisers (*Star Wars*, 1977)	A.41	Phys	Orgn	L	Bilateral	Used	A	Pickle	Wings, propulsion (traditional)	None
Y-Wing fighter (*Star Wars*, 1977)	A.42	Phys	Mech	M	Bilateral	Used	H	Stripped hot rod	Cockpit, wings, propulsion (traditional)	Quadrilateral
Rebel Transport (*Star Wars*, 1977)	A.43	Phys	Mech	L	Bilateral	Used future	H	Worm	Propulsion (traditional)	Complex
Slave I (*Star Wars*, 1977)	A.44	Phys	Mech	S	Bilateral	Used	H	Radar dish	Cockpit, propulsion (exotic)	Complex
Rebel Medical Frigate (*Star Wars*, 1977)	A.45	Phys	Mech	L	Bilateral	Used	H	Outboard motor	Propulsion (traditional)	Quadrilateral, complex
Imperial Shuttle (*Star Wars*, 1977)	A.46	Phys	Mech	M	Bilateral	New	H	Origami swan	Wings, cockpit, propulsion (traditional)	Quadrilateral, complex
B-Wing fighter (*Star Wars*, 1977)	A.47	Phys	Mech	S	Bilateral	Used	H	UNK	Wings, cockpit, propulsion (traditional), weapons	Complex

Continued on next page. . .

Table B.2 – Continued

Name	Figure	Model	Type	Scale	Symmetry	Tech	Race	Inspiration	Funct. Comp.	Panel Shape
Trade Federation Battleship (*Star Wars* prequels, 1999)	A.48	Comp	Mech	L	Radial	New	H	Updated saucer	None	Roughly quadrilateral
Naboo Royal Starship (*Star Wars* prequels, 1999)	A.49	Comp	Mech	M	Bilateral	New	H	Hood ornament	Cockpit, wings	None
Narn G'Quan Class Heavy Cruiser (*Babylon 5*, 1994)	A.50(a)	Comp	Mech	L	Bilateral	New	A	Tribal	Wings, propulsion (traditional)	None
Narn Cruiser (*Babylon 5*, 1994)	A.50(b)	Comp	Mech	UNK	Bilateral	New	A	Tribal	Wings, propulsion (traditional)	None
Earth Alliance Babylon 5 station (*Babylon 5*, 1994)	A.51(a)	Comp	Mech	L	Bilateral	New	H	UNK	Propulsion	Quadrilateral
Earth Alliance Asimov class luxury liner (*Babylon 5*, 1994)	A.51(b)	Comp	Mech	L	Bilateral	New	H	*2001: A Space Odyssey*'s Aries IB	Propulsion (traditional), cockpit	Quadrilateral
Earth Force Starfury (*Babylon 5*, 1994)	A.52(a)	Comp	Mech	S	Bilateral	New	H	Aircraft, *The Last Starfighter*'s Gunstar	Wings, cockpit, propulsion (traditional)	Quadrilateral
Earth Force Omega Class Destroyer (*Babylon 5*, 1994)	A.52(b)	Comp	Mech	L	Bilateral	New	H	Steam engine, *2010*'s Leonov	Wings, cockpit, propulsion (traditional)	None
Minbari Sharlin Class Warcruiser (*Babylon 5*, 1994)	A.53(a)	Comp	Orgn	L	Bilateral	New	A	Fish	None	None
Minbari Tinashi Destroyer (*Babylon 5*, 1994)	A.53(b)	Comp	Orgn	L	Bilateral	New	A	Fish	None	None
Centauri Primus Heavy Cruiser (*Babylon 5*, 1994)	A.54(a)	Comp	Mech	L	Bilateral	New	A	UNK	Wings, weapons, propulsion (exotic)	None
Centauri Vorchan Destroyer (*Babylon 5*, 1994)	A.54(b)	Comp	Mech	M	Bilateral	New	A	Klingon Bird-of-Prey	Wings, cockpit, propulsion (exotic)	None

Continued on next page. . .

