Mask of the Hive, a virtual embodiment of a simulated bee colony, an inherently decentralized interactive character, built under the Synthetic Characters behavioral framework.

Courtesy Michal Hlavac, Synthetic Characters Group, Media Lab, Massachusetts Institute of Technology, Cambridge, Mass.

George W. Fitzmaurice, Ravin Balakrishnan, and Gordon Kurtenbach

NOW INPUT TECHNOLOGY ALLOWS HIGH-FIDELITY SAMPLING OF THE PHYSICAL WORLD, DRAMATICALLY CHANGING THE METHOD AND USER INTERFACE FOR CREATING COMPUTER GRAPHIC IMAGERY.

SAMPLING, SYNTHESIS, AND INPUT DEVICES

Many efforts in computer graphics focus on mimicking reality to generate images and 3D models that capture the same visual fidelity and realistic properties as the physical world. Traditionally, these efforts start with an empty canvas. A combination of algorithmic techniques and user input is then applied to synthesize each element and layer visual effects until the desired fidelity and expression are achieved. Recent innovations with input devices promise to significantly alter this process from a start-from-scratch synthesis procedure to a sampling procedure. Elements from our physical environment are scanned to capture relevant 2D images or 3D content and then imported, manipulated, and merged with other imported artifacts or computer-generated elements. These "spatial sampling" approaches will drive some significant future trends in computer graphics.

Spatial sampling promises to save enormous amounts of time by allowing the import of preexisting spatial data. Moreover, sampling gives the user a large library of things to choose from—covering the entire physical world—including rich textures and sophisticated shapes, as well as the ability to work with familiar physical objects behaving in familiar ways.

An example of the advantage of sampling over synthesis can be found in the world of musical instructions and electronic sound generation. Initially, natural sounds were synthesized by combining simple sound waveforms, and the music industry was dominated by keyboard synthesizers. But the advent of sound sampling dramatically simplified the reproduction of high-fidelity natural sounds.

Bill Buxton of the University of Toronto and Alias | Wavefront and other researchers recognized the analogy with computer graphics. The typical elements of computer graphics, including object shape and motion, surface texture, scene lighting, and camera position, can all be sampled from the physical world, resulting in higher-fidelity imagery and simplifying user involvement. With sampling technology, users can now either synthesize elements from scratch or sample the physical world.

We take a slightly different perspective on synthesis and sampling, viewing all input devices as spatial samplers, at varying levels of abstraction, of the physical world. We examine sampling and synthesis issues relative to the abilities and compatibilities of spatial input devices used to accomplish a particular computer graphics task.

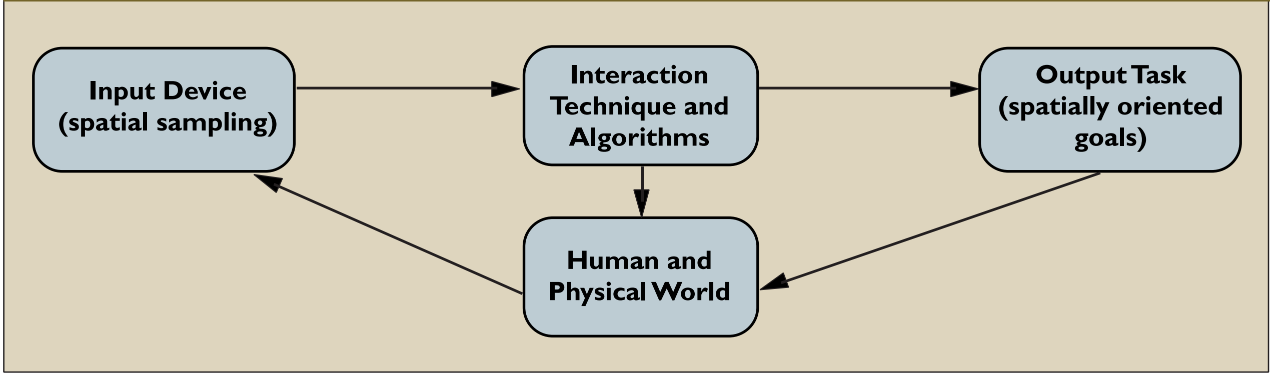
How does spatial sampling support computer graphics? At the highest level of abstraction, many tasks in computer graphics have a spatial quality. They are generally about producing and manipulating data we prefer to perceive spatially instead of symbolically. This abstraction encompasses a variety of tasks, such as designing shapes (in, say, an automobile or running shoe) or visualizing a data set (in, say, global weather patterns). The key concept is that the object of interest is spatial in nature, stemming, perhaps, from the object being a real physical object or a virtual object that will eventually be realized as a physical object. Alternatively, an object can be strictly virtual, though we interpret it as if it were a physical object with size and shape and occupying virtual space. The common element is that we prefer to perceive, reason, and give spatial meaning to these objects.

