From Linear to Interactive Animation: How Autonomous Characters Change the Process and Product of Animating

BILL TOMLINSON
University of California, Irvine

There are significant differences between the art of animating for linear media such as film and video and the art of animating for interactive media such as computer and video games. In particular, these differences arise from the shift from linear characters to autonomous interactive characters. This article describes differences between linear animation and interactive animation in several areas of character design – character intelligence, emotional expressiveness, navigation, transitions among animations, and multi-character interaction. These differences provide insight into the processes of both forms of animation and the final products that they create, and may provide a starting point for linear animators interested in becoming familiar with interactive animation.

Categories and Subject Descriptors: I.3.7. **[Computer Graphics]**: Three-Dimensional Graphics and Realism—*Animation*; I.3.6 [**Computer Graphics**]: Methodology and Techniques—*Interaction techniques*

Categories and Subject Descriptors: I.3.7. **[Computer Graphics]**: Three-Dimensional Graphics and Realism—*Animation*; I.3.6 [**Computer Graphics**]: Methodology and Techniques—*Interaction techniques*.

General Terms: Design, Human Factors

Additionak Key Words and Phrases: Animation, Production, Games, Interactivity

---

Animation is the art of breathing life into things—creating the appearance of motion, behavior and personality in characters and other entities. When most people use the term "animation," they think of Bugs Bunny, Buzz Lightyear, and other traditional animated characters. In the last few decades, animation has also become significantly associated with interactive media such as computer and video games. While there are many similarities between the forms of animation that generate linear and interactive products, there are also profound differences. These differences are not necessarily obvious to the consumers of the two forms, who ultimately experience both as images flickering across screens. This article offers an account of some of the ways in which the interactive animation process is different from (and usually harder than) more linear forms of animation. Considering the differences in the processes of these two kinds of animation may contribute to an understanding of the differences between the aesthetic forms that they produce.

At a very high level, linear and interactive animations share a common goal—to create an engaging experience for the viewer/player. In comparing film and interactive mediated environments, Marsh [2003] offers three levels of experience that produce engagement: voyeuristic (the joy of seeing the new and the wonderful), visceral (thrill of

---

Author's address: ACE (Arts Computation Engineering) program, 430A Computer Science Building, University of California, Irvine, Irvine, CA 92697-3425; email: wmt@uci.edu

Table I. Summary of Five Major Differences Between Linear and Interactive Animation

Topic	Linear animation	Interactive animation
Intelligence	The intelligence and behavior of characters are determined by screenwriters, storyboarders, and animators prior to the audience experiencing the work.	The intelligence and behavior of tcharacters are generated by a computer program in real time.
Emotional expressiveness	Animator controls emotional state exhibited by characters in each shot. Each action a character takes is inextricably tied to an emotion.	Animator creates dynamic emotional ranges explored during game play based on a range factors. Emotions may be layered on top of actions, and controlled independently.
Navigation collision avoidance	Characters only run into things when the animator wants them to.	Characters may accidentally collide with others and with objects in their world. Characters need a mechanism for avoiding collisions dynamically and coping if they occur.
Transitions	Since sequence of events is fixed, transitions between small chunks of animation are controlled by the animator, and may be long or short.	Since the sequence of events is variable, cycles must be kept short or they cause a character to take too long to switch to a different action. (If this is solved by animations not running to completion, characters may exhibit discontinuous motion.)
Multi-character interaction	Two characters may be animated simultaneously.	Enabling two characters to robustly engage in expressive close contact (fighting, affection, etc.) is an unsolved problem.

spectacle and attractions), and vicarious (transfer of emotion through another person,being or object) [Marsh 2003]. The characters in both linear and interactive animation lie at the heart of these three levels of experience, and are particularly central to the third. Accordingly, the differences between the characters in these two forms of media are central to an understanding of the differences between the media themselves.

This article discusses five main differences between linear and interactive characters, including the locus of intelligence in the characters, how the characters are made to express themselves emotionally, how the characters navigate around their world, how the characters transition between actions, and how characters interact with each other. Table I offers a summary of these key differences. Each item gives a brief summary of the topic from the perspectives of linear animation and interactive animation. The brief descriptions in this table are necessarily generalizations; there are exceptions to each of these topics, and active research in all of these areas. These issues will be addressed more fully in the sections that follow the list.

By directly addressing the differences between the two kinds of animation, this article seeks to understand the capabilities of the two media. Many books have been written on the topic of traditional film animation (e.g., [Laybourne 1998; Thomas and Johnson 1981]. In addition, several books have been written on animating for video games (e.g., [Steed 2002]). This article seeks to understand the ways in which the processes of making characters for linear and interactive animation affect the content that these two forms produce.

1 EXAMPLES OF LINEAR AND INTERACTIVE ANIMATION

In order to establish a common background, this article first gives a brief description of the production pipelines for linear animation and interactive animation. In addition, this section provides brief descriptions of two animated projects by the author and his collaborators: one linear and one interactive.

1.1 Linear

The linear animation process may take on a number of forms. The most widely known is the production pipeline used to create animated narrative feature films such as Dreamworks' *Shrek* or Disney's *The Lion King*. While different studios employ different pipelines, a typical production might use the following steps:

- a screenplay is written that prescribes the ordering of events in the final film;
- a series of storyboards is drawn that depicts each event;
- a soundtrack is recorded using human actors;
- numerous scenes are animated simultaneously to match the storyboards and soundtrack with different animators working on different characters and scenes;
- the film is edited into the correct order, and revisions are made to both soundtrack and animation;
- the final picture and sound are printed to celluloid film;
- the film is distributed to theaters and shown to large audiences.

Before a complete film can be created, a great deal of coordination must occur among producers, directors, animators, actors, sound designers, technical directors, editors, publicists, projectionists, and many others. Different animation techniques such as cel animation, 3D computer animation, stop-motion animation, or cut-out animation add additional constraints to this process.

While most animated features adhere to some variant of the above process, independent animators use smaller crews and a wider range of production processes to create their films. As an example of one production pipeline, in 1995-6 the author made

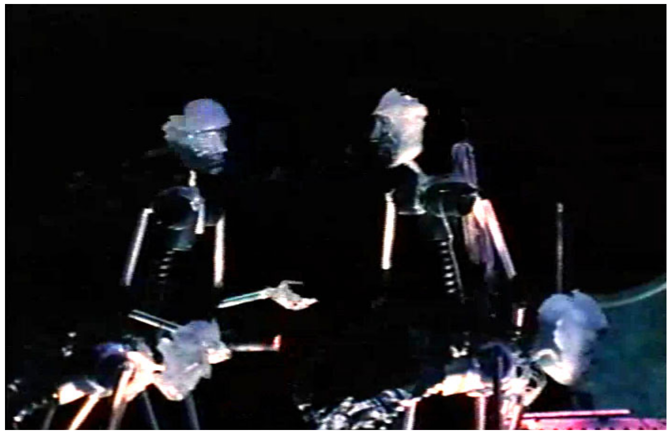

Fig. 1. Still frame from *Shaft of Light*, 1996.

