Elvis: Situated Speech and Gesture Understanding for a Robotic Chandelier

Joshua Juster
MIT Media Laboratory
20 Ames Street
Cambridge, MA, 02142
(617) 818-3476
jjust@mit.edu

Deb Roy
MIT Media Laboratory
20 Ames Street
Cambridge, MA, 02142
(617) 253-0596
dkroy@media.mit.edu

ABSTRACT

We describe a home lighting robot that uses directional spotlights to create complex lighting scenes. The robot senses its visual environment using a panoramic camera and attempts to maintain its target goal state by adjusting the positions and intensities of its lights. Users can communicate desired changes in the lighting environment through speech and gesture (e.g., "Make it brighter over there"). Information obtained from these two modalities are combined to form a goal, a desired change in the lighting of the scene. This goal is then incorporated into the system's target goal state. When the target goal state and the world are out of alignment, the system formulates a sensorimotor plan that acts on the world to return the system to homeostasis.

Categories and Subject Descriptors

H.1.2 [Information Systems]: User/Machine Systems – human information processing, human factors; H.5.2 [Information Interfaces and Presentation]: User Interfaces – input devices and strategies, interaction styles;

General Terms

Design, Human Factors

Keywords

Gesture, speech, situated, grounded, multimodal, input methods, natural interaction, lighting

1. INTRODUCTION

Elvis is a robotic chandelier capable of creating and maintaining complex lighting environments. The system has a target goal state which it tries to preserve by constantly monitoring its environment and adjusting its motors and lights when it detects changes that are beyond its tolerable limits. The control strategy underlying Elvis is closely related to classic cybernetic systems in which closed-loop feedback is used to maintain homeostasis. Changes in the lighting environment trigger action in the robot, which tries to compensate and thus maintain its target lighting conditions. Users can affect change in the robot's target goal state through speech and gesture, causing it to go into action to regain homeostasis.

The system makes a clear distinction between its goals and actions. Its goal state can only be changed by user interaction. Otherwise, the system responds in the same manner both when the environment changes and when the goal state is altered by the user. Elvis compensates for differences between desired and actual environmental state by choosing optimal light and motor actions based on an acquired sensorimotor contingency table. This allows the system to be flexible.

The separation of goals and actions also means that the user never has to be concerned with direct control of the hardware of the system. Rather than expressing which lighting element to turn control, the user expresses desired lighting conditions using speech and gesture, and leaves it to Elvis to map those desires into specific actions of redirecting lights and adjusting intensity levels of lighting elements.

The system learns the effects of its actions on the lighting environment through an automatic procedure. If either the hardware or the environment change (for example, if a lighting element burns out, or the chandelier is moved to a new location), all that is required is for the system to retrain its sensorimotor contingency table. As a result, it would be simple, for example, to add an external lamp or electric window blinds under the system's control.

In this paper, we will first describe Elvis' physical embodiment and the process of sensorimotor learning. Next, we will proceed to focus on the goal system, including goal maintenance, goal shifting through speech and gesture recognition, and the transformation of goals into light and motor actions.

2. EMBODIMENT

Elvis' hardware consists of a custom-built robotic lighting fixture consisting of four two degree-of-freedom (DOF) directed spotlights, all of which swivel around a central ambient light. A video camera with fish eye lens is mounted in the center of the domed central light. Figure 1 shows a design sketch of the lighting system with directed beams of light (above) and a photo of the actual device (bottom). Each of Elvis' spotlights is capable of reaching approximately 80% of a 25 ft. by 25 ft. room. They can tilt 90° and rotate 180° around the ambient light.

ICMI'04, October 13–15, 2004, State College, Pennsylvania, USA.
Copyright 2004 ACM 1-58113-890-3/04/0010...$5.00.

Each spotlight is independently controlled by two servomotors. The first DOF allows each light to move radially along the perimeter of the central dome. The second DOF provides a pivot motion along the vertical plane. The intensity of each lamp (four spots and one central ambient in total) is under computer control with a resolution of 256 intensity settings.

The spotlights are fairly precise, projecting an 8-degree beam that creates a spot with a radius of approximately 1 foot, with a rapid falloff of illumination outside the focal area. The physical design of the system creates complexities that the system handles without the knowledge of the user. Each spotlight cannot reach all parts of the room and the system must take this into consideration when choosing an appropriate configuration.

