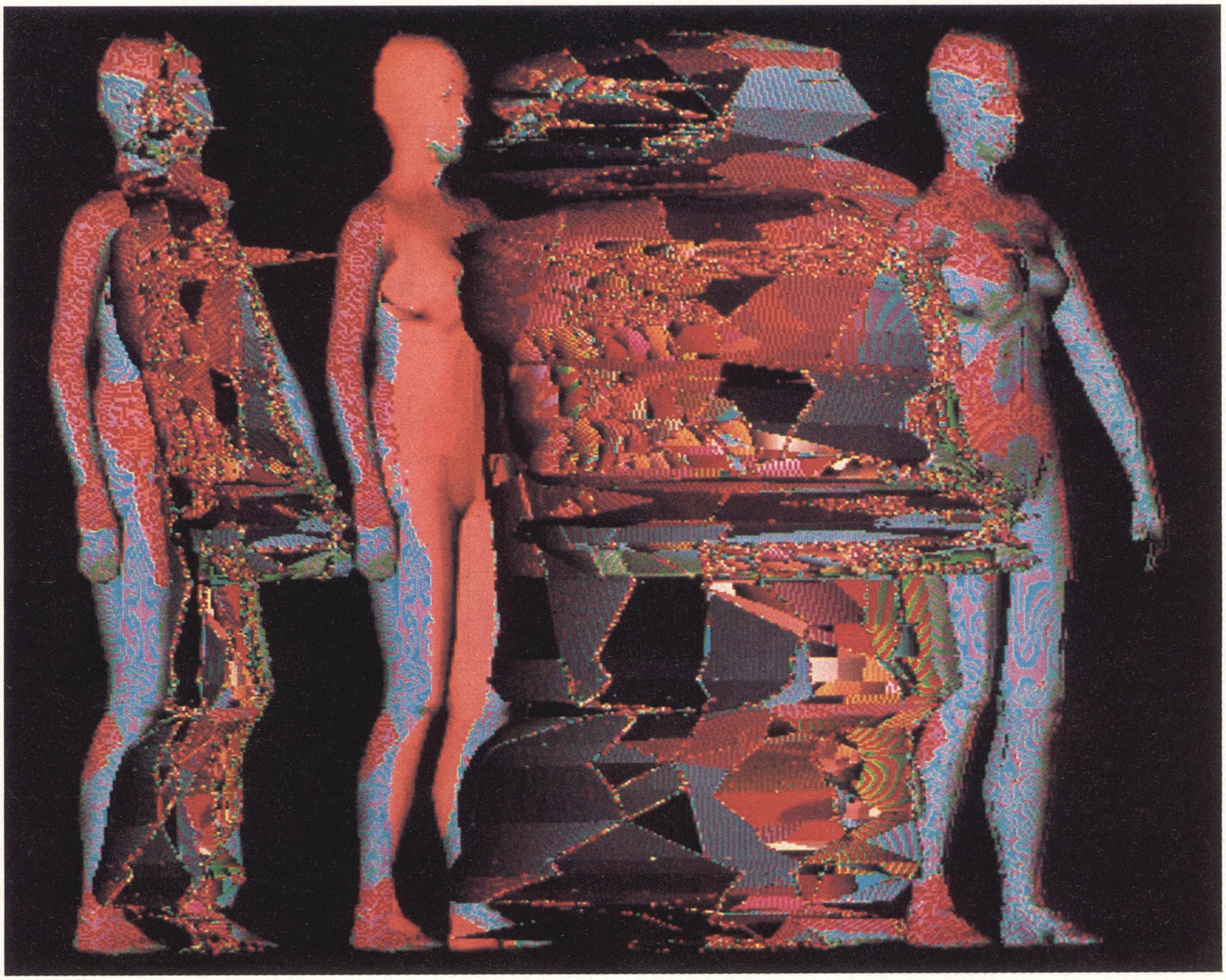
*Western society tends to separate individuals into the archetypal roles of artist and scientist. With computer graphics tools, artists can integrate these roles and produce unique artwork.*



# Computer/Art—Depolarization and Unification

Duane M. Palyka

New York Institute of Technology

No one can deny the separation of art and science in our western society. This duality is also reflected within the individual (a microcosm of the macrocosm) as a right-brain/left-brain polarization. This social separation, emphasizing the scientific side as the dominant side, also separates the individual from his psyche—his imagination, his intuition, human factors, and humanitarian concerns relegated by left-brain thinkers to the realm of the "irrational."

According to Jose Arguelles in *The Transformative Vision,* "...the problem of specialization did not become globally critical until the development of a mechanistic technology in Europe during the late Iron Age. Accompanying this development was the split of cerebral functions, leading to the creation of the two archetypal roles of artist and scientist...Insofar as each is rooted in only one side of the human brain—the scientist in the left, the artist in the right—each is an incomplete being."[1]

As if separation of man into the "technical" man and the "psyche" man was not enough, the former became attached to the money/power structure, and the latter has had to negate his psychic basis to survive. "Thus, as the culture develops along profit-making lines that stress novelty, gadgetry, and innovation for the sake of innovation, the artist consciously or unconsciously is swept along."[2]

Where does this leave the computer artist? Which side of his being forms his work? Is he an artist begging for tools from a technician, or is he a technician making left-brain images? Computer art is a showcase for this duality, yet it also offers a rare opportunity to resolve it. My hope is that the individual can transcend society's definition of scientist and artist and function more freely as a human being. Blurring the social roles should make it psychologically easier for the individual to move between the left- and right-brain hemispheres.

A passive solution to this problem is to dissolve the mechanism that blocks the two hemispheres from functioning together. The ego is the mechanism that prevents the natural flow between hemispheres. We must convert the inner parent/child or master/slave relationship to one of cooperation.

