An Analysis of 3D Data Visualization in Numerical Models

Erik Tollerud

2006

Chapter 1

Introduction

We are all visual learners. While some may claim to be so more than others, research has clearly shown that roughly half of the human cerebral cortex is devoted solely to visual processing [7]. The physical sciences, in particular, make extensive use of a variety of visualization techniques as part of their canonical curricula. Yet most of these visualization techniques are two-dimensional, and most physical data, by its nature, must have at least three. In fact, in some sense, a significant part of the training students of the physical sciences receive is how to perform this visualization in their own minds, and how to project it into two dimensions for others to see. Yet the advent of specialized computer graphics hardware now offers ways to visualize numerical data directly in all three spatial dimensions. The aim of this thesis is to take a closer look at these methods, to determine their value alongside the more traditional two-dimensional techniques, and to determine exactly which three-dimensional techniques are particularly helpful.

To conduct this analysis, however, it must be understood how human binocular (stereoscopic) vision works, as well as the methods for mimicking this effect using computer graphics. Moreover, an understanding of the major types of three-dimensional data must be attained before any analysis can be done. In this thesis, then, I will describe these concepts, and present an example of a computer model that generates these data; the relaxation method and the method of Green's functions will be used to solve problems in electrostatics. This will necessitate a discussion of the issues inherent in these models. With this in mind, an analysis of this visualization method and its value will be completed, partly through use of data gathered at the presentation of this thesis.

Note on Computer Software

A significant portion of this thesis was the writing of the software ("Stereo Data Display Modules (SDDM)") used for the research. The project will be released open-source on the web site `http://anierman.ups.edu/stereo`. Furthermore, a snapshot (as of April 15, 2006) of the source code and an installer for windows 2000/XP is included on a DVD-ROM with this thesis.

Chapter 2

3D Vision and Computer Visualization Methods

Before understanding how an essentially two-dimensional computer or projector screen can appear to be in three dimensions, some understanding must be attained of how human three-dimensional vision is achieved. Once this is understood, the methods available to visualizing in 3D using computer graphics will be explored. Note that when I use the term "3D graphics" I generally refer to current computer graphics that do not use binocular cues, while "Stereoscopic" or "Stereo 3D" will refer to graphics that make use of binocular cues.

2.1 Monocular Visual Cues

There are a number of cues that the visual system uses to *guess* at the three-dimensional structure of something that it is looking at, but which are still essentially possible with only two-dimensions (i.e. with just one eye — hence the term monocular cues).

The simplest monocular cue is interposition (sometimes called partial occlusion). Figure 2.1 is an example of this effect. Interposition is the recognition that when objects overlap, the closest is the one that is visible, while those behind are blocked from view. This technique is one of the most important elements of 3D graphics, as it is particularly useful for fairly simple recognizable shapes with many straight lines. The inclusion of transparency, is, to some extent, an extension of this method.

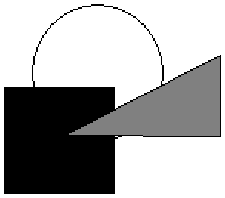


Figure 2.1: The monocular cue of interposition: the shapes appear to be in the order of circle, square, triangle from back to front.

Another important monocular cue used in 3D Graphics is that of shading and lighting. In lighting conditions where there is some directional lighting, certain parts of an object are lit more brightly or in different colors than others. The shading and coloration patterns that result from this provide hints as to how close objects are to the light, as well as in what direction they are angled, because light reflects off many surfaces according to the law of reflection - this is known as specular reflection. An example of the use of lighting (particularly specular) is shown in Figure 2.1, as well as Figure 2.1. For real time 3D graphics, this cue is particularly useful due to the development of the Phong lighting and shading model, a relatively simple equation that produces highly realistic lighting with relatively little computation. Yet it is of somewhat less value in situations with a variety of different shapes with different reflection properties or when there is not a clear sense of where the light is.

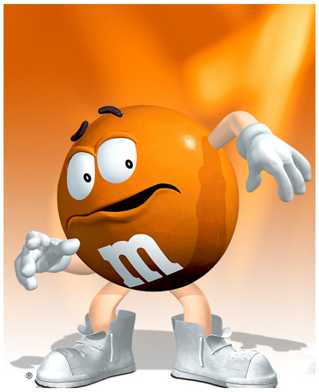


Figure 2.2: Lighting and Shading example. Note the bright white spot showing specular reflection. Courtesy of Microsoft Corp and Mars, Inc.

A third monocular cue is that of perspective. Even classical artists showed some familiarity with perspective (see [4],[8]), and was mathematically first described by the Islamic mathematician Alhazen [9]. Figure 2.1 is an excellent example of the use of perspective to make the cubical shape of an object apparent, and Figure 2.1 also makes significant use of it.

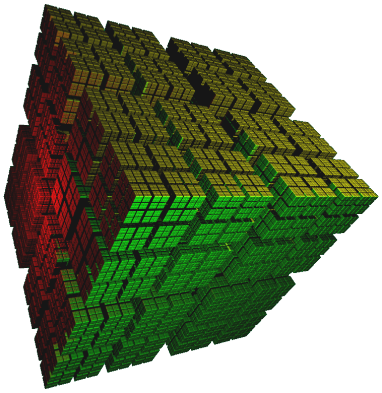


Figure 2.3: A fractal known as a Cantor Cube. Uses colored lighting and perspective to show three-dimensionality. Courtesy [1]

The final monocular cue to be discussed here is that of texture gradients. Figure 2.1 is an example of this in a painting — objects that are intended to be depicted as further have less detail than those near the viewer. While related to perspective, this is not quite the same thing, as texture is not the same as geometry. Still, this is another valuable process for computer graphics, because it involves fairly simple scaling, which is relatively easy to compute.

There are other cues the visual system uses to get depth information about what it sees. The aforementioned four cues, however, are the main cues of of interest because they are the dominant

Figure 2.4: Texture gradients and perspective example: painting "Paris Street: A Rainy Day" by Gustave Caillebotte

techniques used in mono 3D graphics. In modern systems, most of these functions are calculated in the graphics card (hence the term "3D accelerated") - the programmer writes software that specifies where the objects to be drawn are (in three-dimensional space), where the lights are, and how the projection should be computed, and lets the graphics card do the rest. Yet using these cues, the objects still appear to be three dimensional, but still look "stuck" on the screen. This is because they do not make use of the stereoscopic effects that binocular vision provides.

