M.I.T. Media Laboratory Perceptual Computing Technical Report No. 261
Presented at IEEE Workshop for Visual Behaviors, CVPR-94

A Novel Environment for Situated Vision and Behavior

Trevor Darrell, Pattie Maes, Bruce Blumberg, Alex P. Pentland
MIT Media Laboratory

Abstract

We present a new environment for the development of situated vision and behavior algorithms. Our environment allows an unencumbered person to interact with autonomous agents in a simulated graphical world, though the use of situated vision techniques. An image of the participant is composited together with the graphical world and projected onto a large screen in front of the participant. No goggles, gloves, or wires are needed; agents and objects in the graphical world can be acted upon by the human participant through the use of domain-specific computer vision techniques that analyze the image of the person. The agents inhabiting the world are modeled as autonomous behaving entities which have their own sensors and goals and which can interpret the actions of the participant and react to them in real-time. We have demonstrated and tested our system with two prototypical worlds and describe the results obtained with over 500 people.

1 Introduction

In this paper, we explore the use of active vision in a new domain, where interaction with people is the primary focus. We present a world in which simulated agents interact with real people through a video screen and camera. In this environment the agents and a participant can "see" each other–the person can see the agents on the video screen, and the agents can see the person through a computer vision system. The participant's image appears on the video screen, effecting a type of "magic-mirror", in which people see themselves in a different world through the use of a simulated mirror (Figure 1). The relevance of this paradigm to vision is that it provides a domain which is both situated and suitably constrained to allow low-level visual routines to work, but is non-trivial in that people are dynamic and unpredictable. Vision routines are implemented on real camera output, but control the activity of agents which live in a simulated world.

2 The "Looking at People" domain

The "Looking at People" domain provides several challenges for computer vision relative to traditional application areas. Unlike static scenes or scenes with simple object motion, scenes with people are dynamic and have complicated articulated body kinematics, non-rigid motion, as well as gestures and other semantically-laden forms of communication. All of these properties prove difficult for traditional vision algorithms.

First, the shape and motion of human bodies are complicated and can be hard to characterize with precision. For example, Figure 2 shows a set of silhouttes of users of our system. These images are difficult to model precisely: for example, a complete motion model would require both a model of articulated limb dynamics, as well as a non-rigid model of skin and clothing dynamics. Since these are unlikely to be computable in real-time with conventional methods, the people-watching domain is a challenge in that it calls for finding methods which work robustly in the absence of a strict model.

Second, the fact that people are not simply objects, and have intentions and communicate via semantically-laden signs strongly argues for an "active" or "purposive" approach to vision [4, 2, 7]. The traditional stated goal of a computer vision system has been to assume the world is in a particular state and to attempt to recover as complete and accurate a description of that state as possible. But communication requires context, negating the utility of an "absolute state" of the world in this domain. A purely descriptive approach is thus inappropriate with regard to building interactive vision systems. The goal of vision routines for a people-watching system should not be to perfectly estimate and represent the three-dimensional shape of the body, face and hands, but rather to recover and provide some *meaningful* signal in the context of the current interaction. What exactly is meaningful will change over time, so no static description would suffice.

Finally, the presence of people in an interactive system provides a strong pressure to achieve real-time per-

Figure 1: The magic-mirror metaphor: a participant sees his/her mirror image surrounded by autonomous agents.

formance. Quite simply, if the system does not react in real-time, or perhaps what is more appropriately called "interactive-time", the user will get bored and leave. Unlike many static domains in which the agent can stop momentarily if necessary to make a decision, the visual routines and agent models used in an interactive man-machine system must be both robust and fast.

3 Attention and Intention

A key issue for practical real-time vision systems is the question of what to look for and where/when to look for it. Vision algorithms which are too general, e.g. look for everything everywhere, will usually suffer from an explosion in search complexity, and fail to offer adequate performance. As many authors have noted, the solution to this dilemma lies in the use of an attention mechanism [20].

