Proceedings of the 17th IEEE International Symposium on Robot and Human Interactive Communication, Technische Universität München, Munich, Germany, August 1-3, 2008

A Hybrid Control System for Puppeteering a Live Robotic Stage Actor

Guy Hoffman, Rony Kubat, and Cynthia Breazeal

Abstract— **This paper describes a robotic puppeteering system used in a theatrical production involving one robot and two human performers on stage. We draw from acting theory and human-robot interaction to develop a hybrid-control puppeteering interface which combines reactive expressive gestures and parametric behaviors with a point-of-view eye contact module. Our design addresses two core considerations: allowing a single operator to puppeteer the robot's full range of behaviors, and allowing for gradual replacement of human-controlled modules by autonomous subsystems.**

We wrote a play specifically for a performance between two humans and one of our research robots, a robotic lamp which embodied a lead role in the play. We staged three performances with the robot as part of a local festival of new plays. Though we have yet to perform a formal statistical evaluation of the system, we interviewed the actors and director and present their feedback about working with the system.

I. INTRODUCTION

Robotic stage performers have been few and far between. Most work has dealt with fully scripted or extremely simple behavior on one end of the spectrum (for a good review, see Dixon [1]) or on the other, fully teleoperated robots such as the recent production of *Heddatron* [2]. In other work, robots have been partnered with each other on stage without the inclusion of a human scene partner [3]. Still, it is safe to say that fluent theatrical dialog between an autonomous robot and a human scene partner is still an unattained goal.

Robots have held a more significant part in film production, where generally analog-controlled animatronic puppets have played character roles. These systems traditionally employ a controller for each degree-of-freedom (DoF), and require not only expert rehearsal and multiple takes, but also the use of camera techniques such as carefully selected angles and editing to make up for the lack of interactivity between the robot and the human actors. In particular, as elaborated below, eye contact and precise inverse kinematics (IK) are impossible for these direct-drive puppeteering systems.

In the field of on-screen (non-robotic) performance characters, several systems were developed: Pinhanez used an extension of interval algebra to stage synthetic character performances including interrelations between the characters' action durations and timing, as well as methods to parse a human scene partner's actions into a scripted scene [4], [5]. Becker, Wren, Pentland, and others have used top-down gesture recognition techniques to create multimodally perceptive and expressive virtual environments [6], [7], among them *TheaterSpace*, a perceptive dance and theater stage for single performers.

Blumberg *et al.* devised a more comprehensive cognitive architecture for synthetic screen characters which were used—among others—to produce interactive performances [8], [9], [10]. Portions of this work have been conceived and extended by Downie and, in recent years, supported a large number of interactive musical, dance, and abstract graphical performances [11]. However, all of the above work used on-screen animated characters, and did not deal with the challenges of a physically situated robotic stage actor.

We have developed a hybrid control system aimed for rehearsal and production of live stage performances of robots acting with humans. This system is intended to allow a single operator to control a robotic actor using pre-animated gestures and sequences, but at the same time adapting to the rhythm of live performance to the human actors. The result permits the robot to be both expressive and responsive to its scene partner. We draw on lessons from human-robot interaction and film animatronics to purvey animacy to the robot, and we use camera-based feedback and IK to allow the robot to make eye contact with actors.

A core consideration in the design of this system is enabling the various puppeteering components to be exchanged with autonomous modules, eventually allowing the robot to become completely autonomous.

II. HYBRID CONTROL PUPPETEERING

The challenge of designing a system to control a live robot interacting on stage with human actors is to enable the robot to be both expressive and responsive. Most existing systems fall on one extreme of the scripted/direct-drive spectrum:

On one side is direct control of the robot's DoFs through digital or analog controls. Such systems are often used on film sets. They usually require lengthy rehearsal times and are frequently operated by more than one puppeteer. In addition, eye contact with an actor is virtually impossible, due to the fact that more than one operator has access to the joint chain leading to the eye DoF, and the resulting impractical level of coordination required for eye contact. Very simple systems with a limited DoF count, are also sometimes direct-controlled.

