Puppetry is the act of bringing inanimate objects to life through direct manipulation. Cartoon animation breathes life into drawings through image sequences. Combining the two is the delight of computer puppetry.

In the early 1960s, Lee Harrison III, an electronic engineer in Denver, Colorado, experimented in controlling an electronic cartoon character with live, real-time, interactive controls. At first he used dials and sliders, but in 1967 he put a dancer into an armature of Tinker Toys and potentiometers, and created the first “data suit” directly controlling a cartoon character with a human performer1 (see Figure 1). He then went on to evolve that system and commercialized it into one that let a human operator interactively move layered bits of imagery around on a TV screen and then play back the motions to videotape. Called Scanimate, it won him a 1972 National Academy of Television Arts and Sciences award. Almost all the TV flying logos of the late sixties, seventies, and early eighties were made by Scanimate systems.

However, as computer graphics technology evolved, Scanimate was outmoded by the slicker imagery produced using the less immediate methods of keyframing, and it has been all but forgotten. Today, computer-generated imagery can produce high-quality graphics in real time, that is, at or greater than 25 images per second (NTSC). We get computer puppetry by combining this capability with devices to track human motion.

In a computer puppetry system, various devices translate live performers’ body motions into the actions of a computer-generated character. The character appears simultaneously on the video monitor, which lets the performers continually regulate their performance to achieve the desired visual result. In some systems, one-to-one correspondence exists between the performer’s motions and the character’s motions. In other systems, indirect mapping occurs from performer motion to character action.

This real-time puppetry improves on the usual keyframe approach because animation can be done much more quickly. Also, performers, directors, and other artists get immediate feedback regarding the quality of the animation so that they can make corrections on the spot. For example, in Medialab’s production of the Donkey Kong Country cartoon series, 80 percent of the animation was performed with a computer puppetry system and 20 percent with conventional keyframing. Yet the time spent on each was equivalent.2

A little history
The first modern computer puppet that performed for a live audience was Mike the Talking Head at...
the Siggraph 89 electronic theater. It showed the real-time capabilities of the new Silicon Graphics “4D” machines and was driven by a special controller built by deGraf/Wharman. A single puppeteer controlled all the facial expressions, lipsynching, and head rotations. After introducing himself, Mike roused the audience with a lipsynched operatic solo.

In that same year, Pacific Data Images (PDI) created the first computer puppet in production television (not counting Harrison’s oscilloscope characters, which were used in various television animations) for Jim Henson Productions. Waldo C. Graphic appeared on a weekly Muppet TV show along with traditional puppets. He was controlled by a puppeteer using a custom mechanical arm with an upper and lower jaw attachment. For the performance sessions, PDI video composited a low-resolution version of the character with the real puppets and later, fully rendered it offline for the final tape.³

In 1991, Medialab, Paris, started weekly production of Mat the Ghost, a computer puppet using a newly developed in-house system (see Figure 2). Mat appeared daily on French cable TV (Canal Plus) for almost four years. A team of three puppeteers generated the entire week’s worth of animation (five animation minutes) in one afternoon session.⁴ Since then, Medialab has created and performed scores of characters for television, film, and live events.

Other companies have also developed real-time puppetry systems, including

- Simgraphics (South Pasadena, California), which has been doing live characters (mostly heads) since 1992;
- Protozoa (San Francisco), which sells the Alive! puppetry system used for the Cartoon Network’s Moxy and for the NBC/Microsoft Network’s commentator/cappuccino bartender Dev Null;
- Mr. Film (Venice, California), which has an in-house system they have used for several years;
- Windlight (Minneapolis, Minnesota), which also has an in-house system;
- DreamTeam (Israel), which sells a puppetry system that MTV uses for their CyberCindy character; and
- Digits ’n Art (Montreal), which sells a performance animation system using magnetic capture for the body and optical capture for the face.

**A computer puppet’s life**

The first step in bringing a computer puppet to life is creating the character’s concept. How the character will be used forms an important part of its design. To illustrate this process, I will refer to Medialab’s methods and procedures. I do this not to single out Medialab, but to draw on our experience for concrete examples of principles that prove similar for any computer puppetry system.