Attention mechanisms require some state to exist in the perceiving agent. One can say "Attention requires Intention", in that without meaningful states and goals, a vision system has no principled way to prioritize what to look for next. The use of an action-selection system in the simulated agents fills this role; the same mechanisms that govern their locomotion, feeding, and following behaviors can provide perceptual goals as well. In this case, vision behaviors are "situated" in that they are based on these goals of the perceiving creature, and they are implemented on real video input of the user in the scene, not a simulated input scene.

To this end, an important aspect of our system is the use of a behavior-based agent model to drive the simulated creatures and agents in the computer graphics world. A behavior-based approach to modeling agents provides a simulated world which is populated with autonomous, unpredictable creatures, whose action selection model is ethologically plausible. Having agents whose action pattern mimics the behavior of animals in the real world allows for a "natural" and intuitive interface between people and those agents: a user can use his/her pre-existing knowledge of how to interact with a creature.

4 Action Selection with Time-varying Goals

To achieve a believable interaction, and in particular provide the context for interpreting the communications of the user and the attention mechanisms of the agent, a model of goals and intentional behavior is needed. Since multiple goals can exist and conflict with each other in an agent at any given time, the model should be able to mediate between heterogeneous goals in real time.

Recent results with reactive systems [12], routines [1] and subsumption architectures [8] have have demonstrated remarkably reliable and successful performance in performing real-time action selection in autonomous agents. The emphasis in these architectures is put on direct coupling of perception to action, distributedness and decentralization, dynamic interaction with the environment and intrinsic mechanisms to cope with resource limitations and incomplete knowledge.

Unfortunately, with the exception of some of these models, it is impossible for the agent to have time-varying goals that affect the behavior. As a result, agents built this way seem to only engage in very predictable, reflex-oriented behavior. In our system we employ a behavior model combining some of the best of both worlds: it produces fast and robust activity in a tight interaction loop with the environment, while at the same time allowing for some

Figure 2: Binary silhouettes of users after figure-ground processing. Task-dependent vision routines to find hands and pose information use this representation as input.

goal-dependent planning to take place. One of the distinguishing characteristics of the model used is that it is closely based on models of animal behavior. In particular, it borrows heavily from the work of classical Ethologists such as Lorenz, Tinbergen, Baerends, Ludlow and McFarland [14] [18] [3] [15] [17].

Specific ideas from Ethology that are incorporated in our model include:

- *structured behavior repertoire*: behaviors are organized as a loose hierarchy with the top of the hierarchy representing more general behaviors and the leaves representing more specific behaviors.
- *real-time dynamic planning*: all behaviors compete on every time step for control of the creature.
- *exclusivity*: A model of mutual inhibition among competing behaviors is used to insure that the agent engages in a single behavior at a time and does not dither between multiple behaviors.
- *hysteresis*: behavior-specific fatigue is modeled to insure that the temporal pattern of behaviors is believable and so that a creature doesn't mindlessly pursue an behavior indefinitely to the detriment of other needs.

The main difference between our approach and current situated behavior systems is that we neither hard-wire nor precompile the action selection. Arbitration among actions is a run-time process which differs according to the goals of the system and the situation it finds itself in.

Full details of the behavior model and a discussion of typical locomotion, foraging, and exploration behaviors are reported upon in [6]. In the next section we discuss the vision routines we have implemented for perceiving a person in an interactive setting.

5 Routines for Looking at People

We have developed a set of vision routines for perceiving body actions and gestures performed by a human participant in an interactive system with a simulated mirror paradigm. Our routines solve subsets of the perception task which are computationally tractable and still useful: tracking a user's location and posture, and performing simple, context-dependent gesture recognition.

The "magic-mirror" paradigm is attractive because it provides a set of domain constraints which are restrictive enough to allow simple vision routines to succeed, but is sufficiently unencumbered that is can be used be real people without training or a special apparatus. In this paradigm, a person faces a large screen, on which is both an image of the person and an image of the virtual world is presented. Vision routines analyze the image of the person and allow interaction with the virtual world.

