Interactive Robot Theatre

Cynthia Breazeal, Andrew Brooks, Jesse Gray, Matt Hancher,
Cory Kidd, John McBean, Dan Stiehl, Joshua Strickon
MIT Media Lab
77 Massachusetts Ave NE18-5FL, Cambridge, MA 02139 USA
E-mail: [cynthiab,zoz,jg,mdh,coryk,jmcbean,strickon]@media.mit.edu.

Abstract

This work motivates interactive robot theatre as an interesting test bed to explore research issues in the development of sociable robots and to investigate the relationship between autonomous robots and intelligent environments. We present the implementation of our initial exploration in this area highlighting three core technologies. First, an integrated show control software development platform for the design and control of an intelligent stage. Second, a stereo vision system that tracks multiple features on multiple audience participants in real-time. Third, an interactive, autonomous robot performer with natural and expressive movement that combines techniques from character animation and robot control.

1 Introduction

Entertainment robotics is thought of as a relatively new field, but the idea of building lifelike machines that entertain us has fascinated us for hundreds of years with the first ancient mechanical automata (Wood, 2002). Some of the most well known examples of modern entertainment robots include highly expressive animatronics in theme parks that perform fully automated and fixed routines.

As the field of entertainment robotics matures, new applications incorporate autonomous robotic technologies. For instance, small mobile robots with navigational skills have been used in live performances (Werger, 1998), Carnegie Mellon University has incorporated speech recognition and dialog systems to allow animatronics to carry out simple verbal interactions with people (www.etc.cmu.edu/projects/iai/), or to have mobile robot performers carry out simple "improvisational" dialogs with each other (Bruce et al., 1999). New applications for entertainment robots in the home, such as Sony's robot dog Aibo, has sparked companies to create a menagerie of robot toys. Meanwhile, new areas of inquiry for human-robot interaction (HRI) and social/sociable robots in the research arena address the challenging issues associated with developing robots that can interact naturally and appropriately with people while serving as helpers for the elderly, teammates for astronauts, museum docents, domestic assistants, and more (Fong et al., 2002; Breazeal, 2002).

We argue that Interactive robot theatre with human participants could serve as a viable test domain for developing the social interactivity of robots – similar to how *RoboCup Soccer* serves as a suitably constrained yet interesting test scenario for the field of multi-agent robotics (Kitano et al., 1997). For instance, the script places constraints on the dialog and interaction, and it defines concise test scenarios. The stage constrains the environment, especially if it is equipped with special sensing, communication, or computational infrastructure. More importantly, the intelligent stage, with its embedded computing and sensing systems, is a resource that autonomous robotic performers could use to bolster their own ability to perceive and interact with people within the environment. Finally, a robot actor must be able to act/react in a convincing and compelling manner to the performance of another as it unfolds. This requires sophisticated perceptual, behavioral, and expressive capabilities. Introducing improvisation, or allowing for greater audience participation, makes the situation more unpredictable and less constrained — approaching open-ended interaction with people.

In the future, we may see more elaborate versions of interactive robot theatre in theme parks, museums, and storefront windows. Someday, there may even be fanciful robotic characters on Broadway performing with human actors on an intelligent stage.

2 The Installation

Our robotic terrarium exhibit (see video) is an early exploration of these futuristic visions, combining three core technologies (i.e., intelligent stage software toolkit, real-time vision for perceiving people, and expressive and lifelike robot movement) to create an intelligent stage with autonomous robotic performers. The installation was exhibited at the SIGGRAPH 2002 Emerging Technologies Exhibit in San Antonio, TX.

The storyline is inspired by primitive life of an alien world. By day, a serpentine, anemone-like creature (called Public Anemone) is awake and interacts with the audience and aspects of its environment. It perceives audience members through a real-time stereo vision system, allowing people to compete for the anemone's attention and to distract it from performing its daily chores such as "watering" the nearby plants, "drinking" from the pond, or "bathing" in the waterfall. The anemone responds to people by orienting towards them and tracking their movements. If someone gets to close, however, the anemone becomes "frightened" and recoils defensively like a rattlesnake. Because the robot makes decisions based on its internal drives and audience interaction, its behavior is different from each day to the next but follows a coherent theme.