On one hand, the "rational" left brain cannot do everything itself. And on the other hand, when the artist swings the other way, there is usually equivalent resistance to getting involved with technical tasks such as programming. Perhaps future generations of artists who grow up programming computers and making computer art will have less rigid brain mappings than we have. Perhaps too, less rigid technical tools will replace programming and make the flow between hemispheres easier. In any case, we should really be interested in developing the whole person, not an incomplete one who functions on only one side.

Currently, computer graphics/art exemplifies this duality and dominance even further by tying the creation of images to commercial production, bringing with it the superficial gadgetry and the "innovation for the sake of innovation" mentioned above. Under commercial emphasis, computer graphics tool development dominates creative efforts and brings along with it competition and secrecy. The computers in use now are expensive, and the artist must create on the scientist's terms just to get access to the equipment. Furthermore, if the artist is not a

An earlier version of this article appeared in *Proc. Symp. on Small Computers in the Arts,* Philadelphia, Penn., 1984, pp. 7-16.



0272-1716/85/0700-0047$01.00 © 1985 IEEE

programmer, he must depend on the scientist to provide the software tools for him to realize his aesthetic needs—tools which the scientist thinks the artist should have based on what he has seen as artistic tools in the past.

## Paradigm

One of the paradigms of computer graphics/art is the desire and competition to design tools for generating the most "realistic" images. This attitude revolves around the scientist's myopic view of art as seen only from the technical side. An artist's work is a balance of psyche and *techne,* but it is the latter aspect that the scientist relates to and identifies with art.

"Both art and science rely on technique; the difference between them is in the degree of contact with psyche."[3] From this viewpoint the "scientist-become-artist" deals with spaces that have mathematically correct perspective with detailed and rigid object definitions. He becomes concerned with issues of how to make clouds, trees, and mountains look "accurate," without considering that this could be a subjective determination. It appears to him that the "jaggies," the staircasing effect inherent in digital images, serve no aesthetic purpose but interfere with the technical (and, hence, commercial) conception of visual reality. Thus, he forces art to be pulled further from the psyche and into the linear time/space left-brain domain. To be fair, it is a left-brain-based society that demands these tools, and the scientist is fulfilling its needs for competitive commercial and military applications. It's frightening to think of videogames as being a training ground to kill people, but military simulators are computer graphics products not unlike videogames.

It's easy enough to point out deficiencies, but it is another matter to show positive alternatives. My own work has been concerned with resolving these issues, and I would like to share with you my personal exploration toward right/left brain unification and art/psyche involvement.

## Working concepts

In all my computer artwork to date I have done my own programming to merge my aesthetics into the software. In the main body of artworks I wish to discuss, I accept standard concepts of object-oriented computer graphics as my basis—for example, solids of revolution, polygon tilers, B-spline patches, and polygonal databases. Object-oriented computer graphics involve tools designed to build objects, not spaces. Spaces incorporating objects made with these left-brain tools usually have the feeling that the elements fit together arbitrarily. This feeling is more obvious in spaces constructed by engineers than those made by artists, since artists are sometimes able to transcend the design limitations of the engineer's tools. In any case, a spatially oriented computer graphics tool set would promote better holistic thinking—a right-brain function in the right-brain realm of visual spaces. Later, I shall discuss how spatial tool design relates to my work.

Using these programming concepts, I proceeded to design and build instances of these tools with particular orientation toward my artistic sensibilities. For example, I accept the notion that programming accidents and complex interaction sometimes contribute to interesting visual effects that may relate to my aesthetic sense. I try to be open to creative associations. I then use these tools to build imagery based on poetic mathematical concepts and my own psychic sense, which I describe as surrealistic in its patterns of free association. Allowing mathematical/programming ideas to merge with visual ones, I attempt to find a unique form of personal right/left hemisphere expression—an expression controlled and sometimes hampered by my initial promise of starting with traditional left-brained computer graphics tools. The final work still reflects a stiff, object-oriented space with mathematically calculated light sources, shading, and perspective. Frankly, the design of the original computer graphics tools is strong, and it is hard to resist using them as a starting point. It is much harder for me to design personal tools of equivalent strength completely on my own.

***To be fair, it is a left-brain-based society that demands these tools, and the scientist is fulfilling its needs for competitive applications.***

In describing my work, I concentrate on artistic aesthetic ideas and artistic technical ideas that are unique to my work, and I minimize the description of general computer graphics tools, which can be found elsewhere. Consequently, if the reader is not familiar with general computer graphics tools, I refer him to Andrew Glassner's recently published book *Computer Graphics User's Guide* and Judy Sachter's Master of Arts thesis for Ohio State University entitled "The Basic Concepts of Three-Dimensional Computer Graphics for Artists." Both are oriented toward the artist as user. Sachter's thesis hopefully will be coming out in book form in the near future.

## Related spaces

In thinking about three-dimensional computer graphics, one must deal with two related spaces: One is the two-dimensional visual space in which the work is seen, and the other is the multidimensional mathematical space that contains the numeric information and algorithmic schemes to generate the work.[2] The visual space, being projected from the mathematical space, can be thought of as a certain way of looking at the latter space. The artist uses a rendering technique to take a "snapshot" from one point of view of a more complete and complicated space that really has many points of view. This relationship of spaces is similar to the esoteric idea that our own view of experiential reality is being projected for us by a more fundamental and complete "greater reality"; and, as our consciousness grows, we are likely to experience more of the greater reality and develop greater understanding of how the lesser realities are projected from it and by it. Hence, in greater realization, we are able to free ourselves from individual life patterns.

Likewise, in computer graphics, the more we understand the mathematical reality behind the images, the more we are able to understand the projection of the individual images and see them in a different light. Our added awareness frees us from old patterns of thinking about images. Similarly, once we see how software systems logically work, we can free ourselves from being "stuck" to one particular software system. Instead of jumping around the branches of a tree trying to familiarize ourselves with each new branch, we should ground ourselves in the roots and see where the branches originate.