**Live performance**

Real-time animation can be used in many ways. The first and simplest is live performance. The character is broadcast directly from the computer’s image buffer with no postprocessing. Often the puppeteer/actor controlling the mouth movement provides the voice, but occasionally a particular voice talent must be used, as was the case for Bugs Bunny Live. Used both in France and Italy, Bugs’ voice was supplied by the local voices who have always done Bugs Bunny in those countries. One of the local puppeteers’ challenges was to adapt themselves to the actors’ speech patterns and cadences.

TV talk shows, trade-shows, and industrial sales and marketing conferences all use live performance. Medialab and Simgraphics lead the field in producing this type of computer puppetry.

**Recorded performance**

Performance for later broadcast proves slightly more complex. This also takes the imagery directly from the real-time generated computer frame buffer, but records it to video for postproduction. In postproduction, shadows, backgrounds, foreground masking, and other special effects are added, but for the most part the character appears in much the same way as it does on the computer screen. This form of production usually uses a pre-recorded voice track, which is delayed by a few milliseconds before being recorded with the video image. This short time period lets puppeteers react to the sound track and provide motion in proper synchrony. (Delayed voice output is used in live situations as well if the mouth puppeteer is not the same as the voice talent.)

If a performance is not going out live, then the puppetry team can retake sequences, repeating the whole action or just a part of it. For instance, if the facial expressions were correct but the head turned a little too quickly, then the head can be performed again, keeping the facial motion from the original take. Complex characters also can be performed in several passes. For a computer-generated character—Pepe the cricket, in the 1996 film *The Adventures of Pinocchio*—Medialab performed the two pairs of arms in two passes, first the upper set, and then the lower set.

Both performance models using real-time computer imagery as the final product require careful character design to optimize the polygons in the model, since polygon counts affect the real-time rendering speed. Experienced designers and modelers know how to exploit
polygons and texture mapping to provide maximum
detail with minimal resources. Greater movement areas,
such as the mouth and eyes, get more polygons. Lesser
movement areas, such as the forehead, torso, and back get
fewer polygons. Texture mapping can be used to add
detail in the place of geometry, but the texture mapping’s
size and quantity must be matched with the computer’s
capabilities. Character design sheets, which show the
character in its full range of expression and poses, have to
take these constraints into account, thus digital modelers
must be involved in the process from the beginning.

Another constraint on character design is the char-
acter’s performance requirements. How many puppeteers
will be required to animate the character? Can the
combination of available motion-capture devices support
the input requirements? The more expressions a character has, or the more things it can do, the greater
the demands on the puppeteers, leading to a potential
need for several performers. For simple characters in a
live situation, Medialab usually uses two performers: one
for the body and one for facial expressions. For
recorded performances, a third puppeteer is added to
achieve greater expression. Budget constraints also
determine the number of puppeteers. Typically, a
recorded program has a larger budget than a live per-
formance, thus more puppeteers.

The input devices for manipulating characters are also
important. It’s easiest to animate humanoid characters.
Multi-legged creatures such as insects, or no-legged
creatures such as fish, mermaids, and snakes, require
some thought as to how they will be animated: what
devices will be used, and how will the puppeteers con-
trol the movements? The results of these studies guide
the character’s design (see Figure 3).

The character’s complexity affects the design in
another way. Geometric deformations, joint smoothing
(“skinning”), dynamics, collision detection, and other
processes influence the character’s computational
expense. The computer’s processing power must be con-
sidered when including, for example, dynamic elements
in the character’s design. Although less costly, small
machines like the SGI Indigo cannot support as much
real-time computation as large multiprocessor systems
such as an SGI Onyx.

Designs that include loose clothing, flowing hair,
capes, and other computationally
expensive elements should be
approached carefully. Although
these can be handled in postpro-
duction, effective real-time solu-
tions have not yet been
commercialized.

**Postprocess rendering**

Other modes of computer puppetry have less stringent constraints
because they involve a postrendering phase. In postrendering, the
character’s complexity need not be
limited by real-time requirements.
At Medialab, we use two postrender
methods. The first is to operate the
hardware renderer at lower speeds, measured in
seconds per frame rather than frames per second. We use
a lower polygon resolution version of the character to
perform the motion in real time, then switch to a high-
resolution version for frame-by-frame rendering. As the
machine renders each frame, the frame is saved onto
disk. Another method involves exporting the real-time
motion to a commercial animation system and render
there. This also requires two versions of the model, one
low-resolution model for real time, and another for use
in the commercial system. In the commercial system,
we can add extra dynamics, shadows, textures, and
effects that may not be possible for the real-time system.