2.2 Binocular Vision

To achieve vision that truly appears three dimensional to the observer, use must be made of both eyes. The human visual system has evolved in such a way as to use this binocular vision as the primary (although less obvious) means of determining the distance of an object. Figure 2.2 shows the fundamentals of this — the parallax effect. If two objects are placed at different distances, but along the visual axis, they will project differently onto the focal plane (the dashed lines in the figure shows two possible such planes). An object that is much closer will appear in two different places if the focal plane is far away from it, while a more distant object will appear in focus at just one point. This is the basis for depth perception through binocular vision. By itself, this does not show for certain where the nearer object is (it could be a different-shaped object), but if the focal plane or the various relative positions change, the shift in all the images is quickly used by the brain to order the distance of the objects.

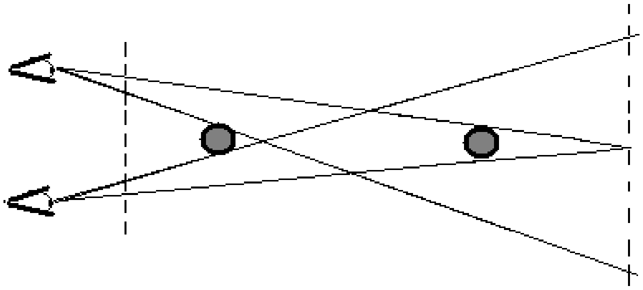


Figure 2.5: Parallax in binocular vision

Note that all this depends on is recognizing that the two eyes are projecting the same focal plane onto slightly different locations. 3D computer graphics systems (like OpenGL) in fact have built-in functions for setting a projection of a plane to a point In mono 3D, this of course is just done once, but there are a variety of methods of generating stereo pairs(see [2], Stereographics section). Hence, the primary technical issue is finding a way to send a different image to each eye of the viewer.

2.3 Stereoscopic Graphics Methods

There are three common methods used to perform this eye selection, given two images appropriately prepared. The three methods are anaglyph, active stereo, and polarized projection.

Anaglyph stereo is the most well known version of stereoscopic graphics. It requires two colors, and special glasses with a filter with each of the two colors over each eye - the most common colors used are red-blue or red-cyan. The two images are colorized with one the color for the right eye, and the other the color for the left eye, and are displayed on the screen in those colors. The filters remove the other color, and ensure that each eye only sees the color intended for it. This produces fairly decent stereoscopic images (although some people report headaches after use). Moreover, given that most computer and TV monitors are already color, all that is required for viewing are the glasses (which are readily and cheaply available). Unfortunately, anaglyph has the disadvantage that it does not permit color. In most data visualization, color is important for creating essentially an "extra dimension", so it is often very important, and the lack of it is a major drawback for visualization.

A variety of methods for implementing *active stereo* are available, but they are all conceptually similar. They have some means of directing the correct image to the matching eye directly. The most common example is shutter glasses, which have an small black/clear filter over each eye. When activated, the filters alternate flipping on and off (i.e. the right shutter is open while the left is closed and vice versa), while the image on the image source is flipped between the two images at the same frequency. Hence, when one shutter is open, the appropriate image for that eye is active, and when the flip occurs, the other image is shown. If this occurs at 60 Hz or higher, the human eye can't perceive the difference between it and a steady image, so it appears as a constant stereoscopic image. While giving a very high quality stereoscopic image, as well as providing for color, this method requires the purchase of glasses for each viewer, which is expensive for a large presentation. Moreover, because of the 60 Hz requirement, only Cathode Ray Tube (CRT) monitors or projectors can be used currently; Liquid Crystal Display (LCD) monitors and projectors simply cannot switch that quickly.

The method best suited for presentation is the *polarized projection* method. In this setup, two projectors must be used, each with a polarizing filter. Note that the projectors must produce unpolarized light, something that is not true for many LCD projectors. These filters are set with their polarization axis perpendicular to each other, and one image is directed to the first projector, while the other image is sent to the second. The viewers wear special glasses that have polarizing filters over each eye oriented appropriately so that one filter blocks out one of the images, and the other blocks out the other. Assuming polarization is preserved in the reflection off the screen (usually requiring a special kind of screen), the polarized glasses then block out one of the projectors' image, but allows the other through. Hence, the stereoscopic effect is seen, and depth is perceived. Because polarized glasses are very cheap to obtain, this method is fairly cost-effective.

While all of these methods result in the stereoscopic effect, the main limitation is in the hardware used to produce these image pairs. Most consumer graphics cards are not designed to support this functionality as efficiently as they could, and to access the built-in OpenGL stereo functionality, special workstation graphics cards (that are somewhat more expensive) are required. Anaglyph can be performed on any graphics card, however, and dual-output graphics cards can be used with the projection/polarization method (although somewhat less efficiently than the OpenGL stereo mode).

Chapter 3

Forms of 3D Numerical Data

Having established a method to use to show three-dimensional data in stereoscopic 3D, some sense must be made of what sorts of numerical data can or should be visualized in three dimensions. While almost any data can be projected into two dimensions (usually by taking two-dimensional slices of the data and showing enough of them to see details of the higher dimensions), many types of data are inherently three-dimensional and are more easily visualized in their natural space. The five primary types of data used in numerical modeling are dot fields, height fields, parametric systems, scalar fields, and vector fields. Note, however, that many of these data are based on systems that are in some sense dependent on a fourth dimension (generally time), and hence these three-dimensional data are also a kind of slices of a four-dimensional data set.

One important distinction must be recognized between these data types: height fields, scalar fields, and vector fields are functions imposed on a rectilinear coordinate system — the data is considered to be a continuum that fills a space, and the finite data used in generating the field is usually viewed as just a sampling of the continuous function. Because of this, they can be imposed upon a rectilinear coordinate system (Cartesian, in the software used for this thesis). By contrast, dot and parametric fields have data that can take any position in the three-dimensional space, and thus are not viewed as filling the space.

I will also present a hypothesis for each regarding the value of the stereoscopic visualization method for improving understanding of the data. This will be compared with the results from the thesis presentation.

3.1 2-D Scalar Fields (Height Fields)

While not, strictly speaking, three-dimensional data, a data type that is still valuable to visualize in three dimensions is that of the height field. This data is a function of two dimensions (usually x and y) that has a single scalar value at each point. By using this scalar value as the third coordinate (usually z), the data can be visualized in three dimensions. Figure 3.1 shows an example of this representation. This method is commonly encountered in topographic maps of elevation on the earth's surface — the z coordinate is a natural choice here because it shows the actual three-dimensional shape of the topography in question (ignoring the earth's curvature). This data type is also common in two-dimensional scientific models, such as potential maps of two-dimensional electrical systems.