The artwork that I'm presenting has aesthetic considerations that include more than "meets the eye." Although one can appreciate it on that level alone, if one reaches into the greater reality of the work, one gets more from it. This is like understanding the formal logical organization of Rembrandt's "Polish Rider" (The Frick Collection, N.Y., 1655) or the symbolic meaning within Robert Campin's masterpiece "The Annunciation" (The Metropolitan Museum of Art, N.Y., about 1425-28). Except for its deeper mathematical basis, this method of relating an artwork to its greater reality is merely an extension of what artists, art historians, and art lovers normally do with works of art.

## Carrots and eggs

As in nature, my three-dimensional computer artwork has its evolutionary roots based on the egg. Various parts of the work "Space Carrots" (see Figure 1) are "mathematical eggs" formed using the concept of "solid of revolution." Although the meaning of this term may be

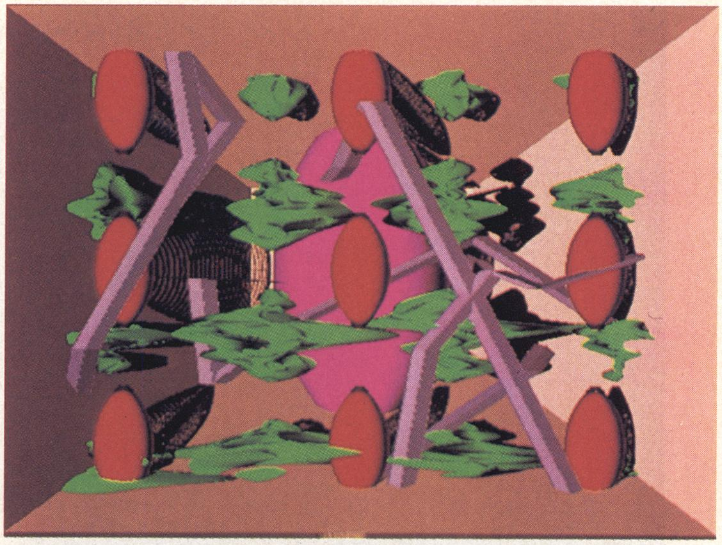


Figure 1. "Space Carrots."

inferred from the following text, please refer to Sachter for a better description.[2] A vector that forms the radius of a circle in the $x,z$ mathematical plane grows larger as it moves downward in the $y$ direction toward the center of the form and then grows gradually smaller again toward the bottom to form an egg. The varying radius, using the vertical egglike curve to determine its length, changes as it lofts circles in the $x,z$ plane. In mathematical space the circles themselves are really nothing but $(x,y,z)$ point calculations that are deposited equidistantly from each other at the edge of the radius, which spins in a circular path. Looking at it from the top down, the viewer sees that each set of equidistant points vaguely resembles a circle. To give our egg visual substance, we use our calculated points to attach B-spline patches to our etheric form much as a roofer would use nails to attach shingles to a curved house.[3]

Even though the use of patches may seem to be overkill to the technically sophisticated reader for such simple "egg-carrots," they are indispensable for generating the green, amorphous forms in "Space Carrots." These forms are generated simply by adding some restrained random numbers to the radius lengths that form the egg.

Although I could generate a wide variety of interesting forms by varying my egg radii randomly, I decided to utilize solids of revolution for more controlled sculpting effects.[2] However, instead of using multiple curves to define the object, I used a look-up table technique similar to that used in texture mapping and bump mapping.[3] In a frame buffer other than the rendering buffer, I drew an unusual picture of my face using conventional paint system techniques (see Figure 2). The drawing's uniqueness comes from the fact that during its execution I visualized the drawing as an unwrapped soup-can label to be wrapped around an egg. I used my imagination to

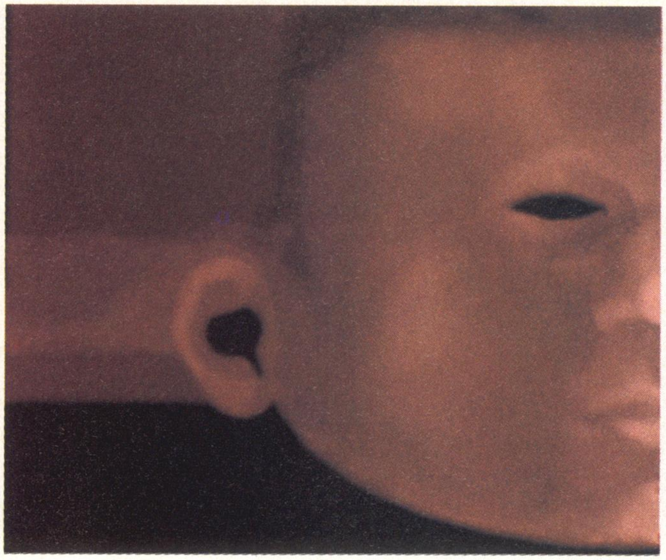


**Figure 2. "Face."**

visualize my head as an egg with shades of light and dark representing the spatial differences between my head and the egg. In the "null" case, if I had presented to my rendering algorithm a totally gray drawing in place of "Face," it would have sculpted a faceless Humpty Dumpty.

In generating the egg, the program would add a number from the frame buffer that contained the face drawing each time the radius required a new length, instead of adding some random number to it. If the number showed white in the "face drawing" buffer, the radius would move out the farthest, and if the number showed black, the radius would shrink toward the center of the egg.[3] As you can see by the resultant "Egghead" picture (see Figure 3), the experiment was not as successful as one would have hoped. My final result resembles what I look like in someone else's reality projection but not what I look like in my own!

Actually, the problems I encountered with this experiment stem from two sources: my preconditioning as an artist in seeing light and shadows in figure drawing and the grossness of the tools I'd developed to do the task. That experience taught me not to get caught up in the above-mentioned "realistic rendering" paradigm!