On the other side of the spectrum are fully-scripted multi-joint articulated animatronics, common, for example, in theme parks. These systems don't usually require any operator intervention, but since their motion is completely predetermined, there is no possibility for changes of timing, responsiveness to human actors, or alteration/improvisation in performance. In the rare cases where there is a human

G. Hoffman, R. Kubat, and C. Breazeal are with the Media Laboratory, Massachusetts Institute of Technology, 20 Ames Street, Cambridge, MA 02142, USA {guy,kubat,cynthiab}@media.mit.edu

978-1-4244-2213-5/08/$25.00 ©2008 IEEE

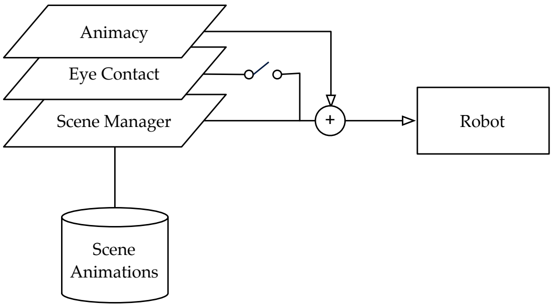


Fig. 1. Schematic layout of the hybrid puppeteering architecture.

scene partner, the complete onus of collaborative behavior and timing is on the human.

As a first step towards a fully autonomous robotic actor, we have developed a hybrid control puppeteering system, which consists of components that not only overcome many of these restrictions, but are designed to be gradually replaced with autonomous processes (see: Section V). The system consists of three layers which are combined to generate the robot's behavior (Figure 1).

A. Scene Manager

The base narrative layer is structured around the play's *scenes*. A scene is a sequence of short *beats*, each of which describes a gesture on the robotic character's part.[1]

To allow for complex gesture expressiveness, a scene is animated in a 3D animation software, using a physically structured model of the robot. This results in a sequence of positions for the robot throughout the scene, broken into "frames". We denote a frame at scene time i in scene s as the column-vector of joint configurations $\mathbf{q}_i^s$. A custom-written exporter to the animation program exports the robot's DoF positions in radians for each of the frames in the scene, which are saved in the scene animation database.

Next, beats are identified and delimited in each scene. A beat is defined by an onset frame and end frame. During performance, a beat is expressed in two parts: the *impulse* and the *cue*, two terms borrowed from acting method: "[T]he impulse comes early in the speech, and the cue then plays that out." [12] The beat's impulse is the preparatory behavior of the character, which happens before the character's cue to perform an action, as an initial reaction to the scene partner's action. In order to support this in our system, a beat is assigned two speeds, in frames per second, for the impulse-to-cue, and cue-to-end parts of the beat.

For example, a beat may run between frame 20 and frame 65 of the animation, with a 2fps impulse speed, and a 20fps cue speed. When the impulse of that beat is triggered the animation runs at 2fps from frame 20 until the cue it struck. At that point, the animation accelerates to 20fps, a speed that is maintained until the end of the gesture. If the desired frame rate is below the originally animated frame rate, we perform a linear interpolation of the joint positions. The result of this impulse-to-cue architecture is to prevent a stop-and-go delayed performance on the robot's part, and allowing for a fluent exchange of movement on stage.

The triggering of impulses and cues thus maps real performance time t to scene-based frame time t_f. Below, we will refer to this temporal mapping function as $t_f = M(t)$. Thus, at performance time t, the scene manager produces the joint configuration $\mathbf{q}_{M(t)}^s$.

Note that the complete scene is designed as a single animation to prevent discontinuities in the robot's movement, as each beat flows into the next.

