Metarealistic Rendering of Real-time Interactive Computer Animations[*]

George K. Francis
Department of Mathematics
University of Illinois
gfrancis@uiuc.edu
http://www.math.uiuc.edu/∼gfrancis/

13 May 2001

Abstract

We develop a vocabulary for real-time interactive computer animation of mathematical homotopies. We distinguish metarealistic from photorealistic rendering and classify homotopies by the way they are implemented in immersive virtual environments. The nature of the geometrical pipeline and its OpenGL C-library suggest this progression: rigid motions, articulations, topological distortions of space, and general deformations.

Apologia

Two decades ago, when I coined the acronym RTICA (for Real-Time Interactive Computer Animation), it was wishful thinking that we should be able to do such a thing in the classrooms, in our student labs, in our offices, at home! By the end of that decade Silicon Graphics Iris workstations had made all this possible. By 1992, the "C" in the acronym could be expanded to CAVE, the recursively expanding acronym for a new technology[10] in immersive virtual environments. Along the way, what at first was mostly "a manner of speaking" has matured into a reusable vocabulary.

My acronym is intended to be pronounced, not spelled out. Try it! Make it sound like "articae": many artful things. The A also expands variously to Animation, Animator, Animatrix. And, "animation" itself has some welcome double

[*] An abridged version of this paper appeared in the proceedings of the MOSAIC 2000 Conference, Seattle, 2000. The real-time interactive computer animations mentioned herein, were presented at the conference *Matematica e Cultura: Arte, Teconologia, Immagini,* Bologna, 2000.

meanings: the process of bringing to life, and that which has been so enlivened. Similarly, the "animator", like the "editor", is both the human operator and the graphic tool used to produce the animation. There is, as yet, no such word as "animatrix". Perhaps there should be: for the tenth muse, who inspires us to write computer programs to make pictures come alive.

In this essay I shall explore computer animation from the viewpoint of a mathematician for whom drawing mathematical pictures came as a late vocation, and for whom teaching what he learned became the principal arbiter for many choices and decisions. I hope this attempt to formalize my "axioms" (if only to rationalize my ideosyncrasies) will at least amuse the reader.

First, I draw a sharp distinction between photorealistic rendering of computer images and a species of non-photorealistic rendering,[1]

Next, I choose a particular family of mathematical phenomena to illustrate, namely several kinds of *homotopies*. These can be classified according to the difficulty of rendering them as real-time interactive animations. Two related areas of mathematics are particularly hard to imagine and therefore good subjects for metarealistic rendering. They are 3-dimensional non-Euclidean geometry, and the visualization of processes extended in 4 isotropic geometrical dimensions.

Finally, I adopt a more polemical turn of speech which touches on the rightful place of mathematical illustration in the brief history of computer graphics. To maintain focus, I trace the evolution of my metarealistic enterprise, *illiView*, as an example of technology driven research.

Metarealism vs. Photorealism

There is no ambiguity in the meaning of photorealism in computer graphics. After $2\frac{1}{2}$ centuries of photography[2] and over a century of movies to fall back on, we have no difficulty in deciding when a computer animation really looks and acts "real". We have an even longer collective pictorial experience, formed in the eons during which artists developed (hardly photorealistic) renderings of every conceivable scene and action. By and large — and exceptions are worth pondering elsewhere — artists drew on the real world. Their pictures used real and familiar objects even when the intended meaning was supernatural, spiritual, or just fanciful. Even impressionism, which Webster[35] defines as "a type of realism the aim of which is to render the immediate sense impression of the artist apart from any element of inference or study of the detail," and subsequent abstract styles (by-and-large !),

[1] This new discipline [27, 19, 1, 23] treats computer simulations of all graphics media other that photography.

[2] In 1727 Prof. Johann Heinrich Schulze, discovered the photographic properties of silver-halide [2, 13ff]. Daguerre, 1839, received a life-long pension for his discovery from the French government.

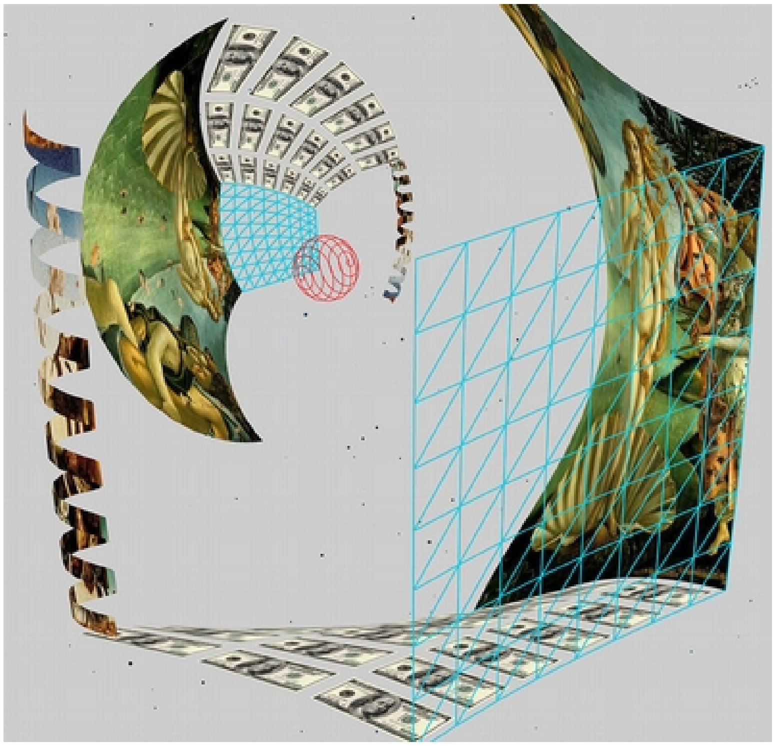

Figure 1: Snapshot into a gravitational lensing project by John Estabrook, Ulises Cervantes-Pimentel, Birgit Bluemer and George Francis. In this real-time interactive CAVE animation an invisible mass distorts our view of the world. A second image forms within the Einstein radius about the mass, which is inside the spiral ball. The ball is confined to bounce about the cubical stage. The rules of this CAVE-to-CAVE game, based on the Prisoner's Dilemma and implemented on the DuoDesk, amused the NCSA PACI-Partners and visitors to Supercomputing98[12].

mean to evoke an emotional or intellectual response which the viewer might have had on looking at the actual model for the picture.