5.1 Domain constraints

The "looking at people" domain provides constraints on the recognition and tracking problems that need to be tackled. We take advantage of these constraints in constructing our vision system. The most basic constraint is that we are looking at people, and can exploit specific domain knowledge about human anatomy. Hands are connected to a torso, which is usually below a head and above two feet, etc. A system can explicitly or implicitly use this knowledge to guide the recognition and/or tracking process.

Another important set of constraints derive from the fact that we can arrange the imaging geometry of the camera such that the user is almost always in a frontal pose. In our system, the same camera used for vision processing is also used for acquiring the image of the user which is composited into the graphics display. For the "magic mirror" effect to work, the image of the user used for the display must come from a camera position which is located approximately at the position of the screen. Since in this paradigm the user will be watching the screen almost continuously, we can assume with some degree of confidence that they will face the screen and thus their body will be oriented parallel to the screen most of the time. With this assumption, and the assumptions about the human figure, we can make relatively strong inferences about the position of the user's head and hands.

Below, we will describe the algorithms we use which

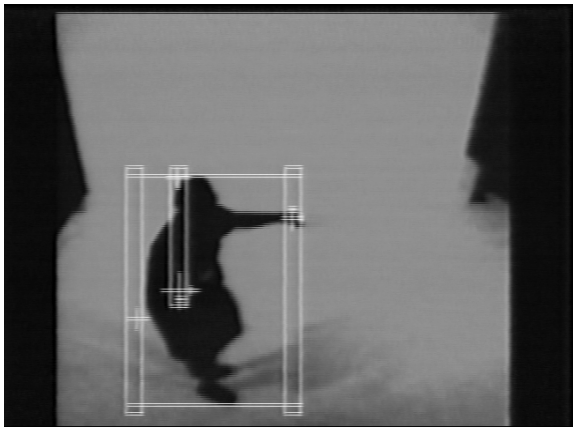

Figure 3: The vision system works off a silhouette of the user. It computes a range of features including the bounding box, which is used to project the user's location in 3D.

exploit these domain constraints and recover information about the user's position and pose. While we believe the constraints will be valid most of the time, we cannot always guarantee their validity. People may attempt to use the system in unforeseen ways, their dimensions may be far from the norm, and/or they may have therapeutic or prosthetic devices such as crutches or wheelchairs. When someone who does not fit the "prior model" implicit in the vision system attempts to use the system, it is important that the system not fail catastrophically. (In the cases described above, it is typically possible to continue to estimate basic position information about the user, but limb position information becomes unreliable.)

5.2 Figure-ground processing

Certain basic tasks and 3D information required to place the user in the graphical world *are* computed continuously (descriptively) and without regard to the state of agents in the world. For example, simple figure-ground segmentation is continuously performed by our system, as is the computation of bounding box information and its 3D position in the world. Other tasks, such as localizing the head or hands of the user, or interpreting body state (such as whether the user is bending over or pointing), are performed conditional on agent state.

Before other processing can occur, the vision system must isolate the figure of the user from the background (and from other users, if present). Our approach is to use low-level image processing techniques to detect differences in the scene, and use connected-components analysis routines to extract objects.

The simplest method of detecting differences is to constrain the background to be a known color, and detect instances of that hue in the input images. Several commercially available video image processing boards are available which can automatically isolate figure/ground and perform video compositing based on this approach [21]. This approach essentially performs clustering in color space to determine pixel membership in figure/ground classes. The advantages of this approach are ease of implementation, and the availability of real-time hardware solutions to the problem. Our first implementation used a single color background subtraction method to remove the background of the scene.

Constructing a uniform color stage for the user to act on is possible in many laboratory and exhibition environments, but is cumbersome in more constrained user environments such as offices or homes. A slightly more sophisticated approach is to allow the background to be an arbitrary, but static, pattern. Mean and variance information about the background pattern are computed using samples collected over a specified time-window. Using these statistics to determine thresholds for pixel class membership, accurate figure-ground membership is possible [5]. Typically the background statistics are recomputed continuously over a sliding time window, so that slow variation in the background pattern (say from changing illumination due to time of day) has no effect, and even large scale changes are adapted to on the order of a minute. [1]

Once a difference signal has been computed we apply connected components analysis to find a foreground region. We binarize the difference image, find connected regions, and compute the image bounding box and first-order moments of the largest connected region. (For a review of connected components and binary image processing methods see [9] and [16].)