An alternative (two-dimensional) representation of this data type is contour maps. Contour maps plot curves of constant value — they indicate where the field has the same scalar value. An example of this (for similar data as that represented in Figure 3.1) is shown in Figure 3.1. While this is a two-dimensional representation of two-dimensional data, it is still in some sense composed of slice of three-dimensional data, because the height values used to determine the contours are the third dimension. Furthermore, this representation is a more familiar version of the technique of level surfaces that can be used for representing 3-D scalar fields.

The hypothesis for this data type is that the stereoscopic effect is of limited utility. Because this data generally has little overlap (due to the two-dimensional data), depth perception only provides a cosmetic improvement — the patterns can still be ascertained without it.

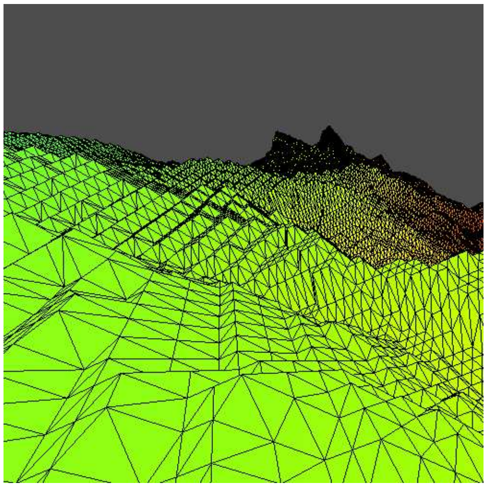


Figure 3.1: A representation of a height field (topographical data of Mt. Rainier in Washington, USA)

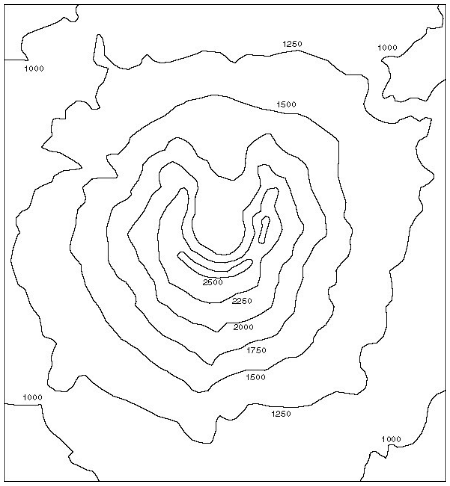


Figure 3.2: A contour map of a height field (topographical data of Mt. St. Helens in Washington, USA)

3.2 3-D Parametric Systems

Parametric systems are described by a set of functions that are dependent on an independent variable. In three dimensions, there are three of these functions, so that each value of the independent variable can be mapped to a unique point in three-dimensional space. By plotting a set of points over regular time intervals, a three-dimensional curve (sometimes called a "trajectory") is obtained that represents the parametric data. While simple, these systems have a wide variety of applications in numerical modeling and data visualization. Examples include the path of a particle (usually under the influence of some force), where time is the independent variable. Another example could be the pressure, volume, and energy of a gas as temperature is varied.

The hypothesis for this data type is that stereo 3D is useful for systems with non-linear variation in all three dimensions. Systems that vary linearly in at least one dimension can be easily represented in the other two, and hence the system has little use for the additional depth information. In systems with variation on all three dimensions, however, no choice of viewing location will show the variation in all three dimensions without depth information, rendering stereoscopic visualization useful.

A famous example of this in mathematics is the Lorenz-Saltzman equations, with the x, y,and z coordinates obey the differential equations

$$\frac{dx}{dt} = -sx + sy \qquad \frac{dy}{dt} = -xz + rx - y \qquad \frac{dz}{dt} = xy - bz \tag{3.1}$$

where s,b,and r are parameters. This system is famous because it was the original example illustrating the principles of chaos theory (i.e. extreme sensitivity to initial conditions and parameters). A sample output of this system is in Figure 3.2 using red-cyan anaglyph stereo 3D (also compatible with red-blue glasses).

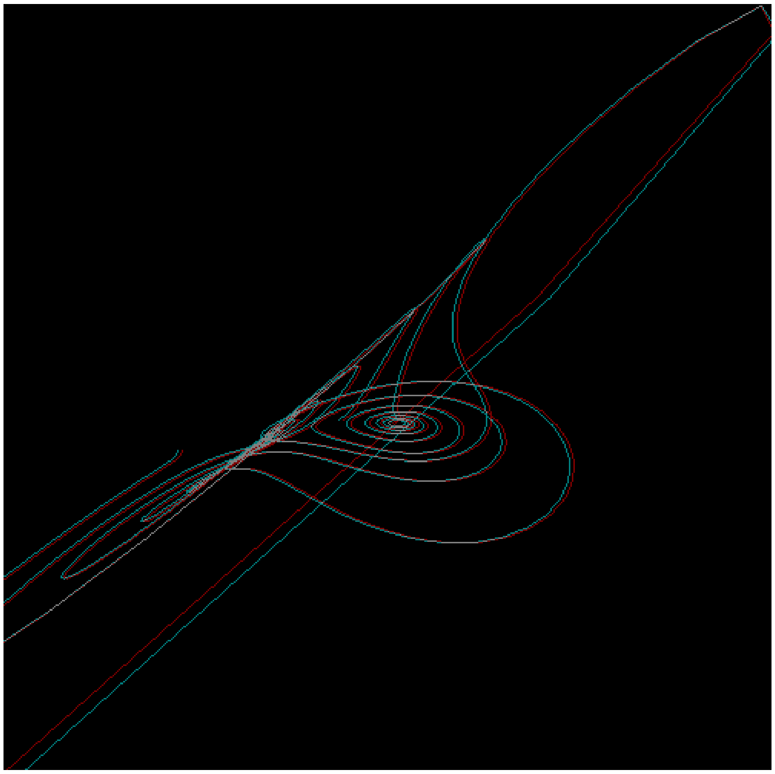


Figure 3.3: A representation using red-cyan anaglyph stereo of 1000 iterations at a Δt of .01 of the Lorenz-Saltzman equations with initial position (1,0,0) and parameters $b = 2.66667$,$r = 30.0$,$s = 10.0$

3.3 3D Scalar Fields

Three-dimensional scalar fields are a higher-dimensional analog to height fields. In 3D fields, however, a scalar value is associated with each point in three-dimensional space. These data types are exceedingly common because we live in a world that has three spatial dimensions[1]. Temperature in a room, electrical potential around a charged sphere, potential of a molecule, etc. are all examples of this data type. There is a difficulty in visualizing this data compared to the height field, however. There is no way to represent the scalar value using spatial dimensions, as they are all used up in indexing a particular location. Hence, representations must rely on other means of conveying this information. A simple method (as shown in Figure 3.3 represents the value at each point as a sphere of size proportional to the value. Another method (which I use in conjunction with the scaled spheres method in the Stereo Data Display Modules) colorizes the spheres according to their value and a color map or mapping function.