Given this interpretation, computer graphics can be seen as a means of supporting "spatial" computing with spatially oriented input and output goals. The core components of spatial computing are a user, input to the system, algorithms or ways the computer assists in a task, and some form of output (see Figure 1).

What is interesting about these simple components is that compatibilities (or incompatibilities) among them produce challenging problems and opportunities in computer graphics. For example, consider some of the problems caused by having to project 3D objects onto a 2D display. On the input side, users need devices they can point in 3D on a 2D projection. On the algorithm side, they need projection algorithms. On the output side, if the object being worked on is to be realized in the physical world, the 2D projection gives the user no haptic evaluation. Finally, for the human perceiving the 3D object, a 2D projection may at times be misleading or confusing. A change or limitation in the ability of any of the these key components of spatial computing can dramatically affect the nature of the technology as a whole.

In a "spatial sampling" input approach, the sampling device is pivotal in determining the abilities and design of the rest of the system. For example, an input device that samples only one 3D point at a time is suited to different tasks and may require a very different user interface from a system with an input device that samples thousands of 3D points at a time. New advances in input technology allow the physical world to be digitally sampled with greater ease, accuracy, and frequency.

Figure 1. Components of spatial computing.


Input Device (spatial sampling)
Interaction Technique and Algorithms
Output Task (spatially oriented goals)
Human and Physical World

Sampling and Synthesis

To explore the interplay among sample, sampling, and synthesis, we first define these terms:

Sample. A small part of something intended as representative of the whole.
Sampling. The act of collecting one or more samples.
Synthesis. The combination of several samples to better represent the intended whole.

Suppose a user wants to input a curve (the intended representation) using a stylus and digitizing input tablet. When the user makes a stroke with the stylus, the system senses a small series of points, or samples. These points are then combined by the system to form a curve, or synthesized into a higher-level representation. Note that sampling can include sensing human activity, as well as objects and phenomena from the physical world. Behavior, such as walking and hand gestures, can be sampled; so can objects, such as chairs and tables, and natural phenomena, such as waterfalls and trees.

We organize input technology trends in a framework highlighting the interplay between sampling and synthesis in relation to the resulting spatially oriented output. Such a framework could serve as a guide to designers when discussing the appropriateness of sampling devices and synthesis techniques for particular tasks.

One approach to classifying devices and techniques for interacting with computer graphics is to consider the input devices being used. One such taxonomy is based on the sensed properties of input devices, their degrees of freedom, and the type of human motor control required to operate them [4]. For example, the standard mouse senses motion in two degrees of freedom (the x and y position on a mousepad) and is operated by the user's hand and fingers. An isometric joystick (like those found on many laptop computers) senses force in two degrees of freedom and is operated by a finger. Another classifying approach focuses on defining a set of virtual devices, such as locator, stroke, valuator, pick, string, and choice, as a way of providing abstract mappings between input values and input devices [7]. For example, the virtual device "string" sends input values to the application from any device, including keyboards and speech recognition systems. While these classifications are valuable, they often do not capture the intent of the spatially oriented output.

Our approach considers how three main steps influence how users perform computer graphics tasks, as in Figure 1. First, input devices are used to sample human activity or physical-world objects and activity. Second, interaction techniques and algorithms help regulate how users communicate with the computer via input devices, as well as assist in the performance of the task at hand. Third, output displays track the progress of the user's intended spatially oriented output task. This information can be communicated to the user through a variety of ways, including display monitors, printouts, audio, and haptic-feedback interfaces. Note too that the output task and interaction technique can in turn influence how the user conceptualizes and performs the task.

A way to organize and unify this input/output framework is through a hierarchy of spatial primitives to determine compatibility between the various components. Most tasks in computer graphics—and the physical world as well—consist of spatial primitives at different levels of abstraction. These fundamental spatial primitives can be abstracted as point, shape, surface, volume, and scene (see Table 1). Note that further refinements and additions to this hierarchy are possible, though Table 1 is sufficient for our discussion here.

