COMPUTER GRAPHICS WORLD

GRAPHICS TECHNOLOGY

Geometry and Imaging

Clarifying the major distinctions between the two domains of graphics

By Alvy Ray Smith

There are two quite distinct ways of making pictures with computers. The geometric way is widely understood, and often thought to be the only way. The imaging way is less intuitive, but addresses a market that's probably as large or larger than that for geometry.

Worlds Apart

The terminologies and theories of the two worlds are also strikingly distinct as are the hardware devices to implement them. Yet understanding the differences helps users know which technology to use for their applications today, and allows both developers and users to take advantage of the best of both worlds tomorrow.

Geometry-based picturing begins with the description of objects or scenes in terms of common geometric ideas: polygons, lines, spheres, cylinders, patches, splines, and so forth. Of course, these are mathematical abstractions, not pictures. So to make a digital picture of a geometrically described object requires that it be rendered (or rasterized or scan converted) into pixels.

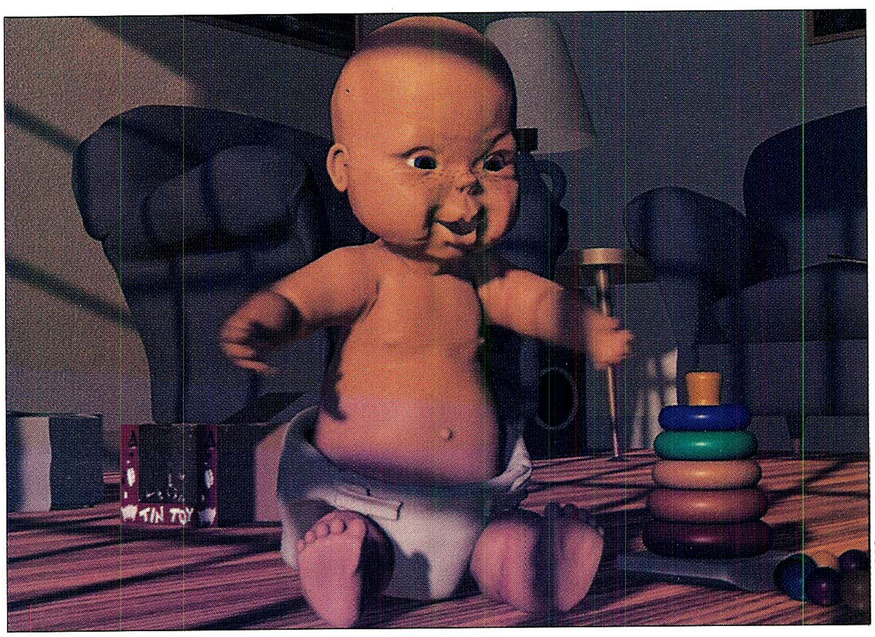
Imaging-based picturing, on the other hand, begins with a set of discrete samples—pixels—of a continuum, usually placed on a uniform grid. Sometimes these samples come from scan conversion of geometry, but in the majority of cases they come from nongeometric sources, such as digitized satellite photographs, computed tomographic (CT) or magnetic resonance (MR) scans, digitized X-radiographs, electronic paint programs, seismic sensors, supercomputer simulations of partial differential equation systems, or laboratory measurements. In all cases, an array of samples, or numbers, comprises the original data, rather than a display list of geometric primitives.

The imaging domain is discrete by definition and integer arithmetic typically suffices, whereas geometric concepts live in real continuous space, requiring floating-point precision arithmetic for accurate computer representation.

In some cases, it is possible to extract geometric data from sampled data, reenter the geometric domain, and then render the geometry to reenter the image domain. However, this process is not required to make pictures. In fact, doing so frequently introduces thresholding artifacts (jaggies) that may be highly undesirable, as in medical diagnostic imaging, which insists and depends on no alterations of its data. Direct imaging of data arrays avoids such undesirable artifacts.