B. Eye contact

The second control layer is responsible for making eye contact with human actors. Traditionally, in film animatronics, eye contact between human actors and all but the simplest figures was virtually impossible. This is because if more than one puppeteer controls the robot, the end effector state (usually the eyes) is dependent on the motion of more than one operator. We overcome this restriction with the eye contact layer of our system.

This layer computes the robot's inverse kinematics using Cyclic Coordinate Descent [13] for the eye end effector, pointing it towards an arbitrary 3D position. This results in a joint configuration $\mathbf{c_t}$ at time t. The 3D position is determined by the operator clicking on a cylindrical projection of the space surrounding the robot (The white box below the robot's video POV in Figure 2). The operator also views the scene through a long focal point camera mounted in the robot's eye. This narrow-field camera can be used to fine-adjust the eye contact, by controlling to keep the gaze target centered in view. The mapping from the 2D location on both the cylindrical projection view and the narrow camera views to the 3D gaze target in the robot's coordinate space is learned by training a mixture of gaussian model with labeled data.

The eye-contact IK does not necessarily involve all the DoFs of the robot. We denote the set of IK-related joints in an m-DoF robot by the binary vector

$$\mathbf{e} = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_m \end{pmatrix}$$

where

$$\epsilon_i = \begin{cases} 1 & \text{if DoF } i \text{ is part of the IK solution} \\ 0 & \text{otherwise} \end{cases}$$

C. Animacy

An animacy layer ensures the robot is never completely still. It resides above the scene and eye-contact layers, which make up the major motor activity. Eschewing stillness follows from our experience with theater practice, socially expressive robots, and synthetic characters. Theater practice prescribes continuous internal activity even when the actor does not have stage or line instructions at the moment: "There's always some physical expression of internal states,

[1] The nomenclature of "beats" and "scenes" used here, though borrowing from the vocabulary of theatrical practice, is distinct.

even if it's the movement of a finger;" "If you stop thinking as the character, the character is dead." [14].

The animacy layer is implemented as an additive smoothed-noise sinusoidal movement of the robot, akin to breathing. The motion is influenced by two parameters of frequency f and amplitude α, setting the extent of the offset from the scene-prescribed position of the motor. We thus denote the instantaneous additive component to the robot's joint positions as $\mathbf{a}_t$.

D. Arbitration

The motor position for each joint is composited as follows: first the Scene Manager sets the position for each of the robot's DoFs. Then, if the eye-contact layer is active, it overrides the DoF position for the DoFs needed for IK. Finally, the animacy layer offsets the computed position based on its own position.

More formally, using the above notation, we derive the instantaneous configuration of the robot $\mathbf{p}_t$ at time t during scene s, as follows. First let $\lambda \in [0, 1]$ be the extent to which the eye-contact IK module is activated. Then $\mathbf{p}_t$ is given as:

$$\mathbf{p}_t = \mathbf{q}^s_{M(t)} \cdot (1 - \lambda)\mathbf{eI} + \mathbf{c}_t \cdot \lambda\mathbf{eI} + \mathbf{a}_t$$

Eye-contact is activated whenever a position is selected by the operator. It is disabled whenever a new impulse is triggered, if the beat of this impulse is marked as "disabling eye contact". This distinction is important because some of the beats only include degrees of freedom that are not related to the IK. For example, some beats in this production only change the color of lights which are part of the robotic actor, and therefore should not interrupt the eye contact. Finally, in order to prevent motion discontinuities, the control of the eye contact module is faded in and out with a linear fade (the above-mentioned λ).

III. USER INTERFACE

Figure 2 shows the puppeteer's user interface, designed for single-operator live performance. The screen is divided into three parts:

Along the top is a status bar which indicates—left to right—the currently loaded scene (in this case: "Scene 1"), the description of the currently running beat ("Notice F") and whether this beat disables eye contact ("Y(es)"). To the right, sliders indicate the scene-based frame position of the scene manager.