**Modeling**

Once we complete the design and the client approves
it, modeling begins. Although no standards exist for
character creation, the procedure at Medialab is proba-
bly typical.

Using a commercial modeler, we create the base (or
neutral) version of the character’s form and a skeleton
to support and move the body parts. We find that a
nonuniform rational B-spline (NURBS)-based modeler
works best for the organic forms that our puppets
ormally take and gives us more flexibility to make modifi-
cations as the model progresses. Eventually, however,
we convert the NURBS representation into polygons for
greater efficiency in the real-time system.

The skeleton that underlies the objects must be cre-
ated in such a way that we can map the motion-capture
sensors to the character. At Medialab, we try to fit all our
characters to a standard humanoid mapping for conve-
nience. However, we occasionally have to resort to cre-
ating unique mappings for fish or insects.

Along with neutral expressions, we design extreme
expressions (see Figure 4). Medialab’s puppetry system
uses multi-target interpolation for facial expressions, a
technique pioneered for the 1985 computer animated
short film *Tony de Peltrie.* Each preset extreme—happy,
sad, surprised, angry—represents a modified version of
the neutral expression. The performer uses the input
deVICES to mix varying levels of these extreme expres-
sions in real time to create an emotive face.

Once we finish the model, we import the character into
the real-time system and test it with performers. A char-
acter in motion is not the same as a character on a drawing or in a rendered frame. Many artifacts and problems not considered when examining static models come to light when we add motion. We carefully examine joint movement to make sure the parts all fit together when moving. We look at texture motion to watch for aliasing artifacts, distortions, and how they look across joints. We benchmark the character’s speed and make recommendations as to where polygons need to be removed for speed or added for better visual effect. After these observations, we often return the character to the modeling stage to adjust joint centers of rotation, modify geometry to add polygons in areas of extreme motion, remap textures, and reform contours that didn’t look as good in motion as they did in the static model.

During this stage we also add dynamic elements such as earrings, hair, feathers, and tails; behaviors like blinking, winking, and breathing; and special effects such as bubbles coming out of a fish’s mouth, color changes, texture changes, or preprogrammed motions. In Medialab’s system we do this by writing animation and behavioral scripts, which the system interprets in real time. Once the final character is ready, we test again for speed, optimizing a few elements here and there. On larger machines, we parallelize the per-frame computations as much as possible. The character is now ready to perform.

Performance

As described earlier, Medialab typically uses two to three performers for each character, one for the body and another one or two for facial animation, lip-synch, and special effects. Each of these tasks requires great skill and concentration. The performers must monitor their own actions, while at the same time, work together to create a unified personality. Occasionally, an unusually complex sequence or exceptionally expressive character requires the addition of a fourth performer to complete the team.

The body

The body performer wears a suit supporting 16 to 18 electromagnetic sensors that record the motions of the head, torso, hips, and limbs (see Figure 5). Other computer puppetry systems’ configurations range from 8 to 24 sensors. See the sidebar “Sensor Technology” for more information.

Performance issues

Since the sensors respond in real time, the performers drive the character’s motion directly and get immediate graphic feedback on a video monitor. It’s puppetry with a virtual puppet. This is an important distinction from just motion capture. The fact that feedback is

Sensor Technology

Electromagnetic sensor technology was originally developed in the early seventies by Polhemus to track jet pilots’ helmet motions for “heads-up” cockpits. It has a central source, which creates a pulsed magnetic field. A sensor in the field records its immediate position and orientation with respect to the source.

In the late eighties, people began to realize that if the system were to support multiple sensors, these could be attached to a body to record motion in real time. To this end, Polhemus released a multiple sensor system, and Ascension, a competitor started by former Polhemus employees, came out with Flock of Birds. These were the first commercial systems that provided multiple sensors capable of transmitting positions of the human body’s limbs at a reasonable sampling rate (20 or more samples per second). Flock of Birds was successful enough that several computer animation companies and research labs bought into the technology and began to experiment with motion capture and computer puppetry.

Since that time, the competition between Polhemus and Ascension has matured to the technology. Now they provide wireless systems with up to 32 sensors, enabling multiple performers to work simultaneously.