My hypothesis for this representation is that while 3D is absolutely necessary for viewing these data sets, the value of the stereoscopic effect is only realized with data sets that show nonlinearity in at least two dimensions, as that prevents a clear view into what is occurring in the interior of the data set, even if one of the dimensions is mostly linear.

Another important representation method available in this field is that of level curves. These are analogous to the contour map in Figure 3.1, but instead of curves of constant value, this data set uses surfaces of constant value, because the data is three-dimensional (levels use the space orthogonal to the gradient of the field, which has a dimension one less than that of the data). Because of the three-dimensional form space they are in, however, level surfaces will tend to obscure each other. Hence, graphical techniques of transparency are crucial in generating level surfaces, which is a further level of complexity from that of contour curves of height fields.

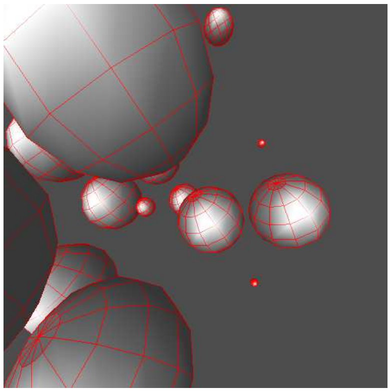


Figure 3.4: Scaled sphere representation of 3D scalar field

3.4 3-D Vector Fields

Functions that map from one dimension to three represented by parametric systems, while those mapping from three to one are represented using scalar fields. However, for functions that have a three-dimensional output for each three-dimensional input require a third type of field: the vector field. These fields are also very common in modeling and nature — wind, water flow in the ocean, and magnetic and electric fields are all vector fields.

[1] Some recent theories in physics suggest there may be more than three spatial dimensions - but these additional dimensions are always much smaller than the three well-known dimensions, so I will ignore this fact

The representation for these is also fairly intuitive — at grid points in the coordinate system, a three-dimensional arrow is drawn pointing in a direction dictated by the three output values, and the arrow is scaled or colored in the same fashion as the spheres in a scalar field. Another version more in line with the traditional representation of vector fields in two dimensions draws an arrow using line segments, and scales the arrow lengths to represent the magnitude of the vector.

Note that the two-dimensional analog of a three-dimensional vector field is much more direct than that for the scalar field — it is just a slice in two dimensions, because there is no way to completely represent a two-dimensional vector as the z coordinate of a height field (the magnitude can be used, but direction is lost). Instead, two-dimensional arrows are used (usually with line segments instead of two-dimensional block arrows).

An additional representation method is available for vector fields that is not present in scalar fields: streamlines. Streamlines are parametric curves that follow the vectors — they are generated by finding the direction of the vector at a starting point, and proceeding in that direction a fixed distance, and repeating the procedure. The resulting curve follows the path of the vectors, and with enough streamlines, can sample most of the vector field. However, this method can easily miss details of the field if none of the streamline (or streamlines, as more than one are usually drawn simultaneously) passes through an area that differs in pattern from the area it does pass through.

The hypothesis for this visualization method is that stereoscopic 3D is crucial. Because of the additional complexity introduced by the presence of the arrows, stereoscopic should be crucial in picking out details like the direction and magnitude of individual (relatively small) arrows. However, because of the availability of the streamline method, for simple vector fields with fairly uniform or linearly varying vector directions, the streamline method can be used and may not require stereoscopic 3D (see Section 3.2).

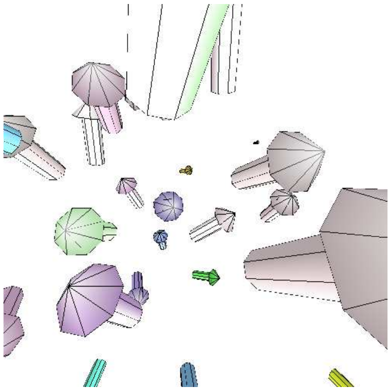


Figure 3.5: 3D vector field representation

3.5 Point Fields

At first, it seems that of the fully 3D data types, the point field is one of the simplest. A point field is a set of three-dimensional points, and hence is most easily represented as a set of dots in space. Figure 3.5 is an example of a dot field representation. The complexity of point fields arises from the fact that they have data attached to them in the sense of a scalar or vector field. In fact, in some sense the 3-D scalar and vector fields are specific cases of the point field. This is not as accurate of an interpretation as it first seems, however, because the inherent assumption in scalar and vector fields is that they are representing a continuous (or at least regularly periodic) field that permeates the space they are in, rather than a discrete set of points that are finite in number and extent. Because mathematical functions are more suited to this interpretation, non-rectilinear point fields are actually less common than the other two. Still, they are commonly used in fields such as astronomy (representing the location or other properties of, for example, stars in a galaxy), o statistical physics (where they may represent, for example, the location of molecules in a gas).

There is no hypothesis for this data type because it was not implemented in time for the presentation.

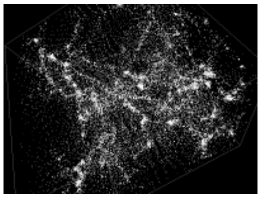


Figure 3.6: A simple dot field representation of a galactic star distribution model

Chapter 4

Electrostatics Model

Having established a sense of what sorts of data are visualized in stereo 3D, I turn to a particular numerical model that produces a number of these data types: electrostatics. Essentially contained within Poisson's equation and the definition of the electrical potential, electrostatics can generate a wide variety of both simple and complicated scalar fields and vector fields in two or three dimensions. Hence, it is an excellent testing ground for the utility of these visualization methods.

4.1 Essential Physics

The key equations of electrostatics are Poisson's equation and the (definitional) relation between electric fields and electric potential:

$$\nabla^2 V(\mathbf{r}) = \frac{\rho}{\epsilon_0} \tag{4.1}$$

$$\nabla V(\mathbf{r}) = -\mathbf{E}(\mathbf{r}) \tag{4.2}$$

where V is the electrical potential, ρ is a charge density, ϵ_0 is the permittivity of vacuum, and $\mathbf{E}$ is the electric field.

Given these relations, a method is available to solve any problem in electrostatics: given a charge distribution (determined by ρ) and boundary conditions, we can solve the differential equation (4.1). The resulting V is a scalar field (with dimension determined by the dimension of the charge distribution). $\mathbf{E}$ can be computed from V using (4.2), and it is a vector field (of the same dimension).