The center contains the operator's action controls. It is also divided into three parts: to the far right, the impulse/cue button advances the scene manager's beats. This button, when it is pushed down, triggers the next beat using the impulse frame rate. When released it switches to cue mode and continues to advance the beat at the cue frame rate. This spring-loaded behavior makes sure that the robot is never left in 'impulse' mode.

The large white box at the bottom left of the action control area is the cylindrical projection of the space for large-step eye contact movements. It covers the entire eye contact workspace. The system defines a 3D position for the eye contact IK when the box is clicked. In the case of a statically mounted robot, a projection of the robot's surrounding can be positioned in this view. This control is used mainly for large movements, directing the robot's gaze towards areas which are outside of the camera view. When this control is clicked, the eye-contact IK module is activated.

The operator can see "through" the robot's view in the top-left corner's camera window. Clicking in this window refines the robot's gaze direction. This control can be used to enable closed-loop feedback to keep the scene partner's face centered.

In the top-center of the control section, a toggle button indicates whether eye-contact IK is currently active. It can also be used to manually disable eye-contact. Below are sliders to control the two animacy parameters, α and f.

The bottom third of the screen is taken up by a 3D model of the robot which incorporates all of the control layers and enables the operator to see the robot's full configuration. This segment is invaluable both for debugging, which can be done without using the physical robot, and if the robot is occluded from the operator's view. The slider at the bottom enables the operator to rotate the 3D view of the robot.

Finally, the second window is the Scene Viewer, showing the impulse and cue speeds. The window also contains a flag indicating whether the beat disables eye-contact IK for each of the beats in the scene. In this window, the operator can change the impulse and cue speeds for the current scene on the fly.

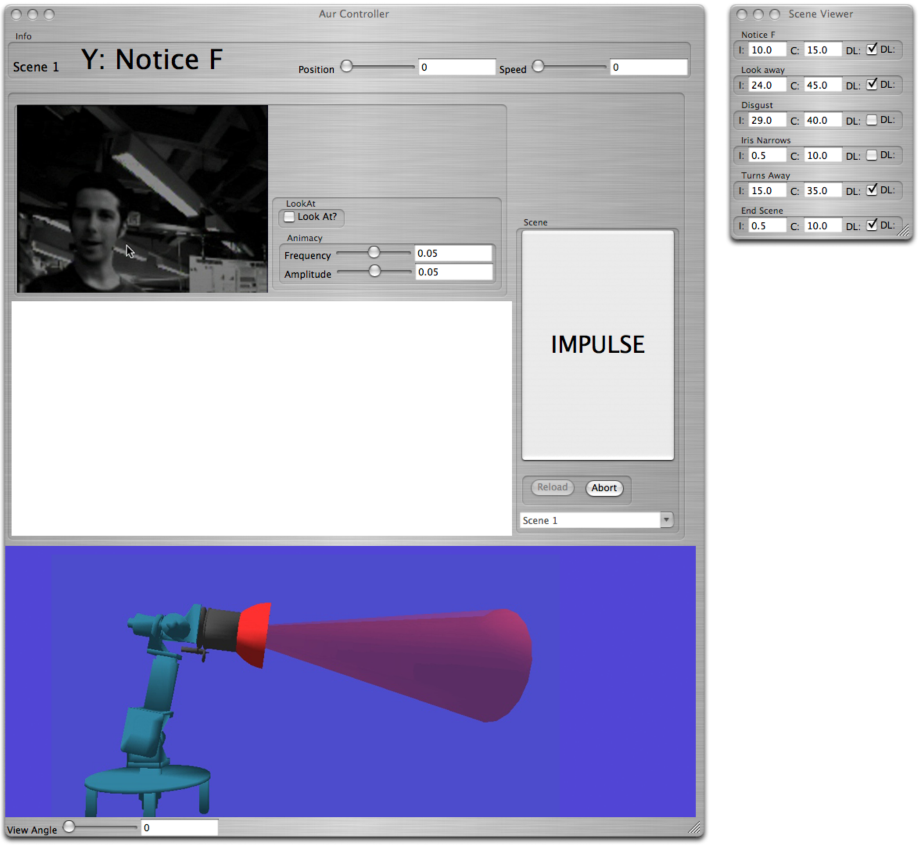
In order to allow a single operator to control the robot, we have mapped the impulse/cue spring-loaded button to an external device with a push-button. During live performance, the operator can thus use the push-button device in one hand to trigger the beat impulses and cues, and the mouse to control the eye contact and animacy parameters.

