Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

J. Hübl, I. Herman

Modeling clip: some more results

Computer Science/Department of Interactive Systems Report CS-R9008 February

The Centre for Mathematics and Computer Science is a research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a nonprofit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands Organization for the Advancement of Research (N.W.O.).

Modeling Clip: Some More Results

Josef Hübl

DIGIDATA
Gesellschaft für Digitale Datensysteme mbH
Goethestr. 17
D-8000 München 2
Federal Republic of Germany

Ivan Herman

Centre for Mathematics and Computer Science
Dept. of Interactive Systems
Kruislaan 413
NL-1098 SJ Amsterdam
The Netherlands
ivan@cwi.nl

ABSTRACT

Abstract The modeling clip of the PHIGS ISO Standard is mathematically analysed. The most important result of this analysis is the fact that the projective image of a modeling clip body (that is a not necessarily bounded convex body in space) is simply the union of two convex bodies. Furthermore, it will also be proved that in some cases one of these two bodies is empty. This fact makes the implementation of the modeling clip fairly straightforward and makes it also possible to use all already existing results on clipping against general convex bodies without change.
1983 CR Categories: G.0,I.3.2, I.3.5
Keywords & Phrases: projective geometry in computer graphics, computer graphics standardisation, PHIGS, PHIGSPLUS, clipping.
Note: the present text is published in: *Computer Graphics Forum*, **9**(1990), No. 2.

1. Introduction

The modeling clip as defined in the PHIGS Standard[6] has already stirred quite a lot of interest in the past years[3,8,9,12]. The reason is that the way modeling clip is described in the ISO document generates a range of algorithmic problems; indeed, the convex body which is used for clipping purposes has to be transformed by a projective transformation first and the clipping step itself can be done only after having performed this transformation. The reason of this is bound to the way the modeling clip is specified in PHIGS; more details about it can be found for example in the paper of Herman and Reviczky[3]. This fact leads however to the problem of determining what exactly the image of a modeling clip body will be under the effect of a projective transformation or at least how this clipping can be performed without getting into conflict with the official PHIGS specifications.

In the paper cited above a solution is also proposed to perform this clip. It is based on the idea of assigning a hyperspace of the four dimensional Eucledian space to each of the planes defining the original clipping body. In some cases, however, it is necessary to perform the clipping after the *full* transformation (that is including the projective division). There exist for example algorithms which enable the generation of more complicated output primitives once the projective transformation has been performed. Examples are given in the two papers of Herman[4,5]. For these

Report CS-R9008
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

methods to work neatly with modeling clip an approach should be necessary which would perform this clip after the whole projective transformation.

A significant step in this direction has been made by Krammer in[8]. His idea was to introduce the so-called "conic sectors", which are areas defined by a pair of planes in the projective space. A usual halfspace is also a special case for a conic sector by taking the original boundary plane of the halfspace and the ideal plane as generating planes. Krammer has shown that such conic sectors are projective invariant, that is that the image of a conic sector remains a conic sector. In his approach, the modeling clip is done a series of clips against such conic sectors much in the same way as a usual clip is performed as a series of clips against halfspaces.

Unfortunately, Krammer has not made the last step, that is to try to describe what the intersection of all these conic sectors will be. As it will be shown in the sequel, this point-set (which is therefore the clipping body to be used for modeling clip) is surprisingly simple: it is the union of two "traditional" convex bodies; furthermore, both of these bodies can be described very easily.

2. Notational Conventions and Some General Remarks

In the sequel, we will denote by $\mathbf{R}^3$ and by $\mathbf{R}^4$ the set of three and four dimensional vectors respectively and by $\mathbf{PR}^4$ the set of four dimensional homogeneous vectors. All vectors are considered to be column vectors; we will use the notation v^T for all vectors $v \in \mathbf{R}^3$ or $v \in \mathbf{R}^4$ to denote the *transpose* of the vectors (that is row vectors). To make a clear difference, lower case letters will be used for 3-element vectors (that is elements of $\mathbf{R}^3$) and capital letters for the homogeneous 4D vectors. If $p \in \mathbf{R}^3$, we will denote by $P \in \mathbf{R}^4$ the homogeneous version of this vector (by putting the value of 1 as last coordinate) Matrices will be denoted by bold capital letters like $\mathbf{V}$.

A general convex body, whether bounded or not, is described as the intersection of a finite set of halfspaces. In PHIGS, one defines such a convex body by defining a series of halfspaces in the structure store; at structure traversal, an actual set of such halfspaces defines the current modeling clip body.