Table B.2 – Continued

Name	Figure	Model	Type	Scale	Symmetry	Tech	Race	Inspiration	Funct. Comp.	Panel Shape
Shadow Battlecrab (*Babylon 5*, 1994)	A.55(a)	Comp	Orgn	S	Bilateral	New	A	Insect, spider	None	None
Shadow Fighter (*Babylon 5*, 1994)	A.55(b)	Comp	Orgn	S	Bilateral	New	A	Insect	None	None
Vorlon Transport (*Babylon 5*, 1994)	A.56(a)	Comp	Orgn	S	Bilateral	New	A	Bulb of garlic	None	None
Vorlon Fighter (*Babylon 5*, 1994)	A.56(b)	Comp	Orgn	S	Bilateral	New	A	Flower, squid	Propulsion (exotic)	None
Drakh Marauder Heavy Fighter (*Babylon 5*, 1994)	A.57	Comp	Orgn	UNK	Bilateral	New	A	Insect stinger	Cockpit, weapons, propulsion (exotic)	None
Drazi Sunhawk (*Babylon 5*, 1994)	A.58	Comp	Orgn	UNK	Bilateral	New	A	*Blake's 7* Liberator	Weapons, propulsion (exotic)	None
White Star (*Babylon 5*, 1994)	A.59	Comp	Orgn	L	Bilateral	New	A	Pelican	Wings	None
Brakiri Dikati Transport (*Babylon 5*, 1994)	A.60	Comp	Orgn	UNK	Radial	New	A	A sphere	None	None
Battlestar Galactica (*Battlestar Galactica*, 1978)	A.61	Phys	Mech	L	Bilateral	Used	H	Naval	Propulsion (traditional)	Complex
Battlestar Galactica (*Battlestar Galactica*, 2004)	A.62	Comp	Mech	L	Bilateral	Used	H	Naval	Propulsion (traditional)	Roughly quadrilateral
Colonial Viper (*Battlestar Galactica*, 1978)	A.63	Phys	Mech	S	Bilateral	Used	H	Aircraft	Wings, propulsion (traditional), cockpit	Quadrilateral
Colonial Viper (*Battlestar Galactica*, 2004)	A.64	Comp	Mech	S	Bilateral	Used	H	Aircraft, remake or original	Wings, cockpit, propulsion (traditional)	Quadrilateral, complex
Cylon Raider (*Battlestar Galactica*, 1978)	A.65	Phys	Mech	S	Bilateral	Used	A	Updated saucer (unconfirmed)	Wings, weapons, propulsion (traditional)	Complex

Continued on next page. . .

Table B.2 – Continued

Name	Figure	Model	Type	Scale	Symmetry	Tech	Race	Inspiration	Funct. Comp.	Panel Shape
Cylon Raider (*Battlestar Galactica*, 2004)	A.66	Comp	Mech	S	Bilateral	New	A	Aggressive, remake of original	Wings, cockpit, weapons	Complex, curved
Civilian ship (*Battlestar Galactica*, 2004)	A.67	Comp	Mech	L	Asymmetric	New	H	UNK	Propulsion (exotic)	Complex
Civilian ship (*Battlestar Galactica*, 2004)	A.68	Comp	Mech	L	Bilateral	New	H	*2001: A Space Odyssey*'s Space Station V (unconfirmed)	None	Complex
Amarr Battlecruiser (*EVE Online*, 2003)	A.69	Comp	Mech	L	Bilateral	New	H	Orgn, gold, status	Wings, propulsion (traditional)	Complex, curved
Gallente Battlecruiser (*EVE Online*, 2003)	A.70	Comp	Mech	L	Bilateral	Used	H	Orgn, individuality	Propulsion (traditional)	Complex, curved
Minmatar Battlecruiser (*EVE Online*, 2003)	A.71	Comp	Mech	L	Bilateral	Used	H	Crude, junky	Propulsion (traditional)	Quadrilateral
Caldari Battlecruiser (*EVE Online*, 2003)	A.72	Comp	Mech	L	Bilateral	Used	H	Aircraft, naval, utilitarian	None	Quadrilateral
Amarr Frigate (*EVE Online*, 2003)	A.73	Comp	Mech	UNK	Bilateral	New	H	Orgn, gold, status	Propulsion (exotic)	Complex, curved
Gallente Frigate (*EVE Online*, 2003)	A.74	Comp	Mech	UNK	Bilateral	New	H	Orgn, individuality	Propulsion (exotic)	Complex, curved
Minmatar Frigate (*EVE Online*, 2003)	A.75	Comp	Mech	UNK	Asymmetric	Used	H	Crude, junky	None	Roughly quadrilateral
Caldari Frigate (*EVE Online*, 2003)	A.76	Comp	Mech	M	Bilateral	Used	H	Aircraft, naval, utilitarian	Wings, cockpit	Roughly quadrilateral
Amarr Titan (*EVE Online*, 2003)	A.77	Comp	Mech	L	Bilateral	New	H	Orgn, gold, status	None	Complex, curved
Gallente Titan (*EVE Online*, 2003)	A.78	Comp	Mech	L	Bilateral	New	H	Orgn, individuality	None	Complex, curved

Continued on next page. . .