## Incorporating the Z-buffer

Notice that I incorporated the Z-buffer into the lower half of the picture as part of the image. A Z-buffer is an extra frame buffer or two used to hold the picture's depth information to allow a programming mechanism to resolve depth overlap and intersection on a pixel-by-pixel basis. This mechanism determines how much of the image gets projected from the multidimensional greater reality to the two-dimensional frame buffer space.[3] Actually, exotic uses of the Z-buffer add another chance component to the scheme of things. Since I love to incorporate chance factors into my artwork, I use the Z-buffer extensively, either visibly or behind the scenes throughout my work.

"Warm-Cool One" (Figure 4) is just such an example of one of my "behind-the-scenes" uses of a Z-buffer. This picture was created using both the finished two-dimensional frame-buffer image of "Space Carrots" and its saved Z-buffer. On a pixel-by-pixel basis, I used the following procedure to modify the image. First, I separated the pixel's color component from its intensity component. This was fairly easy to do since I defined the intensity to be the lower five bits in an eight-bit pixel and the color to be the upper three bits. Here, I had eight different colors to work with and 32 different intensities for each color. Then I picked out the pixel's associated Z-buffer value from the same pixel position in the saved Z-buffer. Using this value as an index into a red/green peppermint scheme that I had mapped onto "Z depth," I changed the pixel's color value accordingly. Finally, keeping the old intensity with the new color, I put the new pixel into the old pixel's frame-buffer position.

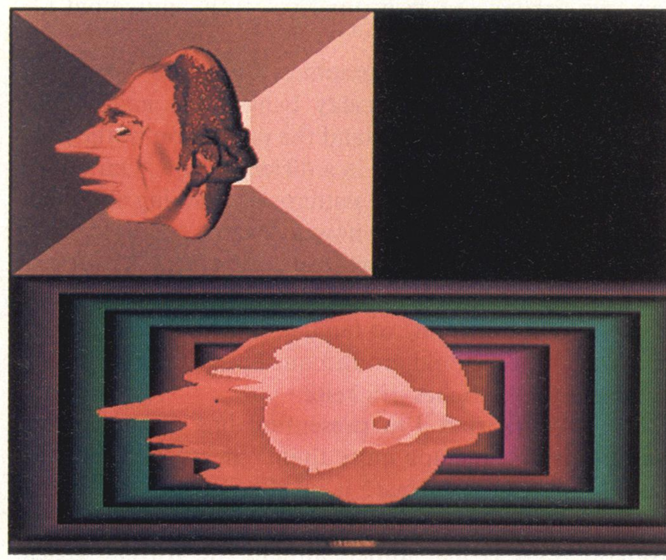


**Figure 3. "Egghead."**

This scheme actually started out as an attempt to make use of artistic warm/cool space to exaggerate depth cues—a trick that makes a two-dimensional picture look more three dimensional. Later pictures along this line were more successful but were less visually interesting. In my work, I have the general attitude that the visual image takes precedence over the concept behind it.

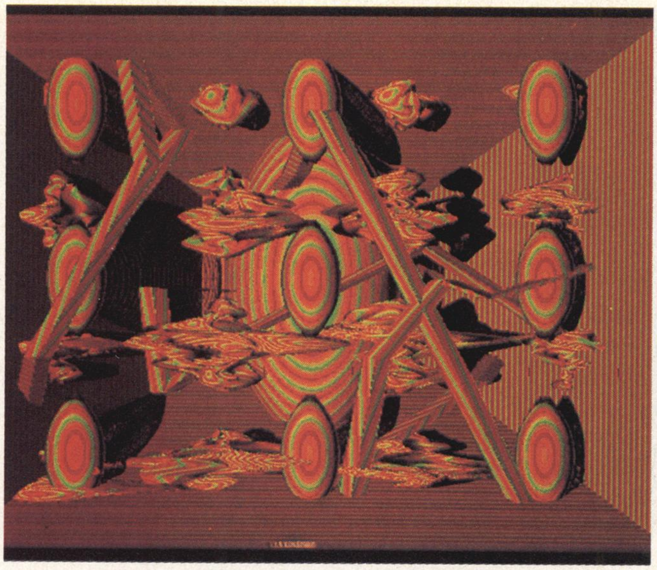


**Figure 4. "Warm-Cool One."**

## The figure series

Two of the most powerful tools designed for three-dimensional computer graphics are the "polygon tiler" (or "polygon renderer") and a three-dimensional points-polygon database to generate images. In the greater reality, the points-polygon database contains numeric information on how to render a particular complex form; the polygon tiler understands this format and renders a view of the form in the visual frame-buffer space. I shall keep the description brief here and again refer the reader to longer descriptions in both the Sachter and Glassner books.

I developed the polygon tiler used extensively in the next set of works to reflect my own aesthetic interests. I actually used it for the first time in "Space Carrots." Notice in "Space Carrots" that the randomly formed beamlike object intersects the space. The sides of the beam, with accompanying aliasing artifacts, were rendered by the tiler.

In "Figure with Aura and Guards" (Figure 5) we have three apparent instances, or views, of a female figure defined as a points-polygon database and rendered using my polygon tiler. I say "apparent instances" because the central form in the picture actually is composed of several exploded instances of the figure, which are overlayed with various degrees of transparency. The form explosion occurs along the vector direction of the polygon normals, which are calculated at each vertex of polygonal intersection. In this case each "normal vector" at a particular vertex is the sum of vectors perpendicular to the surface of every polygon which intersects at that vertex.[3] If the normals were rendered along with the polygons, the figure would look like a porcupine. However, the normals are like ghosts in that they are not rendered from the greater reality into the perceived reality but may affect it nonetheless.

In another behind-the-scenes maneuver I have included the egg concept as part of the greater reality of "Figure with Aura and Guards." The texture on the surface of the figures is a reflection of the egg's presence. The subtle mechanism behind the presence of the egg needs further clarification: The egg, in this case, is a sphere with infinite radius whose center is the center of the female database. Instances of the picture's Z-buffer are randomly placed on the surface of this sphere as part of the following algorithm: Rays from the figure emanate along the normal vectors away from the figure and toward the sphere's surface. They retrieve Z-buffer patterns and colors to be put back on the surface of the figure.