A few problems can be solved analytically - the simplest is that of free space (i.e. $\rho = 0$) with the traditional boundary conditions of $V = 0$ infinitely far from the origin. In this case, the solution is trivial — just $V = 0$ everywhere. Another important analytical solution is that of a $\rho = q\delta(\mathbf{r})$ — that is, a point charge of magnitude q centered at the origin. In this case, the solution (see [5]) is

$$V(\mathbf{r}) = \frac{q}{4\pi\epsilon_0 r} \tag{4.3}$$

where r is the magnitude of $\mathbf{r}$. The corresponding $\mathbf{E}$-field (*Coulomb's Law*) is often taken as the starting point for electrostatics (i.e. Equation4.1 can be derived from it) because it is experimentally verified to a very high degree of precision (see [11],[3],[6]).

Unfortunately, there are a large number of charge distributions that cannot be very easily solved analytically (and some that cannot at all). But because the overall behavior is governed by a fairly simple differential equation (Equation 4.1) and a scalar field (ρ), it can be computed using various numerical computational methods. The two I used were the method of Green's functions and the relaxation method.

4.2 Method of Green's Functions

The method of Green's functions can be applied to a wide variety of differential equations, particularly those of the form

$$(\nabla^2 - k^2)\psi(\mathbf{r}) = c(\mathbf{r}).$$

For the case of Poisson's equation, $k = 0$ and $c = \frac{\rho}{\epsilon_0}$. To solve this, we make use of Green's function $G(\mathbf{r})$, for which

$$\nabla^2 G(\mathbf{r}, \mathbf{r}') = \delta(\mathbf{r} - \mathbf{r}').$$

A Green's function that works here (see [5] or [10] for a proof) is

$$G(\mathbf{r}, \mathbf{r}') = \frac{1}{|r - r'|}.$$

And given that for Green's functions,

$$\phi(\mathbf{r}) = \int_{sphere} G(\mathbf{r}, \mathbf{r}') \frac{C(\mathbf{r}')}{4\pi} d\mathbf{r}',$$

so if we interpret this scalar quantity ϕ as the electrical potential, and $\frac{C}{4\pi}$ as the charge distribution $\rho(\mathbf{r})$, we have a way to calculate the potential at any point as an integral over a set of delta functions. Algorithmically then, we simply visit each point we wish to determine the potential at, and determine the contribution that it receives from each point charge as $\frac{q}{\mathbf{r} - \mathbf{r_0}}$.

An advantage of using this algorithm is that we are not forced to set boundary conditions — it is implicit in the formulation of this solution that the potential is 0 at an infinite distance from the charges. This allows easier comparison to theory, where this is often taken to be the boundary condition. More importantly, this can be computed quickly, or at least in a predictable amount of time, as each point in the field is simply visited once for each point with charge.

An example for this is Figure 4.2, a representation of an electric dipole.

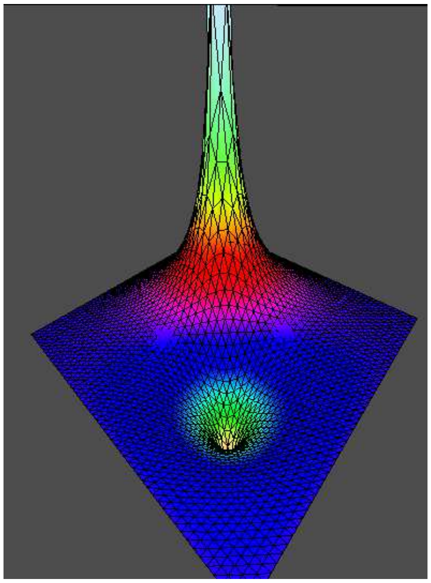


Figure 4.1: SDDM Screenshot of a height field representing the two-dimensional electric dipole as calculated using the method of Green's functions

4.3 Relaxation Method

While the method of Green's functions is conceptually fairly simple and computationally efficient for systems with the common boundary conditions of $V = 0$ at ∞, physical systems often have fixed potentials rather than fixed charges, because most electrical instrumentation is designed to produce voltages. Hence, many interesting problems specify potentials instead of (or in addition to) charge distributions, and want the potential and **E** field everywhere else based on those conditions. To do this, we need a method that allows specification of arbitrary boundary conditions. The *relaxation method* solves Poisson's equation (Equation 4.1) for arbitrary boundary conditions and charge distributions.

First we will use the relaxation method to solve the free-space poisson's equation ($\nabla^2 V = 0$) for a one dimensional system. Then we have:

$$\frac{d^2V(x)}{dx^2} = 0.$$

To perform numerical computation, we convert this from a continuous derivative to a finite difference. If three consecutive, uniformly spaced values of the potential are V_1,V_2, and V_3:

$$\frac{\Delta(\Delta V)}{\Delta x \Delta x} = 0,$$

but because the potentials are on a uniformly spaced grid so that $\Delta x = 1$,

$$\Delta(V_n - V_{n-1}) = V_3 - V_2 - (V_2 - V_1) = V_3 - 2V_2 + V_1 = V_1 + V_3 - 2V_2 = 0.$$

Rearranging, we get:

$$V_2 = \frac{V_1 + V_3}{2}.$$

Thus, the value of a potential at a particular point is the average of the neighboring values. If there is charge at the point in question, the equation changes to

$$\frac{\Delta(\Delta V)}{\Delta x \Delta x} = \rho$$

(in units of $[\rho] =$Cϵ_0). But this can just be folded into the average, giving

$$V_2 = \frac{V_1 + V_3 - \rho}{2}$$

Now to generalize, in three dimensions, Poisson's equation gives:

$$\frac{d^2V(x,y,z)}{dx^2} + \frac{d^2V(x,y,z)}{dy^2} + \frac{d^2V(x,y,z)}{dz^2} = \rho$$

Applying the same argument, the potential is the average of the six nearest neighbors. Furthermore, the charge at a particular point is included in the average, giving:

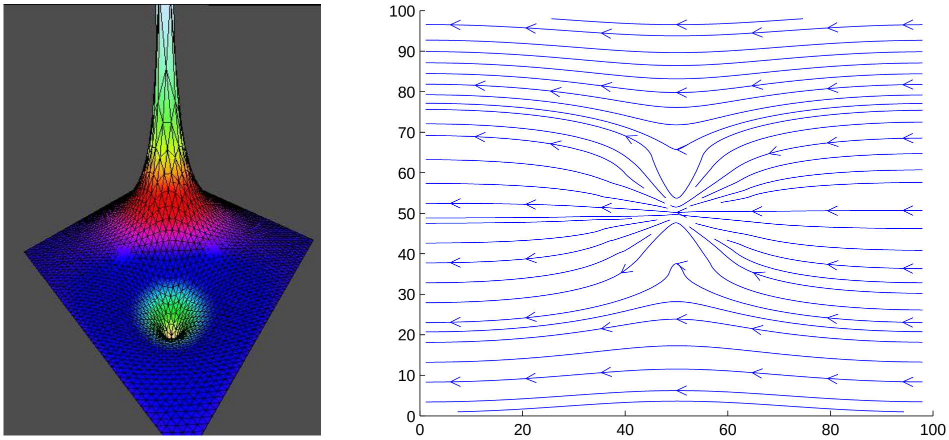
$$V(x,y,z) = \frac{V(x+1) + V(x-1) + V(y+1) + V(y-1) + V(z+1) + V(z-1) + \rho}{6} \tag{4.4}$$

this can be used in two dimensions as well by removing the z terms and changing the 6 to 4.