a stop-motion animated film called *Shaft of Light* (see Figure 1). This production began with a short script, which was produced as a stage play using human actors. Thereafter, the film was storyboarded and the characters were constructed from brass tubing, wire, and hot glue. Once the characters were complete, there was a 12-day film shoot during which approximately 14 minutes of raw footage (~20,000 images) were generated. In the 4 weeks following the shoot, the raw footage was developed and edited down to its final length of 8 min. 46 sec. Then the actors recorded the voices of the characters and the composer created all of the composition and sound design. (This order of production is quite different from the traditional animated feature process described above, where the voice recording is done first and the animation is made to match.) Finally, a 16mm celluloid print was made, and copies of the film were distributed to film festivals. An online version of the film may be seen at the following web address :

http://www.ics.uci.edu/~wmt/movies/ShaftOfLight.mov

1.2 Interactive

Large-scale interactive animation productions, like those used in making character-based video and computer games, employ a very different set of processes than those described above for linear animation. The game industry is significantly younger than the film/animation industry, having been in existence for less than half as long. *Spacewar* in 1962 could be considered the first "real computer game" [DeMaria and Wilson 2003, p.12], whereas filmed animation developed as a medium in the nineteenth century [Bendazzi 1995]. Due to its relative youth, the game industry is still undergoing rapid change in its production processes.

Despite the wide variety of production pipelines in place in game companies, there are certain processes that unify many current 3D animated, character-based computer and video game productions. Interactive animations are conceptualized by a small team of designers, artists, software engineers and producers. A team of software engineers creates a game engine that will serve as a computational platform for the game (or they decide to license and reuse an existing engine). The animated characters are modeled and rigged (that is, given a computationally sound infrastructure so that the animators can produce content that may be integrated into the game) by a group of artists, animators, and technical directors. Animators generate the characters' behavioral repertoire, using a 3D animation program, motion capture technologies, or other custom tools. Software engineers create the code that controls the behavior of the characters and the dynamics of the virtual world and connects the interface elements (buttons, etc.) for game play. This kind of game often consists of a group of levels or areas (similar to scenes in a play or movie); these subsections are created by level designers using tools designed by software engineers and technical directors. Lighting designers, virtual cinematographers, and audio directors contribute additional elements to the game. Finally a team of quality assurance testers plays the game thoroughly to find bugs, which are then sent back to the software engineers, designers, and animators. Once the game is complete, it is distributed to its players through stores and online retailers.

Similar to the independent animators described above, there are numerous smaller teams making independent interactive animations as well. One example is the production of a 3D animated interactive installation entitled *AlphaWolf* [Tomlinson et al. 2001] (see Figure 2) by the author and other members of the Synthetic Characters Group at the MIT Media Lab. The installation featured a virtual pack of wolves that participants could direct by howling, growling, whining, or barking into microphones [Tomlinson et al. 2002]. This production started from a code base for building interactive characters and virtual installations that has been in development and use since 1997 [Blumberg 1998; 1999]. In the winter and spring of 2001, a prototype of the installation was developed with simple 3D models, animation and programmed behavior. In the summer of that year, five programmer/interface designers, two animators, and a sound designer worked in the same room for approximately three months to build the final installation. The development process involved very close collaboration among the creators and a tight feedback loop between the programmers, interface designers, animators, and sound online at designer. This interactive installation premiered at SIGGRAPH 2001, and has been exhibited at a number of other venues since then. For a better sense of the installation, please see the episode of the PBS television show *Scientific American*

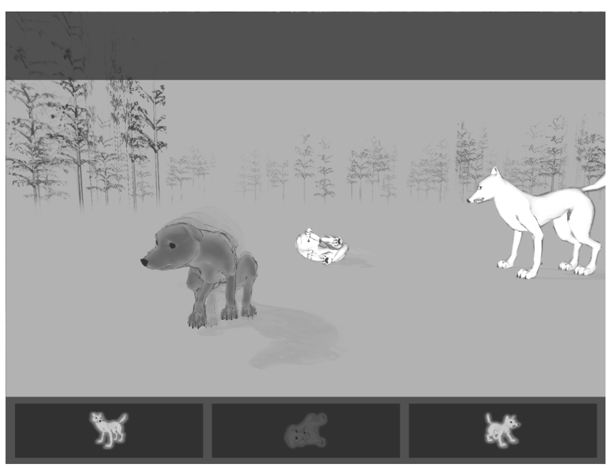

Fig. 2. Still frame from a multi-player interactive installation entitled *AlphaWolf*.

Frontiers, hosted by Alan Alda on 10/22/02, available http://pbs-saf.virage.com/cgi-bin/visearch?user=pbssaf&template=template.html&query=Alpha+Wolf&category=0&viKeyword=Alpha+Wolf

2 DIFFERENCES BETWEEN LINEAR AND INTERACTIVE ANIMATION

Linear and interactive animations employ very different animated production processes and result in quite different final products. One of the main differences between these two formats is the way in which they deal with characters. Numerous terms, such as synthetic characters [Kline and Blumberg 1999], synthespian [Kleiser 1996], and lifelike computer characters [Ball et al. 19997] have been used to describe a range of similar entities. The process by which the characters are created and the final form that they take point to profound differences in the craft of animation employed. This section describes several specific differences between the creation of linear characters and interactive/autonomous characters. In addition, each section offers suggestions for linear animators interested in becoming more familiar with interactive animation processes.

2.1 Intelligence

One of the key differences between linear and interactive animation is the locus of the character's decision-making process. In linear animation, the behavioral modeling happens inside the human head; animators, writers, and directors intuit the behavior that the characters should undertake. These people then work together to transfer this behavior to the audience through images and sound. In interactive animation, the modeling of intelligence that will be seen by an audience is moved out of the heads of the creative team; it happens inside a machine instead. The creative team designs the system to generate behavior. The computer program acts as an intermediary between the creators and the audience. But for the program to perform this task, it is necessary for the behavioral decisions and parameters to be specified explicitly. The act of moving from linear to interactive is more than just a set of small tweaks or a change in complexity. Rather, it entails taking an intuitive process (the way an animator or animation director decides to have a character act) and making it explicit (so that a computer program can make the same decisions on-the-fly).

The computer/video game industry has expended great effort to make compelling real-time characters. Many of these games have been exceedingly successful financially; *Sims* sold over 28 million copies and related expansion packs worldwide [Yi 2004]; the industry as a whole generated $30 billion in sales [Gaudiosi 2003]. Many of these games showcase cutting-edge research in graphics, character motion, AI, and other areas. Nevertheless, game characters are often weak in expressiveness and believability. The content of computer games has been a hotly contested issue for a number of years (e.g., [Anderson and Bushman 2001; ESA 2003]).

When people trained in linear animation techniques move into interactive animation, this new medium's inability to fully specify the behavior of the characters in every frame may be challenging for them. Rather than specifying the characters' exact behavior, interactive animators and character designers instead develop suites of behaviors and the rules that will decide when these behaviors should occur. Unless the animator is interested in learning how to program, the process of making interactive characters will necessarily be collaborative with computer programmers, who will encode these rules in the characters' software.