Not so when we use computer graphics to render abstractions that *do not* have concomitant realities. Lately there has been a welcome resurgence of the fascination with the fourth *geometrical* dimension. In the latter half of the nineteenth century there was some popular interest in worlds of four or more isotropic spatial dimensions. Two generations of relativistic science and science fiction has made it difficult to persuade college freshmen and the general public alike that *the* fourth dimension need not invariably be time[4, 6, 21]. Hypercubes rotating in 4-space are found in screen-savers and are routinely assigned as machine problems in computer science courses. Since the advent of computer animation and virtual reality, non-Euclidean geometry has come into fashion[20, 14, 31, 36]. Java applets for drawing lines in the hyperbolic plane and on a spherical surface can be found on the web. Instructive excursions into special and general relativity, molecular biology, cosmology etc. all benefit from particularly *non-realistic* simulations which bend the laws of physics, use stick and balls to model membranes, enlarge planetary diameters a thousandfold, and travel through the cosmos faster than light. All this so you can see anything at all, and to animate it in real-time.

This enterprise deserves its own name, and I propose *metarealistic rendering*, fully aware of the hazard of coining jargon.[3] This is not a synonym for what is generally called *virtual reality*. The "virtual" in VR just means that we use novel projection (e.g. stereo) and input controls (e.g. head-tracking) to convincingly evoke the illusion of experiencing something "real", much as in a dream. The prefix *meta*, having the innocent meaning of "between, with, after, along side of" (Webster) also has less desirable connotations, such as "metaphysical", "metaphorical" and "metamorphosing". But recall that Aristotle's book on the essential nature of reality merely followed his book on its physical reality. Only later did *metaphysics* suggest something transcendent, philosophical, theological. Similarly "metaphor" has become bent out of shape in computing. No longer just the "poetic reuse of a word," it seems to mean the imposition of a familiar cognitive structure on an unfamiliar one. For example, the "desktop metaphor" trades on our familiarity with desktop furnishings, and then has us "mouse" pictures[4] about the computer screen, and to eject a floppy by depositing it in a trash can (a dubious metaphor!). Finally, in computing jargon, the transformation of the physical characteristic of an object, literally its *metamorphosis* (meta=trans, morphe=form), has lost the prefix to become just *morphing*. So, having demythologized some undesirable connotations, let us briefly consider whether the distinction between *photorealistic* and *metarealistic rendering* deserves a place in the philosophy of technology.

The photorealist has this significant advantage over a metarealistic colleague. The former can compare personal experience with its imitation (counterfeit, not facsimile) at every step along its development. Faithfulness to reality guides the programming strategy; it is an unambiguous criterion for choosing techniques. Thus for photorealism, Phong shading and anti-aliasing are invariably "better" than mere Gouraud shading and the "jaggies", because it makes the image look more like the photograph of the original, of the "real-thing". On the other hand, consider a rendering of what a four-dimensional artist would have to draw on a three-dimensional canvas in order to evoke the same illusion of an impossible fig-

[3] The literature is littered with multiple terms competing for the same fuzzy meaning, and derelict corpses of splendid terms that nobody uses.

[4] Representing files but blasphemously called "icons".

ure as the Penrose tribar popularized by Escher[26, 11]. What does such a 4-D object "really" look like? Should it be shaded at all? How much do we care about its jaggies? Actually, we do care. But only after the depiction is totally convincing in its raw-and-ready draft. Only then, resources permitting, might we engage a computer artist to help us to meet the competition for inclusion in the SIGGRAPH Electronic Theater.

Thus, we find that the metarealist must use other criteria, other standards of comparisons than the photorealist. Among these are fidelity to physical principles, mathematical honesty, even something like a Calvinist economy in programming the RTICA so that its scientific message does not get lost in a baroque programming style. After all, it is usually the student programmer of the RTICA who gains the greatest intellectual benefits from the exercise. Photorealistic perfection intimidates the student programmer and distracts the engaged observer. The latter is expected to think through the mathematical ideas expressed in the RTICA. Thus, the less the artifact looks "like something else" the better.

Animating Homotopies

Real-time interactive computer animation is particularly appropriate for illustrating mathematical objects called *homotopies*. These deal with topological changes that happen over time. As such, a homotopy is the natural mathematical abstraction of any temporal change-of-shape. Rather than proposing some grand scheme for animating all possible homotopies, we identify a few *elementary* and *typical* homotopies and develop a vocabulary for exploring their RTICAs.[5] However, we do propose to transfer the notion of homotopy into the vocabulary of computer graphics.

Turning a sphere inside out is the best known of such homotopies[32]. In the four decades since Smale proved its existence, and the two decades since Nelson Max first captured Morin's eversion in a computer animation, there have been many metarealistic renderings of this "Helen of Homotopies."

Let us review the common meanings of our vocabulary. But, why should we care what the non-specialist, ignorant of the jargon, might hear? Mostly, we do so to anticipate the conscious or subliminal associations our usage will arouse. It also permits us to appreciate the circumstances under which the common term entered the jargon, and the evolution of its meaning along with the technology.

A quick dictionary[28] search yields

[5] The students in my courses on geometrical graphics are my chief assistants in this enterprise. I present them a smorgasbord of sample homotopies, often worked on by previous students. They choose one, or come up with a new one, and build a semester project around it. This pedagogical aspect has a healthy influence on my understanding of what is elementary and typical.