Table B.2 – Continued

Name	Figure	Model	Type	Scale	Symmetry	Tech	Race	Inspiration	Funct. Comp.	Panel Shape
Minmatar Titan (*EVE Online*, 2003)	A.79	Comp	Mech	L	Bilateral	Used	H	Crude, junky	None	Complex
Caldari Titan (*EVE Online*, 2003)	A.80	Comp	Mech	L	Bilateral	Used	H	Aircraft, naval, utilitarian	None	Complex
Hiigaran Mothership (*Homeworld 2*, 2003)	A.81	Comp	Mech	L	Bilateral	New	H	UNK	Propulsion (exotic)	Complex
Hiigaran Carrier (*Homeworld 2*, 2003)	A.82	Comp	Mech	L	Asymmetric	New	H	UNK	Propulsion (traditional)	Roughly quadrilateral
Vaygr Minelayer Corvette (*Homeworld 2*, 2003)	A.83	Comp	Mech	UNK	Bilateral	New	A	UNK	Weapons, propulsion (exotic)	None
Vaygr Bomber (*Homeworld 2*, 2003)	A.84	Comp	Mech	UNK	Asymmetric	New	A	UNK	Propulsion (traditional), wings	None
Covenant Destroyer (*Halo*, 2001)	A.87(a)	Comp	Mech	L	Bilateral	Used	A	Rounded, organic frame	Wings, propulsion (exotic), weapons	Complex, curved
Covenant Assault Carrier (*Halo*, 2001)	A.87(b)	Comp	Mech	L	Bilateral	Used	A	Rounded, organic frame	Propulsion (exotic)	Complex, curved
UNSC Phoenix Class (*Halo*, 2001)	A.88(a)	Comp	Mech	L	Bilateral	Used	H	Naval, battleship	Weapons, propulsion (traditional)	Quadrilateral, complex
UNSC Longsword Interceptor (*Halo*, 2001)	A.88(b)	Comp	Mech	M	Bilateral	Used	H	Manta ray	Wings, cockpit, propulsion (traditional)	Complex
Vree Saucer (*Babylon 5*, 1994)	A.89(b)	Comp	Mech	UNK	Radial	New	A	Saucer	None	Quadrilateral, triangle
Saucer (*Independence Day*, 1996)	A.89(a)	Phys	Mech	L	Radial	Used	A	Saucer	None	Complex
Saucer (*Close Encounters of the Third Kind*, 1977)	A.89(c)	Phys	Mech	L	Radial	Used	A	Saucer	None	Roughly quadrilateral

Continued on next page. . .

Table B.2 – Continued

Name	Figure	Model	Type	Scale	Symmetry	Tech	Race	Inspiration	Funct. Comp.	Panel Shape
Visitor Saucer (*V*, 1983)	A.89(d)	Phys	Mech	L	Radial	Used	A	Saucer	None	Quadrilateral
Nostromo (*Alien*, 1979)	A.90	Phys	Mech	L	Asymmetric	Used	H	Heavy machinery	None	Roughly quadrilateral
Red Dwarf (*Red Dwarf*, 1988)	A.91(a)	Phys	Mech	M	Bilateral	Used	H	UNK	None	Roughly quadrilateral
Starbug (*Red Dwarf*, 1988)	A.91(d)	Phys	Mech	S	Bilateral	Used	H	Insect, ant	Cockpit, propulsion	Roughly quadrilateral
Event Horizon (*Event Horizon*, 1997)	A.92	Phys	Mech	L	Bilateral	Used	H	Gothic design	Wings, propulsion (traditional)	Complex
Gunstar (*The Last Starfighter*, 1984)	A.93	Comp	Mech	S	Bilateral	New	H	Aircraft	Wings, weapons, cockpit	Quadrilateral
Lexx (*Lexx*, 1997)	A.94	Comp	Orgn	L	Bilateral	Used	A	Insect, dragonfly	None	None
Serenity (*Firefly*, 2002)	A.95	Comp	Mech	M	Bilateral	Used	H	Insect, firefly	Cockpit, propulsion (traditional), wings	Quadrilateral
Planet Express (*Futurama*, 1999)	A.96	Comp	Mech	M	Bilateral	New	H	Simple, retro	Cockpit, propulsion (traditional)	None

Appendix C

Parameters

The following is a list of parameters that can be modified by the user to adjust the output of our prototype program. We describe each parameter's possible values, its effect on the output, and its default value. Parameters are grouped into categories based on effect.

C.1 Geometry parameters

The following parameters determine the probability that a panel will be assigned a given base primitive, as well as the range of valid height values for this base primitive. These parameters are described in Section 5.3.3.

MIN_HEIGHT A floating point value that determines the minimum height that can be assigned to a base primitive. This parameter is used to generate a random height for new base primitives. The default value is 10.0.

MAX_HEIGHT A floating point value that determines the maximum height that can be assigned to a base primitive. This parameter is used to generate a random height for new base primitives. The default value is 30.0.

PROB_CUBE This floating point value represents the probability that a panel's base primitive will be assigned a cube. This is computed initially for each panel. Its value ranges from 0 to 1, inclusive. Its default value is 0.4.

PROB_COMPLEMENTARY This floating point value represents the probability that a panel's base primitive will be assigned a wedge. This is computed for all panels that are adjacent to a panel that has been assigned a cube. Its value ranges from 0 to 1, inclusive. Its default value is 0.5.

PROB_JOINT This floating point value represents the probability that two neighbouring panels with model base primitives will be joined with pipes. Its value ranges from 0 to 1, inclusive. Its default value is 0.3.

PROB_RECT_BASE An integer that represents the probability that a panel will be left empty and not assigned a base primitive. This is considered only after cubes and complementary geometry have been assigned. Its value can be any positive integer. The probability is calculated by dividing this value by the total sum of the probabilities of all base primitives. Its default value is 100.

PROB_CONE_BASE An integer that represents the probability that a panel's base primitive will be assigned a cone. This type of base primitive is considered only after cubes and complementary geometry have been assigned. Its value can be any positive integer. The probability is calculated by dividing this value by the total sum of the probabilities of all base primitives. Its default value is 100.

