M.I.T Media Laboratory Perceptual Computing Section Technical Report No. 378
May 1996

Computer Theater

Claudio S. Pinhanez
Perceptual Computing Group – MIT Media Laboratory
20 Ames St. – Cambridge, MA 02139
pinhanez@media.mit.edu

Abstract

The basic argument of this paper is that until recently theatrical performances involving human and computer actors were not possible due to the lack of appropriate computational models for action representation, recognition, and generation. This paper surveys and classifies some recent experiments in computerized performances, and uses this classification as a framework to establish the importance of action as a foundation of computer theater. It also argues that computer theater has many characteristics which qualify it as a good domain for AI and computer vision research on action. Finally, an example of a script for a computer theater performance is provided and analyzed in terms of scientific and technological challenges.

Action is the basis of theater[1] and, as such, needs to be fully incorporated in whatever model a computer is running during a computer-based theatrical performance. I believe the lack of good models for action is the basic reason for the relative absence of experiments involving theater and computers. The attempts to wire up stages or performers were in general concerned with dance (for instance, [20]), depending solely on the detection of positional and spatial movements.

Computer theater not only requires action representation and recognition but it is also an interesting domain for action research. To support this argument I begin by examining the multiple possibilities of making theater with computers, concerning both explored and unexplored developments. Some attempts to represent and recognize actions are examined in the second part of the paper, followed by the analysis of a very simple script written for a computer theater performance.

1 Computer Theater

As much as museums and art galleries seem to depend on the physical presence of objects, a performance requires the sharing of a common physical space. Otherwise the fundamental relation, *the suspension of disbelief*, does not take place.

In this paper I consider as *computer theater* only situations which involve human performers and audience in the same physical space, therefore excluding the idea of "distributed theater" as proposed by Krueger ([17], pg. 221). By doing this I am not ruling out the importance of new theatrical forms of art and entertainment which do not require physical sharing of space. My objective is mostly to delimit the scope of computer theater to simplify the analysis and be able to draw generic conclusions.

I also restrict the usage of the term computer theater to performance situations (ruling out, for instance, user browsing and story-telling). Computer theater, in my view, is about providing means to enhance the artistic possibilities and experiences of professional and amateur actors, or of audiences clearly engaged in a representational role in a performance ([35]).

The classification of interactive computer music systems proposed by Rowe in [32] is an interesting starting point for the understanding of the possibilities of computer theater. Rowe classifies music systems according to three dimensions, two of which naturally extend to theatrical systems.

Particularly important to my analysis is Rowe's differentiation between *player* and *instrument* paradigms for interactive music systems. Based on his classification, I propose the categorization of computer theater systems into three categories: hyper-actors, computer-actors, and computerized stages.

1.1 Hyper-Actors

Rowe ([32]) classifies an interactive musical system as following the *instrument* paradigm if the basic concern is to construct an extended musical instrument. For instance, in the *hyperinstruments* project led by Tod Machover at the MIT Media Laboratory ([21]), musical instruments were built which sense a virtuoso musician's gestures, enabling him/her to control and modulate a computerized counterpart to the acoustic sound produced by the instrument.

An actor's instrument is his body — including voice and facial expression: "virtuosi" actors are able to control their bodies in extraordinary and different ways. Through the centuries actors have relied on masks, make-up, and costumes to alter their bodies, or in the extreme case, on puppets and marionettes.

I suggest the term *hyper-actor* to denote computer theater systems which aim to enhance the actor's body and therefore his expressive capabilities. A hyper-actor expands an actor's body so he is able to trigger lights, sounds, or images on a stage screen; to control his final appearance to the public if his image or voice is mediated through the computer; to expand its sensor capabilities by receiving information through earphones or video goggles; or to control physical devices like cameras, parts of the set, robots, or the theater machinery.

The idea has been more explored in dance and music

[1] Langer, in [19], chapter 17, contains an interesting discussion about the basics of theater.

than in theater. Body suits wired up with sensors having been widely explored with, recently in the works of Troika Ranch[2]. Other examples include the performances of Laurie Anderson involving the processing of her voice and singing through a Synclavier ([2]); George Coates' experimentation with actors receiving the script from Internet users during the live performance of *"Better Bad News"*[3]; and Christopher Janney's performances where a musician and a dancer played with the sound of their heartbeats[4].