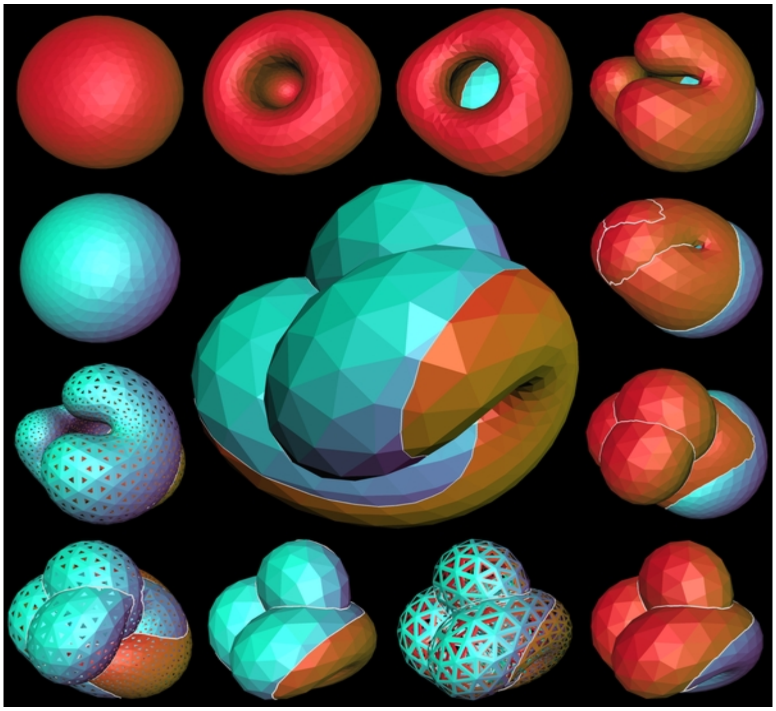

Figure 2: Thirteen topes (stages) of the symmetry-3 eversion in the "Optiverse" by John M. Sullivan, Stuart Levy and George Francis [15, 18, 33]. Clockwise (from top-left), this regular homotopy turns a bi-colored sphere inside out by passing through Boy's Surface (center). It is not an isotopy of the sphere in 3-space, nor can it be regarded as the shadow of an isotopy in 4-space. In metarealistic terms this famous eversion is an essential deformation.

Animate: Create the illusion of motions.

Motion: Change in position, change of body, gesture, gait.

Animated Cartoon: A motion picture consisting of a photographed series of drawings.

Motion Picture: Series of filmed (photographed) images viewed in sufficiently rapid succession to create the illusion of motion and continuity.

We consolidate these into our working definition of animation:

Animation: A sequence of pictures viewed rapidly enough to evoke an experience of continuous motion.

We have less success finding a common word for that which we animate. We borrow "homotopy" from topology rather than using the familiar terms: *transformation* and *metamorphosis*? Indeed, both say "change of shape", one in Latin, the other in Greek. But both also have undesirable connotations. In the common tongue, transformation has come to mean just about any kind of change. In mathematics it is synonymous with every function or mapping between two sets. In contrast, the second term has too restricted (entomological and mystical) meanings in the common language.[6] So we borrow a technical term from topology and popularize it.

All uses of the term *homotopy* in topology have this common denominator. Two continuous mappings, $f_0, f_1 : X \to Y$, from the same source space X into the target space Y are homotopic if there is a continuous mapping $h : X \times [0, 1] \to Y$ so that for each $x \in X$, $h(x, 0) = f_0$, $h(x, 1) = f_1(x)$. One writes $f_t(x)$ for $h(x, t)$ and refers to f_t as a *homotopy.*[7] The place that remains the same throughout the homotopy (from Greek: same+place) is X; the *topes* are its continuous images, $F_t = f_t(X)$, depicting what is being deformed in Y. Note that F_t is the deforming shape, and f_t is its parametrization.[8]

To capture a homotopy into a computer program, both source X^d and target Y^n must be subsets of real Cartesian spaces of dimensions d and n. When $d > n$ we tend to speak of *projecting* X to Y. When $d < n$ we think of *inserting* X into Y.[9] When $d = n$, and often as not f_0 is the identity map, then f_1 represents some "rearrangement" of space, and the homotopy f_t is a continuous *distortion* of space until the rearrangement is complete.[10]

Common examples of insertions ($d < n$) are discrete sets of points ($d = 0$) such as particle systems, curves ($d = 1$), and surfaces ($d = 2$) in space ($n = 3$). To observe the distortion of space ($d = n$) we insert lower dimensional marker objects and follow their fate. We observe projections from space ($d = 3$) to a surface ($n = 2$) either by what appears on the 2-dimensional screen itself, or in terms of pictures affixed to surfaces in space. Ray tracing, texture–, and environment mappings should come to mind.

In practice, we require the in-between maps, f_t, which parametrize the topes of the animation, to be of the same species as the initial and final maps. Thus, when

[6] The popular amputation, *morphing* comes closest, but already has a narrower technical meaning.

[7] Seifert and Threllfall[34] speak of a *homotopic deformation* from f_0 to f_1. The noun comes later.

[8] The neologism "tope" avoids also importing the topological term "image" into a field where "image" already has too many meanings.

[9] Also not a customary word, our use of "insertion" should remind the reader of *immersion, imbedding* but also singular surfaces in general position, called *Whitney maps*.

[10] Think of "warping", though this term is used also for bending an object already inserted in space.

$f_0,\ f_1 : X \to Y$ have some structure or regularity, it shall be preserved throughout the homotopy. For example, when X is homeomorphic[11] to each of its topes F_t, the deformation is called an *isotopy*. When $X \subset Y$ and f_t is the restriction to X of a homotopy g_t of Y into itself, we say that it is an *ambient homotopy*. Combining both, an *ambient isotopy* is what we mean by a *distortion* of X. Though it may be difficult to imagine, almost every isotopy we encounter extends to an ambient isotopy.

Classifying Homotopies

We propose a classification of homotopies which takes into consideration the manner in which their RTICA generates the stream of pictures. It is thus hardware and software dependent, but not determined by it. The brief elaboration after each abstract definition will make this clear.

motion: We are moving about a static scene. Or, in the solipsistic coordinate system of our personal reference frame, the entire world is being rigidly moved about in front of us. Either way, we need to understand clearly what *motion* entails.

The customary metaphor of changing *camera coordinates*, of making a *camera path*, is a vestige of the days that an RTICA was used mainly to produce a videotape for a passive audience. Virtual reality and immersive virtual environments are better served by the "observer" metaphor. The world, represented by a *display list* of polygons, X, is subjected to a succession of Euclidean isometries, $M(t)$, before being projected to the screen. In customary graphics programming libraries, such as OpenGL[25], $M(t)$ is a one-parameter family of 4-dimensional matrices applied to the vertices of X expressed in *homogeneous coordinates*. Thus, the abstract group of 3-dimensional Euclidean *isometries* (rotation followed by translation) is represented as a subgroup of the 4-dimensional linear group. Other 3D geometries, such as spherical and hyperbolic, can also be so represented and are treated in a similar way[14]. To the topologist, $M(t)$ is a path in the isometry group of a 3-dimensional geometry.[12]

articulation: While the scene is not totally static here, and objects move about independently, they retain their geometric shape.