Here, we have a dance that is an interaction between the Z-buffer mapped on the surface of the invisible egg and the figure being rendered at the egg's center. A change in one affects the other. As in a dance of subatomic particles, complex time-space interactions occur that make events appear unpredictable.

In addition, the figure's dance with the egg is symbolized by her luminous emanations toward the egg. In fact, the central figure turned out to be so delicate that I just had to put two guards in the picture to protect her.

The remaining pictures in this series were produced by the same algorithmic mechanisms used in the previous ones, and all contain the dance with the egg. "Sparkling Giacometti Sequence" (Figure 6) reflects the use of the "augmented" transparency mode of my polygon tiler. This renders transparent polygons with sparkles at the edges of the polygons. The sparkles really are the result of

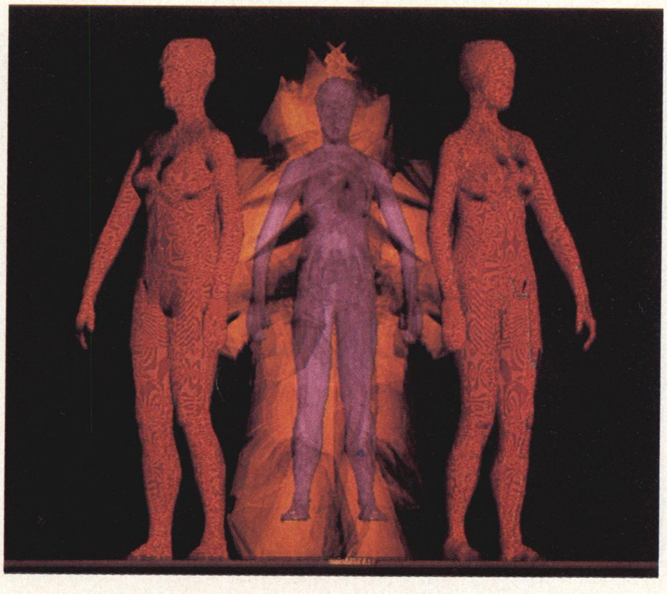


**Figure 5. "Figure with Aura and Guards."**

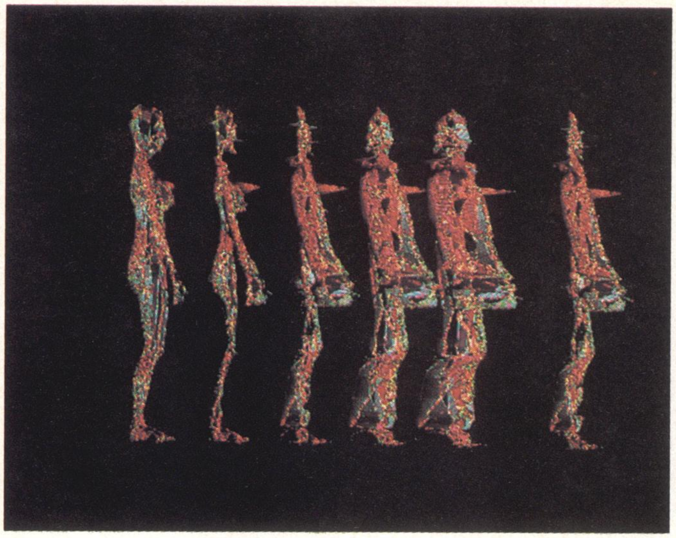


Figure 6. "Tall Movement."

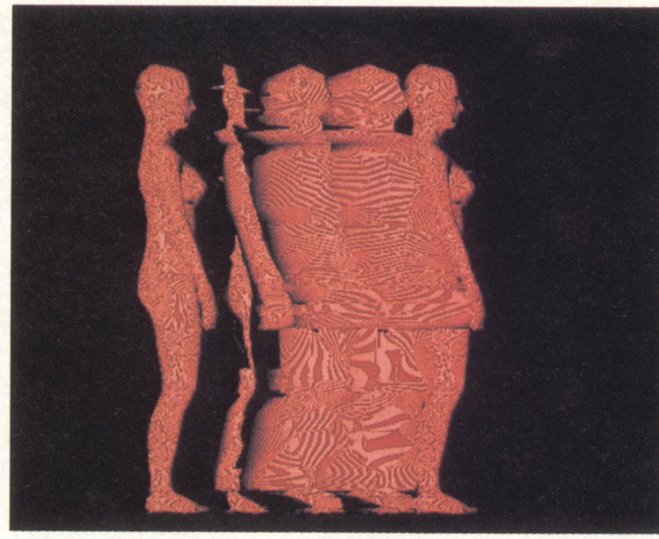


Figure 7. "Sparkling Giacometti Sequence."

a bug, or programming mistake, that occurred during the implementation of the transparency feature. Here is a case where a mistake in the code became a "feature" of the program. I kept the bug as part of the program and worked around it to complete the tiler's transparency option.

The four works "Sparkling Giacometti Sequence," "Tall Movement" (Figure 7), "Stretched Movement" (Figure 8), and "Stretched Movement Zoomed" (Figure 9) are products of a certain spatial tension programmed into their greater reality. This spatial tension causes distortions to figure instances that are placed in a spatial line with each other. The distortion occurs along the direction of the normal component, which is in line with the other figures. However, the end figures remain unaffected when rendered. The distortion also varies from figure to figure depending on where a figure is placed in relation to the others.