At night, Public Anemone goes to sleep and a variety of nocturnal creatures and special effects emerge including glowing fiber optic tubeworms , musical drum crystals, luminescent wall crystals, and a sparkling pond covered in a gentle mist. The audience interacts with these synthetic creatures through touch, eliciting light and musical responses. The tubeworms detect the nearby proximity of people through capacitive sensing, causing them to react musically, optically and mechanically into their shells. The drum crystals allow participants to create rhythm sequences based on how hard they are tapped, synchronized with the glowing crystal wall theme.

3 Theatre Control Software

The terrarium, shown in Figure 1, is an intelligent stage with embedded computation, communication, and sensing systems. Based on the incoming sensory information, the stage autonomously controls a number of special effects elements, including 8 channels of digital audio and music, 40 color controlled lights, 6 ultrasonic foggers, 1 waterfall pump, tubeworm capacitive sensors, tubeworm servo controller, drum crystal triggers, and pond grass lights.

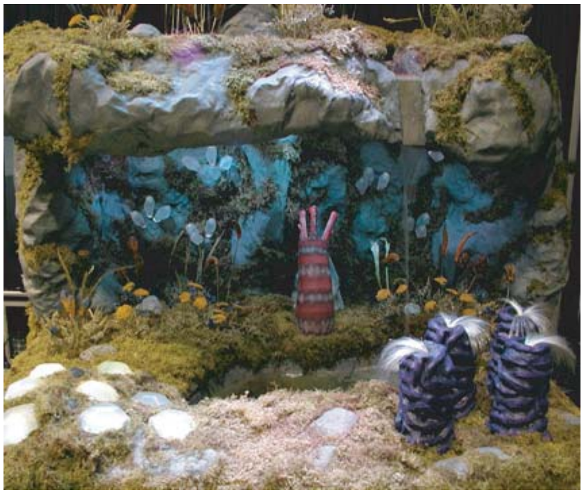

Figure 1: Audience participants are allowed to interact with the cyber flora and fauna of our fanciful robotic terrarium-like stage as it transitions from day to night.

We developed a dedicated software system, called the *Secret Systems* (Strickon, 2002), to design and implement the "brain" of the intelligent stage. The Secret Systems is a new, integrated toolset for the rapid development of interactive, location-based experiences that handles every step of the design process – from the hardware layout to specification of interactive experiences.

The Secret Systems creates an abstraction barrier between the design of the interactions and the incorporation of hardware devices, thereby allowing one to design the interactive experience directly without having to worry explicitly about the underlying hardware constraints. Using this system, equipment can be dragged and dropped onto a layout window. It is then virtually cabled, allowing the system to open appropriate drivers and the correct ports. Detailed cabling and setup instructions are generated as an `.html` file to be used to assemble the entire installation.

For instance, the interaction experiences with the drum crystals and tube worms, as well as the control of music and lighting effects, are implemented as a set of connected modes. Each mode contains an appropriate mapping of the sensory inputs to the special effects outputs. Interactive building blocks are dragged and dropped onto a mode window to be used in a graphical patch. Input and output parsers are automatically placed in the mode window based on the hardware layout. Each mode can have two channels of audio associated with it. The modes are placed into a transition graph that defines the paths through the interaction experience where multiple transitions can be defined for each mode.

4 The Vision System

The nature of the interaction environment at SIGGRAPH presents some unique challenges for using vision as a principal sensor for perceiving and interacting with people. The visual algorithms need to be robust enough to cope with unknown numbers of people at once, all of which may be entering and leaving the field of view at will, ranging in biometric characteristics such as height and race, clad in in possibly unpredictable ways. Furthermore, the processing has a strict real-time constraint, as any delays in the robot's reactions would immediately condemn the interaction as inorganic and unconvincing. The overall performance of the vision system was 11-15 Hz, depending on the number of people within the field of view.

The anemone responds to audience members either by orienting toward them and tracking their hand or face, or by recoiling from them in a fear response if their reach comes too close. Hence, the robot needs to perceive the audience from a front-on perspective to accurately orient itself towards features of interest. It also needs to determine the nearness of these objects to react to invasions of its "personal space". To address these issues, the real-time vision system consists of a pair of fixed-baseline stereo cameras. One camera is mounted behind the robot facing the audience and the other is located overhead looking down at the terrarium.

The vision system consists of several fast modules that perform model-free low-level feature detection and a tracking system that merges the information from these subsystems (see Figure 2). Although individually limited, each module is robust within their own narrow scopes. Furthermore, these low-level functions reinforce one another by searching for conjunctions of multiple features.