5.3 Scene projection and calibration

Once the figure of the user has been isolated from the background, we compute its rough location in the world. If we assume the user is indeed sitting or standing on the ground plane, and we know the calibration of the camera, then we can compute the location of the bounding box in 3D.

Establishing the calibration of a camera is a well-studied problem, and several classical techniques are available to solve it in certain broad cases [9, 16]. Typically these methods model the camera optics as a pinhole perspective optical system, and establish its parameters by matching known 3D points with their 2D projection.

Knowledge of the camera geometry allows us to project a ray from the camera through the 2D projection of the

[1] If the scene conditions are such that static background subtraction is inappropriate, for example if there are multiple moving objects which cover large portions of the scene, then more sophisticated clustering methods are needed. In these cases motion-based grouping methods could be applied to find regions which are moving with coherent motions.[10]

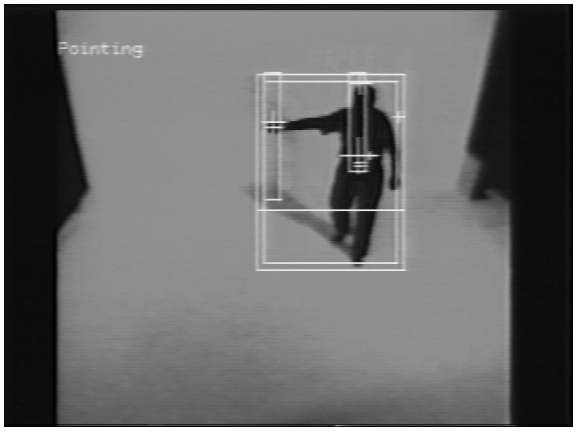

Figure 4: Gestures are modeled as spatio-temporal patterns. For the vision system to recognize a pointing gesture, the arm has to be stretched and the hand has to be held still.

Figure 5: Gestures are interpreted by the agents based on the context. Here, the Puppet walks away in the direction the user is pointing.

bottom of the bounding box of the user (figure 3 shows the bounding box as found in a real image, in this case the user is at the "front" of the virtual world, i.e. close to the mirror). Since the user is on the ground plane, the intersection of the projected ray and the ground plane will establish the 3D location of the user's base. The 2D dimensions of the user's bounding box and its base location in 3D constitute the low-level information about the user that is continuously computed and made available to all agents in the computer graphics world.

5.4 Hand tracking

One of the most salient cues used by the agents in our world is the location of the user's hands. We have implemented a hand search algorithm that uses spatial search patterns to localize hands in the input images. We make heavy use of the domain constraints outlined above, e.g. that people are (mostly) oriented in a fronto-parallel plane with respect to the camera. Figure 4 illustrates the output of the hand recognition system on a real image.

The hand-tracking algorithm we have developed is comprised of several different context-dependent search heuristics. In general, a normalized correlation search is done along the sides of the upper-torso bounding box to find a strong horizontal edge. The upper-torso bounding box is defined in such a way that it discounts to some degree the effect of shadows and feet in the horizontal dimension: it takes the vertical dimensions from the real bounding box and computes horizontal dimensions from the top 66% of the user's image.

Depending on the current context, different search windows and search patterns are used. The main contextual cue is the size of the bounding box, which provides a rough estimate of overall pose. If the box is narrow, we infer that the user's hands are at their side, and we do not attempt to find them in the silhouette. In this case we return the hand position to be located on the side of the bounding box, at the same relative height along the bounding box as it was last reliably seen.

5.5 Gesture interpretation

Hands are relevant to the agents in the world both for their absolute position, and also whether they are performing characteristic gesture patterns. We use simple low-level recognition strategies to detect these characteristic patterns.