The distinction between geometry and imaging is not the distinc-

Alvy Ray Smith is Cofounder and executive vice president of Pixar (San Rafael, CA).



© Pixar 1988

This geometry-based frame of the baby from the film *Tin Toy* was generated from a description containing over 2 million antialiased and shaded polygons. It uses most of the techniques in the Geometry Sophistication Meter on page 92.

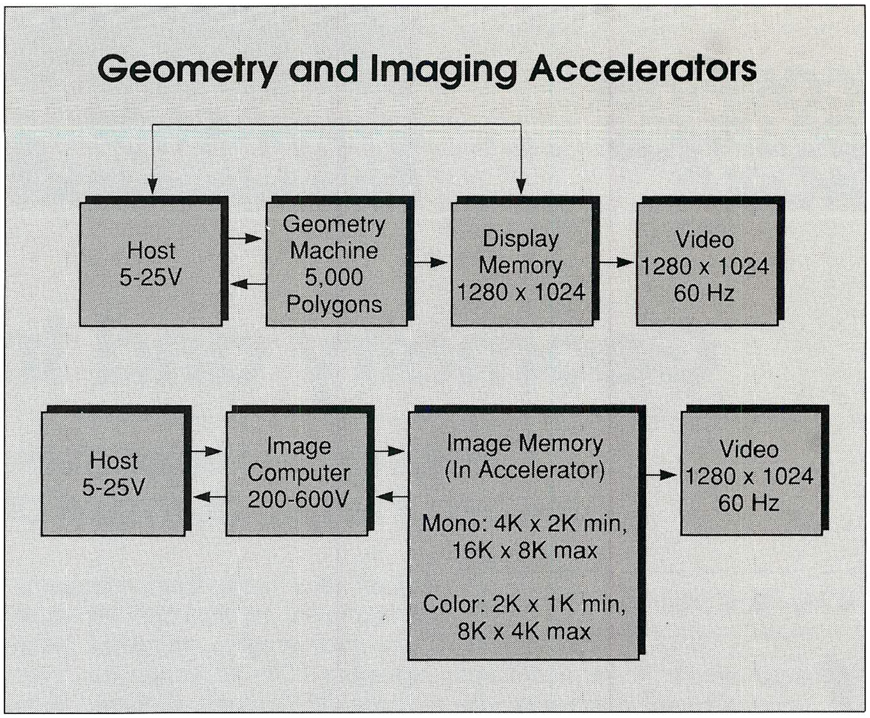
Reprinted from the November, 1988 edition of Computer Graphics World

tion between image synthesis and image analysis; an electronic paint program is an excellent example of an imaging technique. Nor is it the distinction between computer graphics and image processing; image processing is only a subset of imaging, and computer graphics loosely covers picturing from both domains. The fundamental difference is whether the elemental datum is a geometric or a numeric entity, that is, a polygon or a pixel.

Architectural Differences

The distinction between geometry and imaging is reflected in the special-purpose hardware accelerators. All graphics computations discussed here could be implemented on a general-purpose computer, such as a workstation host, minicomputer, mainframe, or PC. But these are insufficient in terms of price/performance for both geometry and imaging computations. The general-purpose machines with relatively lower computational power lack sufficient memory and the ability to perform computations in a tolerable amount of time. The more powerful machines use cycles that are too expensive compared to accelerators.

The figure of Geometry and Imaging Accelerators compares and distinguishes between the modern geometrical and imaging accelerators, that is, between "geometry engines" and "image computers." The following points explain the figure above:



The most obvious difference between geometry and imaging accelerators, besides accelerating different aspects of graphics, is their memory requirements and architectures. (V equals one DEC VAX 11/780 equivalent.)

