Geometry of Binocular Imaging

Victor S. Grinberg
Gregg Podnar
M. W. Siegel

Robotics Institute, School of Computer Science, Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA, 15213

ABSTRACT

We rigorously present the geometric issues related to binocular imaging. We identify the minimum number and most fundamental conceptual set of parameters needed to define 3D-stereoscopic camera and display systems; the fundamental parameter that is needed to specify a 3D-stereoscopic system but not a monocular system is the pupillary distance. We analyse the constraints that are imposed on the values of the parameters by the requirement that the imagery be geometrically indistinguishable from the reality that would be perceived by the "naked" human visual apparatus. We relate our approach to those employed by several well known textbooks and graphics engines.

1. INTRODUCTION

All binocular systems must provide a means for presenting one image to the left eye and another to the right. We use as a basis for discussion a system where images for both the left and right eyes are presented on a single, flat surface and where a means to ensure that each eye sees only the intended image is employed. Our approach derives from first principles a framework for understanding presentation of binocular images and the corresponding display technologies. Irregularly shaped display surfaces, or two separate (not necessarily co-planar) displays can be also be considered using these techniques. This approach can also be adapted to the design of stereo cameras and other image generation systems.

Most monocular "3-D" perspective systems need not consider real sizes and often do not. However, when discussing binocular imaging, it is absolutely necessary to deal with real sizes due to the simple fact that the human pupillary distance is fixed. For a natural view, all binocular perspective projections must be referenced to the pupillary distance. When this fact is ignored distortions appear which result in unrealistic stereo views and often cause discomfort to the viewer. We make a strong distinction between natural binocular imaging and the introduction of artificial aspects which can include changing the pupillary distance, magnification ("zooming"), and others. Artificial distortions can sometimes be used to emphasize or de-emphasize certain aspects of the scene to the viewer, but should be undertaken with the understanding that not only does the scene appear unnatural, but the viewer's perceptual system is further taxed.

Using more than one frame-of-reference in which to calculate perspective images is a typical convenience. We introduce certain frames-of-reference which lend themselves to an understanding of how one may wish to interact with a three-dimensional simulation. We name our primary frames-of-reference the *World Coordinate System* and the *Vehicle Coordinate System*.

We describe all objects in the simulated world in the World Coordinate System. In the World Coordinate System we also describe the position and orientation of a *Vehicle*. By introducing the concept of a Vehicle, movements of the viewer's position in the world (e.g., flying) can be naturally understood and calculated in a straightforward manner. The same is true when considering the interaction of the viewer with the display, such as when head-tracking hardware is employed. The Vehicle Coordinate System is used to describe the position of the viewer's eyes. This approach allows us to represent the display screen as the windshield of the Vehicle. Note that the position of the viewer's eyes is of paramount importance, whereas the "direction of view" is immaterial to calculating the perspective projections.

We establish the connection between this rigorous formal model and the conventional models and parameters of several classical computer-graphics textbooks and computer-graphics company's software systems.

2. CONCEPTS OF BINOCULAR IMAGING

2.1. The role of central projection in stereoscopic imaging*

Stereoscopic imaging is one of the most powerful means to use 2D images for realistically simulating the appearance of 3D objects viewed with two eyes. It involves two technological problems:

1) an image capture tool for two 2D images - one for the left and one for the right eye;
2) an image display tool that allows each eye to see only the image intended for it.