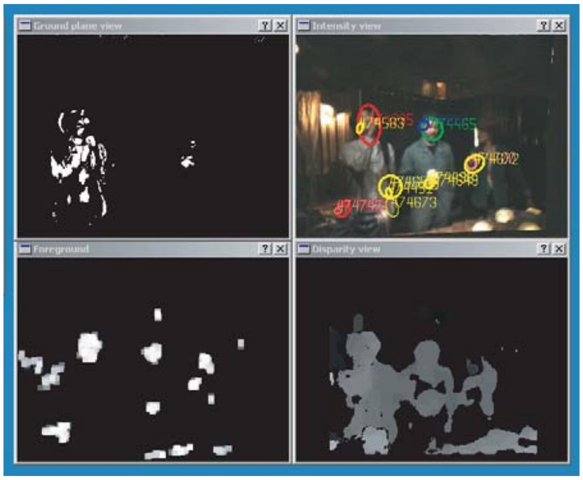

Figure 2: A snapshot of the stereo vision system facing the audience. Motion, human skin chromaticity, foreground depth map and the tracking faces/hands are shown in the image moving counterclockwise from the upper left frame. The vision system was developed in collaboration with David Demirdjian of the Vision Interface Group at the MIT Computer Science and Artificial Intelligence Lab, and is an extension of their earlier systems (Demirdjian et al., 2002; Darrell et al., 2001).

A stereo correlation engine compares the two images for stereo correspondence, computing a 3-D depth (i.e., disparity) map at about 15 frames per second. This is compared with a background depth estimate to produce a foreground depth map. We performed background subtraction on the depth map (instead of raw intensity-based background subtraction) to combat false positives in the resulting foreground image due to time-varying illumination (Darrell et al., 1998). This technique is used on the disparity map—which is produced by performing greyscale *sum of absolute differences* (SAD) stereo correlation on the stereo image pairs (Konolige, 1997)—to produce a binary foreground mask. This is used to screen out background objects and visible portions of the terrarium from further consideration when analyzing the image for human flesh color in the next stage of hand and face extraction. Furthermore, the stereo depth image was used to compute an average z-coordinate for any segmented regions.

Detection of human flesh was achieved directly from one of the raw color images produced by each stereo pair on a given frame. In a normalized color space, human flesh has been shown to have a characteristic chromaticity that is fairly robust to variance in illumination and racial factors, if captured with a sufficiently high quality sensor (Brooks, 2000). The CCD pickups in digital cameras are not ideal for this, but they proved usable with a brief on-site calibration. We employed a trained, unimodal bivariate model of human flesh color in the red and green channels of the intensity-normalized color space similar to the approach described in Habili et al. (2001). The blue channel is unnecessary after intensity normalization as all three channels sum to 1. The probability of a pixel corresponding to human flesh was computed according to its distance from the channel means, and the resulting probability map is smoothed and regions of high probability are extracted. If these regions corresponded to foreground areas, they are selected as candidate hands or faces.

For representational efficiency, each candidate region is then parameterized with an optimum bounding ellipse, allowing it to be represented as a feature vector of x, y and average z location, major and minor axis length, and rotation angle. A binary face classification (Viola & Jones, 2001) is run within the bordered boxed bounds of the ellipse. Finally, an activity score is obtained by summing its intersection with the difference image resulting from subtraction of the previous frame. These abstract “blob” representations are then tracked from frame to frame.

Tracking was achieved in a forward-chaining, predictive fashion using a linear motion model. Motion models for non-kinematically-constrained human movement are notoriously unreliable due to the apparent nonlinearity of the end effector motion when viewed in isolation, but acceptable results were achieved with a second order model. It would have been preferable to include a stochastic component, as in for example particle filters based on (Isard & Blake, 1998), but the overhead of tracking up to three blobs for each person (with possibly up to ten individuals in the scene at once) precluded such an approach. Previous time step detections had their feature vectors perturbed according to the motion model, and were then matched against detections in the current frame. Provision was made for non-immediate decay of unmatched detections, to allow for recovery from occlusion or other temporary detection failure. A reliability measure was computed for all object trajectories. If this measure is over a stability threshold, the perceptual information for these objects are passed to the robotic characters and used as their sensory input.