In PHIGS, each halfspace is described by three data as follows:

- a point of the plane which gives the boundary of the halfspace,
- an interior point of the halfspace and
- a (three dimensional) normal vector pointing toward the demanded halfspace.

By using homogeneous vectors instead of Eucledian ones, the half space can be described by one single vector as well; indeed, a four dimensional vector E can be given so that the set

$$\{ p \in \mathbf{R}^3 : \sum_{i=1}^{3} E_i p_i + E_4 \geqslant 0 \} \tag{2.1}$$

is exactly the demanded halfspace. It is quite straightforward how the vector E can be calculated out of the original data. By taking P instead of p (that is putting 1 as the last coordinate value), the relationship above can also be written by putting:

$$\{ P \in \mathbf{PR}^4 : E^T P \geqslant 0 \} \tag{2.2}$$

The very same E vector can be used to describe the points of the (boundary) plane itself: indeed, instead of the inequalities in (2.1) and (2.2) we have to use equalities. The following statement can be proved easily and can also be found in a number linear algebra textbooks as well as in some papers like the technical note of Zachrisen[12]:

> If E describes a plane in $\mathbf{R}^3$ and $\mathbf{V}$ is a regular projective transformation (that is a transformation described by an invertible 4×4 matrix), the image of the plane is described by the vector
>
> $$E'^T = E^T \mathbf{V}^{-1} \tag{2.3}$$

If two vectors E_1 and E_2 are given, we will use the notation $\{E_1, E_2\}$ to denote the intersection of the halfspaces generated by E_1 and E_2 respectively. By recursion, $\{E_1, \ldots, E_n\}$ will denote the convex area defined by the vectors $E_1, \ldots, E_n$.

3. The Image of a Halfspace

The first question we have to ask ourselves is: what will be the image of a halfspace under the effect of a projective transformation? In the paper of Herman and Reviczky[3] it has already been shown that this image can be distorted in the course of a transformation; in most of the cases it will *not* be a halfspace again.

Let us take a regular projective transformation $\mathbf{V}$ and a halfspace defined by the vector $E \in \mathbf{R}^4$ (using formulae (2.1)/(2.2)). Let $P' \in \mathbf{R}^4$ be the four dimensional vector which is the result of the matrix vector multiplication and by $p' \in \mathbf{R}^3$ the vector derived from P' by the projective division. Clearly, p' is the image of p under the full projective transformation. We will denote by $V_{i,j}$ the elements of the matrix representation of $\mathbf{V}$ with $i,j = 1,2,3,4$.

Let us suppose that the projective division can be done, that is, $P_4 \neq 0$. (In his already cited paper Krammer has shown how to get rid of such singular points before the projective transformation; this supposition can therefore be done). In this case equation (2.3), which is equivalent to:

$$E'^T P = \sum_{i=1}^{4} E'_i P'_i \tag{3.1}$$

can be rewritten in the form:

$$E'^T P = P'_4 \sum_{i=1}^{4} E'_i (P'_i / P'_4) \tag{3.2}$$

We can change the second multiplicative term of (3.2) to:

$$\sum_{i=1}^{3} E'_i p'_i + E'_4 \tag{3.3}$$

What we get is as follows: the point $p \in \mathbf{R}^3$ is a point of the half space defined by E if and only if

$$P'_4 \left(\sum_{i=1}^{3} E'_i p'_i + E'_4 \right) \geqslant 0 \tag{3.4}$$

holds. In the second term of (3.4) we may recognise the same formula as in (2.1); indeed, it refers to the half space described by E'. Let us also remark that the plane defined by E' is just the image of the plane defined by E.

To get a clearer formula we have to concentrate on the first term of (3.4). Clearly, we are not interested in the actual *value* of this term but only in its sign; what we will do is to give an equivalent formulation for $sign(P'_4)$.

We have already used the fact that by choosing an appropriate vector $E \in \mathbf{R}^4$ we can describe a plane. In fact, the very same formulation works if we want to describe the *ideal* plane as well: indeed

$$E_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \tag{3.5}$$

describes all ideal points. Furthermore, formula (2.3) remains also valid to describe the image of the ideal plane. This image may be ideal again (which means that the transformation is affine, like for example a parallel projection) or it will be a "normal" affine plane. The reader may find additional details about ideal points, lines and other elements of projective geometry in the book of Penna and Patterson [10] or any other traditional textbook or tutorial of projective geometry.