The initial problem that has to be solved is: what is the connection between the "real world" and the corresponding 2D images? This question has a universally recognized answer that we cannot formulate better than in Silicon Graphics™ manual[3] (p. 7-4):

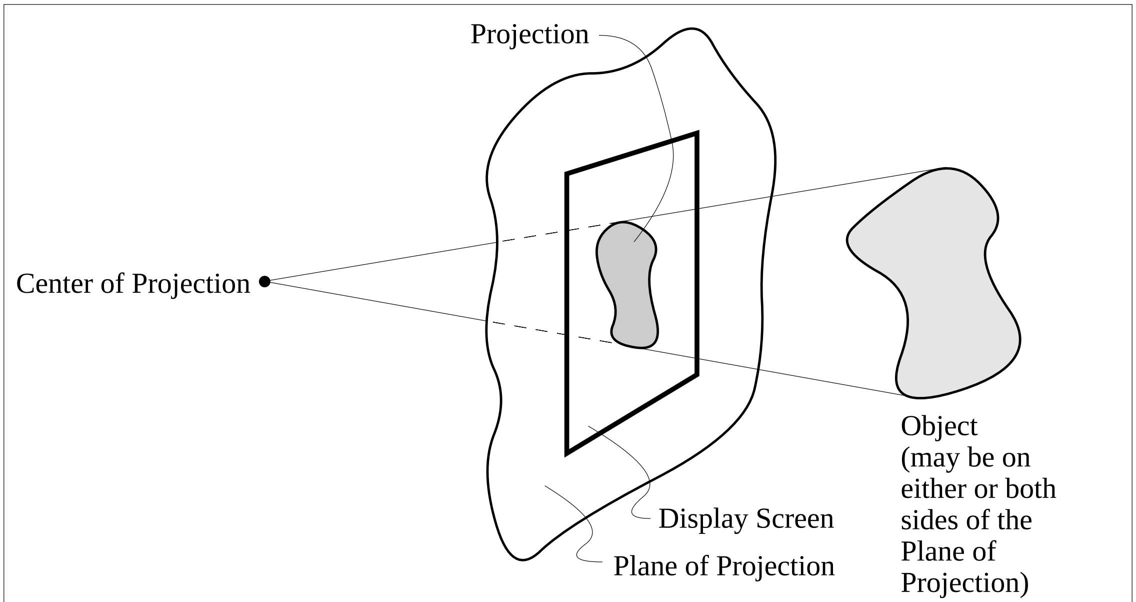
> "Viewing items in perspective on the computer screen is like looking through a rectangular piece of perfectly transparent glass. Imagine drawing a line from your eye through the glass until it hits the item, coloring a dot on the glass where the line passes through the same color on the item. If this were done for all possible lines through the glass, if the coloring were perfect, and if the eye not allowed to move, the picture painted on the glass would be indistinguishable from the true scene."

Of course this is true not only for the computer screen but for any viewing surface on which we draw the image.

The described procedure is called central projection. In this paper we consider only the geometrical aspects of central projection, putting aside intensity, color, etc. To create a central projection one needs to know:

1) the position and orientation of the screen; keeping in mind that the projection can be created on any surface (for example on some part of a computer screen) we use the term *Display Screen* for denoting the bounded part of a surface within which the projection is created;
2) the position of the eye, called the *center of projection*.

Thus the Display Screen simulates the transparent window through which a viewer looks (Fig. 1). In the rest of this paper we use the schematic two-dimensional representation illustrated in Fig. 2.

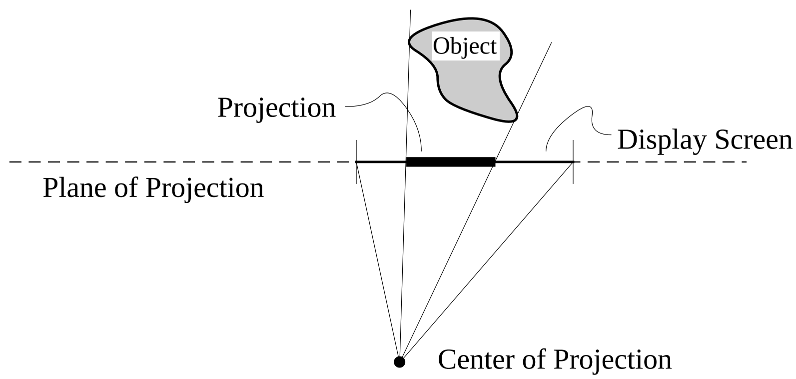


Perspective Projection of an object onto a planar Display Screen.