IV. PRODUCTION

We staged a theater production using the above-mentioned system in three live performances in front of an audience of roughly 50 each night. The performed play was entitled *Talking to Vegetables*.

As the character named "The Confessor", the robot plays foil to René and Fossarius, two human characters struggling with guilt from the death of a beloved friend. In paired scenes, both human characters come independently to the robot to make a confession. The robot—though its physical gestures alone—implores, comforts and accuses the human characters, eliciting a deeper reaction and driving forward the story. Words given to each human character are nearly identical, though each follows a unique arc, driven largely by the Confessor's reaction to the monologues.

The play was written to fit the strengths and weaknesses of the specific robotic platform. An early decision was to make the robotic character mute and to focus performance on its gestural, rather than verbal vocabulary. The reasons for this choice were both technical and artistic. We felt that current speech synthesis technology lacks the nuance necessary for live performance, especially when reacting to



Scene 1 Y: Notice F

Position 0 Speed 0

LookAt

□ Look At?

Animacy

Frequency 0.05

Amplitude 0.05

Scene 1

View Angle 0

Notice F

I: 10.0 C: 15.0 DL: ☑ DL:

Look away

I: 24.0 C: 45.0 DL: ☑ DL:

Disgust

I: 29.0 C: 40.0 DL: □ DL:

Iris Narrows

I: 0.5 C: 10.0 DL: □ DL:

Turns Away

I: 15.0 C: 35.0 DL: ☑ DL:

End Scene

I: 0.5 C: 10.0 DL: ☑ DL:

Fig. 2. Hybrid control puppeteering user interface. The top bar in the main window (left) shows the currently loaded scene and beat, as well as the frame rate and position within the beat. Below is the robot's point-of-view camera, used for the closed-loop feedback of the eye-contact IK. To the right of the camera view are the eye-contact trigger and animacy parameter sliders. Below, the full-stage eye-contact IK controller; to the right—the beat trigger controls and scene loader. The bottom of the main window shows the 3D real-time simulation of the robot. The small window to the right displays a list of all beats in the current scene, along with their associated frame rates and IK override states.

a specific action of a human player. The idea of using a human voice actor to perform any vocalization of the robot was also struck because we wished the robotic character to be distinctly non-human. Although a work of science fiction (we don't have robotic confessionals yet), we wanted the play to reflect a plausible reality in which robots are social companions which can react to human gestures and emotions. We believe that a world where robots can maintain meaningful verbal conversations with human companions falls more in the domain of implausible science fantasy.

A. Robotic Platform

The robotic performer used in *Talking to Vegetables* is *AUR*, a robotic desk lamp [15]. AUR has a five DoF arm ending with an LED lamp which can illuminate in a range of the red-green-blue color space. A variable aperture can change the light beam's width. AUR is stationary and mounted on a steel and wood workbench which locates its base approximately 90 cm above the floor. Figures 3 and 5 show a photo of the robot.

The robot arm is controlled using optical encoders and off-the-shelf motor control boards. The light aperture is position-controlled using a potentiometer and custom electronics. This iris changes the width of the light beam and changes the lamp's "facial expression". The color and intensity of the light is controlled with a DMX light controller. All three modules are interfaced to the main character software described below using a custom UDP/IP network protocol called the Intra-Robot Communication Protocol (IRCP) [16]. The main hybrid control software runs on a 2x Dual 2.66GHz Intel processor machine located underneath the workbench. For a hardware and software component layout of the system, see Figure 4.

