Introduction

In 1435, the Italian scholar Leon Battista Alberti wrote a Latin treatise titled *De Pictura* (on painting). In 1436, it was translated into Italian and distributed. It is the first known publication on the subject of *linear perspective*, a subject very closely related to the very heart of modern computer graphics. The treatise was partly based on observations made by the great Florentine sculptor and architect Filippo Brunelleschi, though other artists from the same period were also experimenting with the technique. The Italian Renaissance in painting was to some considerable extent dependent on this major discovery.

Linear perspective demonstrated that a realistic representation of a 3D environment could be calculated based on rules that govern how our eyes see the world around us. Because these rules could be written down, and because they worked, innumerable artists were able to replicate the results Alberti described and linear perspective became a standard tool for most artists all the way up to the present day (Figs. 0.1 and 0.2).

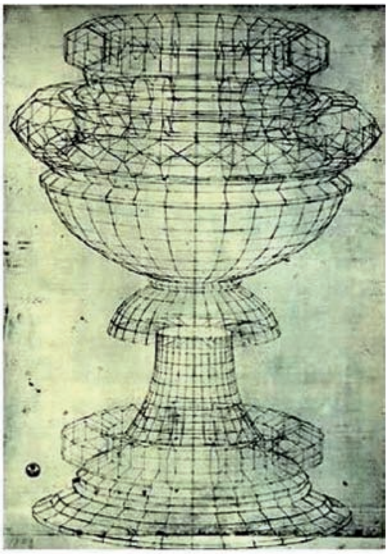



Fig 0.1
Piero Della Francesca, *Brera Altarpiece* 1472–1474
Fig 0.2
Paolo Uccello, *Chalice* 1450

With the invention of the computer, it was inevitable that this knowledge would be turned into software, and it was. When this happened, the modern era of 3D computer graphics, perhaps the most significant, advance in the visual arts since the discovery of perspective was made.

The work of other artists was also used as the basis for innovations in computer graphics. Leonardo Da Vinci's work on aerial perspective was incorporated into 3D rendering software, just as the work of Impressionist and Pointillist artists like Claude Monet and Georges Seurat became the basis for pixel-based representations of visual information (Fig. 0.3).

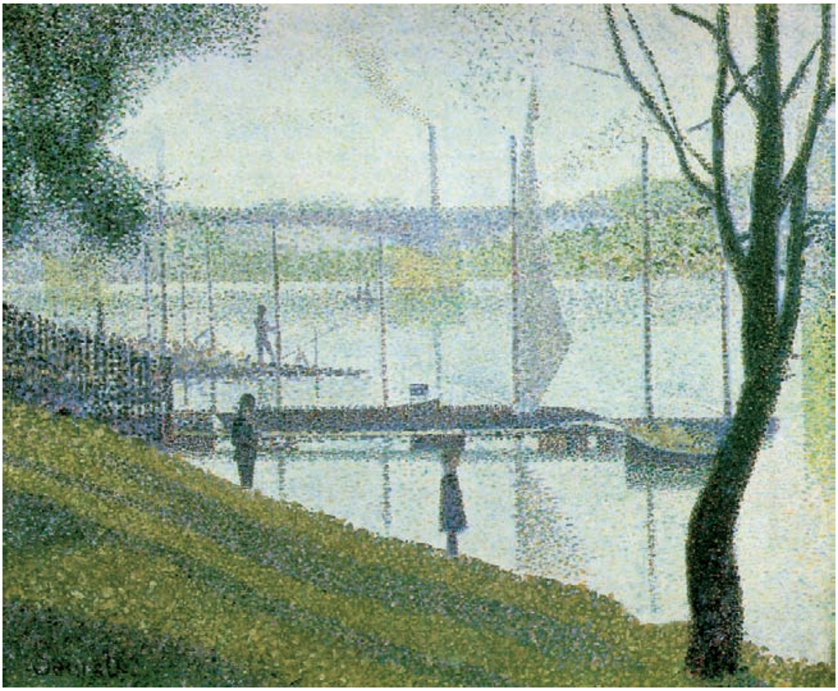


Fig 0.3
Georges Seurat, *Bridge of Courbevoie* 1886/1887

Without the observations of these artists, there would be no computer graphics. Computer graphics are possible only because there are people who looked at the world around them, and went to the trouble not only to describe what they saw, but also to understand what they saw.