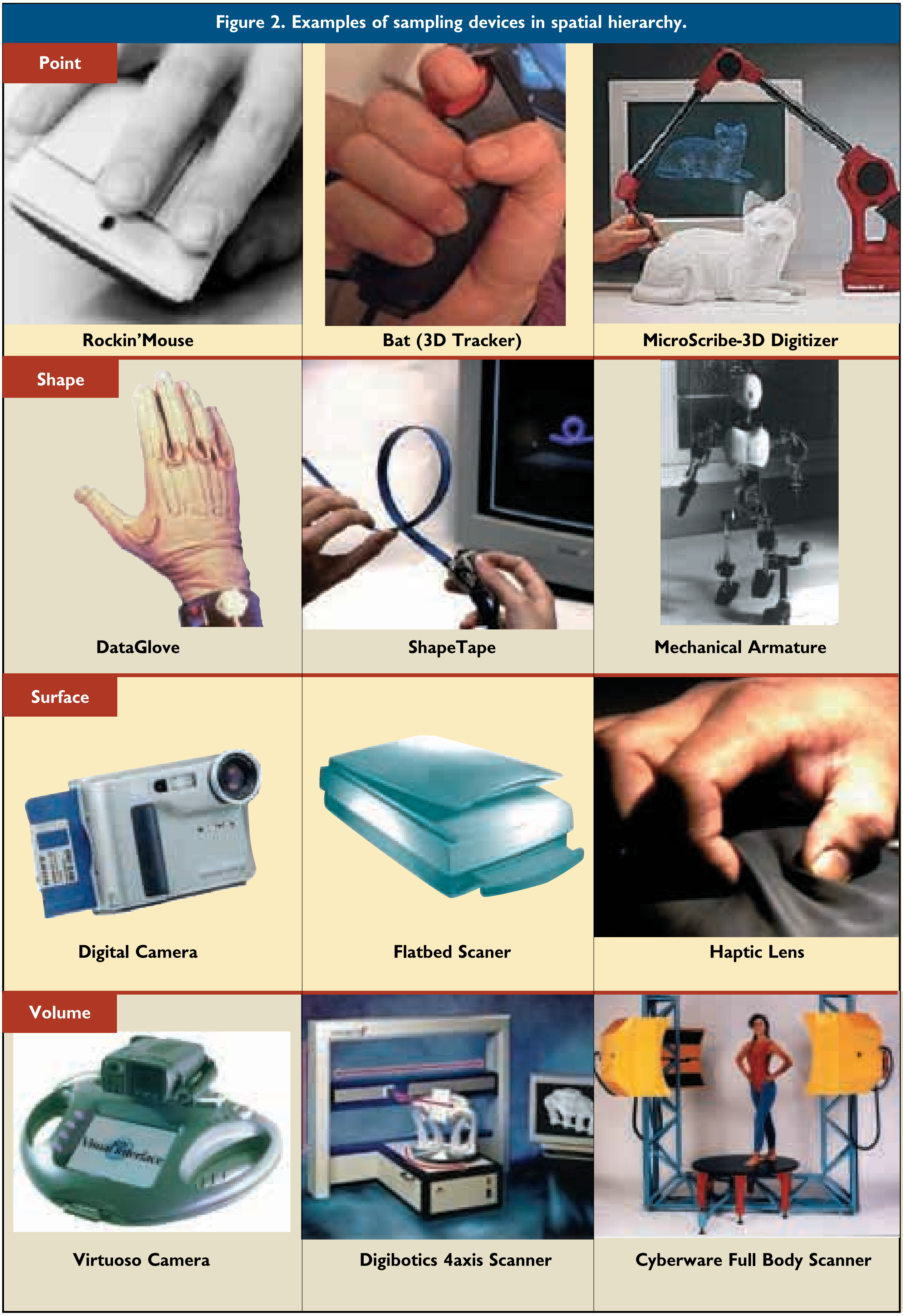
Table 1. Hierarchy of spatial primitives.

Spatial Primitives	Example
Point	2D, 3D location
Shape	curve, straight line
Surface	sheet of paper, photograph
Volume	box, bottle
Scene	room with furniture, streetscape

This classification reflects a hierarchy of spatial representations and abstractions in which each primitive can consist of several lower-level primitives. For example, two points define a straight line (shape), several lines define a surface, several surfaces define a volume, and many volumes are placed within a scene. The time dimension can also be applied to this hierarchy. For example, these spatial primitives can be sampled over time to capture such dynamic properties as motion or changing behaviors. In addition, a series of samples of one primitive over time can sometimes be interpreted as a higher-level primitive. For example, sampling a point in different positions over time yields a curve. However, while time is an important variable, spatial information is still the fundamental and more difficult property to sample and is therefore our focus here.

We can now classify input based on how well an input device supports the direct creation of these spa-

Figure 2. Examples of sampling devices in spatial hierarchy.