B. Rehearsal and Performance

One month of rehearsals preceded performance. For three weeks, while the robot and software were being prepared, the cast rehearsed without robot, using a prop as stand-in for AUR. A single puppeteer (one of the authors) was used for rehearsals and performance, and was present for all rehearsals. During these early rehearsals, the actors and director discussed the gestures most appropriate for the robot to make. The last week of rehearsals incorporated AUR. In performance, the puppeteer was offstage and the robot operated in a "wizard-of-oz" mode.

Talking to Vegetables was performed as a part of a festival of new short plays. Because the performances took place

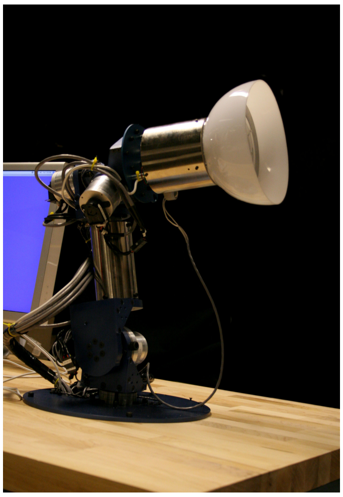


Fig. 3. AUR, the robot used in the play.

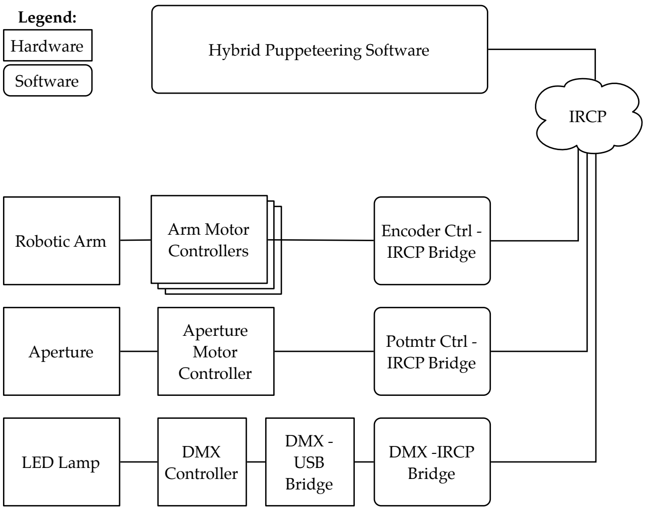


Fig. 4. Schematic layout of the robot's hardware and software systems.

outside the laboratory context and were incorporated into an ongoing festival with limited performances, we did not receive feedback from the audience by questionnaires or other means. A more formal evaluation of the system's performance, both in terms of user interface and importance of individual components, is left to future work. We did, however, solicit feedback from the cast and crew.

Reaction to the rehearsal process and performance was generally positive from both cast and director. Regarding the balance between pre-scripted gestures and the interactive gaze following, director Kate Snodgrass observed:

> It took some work on everyone's part to get this right (actors responding and [puppeteer] reacting), and I'd like to think that the performances went beautifully. It was always a matter of [the puppeteer] understanding what the play was saying (he asked questions like any other actor) and then incorporating a movement or gaze that might be interpreted as meaningful. I was very fond of the way the robot gazed at the actors and then followed their movements. For me, the most successful parts of the robot as actor were the gaze-following interactions. These movements made it seem as if the robot was listening to the actors, intent upon their reactions. Since we could not see the "face" of the robot (we could see colors change, but not the countenance), we could not gauge expression; therefore, the movements and the silences were paramount.

Even though the audience may not have been able to see the robot's "face," it did have an effect on the actors, who often faced the machine. Laurel Ruhlen (who played René) reported an effective robotic gesture: "The sudden narrowing of the robot's iris—kind of had the same effect as someone raising their eyebrows and/or squinting."