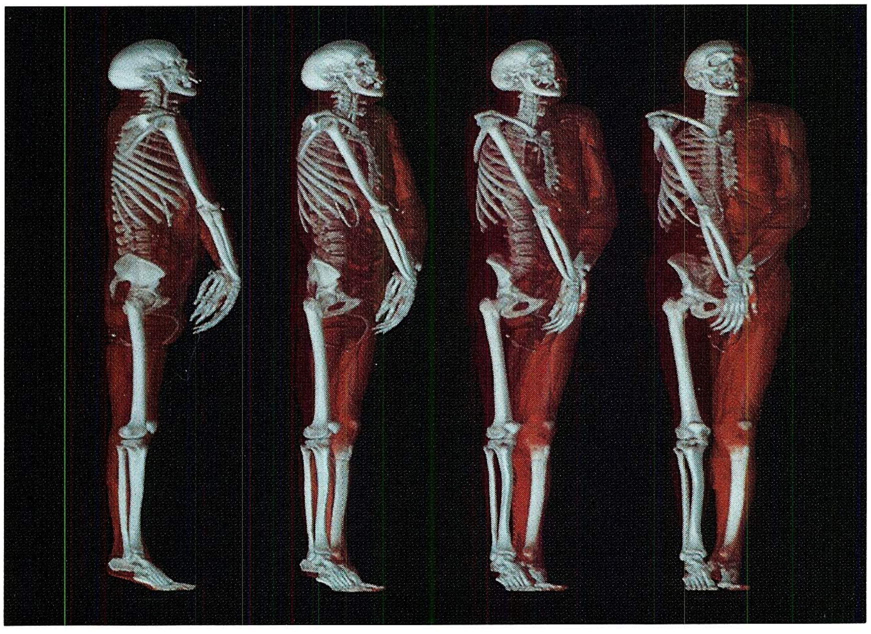
By definition, both kinds of accelerators require a host. Typical workstation hosts have computational power in the range of 5 to 25 times that of the VAX 11/780 from DEC. Host machines may have very large memories that can be copied into a display memory for either type of accelerator.

The geometry accelerators have the ability to generate about 5000 shaded geometric polygons per *picture* in real time, which is defined as 10 to 30 pictures per second. To put this in context, the frame of the baby from *Tin Toy* is a geometry-based picture with 2 million polygons. There is no machine today, regardless of price, that could generate this picture in real time.

Imaging accelerators are specialized to compute on images only. They attain computational power 200 to 600 times that of a VAX 11/780. This one-to-two order-of-magnitude increase in power over the host is what makes an imaging accelerator attractive.

Geometry machines typically generate, as their final output, pixels that are written into a display memory of 1280-by-1024 resolution. In some cases, host memory is also available to the display, but this memory is computed on by the host computer, not the accelerator. The figure distinguishes between memory in the accelerators only.

Imaging accelerators include huge image memories that can



Dr. Elliot Fishman, Johns Hopkins Med. Inst. © Pixar 1988.

In this imaging-based reconstruction of a human being from computed tomography (CT) scan data, each copy of the man was generated from a description consisting of over 40 million sample points on a 3D grid.

dwarf the display memory. This memory is distinguished from host memory by being directly accessible by the accelerator. A few examples demonstrate the requirement by imaging for lots of pixels. A remotely sensed satellite image might be 6K-by-6K pixels per single image. Three-dimensional datasets from medical diagnostic imaging are about 256-by-256-by-256 pixels each; some of them are 512-by-512-by-512 pixels each. A professional-resolution graphic arts page, at 300 pixels per inch, requires 8 million pixels of four-color channels (CMYK) each, for a single page of 8½ by 11 inches.

In summary, geometry accelerators are measured in terms of the number of geometrical objects they can manipulate in real time. Imaging datasets are typically so large that real time is not yet an appropriate measure for them (unless programmability is sacrificed). So imaging accelerators are measured in terms of the number of pixels they can comfortably manipulate on the order of 100 times faster than a host computer.

Sophistication Meters

Some geometry engines do a little imaging, and some imaging computers do a little geometry. The two charts—The Geometry Sophistication Meter and the Imaging Sophistication Meter—attempt to clarify the distinctions between the two domains and summarize the major techniques and terms in each. Both charts are ordered from bottom to top in each column by increasing sohphistication. The horizontal lines in the columns mark the real-time lines for each chart. Notice that the terms used on the two charts are almost completely different.