5 The Robot Performer

The principle robotic performer is the Public Anemone robot. We designed Public Anemone to have an organic appearance and natural quality of movement. The robot is highly articulated, having 13 degrees of freedom consisting of 8 body stages and 5 tentacles. Each body stage is rotated 90 degrees relative to the stage below with servomotors mounted at each stage. This gives the robot a broad range of motion with the ability to precisely control each stage. The body motors are equipped with a combination of optical encoders and potentiometers, enabling both absolute position sensing and precise control via a custom high-density motor control platform Hancher (2003). To achieve a smooth quality of motion by avoiding the effects of cumulative backlash from the planetary gearboxes, torsional springs at each stage preload every gearbox against one side of the backlash region. The robot’s skin is made of a highly elastic silicone rubber and is designed to fit over the mechanics in a way that maximizes the realism of motion.

5.1 The Behavior Engine

The high-level behavior of the robot is controlled by the *C4 System* developed by the Synthetic Characters group at the MIT Media Lab. This behavior-based A.I. code base, inspired by models of animal behavior, has traditionally been used to create graphical creatures that interact with and learn from people (Isla et al., 2001). Together, we adapteded it to the task of controlling a robot for this project, using the animation capabilities as a simulator.

Within the simulator, the anemone is represented within a small virtual world. Hand and face data from the stereo vision system are used to position virtual faces and hands in the anemone’s simulated world. When the anemone receives sensory data from the physical to its virtual world, this information is passed through the anemone’s *percept tree* that contains nodes designed to extract any features from the data that may be relevant to the anemone. For example, a waving hand triggers the “highly active hand” percept, while a hand that approaches too close to the anemone fires the percept that identified “threatening hands”. Features that arise from the same object in the world are bundled together. On the next cycle, when a new group of percept evaluations are produced by the percept tree, the anemone updates its information about objects that are already present. Otherwise a new group is formed in the robot's memory and is attributed to a new object. These groups of percept evaluations form a large part of the anemone's persistent model of the world.

In keeping with models of animal behavior, the robot has several behaviors that compete for activation based on the robot's memory, drives, and the perceptual contributions coming from the vision system. The drives of the anemone relate to its daily chores; it needs to "water" some adjacent plants by sprinkling water from the nearby pond on them, it needs to "drink" from the pond, and it needs to "wash" in the waterfall. Depending on the level of these drives and the activity of participants, the anemone will choose either to engage the participants or work on its chores. Audience participants compete for the anemone's attention to have the robot track their movements — active participants are more interesting to the anemone, so they can draw its attention away from chores more easily. Aggressive participants, however, move in a threatening manner and evoke the anemone's fear response that interrupts its other activities so that the robot can protect itself.

5.2 Generating Natural and Expressive Movement

Once the action to be taken is decided upon, the anemone's motor system attempts to bring the action about in a fluid way. To present a life-like quality of motion, the movements of the anemone are based on handmade animations made by professional computer animators. We created a scale 3-D model of the anemone in a computer animation toolkit. Once completed, the model and the animations are imported into the C4 System; the model becomes the virtual body of the anemone that is situated in the same world as the virtual hands and faces.

These animations are treated as data, and are combined in various ways, to generate the robot's joint angle trajectories over time. The simplest combination is layering, where different parts of the robot are controlled by multiple independent animations. For instance, a separate animation for moving the tentacles may be combined with another animation that controls the body. Animations that control the same joints on the robot are blended together to produce a new animation that is a combination, on a per-joint basis, of these input animations.

An arbitrary number of dynamically weighted source animations can be blended together to produce the actual movement of the model. The technique is based on the *verb-adverb* technique of Rose et al. (1998). All of the joints in the virtual model are represented by quaternions. This allows the blending to take place in quaternion space (Johnson, 2002), providing much more appropriate and stable results than blending in Euler space.

