Virtual Autonomous Agents with Vision

Carlos H. Morimoto*
Departamento de Ciência da Computação do IME-USP
Rua do Matão 1010, São Paulo, SP 05508, Brazil
hitoshi@ime.usp.br

Abstract

This paper presents our ongoing work in developing virtual autonomous agents to facilitate human computer interaction, and in particular, perceptual user interfaces (PUIs) for desktop applications based on computer vision techniques. The desktop environment sufficiently constrains the computer vision problem in order to allow robust real-time performance for our agents. Currently the computer vision module is able to detect and track faces in real-time, and broadcast their positions to autonomous software agents. We are using a synthetic face to visualize the behaviors of the autonomous agent, which are very simple so far, and to give the user graphic feedback of the agent's status. Basically the agent detects the presence of the user by showing a happy face expression, tracks the user's face, becomes sad once the user is gone, and angry if s/he is gone for a long time. Extensions to the vision module to track facial features and recognize facial expressions, as well as extensions to the agent's behaviors, are being implemented.

Keywords: Human Computer Interaction, Computer Vision.

1 Introduction

Since computational power keeps becoming more affordable every day, we expect to start enjoying new and more powerful computer interfaces in the near future, specially within future pervasive devices. To create such new interfaces it will be necessary to move from the currently dominant WIMP (Windows, Icons, Menus, and Pointing) interface paradigm to a somewhat non-standard one.

In order to facilitate the user's activities, it is natural to automate as many tasks as possible using, for example, autonomous agents [11]. Such virtual agents require some kind of sensor capabilities to be able to interact with the user, and this new paradigm is sometimes called perceptual user interfaces (PUIs) [16].

This paper presents our ongoing work in developing virtual autonomous agents using computer vision techniques to facilitate human computer interaction (HCI), and in particular, PUIs for desktop applications. Desktop applications basically requires face and gesture recognition from the vision module, which greatly restricts the problems of general computer vision systems. Although one of the major problems with PUIs is how to integrate the information coming from different sensory channels, such as speech and touch, this topic is out of the scope of this paper.

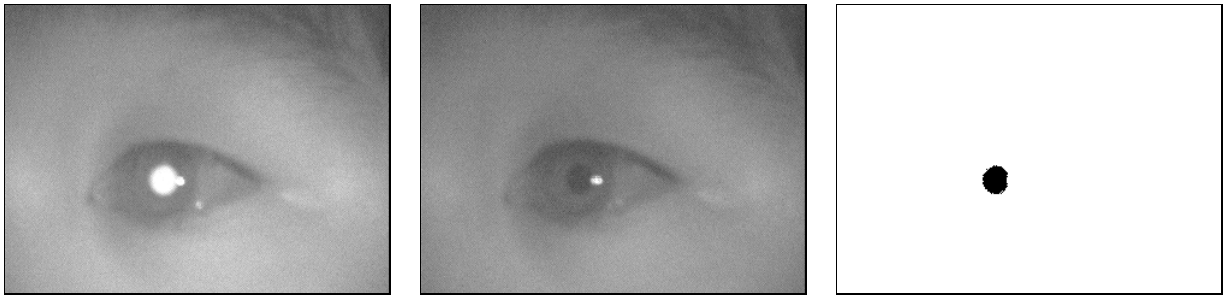
The vision system we are using is able to detect and track faces in real time using active illumination, but faces could also be detected using skin color [5, 15], motion subtraction [4], geometric models and templates [1, 14], artificial neural networks [3, 13], etc.

With small modifications of the camera's optical system, the vision module can also be used for eye-gaze tracking, i.e., it is possible to determine the screen coordinates to where the user is looking at, and send this information to gaze aware applications. Jacob [6] discusses several ways of using eye-tracking information as input for HCI, and it has been used, for example, as pointing devices for handicap people, to reduce the computational burden of large high resolution displays, and to increase the security in public data entry devices.