In figure drawing, the principle of "negative space" is often employed to help guide the draftsperson. This principle suggests that the draftsperson should consider the shape of the blank space around the object being drawn, rather than just the space occupied by the object being drawn. The rationale for this principle is that people are biased to perceive objects in a certain way – heads are roundish, flowers have petals radiating out from a central point. When trying to draw an object, the draftsperson may fall into the trap of drawing the generalization in his or her heads, rather than drawing the form that actually exists in the world. Attending to the negative space, whose shape has no pre-existing impression in the draftsperson's head, will allow him or her to reproduce reality more faithfully.

This notion may be extrapolated to the behavior of animated characters as well. A character is defined not just by what it does, but by what it fails to do or what it is unable to do. In a linear animation, the behavior of each character may be witnessed in is entirety; a linear animator can see the actions that a character takes and the actions that a character does not take. By thinking about the character's "negative behavior space," the animator may gain another perspective on the character as a whole.

In interactive animation, however, viewing a character in its entirety is not nearly as feasible. It is easy to see what a character does by simply playing the game as it was designed. But comprehending the full extent of its "negative behavior space" is very difficult. In an effort to address this challenge, game companies employ small armies of play-testers whose job is to play the game for hours on end and document any problems they see. These play-testers work to insure that the characters never exhibit behaviors that might compromise their appearance of intelligence and realism. If it is not possible to remove all of the bugs from a certain subset of a characters' behavior, it may be better to remove that suite of behaviors than to allow the character to break from time to time. To quote Mark Twain, "It is better to sit in silence and appear ignorant, than to open your mouth and remove all doubt."[1] Better mechanisms for comprehending the scope of a system's behavior could help with the problem of negative behavior space in autonomous characters.

During an interactive animation production process, animators and programmers produce the raw material for believable behaviors and the rules for connecting them, while linear animators produce the behaviors themselves. This difference can be unsatisfying to a linear animator; it is not nearly as clear when the work is complete, and it is more difficult to document the work that has been done (e.g., for a show reel.) However, for an animator it can be an interesting new undertaking to think through a character in sufficient depth to determine all possible actions and reactions that the character could take.

The broad goal of both linear and interactive characters may be similar – interesting characters engaged in intelligent (or intentionally not-so-intelligent) interactions with each other and with their world. However, the specifics of the process of encoding a characters' behavior and intelligence biases the kind of characters that are usually created. Interactive characters are less likely to exhibit an abundance of behavioral complexity due to the challenges of programming that behavior and insuring that all of the behavioral patterns work well together. In addition, interactive characters are prone to repeating their actions and getting stuck in behavioral traps (e.g., trying unsuccessfully to do the same action over and over again). These shortcomings of interactive characters, though, are offset by the potential for these characters to be more believable and viscerally compelling due to their ability to react in real time to the player's actions.

[1] This quote is Twain's story, *The Tragedy of Pudd'nhead Wilson*. [Twain 1893].

2.2 Emotional Expressiveness

In drawn animation, key frames define the characters' positions at various points in the story and in-between frames (or "tweens") interpolate between the key frames. In stop motion, cut-out and other forms of animation, the animator must pose every frame independently. Either way, when animating for a linear product, the animator is able to control the expressiveness of every character at every frame. Giving characters expressive behavior is one of the key principles of animation [Thomas and Johnson 1981]. Even when the broad strokes of an animation are determined by a computational system, as in the pitched battles in the *Lord of the Rings* trilogy or the wildebeest stampede in *The Lion King*, it is still possible to tweak certain key positions or actions if the scene does not satisfy the animators. Ultimately, a human has the last say.

This ability to tweak is especially important in 3D computer animation. In computer animation it is relatively easy to duplicate a cycle and then revise it slightly. The spine may be curved a bit more to show deepening sadness, a head turned to one side to break the repetition. One of the strengths of 3D computer animation is the ability to reuse pieces of animation in modified form.

When animating for interactive products, however, it is not possible for the animator to see the final form that the animation will take in every situation. The animator can play-test the game and get a broad sense for the way an animation will look when it is finally rendered and make revisions based on that process. However, due to the very broad range of potential interactions among interactive characters and their worlds, there can be no final round of hand- tweaking by the animator to make sure that every element of the scene works together and that every action is as expressive as desired. This is not to say that interactive animation is doomed to be repetitive, emotionless, and boring; there are a number of ways in which a system may be engineered to work around these issues. One possibility is to have multiple animations for each action. Rather than having just one animation for a given behavior, a character may have several subtly different alternating variants. Another possibility is to add procedural noise to the action to give it a more vital quality [Perlin and Goldberg 1996]. Various other researchers have also generated procedural emotional modifiers for animation (e.g., [Amaya and Bruderlin 1996; Unuma et al. 1995]).

Academic researchers have looked closely at the topics of expressiveness and believability, focusing most frequently on interactive virtual humans (e.g., [Gratch et al. 2002]). Perlin and his colleagues have done pioneering work in creating synthetic actors (e.g., [Perlin and Goldberg 1996]). By working closely with the natural style of their characters' motion, they have created virtual characters that move and interact very naturally. Blumberg and his colleagues have contributed in several areas including action-selection [Isla et al. 2001], learning [Blumberg et al. 2002], and interactive cinematography [Tomlinson et al. 2000]. Cassell and her colleagues built virtual humanoids with the ability to express themselves like real people, in particular as embodied conversational agents (e.g., [Cassell et al. 1999]). Bates and his colleagues have created computational characters who appear lifelike and are able to interact with people in real time [Bates et al. 1992; Reilly 1996]. More recently, Schaub et al. [2003] and his colleagues have presented research on empathic characters. Robots have also been endowed with characteristics that allow them to engage people in expressive ways [Breazeal 2000; Velasquez 1998]. Many other researchers have also contributed to the creation of compelling synthetic characters, e.g., [[Assanie 2002; Funge et al. 1999; Hayes-Roth et al. 1995; Hodgins and Pollard 1997; Paiva et al. 2001; Thalmann, et al. 1997]. Liu and Popovic [2002] presented a method for producing realistic character

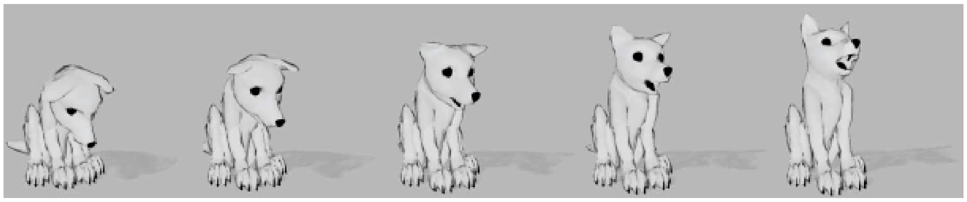

Fig. 3. An expressive emotional range generated by *AlphaWolf's* motor system. The far left and far right poses were created by our animator; the three intermediate poses were generated by the computational system.

motion. Commercial games such as Maxis's *The Sims2* are making substantial advances in expressive characters as well. Several start-up companies have undertaken to develop expressive real-time characters, including Ingeeni Studios, Zoesis. and Nearlife.