[11] Essentially the same shape. Technically, each f_t is one-to-one and maps X onto F_t; its inverse is also continuous.

[12] In this context the letter "M" is a mnemonic for "matrix", "motion", as well as "MODEL_VIEW" in the vocabulary of OpenGL.

Such a clockwork of rigidly moving components (a marionette) would seem to differ qualitatively from a simple motion. In fact, it does so only quantitatively. One builds a *hierarchy*[13] which assigns to each component its proper motion and place in the world scene. That is, the i-th object has a display list X_i and an associated isometry, P_i, which places it into the coordinate system of the next higher object.

For Jim Blinn's *Blobby Man*[8], for example, each finger $X_i, i = 1, .., 5$ would become $P_i X_i$ when it is attached to the hand, X_6, which is attached to the arm X_7, the shoulder, the torso etc. Thus the thumb is rendered as $M_0 ... P_7 P_6 P_1 X_1$, where M_0 incorporates the observers motion. Of course, all the matrices are multiplied together into one before any vertex in the finger is sent down the pipeline. Thus the matrix for the arm is $M_7 = M_0 ... P_7$, and $M_6 = M_7 P_6$ is the matrix for the hand. The world motion, M_0, affects all objects and plays a special role as the inverse of the camera matrix. Suppose we place the observer (ourselves or the camera) somewhere, P_0, in the world. We could build an avatar, X_0, for ourselves so that we appear as $M_0 P_0 X_0$. If now we wish to see through our own eyes then X_0 is rendered without modification, whence $M_0 P_0 = I$ and $M_0 = P_0^{-1}$. If we wished to see through a camera held in the hand, for example, we would replace each M_i by $M_6^{-1} M_i$.

distortion: Here the motion $M(t)$ is not Euclidean, spherical or hyperbolic. We do not assume that the M belongs to any particular group of isometries. But the action MX remains linear (a matrix multiplication). In general, the motion, $M(x, t)$, depends also on the location where it is applied, as well as on time.

A trivial example of this is a change in scale, especially a non-isotropic one, as when a sphere turns into an ellipsoid. Any decoration on the sphere, or an entire articulated hierarchy of objects subordinate to the sphere, would be similarly distorted in the graphics pipeline.

The general case, where M also depends on location, is rarely implemented in a graphics library or in the hardware of a graphics system. We would have liked to use such an acceleration Lou Kauffman's ambient isotopy of 3-space which illustrates Dirac's String Trick. Here, independent rotations of concentric spheres, parametrized by their radius, have the effect of straightening a twice twisted ribbon[30, 17, 22]. Since it was a distortion of space itself, everything in it (such as very many twisted ribbons) followed along, and no ribbons passed through themselves. Unlike an articulation, a distortion of space guarantees that initially disjoint elements will not collide during the homotopy. As we had no hardware

[13] Also called a "scene graph" because the order of the matrix multiplications forms a directed tree.

acceleration for this kind of homotopy, it was implemented as a time-consuming per-vertex deformation and the videotape was ultimately made in the traditional frame-by-frame manner rather than capturing a real-time, interactive animation.

deformation: All changes from frame to frame involve changes in the vertices that make up X. Instead of time acting on M, it now acts on $X(t)$. That is, the coordinates of each vertex of an object are changed in time, $x \rightarrow h(x,t)$, before they are sent down the pipeline.

In a sense, all homotopies can be animated as deformations. It is the only way to do it in a graphics system that lacks a hierarchical geometric library with matrix algebra and hardware acceleration. However, when such shortcuts are available, it makes sense to disinguish this residual category from the previous three, which can be more easily programmed and more speedily displayed.

We often combine motions, $M(t)$, with deformations, $X(t)$. But for unfamiliar events it is best to alternate them, drawing $M(\mu(t))X(\xi(t))$, where $d\mu/dt = 0$ whenever $d\xi/dt \neq 0$ and vice-versa. Doing both simultaneously can be distracting unless one *time scale*, $\mu(t)$, is very much slower than the other one, $\xi(t)$.

The Geometry Pipeline

We shall now trace what befalls a display list as it passes down the graphics pipeline by working backward from what you see on the screen. Window systems are ubiquitous, so the absolute *screen coordinates* of a figure have no primary importance. To locate a point in a visible region of the screen (namely in a *window*), we specify fractions of the width and heights taken in the appropriate direction. The x-coordinate, or abcissa, always goes from left to right, but the ordinate is as often top-down as bottom-up. This way the numerical description (of the geometrical objects to be drawn on the screen) is invariant under moving and resizing the window. What to do if the aspect ratio of the resized window has changed, is a matter of choice. Technically, one distinguishes between the window and the *viewport* into which a view of the world is transformed, either orthographically or perspectively. The window then need not coincide with the viewport, especially if they have different aspect ratios. Also, the window may have more than one viewport, as for stereographic pairs.

It is, however, more practical to consider an abstract 3-dimensional space described in *world coordinates*, in which everything happens. On the screen, we see a certain portion of this world through the window. An important motion through this world is a steerable flight through a scene, X.[14] An incremental adjustment of the

[14] Here X represents all objects already placed into the world coordinates.

controller (mouse or CAVE wand) at moment t is imposed on M_0. We multiply on the left[15] by an "small" Euclidean isometry, $\tilde{M}_t$. Thus $M_0(t{+}dt)X = \tilde{M}_t M_0(t)X$. The $\tilde{M}$ is a rotation by a tiny angle followed by a translation by a tiny vector. An accumulation of such increments applied to an initial $M_0(0)$ (which is usually the identity) constitutes a flight-path or motion through the world.