Extensions to the vision system are under development to allow for other interface modalities with the virtual autonomous agents, such as head gestures and facial expressions. Once these new modalities are implemented and included as part of the agent's capabilities as described in Section 3, usability studies will be conducted to verify the effectiveness of these interaction modes.

The next section describes some related work and the vision system in more details. Section 3 introduces some of the agent applications under development with preliminary experimental results. Section 4 concludes the paper.

*This research was supported by FAPESP - Fundação de Amparo à Pesquisa do Estado de São Paulo - under the contract 99/12176-7



(a) (b) (c)

Figura 1: (a) Bright and (b) dark pupil images. Notice the glint near the edge of the pupil. (c) Difference of the dark from the bright pupil after thresholding.

2 Vision for autonomous agents

One nice example of a computer vision system used for human interaction with virtual autonomous agents is the ALIVE system presented in [7], which demonstrates the effective use of virtual agents for entertainment applications. The system uses a vision-based interface that allows humans to interact with pet like virtual agents, such as parrots and dogs, in a non-intrusive way, i.e., without the use of goggles-and-gloves interfaces. ALIVE's vision module uses a single wide field-of-view camera to determine the 3-D position of the head, hands, and other salient body features from figure/ground segmentation of a known fixed background.

ALIVE is very robust and reactive but it is not quite suited for desktop applications, since its vision system processes mostly body gesture recognition. One lesson to be remembered from ALIVE is that the emotional display of the agents can help interaction. This is clear in the case of ALIVE since the user is interacting within the agent's environment, which is realistic and familiar to the user. For PUIs, the agent is the interface and its form becomes application dependent and many times unfamiliar, and although an interface is allowed to have several associated graphical displays, we believe that an explicit agent graphical output can be an important means for user feedback about the status of the task, as described in the next section.

The computer vision system we are using to detect and track faces is based on the system presented in [9]. It uses two infrared (IR) light sources to create dark and bright pupil images, as shown in Figure 1. One IR light is placed very near the camera's optical axis to create the bright pupil image as seen in Figure 1a (similar to the red eye effect seen on flash photography), and a second IR light source is placed distant from the camera's optical axis, in order to keep about the same illumination but with a dark pupil, as seen in Figure 1b. Simple subtraction with thresholding segments the pupil as seen in Figure 1c. Once the pupils are detected, they can be grouped into faces using heuristic and geometric rules. Each face could be independently tracked, but for desktop applications, tracking the most salient face is in general enough.

Another important application of the vision system is for eye-gaze tracking. Knowing where the user is looking can help the agent to determine its course of action, or simply tell something about the user's behavior and regions of interest The gaze tracker [10] basically uses the same hardware but with a longer lens, so that the pupil can be seen at the largest possible magnification. Once the pupil is detected using the differential lighting scheme, its center of mass is computed and tracked. The IR light sources create glints on the cornea, seen as the very bright spot near the edge of the pupil in Figure 1a and b. Assuming small head motion the glint can be used as a reference point. The distance from the center of the glint to the center of the pupil define a vector which is used to estimate the coordinate on the screen where the user is looking at, after a brief calibration process to determine the mapping between the glint-pupil vector to screen coordinates. This method works well for small head motion, and we have been working to make the system more robust to free head motion.

Several applications of eye-tracking to help human computer interaction has been recently demonstrated by the IBM Blueeyes project. For example, Suitor [12] helps the user browse the Internet by automatically bringing more information related to the subject that the user is currently reading on a ticker. Suitor is very pro-active, i.e., it does not ask if you want more information or not, thus, as soon as the user finishes reading the headline on the ticker, it brings the corresponding page up on a browser. If the user quits reading before the end of the headline is reached, Suitor understands that the news is not interesting, and does nothing.

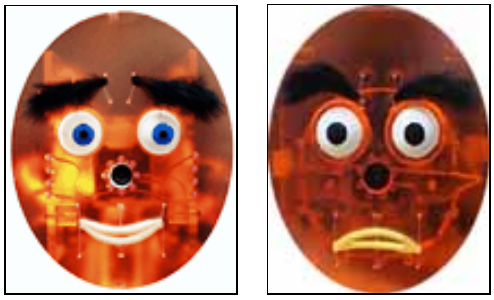


Figura 2: Some of Pong's facial expressions.