Let us come back to our original problem. The following is true:

$$1 = E_0{}^T \begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ 1 \end{pmatrix} \tag{3.6}$$

that is

$$1 = E_0{}^T \mathbf{V}^{-1} \begin{bmatrix} P'_1 \\ P'_2 \\ P'_3 \\ P'_4 \end{bmatrix} \tag{3.7}$$

The first two terms give (according to (2.3)) $E_0{}'^T$, that is the vector representing the image of the ideal plane. We can multiply the second term with $P'_4(1/P'_4)$. According to the definition of p' we get:

$$1 = P'_4 E_0{}'^T \begin{bmatrix} p'_1 \\ p'_2 \\ p'_3 \\ 1 \end{bmatrix} \tag{3.8}$$

In other words:

$$sign(P'_4) = sign(E_0{}'^T p') \tag{3.9}$$

By combining the formulae (3.4) and (3.9) we arrive to the following statement.

If E defines a halfspace in $\mathbf{R}^3$ and $\mathbf{V}$ is a regular projective transformation, the image of the halfspace is the union of two convex areas, namely $\{E_0', E'\}$ and $\{-E_0', -E'\}$

This fact is nothing else then a more "analytic" version of the conic sectors described by Krammer. Let us also note that if $\mathbf{V}$ is affine, E' will be $(0,0,0,\lambda)^T$ (where λ is a non-zero number); in this case the one of the two terms automatically leads to an empty set (if λ is positive, the second term will be empty, the first otherwise).

The great advantage of this analytical description is that it is very easy to extend via induction to a general convex body. Indeed, the following is true:

If $C = \{E_1, \ldots, E_k\}$ is a convex body in $\mathbf{R}^3$ and $\mathbf{V}$ is a regular projective transformation, then the image of C will be the union of two convex bodies C' and C'' namely:

$$C' = \{E_0', E_1', \ldots, E_k'\} \tag{3.10}$$

and

$$C' = \{-E_0', -E_1', \ldots, -E_k'\} \tag{3.11}$$

As a result of this formulation the modeling clip has definitely lost its "frightening" nature: it can be performed after the projective transformation without problems. The only additional difficulty is that instead of one convex body two should be considered; however, both can be described easily with well known formulae.

When effectively performing the modeling clip against these convex bodies, what has to be done in a program is not much different of what Krammer has done in his description: in both cases, a series of halfspace clips should be done. However, the fact the *whole* clipping body can be described in one step as in (3.10)/(3.11) may have a great significance when some optimalisations are sought for clipping more complicated objects. We have already referred to the primitives described in the papers of Herman[4,5]. As a very different example let us refer to the problem of performing a modeling clip of a NURB (Non Uniform Rational B-Spline) surface or a NURB curve. Clipping these surfaces or curves is a significant algorithmic problem; however, very useful optimalisations can be achieved if we make use of the fact that the curve and/or the surface is in the convex hull of the NURB control points (see for example Farin[2] or Barsky et al[1] for more details). It is therefore a natural idea to compare these control points first to the whole clipping body to see whether something is visible at all and/or the overall clipping process of the NURB can be scaled down to some subarea of the NURB. We should not forget that NURB-s are basic primitives for example in the PHIGS PLUS specification[7] and, consequently, the modeling clip of NURB-s has become a very actual problem indeed!

Another advantage we can cite is the fact that it becomes straightforward to apply the filtering technique of O'Bara et al[9] for the construction of more complicated clipping bodies. Following their approach, it becomes possible to apply a clip against a more general point set in space, which are the result of some Boolean-like operations on convex bodies. Such operations are referred to in the PHIGS document although their implementation is not mandatory. However, because of the practical use such operations may have, it is appealing to have the possibility to implement them.

However, the most important consequence of our formulation is the fact that in some cases it is possible to reduce the number of clipping bodies from two to one. If this is the case, all possible complication with modeling clip can be forgotten: modeling clip can be done after having performed the projective transformation and it is just a normal clip against a well defined convex body. This is what we will do in what follows.

4. Reducing the Number of Clipping Bodies

The previous description of the modeling clipping body is surprisingly simple at a first glance. It is therefore worthwhile to make a little detour and to give a more intuitive picture of this description. On figure 1 the usual approach of presenting the effects of a projective transformation is shown; instead of projective space we have to show what happens in a projective plane but this is enough to help our intuition.