Here is what happens to the homogeneous coordinates, (x, y, z, w), of a vertex after it leaves interactive control and passes into the *clipping* part of the geometry pipeline. Only vertices inside the *clipping cube*

$$-1 \leq \frac{x}{w}, \frac{y}{w}, \frac{z}{w} \leq 1$$

are admissible. Line segments and triangles that cross its boundary are cropped, $(\frac{x}{w}, \frac{y}{w})$ is scaled into the viewport, and $\frac{z}{w}$ is discretized into the depth buffer. The division of (x, y, z) by w is the reason that perspective projections can also be implemented by one last matrix multiplication in the 4-dimensional linear group. The very cleverly designed (but misnamed) *projection matrix* maps a *viewing frustum* homeomorphically into the clipping cube. The viewing frustum consists of a rectangular window and the portion of the rectangular cone from the origin through this window located between the window and a parallel clipping plane further back. One is reminded of Albrecht Dürer's etching of the artist sighting a reclining nude, and marking a transparent canvas where the sight lines pierce it.

We have seen, then, that a vertex in the display list of an object undergoes a sequence of multiplications by matrices representing the Euclidean motion of rotations (about the origin) followed by a translation.[16] All the power (and frequent confusion) of interpreting the meaning of multiplying a vector by a matrix, and, by extension, the interpretation of the associative law of matrix multiplication, comes to into play here. In addition to Birkhoff and MacLane's[7] classical *alibi* and *alias* interpretations, we have a *placement*. An object, described by a display list of coordinates in "its own natural" Cartesian framework, is *placed* into the reference frame of another object higher in the articulation hierarchy.

Let us illustrate the subtle differences in the case of a planar rotation:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x\cos\theta - y\sin\theta \\ y\cos\theta + x\sin\theta \end{bmatrix} = \begin{bmatrix} x\cos\theta - y\sin\theta \\ x\sin\theta + y\cos\theta \end{bmatrix}$$

[15] Unfortunately, OpenGL, only multiplies an existing matrix on the right since that is the natural place for implementing articulated hierarchies. Some matrix arithmetic is therefore inevitable.

[16] More precisely, these modeling matrices are in the 3-dimensional affine group, where the rotation is relaxed to be an arbitrary invertible linear transformation. This allows uniform, and non-uniform scaling.

Customary matrix muliplication “row vector dot column vector” reveals (u, v) to be the coordinates of the same point in a coordinate system which was rotated by an angle of $-\theta$. The point has an “alias”.

$$\textbf{alias} \quad \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \cos\theta \\ -\sin\theta \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} \\ \\ \begin{bmatrix} +\sin\theta \\ \cos\theta \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} \end{bmatrix}$$

Factoring the second expression, we have a trigonometric interpolation which move the point (x, y) towards its left-perpendicular $(-y, x)$ by an angle of θ along a circular arc. Thus the point has an “alibi” as to its whereabouts.

$$\textbf{alibi} \quad \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} \cos\theta + \begin{bmatrix} -y \\ x \end{bmatrix} \sin\theta$$

Rearranging and factoring a different way, shows that (u, v) is what you get if you follow the recipe “go x units along the abcissa followed by y units along the ordinate” when the meaning of abscissa and ordinate has changed by a rotation of the coordinate frame of orthogonal unit vectors along the axes. The point has a new “place” inside a different frame of reference.

$$\textbf{place} \quad \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} x + \begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix} y$$

To illustrate the flexibility of our language, let us read

$$P_1P_2P_3X = P_1(P_2P_3X) = (P_1P_2)(P_3X)$$

in two ways. In the first, the object X has been moved about it’s own space by the composite motion P_2P_3 and then placed into the reference frame (coordinate system) P_1. In the second, X has been placed by P_3 into a new reference frame, where it is then moved by the composite motion P_1P_2.

Permutations of the *alias-alibi-placement* attitudes generate many more verbalizations of varying practicality. However, it time to apply our new vocabulary to a perennial problem in mathematical visualization, that of seeing into the fourth dimension.

The Fourth Dimension.

It is curious how very much the popular mind is intrigued by the“Fourth Dimension” and it would be interesting to speculate why this might be so. But here is not the place for even the most cursory review of this issue. The reader should set aside an evening for browsing the web on the subject. Special attention to the work of Tom Banchoff [4] on the subject will be richly rewarded. What we can do here is to extend our vocabulary for metarealistic rendering a short distance in the direction of the Fourth Dimension.

First of all, we should distinguish between attempts to *visualize* phenomena extended in 4 isotropic dimensions, and our recognizing four and higher dimensional reality by its *special effects* in 3D. Borrowing from media jargon, we might call the former *4D-viz* and the latter *4D-fx*. It is appropriate here to remind the reader that, when all is said and done, we can see only curves and surfaces. We draw inferences about space and bodies by the way curves and surfaces arrange themselves under lighting, occlusion, motion parallax, binocular vision etc. We can learn to draw similar inferences about higher dimensions from the way we furnish ordinary space. We do this in basically four different ways.

decorations: Curves, surfaces and *rooms*,[17] whose points are endowed with more attributes than fit into 3-degrees of freedom, are equipped with visual artefacts, *glyphs*, which express the attributes.

The whole of scientific visualization might be subsumed under this first rubric. For example, we might paint temperatures on heated objects in vivid color. We distinguish two *hyper-attributes*, say pressure and temperature, by color and texture. Visualizing surfaces in 4-space by painting the fourth dimension on a 3D slice or shadow of it is less successful. Graphs of complex functions are particularly sensitive to such graphical misadventures. The reason is simple. While we are not apt to rotate pressure into temperature, we do want to rotate an object in 4D every which way.

A common way of indicating many hyper-attributes is to place *glyphs* at a sample of points. We remember the little arrows representing velocity vector field from our calculus courses. Glyphs generalize this idea in the form of small, solid shapes, which have an obvious direction. Limited ranges of other attributes are mapped to physical features of the glyph, for example, its size, color, shape, texture, and other details.

[17] The correct technical term is “volumes”, as in “volumetric rendering”. But non-specialists might be misled into thinking of books.

charting: Three dimensional subspaces of R^4 are faithfully mapped to a flat, 3-dimensional canvas.[18] Think of a Mercator or a polar projection of a sphere to a page in an atlas, only one dimension higher. Topologists prefer to call this a *chart* rather than a "map" to emphasize that the the mapping is one-to-one.