Point

Rockin'Mouse

Bat (3D Tracker)

MicroScribe-3D Digitizer

Shape

DataGlove

ShapeTape

Mechanical Armature

Surface

Digital Camera

Flatbed Scaner

Haptic Lens

Volume

Virtuoso Camera

Digibotics 4axis Scanner

Cyberware Full Body Scanner

SOURCES: MICROSCRIBE IMAGE COURTESY IMMERSION, SAN JOSE, CALIF.; CYBERGLOVE IMAGE COURTESY VIRTUAL TECHNOLOGIES, PALO ALTO, CALIF.; ARMATURE IMAGE COURTESY PUPPETWORKS, TORONTO; DIGITAL CAMERA IMAGE COURTESY SONY CANADA, TORONTO; VIRTUOSO IMAGE COURTESY VISUAL INTERFACE, PITTSBURGH, PA.

tial primitives and how much algorithmic synthesis is required to achieve the final desired output, including line drawings and 3D models. In other words, how close is the match between the sampled input primitive and the target output primitive. In contrast to earlier taxonomies [4], which were concerned with input from the perspective of properties sensed, we are concerned only with the resulting output sample of the input device. For example, Buxton's taxonomy distinguishes between the mouse and the isometric joystick [4], though we classify them as similar devices because each results in sampled points.

If input/output compatibility between the input device sample and the desired output primitive is imperfect, some synthesis, inferencing, or decomposition of the input data is required. For example, if the input device generates curve samples and the desired output primitive is a curve, there is strong compatibility between input and output. But if the input device generates point samples and the desired output primitive is a curve higher in the hierarchy of spatial primitives, as in Table 1, a synthesis or inferencing algorithm is needed to generate a curve from the given points. The inverse situation occurs when the input device generates samples that are higher in the hierarchy than the desired output primitive. A decomposition process has to be performed on the sampled input to create the desired output primitives. An example of this situation is when the input device generates a sample of a scene and we want to infer the main objects in the scene.

If an input device samples at higher levels of spatial representation, the amount of synthesis required to achieve the final output might be reduced. Indeed, current trends in developing input devices focus on such "high-level sampling" devices. But this focus does not imply that synthesis techniques are no longer required. On the contrary, the new high-level sampling devices serve as catalysts for new approaches to synthesis. New techniques are required and are being developed to intelligently interpret these new forms of raw sampled data.

IN ADDITION TO SPATIAL AND FORCE SAMPLING, WE COULD SAMPLE TEMPERATURE, AUDIO, SMELL, AND SPEECH, AS WELL AS SUCH COMPLEX INFORMATION AS HUMAN MOTION, BEHAVIOR, AND EMOTIONS.

Devices and Techniques

Using this input/output framework as a guide, consider how the following sampling devices and synthesis techniques support creation of computer graphics images.

Sample points. Computer graphics has a long tradition of sampling points using various 2D locator devices, including the mouse. A whole family of devices now provides a sampled stream of 2D points, including pens on digitizing surfaces, trackballs, touchscreens, and joysticks. In a coarse sense, all of these devices sample the position of a user's hand along a 2D work surface.

More recent innovations allow the user to specify a spatial position in 3D. The Rockin'Mouse (see Figure 2) has a curved base allowing it to sense its position on a 2D plane, as well as its tilt about a perpendicular plane [1]. This design allows a user to specify all three degrees of freedom of a point in 3D space at the same time. A variety of mechanical and electromagnetic trackers also makes it possible to sense 3D position. For example, the MicroScribe-3D device, from Immersion, San Jose, Calif., is a mechanical armature that samples points, while a variety of 3D trackers, such as the Fastrak, from Polhemus, Colchester, Vt., and the Bird, from Ascension Technology, Burlington, Vt., use electromagnetic technology to sense 3D position and the sensor's orientation [3]. A variety of trackers are also useful for simultaneously sampling a set of points in 3D space. On a much larger scale, the global positioning system (GPS) can sense the absolute position of any point on Earth, though such sampling is at reduced frequency and resolution compared to a mouse.

While the technologies for sensing points in space may differ drastically—from a simple mouse to GPS—a point is the fundamental primitive being sensed. In order to create higher-order primitives, such as shapes and surfaces, various synthesis techniques and a great deal of user effort are usually required. On the other hand, the volume of data that needs to be processed at any one time by the user and the system is relatively small and thus handled easily. Interactions using point samplers are often fairly simple, due to the limited number of dimensions being controlled, though this ease of interaction sometimes is at the expense of limiting the user's ability to express artistic intent.

Sample shapes. Some input devices enable sampling of multiple points at the same time, allowing the capture of shapes as input. For example, the CyberGlove from Virtual Technologies, Palo Alto, Calif., [3] is an instrumented glove that simultaneously samples the joint angles of a user's hand. The computer can use this data to create a representation of the hand's shape. Another device is the ShapeTape from Measurand, Fredericton, Canada, a flexible tape that senses curvature and twists with very high fidelity along its length, so curves are input directly [2]. Finally, instrumented mechanical armatures, such as those from Puppetworks, Toronto, [3] can be configured to represent a variety of articulated objects as stick figures and used to manipulate similarly articulated virtual creatures, as in Figure 2.