The robot successfully became a character in its own right. Laurel Ruhlen wrote, "The robot was weirdly adorable. I felt surprisingly protective of it." Snodgrass remarked, "I found myself thinking of the robot as a 'real' actor because it had expressions (at least, the movements conveyed this to me)." She continued:

> I know that [the actors] seemed as if they were really talking to the robot in the rehearsals, and a couple of times mid-way through the rehearsal process, when the robot was not reacting in a way that they could decipher, they asked [the puppeteer] to help the robot "understand". As to the audience, personally, I think they enjoyed themselves immensely. They smiled at first because the robot was "acting" and we're not used to a mechanical figure on stage. However, as the play went on, I think they forgot that the robot was being manipulated (if they ever realized this) and began to see the robot as another character in the play.

Actor Jonas Kubilius (Fossarius) noted a similar transition of the audience's reaction to the robot: "It seems to me that it was treated more like a toy (unfortunately), so both me and the audience were like 'wow, it's actually reacting as a human being'."

The time-consuming process of making the robot "understand"—adjusting gesture animations and exporting them into the performance software—was one of the difficulties encountered. Spontaneity was limited to eye-tracking and gesture experimentation was bounded by time. This criticism was reflected in Kubilius's comments: "I was excited to see how a robot could actually participate in a meaningful way in a production. But really, [AUR] did not affect me that much. I do not think I started treating it as a human being; rather, it was like an external trigger to whose actions I could respond." Ruhlen: "It was rather like working with a puppet. I had to tailor my delivery and reactions to match the robot; when you're working with other humans, you sort of meet each other halfway in that respect."

Another criticism from both director and actor was the

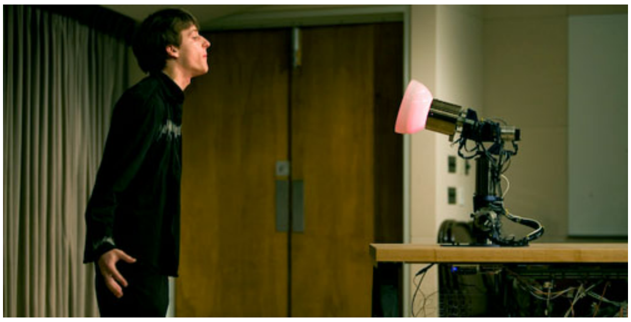


Fig. 5. Scene from a stage production employing the described hybrid puppeteering system.

robot's lack of mobility, which they believe hindered the robot's emotive capacity. Kubilius:

> I guess the problem was that there was this table attached to the robot—or, rather, the robot being attached to a massive table did not allow to connect more with the robot.

Asked whether the robot was a fluid performer, Snodgrass answered: "Yes, it was up to a point. The robot was stationary on a rolling table, and it could not move the way an actor can, crossing the stage on foot, turning, etc. But it was fluid in the sense that it was a realized character on stage who interacted with the other actors and who had a point of view."

V. CONCLUSION AND FUTURE WORK

We presented an approach and system for live robotic stage performers, a virtually untapped area of entertainment technology research. To allow for a performance that is both expressive and reactive to the robot's human scene partners, we developed a hybrid control system designed for use by a single operator in a live stage setting. This system combines dynamic triggering of pre-scripted animation, parametric motion attributes, and real-time point-of-view eye-contact IK, a previous unachieved task. We have staged a production of a play specifically written for a robot and two human actors, and performed it three times.

The system was modularly designed to increasingly be replaced by autonomous systems. Using motion- and face-detection techniques, the eye-contact module can be automated. Similarly, an emotional model, along the lines as the one described in [17] can be used—in conjunction with pre-scripted scene analysis and prosody detection—to drive the parametric attributes used in our system. Finally, a word-spotting and gesture-recognition system, such as [18], can be imagined to replace the triggering of the impulses and cues.

While a complete autonomous robotic stage acting is still ways off, we hope to have laid the groundwork for such an endeavor.