Figure 1

---

*After we finished writing this paper we discovered that the facts and conclusions in this section were previously noted in the first volume of the present series[6]. The title of the article, "Stereoscopic Displays for Terrain Database Visualization", gives no hint of the fundamental contribution it makes (in its section 3.3) to the topic. Furthermore the existence of three more volumes in the series in which many authors casually disregard the facts suggests that this material bears repetition in a complete, rigorous, and accessible format.



Schematic two-dimensional representation of Figure 1.

Figure 2

It is easy and worthwhile to enumerate the number of parameters that are necessary and sufficient to create a central projection. The position of a Display Screen as a solid body is defined by six numbers, and the position of a center of projection requires three numbers more. To fix the size of the (rectangular) Display Screen, two additional parameters are needed, e.g., the width and height of the Display Screen. For two eyes that use the same Display Screen one has fourteen independent parameters because the size of the Display Screen and its position are the same for both eyes.

Additional prevalent concepts and terms such as "line of sight", "direction of view" and the associated parameters (see J. D. Foley et al.[1], Silicon Graphics™ manual[3], S. Wolfram[4], A. Woods et al.[5]) are irrelevant in our context; all that one needs to describe a pair of perspective projections on a single Display Screen is the fourteen enumerated parameters.

These considerations dictate how to create stereo images in the case when they are being created directly on the Display Screen: the projection plane must coincide with the Display Screen and the centers of projection must coincide with the viewer's eyes. Sometimes, however, images are created in several steps, for example one can create images on an image sensor and then project them onto the Display Screen. The problems that arise in such cases can be solved by simple geometrical considerations. Let us consider this case in detail as an example.

2.2. Should left and right image sensor planes be parallel ?

Assume two cameras are used to create a stereoscopic image pair. Old questions that are still discussed widely are: What is the correct distance between them? What is the correct angle between image sensor planes? Should the lenses be offset relative to the centers of the frames? We assume that cameras are ideal, that is they create central projections of the scene on the image sensor planes through well defined centers of projection in the lenses that are called the *optical centers* of the cameras.

The answers to these questions depend on how the resulting images will be displayed and viewed! The difficulty lies in the fact that rather rarely are the recorded projections intended for direct viewing; usually they are again projected (not necessarily by central projection) onto another surface (or multiple surfaces) for viewing. The answer depends on how these second projections are done and how they are viewed.

Let us make natural assumptions: the recorded frames are fully projected on the same Display Screen in such a way that their images on the Display Screen are similar (i.e., scaled without distortion, all sizes are proportional) to the frames. To be specific and simple, let us further assume that both of the viewer's eyes are the same distance from the Display Screen and the point half-way between the eyes lies on the perpendicular to the Display Screen that passes through its center. In this case each frame must be similar to the corresponding central projection of the scene onto the Display Screen. Inspection (simple geometrical considerations, Fig. 3) shows that in this case when taking the picture:

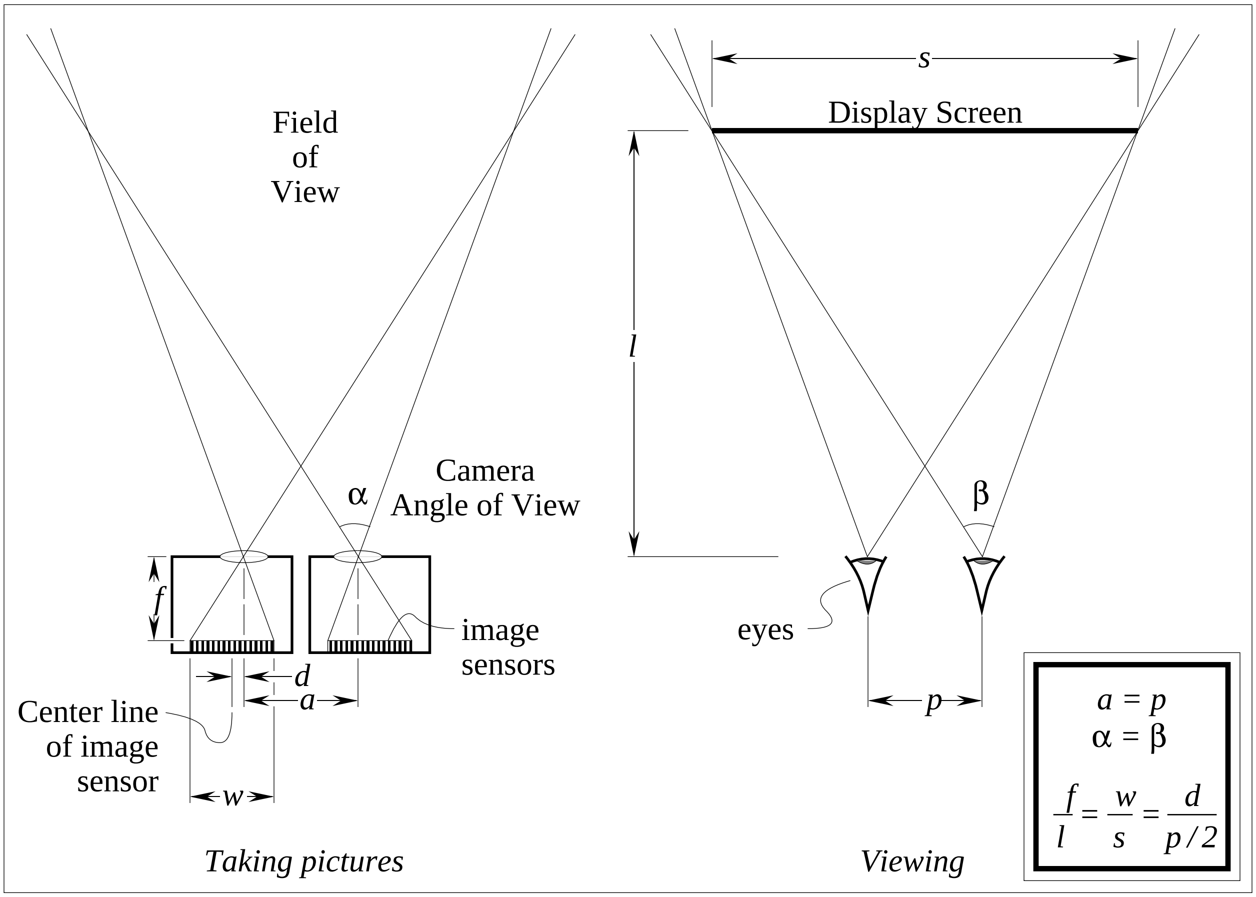


Figure 3

1) the image sensors must be parallel;
2) the optical centers should be offset relative to the centers of frames;
3) the distance between the two optical centers must be equal to the pupillary distance**: $a = p$;
4) the following equalities must be kept:

$$\alpha = \beta$$

$$\frac{f}{l} = \frac{w}{s} = \frac{d}{p/2} \quad .$$

The set of equalities is redundant. We state all because which are more fundamental is a matter of taste.

These conditions are necessary and sufficient for taking and viewing perfect stereo frames as long as our cameras and projectors are ideal (as assumed). The old masters knew this well: their stereoscopic cameras were designed exactly according these rules; a good example is the popular Stereo Realist of the 1950's.

**This policy is sometimes intentionally violated for important technical reasons, e.g., binoculars have objectives that are further apart than the oculars in part to counteract the scene flattening that accompanies their telescopic effect; instruments that provide enhanced depth perception, e.g., for the participants in tank battles, are widely used, but the result is "unreality", albeit a desired kind of unreality.