While points are the underlying sampled data from these devices, the physical structure imposed by a device results in a corresponding structuring of the points into a unified shape entity. This structure ultimately allows for the input to be treated at a higher level of abstraction—as a shape.

Input/output compatibility is high if the output primitive matches the input sample, so little or no synthesis is needed to get from input sample to output primitive. However, such compatibility should not be confused with ease of interaction. For example, the CyberGlove allows for systems that mimic natural interactions for dealing with objects in a 3D scene, though studies have shown that using a mouse (a point-sampling device) with the status quo "ray-casting" technique is faster for virtual object selection [12]. This is due to several factors, mainly the impoverished visuals and depth cues in virtual 3D displays. However, we recognize that a combination of the appropriate interaction technique and an input device that samples lower in the hierarchy of primitives can outperform a device that generates samples higher in the hierarchy.

On the other hand, if the desired output primitive is lower in the hierarchy, a deconstruction process is required to isolate points from the shape sample. This deconstruction is often done directly at the device-driver level, since the sample is a combination of points, or algorithmically from the shape representation, such as by extracting key points from a curve.

Sample surfaces. Moving up the spatial hierarchy, input devices whose output sample is a surface, such as a 2D plane that can be deformed in 3D space, have been available for years in the form of photographic cameras. But only recently have photographic images been used directly in computer graphics. Affordable flatbed scanners and digital cameras now allow the import of photographs and 2D textures into the computer. Initially, computer graphics techniques were used to visually modify the raw 2D images. Subsequently, images were used as texture maps to provide photorealistic detail for both 2D and 3D geometric objects.

More advanced synthesis techniques have since been developed to infer and extract more information from 2D images. Image-based rendering (IBR) is a set of synthesis techniques that creates—from multiple 2D images (or even a single image) of a 3D scene—new images of the scene from different camera perspectives within a limited range [8]. For example, such synthesis techniques as QuickTimeVR from Apple Computer, Cupertino, Calif., use 360-degree cylindrical panoramic images as input and digitally warp the image on the fly to simulate camera panning and zooming [5]. These approaches are unlike conventional computer graphics rendering, which requires 3D models of objects in a scene, as well as texture and lighting models, be created before a rendered image can be generated.

Image-based modeling uses two or more images from different camera perspectives of a single scene to generate virtual 3D models of objects in the scene [6, 10]. Since a computer model of the scene is created, images of the scene from any viewpoint (unlike IBR) can now be rendered as required. This synthesis technique takes 2D images as input and generates a single geometric volume or multiple volumes situated in a scene. While these techniques are new to computer graphics, the field of computer vision has long toiled over the problem of extracting depth and structure from multiple 2D images.

Image-based rendering and modeling techniques typically require some manual human intervention and are suitable for static scenes. A newer technique called "dynamic image-based modeling" processes a continuous stream of images acquired from a video camera to create models of dynamically changing scenes [10]. This process uses "imperceptible structured light," or special light transmitted for cameras to see but which human eyes cannot detect; the light is minimally intrusive to people working in the scene, unlike, for example, when using lasers for image capture.

Figure 3. Various stages in reconstructing a scanned vase.

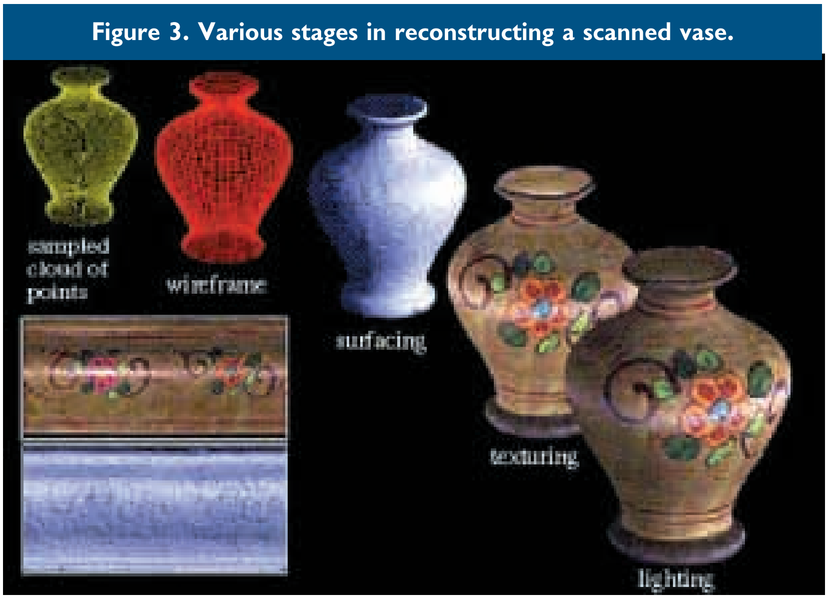


IMAGE COURTESY VISUAL INFORMATION TECHNOLOGY GROUP, THE INSTITUTE FOR INFORMATION TECHNOLOGY, OTTAWA, CANADA (WWW.VIT.IIT.NRC.CA/)