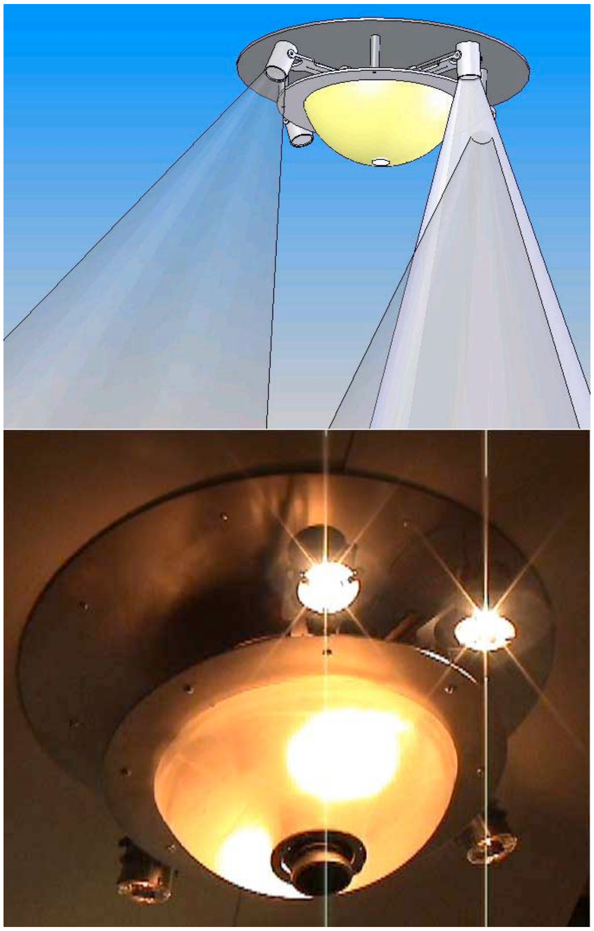


Figure 1. Elvis' hardware embodiment.

Additionally, there are multiple ways to shine light on most areas of the room. The optimal light choice depends on the availability of the other lights as well as the quality of coverage provided by each light. Finally, each spotlight can interfere with 90° of the span of its left neighbor and 90° of the span of its right neighbor. This must be taken into consideration in order to avoid collision.

The user wears a wireless microphone in order to interact with the system. Speech is converted to text using the Sphinx 4 speech recognizer [1]. The recognizer is trained on a trigram model of approximately sixty words.

3. SENSORIMOTOR LEARNING

Elvis utilizes direct inverse modeling [2] in order to learn how its spotlights affect its environment. A training phase involves two stages: motor babbling and scene analysis. First, the system captures an image of the environment as viewed through its camera with all of its lights turned off. Elvis then activates one of its lights and resamples the camera image, resulting in a difference map that is tied to the position and intensity setting of the light. This process is repeated for each of the eight motors (two per light) at ten-degree intervals, leading to a motor-lighting map (which may also be thought of as a sensorimotor contingency table). The procedure takes about twenty minutes and needs to be run once during initial setup. This feature allows the system to be easily moved to new locations. Once the training phrase is complete, Elvis is ready to operate in its new environment.

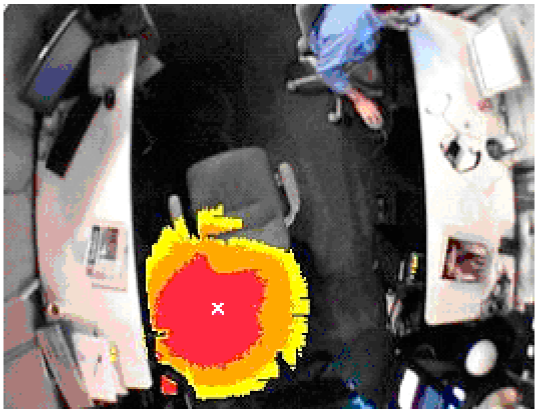


Figure 2. Elvis' spot detection. The white 'X' represents the weighted center of the spotlight.