With this relation in hand that holds for all solutions of Poisson's equation, a computer algorithm becomes apparent. The steps in the algorithm are:

1. Specify an initial state ($V = 0$ everywhere, random values, or some cleverer field to make the algorithm run faster).
2. Set fixed potentials.
3. Specify the charge distribution.
4. Visit each point except the fixed potential points, and set the value based on Equation 4.4
5. Repeat Step 4 until the change in the field at each iteration is below the required tolerance

This can be used in any number of dimensions to generate the potential field as necessary. From this result, Equation 4.2 gives $\mathbf{E}$, and the problem is solved. Figure 4.3 is an example potential field output that uses the relaxation method to generate a uniform electric field and places a dielectric in this field.



(a) (b)

Figure 4.2: (a)SDDM Screenshot of a height field representing a dielectric circle in a uniform electric field. (b) 2D electric field map.

4.4 Model Behavior and Experimental Comparisons

An important fact to recognize regarding these numerical models is that they both operate on discrete grids, while the analytical methods (and the usual physical interpretation) view V and $\mathbf{E}$ as continuous fields. While this does not affect V or $\mathbf{E}$ significantly as long as the grid size is small compared to the features of interest, one serious problem with a discrete grid size is in the computation of ρ. In the case of the relaxation method or in the presence of dielectrics, the charge distribution of the resultant field is not the same as the initial ρ used in the calculation. For a continuous field, the charge distribution is $\rho = -\nabla \cdot \mathbf{E}$, which when changed to a finite difference, becomes

$$\rho = -(V_1 - V_2 + V_3 - V_2) = 2V_2 - V_1 - V_3$$

which has the same form in three dimensions. For many charge distributions (even, for example, the point charge: see Figure 4.4), this computation produces spurious charge distributions of a significant magnitude where "phantom" charges seem to exist that are not in the physical situations.

This is a result of the discrete Cartesian calculation making errors of what should be a radial calculation (due to the potential's $1/r$ dependence for electric charges). The divergence is computed using the difference between the nearest neighbors of the point it is being calculated at, but if the variation is radial, these calculations will not cancel out as it should if the cartesian grid is not aligned along the radial direction (and it is not except along the lines that pass through the point). Hence, the calculation should be zero around a point charge, but it is not computed as such (see Figure 4.4) This is an inherent limitation in the grid system, and can only be rectified through use of special divergence calculation methods that account for the expected dependence.

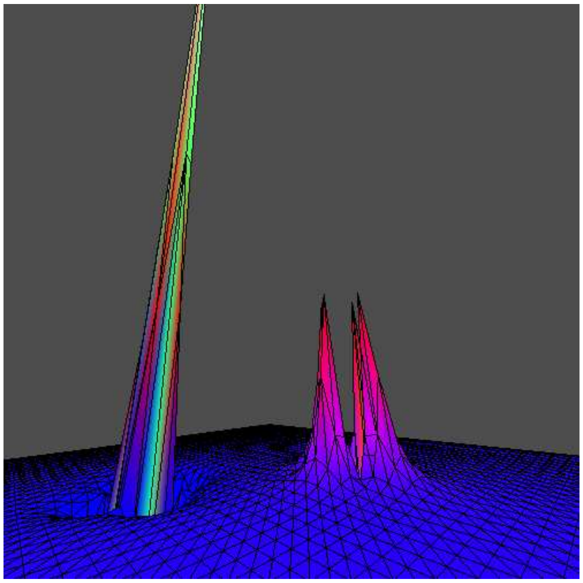
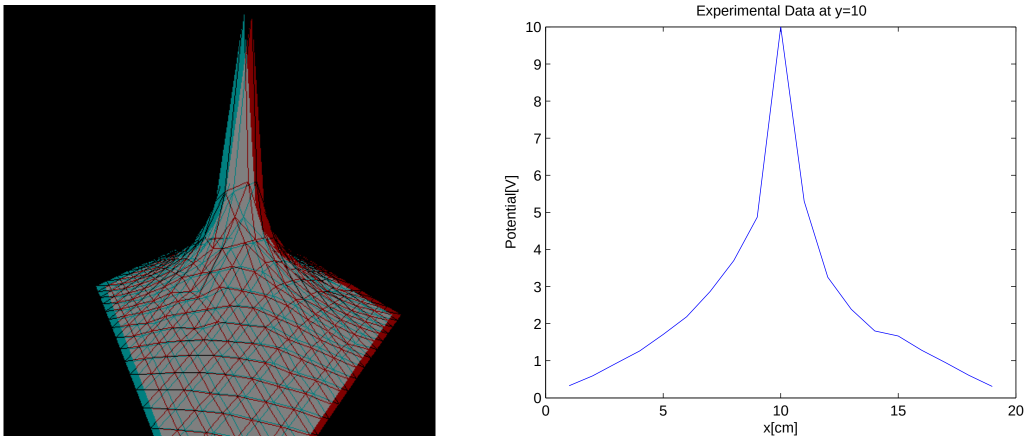