PROB_CYL_BASE An integer that represents the probability that a panel's base primitive will be assigned a cylinder. This type of base primitive is considered only after cubes and complementary geometry have been assigned. Its value can be any positive integer. The probability is calculated by dividing this value by the total sum of the probabilities of all base primitives. Its default value is 200.

PROB_PYR_BASE An integer that represents the probability that a panel's base primitive will be assigned a pyramid. This type of base primitive is considered only after cubes and complementary geometry have been assigned. Its value can be any positive integer. The probability is calculated by dividing this value by the total sum of the probabilities of all base primitives. Its default value is 100.

PROB_MODEL_BASE An integer that represents the probability that a panel's base primitive will be assigned a kitbashed model. This type of base primitive is considered only after cubes and complementary geometry have been assigned. Its value can be any positive integer. The probability is calculated by dividing this value by the total sum of the probabilities of all base primitives. Its default value is 800.

C.2 Panel parameters

The following parameters determine the size of the grid panels, the number of splits performed to create subpanels, and whether panels are split symmetrically. These parameters are described in Section 5.3.1.

GRID_SIZE_X An integer representing the size of the grid in x for the frame node's grid panels. The value of this parameter must be at least one. Its default value is 10.

GRID_SIZE_Y An integer representing the size of the grid in y for the frame node's grid panels. The value of this parameter must be at least one. Its default value is 10.

SPLITS_PER_PANEL An integer representing the number of split points that will be generated for a single panel's split. The value of this parameter must be at least one. Its default value is 8.

NUM_RECURSIONS This integer is the number of recursive splits that will be performed to generate panels. It represents the depth of the panel tree. Its default value is 3.

SYMM_PROBABILITY This floating point value represents the initial probability that two panels in symmetric frame nodes will be split the same. For each further recursive split, this probability is divided in half. Its value ranges from 0 to 1, inclusive. Its default value is 0.5.

ENFORCE_SYMMETRY A boolean value that determines whether symmetric frame nodes should be processed together. This parameter determines whether pattern panels are symmetric across different sections of the frame. Its default value is true.

C.3 Pattern Parameters

The following parameters determine the size and regularity of the pattern panels. These parameters are described in Section 5.3.2.

MAX_PATTERN_BLOCKS An integer that gives the maximum number of grid panels that can be used for the pattern panels. The actual number of pattern panels used for a given frame node will be a random number between 1 and MAX_PATTERN_BLOCKS. The default value is 4.

PATTERN_REGULARITY This floating point value is a measure of the regularity of the pattern panels. For each set of pattern panels, this value is queried to determine the degree to which this instance deviates from the regular grid. Its value ranges from 0 to 1, inclusive, where 1 results in complete regularity and 0 results in complete randomness. The default value is 0.3.

C.4 Pipe parameters

The following parameters determine the behavior and appearance of the pipes. These parameters are described in Section 5.3.4.

PERPENDICULAR_FACTOR This integer is the value of the weight penalty assigned to nodes in the pipe graph that are perpendicular to the current path. It discourages pipes from creating unnecessary bends. Its default value is 25.

COMPLEMENT_FACTOR This integer is the value of the weight penalty assigned to nodes in the pipe graph that are between panels with complementary geometry. It prevents pipes from selecting a path between complementary geometry unless no other path exists between the source and destination. Its default value is INT_MAX.

EXISTPATH_FACTOR This integer is the value of the weight bonus assigned to nodes in the pipe graph that have already been assigned a pipe. It provides incentive for pipes to share the same path. Its default value is -10.

PATTERN_FACTOR This integer is the value of the weight penalty assigned to nodes in the pipe graph that are between special panels, such as pattern and edge panels. It prevents pipes from selecting a path between these special panels unless no other path exists between the source and destination. Its default value is INT_MAX.

PIPE_SIZE This floating point value represents the width of the pipes. It is used to both create the pipe geometry and to adjust base primitives to create room for the pipe geometry. Its default value is 8.0.

C.5 GUI parameters

The following parameters are used to determine which geometry is displayed in the GUI. A screenshot of the GUI is shown in Figure C.1.

DRAW_PANELS A boolean value that determines whether panel geometry is drawn in the GUI. Its default value is true.

DRAW_PIPES A boolean value that determines whether pipe geometry is drawn in the GUI. Its default value is true.