Another possibility is having the actor not on stage, and providing him the means to control the physical appearance of his image to the audience. Mark Reaney's "virtual theater" ([30]) is a very curious illustration of this concept. In a typical scene an actor on stage plays with an off-stage actor whose image is seen by the audience on two large stereo-graphic video screens (the audience wears special 3D-goggles). The off-stage actor's image expands and contracts according to the play events and is used to symbolize and enrich the power struggle portrayed in the scene.

1.2 Computer-Actors

The *player* paradigm in interactive music systems corresponds to situations where the intention is to build *"... an artificial player, a musical presence with a personality and behavior of its own..."*[5]. In the computer theater realm the player paradigm corresponds to the *computer-actor*, a computer program which assumes the role of one of the characters of the play. The computer displays the characters' actions in some sort of output device such as video screens, monitors, speakers, or physical devices.

The distinction between hyper- and computer-actors is important because computer-actors require a control system which decides what to do "independently" of the desires of the human partners. A computer-actor must be able to follow the script — if there is one — and react according to its own role; here, the issues of decision and control of expressiveness seem to be more relevant than in the case of hyper-actors. In contrast, hyper-acting (without the bad connotation of the term) is likely to require much better sensing and understanding of the human performer actions.

A straightforward implementation of computer actors would be human-like or cartoonish characters displayed on a stage screen. Most of the interesting examples come from the research oriented towards direct user interaction with computer-generated characters ([41]). Worth of mention is the work of Bruce Blumberg ([22]) in building a computer graphics generated dog which interacts with the user, obeying simple gestural commands ("sit", "catch the ball") while satisfying its own agenda of necessities ("drink water", "urinate"). The interaction occurs through a large video screen which simulates a mirror where the user and the dog occupy the same space.

An interesting alternative is being developed by Flavia Sparacino ([38]) who is incorporating behavior-based interaction into text, pictures, and video, constituting what she calls *media-creatures*. The project also involves exploring media-creatures in dance and theater performances — *media-actors*. Similarly, computer-actors can be computer-generated objects which do not exist in the real world (or do not normally interact with people). As an example of the idea, Sommerer and Mignonneau[6] developed an art installation where fractal-based images of plants grow when the user touches a real plant in the space. Actors and dancers can also be embodied in robots[7] and kinetic sculptures.

As a performer and director I find the possibility of writing for, directing, or performing with non-existent objects and media-actors a dazzling way to expand the realm of expressiveness of theater in novel directions, as much as synthesizers have expanded the concept of music and musical events.

1.3 Computerized Stages

It is worth mentioning another type of computer theater systems which are concerned with the expansion of the possibilities for the stage, set, props, costumes, light and sound. The fundamental distinction with the hyper- and computer-actors is that elements of *computerized stages* are not characters or representations of characters of a play.

A stage can react by changing illumination, generating visual and sound special effects, changing the appearance of backdrops and props, or controlling machinery. An example is the Intelligent Stage project at Arizona State University ([20]) which enables the mapping of volumes in the 3D space to MIDI outputs. Movement and presence of dancers are monitored by 3 cameras, triggering music and lights accordingly.

2 Dimensions of Experimentation

It is also important to distinguish between different aspects to be considered for investigation and experimentation in computer theater. Particularly because some modes and possibilities have hardly been addressed by past and current computer theater systems as, for instance, it is the case of rehearsable computer-actors and script-following systems.

2.1 Rehearsal vs. Performance

Ensemble rehearsing is a key part of the artistic process of theater. Compared to music, the ensemble rehearsal process in theater is longer and richer in experimentation. Characters are usually built through the interaction between actors on the stage with supervision and guidance from the director.

Performing with a bad actor is bad, but rehearsing with a bad actor is quite worse. An unmotivated or limited actor in rehearsal can stop the creative process of the whole company. The importance of rehearsal is a major point that most of the computer theater experiments so far have inadequately addressed, especially in the case of computer-actors.