"Picasso Two" (Figure 10) is the pinnacle of this series. It was generated after Figures 11 and 12, which were done in conventional media, and shows the relationship between the two media and how they influence each other. Figure 10 is the product of the complex interaction of every element mentioned so far. In this work the effects created by chance, grouped with those created by unpredictably complex algorithmic interaction, form

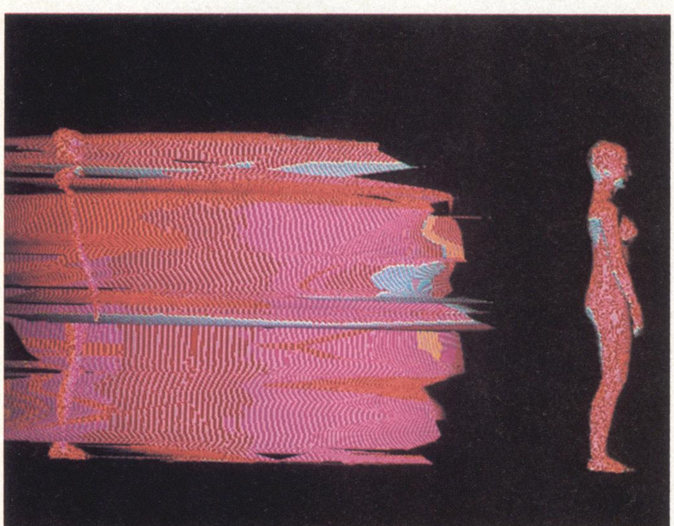


Figure 8. "Stretched Movement."

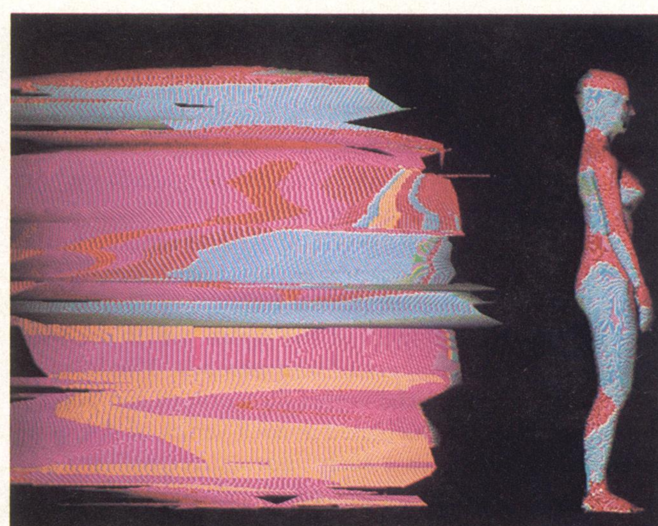


Figure 9. "Stretched Movement Zoomed."

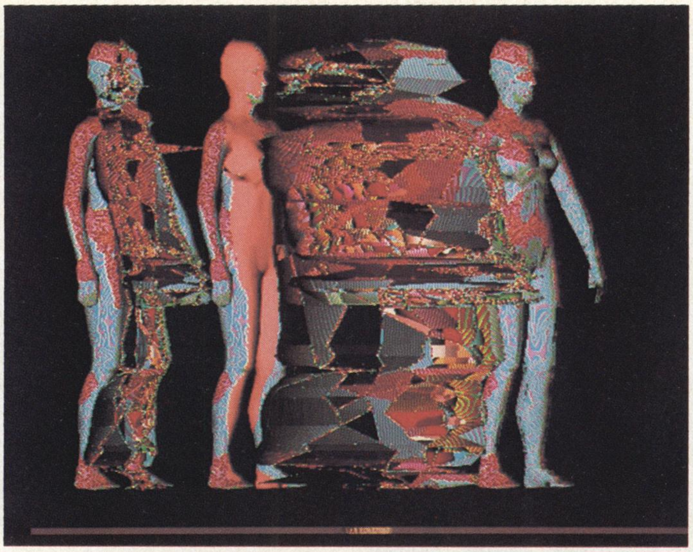


Figure 10. "Picasso Two."

a visual tension when played against the solid definition of the figure. This tension in visual space is similar to that which I find in many of Picasso's paintings.

## Space forms

Spatial tension is also an integral component of the next series I attempted—a series much different from the previous one. These new images are the product of a unique attempt to create a form-generating space that resembles, in some sense, our own real star-filled "outer" space. This new visual space has stars in it just as does cosmic space. They create gravitational forces on objects that enter the space. However, that's where the similarities end. The visual space differs from outer space in that these stars have no mass, and their interacting gravitational forces actually create visual forms. Within each of these contained spaces are placed several random points, or stars, that exert gravitational forces on the space. A blobby form is then generated in their midst by the forces created by the stars. "Space One" (Figure 13) is a six-by-six matrix of star spaces with their associated blobby forms. "Tex Blob" (Figure 14) is one of those space blobs with patterns texture-mapped onto its surface. And "Warm-Cool One Dipped into Gravity

Figure 11. "Pablo."

Figure 12. "Painting 8-72."