For each motor position, Elvis creates and stores a lighting map representing the change in lighting location and brightness (Figure 2). A lighting map is represented by a 3-tiered system, where red represents 95% to 100% of the maximum detectable brightness level, orange represents 85% to 95%, and yellow represents 75% to 85%.

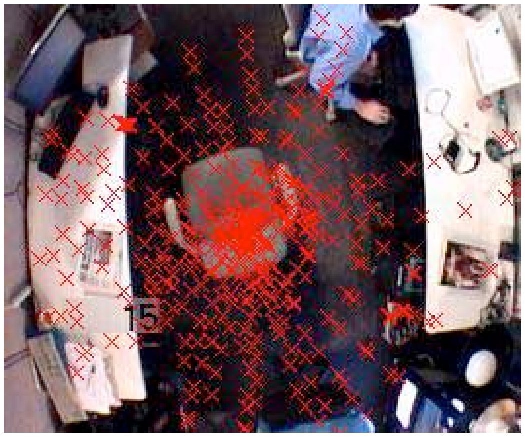


Figure 3. Sensorimotor Map

The center of the light's spot is determined by taking a weighted average of the points in each of the three lighting tiers. When

Elvis needs to find a way to shine light in a certain area, it first looks up lighting maps with centers close to the desired lighting and then compares lighting maps to determine the optimal choice.

Figure 3 shows a visualization of Elvis' motor-lighting map after training. Each 'X' marks the center of focus for a spotlight position that is stored it its learned sensorimotor map.

4. GOALS

Elvis views the world as a two dimensional map of lighting intensities. As a result, when Elvis intends on changing its environment, it views this change in terms of a lighting map, where each pixel represents the desired change in state at that point. A change in state can either be absolute or relative. For example, an absolute change might be to set a certain lighting area to half brightness. We refer to this map as a *goal*, since Elvis formulates this intermediate representation before utilizing its sensorimotor memory to devise a motor plan.

Additionally, individual goals can be combined to form a *composite goal*. Figure 4 shows a conceptual drawing of a composite goal. The figure might represent the command "move that light to the left", which involves a composite of two goals. The first removes the initial light and the second puts a spot with the same luminosity in the new location.

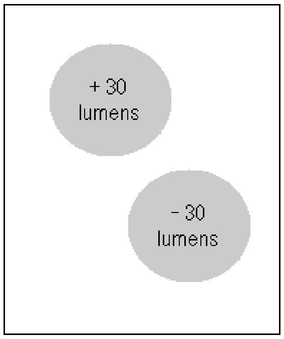


Figure 4. Composite goal corresponding to the command "Move that light to the left".

5. GOAL MAINTENANCE

When Elvis is turned on for the first time, it creates a lighting map from what it sees with its camera. This initial map is referred to as the *target goal state*. Every second, Elvis takes a snapshot of its environment and compares it to its target goal state. This snapshot will be referred to as the *world state*. When Elvis detects a significant difference between its two states, it uses its set of available action primitives to manipulate the world state until it looks as similar to its target goal state as possible. Figure 5 illustrates this process of maintaining homeostasis.

In order to determine how to make the appropriate adjustments, Elvis creates an intermediate *difference state*. This is attained by first finding the differences between the target goal state and the world state and then removing the impact that all of Elvis' lights have on the scene. This difference map represents all of the changes that Elvis hopes to make to his world.

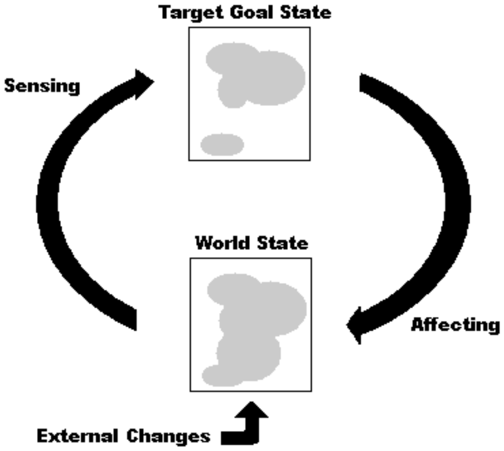


Figure 5. The closed-loop homeostasis process attempts to compensate for external environmental changes.