Knowledge of a subject is what makes the difference between a novice and an expert. If I look under the hood of a car, I see a bunch of blackened metal objects. A mechanic sees an engine, gaskets, spark plugs, hoses, and many other things he can identify, knows the purpose of, and can assemble himself if necessary. For a computer graphics artist, the difference between a true expert and a novice who just sees a jumble of stuff under the hood is observation (Fig. 0.4).

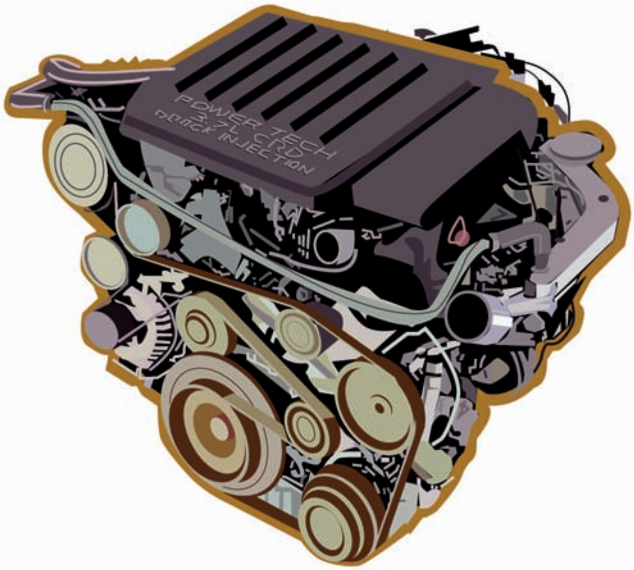


Fig 0.4
Power tech engine block

Linear perspective stems from the *observation* that parallel lines seem to converge as they move farther away from our eye. The reason isn't that they are actually coming together, but that the human eye is nearly spherical in shape to accept visual information from all sides. The result is shapes that appear to converge the closer they are to the pupil (Fig. 0.5).

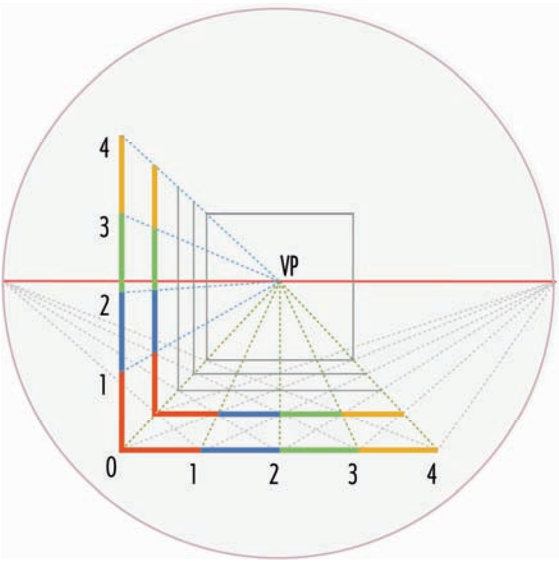


Fig 0.5
Linear perspective

The red line in this illustration is the horizon line, the point labeled “vp” is the vanishing point, the circle is the total field of view (fov), the vanishing points that connect with the fov are *diagonal* vanishing points, and the colored bars are for measurement.

To reproduce this on paper, artists would draw a straight horizontal line to represent the horizon, then a point in the center of the line to represent the *vanishing point* (this corresponds to the center one’s pupil). Another horizontal line is then drawn either above or below the horizon line and subdivided equally, with evenly spaced lines drawn directly into the *vanishing point*. These lines represent parallel lines and may be used to measure height, width, or depth. These lines simulate how parallel lines as seen by the human eye converge in toward the pupil (Fig. 0.6).

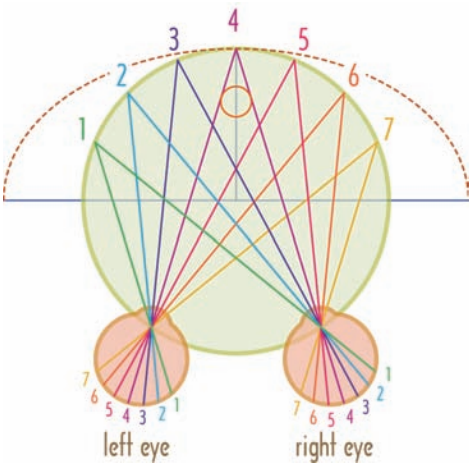


Fig 0.6
Stereovision

As the point of focus shifts to the right or left of center, the angle formed by the ray from each eye to the subject becomes more acute. This causes some distortion and an exaggerated impression of curvature.