Our model of gestures is highly reduced, and comprises two possible spatio-temporal hand patterns: pointing and waving. Each are defined in terms of the 2D motion of the location of the hand in the image plane. Pointing requires a particular relative hand location (an extended arm), and a steady position over time (Figure 4). Waving requires a predominantly side-to-side motion of the hand, while the user is otherwise stationary. These are special cases of our more general work on gesture recognition, which builds space and time separable template patterns for recognition[11]. However this work assumed a high resolution image of the object performing the gesture. As we have as yet no high-resolution camera to provide a foveated image of the hands (or face) of the user, we presently only model gesture as change in position (or lack thereof) over time.

Each gesture may be interpreted by different agents in the world depending on their context. For example, in the implementation described below, the waving gesture elicits a response from the Puppet agent, but not from the Hamsters. And the response given by the Puppet is state dependent; it will return when waved at only when it has been sent away or otherwise ignored.

6 An Example: ALIVE

We have combined these ideas in a system designed to allow a user to interact with an immersive visual environment without any physical apparatus. Our system, ALIVE, or "Artificial Life Interactive Video Environment" uses the agent and vision modeling techniques described above. With few exceptions (i.e. [13]), to experience these environments previously required the use of gloves, goggles, and/or a helmet, and most likely a wired tether to a computer graphics workstation [19].

We implemented the magic-mirror model in ALIVE using a single CCD camera to obtain a color image of the scene. The image of the user was separated from the background using color differencing with a known background, and then composited into the 3D graphical world. The composite world was projected onto a large screen, which faced the user. The polarity of projection was reversed so that the image indeed acted as a mirror.

ALIVE consisted of two worlds inhabited by different creatures; the user could switch between these worlds by pressing a virtual button. (To activate the button, the users hand had to be in the correct 3D location, not just the same image position.) One world was inhabited by a Puppet and the other by a Hamster and a Predator. The Puppet had behaviors to follow the user around, try to hold the user's hand, and imitate some of the actions of the user (sitting down, jumping, etc). It would be sent away when the user pointed away and come back when the user waved. The puppet employed facial expressions to convey some of its internal state. For example, it would pout when the user sent it away and smile when the user motioned it to come back. It giggled when the user would touch its belly.

Similarly the Hamster had behaviors to avoid objects, follow the user, and to beg for food. The user was able to feed the Hamster by picking up food from a virtual table and putting it on the floor. The user could open an adjoining cage and release a Predator, which would then chase the Hamster (but avoid the user).

The user could interact with the agent using certain hand gestures, which were interpreted in the context of the particular situation. For example, when the user points away (Figure 5) and thereby sends the puppet away, the puppet will go to a different place depending on where the user is standing. If the user waves or comes towards the puppet after it has been sent away, this gesture is interpreted to mean that the user no longer wants the puppet to go away, and so the puppet will smile and return to the user. In this manner, the gestures employed by the user can have rich meaning which varies on the previous history, the agents internal needs and the current situation.

6.1 Implementation

ALIVE was demonstrated for 5 days at SIGGRAPH-93 as part of the Tomorrow's Realities show. The user moved around in a real-world space of approximately 16 by 16 feet. A video camera at the front of the space captured the user's image. This version of the ALIVE system employed Chroma-keying to compose the user's image with the computer animated virtual world, which was performed by an Ultimatte system [21]. The composited image was displayed on a large screen (10' by 10') which faced the user. The machine used for the graphics and behavior modeling was an SGI Indigo Elan. We implemented the visual search routines on a dedicated image processor built by Cognex, Inc. for the vision algorithms, which was connected to the Indigo via serial line and ethernet.

In terms of the performance, as might be expected, rendering and sensing were the two bottlenecks. The agents update their state 6 times per animation frame. An individual update (including numerical integration) takes approximately 5 milliseconds per agent of which 60% is taken up by the simulated sensing. Rendering of the agents and the world takes another 75 milliseconds. The combined frame rate was roughly 10 frames per second. The vision sensing had a minimum update rate of 6 hertz, but can be quite a bit faster depending upon the complexity of interpretation required.