According to this view, one of the biggest challenges in computer-theater is to build hyper- or computer-actors which can actively respond to variations in the staging of

[2] `http://www.art.net/Studios/Performance/Dance/Troika_Ranch/TroikaHome.html`

[3] `http://www.georgecoates.org/`

[4] `http://www.janney.com/heartb.htm`

[5] [32], pg. 8.

[6] `http://www.mic.atr.co.jp/~christa/`

[7] `http://guide.stanford.edu/people/curtis/machoreo.html`

a script as they are discovered and proposed during rehearsal time by the human actors and by the director. Such *rehearsable* computer-actors seem to require more action representation and recognition than *performance-only* computer-actors.

2.2 Scripts and Improvisations

Rowe ([32]) also distinguishes between *score-* and *performance-driven* computer music, which I map to the concepts of scripted and improvisational computer theater. *Scripted* computer theater systems are supposed to follow totally or partially the sequence of actions described in a script. During the performance the system synchronizes the on-going action with the script, giving back its "lines" as they were determined during the rehearsal process or, less interestingly, in an off-line mode by the director.

Improvisational theater relies on well-defined characters and/or situations. This type of computer theater has immediate connections with the research on developing characters for computer games and software agents ([3, 22]). However, the distinction between actor/situation is not present in most game-agents, impoverishing the theatrical interest in such creatures since a major source of dramatic conflict in an improvisation is setting the characters in unexpected situations and environments.

Yet more important is the fact that good improvisation requires recognition of intentions. Knowing what the other character wants to do enables interesting and rich counteracting behavior. Otherwise, the resulting piece is flat, structurally resembling a "dialogue" with an animal: the sense of immediacy dictates most of the actions.

It is interesting to notice that most of the initial research in interactive computer music was concerned with score-driven systems ([36]), while computer theater seems to be steering towards improvisation, perhaps following the present interest in interactive spaces for computer games ([33]).

2.3 Performers and Users

Most of the experiments on building interactive computer characters have been targeted towards non-actors, people unfamiliar with the computerized environment/space. This is not the case in computer music, where the fundamental concern has been to provide tools for people with some musical training.

Although the development of ideas and methods to concretely engage users is very important, I believe it is fundamental to first concentrate efforts on understanding and reacting to actors and audience in a performance situation. There is an important reason to do so: users are boring from the action point of view. Users are motivated mostly by curiosity, and their repertoire of displayed actions/reactions is normally very restricted.

It is very hard to develop a performance with people who are not committed to engage — as street performers well know. A system assuming non-engaged users must rely on story-telling techniques to create an interesting environment. Theater — and, in general, performance — can go beyond story-telling by assuming that performers and audience know and are engaged in their roles.

Galyean's work ([9]) in interactive narrative is an interesting example where the user navigates a virtual world which coerces him to watch a story. At some point of the narrative, the user becomes — unwillingly — the central character of the plot. There is a great deal of control embedded in the program to assure the engagement of the user: if the user does not pay attention to key events, they are repeated, and the story includes moments designed to attract attention, as for instance, when a car is made to crash in front of the "virtual" user. Immersive story-telling environments like Galyean's seem to require different techniques for interaction with their audience/users, relying more on interactive narrative than on action-based dramatic structure.

My view is that user-oriented computer theater systems should be concerned with the provision of true theatrical experiences, transforming the user into a performer, an actor, or a director. The trivial example of a performance experience is a karaoke-style system where the user interacts with a computer-actor in front of an audience — a *theatroke*[8] — like the "Interactive Shakespeare" built by Michael Naimark[9].

More interesting is the possibility of giving the user the experience of acting, in the sense of building a character or living intensively a theatrical situation. Rehearsal improvisations and theatrical games ([15]) are, sometimes, the most rewarding aspects of the theater craft. In computer music, a similar issue is being addressed by the new generation of hyper-instruments built by Tod Machover, aimed to *"... make the interaction with such sophisticated systems/instruments as intuitive and natural as possible for the general public (...) but that do NOT eliminate the necessity of concentration, skill and imagination (so make challenging MENTAL interfaces)"*[10].