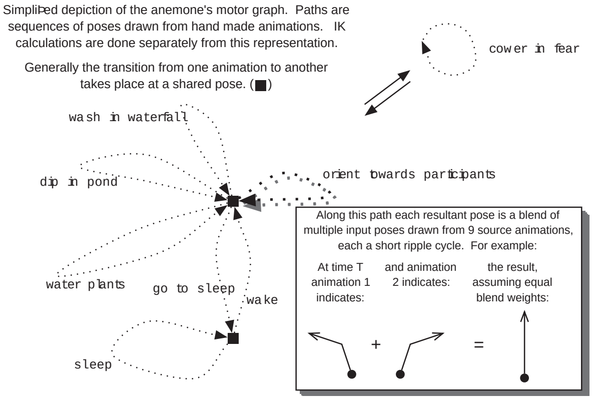
Simplified depiction of the anemone's motor graph. Paths are sequences of poses drawn from hand made animations. IK calculations are done separately from this representation.
Generally the transition from one animation to another takes place at a shared pose. (■)
cower in fear
wash in waterfall
dip in pond
orient towards participants
water plants
go to sleep
wake
sleep
Along this path each resultant pose is a blend of multiple input poses drawn from 9 source animations, each a short ripple cycle. For example:
At time T animation 1 indicates:
and animation 2 indicates:
the result, assuming equal blend weights:
+
=

Figure 3: The pose graph of the anemone which represents all the animation paths between the robot's various behaviors.

The representation of an animation in the system is an ordered list of poses. These *pose lists* are linked together to form a directed graph of poses through which the anemone travels, called a *pose graph* (Downie, 2000). This representation allows the system to generate a path from one pose (i.e., a particular configuration of the model joints) to another pose by traveling along paths that the animator intended (see Figure 3). In general, the movement of the joints in the model are restricted to these paths, some of which are blends of multiple animations.

However, for certain types of activity, a more immediate transition strategy is called for. For instance, the fear response of the anemone is an animation that has it cowering defensively. This response needs to happen immediately upon receiving input about a threatening object. Hence, animations must be able to be interrupted in a way that looks natural. Creating hand animated transitions from every possible position of the anemone to the "cower" animation would be untenable. As a result, a programmatic blend is used. This mechanism uses simple per-joint interpolation to change the joint angles from their current angle to the positions required for the "cower" animation. A straight programmatic blend from one position to another turned out to look fine for a quick dart like this, but not for other, more prolonged actions, such as orienting towards participants.

Orienting towards participants and tracking their movements is the primary way the anemone conveys that it is interacting with people. This behavior is exhibited frequently and must be performed in a life-like way. Our animators produced a set of animations depicting the anemone orienting at a particular location while rippling its body segments. We call this an *active hold*. We want to maintain this style of movement while the anemone orients to the participant's location within its visual field and smoothly tracks them. We chose a strategy that combined animation blending with a small element of inverse kinematics (IK) for minor corrections of the top two stages of the robot (i.e., those closest to the tentacles which might be interpreted as being the anemone's "head").

Specifically, the animators created nine orientation animations; in each the anemone ripples its body in place while orienting towards a different location in space. These orientation poses form a 3×3 grid that covered the field of view of the robot. The rippling phase of the anemone in each of these different positions was synchronized in time so it would not be canceled out by blending animations with opposing ripples. All but the final stages (those directed by IK) are controlled by a programmatic blend of the 4 animations closest to the target orientation. By varying the weights and the animations involved, the anemone can track an object smoothly throughout its space while maintaining its characteristic ripple. In this mode, the animations provide a certain quality to its movements (i.e., the rippling) while the actual task-oriented movement (i.e., the tracking) is controlled solely through the blend weights of the animations. Finally, since the 9 orientation animations were mapped roughly by hand into the coordinate system of the virtual robot, a cyclic coordinate descent algorithm was used to get a more exact orientation in the final stages of the robot (Wang & Chen, 1991). After some experimentation, it became clear that only the final two stages were necessary for this correction (see Figure 4).

This discussion has been about how to move the virtual anemone model; however, since the model was designed with the same scale and articulation as the actual robot, driving the robot is as simple as translating the rotation information from the model into values for the physical motors. For each single degree of freedom joint, its quaternion representation is converted into an axis/angle representation (in radians), converted to encoder ticks, and sent to the motors at 50 times per second. The motor controllers are directly connected to the computer running the behavior via a serial connection that allows for two way communication, so the behavior system has the option of using the position data from the actual robot.