Our conclusions are experimentally confirmed by A. Woods et al.[5]. The authors used videocameras instead of film, and so were able to "project" the image created on their image sensors onto the screen idealy, without distortion, which is much easier to do electronically than optically. The authors conclude that "the parallel camera configuration is used in preference to the toed-in (converged) camera configuration".

3. PROPOSED FRAMES-OF-REFERENCE

We restrict our considerations to the case when both images are projected on the same flat rectangular surface (Display Screen), and suitable hardware is used to restrict each eye to seeing only the view intended for it.

3.1. World Coordinate System

The World Coordinate System (WCS) is an arbitrary orthonormal coordinate system. We refer to the WCS axes as the 0-axis, 1-axis and 2-axis. The WCS is used to describe the real-world objects that will be imaged. The WCS is also used to describe the position and orientation of the imaginary Vehicle defined below. For any point object **O** we denote by $\mathbf{O_w}$ the triple of coordinates of **O** in the WCS.

3.2. Vehicle Coordinate System

The Vehicle Coordinate System (VCS) is a left-handed orthonormal coordinate system embedded in the Display Screen. It has its origin in the center of the Display Screen. We refer to the VCS axes as the 0-axis, 1-axis and 2-axis. The 0-axis is parallel to the horizontal edge of the Display Screen and directed right for a typical view; the 1-axis is parallel to the vertical edge of the Display Screen and directed up for a typical view; the 2-axis is orthogonal to the Display Screen and directed away from the typical viewer. For any point object **O** we denote by $\mathbf{O_v}$ the triple of coordinates of **O** in the VCS.

3.2.1. Display Screen

The position and orientation of the Display Screen (i.e., of the VCS) is defined by:

1) the triple $\mathbf{V_w}$: the WCS-coordinates of the VCS origin;
2) the 3x3 transformation matrix **T** whose columns are the VCS-coordinates of the WCS basis vectors.

Because both the WCS and the VCS are orthonormal coordinate systems, **T** is unitary, and thus can be also defined by the rows being the WCS-coordinates of the VCS basis vectors.

There are simple formulas that link the WCS- and the VCS-coordinates of any point **O**:

$$\mathbf{O_v} = \mathbf{T}(\mathbf{O_w} - \mathbf{V_w})$$

and

$$\mathbf{O_w} = \overline{\mathbf{T}}\,\mathbf{O_v} + \mathbf{V_w} \quad ,$$

where $\overline{\mathbf{T}}$ stands for **T** transposed.

3.2.2. Eyepoint

The location of an eye is defined by the triple $\mathbf{Y_v}$ - the VCS-coordinates of the eye.

Triple $\mathbf{V_w}$ contains three independent parameters; the same is true for triple $\mathbf{Y_v}$. The matrix $\mathbf{T}$ contains nine terms, but because it is unitary these terms cannot be chosen arbitrary: they are bound by six independent relations

$$(\overline{\mathbf{T}}\,\mathbf{T})_{ij} = \delta_{ij} \quad ,$$

where δ_{ij} - Kronecker symbol:

$$\delta_{ij}=\begin{cases} 1 & \text{if } i=j \\ 0 & \text{otherwise} \end{cases} \quad .$$

So the nine components of $\mathbf{T}$ depend on only three independent parameters, essentially the three angles needed to describe the orientation of the VCS in the WCS or vice-versa. In all we have nine independent parameters that are necessary and sufficient for projection calculations. We omit the corresponding formulas that are simple and straightforward. The position of the projection of a point is naturally defined in the VCS. Its 2-coordinate is equal to 0 since the projection lies on the Display Screen.

For creating the two projections for the left and right eyes one needs three more parameters, namely the position of another eye. For doing this instead of $\mathbf{Y_v}$ we use triples $\mathbf{Y_v^{\mathit{l}}}$ and $\mathbf{Y_v^{\mathit{r}}}$ that are the VCS-coordinates of the left and right eye respectively.

The advantages of these frames-of-reference become manifest when we describe below the natural changes of positions of the Display Screen and the eyes.