Most of today’s computer puppetry systems use either Ascension or Polhemus electromagnetic sensing systems to capture body movement. A brief calibration period “zeros” the sensor positions with respect to the body and to the performance space. More sophisticated systems, such as Medialab’s, also calibrates the performer with respect to the floor to prevent the appearance of sliding and passing through the ground. This calibration considers the differences in limb proportions between the performer and the character. Thus, a tall performer can play a squat monkey (like some of Medialab’s Donkey Kong Country characters) or a short performer can play a long-legged beauty queen.

Authorized licensed use limited to: TONGJI UNIVERSITY. Downloaded on August 03,2010 at 10:10:55 UTC from IEEE Xplore. Restrictions apply.
instantaneous, continuous, and complete lets the performer continually adjust the performance. Differences in morphology between the performer and the character, self-intersections (due to the morphological differences), changing camera positions, and visual aesthetics require a constant real-time tuning of the performance. For example, if the animated character has a dog’s snout, the nose may be several virtual inches in front of the face. If the sequence calls for the character to touch his nose, the performer must be able to find the exact spot in the performance space, perhaps several inches in front of his own nose. With a puppetry system the visual feedback lets this happen quickly.

In fact, one of the first actions a performer does when working with a new character is to map out the character’s shape in real space. For a long-armed character to touch its toes, the performer only needs to reach as far as his own thighs. A character with a large head combing his hair may require the performer to make motions several inches above his own head. In a motion-capture system, the performer does not know the extent of the character’s body, so either the motion is done in key-frame as a postproduction step or the performer must wear a precisely calibrated prosthetic.

Not only can the performer monitor the effect of his actions, the director can as well. This means that the director can work with an animated character as if it were a live actor (it almost is), giving stage directions, asking for a slight change in movement, or specifying a particular tilt of the head. The immediacy and authenticity of the feedback fuels the creative process. Frequently, the director and performers experiment with a scene, animating it several different ways before settling on a final version.

The idea of having only one performer appeals to producers trying to control costs, thus there is the pressure to load all the controls onto one performer. If the designers carefully construct the control system with an optimized set of expressions, it’s possible to have one or two performers without a noticeable reduction in expressiveness. However this works only to a certain point and depends on a highly skilled performer. We find these people to be rare.

In any performance system, the performer’s skill can make all the difference. In setting up a studio in Los Angeles, Medialab auditioned more than 40 performers. Surprisingly, we found that actors and mimes had a much harder time bringing the characters to life than did professional puppeteers when wearing the body suit. We theorized that mimes and actors had difficulty referencing the character from the screen and not from their own physical performance. If a sneaking walk didn’t make the character look like it was sneaking on the screen, the actors and mimes were much slower to recognize and correct their body motion appropriately than were the puppeteers. The puppeteers were accustomed to using the screen as a reference and creating an expression with a seemingly unrelated motion. They were much more successful driving our characters.

Facial animation

Facial expressions can be accomplished with several methods. Medialab performers use many devices, including electronic gloves, pedals, and joysticks to effect multi-target interpolation between key expression forms (see Figure 6). This allows natural artistic control over a range of expressions including those not in the human repertoire—bulging eyes, wiggling noses and ears, or the spreading of a peacock tail. We have had much success with this method, and in my opinion it proves the most effective for real-time cartoon facial animation.

Other groups, such as Protozoa, use preprogrammed expressions controlled by sliders or triggered by buttons or hand postures. When used properly, this technique yields good results, although expressions are limited to the given set.

Most other systems opt for a more straightforward method of facial animation using optical capture (see the sidebar “Optical Motion Capture”). These systems are similar to optical body systems, except they can operate in real time because they don’t lose markers due to occlusion. The markers are millimeter-sized reflective dots glued to the face at strategic points—along the nose, around the eyes, cheeks, mouth, and so on. They mount one or more cameras on a helmet to fix the head in camera space. Software tracks the markers and (in almost all systems) deforms a computer graphic facial model in conformation with the perceived positions of the points. Some systems use a muscle model based on Waters® model, where points activate simulated facial muscles with specific areas of influence.

The purported advantage of optical facial capture is that anybody can perform the face, including the body actor. Wearing a body suit, a facial capture system, and a microphone, one person can animate an entire character—a great advantage in a live situation. In practice, however, active body performers find wearing a facial system encumbering. In addition, facial motions need to
be exaggerated and “played” to have the desired effect (much like actors must overact to appear natural on stage or camera). However, in a recording situation, a voice professional, not someone skilled in facial pantomime, does the voice separately. The systems also break down with overly cartoon-like faces where the markers no longer correspond well with the computer graphic face. With optical systems, the best results are with faces similar to the actor’s face, as with the body optical systems.