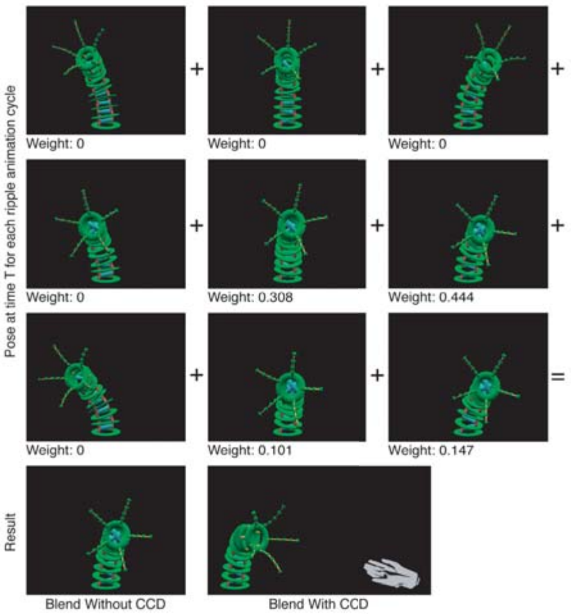

Figure 4: The motion blending of the simulated anemone for its orientation motor skill. A single frame of each of the nine active-hold animations in different orientations are shown above. One frame of the resulting weighed blend from the contributing animations (lower left) generates an active-hold in approximately the desired orientation. The final, more accurate orientation after layering IK on the top two stages (lower right).

The scale of the anemone's model in its virtual environment and the transformation of coordinates of incoming vision data were carefully determined to correspond to the actual scale and coordinates. Hence, when the virtual version of the anemone orients to virtual hands in its simulated world, the actual robot orients towards the physical hands in the real world. Once this link was set up between the virtual anemone and the real robot, the animated version was very helpful for debugging and development without risking damage to the actual robot during the preliminary stages.

Results of Robot Evaluation Survey		
Quality	Mean	Std.Dev.
Likable	5.8	1.5
Engaging	5.5	1.2
Responsiveness of Terrarium	5.4	1.3
Liveliness	5.4	1.3
Reponsiveness of Robot	5.2	1.3
Personal	5.1	1.2
Naturalness of Interaction	5.2	1.5
Engaging of Senses	5.2	1.5
Smiled Often	5.3	1.7

Table 1: Summary of results from the questionnaire. Participants evaluated each quality on a 7-point scale (1 weak, 7 strong).

6 Evaluation

The robotic terrarium was exhibited for 5 days, 8 hours per day, as part of the SIGGRAPH 2002 Emerging Technologies Exhibit to hundreds of visitors. Often large groups of people interacted with the installation at once, crowding around the terrarium.

To evaluate the audience's reaction to our installation, we asked participants to fill out a survey after they interacted with the terrarium. Each question asked to rate the experience on a 7-point scale (1 weak, 7 strong). Most of the survey questions focused on the evaluation of the Public Anemone robot, however, a few addressed the interaction with the installation as a whole. A total of 126 surveys were completed by 79 men, 45 women, and 2 who did not state their gender. The participants ranged in age from 16 to 67 years. A diverse occupation background was represented including people from academia, the arts (e.g., music, animation, etc.), engineering (e.g., video game designers, etc.), and the media (e.g., TV and film). Overall, the mean scores reflected generally positive scores for all questions on the survey, though some more strongly positive than others. The most notable results are reported in Table 1.

7 Conclusion

Although this work targets robots in the entertainment domain, it has practical implications as well. First, it presents an interesting test scenario for exploring challenging research questions in the development of sociable robots. One day, such robots will serve as our teammates and assistants in a variety of domains (including health, domestic, scientific, military, and educational venues). Within the scope of this paper we address two core challenges: real-time perception of multiple people in a highly dynamic and challenging venue, and the generation of lifelike and expressive movement for a highly articulated autonomous robot. Second, this work dovetails two important computing trends of the twenty-first century, autonomous robots and intelligent environments, to explore how autonomous robots might exploit the infrastructure of embedded computing environments (i.e., the intelligent stage in our case) to support their ability to richly interact with people and the world around them.

Acknowledgements

This installation never would have made it to SIGGRAPH 2002 if it were not for the dedicated work and long hours of many others. Jeff Lieberman, Heather Knight, Dan McAnulty, Adam Kraft, and Aileen Kawabe of the MIT Media Lab's Robotic Life Group were of great help in the construction and shipping of this installation. The vision system was developed in collaboration with Trevor Darrell and David Demirdjian of the MIT CSAIL's Vision Interface Group. Mike Jones granted permission to use the Viola-Jones face detector. Many thanks to Bruce Blumberg, Matt Berlin, Bill Tomlinson, and Marc Downie of the MIT Media Lab's Synthetic Characters Group for helping us to adapt their interactive characters code base to control our robot. Geoff Beatty and Ryan Kavanaugh provided animation and graphic arts support. Funding for this project was provided by the Media Lab's *Things That Think* and *Digital Life* Consortia.