Once calculated, Elvis finds the lighting maps and corresponding lighting element positions in his sensorimotor memory which minimize the amount of fluctuation in his difference map. This process makes Elvis very robust in that he will always find his optimal configuration, regardless of the complexity of a scene. For example, if the user wants to light up five regions but Elvis only has four spotlights, Elvis will either cover only the four regions which most strongly affect his difference map or come up with some intermediate configuration. Additionally, Elvis has a built-in tolerance for minor changes in his world and will only volunteer an additional spotlight when this tolerance is exceeded.

6. GOAL SHIFTING

Elvis is designed to maintain homeostasis between its target goal state and the world state at all times. If a user wants to change the lighting scheme in a room, s/he indicates her/his desired change using speech and gesturing. The system then interprets these modalities in order to generate a goal. This goal is added to the *goal stack*. The goal stack consists of a history of all changes to the initial target goal state. The system can start at the initial goal state and apply each goal in the goal stack in order to determine the current target goal state. The goal stack will be useful in the future to keep track of prior changes and make it simple for the system to return to a previous state. The target goal state is, in effect, a composite goal in which all lighting areas in the map represent absolute levels. We refer to the idea of a user's input affecting the target goal state as *goal shifting*. Figure 6 illustrates this process.

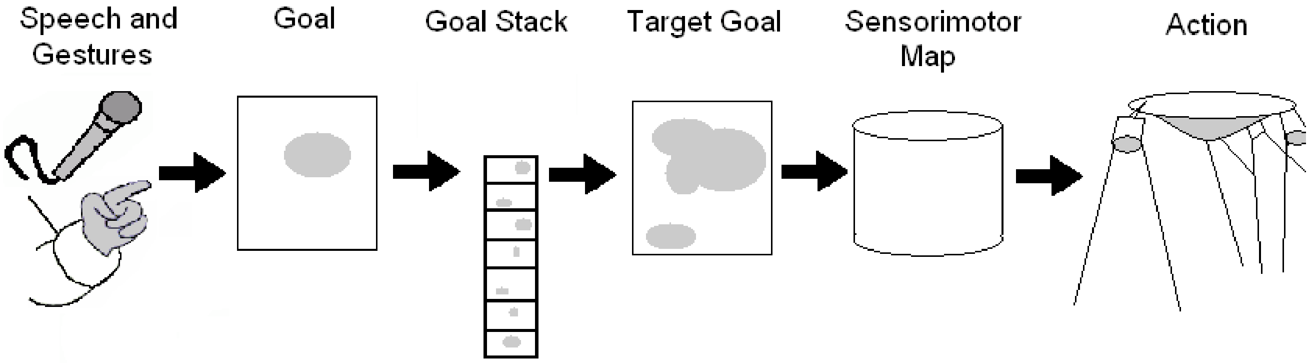


Figure 6. Goal Shifting

The following sections describe how the system analyzes speech and gestures in order to create a goal.

6.1 Speech Analysis

In this initial implementation, keyword spotting is used to analyze the semantic content of speech. Words are divided into four main categories: actions, areas, conjunctions, and intensifiers. Actions are structured in a hierarchy, as shown in Figure 7.

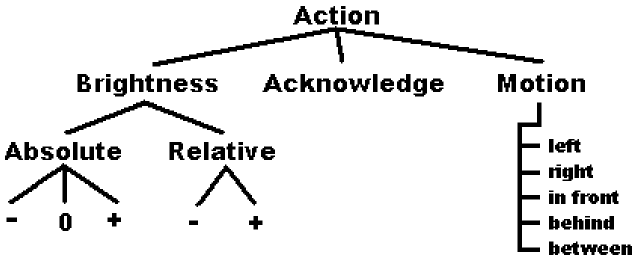


Figure 7. Action Hierarchy

Absolute actions are commands that request an absolute change in lighting, such as “make the room bright”. Relative actions are requests for relative changes in lighting such as “dim that area”. Acknowledge actions are requests for attention, such as simply calling the system’s name, “Elvis!”. In this case, Elvis will respond by *nodding* its lights (all four lighting elements “nod” by moving in and then out to acknowledge that the robot is ready for multimodal input). Motion actions can be more complex and usually supplement a gesture in describing a location (e.g., “shine light between here and there”). In this case, the system selects a region in the middle of the two areas indicated by the gestures.