The capabilities of real-time geometry accelerators in 1988 lie below the real-time lines in the Geometry Sophistication Meter. Most of what is known about geometry-based graphics has not yet been pulled into real time. This is true of the shading, or visual content, of geometric objects, which is more difficult than the shaping, or geometry, of the objects.

Although it is not the purpose of this article to explain all the terms, note the following:

Under the Shading column in the Geometry Sophistication chart, procedural texture mapping should be carefully distinguished from full texture mapping, which is termed "image texture mapping." Procedural texture mapping is simpler than full texture mapping because it implies a short, simple description of a texture with a procedure. Full texture mapping is really an imaging problem since it requires the mapping of an array of pixels onto a 3D geometric surface with correct filtering and resampling.

Under the Geometry column, the concept of hyperpatches, that is, the rendering of geometric volume elements, is frequently confused with the imaging of volume-filling datasets. Geometry and imaging companies both use the terms "volume rendering" or "volume visualization" to describe the distinct processes. However, while geometry accelerators scan convert the 3D generalization of patches—hyperpatches—to obtain volume information, imaging accelerators directly display 3D datasets sampled on an integer grid.

Therefore, I propose that volume rendering be used henceforth for scan conversion of geometric volume descriptions (not their surfaces), and volume imaging be used for direct data display of 3D sampled datasets of arbitrary origin. That way, rendering maintains its association with geometry-based graphics into 3D. Volume visualization, then, is a general term for both. Neither of these techniques is possible in real time.

Under the Antialiasing column, the term "lines" falls within the real-time capabilities, and denotes that some of the more sophisticated geometry accelerators can now render antialiased line drawings. Even the ones that offer shaded polygons do not yet offer antialiased polygon edges, however. And they are a long way from completely solving the problems of jaggies on specular highlights and strobing on motion sequences.

Geometry Sophistication Meter

Shading "Shade"	Geometry "Shape"	Anti-aliasing	Complexity (Number of Primitives Per Picture)
Shading Language			
Materials			
Shaped Lights			
Distributed Lights			
Matte/Glossy			
Displacement Maps			
Environment Maps			
Bump Maps			
Image Texture Maps			
Radiosity			
Refraction			
Procedural Textures			
Phong Shading	Hyperpatches	Motion	100,000,000
Transparency	Patches	Textures	10,000,000
Gouraud Shading	Quadrics	Specular	1,000,000
Multiple Lights	Nurbs	Edges	100,000
Flat Shading	Polygons	Lines	10,000
None	Lines	None	1,000

Geometry sophistication increases from the bottom to the top of each column. The state of real-time geometry machines today is below the lines in each column.

The first thing to notice about the Imaging Sophistication Meter is that imaging applications are tremendously complex in terms of pixel count. This is the world addressed by imaging computers. Some geometry accelerators offer restricted imaging capabilities as well as geometry. The real-time lines in this chart indicate where in the scheme of things this limited imaging lies. In particular, it is restricted to essentially 1.25K-by-1K display memories. Of course, the host computer in geometry workstations can always do the imaging, but not at the order-of-magnitude lower price/performance that a special-purpose imaging accelerator offers.

Fine Points of Imaging

Again, it is not the purpose to explain all the terms, but note the following points:

Perhaps the most important column in this chart is Filtering. Image computers must implement sophisticated filtering based on the Sampling Theorem. Correct sampling, filtering, and resampling, are crucial to good imaging implementation. As indicated, the filtering, if any, done by geometry machines is first-order filtering for simple antialiasing of lines, often called "box" filtering. Many imaging problems require filtering solutions that much more nearly approxmate the ideal Sampling Theorem solution, such as those provided with windowed sinc functions or cubic filters based on Catmull-Rom basis functions.