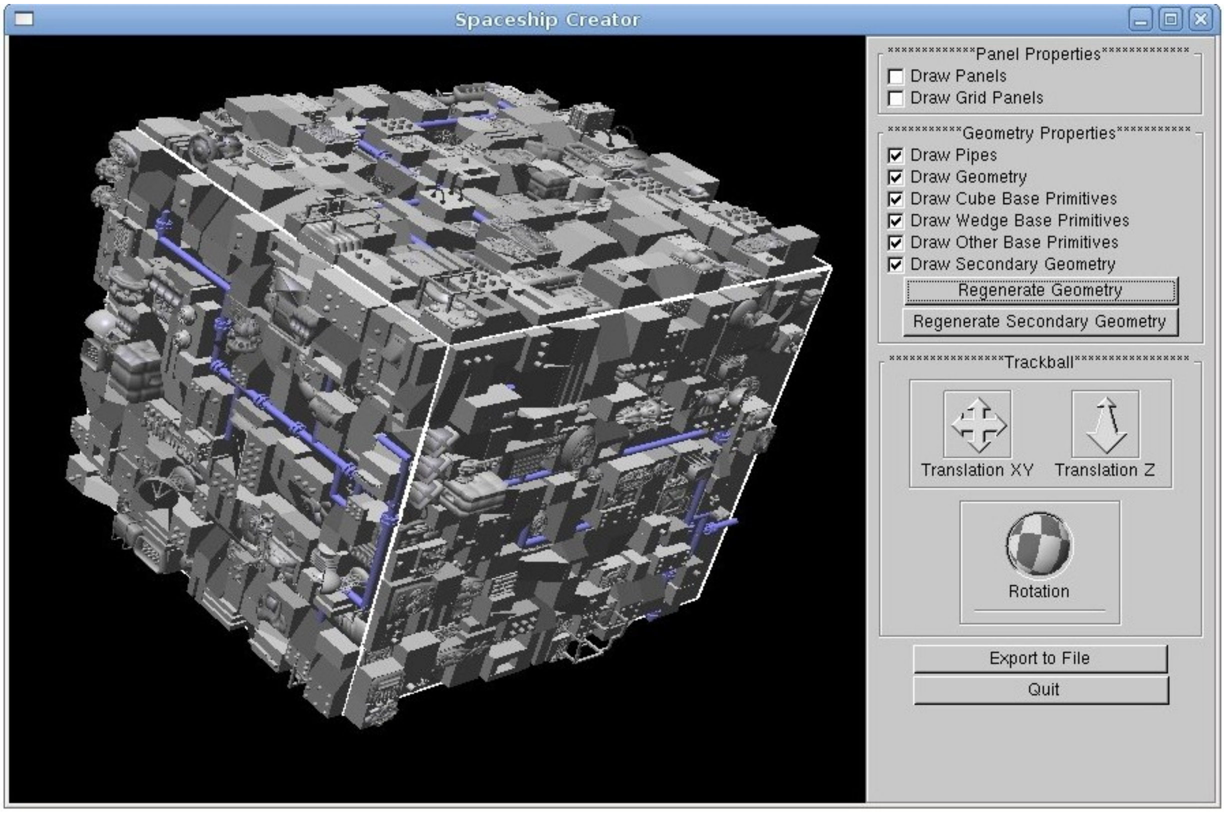
DRAW_GEOMETRY A boolean value that determines whether nurnie geometry is drawn in the GUI. The value of this parameter affects DRAW_SECONDARY, DRAW_CUBES, DRAW_COMPLEMENTARY, and DRAW_OTHER. Its default value is true.

DRAW_CUBES A boolean value that determines whether cube base primitives are drawn in the GUI. This parameter only takes effect if DRAW_GEOMETRY is also true. Its default value is true.

DRAW_COMPLEMENTARY A boolean value that determines whether complementary base primitives are drawn in the GUI. This parameter only takes effect if DRAW_GEOMETRY is also true. Its default value is true.

DRAW_REST A boolean value that determines whether other base primitives (aside from cubes and complimentary primitives) are drawn in the GUI. These primitives include both kitbashed models and pattern tile models. This parameter only takes effect if DRAW_GEOMETRY is also true. Its default value is true.

DRAW_SECONDARY A boolean value that determines whether secondary geometry is drawn in the GUI. This parameter only takes effect if DRAW_GEOMETRY is also true. Its default value is true.



*************Panel Properties*************
☐ Draw Panels
☐ Draw Grid Panels

***********Geometry Properties***********
☑ Draw Pipes
☑ Draw Geometry
☑ Draw Cube Base Primitives
☑ Draw Wedge Base Primitives
☑ Draw Other Base Primitives
☑ Draw Secondary Geometry

Figure C.1: A screenshot of the graphical user interface.

References

[1] John Harris official website. `http://www.alisoneldred.com/artistJohnHarris.html`. 87, 89

[2] Realsoft 3D: Real GreebleZ! generator. `http://catmtn.com/realgreeblez.php`. 14, 15

[3] Sci-fi Meshes. `http://www.scifi-meshes.com/`. xi, 29

[4] Star Wars Official Website Starship Databank. `http://www.starwars.com/databank/`. 84, 104, 105, 106, 110

[5] The Internet Movie Database. `http://www.imdb.com`. 119

[6] TrueSpace Greeble maker. `http://www.pinacoderm.com/tsx/greeble/index.html`. 14, 15

[7] The Beast Within: The Making of Alien. DVD documentary, 2003. 130

[8] 3DXtras Free 3D Models Library Online. `http://www.3dxtras.com`, 2008. 67

[9] John Berkey Remembered. `http://www.starwars.com/community/news/family/news20080513.html`, May 2008. 88

[10] Star Trek official site. `http://www.startrek.com`, 2008. 91, 96, 97, 98, 99, 101

[11] Scifi-Meshes: The Largest 3D-Scifi Based Online Community. `http://www.scifi-meshes.com/`, 2009. 67