Reference to areas is signaled by the detection of pronouns such as “it” and “there”. Conjunctions are used to split compound sentences into multiple commands or parse out multiple areas in a single command. Finally, intensifiers are used to modify the degree of effect of an action. For example, a request to make an area “very bright” is translated into a higher level of desired illumination compared to just “bright”. An example of a motion intensifier is the word “far”, such as in the phrase, “move the light far to the left”.

Words are used to fill command frames, which each consist of one action, one or more areas, and an optional intensifier. The structure of a frame, and the lexicon understood by Elvis is currently completely hand coded. As a result, the system in its current form is fairly brittle when used by unfamiliar users. We plan to soon collect “Wizard of Oz” training data to design more robust data-driven frame structures and lexical items.

If a frame consists of an acknowledge action, goal formulation is bypassed and Elvis immediately responds. Otherwise, the action, areas, and intensifier are analyzed in order to create a goal. In the simple case, when a user requests for a change in the global lighting, no additional information needs to be obtained. A goal can be formulated by setting the luminosity value based on the action word and intensifier, and the lighting map would simply consist of the entire room. However, in most cases, the speech only partially fills a frame since the actual location of areas is specified by gestures.

When the frame contains the word “it” in certain linguistic contexts, the system utilizes a simple heuristic to determine whether the area refers to a previous location or the global lighting. For example, if Elvis receives an utterance such as “make it brighter”, it will look in its action memory to determine the context of the word “it”. If a previous utterance was spoken within its attention span (10 seconds), Elvis will select the previously mentioned location. Otherwise, the system will assume that the user is referring to the entire room.

Often, disambiguating areas requires the analysis of the system’s gesture input. For example, if Elvis hears a phrase such as “Put some light over there”, the phrase “over there” must be disambiguated by sensing where the speaker is, and which area of the room she/he has in mind.

6.2 Gesture Analysis

The gesture analyzer utilizes the speech utterance to guide classification. The semantic class of co-occurring speech is used to bias the expected kind of gesture. For example, the phrase “over there” implies a distant place and the system therefore weights the gesture recognizer on trying to find a pointing gesture.

Elvis’ camera provides visual observations not only of the lighting conditions in the room, but also the position and gestures of the user (if the room is dark, the user must first use speech alone to turn on the lights so that Elvis can then see gestures for consequent commands). The user’s skin is detected using background subtraction and thresholded chrominance histograms [3]. For each frame, skin segments are discretized by performing a connected components analysis and recording the location of the center of each skin component into a gesture map. Elvis stores a cache of the last ten seconds of gesture maps. When the system receives speech input, it sends the *area* and *action* to the gesture system to determine if either term provides any useful information.

The system utilizes Kendon's notion of G-phrases that are made up of a *preparation-stroke-hold-retraction* sequence [4]. Our system primarily deals with transitive and intransitive deictic gestures [5], much like those used by weatherpersons [6]. As a result, our system uses the same primitives used by Kettebekov and Sharma's iMap system to recognize gesturing [7]. These primitives are *pointing, circle,* and *contour* gestures. They are also sufficient in capturing manipulative gestures [8]. Table 1 gives a list of areas, action types and their associated strokes. The more likely stroke is listed first. The first item in the table has an area of h*ere* or *this* and action of *Brightness*. Based on this combination, it is more likely that the user is referring to a local region and is circling it. However, it is also possible that the user is simply pointing at the region.

Table 1.Areas, action types, and probable strokes

Area(s)	Action Type	Probable Stroke(s)
Here, This	Brightness	Circle, Pointing
There, That	Brightness	Pointing
Here, This	Motion	Contour, Pointing
There, That	Motion	Contour, Pointing
(Here, This) + (There, That)	Between	Pointing, Contour

We are currently conducting a study to determine the likelihood that each stroke type occurs given an area and action type. Preliminary data has been collected for the *Brightness* action type. Five users were each given 10 photographs with marked regions indicating the area that they must attempt to illuminate. They were asked to speak naturally to Elvis, as if he were an intelligent being. The analysis of the resulting annotated video can be found in Table 2. Additional findings are highlighted in section 6.4.