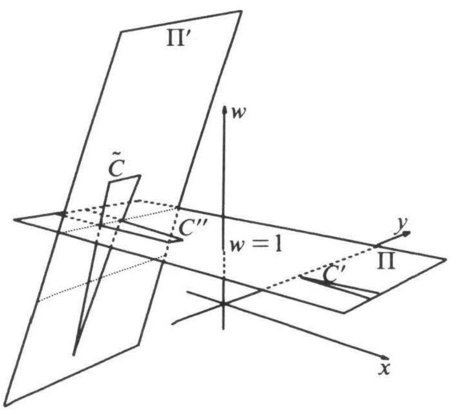

Figure 1

The plane Π is the original plane we start from; it is embedded into $\mathbf{R}^3$ by setting the last coordinate value to 1. The first effect of $\mathbf{V}$ is to transform via a linear mapping this plane onto another one in $\mathbf{R}^3$; this is denoted by Π'.

The modeling clip body (here a triangle) is transformed into another triangle of Π', denoted by $\tilde{C}$ on the figure. As a second step, this polygon is projected back onto Π using a central projection via the origin; this corresponds to the projective division. In our case, the result of this projection is the image of the original clipping body. It is now clear that as the polygon $\tilde{C}$ is cut into two by the $w=0$ plane, the result of the projection will effectively be the union of two convex bodies; this is exactly what our previous result states in a more precise and analytical form.

One would think that by having two concatenated transformations the number of convex bodies might be doubled at each step (if we were to transform the original clipping body through the concatenated matrices). Figure 1 also shows why this cannot happen: concatenation of two projective transformations can also be described by taking the product of the two corresponding matrices. However, this would just mean that the plane Π' should be transformed again into a second plane Π''; as the projective division has to be made after all matrix-vector multiplications only, there is no danger of having a higher number of convex bodies at the end.

If we look now at figure 2, we can remark that $\tilde{C}$ is now *not* cut by $w=0$. In this case, by projecting $\tilde{C}$ back onto Π, we get only one body instead of two! In other words, in such cases either C' or C'' is empty. It is clearly important to find out when this situation effectively happens. This is done more precisely in what follows.

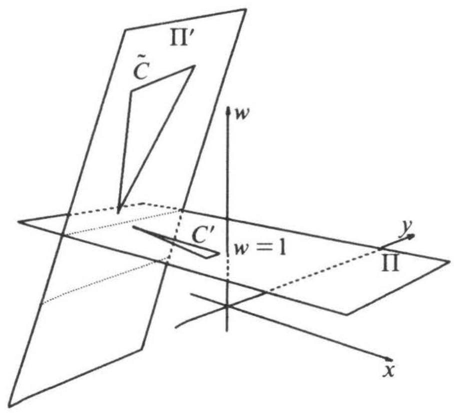

Figure 2

First of all we have to remind what the *vanishing plane* of $\mathbf{V}$ is. By definition, the vanishing plane is the plane which is mapped onto the ideal plane by $\mathbf{V}$. A vectorial representation of this plane can be given easily; indeed

$$E_V = \begin{pmatrix} V_{4,1} \\ V_{4,2} \\ V_{4,3} \\ V_{4,4} \end{pmatrix} \tag{4.1}$$

(that is the elements of the last row in the matrix) will do. This can be seen easily: for each $P \in \mathbf{PR}^4$, the following holds:

$$P'_4 = E_V{}^T P \tag{4.2}$$

that is this value is zero (in other words P' is ideal) if and only if P (more exactly the corresponding $p \in \mathbf{R}^3$) is on the plane defined by E_V and this is exactly the definition of the vanishing plane. Setting (eventually) the negative of this vector we can also use it to describe a given halfspace of the vanishing plane. We have to remark that the vanishing plane of an affine transformation (like the parallel projection) is the ideal plane.

With this definition at hand, we can state the following:

If the modeling clip body C is *fully* in one of the halfspaces of the vanishing plane of $\mathbf{V}$, either C' or C'' is empty.

Indeed, we can set the value of E_V so that it would define the halfspace containing C. If C is defined by $\{E_1, E_2, \ldots, E_k\}$, then clearly:

$$\{E_1, E_2, \ldots, E_k\} = \{E_1, E_2, \ldots, E_k, E_V\} \tag{4.3}$$

According to our previous theorem we have:

$$C' = \{E_0', E_1', \ldots, E_k', E_V'\} \tag{4.4}$$

and

$$C' = \{-E_0', -E_1', \ldots, -E_k', -E_V'\} \tag{4.5}$$

However, either E_V' or $-E_V'$ represents the ideal plane, consequently the last coordinate value of either E_V' or $-E_V'$ will be negative (the other three being 0); clearly, such a vector defines an empty set in $\mathbf{R}^3$.