The mechanism used in the *AlphaWolf* project to give the characters expressiveness is to have emotion layered on top of the action. The animators made multiple example animations (e.g., "dominant walk" and "submissive walk") in 3D Studio Max, which were blended together in real time by the characters' motor systems while the installation was running [Downie 2001]. In addition to smoothing the transitions between one action and another, the motor system allows different emotional styles of an animation to be blended on the fly, thereby giving the characters a continuous emotional range (see Figure 3). This structure, with an action and an emotion layered together, is derived from the work of Rose [1999].

Despite the range of work done to make emotionally expressive interactive characters, it is not clear that they have the same core need for emotion that linear characters do. While movies often focus on empathy (identifying with the emotions of others), interactive animation places much greater emphasis on agency (having an impact on events). It could be argued that the most important feature of interactive characters is responsiveness. Nevertheless, since emotional responsiveness is important, the ability for interactive characters to manifest emotions remains important. The relationship between agency and empathy has also been discussed by Pearce [2004].

The process of making emotionally expressive characters in an interactive domain is different from making them in a linear domain, for many of the reasons described above regarding intelligence. The neurologist Antonio Damasio might offer that this is because emotion lies at the core of intelligent decision-making [Damasio 1994]. To understand the emotions of an interactive character, animators may need a more principled and explicit understanding of emotion in general. A great deal of work has been done in biology, psychology, and other fields on the phenomenon of emotion, both in general and as it relates to computational systems (see [Picard [1997] for a review). A principled approach to emotion requires additional effort to conceptualize and set up, but ultimately allows the system to become more complex without becoming unnecessarily complicated. Different models of emotion – such as a categorical model [Ekman 1992] (where emotion is broken down into different categories such as happiness, sadness, anger, fear, surprise, and disgust) or a dimensional model [Mehrabian and Russell 1974] (where a character's emotional state is represented as a moving point in an emotion space with axes such as pleasure, arousal, and dominance) – may be appropriate for different kinds of characters. Ortony [1988] offered a collapsed model of the OCC model of emotion [Ortony et al. 1988] designed specifically to produce believable emotional agents.In terms of process, interactive animators and programmers build the raw materials and rule sets for emotional behavior, rather than the emotional behavior itself. This requires a more explicit understanding of the characters' emotional capabilities. For example, rather than simply having one character scowl at another (as might occur in an animated film), it becomes necessary to specify the conditions that should cause the scowl. The process of interactive animation therefore requires animators to understand their characters' behavior more fully, and in a wider range of possible circumstances, than in linear formats.

The effort necessary to build an emotionally expressive interactive character lends itself to a different kind of expressiveness than the process of linear animation. Interactive characters are more likely to be broadly expressive, i.e., able to express simple emotions in a variety of appropriate contexts. Linear characters, on the other hand, are deeply expressive; they encounter very few situations, and are much more effective at "getting it right." As techniques are developed to create emotionally expressive interactive characters, interactive characters may begin to have the depth of linear characters without sacrificing the breadth necessary to be interactive in a significant way.

2.3 Navigation/Collision Avoidance

In a linear animation, the animator can be in complete control of all characters at all times. If two characters run into each other in a movie, it is because the animator decided that they should. Getting characters to collide in a plausible way may be an animation challenge; but collisions occur only when the animator has decided that it is the appropriate action for this point in time.

In interactive animations, characters run into each other for various reasons, for example, players due to inexperience, mischief, or the design of the game, may steer their characters into objects and other characters. Also, computer-controlled characters may not have completely effective autonomous behavior, and may end up running into things. The characters' animation needs to be able to cope with the potential collision in some plausible way. A great deal of work has been done in the robotics community on navigation and collision avoidance (e.g., [Siegwart and Nourbakhsh 2004]). Navigation research has also been done on autonomous characters [Brogan and. Hodgins 2002]. Craig Reynolds has produced a number of simulations that show a range of collision, steering, and related behaviors in action [Reynolds 1999]; applets showing these behaviors are available on his web site: http://www.red3d.com/cwr/steer/.

Collision avoidance behaviors often require a set of animations that allow for fine-grained movement and orientation. For example, a character may need animations for "step back," "turn left," "turn right," and other related actions. Alternately, there may be procedural mechanisms in place for the characters to avoid walking into each other. In the procedural case, it is important for the animators to provide feedback to the programmers to ensure that the character stays "in character," even when undergoing the procedural manipulation. Ultimately, the animator should take responsibility for the quality and consistency of the characters' motions. Collision avoidance is an area of active research in fields such as crowd modeling [Musse et al. 1998].

The process of making interactive characters able to navigate and avoid each other may be tedious for a linear animator, since they come from a medium where unintended collisions are unusual. Navigation and collision avoidance may seem like problems caused by the limitations of artificial intelligence; but this process may also be seen as an opportunity to build the personality of the character. How does the character find its way from place to place? Does it forcefully march toward its destination, or sneak through the shadows? When a character runs into another character, is it apologetic or indignant? These kinds of actions have potential as moments for character development.

The prevalence of navigation and collision avoidance behaviors in interactive characters means that navigation and collision avoidance have different relative proportions in the behavior of interactive characters than in linear characters. Linear characters may spend more time emoting, talking or reacting, since cuts reduce the need for navigation and deterministic blocking eliminates accidental collisions. Navigation is one of the main forms of control that players can exert over an interactive character – steering it from place to place. As such, it provides an important opportunity to layer expressive content over user-directed actions.

2.4 Transitions

In addition to expressing its emotions during specific actions (such as walking), a character may also be expressive during a transition from one action to another. An expressive transition is important because it helps demonstrate how the character feels about switching to the new action. In a linear animation, an animator may consult the script or storyboard to find out what action will be coming next. Therefore, the transitions from one animation to another may be done by hand, may be as long and expressive as is appropriate for the scene, and may blend subtly into the actions coming before and after.