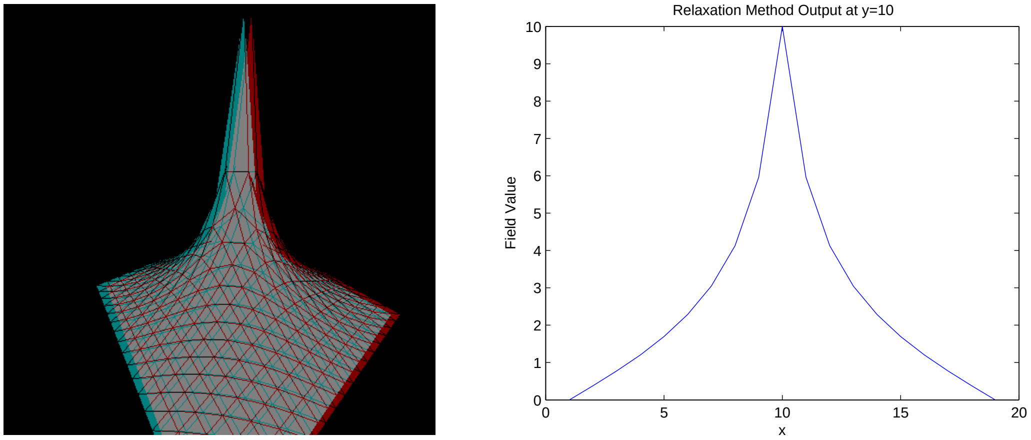
Figure 4.3: SDDM Screenshot of the calculated divergence of a dipole electric field generated using the Green's function method. Electrostatic theory predicts this should be one each of a positive and negative spike filling in only a single grid point.

Another important question is how well the relaxation method really matches experiment. Figure 4.4 shows the data of an experiment performed using poorly conducting paper with conducting paint placed in a square box and grounded, corresponding to a 2D relaxation method simulation with boundary conditions of all 0 at the outer edges. A dot was added in the center of the box, and set to a potential of 10 V, matching the numerical simulation with a point charge in the center. The potential from ground was measured at each point in a 19-by-19 grid inside the boundary, resulting in the the field in Figure 4.4. An equivalent field (Figure 4.4) was generated with the Relaxation method using a fixed potential of 10 in the center and 0 at the boundaries. When compared to the experimental data, this model has an RMS deviation of 0.0447 V, showing that the relaxation method can closely match real physical potentials. For comparison, a Green's function field was generated with the appropriate charge to produce a central peak of 10 V (Figure 4.4). This field has an RMS deviation of 0.2741 V, due to the fact that that method uses boundary conditions of 0 at infinity, rather than at the edge of the field.



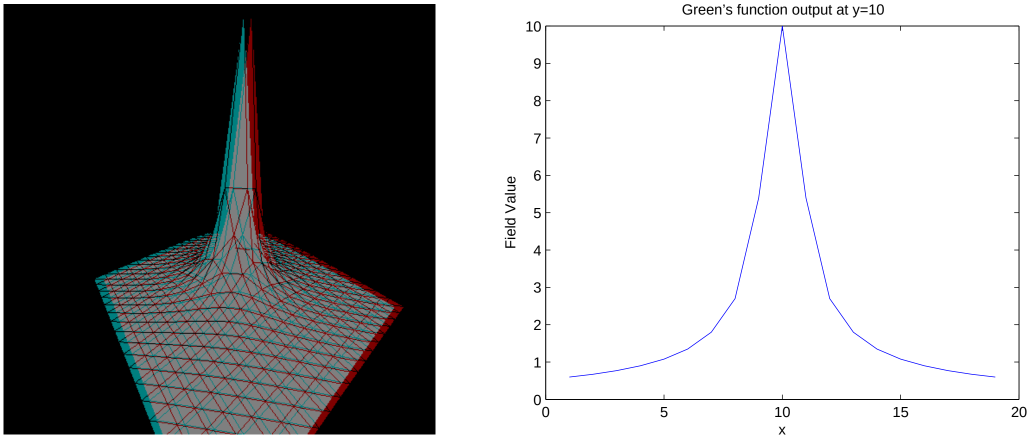
(a) (b)

Figure 4.4: (a)SDDM screenshot in red-cyan anaglyph of experimental data from conductive paper with grounded outer boundary and central 10 V point potential. (b) Plot of y=10 line of same data.



(a) (b)

Figure 4.5: (a)SDDM screenshot in red-cyan anaglyph of relaxation method output. (b) Plot of y=10 line of same data.



(a)

(b)

Figure 4.6: (a)SDDM screenshot in red-cyan anaglyph of Green's functions method output. (b) Plot of y=10 line of same data.

Chapter 5

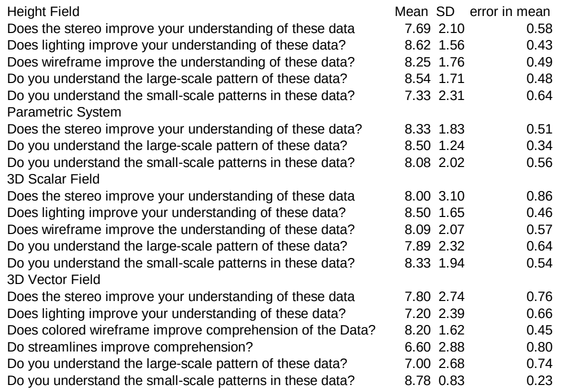
Results from Presentation

5.1 Presentation

This thesis was presented February 24, 2006 by showing a set of height, parametric, 3D scalar, and 3D vector fields using the polarized projection method. A survey was provided to those who attended, with questions as shown in Figure 5.2. For each data type, two data sets were used. The first height field was that of a part of Mt. Rainier (beginning with a familiar object), while the second was the potential of an electric dipole in two dimensions. For the parametric system, two sets of parameters were used to generate Lorenz-Saltzman (Equation 3.1) trajectories, one of which was mostly flat (two-dimensional in $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$), and the other had variability in all three dimensions (Figure 3.2). For the 3D scalar field, the two data sets used were an electric dipole's electrical potential, and a geopotential field (weather data from the National Oceanic & Atmospheric Administration). Finally, for the 3D vector field, I used the electric field from an octopole (a 3D charge distribution that cannot be represented in 2D), and the geostrophic wind (a vector field derived from the geopotential that gives wind direction and magnitude). These data sets were each presented in turn, with various settings turned on and off to give the audience a way to determine which factors made the patterns in the data set more obvious.

5.2 Results

Statistical results of the survey are given in Figure 5.2.