[12] TurboSquid 3D Marketplace. `http://www.turbosquid.com`, 2009. 67

[13] Attene, M., Falcidieno, B., and Spagnuolo, M. Hierarchical mesh segmentation based on fitting primitives. *The Visual Computer: International Journal of Computer Graphics 22*, 3 (2006), 181–193. 81

[14] Bartz, D., Cunningham, D., Fischer, J., and Wallraven, C. The role of perception for computer graphics. In *Proceedings of the 29th Annual Conference Eurographics* (2008), pp. 65–86. 16

[15] Berlyne, D. *Aesthetics and Psychobiology.* Meredith Corporation, New York, 1971. 20, 21, 22, 23, 24, 25, 26

[16] Berlyne, D., and Boudewijns, W. Hedonic effects of uniformity in variety. *Canadian Journal of Psychology 25*, 3 (1971), 195–206. 24, 25

[17] Birkhoff, G. D. *Aesthetic Measure.* Harvard University Press, University of Michigan, 1933. 21

[18] Boyce, T. Interview with Todd Boyce, 2008. xi, 7

[19] CCP Games. Eve Online. `http://www.eve-online.com`. 122

[20] Chen, G., Esch, G., Wonka, P., Müller, P., and Zhang, E. Interactive procedural street modeling. *ACM Transactions on Graphics 27*, 3 (2008). 12

[21] Chiang, D. *Mechanika: Creating the Art of Science Fiction with Doug Chiang.* IMPACT books, New York, NY, 2008. 38, 44

[22] Choy, L., Ingram, R., Quigley, O., Sharp, B., and Willmott, A. Rigblocks: Player-deformable objects. In *ACM SIGGRAPH 2007 sketches* (San Diego, California, 2007). 14

[23] Cole, F., Golovinskiy, A., Limpaecher, A., Barros, H. S., Finkelstein, A., Funkhouser, T., and Rusinkiewicz, S. Where do people draw lines? *ACM Transactions on Graphics (Proc. SIGGRAPH) 27*, 3 (Aug. 2008). 16

[24] Cole, F., Sanik, K., DeCarlo, D., Finkelstein, A., Funkhouser, T., Rusinkiewicz, S., and Singh, M. How well do line drawings depict shape? In *ACM Transactions on Graphics (Proc. SIGGRAPH)* (Aug. 2009), vol. 28. 16

[25] Colyar, K. S., and Matthews, G. B. Procedural modeling of medieval castles. In *ACM SiGGRAPH 2005 Posters* (2005). 14

[26] DeCarlo, D., Finkelstein, A., Rusinkiewicz, S., and Santella, A. Suggestive contours for conveying shape. *ACM Transactions on Graphics (Proc. SIGGRAPH) 22*, 3 (July 2003), 848–855. 16

[27] DeCarlo, D., and Rusinkiewicz, S. Highlight lines for conveying shape. In *International Symposium on Non-Photorealistic Animation and Rendering (NPAR)* (August 2007). 16

[28] Dennett, D. C. *The Intentional Stance.* The MIT Press, Cambridge, MA, 1987. 42

[29] Deussen, O. *Digital Design of Nature: Computer Generated Plants and Organics.* Springer, 2005. 11

[30] DiFate, V. *The Science Fiction Art of Vincent DiFate.* Paper Tiger, London, 2001. 87, 89, 90

[31] Discoe, B. Virtual Terrain Project. `http://www.vterrain.org`. 11

[32] Drexler, D. Designing the Ti'Mur. *Star Trek: The Magazine 3*, 3 (July 2002). 97

[33] Duke, D., Barnard, P., Halper, N., and Mellin, M. Rendering and affect. *Computer Graphics Forum (Eurographics 2003) 22* (2003), 359–368. 40

[34] Ebert, D. S., Musgrave, F. K., Peachey, D., Perlin, K., and Worley, S. *Texturing & Modeling: A Procedural Approach*, third ed. Morgan Kaufmann, 2003. 10, 13

[35] Fleming, R. W., Torralba, A., and Adelson, E. H. Specular reflections and the perception of shape. *Journal of Vision 4*, 9 (9 2004), 798–820. 16

[36] Frank, J. *The Art of John Berkey.* Paper Tiger, London, 2003. 32, 87, 88

[37] Gage, J. *Color and culture: practice and meaning from antiquity to abstraction.* University of California Press, 1999. 32

[38] Gal, R., Sorkine, O., Popa, T., Sheffer, A., and Cohen-Or, D. 3d collage: Expressive non-realistic modeling. In *Proceedings of 5th international symposium on non-photorealistic animation and rendering* (2007). 79

[39] Geodakyan, V. A. Evolutionary role of asymmetrization of organisms, brain and body (model and right-hand rule). In *20th Congress of the I.P. Pavlov Physiological society* (Moscow, 2007), p. 28. 35

[40] Glass, K. R., Morkel, C., and Bangay, S. D. Duplicating road patterns in south african informal settlements using procedural techniques. In *Proceedings of the 4th International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa* (2006), pp. 161–169. 12