In interactive animation the situation is more complicated. Characters may need to transition among fixed or blended animation cycles, or transition from an animation cycle to a procedural animation sequence [Liu and Popovic 2002]. Since it is difficult to predict the sequence of events that a character may take and each action may need to commence at any time, the mechanism for transitioning between animations must be robust and rapid. For example, imagine a character with four animations: STAND_CYCLE, STAND_TO_WALK, WALK_CYCLE, and WALK_TO_STAND (see Figure 4). The STAND_CYCLE is a "moving hold" in which the character stands still, but shows a small amount of bodily motion so that it still looks alive. The WALK_CYCLE animation is a full walk cycle, starting with the right foot forward and ending with the right foot forward. STAND_TO_WALK is a transition animation from the last frame of the STAND_CYCLE to the first frame of the WALK_CYCLE, and

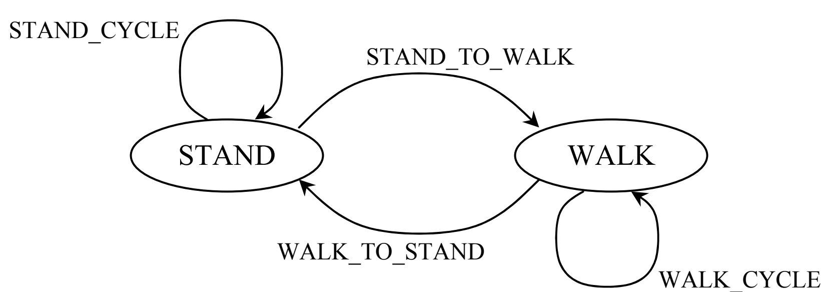

Fig. 4. A finite state machine diagram showing the relationship between the STAND and WALK poses and four animations: STAND_CYCLE, STAND_TO_WALK, WALK_CYCLE and WALK_TO_STAND.

WALK_TO_STAND is a transition from the last frame of the WALK_CYCLE to the first frame of STAND_CYCLE. If the character has just started playing WALK_CYCLE and suddenly needs to stop, it might do one of two things. The first possibility is that it will finish its current WALK_CYCLE and then play WALK_TO_STAND, before finally ending in STAND_CYCLE. This character will travel forward quite a distance (up to a maximum of just less than one full WALK_CYCLE-length plus one full WALK_TO_STAND-length) and will take a relatively long time to do it (perhaps more than a second to completion, which is a long time in an interactive experience). The second possibility is that the character, upon deciding that it needs to stop, will immediately begin STAND_CYCLE. In this case, there will be no time lag, but will result in a motion discontinuity as the character instantaneously moves parts of its body into a completely new pose.

There are several ways to make this system as effective as possible, each of which requires a different kind of effort. One way is to make sure that the WALK_TO_STAND transition is short, so that it causes a small a lag in time and distance as possible. This option still suffers from a significant amount of lag, if only from the WALK_CYCLE itself. A second way is to give the system the procedural ability to interpolate between two poses, so that when the character needs to stop it will simply interpolate from its current pose into the desired pose. This option suffers from the character's poorly animated and physically implausible actions, in exchange for speed of execution. We could attempt to cover the weak animation with some sort of special effect, such as motion blur or fog, but then we are hiding from the true problem and the character loses plausibility. A third way is to ask the animator to do more work. If the character has an additional animation, i.e., WALK_WITH_LEFT_FOOT_FORWARD_TO_STAND, the lag can be cut nearly in half, since WALK_CYCLE no longer has to run to completion. The addition of an animation decreases the "level of granularity" of the character, as it allows for a finer level of behavioral control. The downsides are that it takes the animator away from other animation tasks, that it requires more programming to integrate the animation, and that it requires more disk storage space and load time to include the new animation in the interactive work.

These possibilities point to one of the recurring themes in both the process and product of interactive animation: the trade-offs among animation quality, animator's work, programmer's work, storage space, and speed of execution. Many problems may be solved by increasing one or more of the above four elements. One of the great challenges for the producers and designers of games is, given the human and technical resources, how to optimize the production so that the characters look as good as possible and are as interactive as possible.

The inclusion of emotion in interactive characters makes this problem even harder. For the core actions to have an expressive emotional range, it is necessary for transitions to have a significant emotional component as well. In fact, transitions between animations often occur at moments of emotional change, and thus should be very expressive. For example, imagine that a character is walking happily, but then sees another character it dislikes. It will stop walking and rapidly shift its emotional state. On a continuous emotional scale of happiness from 0.0 to 1.0, the character might be at a 0.9 happiness at the end of WALK_CYCLE, but down at a 0.1 happiness by the beginning of STAND_CYCLE. The reason for the transition is also the reason for the emotional change, and WALK_TO_STAND will need to be able to exhibit the relevant emotional transition.

For animators accustomed to linear work, the need to break up action into discrete chunks and connect them to each other through fixed transitions may be seen as a substantial limitation. It is significantly more challenging to create a fluid and natural-looking series of actions in interactive than in linear animation. A hybrid approach of example-based and procedural animation techniques may be helpful in creating interactive characters with smooth transitions. Regardless of whether example-based or procedural animation techniques are in place, the approach that an animator takes is often more technically formal than the process of linear animation.

When working with programmers (for example, to develop a convincing transition or navigation mechanism) the animator takes on an editorial role as well as a creative one. The programmers' programs will produce animation content; the animator's job is to make the content aesthetically strong and consistent with other parts of the characters' motions. Due to this collaborative process, it becomes more important that animators be able to express why a character should animate a certain way, rather than simply being able to make a character move in a certain way.

The impact of computational transition mechanisms on interactive characters is subtle but significant. It is often easy to see the animation cycles and the transitions that connect them when playing a game (e.g., Lara Croft's various runs and jumps in *Tomb Raider*). Due to this formulaic behavior, and the control that players may exert over it, interactive characters may lose much of their separation from the player. This lack of separation may increase the players' feeling of agency, but may compromise the empathy they feel for the characters.

2.5 Multi-Character Interaction

In a linear animation, two characters may be animated simultaneously. Because the animator knows precisely where the characters will start (i.e., the last frame of the previous scene, or, after a cut, a new position altogether), he or she may animate the two characters to engage in close physical contact. They may dance, fight, embrace, wrestle, or do any of the actions that we might expect from two or more characters in the same space. Doing a linear animation of two characters is more difficult than animating the two characters separately. Warner Brothers' cartoons get around this problem by representing fast-paced or multi-character interactions as a cloud of dust with occasional limbs sticking out of it. This phenomenon is referred to as a "cloud of altercation" in the Cartoon Laws of Physics, a widely circulated humorous newsgroup posting. Nevertheless, when creating a linear animation, it is possible to animate two or more characters engaging each other in a wide range of ways.

In an interactive animation, interactions between characters are much more difficult. For example, consider that one character wants to hug another character. In order to hug, the character must first move to a location in front of the other character and orient to be facing that character. This fine motor control is a challenging computational problem. While it is certainly possible to get a character to the correct position (for example by calling the "set position" and "set orientation" functions on the character), it is difficult to get the character to go there in an expressive and physically plausible way.

Even if the characters do manage to find their way to an appropriate starting point, the challenges continue. If they are in exactly the correct configuration, then it is possible for them to run two matched example animations that have been carefully crafted to synchronize well. This synchronization requires the characters to launch the animations at precisely the same time, a difficult timing task. If the characters are not in exactly the correct configuration, but nevertheless in a fairly close arrangement, they will need to be able to change their animations dynamically to adapt to the subtle variations in their starting arrangement. Failing at these registration and animation tasks may cause

Fig. 5. Two autonomous characters make eye contact.

characters' limbs to pass through each other and a whole array of other problems to ensue. Fighting games often solve this problem by allowing characters a fairly limited number of constrained "moves" such as kicks and punches, each of which may be coupled with a reaction animation from the target character. The close registration between characters that is necessary for them to interact in an unconstrained way is an open area of research in interactive animation. Despite these difficulties, it is important that characters be able to look at each other (see Figure 5) and interact closely if they are to appear to engage in social and emotional relationships.