3 Computer Theater Based on Action

It is certainly possible to have a computer theater system which just produces output from a pre-determined and pre-timed computer "script". As soon as the performance starts, the computer generates output according to a list of timed "cues" determined precisely during rehearsal. Although human actors (and especially dancers) can adjust their performances to such situations, the results can be expected to be devoid of richness and life. Computer theater seems to be worthwhile only if the hyper- or computer-actor also follows the actions of its human partners and adjusts its reactions accordingly.

In the case of scripted theater the computer system must recognize the actions being performed by the human actors and to match them with the information from the script. We could conceive a "minimal" system as using a detailed list of sensory inputs and the corresponding computer-generated outputs. The list can be provided manually by the director or technical assistants, and, during performance, the computer system just matches the listed sensory inputs and generates the pre-determined output.

Although the "simple" system just described is hard to implement in practice due to noisy sensors and performance variability, I believe there is a much more interesting

[8] Since *karaoke*, in Japanese, literally means "empty orchestra", a more appropriate word would be *karayaku* which can mean something like "empty role" or "empty actor".

[9] See a description in [17], pg. 223.

[10] `http://brainop.media.mit.edu/growth/`

approach to computer theater based on *action understanding*. Instead of providing a computer-actor with such a cryptic list of sensor-reaction mappings, the challenge can be to use as input the actions and reactions as determined textually in a script or by the director.

The textual description of the actions corresponding to the human part can then be analyzed by the computer producing visual and auditory components which can be detected by sensory routines (see [26]). On the other hand, the hyper- or computer-actor actions in the script can be used to directly generate low-level instructions for computer-graphics or external physical devices (see [16]). According to this view, a computer-actor should be instructed by words like "shout" or "whisper", and be able to recognize automatically an action described simply as "actor walks to the chair".

A positive feature of action-based verbal descriptions is precisely their vagueness. A description like "actor walks to the chair" does not specify from where the actor comes, the path taken, etc. Instead, it highlights only the final destination enabling the actor to explore different ways of performing it without disengaging the recognition system. Similarly, describing the computer-actor's actions in textual mode provides room for reactive mechanisms during rehearsal or performance time. An instruction like "computer shouts back" can be translated not as a fixed volume level, but as a volume proportional to the current scene's sound level.

Action-based description of scenes can also be employed to describe the structure of improvisational systems with the same advantages, although the recognition of actions is complicated here by the lack of pre-determined ordering. However, as mentioned before, to implement improvisational computer-actors it is necessary to deal with the recognition of the intentions behind the actions of the human actor. For example, in an "escape sketch" it is not only necessary to recognize that an actor is moving erratically in a room, but also whether he is trying to escape from it and which exit he intends to use. Perceptual recognition of intentions has been hardly explored, and I believe it constitutes a major challenge for the design of interesting improvisational computer-actors.

4 Computer Theater as a Domain for Action Understanding

Not only the action approach seems to be appropriate to computer theater, but I also believe that computer theater is a very good domain for research in action understanding. Action is the basic underlying foundation of theater, and specific characteristics of the theatrical activity provide a range of interesting elements for the AI research on action.

In the simplest analysis, it is easy to see that the gestures employed by actors are more explicit and determined than ordinary human activity. For instance, if holding a glass of whiskey is important, the actor makes sure that the audience notices when the glass is picked up. Theater also naturally provides a wider range of gestures, postures, and situations than normal life.

Moreover, actions in theater are staged such as every member of the audience can see them, and minuscule gestures are rarely used. Therefore, visual action recognition can employ long-shot, wide-angle cameras which correspond to the audience's field of view, avoiding the problems of having different image resolution needs which plague, for instance, the building of an action recognizer for an office space.

There are also more interesting and deeper reasons to use theater as a domain for action understanding. Theater defines clear and defined contexts which provide natural limits for reasoning and recognition processes. The context is described in the script, as well as the basic repertoire and sequence of actions and movements of the actors. Also, the mechanics of the dramatic text makes attention to be driven by actions of the performers, and only quite rarely by non-human caused events.