[41] Gombrich, E. *The Sense of Order: A Study in the Psychology of Decorative Art*, third ed. John Wiley & Sons, New York, 1975. 17, 20, 22, 24, 25, 26, 30, 32, 42, 43

[42] Gurney, J. *Imaginative Realism: How to Paint What Doesn't Exist.* Andrews McMeel Publishing, Kansas City, Missouri, 2009. 2

[43] Hagery, J., and Rogers, J. C. *Spaceship Handbook.* ARA Press, Livermore, CA, 2001. 85, 102

[44] Hammes, J. Modeling of ecosystems as a data source for real-time terrain rendering. In *Proceedings of the First International Symposium on Digital Earth Moving* (2001), pp. 98–111. 11

[45] Hart, G. Growth forms. In *Proceedings of the 2009 Bridges Conference on Mathematical Connections in Art, Music, and Science* (2009). 14

[46] Herbruck, A. Federation starship datalink: Star trek concept art, 2009. 99

[47] Home, L. K. H. *Elements of Criticism*, seventh ed. 1788. 26

[48] Hudson, T. 3DStudioMax Greeble Plug-in. `http://max.klanky.com/`. 14, 15

[49] Hutcheson, F. *An Inquiry into the Original of Our Ideas of Beauty and Virtue.* Liberty Fund Inc., 1726. 17, 20, 25, 26, 32, 42

[50] Igarashi, T., and Hughes, J. F. Smooth meshes for sketch-based freeform modeling. In *Proceedings of the 2003 symposium on Interactive 3D graphics* (2003), pp. 139–142. 80

[51] Jeffries, M. Designing the Starship Enterprise. *Star Trek: The Magazine 1*, 10 (February 2000). 94

[52] Kelly, G., and McCabe, H. Citygen: An interactive system for procedural city generation. In *The Fifth Annual International Conference in Computer Game Design and Technology* (2007), pp. 8–16. 12

[53] Koffka, K. *Principles of Gestalt Psychology*. Routledge, London, 1999. 22

[54] Labedade, A. Lightwave greebles and nurnies script. `http://earthwormjim.free.fr/lscript/modeler/greeble/index.html`, 2008. 14, 15

[55] Lai, Y.-K., Zhou, Q.-Y., Hu, S.-M., and Martin, R. R. Feature sensitive mesh segmentation. In *SPM '06: Proceedings of the 2006 ACM symposium on Solid and physical modeling* (New York, NY, USA, 2006), pp. 17–25. 79

[56] Larsson, M., Zalzala, J., and Duda, G. A procedural ocean toolkit. In *ACM SIGGRAPH 2006 Sketches* (2006). 14

[57] Lasseter, J. Principles of traditional animation applied to 3d computer animation. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1987), pp. 35–44. 44

[58] Lechner, T., Ren, P., Watson, B., Brozefski, C., and Wilenski, U. Procedural modeling of urban land use. In *International Conference on Computer Graphics and Interactive Techniques* (2006). 12

[59] Lee, R. A. *IPT's Pipe Trades Handbook*, eleventh ed. IPT Publishing and Training Ltd., Edmonton, Alberta, 2008. 49, 50

[60] Li, M., and Vitányi, P. *An introduction to Kolmogorov complexity and its applications*, second ed. Springer, 1997. 19

[61] Lipp, M., Wonka, P., and Wimmer, M. Interactive visual editing of grammars for procedural architecture. *ACM Transacations on Graphics 21*, 3 (2008). 12

[62] Lluch, J., Camahort, E., and Vivó, R. Procedural multiresolution for plant and tree rendering. In *Proceedings of the 2nd international conference on Computer graphics, virtual Reality, visualisation and interaction in Africa* (2003), pp. 31–38. 11

[63] Locher, P. J., and Nodine, C. G. Influence of stimulus symmetry on visual scanning patterns. *Perception and Psychophysics 13*, 3 (1973), 408–412. 22

[64] Martinet, A., Galin, E., Desbenoit, B., and Akkouche, S. Procedural modeling of cracks and fractures. In *Proceedings of the Shape Modeling International 2004 (SMI'04)* (2004), pp. 346–349. 14

[65] May, J. Perceptual Principles and Computer Graphics. *Computer Graphics Forum 19*, 4 (2000), 271–279. 16

[66] McFayden, T. B5 Scrolls website. `http://themadgoner.com/B5/B5Scrolls/B5Scrolls.htm`. 84, 113, 115, 116, 117, 118

[67] Moles, A. *Information Theory and Esthetic Perception.* University Of Illinois Press, Urbana and London, 1966. 18, 19, 20, 21, 23, 24, 25

[68] Müller, P., Wonka, P., Haegler, S., Ulmer, A., and Gool, L. V. Procedural modeling of buildings. *ACM Transactions on Graphics 25*, 3 (2006), 614–623. 12

[69] Müller, P., Zeng, G., Wonka, P., and Gool, L. V. Image-based procedural modeling of facades. *ACM Transactions on Graphics 26*, 3 (2007), 1–9. 12

[70] Oppenheimer, P. E. Real time design and animation of fractal plants and trees. *ACM SIGGRAPH Computer Graphics 20*, 4 (1986), 55–64. 11