Table 2. Preliminary Study Results for Brightness Action Type

		Area	
		There, That	Here, This
Stroke	Pointing	88%	36%
	Circle/Wave	22%	42%
	Touching	0%	11%
	No Gesture	0%	11%

After a list of gesture strokes and corresponding probabilities is obtained, Elvis reverts to its gesture memory and begins analyzing gesture segments starting one second prior to the speech segment and up to the end of the segment. We have found this gives enough time to allow for the pre-stroke hold, the preparation portion of the gesture prior to the start of speech [9]. This speech-based bracketing of gesture data successfully captures the gesture, since gestures typically begin slightly before or are temporally aligned with speech [10].

The system searches the gesture for evidence of any of the strokes in the aforementioned list and weights the likelihood of each of these strokes by their probabilities.

A rule-based approach is used to characterize gestures. A circle consists of any series of points that form a closed shape with a minimum area. A contour is a series of points in which the start point is distant from the end point. A point is a region in which points remain fairly constant in space over many frames of gesture maps. Since gestures don't necessarily last throughout the entire window of time, multiple gestures of the same type may be detected. The system chooses the gesture which best meets the criteria mentioned above.

Currently, pointing and contour gestures are fairly ambiguous since three-dimensional gestures are interpreted in a two-dimensional space. In the near future, we will attempt to incorporate image segmentation data when choosing regions. For example, if the user points to the edge of a table, the region will be limited to the table and not the surrounding floor.

Once the gesture is recognized, the system checks for motion words. If one exists, the system applies a hard-coded transform to the region to obtain a desired lighting map. For example, the motion word, *between,* would choose a region between the two detected points or between the start and end of a contour. This lighting region is added to the frame.

6.3 Goal Creation

Once a frame is filled, it is able to be transformed into a new goal, consisting of a lighting map with either absolute or relative lighting values.

6.4 Discussion

The initial study has surfaced a few noteworthy statistics. It should be noted that because the study was extremely limited in size, these preliminary results are not conclusive. Firstly, although participants were given several opportune scenarios, special relations such as "next to" and "between" were almost never used. We speculate that this is due to the fact that the room is relatively small and any area can be sufficiently described by simply pointing or motioning over a region.

It was also noted that 33% of all utterances contained a generic name of an object such as "chair" or "table" (this does not include more ambiguous nouns such as "thing", "part", or "end") despite the fact that the users were explicitly told to try to avoid using such terms.

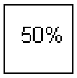
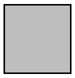
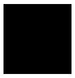
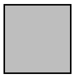
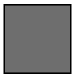
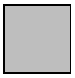
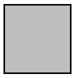
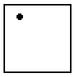
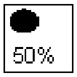
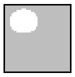
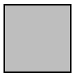
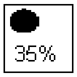
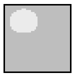
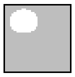
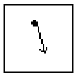
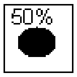
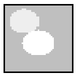
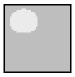
Finally, 27% of all pointing gestures involved two hands. Future versions of the gesture analyzer will attempt to account for this fact.

7. SAMPLE INTERACTION

This section illustrates a typical interaction with the system. Table 3 shows the state changes and corresponding output of the system for several different inputs. In the maps, white represents bright and black is dark.

(1) To begin, the user enters a dark room and says "Elvis, make it brighter". The robot first "nods"

Table 3.Working example. The parentheses in the Action column represent brightness levels.

	Speech / Change	Gesture	Goal	Target Goal State	World State	Action
1	Make it brighter	<none>				Ambient light brightens (50%)
2	*Blinds are closed*	<none>	<none>			Ambient light brightens (70%)
3	Elvis	<none>	<none>			All spotlights *nod*
4	Make it lighter over there					Spotlight-1 moves and turns on (50%)
5	Dim it a little	<none>				Spotlight-1 dims (35%)
6	Shine light behind there					Spotlight-2 moves Spotlight-3 moves and turns on (50%)

its actuated lights to acknowledge that it heard its name and is processing the command. Based on the anaphoric resolution heuristic described earlier, the system will decide that *it* refers to the room. By default, Elvis will brighten the room to 50% of full illumination.

(2) The user walks over to the window and closes the blinds. As a result, the room darkens and the target goal state and the world state no longer match. Elvis brightens the ambient light to compensate.

(3) The user simply says "Elvis!" and waits for a response. All of Elvis' motors nod in reply.