However, my greatest hope is that computer theater might enable action research to start tackling the hard issue of intentionality. In theater, the intentions of the play-writer are translated into physical activity (here including voice punctuation and intonation) by the joint effort of the director and actors during rehearsal. Traditionally, the director analyses the text and assigns intentional actions to parts of the script, and general objectives to different characters (see [7], chapter 7 and part IV). During rehearsal, the actors are guided to find physical activities which correspond to the intention of the characters, in a process which often involves improvising and relating the context to personal experience ([5]). Thus, the resulting activities are loaded with explicit intentional elements by design.

Therefore, theater enables us to assume that every action is intentional and the result of the conflict between the character's objectives and the other actors' actions. In other terms, intentions can be expected to be explicit and present in every activity, creating situations where the relationships between intention, action, and physical activity are easier to study and model.

5 Action Representation, Recognition, and Generation

As I argued above, computer theater can become a good domain to investigate action representation and recognition. In this section I briefly survey the current state of the research in action representation, recognition, and generation.

5.1 Representation

Representing actions has been the object of research of linguistics ([14, 34, 27]), computer graphics ([16]), and computer vision ([37]). However, the research is still struggling against traditional AI problems like context use and common-sense reasoning.

As part of my Ph.D. research with Prof. Aaron Bobick at the MIT Media Laboratory, a representation scheme has been developed based on Schank's *conceptualizations* ([26]), called *action frames*. Our research considers the domain of TV cooking shows, and has been applied in the development of *SmartCams*, automatic cameras for TV studios ([25]). This work should extend naturally to computer theater, since the camera control system is designed to be independent of the general system which uses and understands actions from the script of the cooking show.

Part of our current research efforts are focused on designing a better representation for actions. We are still debating the convenience of using Schank's primitives to describe every action. Also, action frames need to be augmented by incorporating visual elements, as in [16], and time references, possibly using Allen's interval algebra ([1]). Another important element missing in our representation system is a mechanism to specify the intensity of an action. For computer theater purposes, the difference between "talking" and "shouting" is crucial.

5.2 Recognition

Research in visual action recognition has been restricted to recognizing human body movements ([13, 31, 29, 10, 4]). [18, 37, 12] are among the few works which actually examined some of the issues related to understanding actions and their effects in the world.

Bruce Blumberg's dog mentioned above uses the recognition capabilities of ALIVE ([22]) to react to commands like "pointing", "sitting", and "catch the ball". The limited vocabulary and precise context enables the translation of hand positions directly into actions: an extended arm into the ground means "sitting" independently of the actual shape of the hand or the direction of sight.

Aaron Bobick and I have been conducting research (unpublished) addressing visual action recognition. The key idea behind the proposal is to represent time constraints using Allen's interval algebra ([1]), enabling vaguely specified relationships among the actions, sub-actions, and visual features. The visual features are obtained from a dictionary of action verbs which translates the actions from the script into information about features detectable by visual routines.

Recognition of displayed emotional states can determine the appropriate reaction of a computer-actor. [8] describes some work on recognition of human facial expressions. In the case of hyper-actors, detection of true emotional states can be used to control the augmented body. Affective states have been addressed by Roz Picard at MIT ([24]) through direct sensing of body variables such as blood pressure and heartbeat. However, it is hard to predict if such techniques can work in performance or rehearsal since often the displaying of emotions is precisely — and coldly — controlled by the actor.

Speech recognition can be a simple way to synchronize performance and scripts, by matching the spoken words with the lines in the script. However, in improvisational theater it is necessary to go a step beyond, and add understanding capabilities to the system. Another dimension of voice is the expression of emotions, although recognition research in this case is only beginning ([28]).

5.3 Generation

There has been a significant amount of work to incorporate action into computer-graphics: [16, 40] are good examples. Perlin's work ([23]) is particularly interesting because the computer-actor receives commands directly as action verbs.