While photo and video cameras represent most surface-sampling technology, other technologies and techniques sample *internal* surface structure. For example, X-ray imagers scan an object and generate a cross-sectional image of its high-density internal contents. The key difference is that instead of imaging the object's external surface, internal surfaces are imaged. Apart from traditional use of X-rays in medicine, we could combine X-rays with image-based modeling techniques to create models of the internal structures of objects and scenes. These internal-structure models can then be used to generate more accurate and complete models of these objects. For example, if we want to model a human body, having an accurate skeletal model, or his or her internal structure, will likely facilitate more accurate modeling of that person's exterior surfaces and structural behavior.

Innovations in sensing material also represent new styles of sampling and interaction techniques. The haptic lens, as in Figure 2, is a prototype device whose output sample is an intensity map of the deformations of its half-inch-thick input surface (a pliable silicone membrane) [11]. This device can be used to scan the surface of physical objects pressed against the input membrane. Alternatively, users can manipulate the membrane with their fingers, using it as a dynamic input device for editing virtual surfaces in a very direct way.

A common theme in all these examples is that the samples are close enough to the target representation that the synthesis required is simply a process of combining multiple samples and adding some desired deviations from the original samples. In contrast, when point or shape samplers are used, a far more complicated synthesis process is needed to achieve the end result, since the goal is rarely a point or shape but an image or a model. For example, a user has to construct a 3D virtual scene iteratively, based on point and shape input, then create the final rendered image. Being able to sample at higher levels in the spatial hierarchy is valuable when the end result is also high in that hierarchy.

There are also disadvantages to using surface samplers. One is the difficulty of accurately synthesizing the intended result from a collection of samples; another is the increased volume of data that needs to be processed by the computer at any one time. And current system architectures are often challenged when required to process such high-bandwidth information in real time.

Sample volumes. A number of optical 3D scanners, as in Figure 2, sample object volumes using two approaches: passive and active scanning [9]. Passive scanning uses multiple stereoscopic images or video to reconstruct 3D volumes, similar to the human binocular visual system, which compares images taken from slightly different known positions to infer depth information. Some scanners, like the Virtuoso Shape camera from Visual Interface, Pittsburgh, Pa., project a stripe pattern on the target object to assist the reconstruction process [3]. The 3D volume data and the sampled texture maps can be fused to form a high-fidelity virtual representation of the volume.

Active scanning uses point or line stripe lasers and optical triangulation to sample 3D shapes. The laser or light sensors are usually placed on mechanical computer-controlled structures, such as the body scanner from Cyberware, Monterey, Calif., [3], orbiting the target object. Alternatively, the target object itself rests on a computer-controlled turntable, as in the Digibotics, Austin, Tex., four-axis Laser Scanner [3].

3D scanners generate sampled data consisting of a "cloud of points," and various synthesis techniques are needed to interpret this data. The cloud-of-points data is typically a collection of thousands and some-

Figure 4. Outputs of 3D scene sampling using a Cyrax 3D laser scanner.