The ALIVE system was demonstrated and used by more than 500 users at the Tomorrow's Realities track of SIGGRAPH-93. Users generally enjoyed interacting with the system and considered the actions and reactions of objects and agents believable.

7 Conclusion

The ALIVE environment provides an interesting new domain to test behavior-based vision routines. The domain is dynamic, situated, and difficult to predict using conventional models. The "magic-mirror" metaphor allows a user to interact and navigate in a virtual world, using familiar means and without being disoriented. Unlike in viewer-centered virtual reality systems, the user did not get disoriented. They knew at all times where they were in the artificial world and could observe the actions of the other agents as well as their own.

Simple real-time vision routines can be successfully used in this context to provide an interaction between people and virtual agents or creatures. The gross 3D position of the user, the location of hands and head, and coarse information about overall pose (bending over, arms outstretched, etc) can be recovered using classical image processing techniques: figure-ground extraction, connected components, and context-based correlation search.

With these routines, users can directly manipulate objects and agents in the world. The autonomous agents populating the world have time-varying internal needs and motivations which determine what aspects of the user's state they are interested in, what visual search processing should occur, and how the results will be interpreted. The first version of ALIVE was demonstrated in a installation where over 500 people successfully used the system.

References

[1] Agre P. and Chapman D., Pengi: An Implementation of a Theory of Activity, Proc. AAAI-87, Morgan Kaufmann, Los Altos, California, 1987.

[2] Aloimonos Y., Active Perception, Lawrence Erlbaum Associates, Inc., Hillsdale, 1993.

[3] Baerends, G., On drive, conflict and instinct, and the functional organization of behavior, in: Perspectives in Brain Research 45, Corner M. and Swaab, D. (eds), 1976.

[4] Bajcsy, R., Active Perception, Proc. IEEE, Vol. 76, No. 8, pp. 996-1005, 1988

[5] Bichel, M. and Pentland, A., Topological Matching for Human Face Recognition, in: Looking at People Workshop, IJCAI '93, Chamberry, France, August 1993.

[6] Blumberg, B., et. al, in Proc. Conference on Simulation and Adaptive Behavior (SAB-94), Brighton, U.K., 1994

[7] Ballard, D., Animate Vision, Artificial Intelligence, Vol 48, pp. 57-86, 1991

[8] Brooks R.A., A robust layered control system for a mobile robot, IEEE Journal of Robotics and Automation, Volume RA-2, Number 1, 1986

[9] Ballard, D., and Brown, C., Computer Vision, Prentice-Hall, Englewood, 1982.

[10] Darrell T., and Pentland A., Robust Estimation of a Multi-Layer Motion Representation, in IEEE Motion Workshop, Princeton, 1991.

[11] Darrell, T. and Pentland, A., "Space-Time Gestures" Proceedings IEEE CVPR-93, New York, 1993.

[12] Kaelbling L.P., An architecture for intelligent reactive systems, in: Reasoning about Actions and Plans: Proceedings of the 1986 Workshop, Morgan Kaufmann, Los Altos, California, 1987.

[13] Krueger M.W., Artificial Reality II, Addison Wesley, 1990.

[14] Lorenz, K., Foundations of Ethology, Springer-Verlag, New York, 1973.

[15] Ludlow, A., The Evolution and Simulation of a Decision Maker, in: Analysis of Motivational Processes, ed. Toates, F. and Halliday, T., Academic Press, London, 1980.

[16] Horn, B.K.P.S., Robot Vision, M.I.T. Press, Cambridge, MA, 1991.

[17] McFarland, D. and Sibley, R., The behavioral final common path, Philosophical Transactions of the Royal Society, B. 270, 1975.

[18] Tinbergen, N., The Study of Instinct. Clarendon Press, Oxford, 1950.

[19] Rheingold H., Virtual Reality, Simon and Schuster, 1991.

[20] Tsotsos, J., A 'Complexity Level' Analysis of Immediate Vision, Int'l J. Computer Vision, pp. 303-320, Vol. 1, No. 3, 1988

[21] Ultimatte-300, Ultimatte Corp, Chatsworth, CA.