The previous proof gives us also the method to handle such situation: an appropriate E_V should be chosen and then the last coordinate value of E_V' should be examined; the result of this step will decide whether C' or C'' is the empty set.

5. Simple Projective Transformations

It is not easy to use the previous theorem in practice. Indeed, deciding whether a (not necessarily bounded) convex body is contained in a given halfspace is not a simple question; a general solution to handle it leads to a typical linear programming problem. There is, however, a special class of projective transformations where this question becomes easier to cope with; as this class has a number of additional nice properties as well, it is worthwile to make a little detour in this direction.

Let us define a *simple projective transformation* to be a regular projective transformation $\mathbf{V}$ for which the following property holds:

There exists a real number $f \neq 0$ so that

$$\begin{aligned} V_{3,1} &= fV_{4,1} \\ V_{3,2} &= fV_{4,2} \\ V_{3,3} &= fV_{4,3} \end{aligned} \tag{5.1}$$

This class is fairly general. Indeed, it can easily be proved that the product of an affine and a simple projective transformation is a simple projective transformation again. This also means that the transformations defined by the utility functions for viewing defined eg in PHIGS, which realise the synthetic camera model, are in fact simple projective transformations. Indeed, the view orientation matrix produced by this utility is always affine while the view mapping matrix has its last row of the form $v_{3,1} = v_{3,2} = v_{4,1} = v_{4,2} = 0$ (see eg Singleton[11] or the book of Penna and Patterson[10] for details); the final viewing matrix to be used by PHIGS is the product of these two. The fact that a set of conditions of the form (5.1) can be given is of a great importance; indeed, in PHIGS (as well as in GKS-3D) the user calculates the viewing matrices *independently* of the function which actually sets them for the system. In other words, a proper implementation should be prepared to handle all kinds of projective transformations, even those which are not the realisations of the synthetic camera model. It becomes therefore of a great importance that a class of transformation would be found for which the decision whether a given transformation belongs to this class or not can be made solely by inspecting the matrix itself.

Two properties of simple projective projections may be of a general interest. If we calculate the image of a point $p \in \mathbf{R}^3$, we get the following formulae:

Let us define $\alpha = -fV_{4,4} + V_{3,4}$. Then:

$$P_3' = fP_4' + \alpha \tag{5.2}$$

and

$$p_3' = P_3'/P_4' = \alpha/P_4' + f \tag{5.3}$$

As p_3' corresponds to the z (that is depth) value of the image coordinate system, the second equation says that all planes which are parallel to the vanishing plane are mapped onto planes parallel to the xy plane. Indeed, all planes parallel to the vanishing plane can be characterised by the fact that their points have the same P_4' value. It is therefore much faster to compute p_3' using (5.3) than to perform the full matrix-vector multiplication for all coordinates. Furthermore, if we use the value of p_3' to perform Hidden Line/Hidden Surface calculations only (which is usually the case), it is enough to set

$$p_3' = -1/P_4' \text{ if } \alpha \leqslant 0 \tag{5.4}$$

or

$$p_3' = 1/P_4' \text{ if } \alpha \geqslant 0 \tag{5.5}$$

Krammer has described in[8] a way of getting rid of singular points of a projective transformation. His approach is roughly as follows. Out of the view volume defined in PHIGS (which is essentially a rectangular box) one has to construct two planes (denoted by U' and W') in $\mathbf{R}^3$ which are 1) parallel to the image of the ideal plane and 2) which contain the whole view volume (see figure 3). These planes have to be transformed by $\mathbf{V}^{-1}$ to get U and W; it can be proved that they will still be parallel planes and, furthermore, they will be parallel to the vanishing plane of $\mathbf{V}$.

In case the image of the ideal plane is not contained in the stripe defined by U' and W' (like on figure 3) it can also be proved that both U and W will be on the same side of the vanishing plane of $\mathbf{V}$ and, furthermore, by performing a clip against the stripe defined by U and W one gets rid of all those points which might lead to singularities without loosing any primitives (all primitives are to be clipped against the view volume anyway).