[71] Parish, Y. I. H., and Müller, P. Procedural modeling of cities. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (2001), pp. 301–308. 12

[72] Peterson, L. *Sculpting a Galaxy: Inside the Star Wars Model Shop.* Insight Editions, San Rafael, CA, 2006. 4, 5, 6, 29, 36, 42, 103, 104, 106, 108, 109, 110, 111, 119, 130

[73] Platt, J. *Perception and Change: Projections for Survival.* The University of Michigan Press, 1970. 25

[74] Provins, K., Stockbridge, H., Forrest, D., and Anderson, D. The representation of aircraft by pictorial signs. *Occupational Psychology 31* (1957), 21–32. 40

[75] Prusinkiewicz, P., James, M., and Měch, R. Synthetic topiary. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques* (1994), pp. 351–358. 11

[76] Prusinkiewicz, P., and Lindenmayer, A. *The algorithmic beauty of plants.* Springer-Verlag, New York, 1990. 10, 11

[77] Quan, L., Tan, P., Zeng, G., Yuan, L., Wang, J., and Kang, S. B. Image-based plant modeling. *Proceedings of ACM SIGGRAPH 2006 25*, 3 (2006), 599–604. 11

[78] Ramachandran, V., and Hirstein, W. The Science of Art: A Neurological Theory of Aesthetic Experience. *Journal of Consciousness Studies 6* (1999), 15–51. 22, 25

[79] Reeves, T. W., and Blau, R. Approximate and probabilistic algorithms for shading and rendering structured particle systems. *Computer Graphics 19* (1985), 313–322. 11

[80] Reeves-Stevens, J., and Reeves-Stevens, G. *The Art of Star Trek.* Pocket Books, Cincinnati, Ohio, 1995. 36, 91, 93, 98

[81] Reynolds, D. W., and Saxon, C. *Star Wars Complete Cross-Sections.* Dorling Kindersley, 2007. 83

[82] Robertson, S. *Lift Off: Air Vehicle Sketches & Renderings from the Drawthrough Collection.* Design Studio Press, Culver City, CA, 2006. 33

[83] Rosky, E. Blender Discombobulator Plug-in. `http://www.nccn.net/~w_rosky/evan/evan/programs/discombobulator/index.html`, 2005. 14, 15

[84] Schneider, B. Ex Astris Scientia. `http://www.ex-astris-scientia.org/articles/design.htm`, 2009. 92, 93, 95

[85] Shannon, C. A mathematical theory of communication. *The Bell System Technical Journal 27*, 3 (1948), 379–423 and 623–656. 18, 19

[86] Side Effects Software. Houdini Apprentice. `http://www.sidefx.com/`, 2009. 67

[87] Smelik, R. M., de Kraker, K. J., and Groenewegen, S. A. A survey of procedural methods for terrain modelling. In *CASA Workshop on 3D Advanced Media in Gaming and Simulation* (2009). 11, 12

[88] Stableford, B. M. *Science fact and science fiction: An Encyclopedia.* Routledge, 2006. 1, 84, 85

[89] Sternbach, R., and Okuda, M. *Star Trek: The Next Generation Technical Manual.* Pocket Books, New York, NY, 1991. 83

[90] Stiny, G., and Gips, J. Shape grammars and the generative specification of painting and sculpture. *In Proceedings of IFIP Congress71* (1972), 1460–1465. 11

[91] Sun, J., Yu, X., Baciu, G., and Green, M. Template-based generation of road networks for virtual city modeling. In *Proceedings of the ACM symposium on Virtual Reality Software and Technology* (2002), pp. 33–40. 12

[92] Teoh, S. T. Generalized descriptions for the procedural modeling of ancient east asian buildings. In *Proceedings of the International Symposium on Computational Aesthetics in Graphics, Visualization, and Imaging* (May 2009). 14

[93] Tyler, G. An exclusive interview with Andrew Probert. `http://www.trekplace.com/ap2005int01.html`, 2005. 93

[94] Weyl, H. *Symmetry.* Princeton Science Library, 1989. 34

[95] Whiting, E., Ochsendorf, J., and Durand, F. Procedural modeling of structurally-sound masonry buildings. *ACM Transactions on Graphics 28*, 5 (2009), 112. 12

[96] Winkenbach, G., and Salesin, D. H. Computer-generated pen-and-ink illustration. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1994), ACM, pp. 91–100. 20

[97] Wonka, P., Wimmer, M., Sillion, F., and Ribarsky, W. Instant architecture. *ACM Transactions on Graphics 22*, 3 (2003), 669–677. 11, 12

[98] Yang, C., Sharon, D., and van de Panne, M. Sketch-based modeling of parameterized objects. In *2nd Eurographics Workshop on Sketch-Based Interfaces and Modeling* (Dublin, 2005). 80

[99] Zhou, K., Huang, X., Wang, X., Tong, Y., Desbrun, M., Guo, B., and Shum, H.-Y. Mesh quilting for geometric texture synthesis. *ACM Transactions on Graphics (SIGGRAPH)* (2006), 690–697. 13