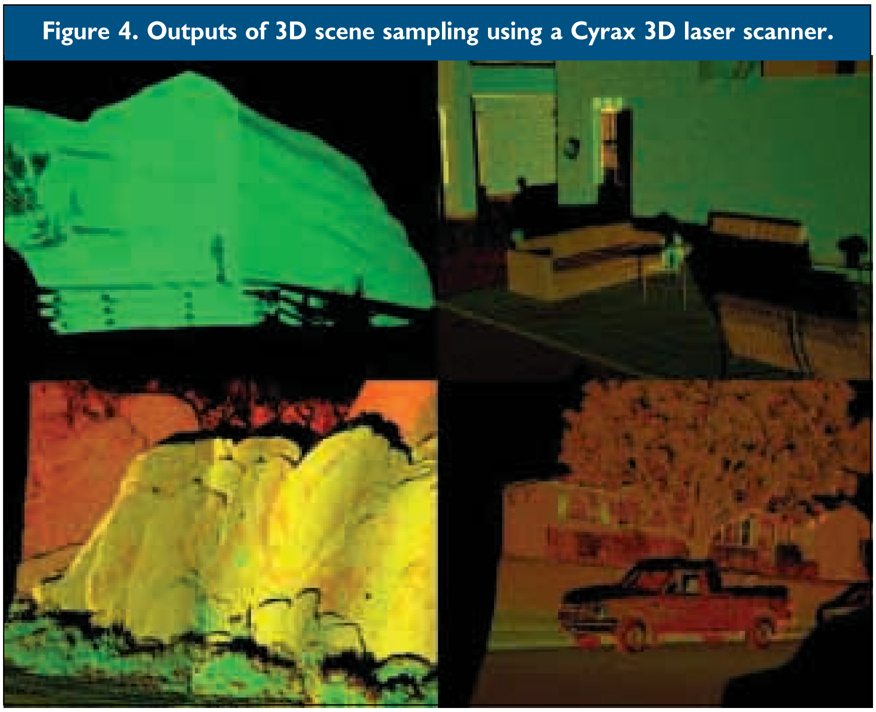


IMAGE COURTESY CYRA TECHNOLOGIES, OAKLAND, CALIF.

times millions of x-y-z points. More efficient data representations, known as wireframe models, are sought after to make the data more manageable for representing and manipulating the 3D object. The raw cloud-of-points data undergoes a process of coordinate smoothing, noise reduction, cluster analysis (to divide the data based on point density in space), and multiresolution analysis (to systematically reduce the number of points needed to represent the volume at different levels of detail). Finally, triangulation converts the set of cloud-of-points data into a set of wireframe triangles.

Recent advances in 3D laser scanning being pioneered by the National Research Council of Canada (www.vit.iit.nrc.ca) involve tricolor laser technology to simultaneously capture the range data and the color at each sampled point. Thus, the data is a six-tuple of x, y, z, red, green, blue and provides perfect registration of the geometric and color data.

Figure 3 shows the various stages involved in scanning a 3D object. The inset shows the raw range and color data of the vase obtained by the NRC Synchronized Laser Scanner taken at 1-degree of rotation increments. Yellow indicates the raw cloud-of-points data; red, the resulting wireframe model; and white, the surface generated from the wireframe. The color information is then mapped onto the surface, and the final image shows a synthetic view with a light source from the right side added to the rendering.

It may seem that the advent of volume samplers would make traditional computer modeling techniques obsolete, though this is clearly not the case in practice. While cloud-of-points samples generated by these devices allow the geometry of 3D objects to be imported directly into the computer, the data is often not the best representation for subsequent manipulation of the related virtual object for two reasons: We already have a sophisticated toolbox of techniques for manipulating curves and surfaces. And curves and surfaces are often already part of the user's mental model of what defines these objects. Thus, curve and surface representations have to be synthesized from the cloud-of-points data.

Sample scenes. Sampling a scene is much more challenging than sampling an object, because a scene consists of a spatial arrangement of many 3D objects. Some 3D scanners allow for detailed scanning of large volumes containing multiple objects (see Figure 4). Weather and aircraft radar systems are an interesting precursor to these concepts of scene sampling in that they sample a very large airspace looking for objects. The sampled data is often a collection of 2D images serving as horizontal cross sections of the sky at various altitudes.

Scene sampling is by far the newest and least developed of the technologies we have considered. Since these scanners generate cloud-of-points data, volume samplers could be used as a starting point for dealing with scene data. In addition, new synthesis techniques have to be developed if we want to be able to extract individual objects from a scene.

Sampling to the Extreme

Sampling technology and techniques could evolve in many directions. For example, sampling could change into an adaptive, iterative, non-uniform scanning process. Instead of having the sampling hardware generate a complete sample and send it to the synthesis engines, these two processes—sampling and synthesis—could communicate to provide a more optimized process. Creating a sampling library that automatically compiles what it has learned over the lifetime of the device could support future scanning processes in which the system detects familiar textures, surfaces, and objects, then adapts its current sampling procedure based on this prior knowledge.

One could imagine a continuous sampling procedure whereby 3D models are progressively built up as they are used. Imagine setting up a 360-degree scanner in your office to continuously scan the environment. Some objects are partially hidden so only partial geometry can be extracted. For example, a book on a bookshelf may expose only its spine to the scanner, but when the book is used, the intelligent scanning system can detect that new information on this object has been exposed and can add the new data to the book element registered so far. This approach builds up 3D geometry while the book is being used. Once the geometry is captured, individual pages can be scanned as they are exposed to the user. The scanner learns about objects as they are used in terms of their geometric structure, as well as their deformable and dynamic properties. Research systems are beginning to explore aspects of this sampling vision [10].

Although we often use sampling to get geometric structures and textures, it can capture a variety of other information. For example, a force sensor can sample the hardness/softness of a surface at a particular point. A series of such samples over a surface yields a realistic model of how the surface responds to touch. These models can then give users touch feedback when interacting with the virtual object through a force-feedback device. This force sampling approach can be used in place of current force synthesis techniques involving complicated mathematical models of the surface (see Salisbury's "Making Graphics Physically Tangible" in this issue).