3.3 "Flying"

"Flying" is changing the position of the Vehicle (i.e., a Display Screen) in the real world without changing the position of the eyes relative to the Display Screen. It represents what a person sees through the window of a moving Vehicle. Insofar as the eyes remain stationary relative to the Display Screen, parameters $\mathbf{Y_v^{\mathit{l}}}$ and $\mathbf{Y_v^{\mathit{r}}}$ do not change as a result of Flying. Parameters $\mathbf{V_w}$ and $\mathbf{T}$ change according to the nature of the flight path.

3.3.1. Translation

Assume that the Vehicle is shifted without changing its orientation. Let $\mathbf{S_v}$ be the shift vector in the VCS. In this case matrix $\mathbf{T}$ is not changed, and the new value of $\mathbf{V_w}$ can be calculated as $\mathbf{V_w^{\mathit{new}}} = \mathbf{V_w^{\mathit{old}}} + \overline{\mathbf{T}}\,\mathbf{S_v}$.

3.3.2 Rotation around a VCS axis

In this case the triple $\mathbf{V_w}$ is not changed. For rotation through angle α around the 0-axis the new value of matrix $\mathbf{T}$ can be calculated as

$$\mathbf{T}_{0j}^{new}=\mathbf{T}_{0j}^{old}$$

$$\mathbf{T}_{1j}^{new}=\mathbf{T}_{1j}^{old}\cos(\alpha) - \mathbf{T}_{2j}^{old}\sin(\alpha)$$

$$\mathbf{T}_{2j}^{new}=\mathbf{T}_{1j}^{old}\sin(\alpha) + \mathbf{T}_{2j}^{old}\cos(\alpha) \quad .$$

Rotation around other axes is described in an analogous manner.

3.3.3 Arbitrary displacement

An arbitrary displacement is a superposition of an elementary translation and the three rotational displacements described above, so new values for $\mathbf{V_w}$ and $\mathbf{T}$ can be calculated for any displacement of the Vehicle. We believe that defining a displacement in the VCS is more natural than in the WCS. It imitates the driving of the Vehicle when a driver thinks in terms of "forward", "backward", "left turn", "right turn" defining the desirable displacements *relative to the current position of the Vehicle* rather than to the objects he sees through the window. Therefore the formulas above are written for the case when the VCS is used for defining the translation and the axes of rotation. However if the WCS reference frame is for some reason preferred, these formulas can be easy modified by applying the transformation from the WCS to the VCS and back.

3.4. Eye's displacement inside the Vehicle

Now assume that the Vehicle is not moving but the viewer's eyes are moving relative to the Vehicle. The advantages of our frames-of-reference are especially apparent in this case. The parameters that make up $\mathbf{V_w}$ and $\mathbf{T}$ are unchanged, and the new positions of the eyes are explicitly defined in the VCS. For example a shift of both eyes is described by formulas

$$\mathbf{Y}^l_\mathbf{v}(new) = \mathbf{Y}^l_\mathbf{v}(old) + \mathbf{S_v}$$
$$\mathbf{Y}^r_\mathbf{v}(new) = \mathbf{Y}^r_\mathbf{v}(old) + \mathbf{S_v} \quad ,$$

where $\mathbf{S_v}$ is the shift of the eyes in the VCS.

4. CONCEPTS, TERMINOLOGY AND FRAMES-OF-REFERENCE USED IN STANDARD TEXTS, ETC.

4.1. W. M. Newman and R. F. Sproull, Principles of Interactive Computer Graphics[2]

The book doesn't say much about the things that we consider, perhaps because it is rather old. However we would like to note that the authors use the term "viewing direction", a concept which is superfluous in our formulation. The authors omit formal definitions of parallel and perspective projections and sometime neglect to distinguish between them.