Significant advances are occurring in procedural animation techniques for enabling interactions between characters. For example, the Endorphin motion synthesis system by Natural Motion (http://naturalmotion.com/pages/demos.htm) offers a mechanism by which characters may interact in physically plausible ways. Sports games such as Madden NFL Football use procedural animation to great effect, as close contact is an important part of many sports. These games may treat the two separate characters as a single animated entity for the duration of the tackle or other contact. Incorporating procedural techniques with hand-animated techniques in the same interactive characters is an additional research challenge, as a blended approach may be superior to either purely example-based or purely procedural animation systems.

The process of enabling interactive characters to engage in close interactions with each other is a challenge that combines both animation and programming. Since, ultimately, the product will reflect on the quality of the characters' animation, it is important that animators take an active role in this process. Watching for moments where the animation does not work well, and understanding enough about the system as a whole to help determine what is causing the problem, can be an important part of an interactive animator's job, especially in a small production.

The limitations of physical contact in multi-character interactions may skew the content that can be delivered in current interactive animated projects. For example, games with guns (first-person shooters, etc.) are relatively easy in this regard, since gun combat does not require the same level of contact as hand-to-hand contact. As computational techniques for multi-character interaction become more prevalent, there will be a significant broadening of the kinds of interactive experiences that may be created.

2.6 Other Differences

The items described above may be among the most evident differences between linear and interactive animation, but there are many others. This section gives brief mention to several of the additional differences.

Cinematography is very different in the two forms of animation. Linear cinematography is predictable, so animations only have to look good from one angle; interactive cinematography is dynamic, so animations have to look good from any angle. In linear animation, the position, orientation, size, and other features of cinematic elements may be adjusted ("cheated") to make a scene look better. In an interactive world, cheating is much more difficult, since it may compromise other networked views of the world, and so on. In a linear animation, the characters' faces are more important than the backs of their heads, since the face shows a lot of emotion. In interactive animation, main characters often face away from the camera, so that players are able to see where the characters are going, rather than seeing the characters' faces. Finally, in a linear animation, lighting design may be tailored to specific characters, sets, and other elements in the world. Work in this area has been undertaken [El-Nasr and Horswill 2004;Tomlinson 1999], but it is computationally expensive and is still an open research topic. Building interactive cinematography systems that effectively exhibit the actions of the characters while enabling players to interact is an ongoing area of research (e.g., [Bares and Lester 1997; Drucker 1994; El-Nasr and Horswill 2004; He et al. 1996; Tomlinson et al. 2000]).

The timing and narrative processes of linear and interactive animations are also different. In linear animation, time is frequently lengthened, condensed, or reversed by the editor. Script resembles a screenplay, with characters, dialogue, and so on. The script and the animators drive the story during production. Each time an audience member sees the animation, the images will be the same. In an interactive animation, time is largely continuous, with the player contributing significantly to advancing the story at runtime. The narrative path that the player witnesses may be different each time he or she interacts with the system. Many researchers have explored ways of understanding interactive narrative (e.g., [Murray 1998; Pausch et al. 1996]).

The team an animator works with in a linear animation production is quite different from the team used in an interactive production. Linear animators collaborate with musicians, producers, color timers, technical directors, and an array of others; interactive animators work with programmers, interface designers, level designers, sound designers, and others. While there is certainly an overlap, even people with the same job title in the two media may perform very different tasks.

The tools that an animator uses in the two media are also different. Linear animators may use a wide range of digital tools, or their finished product may result from pencil drawings on simple sheaf of paper. Interactive animators deal primarily with 3D animation software such as 3D Studio Max or Maya, and are also more likely to interact with motion capture rigs, in-house digital tools, and other technologically complex systems. The tools that interactive animators work with are often very complex, requiring significant knowledge of math and computer science principles. Again, there is an overlap in these tool sets, but the differences are significant.

Interactive animators need to think about interfaces and controllers and how their characters will relate to those features, whereas linear animators do not need to concern themselves with interfaces. Audiences for linear animation usually sit in a theater or in front of a television. How players of interactive animation engage the work may impact how the animator creates characters and their behaviors. Games use an assortment of input devices, including joysticks, game pads, and keyboards, to enable players to engage with the characters. Research has been done on the intersection of interactive animation and human computer interfaces. Johnson et al. [1999] offers a stuffed animal interface as an appropriate level of control for a synthetic character; Marks [2001] provides a robust computer vision system as a simple and intuitive interface to games. These and other efforts to build compelling interfaces demonstrate an added challenge that does not face the creators of linear animation.

Finally, in linear animation, comic timing is relatively easy to manipulate, whereas in interactive animation it is much more challenging. If a pause in a linear animation is too short, we can add additional frames; too long, take out a few. This is not to say that comic timing is simple; making something funny can be a significant challenge. However, in many forms of linear animation, the technology is not the hard part. The great success of linear animation as a comedic medium can be seen in the abundance of funny work that has been made over the past century, from Bugs Bunny to Pixar's film *Geri's Game*. In interactive animation, however, where users may impact the timing of events, certain kinds of comedy become very difficult. We can animate a character to tell a pre-recorded joke or pratfall, but ultimately that is simply falling back on the idea of pre-recorded dialog/action. Ultimately, while comedy is a staple of linear animation, it is not nearly as prevalent in interactive animation.

3. CONCLUSION

Over the last 30 years, animation has begun a dramatic shift in how it is practiced. Prior to this time, animation focused primarily on the production of linear films, both shorts and features. While there is still a vibrant industry and artistic practice in both short-form and feature-length animated film, new kinds of interactive works have sprung up that embrace animation as one of their key elements. The practice of animation is significantly different in these areas. This article has attempted to point out a few of the ways in which interactive animation is different from, and usually harder than, linear animation. We hope that an understanding of the problems and pitfalls can help animators avoid them when making the transition from linear to interactive animation, and may help point out some of the ways that researchers can help.

Computer game producers and other interactive entertainment companies now employ vast numbers of animators, many of whom have a background in linear animation in film and video. The transition from linear to interactive animator can be very challenging. An awareness of the differences may help make the transition easier, and may contribute to an understanding of the way the discipline of animation is evolving. This article has addressed some of these differences, in particular in the area of character design, and has looked at the impact of these differences on the process and product of animation.

At several points in this article, we have pointed out that interactive animation is usually harder than linear animation. There are two main reasons for this increased difficulty. First, linear animation has an established set of conventions and techniques on which to draw, with numerous examples of well-developed and compelling characters. While computer games and other interactive media are still working out many core questions on how to present believable and engaging characters in an interactive format. For example, for a linear animator, making a character that can smile requires a few strokes of a pencil, but is a non-trivial pursuit in an interactive domain. In areas where linear animation has laid groundwork, interactive animation should be able to draw on the established forms of its linear predecessor.