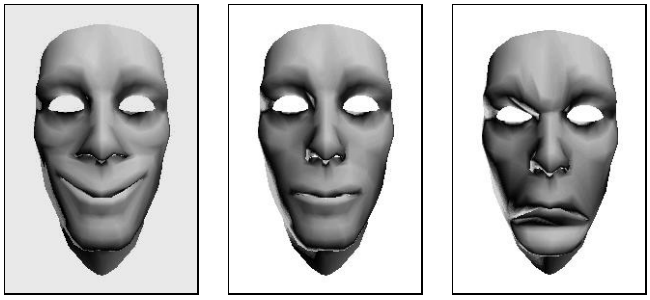


Figura 3: A happy, normal, and angry facial expression from the synthetic DECface.

Another application for eye-tracking is MAGIC pointing [18]. It is hard to use eye-gaze as a pointing device because it is hard to select objects with your eyes by dwell time or blinking [6], since some objects might be selected involuntarily. This is known as the Midas touch problem, with the extra inconvenience that the cursor is always where the user is looking at on the screen. Also, the precision of remote eye-gaze tracking systems is not very good when compared to other pointing devices, requiring large targets to be used, at least about 1 square inch for regular desktop applications. MAGIC solves these problems by combining the strengths of regular pointing devices, such as a mouse, with the eye tracker. When the user is just looking at the screen, the cursor behaves normally, i.e., it does not move. If the user wants to move the cursor, s/he would have to touch and move the mouse to accomplish this task. With MAGIC, the cursor is automatically warped to a position near the desired target as soon as the user touches the mouse, so s/he does not have to drag it, and uses the mouse just for fine adjustments, creating a much more natural and comfortable interface. Next we introduce some agents we are developing using the vision systems described in this section.

3 Interface agents

Besides the agents that populate ALIVE's environment and directly interact with the user, it uses another autonomous agent as an artificial guide, visualized as a parrot, which occasionally makes suggestions based on the types of interactions the user has had with the environment. To fully demonstrate the potential and functionality of our system, a more complex visualization will be required.

The first agent we describe is an eye-contact agent. It is very similar in behavior to the Pong robot developed by the Blueeyes team project. Pong is a robot head that can mimic facial expressions, as seen in Figure 2. Our agent is a virtual one, whose graphical output is based on DECface [17], which was used for speech and lip synchronization for synthetic faces. DECface models facial muscles and lacks other facial features such as eyes, eyelids, and eyebrows, but can be still very expressive as can be seen in Figure 3.

The behavior of our agent is very simple. If no face is being detected, it shows a sad or angry face, depending of how long it has seen a human face. As soon as the system detects a face, it shows surprise and then smiles, as long as the face stays in sight. The eyes are being implemented, and they will follow the user's face, maintaining eye-contact. As soon as our facial feature tracking algorithms become available, a mirror agent that mimics every user facial expression will be also implemented. Facial features and facial expressions can be computed using local parameterized models. Black and Yacoob [2] present a facial feature tracking system based on robust statistics, and Morimoto *et al.*[8] used this system to recognize head gestures using Hidden Markov Models. The facial feature tracking system assumes though that the initial position of the features are known.

The second agent we are working with is a guide to help the calibration process for the gaze tracker. The calibration process requires the user to fixate at 9 screen points and hit the space bar on the keyboard.for each of them. The current agent only tells the user if it was able to compute the calibration or not, but it is being extended with some heuristic rules in order to be able to suggest the user solutions to common calibration problems.

4 Conclusion

We have presented a few virtual autonomous agents that use computer vision for human-computer interaction. We are using a real-time face detection and tracking system that can also be used for eye-gaze tracking. Since our agents are in their early stages of development, their behaviors are still very simple. More complex behaviors are being implemented to experiment with perceptual user interfaces (PUIs).