(4) The user decides to read at a chair in the corner of a room. Before the user gets to the location, she says "Make it lighter over there" and points to the chair. Elvis recognizes the gesture as a pointing gesture (the words "over there" guide the gesture recognition) and determines the location of the pointing. A spot is indicated on the goal lighting map, which is then integrated into the target goal state. Detecting a difference between the target state and world state, Elvis shines Spotlight-1 onto the chair and the user sits down and begins to read.

(5) After a few minutes, our user decides the light on her book is a little too bright. As a result, she requests "Dim it a little". After checking its action memory, Elvis determines that its attention is still on the chair and therefore recreates the previous goal map with a relative decrease in brightness. This changes the target goal state and Elvis responds by dimming Spotlight-1.

(6) Awhile later, the user decides to look for the newspaper but can't remember where she put it. She thinks it might be behind the coffee table. To verify, she commands "shine light behind there!" and motions to a spot behind the table. The system recognizes the contour, which is weighted by the word "behind", and locates the spot referred to by "there". It then applies a transform to change the spot to a location behind the original spot. A goal map is created containing this spot and added to the target goal map. Once again, the target goal state and the world state are no longer in equilibrium. The system chooses to move Spotlight-3 but Spotlight-2 is in the way. Spotlight-2 moves and Spotlight-3 reaches its destination and shines light behind the coffee table.

As currently implemented, Elvis is able to carry out multimodal interactions exemplified in this section in real-time with relatively high accuracy when interacting with familiar users.

8. CONCLUSIONS

We have presented a working robotic lighting system that translates speech and gesture commands into lighting changes. The system uses a set of four actuated lighting elements and a fifth fixed ambient light to set lighting scenes according to the user's requests. The basic control architecture of the system is based on the cybernetic notion of homeostasis. Speech and gestures perturb the robot's desired goal state, whereas environmental changes perturb the robot's perceived world state. Either type of perturbation causes the robot to take appropriate actions to regain homeostasis.

We acknowledge that one weakness of the current implementation is that it relies on hand-coded structures for interpreting speech and gesture. In future work, we plan to collect training data from unfamiliar users to train more robust classifiers to address this limitation.

We believe that a homeostasis control framework is a promising approach for the design of a variety of situated, interactive systems in which a layered approach to interface design may be used to create natural multimodal interfaces.

9. REFERENCES

[1] P. Lamere, P. Kwok, et al. *Design of the CMU Sphinx-4 Decoder.* Eurospeech, September 2003.

[2] M. Kuperstein. *Neural model of adaptive hand-eye coordination for single postures*. Science, 239. 1308-1311, 1988.

[3] M. Petrou, L. Shafarenko, and J. Kittler. *Histogram-based segmentation in a perceptually uniform color space.* IEEE Transactions on Image Processing, vol. 7, pp. 1354-1358, September 1998.

[4] A. Kendon. *Conducting Interaction.* Cambridge: Cambridge University Press 1990.

[5] D. McNeill. *Hand and Mind*. The University of Chicago Press, Chicago, 1992.

[6] R. Sharma, J. Cai, S. Chakravarthy, I. Poddar and Y. Sethi. *Exploiting Speech/Gesture Cooccurrence for Improving Continuous Gesture Recognition in Weather Narration.* In Proc. International Conference on Face and Gesture Recognition, Grenoble, France, 2000.

[7] S. Kettebekov and R. Sharma. *Understanding Gestures in Multimodal Human-Computer Interaction.* International Journal on Artificial Intelligence Tools, vol. 9, no. 2, pp. 205-224, June 2000.

[8] P. R. Cohen. *Synergic use of direct manipulation and natural language*. In Proc. Conference on Human Factors in Computing (CHI), (1989) 227.233. (1997) 415-422.

[9] S. Kita, I.V. Gijn, and H.V. Hulst. *Movement phases in signs and co-speech gestures, and their transcription by human coders*. In Proceedings of Intl. Gesture Workshop, (1997) 23-35.

[10] S. Oviatt, A. De Angeli, and K. Kuhn. *Integration and synchronization of input modes during multimodal human-computer interaction.* In Proceedings of the Conference on Human Factors in Computing Systems (CHI'97), 95.102, ACM Press, New York.