The second area of increased difficulty is that interactive animation has a number of new kinds of challenges that are not present in linear animation. Computational action selection, for example, is not a problem that linear animators have to address. Computer science researchers and others are whittling away at the added difficulty (as mentioned throughout this article), but currently animators are shouldering much of the burden of the vastly more complex possibilities of interactive animations.

Whether making linear or interactive animation, it is important to remember the principles of the traditional kind. As John Lasseter, director of *Toy Story* and other Pixar films, offered in describing an early computer animation he had done, "It was not the software that gave life to the characters, it was these principles of animation, these tricks of the trade that animators had developed over 50 years ago." [Lasseter 2001, p.45]. An awareness of the differences between linear and interactive animation may help interactive animations to exhibit the same core principles that made traditional linear animation so appealing to a broad audience.

ACKNOWLEDGEMENTS

This article reflects numerous discussions over many years of joint work by the author and numerous collaborators. In particular, the members of the Synthetic Characters Group at the MIT Media Lab – Prof. Bruce Blumberg, Michael Patrick Johnson, Kenneth Russell, Chris Kline, Michal Hlavac, Jed Wahl, Marc Downie, Delphine Nain, Geoff Beatty, Scott Eaton, Ben Resner, Robert Burke, Damian Isla, Matt Berlin, Jesse Gray, Adolph Wong, Derek Lyons, Bryan Yong, and Jennie Cochran – were central to the development of the author's style of interactive animation. In addition, the author would like to thank Michal Hlavac, Michael Moshell, Scott Fisher, and Celia Pearce for valuable discussions of the content of this article, and his current research group at UCI for their continued work on interactive animated characters. Finally, the author extends his thanks to the two anonymous reviewers whose careful attention to drafts of this article improved it significantly.

REFERENCES

AMAYA, K. AND BRUDERLIN, A. 1996. Emotion from motion. In *Proceedings of the Conference on Graphics Interface '96* (Toronto, Canada, 1996).

ANDERSON, C. A. AND BUSHMAN, B. J. 2001. Effects of violent video games on aggressive behavior, aggressive cognition, aggressive affect, physiological arousal, and prosocial behavior: A meta-analytic review of the scientific literature. *Psychological Science 12* (2001), 353-359.

ASSANIE, M. 2002. Directable synthetic characters. In *Proceedings of the 2002 AAAI Spring Symposium on Artificial Intelligence and Interactive Entertainment.*

BALL, G., LING, D., KURLANDER, D., MILLER, J., PUGH, D., SKELLY, T., STANKOSKY, A., THIEL, D., DANTZICH, M. V., AND WAX, T. 1997. Lifelike computer characters: The project at Microsoft Research. In *Software Agents*. J. M. Bradshaw, ed. AAAI Press/The MIT Press, Cambridge, MA, 1997, 191-222.

BARES, W. H. AND LESTER, J. C. 1997. Realtime generation of customized 3D animated explanations for knowledge-based learning environments. In *Proceedings of the 14th AAAI National Conference on Artificial Intelligence.*

BATES, J., LOYALL, A., AND REILLY, W. 1992. An architecture for action, emotion, and aocial behavior. In *Selected Papers from the Fourth European Workshop on Modeling Autonomous Agents in a Multi-Agent World*, *LINCS 830.* Springer, New York,1992, 55-68.

BENDAZZI, G. 1995. *Cartoons: One Hundred Years of Cinema Animation*: Indiana University Press, 1995.

BLUMBERG, B. 1998. Swamped! Using plush toys to direct autonomous animated characters. In *Proceedings of the SIGGRAPH 98 Conference*: *Abstracts and Applications.*

BLUMBERG, B. 1999. (void*): A cast of characters. In *Proceedings of the SIGGRAPH 99 Conference: Abstracts and Applications.*

BLUMBERG, B., DOWNIE, M., IVANOV, Y., BERLIN, M., JOHNSON, M. P., AND TOMLINSON, B. 2002. Integrated learning for interactive synthetic characters. In *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques* (San Antonio, TX, 2002).

BREAZEAL, C. 2000. Sociable machines: Expressive social exchange between robot and human. Sc.D. dissertation, Artificial Intelligence Laboratory. MIT, Cambridge, MA, 2000.

BROGAN, D. C. AND HODGINS, J. K. 2002. Simulation level of detail for multiagent control. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems* (2002).

CASSELL, J., BICKMORE, T., BILLINGHURST, M., CAMPBELL, L., CHANG, K., VILHJÁLMSSON, H., AND YAN, H. 1999. Embodiment in conversational interfaces: Rea. In *Proceedings of the CHI'99 Conference on Human Factors in Computing Systems* (Pittsburgh, PA, 1999).

DAMASIO, A. 1994. *Descartes' Error: Emotion, Reason, and the Human Brain*., G. P. Putnam's Sons, New York, 1994.

DEMARIA, R. AND WILSON, J. L. 2003. *High Score!: The Illustrated History of Electronic Games, 2nd ed.* McGraw-Hill Osborne Media, Emeryville, CA, 2003.

DOWNIE, M. 2001. Behavior, animation, music: The music and movement of synthetic characters. S.M. thesis, Media Arts and Sciences. MIT, Cambridge, MA, 2001.

DRUCKER, S. M. 1994. Intelligent camera control for graphical environments. Ph.D dissertation, Media Arts and Sciences, MIT, Cambridge, MA, 1994.

EKMAN, P. 1992. An argument for basic emotions. In *Basic Emotions*, N. Stein and K. Oatley, eds. Lawrence Erlbaum, Hove, UK, 1992, 169-200.

EL-NASR, M. S. AND HORSWILL, I. 2004. Automating lighting fesign for interactive entertainment. *Comput. Entertain. 2* (2004), 15-15.

ESA 2003. Key facts and research on video games and violence. http://www.theesa.com/keyfactsaug2003.htm, vol. 2003. Entertainment Software Association, 2003.

FUNGE, J., TU, X., AND TERZOPOULOS, D. 1999. Cognitive modeling: knowledge, reasoning and planning for intelligent characters. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (SIGGRAPH 99, Los Angeles, CA).

GAUDIOSI, J. 2003. Games, movies tie the knot. *Wired Online* (Dec. 10, 2003).

GRATCH, J., RICKEL, J., ANDRÉ, E., BADLER, N., CASSELL, J., AND PETAJAN, E. 2002. Creating interactive virtual humans: Some assembly required. *IEEE Intelligent Systems* (July/Aug. 2002), 54-63.

HAYES-ROTH, B., SINCOFF, E., BROWNSTON, L., HUARD, R., AND LENT, B. 1995. Directed E. improvisation with animated puppets. In *Proceedings of the CHI '95 Conference*.

HE, L., COHEN, M. F., AND SALESIN, D. H. 1996. The virtual cinematographer: A paradigm for automatic real-time camera control and directing. In *Proceedings of the SIGGRAPH 96 Conference* (Aug. 1996).

HODGINS, J. K. AND POLLARD, N. S. 1997. Adapting simulated behaviors for new characters. In *Proceedings of ACM SIGGRAPH 97 Conference*.