Height Field	Mean	SD	error in mean
Does the stereo improve your understanding of these data	7.69	2.10	0.58
Does lighting improve your understanding of these data?	8.62	1.56	0.43
Does wireframe improve the understanding of these data?	8.25	1.76	0.49
Do you understand the large-scale pattern of these data?	8.54	1.71	0.48
Do you understand the small-scale patterns in these data?	7.33	2.31	0.64
Parametric System			
Does the stereo improve your understanding of these data?	8.33	1.83	0.51
Do you understand the large-scale pattern of these data?	8.50	1.24	0.34
Do you understand the small-scale patterns in these data?	8.08	2.02	0.56
3D Scalar Field			
Does the stereo improve your understanding of these data	8.00	3.10	0.86
Does lighting improve your understanding of these data?	8.50	1.65	0.46
Does wireframe improve the understanding of these data?	8.09	2.07	0.57
Do you understand the large-scale pattern of these data?	7.89	2.32	0.64
Do you understand the small-scale patterns in these data?	8.33	1.94	0.54
3D Vector Field			
Does the stereo improve your understanding of these data	7.80	2.74	0.76
Does lighting improve your understanding of these data?	7.20	2.39	0.66
Does colored wireframe improve comprehension of the Data?	8.20	1.62	0.45
Do streamlines improve comprehension?	6.60	2.88	0.80
Do you understand the large-scale pattern of these data?	7.00	2.68	0.74
Do you understand the small-scale patterns in these data?	8.78	0.83	0.23

Figure 5.1: Results of numerical thesis presentation survey questions. Answer options ranged from 1 to 10. Total number of respondents was 13.

A variety of trends are shown in this data, but the most significant lies in the comparison of the large-scale understanding as compared to small-scale. For the height field and parametric systems, the large scale patterns tended to be the most apparent to audience members. This (as the comments bear out) suggests that the first two suffer from overlap problems - the arrows and spheres overlap and block each other so that only local details are apparent, and there when more information is available (e.g. when the view is zoomed out so the whole system is visible), the information is just too much and it becomes very difficult to understand. Another factor that was important (mentioned in comments) was that motion (particularly rotation) of the camera was very helpful in seeing the three-dimensional patterns, greatly enhancing the stereoscopic effect.

As for specifics on the individual data types, it was confirmed that the stereo effect was less useful for the height fields than it was for the other data types. Lighting was also noted as a very effective indicator of three-dimensionality. This is particularly interesting because the algorithm used to determine the lighting was, in fact, still incomplete at the time the presentation was given, and the lighting was flawed in that it gave erroneous lighting conditions for sharp corners and edges.

The parametric system, however, was surprising: it benefited the most (and the most consistently) from the stereoscopic effect, compared to the other three data types, both for the system that showed great variation as well as the for trajectory that was mostly flat. The large-scale pattern here was also slightly more apparent, although the short size of these trajectories limits the usefulness of the "small-scale" and "large-scale" divisions.

The 3D scalar field, meanwhile, had mixed reactions. Some viewed stereoscopic as crucial, while others said it was not at all important. All agreed, however, that rotation was helpful. Furthermore, lighting was perceived as a valuable technique to use for these fields, as well.

The 3D vector field, despite the similarity in general structure to scalar fields, had fairly different results. The large-scale patterns were relatively difficult to comprehend due to the fact that the arrows all blur together when a wide enough view angle is provided. And while I expected the availability of streamlines to alleviate this problem, the streamlines, in fact, were not seen as a valuable resource. The colored wireframe representation, however, was well-recognized as the method providing the best visual effect.

Chapter 6

Conclusions

In the end, I found that all my hypotheses were correct. However, this is not surprising, given the fact that I wrote the software that realized them. In comparing how other people saw the data, a nuanced overall conclusion is reached: stereo is valuable, but only for some people. As indicated by the comparatively large spreads in the answers to the first question of each data set (and more so by the comments), for some people the effect was striking and useful, while to others it was little more than eye candy. The balance, however, is in favor of the former: this representation does help to improve data comprehension for many viewers.

There are a few areas in particular where further study is warranted. Aside from the dizzying array of representation options available in an implementation of a dot field module, the use of level surfaces to study 3D scalar fields could provide another means to understand the field. Another question that ended up unanswered was what difference, if any, existed between understanding for those trained in math and science and those who were not (the audience at the presentation was roughly half and half). It is reasonable that those so trained would recognize these representations more readily, but whether the stereoscopic effect improves that understanding, or hinders it (because of their training in 2D representations) is an open question as of yet.

In any event, it is clear that this method of data visualization making use of stereoscopic vision has great potential as a method to augment existing visualization methods, and can be accomplished at relatively low cost. Furthermore, studying what factors improve visualization is, in fact, of value. Often in the sciences, specialists get used to their traditional methods of data representation, and ignore the fact that newer methods may be of value for showing their results to those not trained in the field, or even for themselves. This kind of study (done within the scientific community in question) can, hence, be of great value.

Appendix A

Acknowledgments

Thanks to my thesis committee: James Bernhard, Andrew Nierman, and Doug Canon for their support and suggestions. Also to the UPS Honors program for help in the presentation and providing the impetus for this project. In addition, I'd like to thank Edward Tollerud and the National Oceanographic & Atmospheric Administration for use of numerical data produced by one of their models.

Bibliography

[1] Kravchenko Alexei and Mekhontsev Dmitriy. Fractal gallery. Stereographics and OpenGL Sections. URL: http://fractals.nsu.ru/gallery_en.htm.

[2] Paul Bourke. Paul bourke - personal pages. Stereographics and OpenGL Sections. URL: http://astronomy.swin.edu.au/~pbourke/.

[3] Lewis P. Fulcher. Improved result for the accuracy of coulomb's law: A review of the williams, faller, and hill experiment. *Physical Review A (General Physics)*, 33(1):759–761, 1986.

[4] Joseph F. Gregory. Linear perspective in antiquity : a note on reading the evidence. In *Kenneth C. Lindsay : essays on art and literature.* State University of New York at Binghamton, 1999.

[5] David J. Griffiths. *Introduction to Electrodynamics.* Prentice Hall, 3rd edition, 1999.

[6] J. A. Lipa, J. A. Nissen, S. Wang, D. A. Stricker, and D. Avaloff. New limit on signals of lorentz violation in electrodynamics. *Physical Review Letters*, 90(6):060403, 2003.

[7] Harvey R. Schiffman. *Sensation and Perception: An Integrated Approach.* John Wiley&Sons, 5th edition, 2001.

[8] Christopher W. Tyler. Perspective as a geometric tool that launched the renaissance. URL: http://www.ski.org/cwt/CWTyler/ArtInvestigations/PerspectiveHistory/Perspective.BriefHistory.html.

[9] Wikipedia. Alhazen — wikipedia, the free encyclopedia. http://en.wikipedia.org/wiki/Alhazen, 2006. [Online; accessed April-2006].

[10] Wikipedia. Green's function — wikipedia, the free encyclopedia. http://en.wikipedia.org/wiki/Green's_function, 2006. [Online; accessed April-2006].

[11] E. R. Williams, J. E. Faller, and H. A. Hill. New experimental test of coulomb's law: A laboratory upper limit on the photon rest mass. *Physical Review Letters*, 26(12):721–724, 1971.