We also believe that stage performance can be a promising implementation platform and testing ground for many important ideas in human-robot interaction research. It is a relatively constrained yet rich environment in which a robotic agent meshes its actions with a human partner. Surprising as it may sound, robotic theater may prove to be a new "grand challenge" for fluent human-robot joint action, dialog, collaboration, and practice.

ACKNOWLEDGMENTS

We wish to thank Kate Snodgrass, Jonas Kubilius, Laurel Ruhlen, Emilie Slaby and the crew which made the staging of *Talking to Vegetables* possible. This paper is based upon work supported under a National Science Foundation Graduate Research Fellowship.

REFERENCES

[1] S. Dixon, "Metal Performance: Humanizing Robots, Returning to Nature, and Camping About," *The Drama Review*, vol. 48, no. 4, 2004.

[2] Les Freres Corbusier, "Heddatron," http://www.lesfreres.org/heddatron/, 2006.

[3] A. Bruce, J. Knight, S. Listopad, B. Magerko, and I. Nourbakhsh, "Robot improv: Using drama to create believable agents," in *Proceedings of ICRA 2000*, vol. 4, April 2000, pp. 4002–4008.

[4] C. Pinhanez, "Representation and recognition of action in interactive spaces," Ph.D. dissertation, MIT Media Lab, June 1999.

[5] C. S. Pinhanez and A. F. Bobick, ""it/i": a theater play featuring an autonomous computer character," *Presence: Teleoper. Virtual Environ.*, vol. 11, no. 5, pp. 536–548, 2002.

[6] D. A. Becker and A. Pentland, "Using a virtual environment to teach cancer patients t'ai chi, relaxation and self-imagery," MIT Media Laboratory, Tech. Rep. VisMod 390, 1996.

[7] C. Wren, S. Basu, F. Sparacino, and A. Pentland, "Combining audio and video in perceptive spaces," in *Proceedings of: Managing Interactions in Smart Environments (MANSE 99)*, Dublin, Ireland, December 1999. [Online]. Available: citeseer.ist.psu.edu/article/wren99combining.html

[8] P. Maes, T. Darrell, B. Blumberg, and A. Pentland, "The ALIVE system: full-body interaction with autonomous agents," in *CA '95: Proceedings of the Computer Animation*. IEEE Computer Society, 1995, p. 11.

[9] B. Blumberg, "(void*): a cast of characters," in *SIGGRAPH '99: ACM SIGGRAPH 99 Conference abstracts and applications*. ACM Press, 1999, p. 169.

[10] R. Burke, D. Isla, M. Downie, Y. Ivanov, and B. Blumberg, "CreatureSmarts: The art and architecture of a virtual brain," in *Proceedings of the Game Developers Conference*, 2001, pp. 147–166.

[11] M. Downie, "Choreographing the extended agent: Performance graphics for dance theater," Ph.D. dissertation, MIT Media Lab, Cambridge, MA, September 2005.

[12] S. Meisner and D. Longwell, *Sanford Meisner on Acting*, 1st ed. Vintage, August 1987.

[13] L. Wang and C. Chen, "A combined optimization method for solving the inverse kinematicsproblems of mechanical manipulators," *Robotics and Automation, IEEE Transactions on*, vol. 7, no. 4, pp. 489–499, 1991.

[14] S. Moore, *Training an Actor: The Stanislavski System in Class*. New York, NY: Viking Press, 1968.

[15] G. Hoffman, "Ensemble: Fluency and embodiment for robots acting with humans," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, September 2007.

[16] M. Hancher, "A motor control framework for many-axis interactive robots," Master's thesis, Massachusetts Institute of Technology, 2003.

[17] C. Breazeal, *Designing Sociable Robots*. MIT Press, 2002.

[18] C. Wren, B. Clarkson, and A. Pentland, "Understanding purposeful human motion," in *Proc. Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, 2000, pp. 378–383.