Among the most satisfactory examples of this genre are the elegant realizations of non-Euclidean geometries of 3-space in terms of projections from 3-manifolds in R^4. For a positively curved geometry of space we use a conformal projection[19] of the 3-sphere from its pole to flat 3-space. Straight lines become circles, and geometrically flat Clifford tori become the sensuously "oblique" Cyclides of Dupin[3]. Central projection of a 3-dimensional paraboloid to the 3-ball gives it a hyperbolic geometry, exquisitely rendered in the the final minutes of the Geometry Center's classic video "Not Knot"[20].

Our real-time interactive CAVE animation, *A Post-Euclidean Walkabout*[14], has four such charts. In the first act we are still in Euclidean space and merely fly around and through a morphing shape of a sea shell. Act two starts with a suitably triangulated and painted hyperbolic octagon in the Poincaré model of the hyperbolic plane. It lifts off the plane and wraps itself into a double torus. Though topologically distorted, the triangulation remains as a testimony to the conformal structure on the Riemann surface. Next, we enter hyperbolic 3-space and fly through its tesselation by right-angled dodecahedra. In the final act we visit the dodecahedral subdivision of the 3-sphere, the 120-cell.

The efficiency of these *4D-fx* is the result of the intrinsically 4-dimensional nature of the Silicon Graphics geometry pipeline, which powers the CAVE. All three isometry groups, Euclidean, hyperbolic and spherical, have a representation in the 4D general linear group which acts on the *projective models* of the three geometries of space. This *Klein Model* of hyperbolic 3-space is conformal[20] at the center of the unit ball. To give the CAVE visitors the correct illusion of flying while keeping them fixed to the origin, we inflict a 1-parameter family of isometries on the dodecahedral tesselation.

shadows: Generalize to one dimension higher, the familiar perspective, axonometric, and orthographic projections from the fine and graphic arts.

[18] What kind of subspaces, e.g. manifolds, and how faithful, e.g. conformal, are subjects of a finer classification than we have in mind here.

[19] The commonly used term, "stereographic" for this projection is often misunderstood to mean "binocular." The geographer prefers "polar'; the geometer prefers "conformal."

[20] Conformal mean "angle-preserving" and implies that circles map to circles. To be conformal just at one point means that visual angles from that point are "true". In particular, right angles *look* correct.

One immediate problem with this is the fact that we look at a perspective drawing of a 3-D scene from the *outside.* Add one dimension and we now experience the "picture" by being in it.

However, even with 3D perspective our perception depends on a 2D picture having lots of curves in it: edges, profiles, contours, 1-D textures. Similarly, we can learn to recognize 3D projections of surfaces extended in 4D by the way their shadows on our 3D canvas deform even as the object is rigidly rotating in 4D. This school of thought originated with Tom Banchoff's classic computer animations[5].

Somewhat more difficult to follow is a homotopy of a surface in 4D, such as the unraveling of an unknot. An embedding of a 2-sphere in 4D is called a *4D-knot.* It is a trivial 4D knot, an *unknot*, precisely if an isotopy moves it into a 3D-flat, and there, it looks like a sphere. The lower dimensional analogue of such a thing is an unknotted tangle, say your garden hose with end screwed to end, magically untangling itself, without "cutting and pasting," so as to lie flat on the lawn as a perfect circle. This was Dennis Roseman's subject in our CAVE tryptich, *Laterna matheMagica*[15]. There, the "shadow" of the unknot was decorated with a color to indicate the intersection with a 3D-flat slicing through the surface.

slices: If we interpret one axis in R^4 as time, then the succession of orthogonal 3-spaces organize themselves into one spatio-temporal experience.[21] Conversely, any process in space-time can be regarded as a monolithic 4-dimensional entity.

If we slice a plane through a surface in 3D we see curves wriggling in the plane. The curves needn't be amorphous. Floor plans of buildings are also 2D slices of surfaces in 3D, which change discontinuously as we move through the ceiling. Contour lines of maps depicting mountain tops, passes and valleys are familiar examples of slices. If we pass a 3D slice through a surface in 4D we get curves in space which undergo interesting recombinations. Of course, we might start with the recombinations and organize them into a coherent surface in 4-space.

However a better analogy is obtained by considering a maze. A 2D-maze is what we solve on a piece of paper. Even when we are in the maze, as in a park or on a floor of an unfamiliar building, we can find our way out as soon as we see a floor-plan. A 3D-maze is just a stack of floor-plans, a building for example. So a 4D-maze is a stack of buildings, and finding your way around and out of such a maze is best left to Virtual Reality [29].

[21] A movie, and earthquake, a dream.

Technology-driven Research

From the beginning of my topological apprenticeship I was repelled by the customary highly formal, mostly verbal, and poorly illustrated descriptions of homotopies. In time, I also learned that the authors of these descriptions secretly drew pictures for themselves to clarify their thoughts, to simplify, and even to discover new theorems. Only their modest skills and lack of training in the graphic arts kept them from making their expositions easier to follow. Or so I thought! In fact, it was the custom in the heyday of Bourbaki to disparage pictures in mathematics. It was rationalized by claims that pictures lead to erroneous inferences, that they are expensive to print, that they take up space better devoted to elaborate notation.

Determined to illustrate homotopies more effectively, and to teach others to do likewise, I needed to find new graphic media. For my purposes, the medium had to be completely under my control. Inspired by Hilbert Cohn-Vossen[24], I first learned to draw with pencil, chalk and ink pen, on paper napkins, blackboards and drafting vellum[16]. But some homotopies simply would not fit into a set of one, two, or however many discrete pictures; they required animation [21].

Traditional cell-animation (*á la* Disney) was out of the question. This medium requires substantial group effort. Cell-animation teams are highly differentiated and specialized. Having little talent for management and no appreciable resources, I had to look for other ways to animate my homotopies. Computer animation looked attractive, in particular the promise of real-time interactive computer animation. Once early on, I was too late to make a videotape in time for a public presentation. The convention then (and still today) was to capture frames into animation buffers, transfer them over the the net, and lay them onto the tape. My NCSA colleague, Ray Idaszak, suggested that I build flexible steering gadgetry into the program and wire the *Personal Iris* workstations directly to the beta-cam recorder. I rehearsed my talk as I steered the RTICA, and the tape got done in time for the conference.