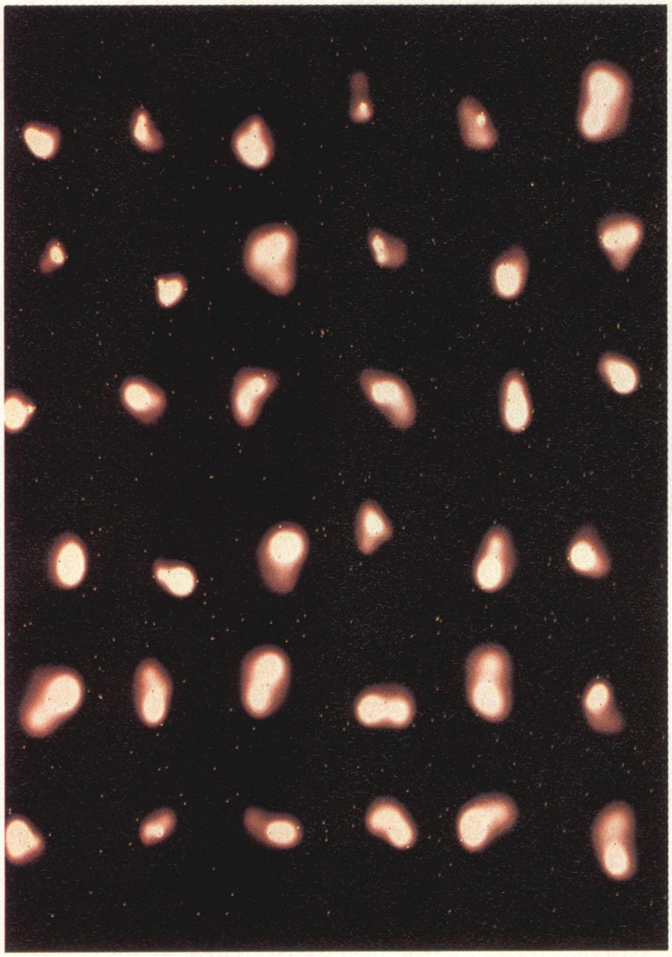


Figure 13. "Space One."

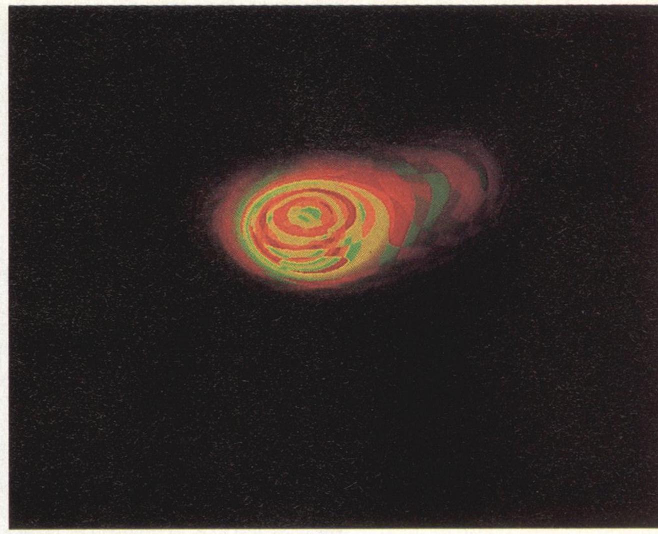


Figure 14. "Tex Blob."

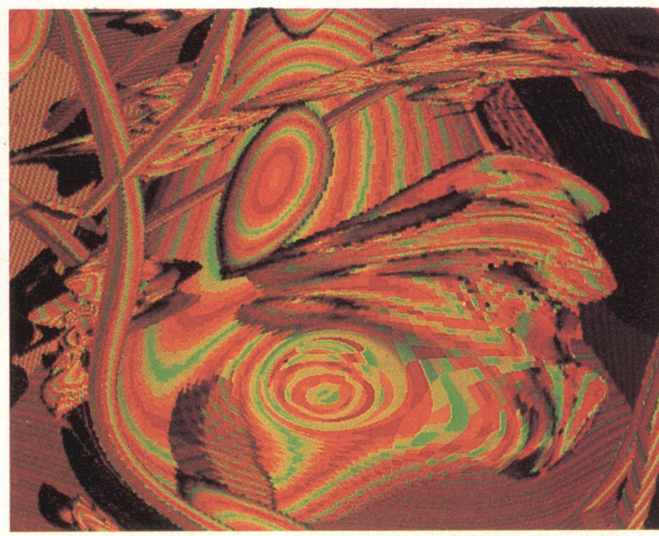


Figure 15. "Warm-Cool One Dipped into Gravity Space."

Space" (Figure 15) is Figure 4 dipped into one of these gravity spaces. Finally, "Folded Space" (Figure 16)—a freer, more complex result—is composed of overlapping spaces that create blobby forms which ripple away from their centers.

Earlier, we mentioned the need to create more space-oriented computer graphics tools for more right-brained, holistic methods of design. In some sense, this last series involves a primitive attempt to do just that. Please do not infer that the visual spaces should resemble our cosmic space—that was simply part of its artistic implementation—rather, the merit of its artist-tool design lies in its facility for designing a visual space, which, in turn, controls the design and placement of objects within it. This reverses the usual method of spatial design within which objects are arbitrarily placed together. Philosophically, the objects in the real greater reality are much more related than what our senses normally lead us to believe. My artwork attempts to suggest that relationship. I would prefer general artistic tools that do the same.

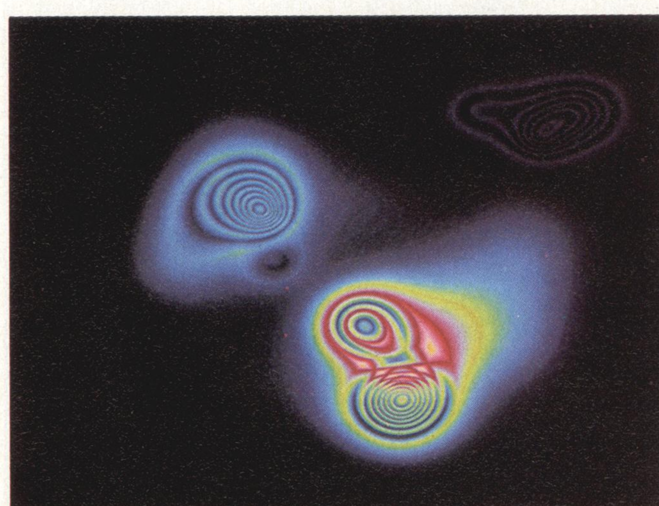


Figure 16. "Folded Space."