Our PUIs will also use synthetic faces with expressions to give the user feedback about the result or the status of their tasks. For example, when asked to perform a task, the face can turn its back, and once its finished, it can face the user again with a "happy" or "sad" expression denoting the results of its processing. After the agent makes eye-contact with the agent, then it returns to a normal facial expression. The expression intensity can also be controlled by a measure of the success of the task. A facial expression could also show that a command (e.g. gesture or voice command) was not understood, warn about the status of its related task, "cry" for more user input/help, and so on.

Referências

[1] S. Birchfield. An elliptical head tracker. In *Proceeding of the 31st Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, November 1997.

[2] M.J. Black and Y. Yacoob. Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motions. technical report CAR-TR-3401, Center for Automation Research, Univ. of Maryland, College Park, MD 20742-3275, May 1995. (abstract only).

[3] A.J. Colmenarez and T.S. Huang. Face detection with information-based maximum discrimination. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 782–787, Puerto Rico,PR, June 1997.

[4] T. Darrell, B. Moghaddam, and A. Pentland. Active face tracking and pose estimation in an interactive room. Technical Report 356, M.I.T. Media Laboratory Perceptual Computing Section, Cambridge, MA, 1996.

[5] P. Fieguth and D. Terzopoulos. Color based tracking of heads and other mobile objects at video frame rates. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 21–27, Puerto Rico,PR, June 1997.

[6] R.J.K. Jacob. The use of eye movements in human-computer interaction techniques: What you look at is what you get. *ACM Transactions on Information Systems*, 9(3):152–169, April 1991.

[7] Pattie Maes. Artificial life meets entertainment: lifelike autonomous agents. *Communications of the ACM*, 38(11):108–114, November 1995.

[8] C.H. Morimoto and R. Chellappa. Fast electronic digital image stabilization. In *Proc. International Conference on Pattern Recognition*, Vienna, Austria, August 1996.

[9] C.H. Morimoto and M. Flickner. Real-time multiple face detection using active illumination. In *Proc. of the 3rd Int. Conf. on Automatic Face and Gesture Recognition*, Grenoble, France, March 2000.

[10] C.H. Morimoto, D. Koons, A. Amir, and M. Flickner. Pupil detection and tracking using multiple light sources. *Image and Vision Computing*, 18(4):331–336, March 2000.

[11] H.S. Nwana and D.T. Ndumu. An introduction to agent technology. In H.S. Nwana and N. Azarmi, editors, *Software agents and soft computing*, pages 3–26, 10662 Los Vaqueros Circle, P.O. Box 3014 Los Alamitos, CA 90720-1314, 1997. Springer Verlag.

[12] IBM Almaden Research Center: BlueEyes Project. URL: http:// www.almaden.ibm.com/ cs/ blueeyes.

[13] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, June 1996.

[14] L.C. De Silva, K. Aizawa, and M. Hatori. Detection and tracking of facial features. In *Proceedings of the SPIE Com. and Image Proc. 95*, volume 2501, pages 1161–1172, Taipei, Taiwan, May 1995.

[15] R. Stiefelhagen, J. Yang, and A. Waibel. A model-based gaze tracking system. In *Proceedings of the Joint Symposia on Intelligence and Systems*, Washington, DC, 1996.

[16] M. Turk. Moving from guis to puis. Technical Report MSR-TR-98-69, Microsoft Research, 1998.

[17] K. Waters and T. Levergood. Decface: an automatic lip-synchronization algorithm for synthetic faces. technical report 93/4, Cambridge Research Lab, Digital Equipment Corporation, Cambridge, Massachussets, August 1993.

[18] S. Zhai, C.H. Morimoto, and S. Ihde. Manual and gaze input cascaded (magic) pointing. In *Proc. ACM SIGCHI - Human Factors in Computing Systems Conference*, pages 246–253, Pittsburgh, PA, May 1999.