The synthesis of facial expressions for human-like computer characters has also received significant attention from the computer graphics community. The work of Terzopoulus [39] is a typical example where the modeling of facial muscles and tissue enables a variety of facial expressions.

Although speech synthesizers are already commercially available, the ability to generate expression-modulated speech is still a subject of research ([6]). Finally, a good example of attributing expressiveness to media objects is the work of Yin Yin Wong ([42]) with expressive typography. In this case, text dynamically changes its shape, typeface, color, and screen position in order to convey temporally the expressive dimension of the message.

6 A Computer Theater Performance

The following is a script for a mime-like computer theater performance which exemplifies some of the possibilities for hyper- and computer-actors. It is not supposed to be artistically complex or ambitious: I intentionally made the option for a simple and direct plot to emphasize the nature of interaction expected to happen between the actor and the computer theater system:

> *A large video screen, covering the diagonal at stage left, shows the image of a sidewalk. Computer graphics images of empty shoes — in fact, computer-actors — walk by the sidewalk once in a while. A human actor uses a sound-generator hyper-actor, being able to produce sound effects accompanying some of his actions. He enters the stage carrying a shoe-shining box. He puts the box in front of the screen. Then he reaches for a bucket lying upstage right, and throw the contents towards the screen. He follows with his arm (producing sound) the invisible path of the bucket's contents till he sees a big spill of mud on the sidewalk. A pair of shoes enters the screen and steps on the mud, getting dirty. He mimes polishing each shoe (as if they are real and outside of the screen), and the screen's shoes become shiny — and happy. Polishing produces a music following the rhythm and the intensity of the shining. He receives money and waves back as the shoes leave the screen. A pair of bare feet enters and gets dirty. He polishes them, while they seem to scream of pain, and run away without paying. He gets someone from the audience to become the shoe-shiner. He steps outside, showing the new shoe-shiner how to react to a pair of woman high-heels. The high-heels are very happy with the member of the audience's work, and the actor becomes jealous, and pushes the member of the audience out of the stage. When he tries to continue polishing the high-heels, they kick him and go away. A giant pair of shoes enters, steps on the mud, and mud is spilled on the actor. He starts polishing himself, visibly sad. When he is done, the giant shoes come back and spill mud on him again. The actor and all the shoes bow acknowledging the applauses while credits are displayed on the screen.*[11]

This script is biased towards visually recognizable actions and it is designed to provide clear physical activity. By this example, we can see the variety of actions a performance can have. However, I am still considering the appropriateness of many of the script elements and the

[11] Adapted from "Shoeshine" by James Gousseff ([11]), pg. 72.

feasibility of detecting many of the actions with current real-time computer vision technology.

The process to produce this performance starts by inputing the script into the computer system responsible for the recognition and generation of CG animation. The script is translated into a representational scheme based on action (ideally, the translation is automatic), which is fed both to the visual recognition system and to the hyper- and computer-actors' systems.

The visual recognition system expands the information from the script to determine proper visual cues for the beginning and ending of all actions (again, ideally, automatically — see a similar inference situation in [26]). The computer-actors also translate the actions into low-level graphics automatically. Actor and computer-actors rehearse together, defining the physical activities implicit in the actions. Here, the director evaluates the performance of the computer-actors and suggest modifications through a computer interface: intensity and path of movements, shape, and color are among the parameters which can be controlled. Similarly, the actor tunes the output of his hyper-actor system by giving instructions associated with the actions. On the other hand, actions which are not being distinguished by the action recognition system can be enriched with more detailed information about position and attitude of the actor, disambiguating the input.

If we compare this example with the previous section on the current state of action understanding, it is clear that much work needs to be done. I do not expect to see such powerful systems running in a short period of time. However, there are many sub-components and sub-problems which can be solved separately and integrated with simplified version of other modules. The important thing about the above script is not how it can be implemented but that it makes clear how rich and challenging is the computer theater domain for action recognition and representation.

7 Final Remarks

Throughout this paper I stressed many times the importance of action in computer theater. Action and reaction are essential to the vitality of theatrical performance and must be incorporated, implicitly or explicitly, into any computer theater system.