In addition to spatial and force sampling, we could sample temperature, audio, smell, and speech, as well as such complex information as human motion, behavior, emotions, and relationships between humans and the surrounding environment.

Conclusion

These trends toward devices that sample ever-higher levels of abstraction could lead to advances along several fronts, including systems architecture and user interfaces, as well as computer graphics. From a systems perspective, the higher bandwidth and volume of data they generate will mandate development and use of new data-transfer standards, storage media, and display technologies. The increasing quality and fidelity of these sampling technologies will also further increase system performance requirements.

From an interaction perspective, these representations and technologies are opportunities for new and improved interaction styles, in much the same way that sampling a point yielded the mouse, which in turn spawned a new way of interacting with computers, including graphical user interfaces. In other words, entire new user interfaces could evolve based on high-level samplers.

High-level samplers are likely to be a powerful tool in spatial computing, though synthesis techniques will not disappear. Instead, new forms of synthesis will be needed to deal with these higher forms of data. Moreover, these devices and techniques promise to reduce the amount of painstaking manual labor required today to create computer graphics imagery, while producing richer and more realistic results.

References

1. Balakrishnan, R., Baudel, T., Kurtenbach, G., and Fitzmaurice, G. The Rockin'Mouse: Integral 3D manipulation on a plane. In *Proceedings of CHI'97 Conference on Human Factors in Computing Systems* (Atlanta, Ga., Mar. 22–27). ACM Press, New York, 1997, pp. 311–318.
2. Balakrishnan, R., Fitzmaurice, G., Kurtenbach, G., and Singh K. Exploring interactive curve and surface manipulation using a bend and twist sensitive input strip. In *Proceedings of ACM Symposium on Interactive 3D Graphics 1999 (I3DG'99)* (Atlanta, Ga., Apr. 26–28). ACM Press, New York, 1999; see www.dgp.toronto.edu/people/ravin/publications/i3dg99/shapetape.ps.
3. Buxton, W. Directory of sources to input technologies (1998); see www.dgp.utoronto.ca/people/BillBuxton/InputSources.html.
4. Buxton, W. Lexical and pragmatic considerations of input structures. *Comput. Graph. 17,* 1 (1983), 31–37.
5. Chen, S. QuickTime VR: An image-based approach to virtual environment navigation. In *Proceedings of SIGGRAPH'95* (Los Angeles, Aug. 6–11). ACM Press, New York, 1995, pp. 29–38.
6. Debevec, P., Taylor, C., and Malik, J. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In *Proceedings of SIGGRAPH'96* (New Orleans, Aug. 4–9). ACM Press, New York, 1996, pp. 11–20.
7. Foley, J., Wallace, V., and Chan, P. The human factors of computer graphics interaction techniques. *IEEE Comput. Graph. Appl. 4,* 11 (1984), 13–48.
8. Horry, Y., Anjyo, K., and Arai, K. Tour into the picture: Using a spidery mesh interface to make animation from a single image. In *Proceedings of SIGGRAPH'97* (Los Angeles, Aug. 3–8). ACM Press, New York, 1997, pp. 225–232.
9. Petrov, M., Talapov, A., Robertson, T., Lebedev, A., Zhilyaev, A., and Polonskiy, L. Optical 3D digitizers: Bringing life to the virtual world. *IEEE Comput. Graph. Appl. 18,* 3 (May/June 1998), 28–37.
10. Raskar, R., Welch, G., Cutts, M., Lake, A., Stesin, L., and Fuchs, H. The office of the future: A unified approach to image-based modeling. In *Proceedings of SIGGRAPH'98* (Orlando, Fla., July 19–24). ACM Press, New York, 1998, pp. 179–188.
11. Sinclair, M. The haptic lens. In *Visual Proceedings of SIGGRAPH'97* (Los Angeles, Aug. 3–8). ACM Press, New York, 1997, p. 179; see www.oip.gatech.edu/imtc/html/hapticsketch.html.
12. Ware, C., and Lowther, K. Selection using a one-eyed cursor in a fish tank VR environment. *ACM Transact. Comput. Hum. Int. 4,* 4 (Dec. 1997), 309–322.

George W. Fitzmaurice (gf@aw.sgi.com) is a research scientist in the interactive research group at Alias | Wavefront, Inc., Toronto, Ontario, Canada.

Ravin Balakrishnan (ravin@aw.sgi.com) is a Ph.D. candidate in the Department of Computer Science at the University of Toronto and a part-time researcher at Alias | Wavefront, Inc., Toronto, Ontario, Canada.

Gordon Kurtenbach (gordo@aw.sgi.com) is director of interactive graphics research at Alias | Wavefront, Inc., Toronto, Ontario, Canada.