4.2. J. D. Foley, A. van Dam, S. K. Feiner, J. F. Hughes, Computer Graphics: Principles and Practice[1]

The book contains a formal definition of perspective projection and a detailed description of its properties. A frame-of-reference is introduced that is sufficient to describe an arbitrary perspective projection. The authors avoid the confusing terms "viewing direction", "line of sight" and the like. Unfortunately the paragraph "Stereopsis" (pp. 616-617) occupies less than a page in the book and contains only basic definitions. Therefore the authors do not have a chance to use their frames-of-reference to describe the changes in stereoscopic images that accompany a viewer's interaction with the system. We believe that our frames-of-reference are more intuitive that those used in the book.

4.3. S. Wolfram, Mathematica™ [4]

The book is a manual of the widespread commercial software package of the same name. This package contains powerful means for 3D imaging. Perspective projection is used as an internal tool to create images. The available set of parameters is not sufficient to create an arbitrary perspective projection: the center of projection is always constrained to lie on the perpendicular to the Display Screen that passes through its center. Thus the high-level features of the package are regrettably unusable for creating perfect stereoscopic images.

4.4. Silicon Graphics™ Computer Systems, Graphics Library Programming Guide[3]

The book is a manual for the computer graphics package supplied with a variety of Silicon Graphics™ computers. The package includes several routines for 3D imaging. Arbitrary projection can be created by combining routines "window" (a rather bad name for a routine that creates perspective projection) and "lookat". However the set of parameters used by the combination is actually redundant i.e., different sets of parameters create exactly the same picture. The meaning of the parameters is described in a rather obscure way; in fact the easiest way to understand the exact meaning of the SGI parameters seems to be to experiment with them, which is what we did. We wrote routines that convert the SGI parameters to our WCS and VCS parameters and vice-versa, resolving the mentioned redundancy in definite way. Then, by experimenting with simple graphics, the exact meaning of all the SGI parameters was discovered. It turned out that the mysterious "line of sight" is the perpendicular from the eye to the plane of the Display Screen. In some situations, e.g., when this line does not even pass through the Display Screen (Fig. 4), this definition seems unnatural.