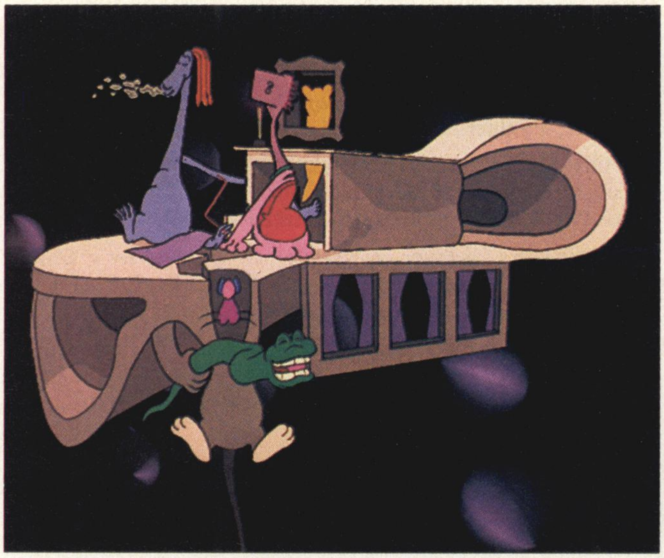


Figure 17. "Apartment Spin 1."

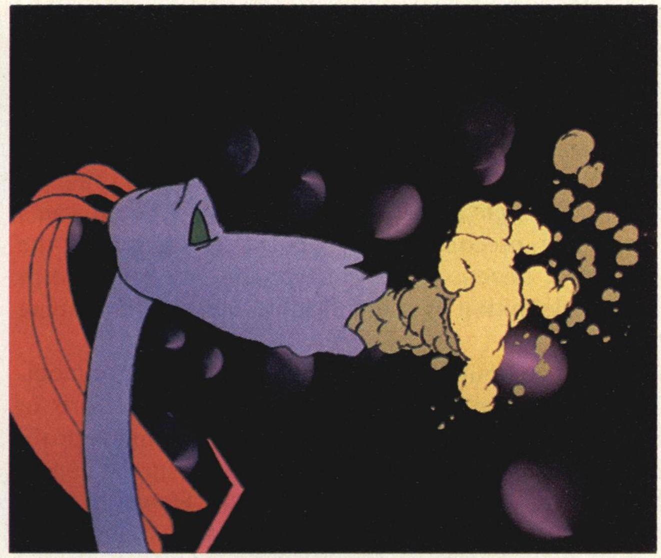


Figure 18. "Eno with Smoke."

## Further work

For my next work I chose a medium more flexible for realizing psyche/art involvement with less emphasis on mathematics. I used traditional animation techniques with a computer production base. This 22-minute piece called "Living Above the Mouse's Ear" (see Figures 17-20) reveals psychic permutations that create their own dreamlike spaces—typical of my conventional painting concerns.

"Apartment Spin 1," "Eno w/Smoke," "Landing on Eggs," and "Mouse in Hall" are four frames from this animation. Stylistically, they can be compared to the computer work above and to the two conventional-medium paintings "Pablo" and "Painting 8-72" (Figures 11 and 12). As you can see, my style has definite consistencies, irrespective of the media used. Notice the consistent use of saturated color within a personal color space. In particular, compare the painting "Pablo" with the computer-generated "Picasso Two." The use of the computer does not reduce my ability to express myself artistically; rather, it enlarges the range of ideas that I can use in my art.

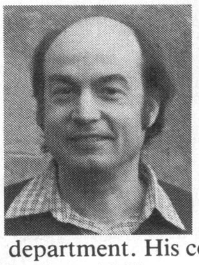
All of my artworks, whether computer-generated or conventional, have a strongly surrealistic basis. Working in conditions conducive to a psychic flow causes various interesting images and ideas to emerge on both conscious

Figure 19. "Landing on Eggs (Full)."


Figure 20. "Mouse in Hall."

and unconscious levels. Here one can relate the creative processes and visual products to theories based on manifestations of the numinous element, or the collective preconscious. J. C. Gowan defines "preconscious" as "...that aspect of the psyche, sometimes, not always, available to the ego. Preconscious insights tend to be expressed through right-hemisphere imagery."[4] Gowan's "collective preconscious" concept is similar to the "collective unconscious" concept of Carl Jung except that its definition is more flexible toward "becoming conscious."

Making computer art through the design and/or modification of programming tools requires great efforts in time and energy. Just to develop the programming proficiency necessary to work this way, I passed up the opportunity to make a lot of conventional art. However, computers are seductive and fun; I enjoy programming. Time and energy permitting, programming by day and painting by night gives a pleasant right-brain/left-brain balance to my life. It even gives me a means of making a living without compromising my art. However, you will notice that these last two statements are indicative of the social aspect of the problem. I feel that resolving this duality on an individual basis is important enough to warrant the effort. Such efforts further the development of consciousness, both individual and collective. ■



**Duane M. Palyka** is a senior research scientist at the New York Institute of Technology Computer Graphics Laboratory. He received both a BS in mathematics and a BFA in painting from Carnegie-Mellon University. During his employment as a research associate and systems programmer in computer graphics at the University of Utah, he received an MFA in painting from their art department. His computer art has been exhibited internationally since 1968.

## References

1. Jose A. Arguelles, *The Transformative Vision (Reflections on the Nature and History of Human Expression),* Shambhala, Boulder, Colo., 1975.
2. Judy E. Sachter, "The Basic Concepts of Three-Dimensional Computer Graphics for Artists," unpublished master's thesis, Ohio State University, 1984.
3. Andrew S. Glassner, *Computer Graphics User's Guide,* Howard W. Sams & Co., Inc., Indianapolis, Ind., 1984.
4. John Curtis Gowan, *Operations of Increasing Order;* by the author, 1426 Southwind Circle, Westlake Village, Calif., 1980.

## Bibliography

Betty Edwards, *Drawing on the Right Side of the Brain,* J. P. Tarcher, Inc., Los Angeles, 1979.

John Curtis Gowan, *Trance, Art, and Creativity,* Creative Education Foundation, Buffalo, N.Y., 1975.

Palyka's address is Computer Graphics Laboratory, New York Institute of Technology, PO Box 170, Wheatley Road, Old Westbury, New York 11568.