For example, a satellite image obtained at great expense is often distorted by surface curvature and sensor artifacts; rectification of the image retaining its high-frequency information, demands the best filtering known. This is what image computers provide. It is extrememly expensive computationally, which is a reason for needing 10 to 100 times more power than a host provides for imaging.

The simplest type of imaging routines compute algorithms based on individual pixels (point operations in the Techniques column), not requiring any information from neighboring pixels, such as compositing or matte algebra routines that use matte algebra to

combine images with an alpha channel.

Higher order imaging operations use neighborhoods of each pixel for the computations (convolutions and filters in the Techniques column). These include all common filtering operations, such as edge sharpening, blurring, defocusing, and smearing.

Even more complex operations, such as the Fast Fourier Transform (FFT, Walsh, and other tranforms, under the Techniques column), require global information from all pixels in an image to compute each pixel. It is neighborhood and global operations on which parallel-computing architectures die because they lack sufficient interprocessor bandwidth and arbitration for the requisite neighborhood requests.

Some geometry accelerators have displays that can access the main memory of the host and "playback" a movie of images or volumes (Dimensions column) held in that memory. The point here is not to display precomputed film loops in real time. Rather it is the capability to compute them at special-purpose price/performance.

Imaging accelerators not only accommodate the sizes implied by higher dimensional images but they have memory architectures to enable efficient computations on them. For example, they do not require packing and unpacking of color components into ordinary computer words.

In the immediate future an understanding will arise of the complementary nature of geometry and imaging and, with it, applications that benefit from exploiting both together. It is not difficult to conceive of these. For example, a real-time geometry engine could be used to plan a flight path over a 3D terrain, which could be generated by an imaging accelerator from digitized satellite photographs and altitude sam-

Imaging Sophistication Meter

Techniques	Dimensions	Filtering	Complexity (Number of Pixels Per Image)
3D Painting, FFT, and Compositing			
Volumetrics			
Volume Imaging			
Warping			
Classification: Thematic, MR, and CT			
FFT, Walsh, and other Transforms			
Compression			
Soft-Edged Painting and Soft Fill		Very Wide Bessel	1,000,000,000
Convolutions and Filters		Sinc	100,000,000
Histograms and	Volume Movies	Cubic	10,000,000
Equalization	Volumes	Gauss	1,000,000
Point Operations	Image Movies	Box	100,000
Matte Algebra	Images	None	10,000

Imaging sophistication increases from the bottom to top of each column. Image computers address the entire chart, while geometry accelerators do limited imaging, as shown below the lines.

ples of the Earth's surface. This is clearly an imaging task requiring large image memory and very fast execution.

Also, in graphic arts, real-time geometry could be used to rough out a page layout, then imaging could be used for final touchup, composition, and correction of the full page makeup at professional resolutions (as opposed to desktop-publishing resolutions). Simple corrections, such as slight rotations of an image on the page, could be made digitally in seconds rather than the hours that would be required by an unaccelerated host. The "modern" way to rotate pictures in an unaccelerated graphic arts system is to tape the original photo on a drum scanner at the desired angle and rescan.

There is nothing, except perhaps a lack of understanding, that prevents these applications from being built today.

In the near future hosts will have the power of 25 to 100 VAXes. Geometry machines will have complexities near 10,000 shaded polygons per frame in real time. A new kind of accelerator will appear—a rendering accelerator—that will accelerate the non-real-time aspects of geometry ("above the line"), which the geometry machines will be incapable of handling. The imaging computers will reach 500 to 1000 VAX equivalents, maintaining their one-or-two order-of-magnitude lead over general-purpose hosts. And an interesting thing will happen at about 1000 VAXes: Real-time imaging will become possible, ushering in a new era in imaging.

Looking Forward

In the far future, of course, all of geometry and all of imaging will be integrated, standardized, and available in real time. Such capabilities will have become a commodity and expected of every computer. There is no reason to doubt that this will happen because digital components will continue to decrease in cost—while increasing in performance and density—for perhaps another 50 years, and people's demand for pictures will always increase and their desire for accuracy of representation will always grow. CGW