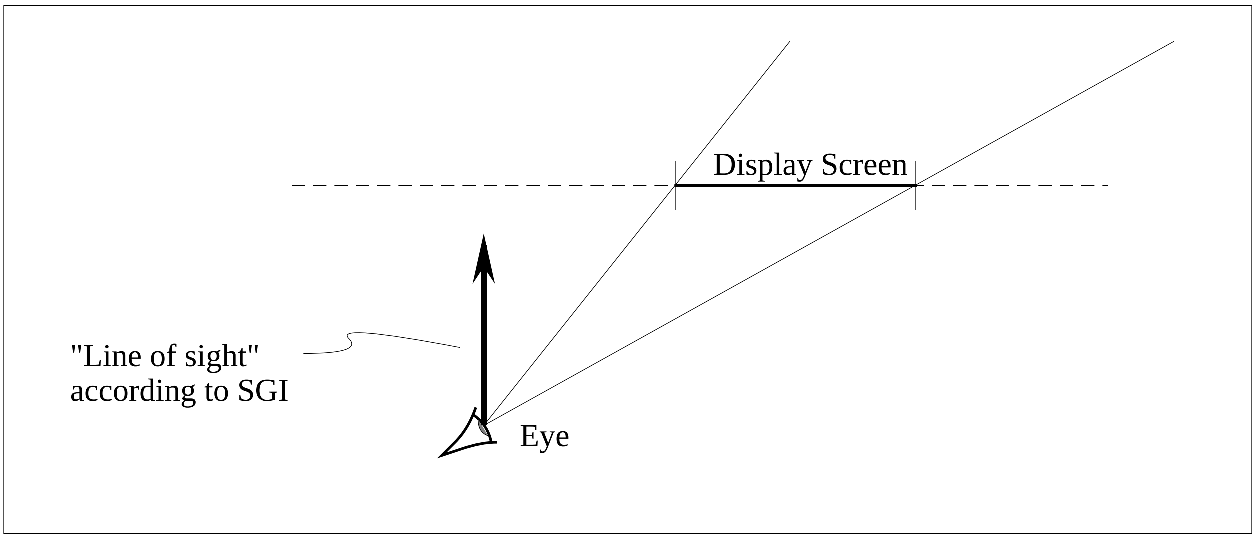


Figure 4

The choice of parameters used in the SGI package we think is rather poor. For example the picture on the screen depends on certain parameters discontinuously (unlike our set). Visualizing the natural motions of the Display Screen and eyes requires extensive and convoluted calculations. Nevertheless the package can be used to create perfect stereoscopic images; however the viewer's interaction with these images is necessarily difficult to program.

Also the paragraphs devoted to the topic (pp. 7-1 through 7-15) contain mistakes, misprints and incorrect figures.***

***We would be happy to share our errata with interested readers.

5. POSSIBLE GENERALIZATIONS

We have discussed in detail the case when both eyes use one flat rectangular Display Screen. We see the following situations in which it might be useful to relax these restrictions.

5.1. Curved screens

Most existing displays have slightly curved screens and some (e.g., Omnimax™) and various dome-simulators are dramatically curved, so it makes sense to create stereoscopic images taking this into account. Projection onto a curved surface is not difficult in principle; however implementation requires detailed knowledge of the screen's curvature and corresponding means for its description. Standard rendering algorithms may not be applicable to non-planar screens, and the computational burden can increase significantly.

5.2. A separate screen for each eye

Some existing stereo systems ("Virtual Reality" with HMDs) use two screens for imaging, one per eye. This case requires that two perspective projections must be created not only with two different centers of projection, but also onto two different planes. The number of necessary parameters and the amount of calculation increase correspondingly.

5.3. Multiple Display Screens

Several stereo Display Screens can be created on the same display simultaneously. This approach opens ways to simulate, for example, a rear view mirror in the Vehicle. This case requires us to expand our formalism slightly. It is unnatural in this case to tie the VCS with a Display Screen, since we have more than one Display Screen. It seems more natural to define the VCS as a coordinate system tied with a "real Vehicle", and then to define the position and orientation of each screen in the VCS.

6. ACKNOWLEDGMENTS

This paper grew out of numerous discussions, sometimes quiet, usually loud, in groups sometimes large, usually small, involving at one time or another all the faculty, staff, and students working on 3D-stereoscopy under ARPA High Definition Systems Grant MDA972-92-J-1010. The authors acknowledge the contributions made during these discussions by Tom Ault, Alan Guisewite, Priyan Gunatilake, Joe Mattis, Jeff McVeigh, Steve Roth, Scott Safier, and Sriram Sethuraman; we also acknowledge that not all of them (yet) fully share our conceptual framework. We of course alone accept responsibility for any errors.

7. REFERENCES

1. J. D. Foley, A. van Dam, S. K. Feiner, J. F. Hughes, Computer Graphics: Principles and Practice, 2nd ed., Addison-Wesley, 1991.
2. W. M. Newman and R. F. Sproull, Principles of Interactive Computer Graphics, 2nd ed., McGraw-Hill, 1979.
3. Silicon Graphics™ Computer Systems, Graphics Library Programming Guide, Document Number 007-1210-050., 1993.
4. S. Wolfram, Mathematica, 2nd ed., Addison-Wesley, 1991.
5. A. Woods, T. Docherty, R. Koch, "Image Distortions in Stereoscopic Video Systems", in *Stereoscopic Displays and Applications IV*, John O. Merrit, Scott S. Fisher, Editors, Proc. SPIE 1915, pp. 36-48, 1993.
6. H. Veron, D. A. Southard, J. R. Leger, J. L. Conway, "Stereoscopic Displays for Terrain Database Visualization", in *Stereoscopic Displays and Applications*, John O. Merrit, Scott S. Fisher, Editors, Proc. SPIE 1256, pp. 124-135, 1990.