Facial tracking can be done in other ways. For example, tracking points on the face and warping the character’s facial “skin” in a like manner, or machine-vision analysis of facial motion—such as smiling or winking—to apply to the animated face. However, commercial systems seem to adhere to the point deformation method.

Early in their history Simgraphics developed a mechanical facial system for their characters. The performer wears a light-weight headset with potentiometers linked to points on the face—lips, chin, cheeks, eyelids, eyebrows, and so on. The potentiometers capture facial motions, which the system maps onto the computer character. This facial system produces results similar to those from optical facial capture systems.

**Lip synchronization**

Lip synchronization forms a special aspect of facial animation. Natural lip movements, a complex phenomena, provide visual cues to help us understand speech. If the speaker’s mouth movement does not correspond with the speech, we experience discomfort and have a harder time interpreting what’s being said.

The following factors help make lip sync believable:

- the quality of the mouth shapes with respect to the character’s realism,
- the quality of the synchronization, and
- the expression that accompanies the speech.

The need for convincing mouth shapes directly correlates with the character’s natural realism. Thus, a robot can be convincing simply with a jaw flapping open and closed (or even a blinking light in synchrony with the speech), while a realistic human character requires a great number of believable mouth shapes and precise synchronization.

At MediaLab, we rely on the puppeteers’ skills to generate the lip sync—either they do the voice themselves or they follow a prerecorded voice track. A skilled puppeteer can provide convincing lip sync for almost any character, but the more human the mouth, the more difficult the task.

Puppeteers do not follow every nuance of the mouth in speech. In fact, often the character isn’t built with the capability to form some of the necessary shapes. Rather, puppeteers move the mouth to make the speech believable. First, they must follow the speech’s cadence: opening and closing the mouth in time with the syllables, sometimes only performing two syllables if three come too close together, but generally maintaining the rhythm. Second, puppeteers should respect the character’s general personality. More relaxed or less intelligent characters may leave their mouth open more often, while prim and proper matrons will have a closed mouth, biting off words before they’re finished. Although the resulting lip sync may not be technically accurate, it will be believable, which is more important in animation. Other facial gestures, such as rolling eyes

---

**Optical Motion Capture**

Motion capture without puppetry does not need to provide complex imagery in real time. The predominant motion capture technology since the mid-eighties has been optically based. Several companies market systems that use a set of three or more infrared video cameras (typically six to eight) to track infrared reflective markers strategically placed on the performer’s body (see Figure A). By comparing the position of each marker in several camera views, the systems can localize the markers’ positions in the performance space. The advantage of these systems is that a large number of markers can be used (more than 100 in some systems), and the markers can be placed on one or more performers, props, or animals. The markers are lightweight, easy to apply, and wireless. The field of motion can be as small as what’s required to track finger motion or as large as a sound stage (although not both at once).

Unfortunately, these optical systems cannot reliably track the position markers in real time due to the loss of data through occlusions. Thus, these systems require a great deal of human intervention after the motion capture session to clean up and calibrate the resulting data.

Nevertheless, optical motion capture remains an excellent tool for simulating realistic characters, especially those modeled from real actors or personalities (for example, Digital Domain’s tracking of music star Michael Jackson for his 1997 Ghosts video).
Computer puppetry combines aspects of live puppetry, acting, mime, and animation.

Computer Graphics in Entertainment

or raising eyebrows, also provide important speech cues and need to correspond well with the mouth motions.

For very accurate lip synch, optical facial capture and audio phoneme recognition systems provide alternative solutions. Current facial tracking systems track markers on the performer’s lips and move the character’s lips in the same way. The problem arises when the performer’s face does not resemble the character’s face (a bird, for example). In this case the animation looks distorted and unconvincing. In simple characters, where puppeteers only track the mouth opening and closing, the problem is less noticeable.

Audio phoneme recognition processes the voice and generates phonemes to guide the shape of the character’s mouth. The simplest real-time systems track only vowel sounds and ignore B, P, M, and F sounds. Some systems compensate by opening and closing the mouth based on volume, but again, this works only for simple and robot-like characters. More complex phoneme generation systems may exist in laboratories, but have yet to make it to the commercial market. One of the problems all audio-based systems have is that it can deduce lip protrusion to improve the system’s overall accuracy.