ISLA, D., BURKE, R., DOWNIE, M., AND BLUMBERG, B. 2001. A layered brain architecture for synthetic creatures. In *Proceedings of the International Joint Conferences on Artificial Intelligence* (IJCAI, Seattle, WA, 2001).

JOHNSON, M. P., WILSON, A., BLUMBERG, B., KLINE, C., AND BOBICK, A. 1999. Sympathetic interfaces: Using a plush toy to direct synthetic characters. In *Proceedings of the CHI 99 Conference.*

KLEISER, J. 1996. Digital stunt doubles: Safety through numbers (panel). In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*. ACM Press, 1996, 493-494.

KLINE, C. AND BLUMBERG, B. 1999. The art and science of synthetic character design. In *Proceedings of the AISB 1999 Symposium on AI and Creativity in Entertainment and Visual Art* (Edinburgh, Scotland).

LASSETER, J. 2001. Tricks to animating characters with a computer. *SIGGRAPH Computer Graphics 35* (Jan. 2001), 45-47.

LAYBOURNE, K. 1998.*The Animation Book: A Complete Guide to Animated Filmmaking-From Flip-Books to Sound Cartoons to 3-D Animation*. Three Rivers Press, 1998.

LIU, C. K. AND POPOVIC, Z. 2002. Synthesis of complex dynamic character motion from simple animations. *ACM Trans. Graphics 21* (2002).

MARKS, R. 2001. Enhanced reality: A new frontier for computer entertainment. In *Proceedings of the SIGGRAPH 2001 Conference: Abstracts and Applications: Emerging Technologies*.

MARSH, T. 2003. Presence as experience: Film informing ways of staying there. *Presence: Teleoper. Virtual Environ. 12* (2003), 538-549.

MEHRABIAN, A. AND RUSSELL, J. 1974. *An Approach to Environmental Psychology*. MIT Press, Cambridge, MA, 1974.

MURRAY, J. 1998. *Hamlet on the Holodeck: The Future of Narrative in Cyberspace*. MIT Press, Cambridge, MA, 1998.

MUSSE, S. R., BABSKI, C., CAPIN, T., AND THALMANN, D. 1998. Crowd modelling in collaborative virtual environments, In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology 1998*. ACM Press, New York, 115-123.

ORTONY, A. 1988. On making believable emotional agents believable. In *Emotions in Humans and Artifacts*, R. Trappl et al., eds. MIT Press, Cambridge, MA, 1988.

ORTONY, A., CLORE, G. L., AND COLLINS, A. 1988. *The Cognitive Structure of Emotions*. Cambridge University Press, New York, 1988.

PAIVA, A., MACHADO, I., AND PRADA, R. 2001. Heroes, villians, magicians, …: Dramatis personae in a virtual story creation environment. In *Proceedings of the 6th International Conference on Intelligent User Interfaces* (Santa Fe, NM, 2001).

PAUSCH, R., SNODDY, J., TAYLOR, R., WATSON, S., AND HASELTINE, E. 1996. Disney's Aladdin: First steps toward storytelling in virtual reality. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*. ACM Press, New York, 1996, 193-203.

PEARCE, C. 2004. Towards a game theory of game. In *First Person: New Media as Story, Performance and Game*. N. Wardrip-Fruin and P. Harrigan, eds. MIT Press, Cambridge, MA, 2004.

PERLIN, P, K. AND GOLDBERG, A. 1996. Improv: A system scripting interactive actors in virtual worlds. In *Proceedings of the ACM SIGGRAPH 96 Conference* (New Orleans, LA, 1996).

PICARD, R. 1997. *Affective Computing*. MIT Press, Cambridge, MA, 1997.

REILLY, W. S. N. 1996. Believable social and emotional agents. Ph.D dissertation, School of Computer Science, Carnegie Mellon Univ., Pittsburgh, PA, 1996.

REYNOLDS, C. W. 1999. Steering behaviors for autonomous characters. In *Proceedings of the Game Developers Conference* (San Jose, CA, 1999).

ROSE, C. 1999. Verbs and adverbs: Multidimensional motion interpolation using radial basis functions. Ph.D. Dissertation, Dept. of Computer Science, Princeton Univ., Princeton, NJ, 1999.

SCHAUB, H., ZOLL, C., AND AYLETT, R. 2003. Modelling empathy: The EU-project VICTEC (virtual information and communication technology with empathic characters). In *Proceedings of the Fifth International Conference on Cognitive Modeling* (Bamberg, Germany, 2003).

SIEGWART, R. AND NOURBAKHSH, I. R. 2004. *Introduction to Autonomous Mobile Robots*. MIT Press, Cambridge, MA, 2004.

STEED, P. 2002. *Animating Real-Time Game Characters*. Charles River Media, Hingham, MA, 2002.

THALMANN, D., NOSER, H., AND HUANG, Z. 1997. Autonomous virtual actors based on virtual sensors. In *Lecture Notes In Computer Science, Vol. 1195: Creating Personalities for Synthetic Actors*, Springer-Verlag, London, UK, 1997.

THOMAS, F. AND JOHNSON, O. 1981. *Disney Animation: The Illusion of Life*. Abbeville Press, New York, New York, 1981.

TOMLINSON, B., BLUMBERG, B., AND NAIN, D. 2000. Expressive autonomous cinematography for interactive virtual environments. In *Proceedings of the 4th International Conference on Autonomous Agents* (Barcelona, Spain, 2000).

TOMLINSON, B., DOWNIE, M., BERLIN, M., GRAY, J., LYONS, D., COCHRAN, J., AND BLUMBERG, B. 2002. Leashing the AlphaWolves: Mixing user direction with autonomous emotion in a pack of semi-autonomous virtual characters. In *Proceedings of the 2002 ACM SIGGRAPH Symposium on Computer Animation* (San Antonio, TX, 2002).

TOMLINSON, B., DOWNIE, M., BERLIN, M., GRAY, J., WONG, A., BURKE, R., ISLA, D., IVANOV, Y., JOHNSON, M. P., LYONS, D., COCHRAN, J., YONG, B., STIEHL, D., SOETJIPTO, R., ZAHAROPOL, D., AND BLUMBERG, B. 2001. AlphaWolf. In *Proceedings of the SIGGRAPH 2001 Conference Abstracts and Applications*. 2001.

TOMLINSON, W. 1999. Interactivity and emotion through cinematography. Masters thesis, Media Arts and Sciences. MIT, Cambridge, MA, 1999.

UNUMA, M., ANJYO, K., AND TAKEUCHI, R. 1995. Fourier principles for emotion-based human figure animation. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*. ACM Press, New York, 1995, 91-96.

VELASQUEZ, J. 1998. When robots weep: Emotional memories and decision-making. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence* (Madison, WI, 1998).

YI, M. 2004. Games of life. *The News Journal* (Jan. 20, 2004). http://www.delawareonline.com/newsjournal/life/2004/01/20gamesoflife.html.

Received August 2004; revised November 2004; accepted December 2004