The classification of computer theater developed in the first half of the paper is important to clarify and compare different techniques and approaches and their dependency on particular computer theater systems. Rather than being exhaustive, the enumeration of different possibilities of computer theater has the aim of guiding the design of new systems targeting specific scientific or artistic concerns.

Finally, I believe action representation and recognition research can be significantly boosted by using the computer theater domain. Defined contexts, exaggerated gestures, controlled environments, known and reliable mappings between symbols and real world, explicit translation of intentions into physical activities, and richness of different situations can provide a fertile environment for action research. The disadvantages are the likely "toy" domains, and the difficulties on devising evaluation methods.

Acknowledgment

Section 4, which examines computer theater as a domain for action understanding, has acquired its present shape due to many suggestions and ideas from my Ph.D. advisor, Prof. Aaron Bobick of the MIT Media Laboratory.

References

[1] James F. Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23:123–154, 1984.

[2] Laurie Anderson. *Empty Places.* HarperPerennial, New York, New York, 1991.

[3] Joseph Bates, A. Bryan Loyall, and W. Scott Reilly. An architecture for action, emotion, and social behavior. In *Proceedings of the Fourth European Workshop on Modeling Autonomous Agents in a Multi-Agent World*, S. Martino al Cimino, Italy, July 1992.

[4] Aaron F. Bobick and James W. Davis. An appearance-based representation of action. Technical Report 369, M.I.T. Media Laboratory Perceptual Computing Section, February 1996. To appear in ICPR'96.

[5] Melissa Bruder, Lee Michael Cohn, Madeleine Olnek, Nathaniel Pollack, Robert Previto, and Scott Zigler. *A Practical Handbook for the Actor.* Vintage Books, New York, New York, 1986.

[6] Janet Cahn. Generating expression in synthesized speech. Master's thesis, M.I.T. Media Arts and Sciences Program, May 1989.

[7] Harold Clurman. *On Directing.* Collier Books, New York, New York, 1972.

[8] Irfan Essa, Sumit Basu, Trevor Darrell, and Alex Pentland. Modeling, tracking and interactive animation of faces and heads using input from video. In *Proceedings of Computer Animation '96 Conference*, Geneva, Switzerland, June 1996.

[9] Tinsley A. Galyean. *Narrative Guidance of Interactivity.* PhD thesis, M.I.T. Media Arts and Sciences Program, 1995.

[10] D. M. Gavrilla and L. S. Davis. 3-D model based tracking of human upper body movement: a multi-view approach. In *Proc. of the IEEE-PAMI International Symposium on Computer Vision*, pages 253–258, Coral Gables, Florida, November 1995.

[11] James W. Gousseff. *Street Mime.* The Dramatic Publishing Company, Woodstock, Illinois, 1991.

[12] Gerd Herzog and Peter Wazinski. VIsual TRAslator: Linking perceptions and natural language descriptions. *Artificial Intelligence Review: Special Issue on Integrating Language and Vision*, 8, 1994–5.

[13] David Israel, John Perry, and Syun Tutiya. Actions and movements. In *Proc. of the 12th IJCAI*, pages 1060–1065, Sydney, Australia, August 1991.

[14] Ray Jackendoff. *Semantic Structures.* The M.I.T. Press, Cambridge, MA, 1990.

[15] Keith Johnstone. *IMPRO: Improvisation and Theatre.* Methuen Drama, England, 1981.

[16] Jugal Kumar Kalita. *Natural Language Control of Animation of Task Performance in a Physical Domain.* PhD thesis, University of Pennsylvania, Philadelphia, Pennsylvania, June 1991.

[17] Myron W. Krueger. *Artificial Reality II.* Addison-Wesley, 1990.

[18] Y. Kuniyoshi and H. Inoue. Qualitative recognition of ongoing human action sequences. In *Proc. of IJCAI'93*, pages 1600–1609, 1993.

[19] Susanne K. Langer. *Feeling and Form.* Charles Scribner's Sons, New York, New York, 1953.