References

Breazeal, C. (2002), *Designing Sociable Robots*, MIT Press, Cambridge, MA.

Brooks, A. (2000), Mechanisms and Processing Techniques for Real-Time Active Vision, Master's thesis, School of Information Sciences and Engineering, Australian National University.

Bruce, A., Knight, J. & Nourbakhsh, I. (1999), Robot Improv: Using drama to create believable agents, *in* 'AAAI Workshop Technical Report WS-99-15 of the 8th Mobile Robot Competition and Exhibition', pp. 27–33.

Darrell, T., Demirdjian, D., Checka, N. & Felzenszwalb, P. (2001), Plan-view trajectory estimation with dense stereo background models, *in* 'Proceedings of ICCV01', Vancouver, Canada.

Darrell, T., Gordon, G. & Woodfill, J. (1998), Integrated person tracking using stereo, color, and pattern detection, *in* 'Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition', Santa Barbara, CA.

Demirdjian, D., Tollmar, K., Koile, K., Checka, N. & Darrell, T. (2002), Activity maps for location-aware computing, *in* 'Proceedings of IEEE Workshop on Applications of Computer Vision (WACV2002)', Orlando, FL.

Downie, M. (2000), Behavior, Animation, and Music: the Music and Movement of Synthetic Characters, Master's thesis, MIT Media Lab.

Fong, T., Nourbakshs, I. & Dautenhahn, K. (2002), 'A survey of social robots', *Robotics and Autonomous Systems* **42**(3-4), 143–166.

Habili, N., Lim, C. & Moini, A. (2001), Hand and face segmentation using motion and color cues in digital image sequences, *in* 'Proceedings of IEEE International Conference on Multimedia and Expo (ICME 01)'.

Hancher, M. (2003), A Motor Control Framework for Many-Axis Interactive Robots, Master's thesis, MIT Department of Electrical Engineering and Computer Science.

Isard, M. & Blake, A. (1998), 'Condensation: conditional density propagation for visual tracking', *International Journal of Computer Vision* **29**(1), 5–28.

Isla, D., Burke, R., Downie, M. & Blumberg, B. (2001), A layered brain architecture for synthetic characters, *in* 'Proceedings of IJCAI 2001'.

Johnson, M. (2002), Exploiting Quaternions to Support Expressive Interactive Character Motion, PhD thesis, MIT Media Lab.

Kitano, H., Tambe, M., Stone, P., Veloso, M., Coradeschi, S., adn H. Matsubara, E. O., Noda, I. & Asada, M. (1997), The Robocup 1997 synthetic agents challenge, *in* 'Proceedings of the First International Workshop on RoboCup, IJCAI-97', Nagoya, Japan.

Konolige, K. (1997), Small vision systems: hardware and implementation, *in* 'Proceedings of the Eighth International Symposium on Robotics Research', Hayama, Japan, pp. 203–212.

Rose, C., Cohen, M. & Bodenheimer, B. (1998), 'Verbs and adverbs: multidimensional motion interpolation', *IEEE Computer Graphics and Applications* **18**(5), 32–40.

Strickon, J. (2002), Smoke and Mirrors to Modern Computers: Rethinking the Design and Implementation of Interactive, Location-Based Entertainment Experiences, PhD thesis, MIT Media Lab.

Viola, P. & Jones, M. (2001), Rapid object detection using a boosted cascade of simple features, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition', Kauai, HI, pp. 511–518.

Wang, L. & Chen, C. (1991), 'A combined optimization method for solving the inverse kinematics problem of mechanical manipulators', *IEEE Transactions on Robotics and Automation* **7**(4), 489–499.

Werger, B. (1998), 'Profile of a winner: Brandeis University and Ullanta performance robotics "Robotic Love Triange"', *AI Magazine* **19**(3), 35–38.

Wood, G. (2002), *Edison's Eve: A Magical History of the Quest for Mechanical Life*, Alfred A. Knopf.