Computer animation, which meets current standards of quality, continues to be produced in the traditional stop-action manner, because computers cannot produce frames of the required quality in the twenty-fourth of a second needed for the animation to pass the threshold of fluidity. As computer speed catches up to yesterday's standards, tomorrow's techniques of photorealism streak out of range of real-time interaction. Typically, an RTICA is used for a computer animation only initially: to explore and refine, perhaps to choreograph the action and plan the camera paths. At this stage of the production, rough, sketch-like rendering suffices. Later, often at tremendous cost in resources (time, cycles and storage) the planned frames are rendered and stored at leisure until they can be assembled into a videotape or digital movie. By and large, this process was not for me.

Some of my projects did require division of labor and specialization to be "successful," to result in SIGGRAPH quality videotapes produced in the traditional manner[9, 18, 30, 33]. These projects required far too much emotional effort and their completion generally left me drained and unproductive for months! Quantitatively speaking, it was more cost-effective to concentrate on the construction of the RTICA than on the production of competitive videotapes.

The decision to optimise the efficiency of an RTICA, and to find homotopies most suitable for the technology at hand, eventually paid off handsomely. Real-time interactive animation was exactly what was needed for immersive virtual environments, such as the CAVE[12, 13, 14, 15, 31]. This new medium presented its own geometrical and pedagogical challenges. Adapting CAVE programming to the classroom involves careful design of examples and prototypes like the *illiShell* (1994) and *illiSkeleton(1998).*

My *illiView* project[22] acquired new watchwords like "transfer of technology" and "rapid prototyping." Each new technological opportunity and demand generated criteria for choosing the homotopy to be taken on next. Not infrequently, a student in a geometrical graphics course would express the desire to experiment with a particular set of graphical features unfamiliar to me. In keeping with the mathematical nature of the courses, this also required finding a suitable homotopy to be illustrated with the new technology.

This, then, is the context in which our present article takes its origins. Our modest expansion of the technical vocabulary for treating real-time interactive computer animation with some precision, at least fits our experience. We hope that it will prove equally useful to other graphicist who practice metarealistic rendering, whether they approve of this name for their work or not.

References

[1] *Proceedings of the First International Symposium on Non-Photorealistic Animation and Rendering*. ACM-SIGGRAPH, Eurographics, Annecy, France, June 5-7,2000.

[2] Hans-Dieter Abring. *Von Daguerre Bis Heute, Band I.* Buchverlag und Foto-Museum, 1990.

[3] Thomas Banchoff. The spherical two piece property and tight surfaces in spheres. *Journal of DifferentialGeometry*, 4(3):193–205, 1970.

[22] Named in admiration of the Geometry Center's geometrical viewing package, *Minneview*, its highly successful successor, *Geomview*, and cousin *Meshview* by Andy Hanson, *illiView* differs from these in many respects, as described in [21].

[4] Thomas Banchoff. *Beyond the Third Dimension*. W. H. Freeman&Co., Scientific American Library, New York, 1990.

[5] Thomas Banchoff and Charles Strauss. *Complex Function Graphs, Dupin Cyclides, Gauss Maps and the Veronese Surface*. Computer Geometry Films: Brown University, Providence, RI, 1977.

[6] David Banks. Interactive display and manipulation of two dimensional surfaces in four-dimensional space. In *Symposium on Interactive 3D Graphics*, pages 197–207. ACM, 1992.

[7] Garrett Birkhoff and Saunders MacLane. *A Survey of Modern Algebra*. MacMillan, 1953.

[8] Jim Blinn. *Jim Blinn's Corner: A Trip Down the Graphics Pipeline*. Morgan Kaufmann, 1996.

[9] Donna Cox, George Francis, and Ray Idaszak. The Etruscan Venus. In Tom DeFanti and Maxine Brown, editors, *Video Review Number 49*. ACM-SIGGRAPH, 1989.

[10] Carolina Cruz-Neira, Daniel J. Sandin, Thomas A. DeFanti, R. V. Kenyon, and John C. Hart. The CAVE: Audio-visual experience automatic virtual environment. *Communications ACM*, 35(6):65–72, 1992.

[11] Michael Daniels and Michael Pelsmajer. *Illusion*. National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign, 2000. An *illiView* real-time interactive CAVE animation.

[12] John Estabrook, Ulises Cervantes-Pimentel, Birgit Bluemer, and George Francis. Partnerball: a mathematical experiment in simulating gravitational lenses in tele-immersive virtual reality. *Supercomputing98*, 1998.

[13] Alex Francis, Umesh Thakkar, Kevin Vlack, and George Francis. CAVE Gladiator: Experiments on the relative value of head-tracking and stere-opsis in CAVE immersive virtual environments. *CAVERNUS: The CAVE Research Network Users Society*, 1998.

[14] George Francis, Chris Hartman, Joanna Mason, Ulrike Axen, and Paul McCreary. Post-Euclian Walkabout. In *VROOM – the Virtual Reality Room*. ACM-SIGGRAPH, Orlando, 1994.

[15] George Francis, John M. Sullivan, Ken Brakke, Rob Kusner, Dennis Roseman, Alex Bourd, Chris Hartman, Glenn Chappell, and Jason Rubenstein.

LATERNA matheMAGICA. In Holly Korab and Maxine D. Brown, editors, *Virtual Environments and Distributed Computing at SC'95: GII Testbed and HPC Challenge Applications on the I-WAY*. ACM/IEEE Supercomputing'95, 1995.

[16] George K. Francis. *A Topological Picturebook*. Springer, 1987.

[17] George K. Francis and Louis H. Kauffman. Air on the Dirac Strings. In William Abikoff, Joan Birman, and Katherine Kuiken, editors, *The Mathematical Legacy of Wilhelm Magnus*, volume 169, pages 261–276. Amer. Math. Soc., Providence, RI, 1994.

[18] George K. Francis, John M. Sullivan, Stuart S. Levy, Camille Goudeseune, and Chris Hartman. The Optiverse. In *SIGGRAPH Video Review*, volume 125 of *Electronic Theater*. ACM-SIGGRAPH, 1998. 3 min narrated and scored videotape.