The original form of linear perspective did not take into account lens curvature, so it was only an approximation of what a human eye sees. Nevertheless, it worked well enough that for 500 years, it was the basis for almost every great work of art made during that period. Later, engineers worked out engineering perspective and they did take *lens curvature* into account. With it, lens curvature could be represented accurately. It is this form of perspective that is now used for computer renderings (Fig. 0.7).

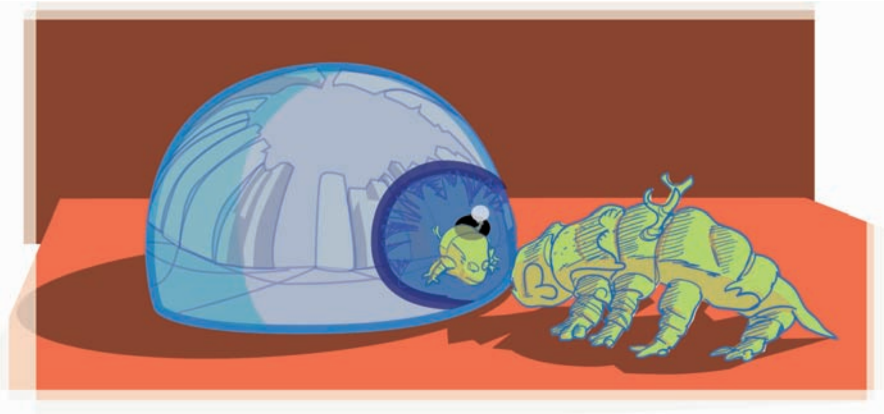


Fig 0.7
Lens curvature

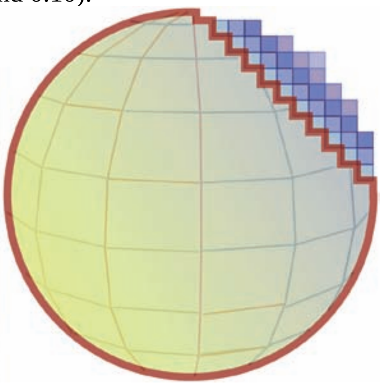
Notice how the curvature of the reflective hemisphere affects the reflection cast upon it. This is the same thing that happens when visual information in three-dimensional space is projected onto your eye.

Leonardo's observation that colors become less distinct over distance became known as *aerial perspective*. He used this to more accurately describe observed details, most famously in the background of the painting *Mona Lisa*. Other artists used the technique as it became better known, and descriptions of it became commonplace in books about art. The phenomenon Leonardo saw, described, and went to some trouble to understand was caused by the fact that our atmosphere contains many tiny light-occluding particles such as dust and fog. At near distances, they do not affect our vision because there are not as many of these particles between our eye and the object we are observing as when we refocus our eyes on a distant object. In computer graphics, *aerial perspective* is known as *environmental fog*. The effect is used effectively to simulate great distances in computer renderings (Fig. 0.8).

Fig 0.8
Shadow in the Summer (© 2005 Stephan Martiniere)

This is an excellent modern example of atmospheric perspective in a painting. Notice how the color in the background is muted because it is flooded with the primary light color. In extreme distances, individual color differences are normally indistinguishable because of this effect.

Without the work of pointillist artists like George Seurat and Henri-Edmund-Cross, or their nearest inspiration, Claude Monet, we might never have seen what we now recognize as 3D computer graphics. What these artists discovered was that if they broke a color into its primary components, it would be seen as the original color. They expanded on this observation by making hundreds of paintings, each of which tested the limits of what came to be known as *pointillism*, a style of painting using nothing but brightly colored dots, or points. These paintings became the basis for pixel-based graphics. Without a method to make a two-dimensional image on a screen, no amount of knowledge regarding perspective would be of any use. With it however, linear perspective could be used to calculate what an image should look like on a 2D plane, and pointillism allows the computer to generate the image (Figs. 0.9 and 0.10).



Figs 0.9 and 0.10
Pointillism and the pixel

These observations by artists led to key discoveries that had (and are still having) far-reaching results. By themselves, the observations would be of little use if the artist were unable to communicate what he had seen or what caused the effect. To be able to articulate, what one has seen is exactly what a computer graphics artist must do. This is the primary skill and it falters if that artist's observation skills are weak.