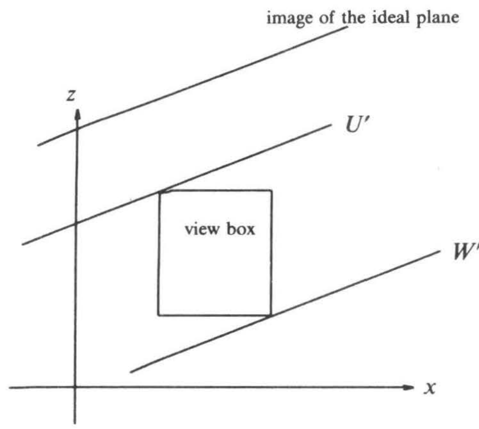

Figure 3

Clearly, in case of a simple projective transformation it is enough to use the planes $z_{\min}$ and $z_{\max}$ of the view volume to get the U' and W' planes. In most cases (eg for a synthetic camera model!) the image of the ideal plane will be outside the view volume as well; and this fact can always be decided fairly easily (the image of the ideal plane will be parallel to U' and W' and, therefore, this fact can be decided by a simple comparison against the z values). Consequently, by performing a clip against U and W we will not loose any output primitives and we will get rid of the possibly singular points.

Coming back to our modeling clip this also means that we might add to our modeling clip body definition the halfspaces defining the stripe between U and W. In this case, the resulting clipping body will effectively be on one side of the vanishing plane; that is, the image of the modeling clip body will still be one side only (by applying our theorem of the previous section). In other words, by adding these two planes to the modeling clip body, we can reduce the modeling clip step to a clip against one single convex body. Taking into account that the special case which has led to this optimalisation possibility encapsulates such important cases as the realisation of the synthetic camera model for viewing, this fact is of a great practical importance indeed.

6. Conclusion

In the sequel we have proved that the image of a convex body (that is a modelling clip volume) under the effect of a projective transformation is very simple to describe: indeed, it is the union of two, well describable convex volumes. In other words, the modelling clip of PHIGS can be performed once the whole transformation of the output pipeline, (that is essentially viewing), is completed and this can be done by using traditional clipping algorithms. Furthermore, we have also proved that in case of some very important special cases (for example when the synthetic camera model is used for viewing) the image of the modelling clip volume will be one volume again (that is one of the volumes cited above will be empty), and it is also computationally easy to find out which of the two volumes will be the empty one.

Acknowledgements

We are both grateful to our colleague and friend Jȧnos Reviczky for the valuable discussions on the first draft of this paper.

References

1. R.H. Bartels, J.C. Beatty, and B.A. Barsky, *An Introduction to Splines for Use in Computer Graphics & Geometric Modelling*, Los Altos California, 1987.
2. G. Farin, "Algorithms for Rational Bezier Curves," *Computer Aided Design* **15** (1983).
3. I. Herman and J. Reviczky, "Some Remarks on the Modelling Clip Problem," *Computer Graphics Forum* **7** (1988).
4. I. Herman, "2.5D Graphics Systems," in *Eurographics'89 Conference Proceedings*, ed. W. Hansmann, F.R.A. Hopgood and W. Strasser, North-Holland, Amsterdam (1989).
5. I. Herman, "On The Projective Invariant Representation of Conics in Computer Graphics," *Computer Graphics Forum* **8** (1989).
6. ISO, "Information processing systems - Computer graphics, Programmer's Hierarchical Interactive Graphics System (PHIGS) - Part 1, Functional description," ISO/IEC 9592-1 (1988).
7. ISO, "Information processing systems - Computer graphics, Programmer's Hierarchical Interactive Graphics System (PHIGS) - Part 4, Plus Lumière und Surfaces (PHIGS PLUS)," ISO/IEC 9592-4, rev. 3 (1989).
8. G. Krammer, "Notes on the Mathematics of the PHIGS Output Pipeline," *Computer Graphics Forum* **8** (1989).
9. R. M. O'Bara and S. S. Abi-Ezzi, "An Analysis of Modeling Clip," in *Eurographics'89 Conference Proceedings*, ed. W. Hansmann, F.R.A. Hopgood and W. Strasser, North-Holland, Amsterdam (1989).
10. M.A. Penna and R.R. Patterson, *Projective Geometry and Its Application to Computer Graphics,* Prentice-Hall, New Jersey (1986).
11. K. Singleton, "An Implementation of the GKS-3D/PHIGS Viewing Pipeline," in *Eurographics'86 Conference Proceedings*, ed. A.A.G. Requicha, North-Holland, Amsterdam (1986).
12. M. Zachrisen, "Yet Another Remark on the Modelling Clip Problem," *Computer Graphics Forum* **8** (1989).