Unfortunately, straight lip or phoneme tracking often results in a stiff and mechanical looking character. Personality must be added on top of this for the character to be convincing. Protozoa’s character Moxy used a voice-driven lip synch process, but a rippling lip effect was added by the software, giving character to the speech and becoming an important part of Moxy’s personality. Puppetry superimposed on automated lip synch may also help.

Production issues

A production is never finished. Directors will want to change things up to the last minute and then go back and change things that were wrapped up long ago. For a system to be successful, it must be viable in a production environment. Just animating a character in real time is not enough. Important production issues for real-time puppetry, based on what we have learned at Medialab in the past six years of production, follow:

- A system must have a clean method of storing, cataloging, and retrieving all animation that was recorded in a session, and from session to session.
- The characters’ movements must be divisible. That is, you must be able to break the animation into its component channels and replay and/or reanimate any combination of channels. For example, we often will perform a sequence and then replay the facial motion, which was perfect, while reanimating the body, which may have gestured a little too early. The resulting combination of old and new then registers as the desired take.
- Time-code registration is also important. Often characters perform against a video background and with a prerecorded sound track. These are always registered with a time code. We animate against this time code, making it easy for the client to take the output of the computer (on time-coded videotape) and properly composite it with their sound track and background videos. Our system has special inputs that allow a video control room to remotely start and stop computer animation sequences, greatly aiding time-code synchronization.
- For productions that use video backgrounds, it’s important to match the camera and lighting of the real scene. For a long time we estimated the camera position based on still images. This worked to a certain extent, but often was a lengthy, error-prone process. Recently we’ve developed a method that automates camera calibration and lets us quickly and accurately duplicate the camera position and focal length of video backgrounds. We still match lighting by hand.
- The director must understand the technology. An art form, computer puppetry combines aspects of live puppetry, acting, mime, and animation. Often directors approach the set with preconceived notions from one of these traditional disciplines. Without understanding that computer puppetry is at once all and none of these, they try to force the medium into something that it’s not. With patience and experience, they discover the best ways to use the strengths and limitations in the system, producing outstanding animation quickly and efficiently.

Conclusion

With improvements in real-time computer graphics machines and motion-capture technologies, computer puppetry has become a viable tool in character animation. Currently, computer puppetry is restricted to those who can afford large machines and expensive motion-capture equipment. As machines get cheaper and faster, computer puppetry systems will become widely available, just as computer animation is today. However, technology comprises just one important aspect in this medium. As in any art, the final product’s success depends on good puppetry, acting, design, and directing talent.

Acknowledgments

Credit for much of the work I describe goes to the dedicated research and development, production, and performance teams of Medialab, without whom the field of computer puppetry would be less rich than it is today. I
especially want to thank the directors, puppeteers, and performers who have contributed to this article by generously sharing insights into their art, and the Medialab research and development team with whom I’ve had the privilege of working for the past six years.

References

David J. Sturman is the former director of Medialab’s research and development group. His technical interests include real-time animation, human-computer interaction, and real-time I/O devices. He earned his BS in computer science from the Massachusetts Institute of Technology and has a PhD in computer animation and virtual environments from the MIT Media Lab.

Contact Sturman through Medialab, 104, av de President Kennedy, 75016 Paris, France, email david@mlab.fdn.org.

ArtSci 98
April 4-5, 1998
The Great Hall
Cooper Union, New York City

Sponsored by Discover and Leonardo Journal, this two-day symposium will gather scientists, artists, educators, writers, and science and technology professionals to discuss how science and technology affect each other. ArtSci 98 aims to dispel misconceptions, confirm suspicions about the advantages of multidisciplinary learning, and take an affirmative step in of narrowing the personal and professional schisms between scientists and artists, and the general public.

Scheduled panel topics include creativity, the transformative functions of science and art, the dialectics of art and science, the artist and scientist in society, technology and creativity, problem solvers and mythmakers, science in art and art in science, and literal and actual collaborations.

For more information contact
Cynthia Pannucci, founder and director,
Art and Science Collaborations (ASCI)
P.O. Box 358
Staten Island, NY 10301
http://www.asci.org

IEEE Computer Graphics and Applications 45