Today, computer animation software has pointillism, aerial perspective, linear perspective, lens curvature, and many other observations built into the program. Now, the software will perform the mechanical calculations for you, just as a calculator will add numbers. The trick is that you need to be able to input the right numbers. In computer graphics, this means you have to be able to see, understand,

and describe your subject to your software in a language it understands. If you do your job properly, you will receive in return a beautifully rendered image.

Knowledge of a computer animation program will not by itself make anyone into a competent professional animator. They may learn the buttons, they may learn the language of the application, but without well-described *observations*, the raw data needed to create a computer rendering, this knowledge is insufficient to be a truly successful computer artist. To be a successful computer artist, you also need to understand how to look at the world around you, or imagine the one within, with great clarity. You must be able to see detail that others miss, understand why it is there, what it is for, how to distinguish it from its neighbors, and describe it to your application.

The skills just described are the basis for this book. They are application independent and are true of every 3D application currently made. As 3D professionals, you will discover that 3D applications change on a nearly annual basis and that every few years the most popular application will have changed to something new. When this happens, artists who understand computer graphics for what it is, a way to place their real-world observations into an application capable of generating a rendering (even if it is a real-time game engine), will never be out of a job. The reason is that the applications are much easier to learn than these other skills, and their employers know that.

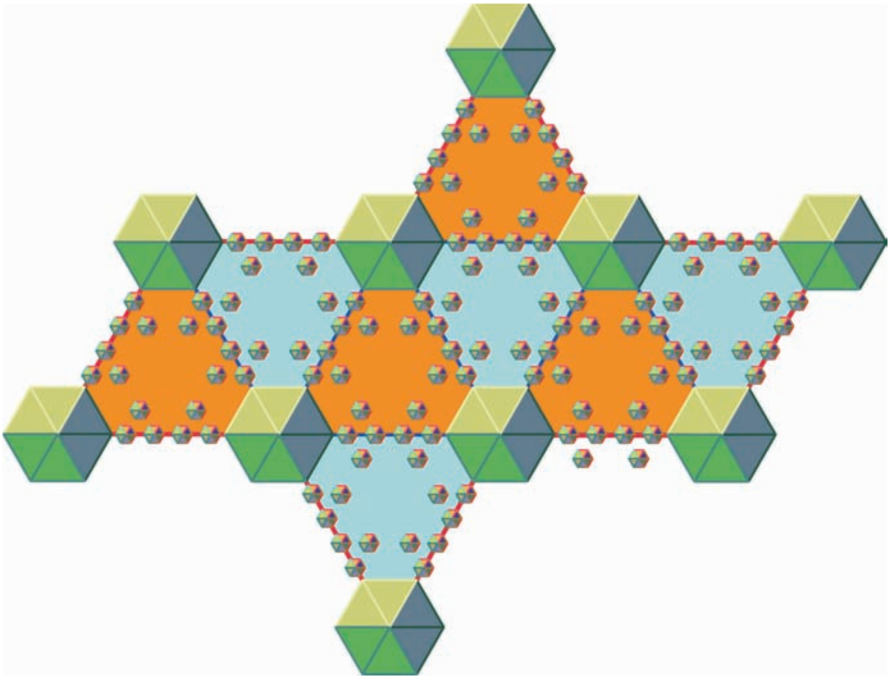
This book is about computer graphics; it is not about computer graphics applications. In this book, you will learn the meaning and usage of computer graphics tools and terminology, but more importantly, the basic observation skills needed to do something great with that knowledge.

The information contained herein is meant to comprise the first portion of a university-level introduction to Computer Graphics course. The layout of the book follows the class structure of my first two modeling classes for freshman students (they are staggered with two animation classes). Although a great deal may be learned by simply reading the text, it is highly recommended that any serious student also performs every exercise offered here.

The first part of the book, 3D1, focuses on fundamental principles of how 3D space is represented in a computer, user interface, basic polygonal modeling tools, and other information essential to getting started as a 3D artist. The second part of the book, 3D2, introduces the more complicated subjects of surfaces, topology, and optimization. Each of these has their own importance, and all help inform the student to make better creative and technical choices with his work.

Andrew Paquette

Part I: 3D1: Introduction to 3D Modeling



Modeling is where all 3D projects must begin, because without a model, there is nothing to animate or render. Studying modeling is also an excellent means of familiarizing an artist with basic concepts of Computer Graphics. By understanding modeling, an artist will also be more comfortable with the software he uses, finding errors in his work, and avoiding problems he might otherwise encounter.

This section is designed to provide a student with a solid foundation in polygon-based modeling and animation applications.