[19] Stuart Green, David Salesin, Simon Schofield, Aaron Hertzmann, Peter Litwinowicz, Amy Gooch, Cassidy Curtis, and Bruce Gooch. *Non-Photorealistic Rendering, Course Notes*. ACM-SIGGRAPH, New Orleans, 1999.

[20] Charlie Gunn and Delle Maxwell. *Not Knot*. A K Peters, Wellesley, MA, 1991. video (17 min) produced by the Geometry Center, U. Minnesota.

[21] Andrew Hanson, Tamara Munzner, and George Francis. Interactive methods for visualizable geometry. *IEEE Computer*, 27(4):73–83, July 1994.

[22] John C. Hart, George K. Francis, and Louis H. Kauffman. Visualizing quaternion rotation. *ACM Transactions on Graphics*, 13(3):256–276, 1994.

[23] Aaron Hertzmann and Denis Zorin. Illustrating smooth surfaces. *SIGGRAPH2000 Conference Proceedings*, pages 517–526, 2000.

[24] David Hilbert and S. Cohn-Vossen. *Geometry and the Imagination*. Chelsea, 1952.

[25] Mark J. Kilgard. *OpenGL programming for the X Window System*. Addison-Wesley, 1996.

[26] Scott E. Kim. An impossible four-dimensional illusion. In David W. Brisson, editor, *Hypergraphics: Visualizing Complex Relationships in Art, Science and Technology*, pages 187–239. Westview Press, Boulder, 1978.

[27] Lee Markosian, Michael Kowalski, Samuel Trychin, Lubomir Bourdev, Daniel Goldstein, and John F. Hughes. Real-time nonphotorealistic rendering. *Proceedings of SIGGRAPH 97*, pages 415–420, August 1997.

[28] William Morris, editor. *The American Heritage Dictionary of the English Language*. Houghton Mifflin, 1969.

[29] Michael Pelsmajer. *4D-Maze*. National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign, 1999. An *illiView* real-time interactive CAVE animation.

[30] Daniel Sandin, George Francis, Louis Kauffman, Chris Hartman, and Glenn Chappell. Air on the Dirac Strings. In Tom DeFanti and Maxine Brown, editors, *Electronic Theater*. ACM-SIGGRAPH, 1993.

[31] Daniel Sandin, Louis Kauffman, George Francis, Joanna Mason, and Milana Huang. Getting Physical in Four Dimensions. In *VROOM – the Virtual Reality Room*. ACM-SIGGRAPH, Orlando, 1994.

[32] John M. Sullivan. The Optiverse and other sphere eversions. In *ISAMA 99*, pages 491–497. The International Society of The Arts, Mathematics and Architecture, Univ. of the Basque Country, 1999. Proceedings of the June 1999 conferece in San Sebastián.

[33] John M. Sullivan, George Francis Stuart S. Levy, Camille Goudeseune, and Chris Hartman. The Optiverse. In Hans-Christian Hege and Konrad Polthier, editors, *VideoMath Review Cassette*, VideoMath Festival at ICM'98. Springer-Verlag, Berlin, 1998. 6.5 min narrated and scored videotape.

[34] H. Seifert und W. Threlfall. *Lehrbuch der Topologie*. Chelsea, 1947.

[35] *Webster's Collegiate Dictionary*. Thomas Allen, Ltd., 5th edition edition, 1936.

[36] Jeffrey R. Weeks. *The Shape of Space*. Dekker, 1985.

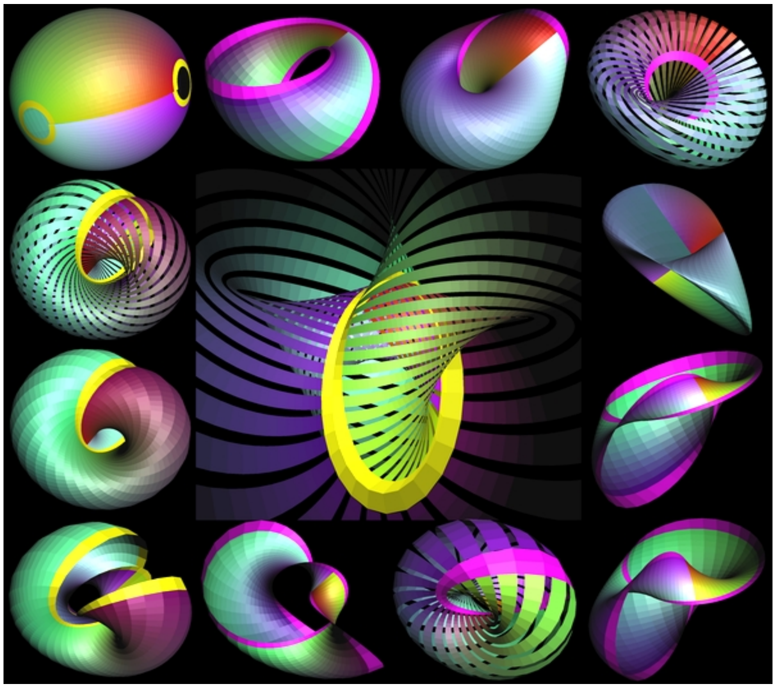

Figure 3: Thirteen topes (stages) of the “illiSnail” animation in the “Post-Euclidean Walkabout” by Chris Hartman, Joanna Mason, Ulrike Axen, Paul McCreary, and George Francis[14]. Clockwise (from top-left), we chart Blaine Lawson’s ruled, minimal surfaces in the 3-sphere by projecting them conformally to 3space, passing through a meridian 2-sphere (1), a half-twist Möbius band (2) with a circular border (3) closing up to Steiner’s cross-cap (4) and Roman surface (5). A once-twisted ribbon (6,7) closes up to the Clifford torus, seen from the outside (8) and inside (center). A 3-half twisted ribbon (9) closes up (10) to half of Lawson’s minimal Kleinbottle. This surface is also the mapping cylinger of $w^2 = z^3$, and Ulrich Brehm’s trefoil knot-box. This real-time interactive CAVE animation has conformal projections (charts), shadows and slices of surfaces embedded, rotating, and isotopically deforming in 4-space.