[20] Robb E. Lovell and John D. Mitchell. Using human movement to control activities in theatrical environments. In *Proc. of Third International Conference on Dance and Technology*, 1995.

[21] Tod Machover. Hyperinstruments: A progress report. Technical report, M.I.T. Media Laboratory, January 1992.

[22] Pattie Maes, Trevor Darrell, Bruce Blumberg, and Alex Pentland. The ALIVE system: Full-body interaction with autonomous agents. In *Proc. of the Computer Animation '95 Conference*, Geneva, Switzerland, April 1995.

[23] Ken Perlin. Real time responsive animation with personality. *IEEE Transactions on Visualization and Computer Graphics*, 1(1):5–15, 1995.

[24] R. W. Picard. Affective computing. Technical Report 321, M.I.T. Media Laboratory Perceptual Computing Section, November 1995.

[25] Claudio Pinhanez and Aaron F. Bobick. Intelligent studios: Using computer vision to control TV cameras. In *Proc. of IJCAI'95 Workshop on Entertainment and AI/Alife*, Montreal, Canada, August 1995.

[26] Claudio S. Pinhanez and Aaron F. Bobick. Approximate world models: Incorporating qualitative and linguistic information into vision systems. To appear in AAAI'96, 1996.

[27] Steve Pinker. *Learnability and Cognition.* The M.I.T. Press, Cambridge, MA, 1989.

[28] Jeffery Pittam, Cynthia Galois, and Victor Callan. The long-term spectrum and perceived emotion. *Speech Communication*, 9:177–187, 1990.

[29] R. Polana and R. Nelson. Low level recognition of human motion. In *Proc. of IEEE Workshop on Motion of Non-Rigid and Articulated Objects*, pages 77–82, Austin, Texas, November 1994.

[30] Mark Reaney. Virtual scenography: The actor, audience, computer interface. *Theatre Design and Technology*, 32(1):36–43, Winter 1996.

[31] K. Rohr. Towards model-based recognition of human movements in image sequences. *CGVIP: Image Understanding*, 59(1):94–115, January 1994.

[32] Robert Rowe. *Interactive Music Systems.* The MIT Press, Cambridge, Massachusetts, 1993.

[33] Kenneth Russell, Thad Starner, and Alex Pentland. Unencumbered virtual environments. In *Proc. of IJCAI'95 Workshop on Entertainment and AI/Alife*, pages 58–62, Montreal, Canada, August 1995.

[34] Roger C. Schank. Conceptual dependency theory. In *Conceptual Information Processing*, chapter 3, pages 22–82. North-Holland, 1975.

[35] Richard Schechner. *Performance Theory.* Routledge, London, England, 1988.

[36] Elliot Schwartz and Daniel Godfrey. *Music since 1945: Issues, Materials, and Literature.* Schirmer Books, New York, New York, 1993.

[37] Jeffrey Mark Siskind. Grounding language in perception. *Artificial Intelligence Review*, 8:371–391, 1994–5.

[38] Flavia Sparacino. Directive: Choreographing media creatures for interactive virtual environments. Technical Report 377, M.I.T. Media Laboratory Perceptual Computing Section, 1996.

[39] D. Terzopoulus and K. Waters. Analysis and synthesis of facial image sequences using physical and anatomical models. *IEEE PAMI*, 15(6):569–579, June 1993.

[40] Nadia Magnetat Thalmann and Daniel Thalmann. *Synthetic Actors in Computer Generated 3D Films.* Springer-Verlag, Berlin, Germany, 1990.

[41] Naoko Tosa, Hideki Hashimoto, Kaoru Sezaki, Yasuharu Kunii, Toyotoshi Yamada, Kotaro Sabe, Ryosuke Nishino, Hiroshi Harashima, and Fumio Harashima. Network-based neuro-baby with robotic hand. In *Proc. of IJCAI'95 Workshop on Entertainment and AI/Alife*, Montreal, Canada, August 1995.

[42] Yin Yin Wong. Temporal typography: Characterization of time-varying typographic forms. Master's thesis, Massachusetts Institute of Technology, 1995.