Position paper for the workshop "Gesture at the User Interface", CHI'95, Denver

Interpreting Human Gesture with Computer Vision

Joëlle Coutaz and James L. Crowley***

* Laboratoire de Génie Informatique (LGI-IMAG)
BP 53, 38 041 Grenoble Cedex 9, France
E-mail: Joelle.Coutaz@imag.fr, tel. +33 76 51 48 54, fax : +33 76 44 66 75

** Laboratoire d'Informatique Fondamentale et d'Intelligence Artificielle (LIFIA-IMAG)
46 Avenue Félix Viallet, 38031 Grenoble Cedex, France
E-mail: jlc@imag.fr, tel. +33 76 57 46 55, fax : +33 76 57 46 02

Abstract

This paper discusses the use of computer vision for interpreting human gesture. We first present the functional roles of human gesture and observe that in real life, gesture may be enhanced with physical instruments. In Human Computer Interaction, input devices such as the data-glove, are convenient tricks for computer scientists to sense human gesture but are not necessarily ecological instruments. We claim that computer vision can turn the human (the whole as well as the parts) into an effective input device without intrusive add-on's. In the last sections of the paper, we present FingerPaint, our demonstration of the use of cross-correlation as a means of tracking natural pointing devices (e.g., bare fingers, pens and erasers) in the context of the digital desk.

1. The Functional Roles of Human Gesture

Human gesture serves three functional roles [Cadoz 94]: semiotic, ergotic, and epistemic.

- The *semiotic* function of gesture is to communicate meaningful information. The structure of a semiotic gesture is conventional and commonly results from shared cultural experience. The good-bye gesture, the American sign language, the operational gestures used to guide airplanes on the ground, and even the vulgar "finger", each illustrates the semiotic function of gesture.

- The *ergotic* function of gesture is associated with the notion of work. It corresponds to the capacity of humans to manipulate the real world, to create artefacts, or to change the state of the environment by "direct manipulation". Shaping pottery from clay, wiping dust, etc. result from ergotic gestures.

- The *epistemic* function of gesture allows humans to learn from the environment through tactile experience. By moving your hand over an object, you appreciate its structure, you may discover the material it is made of, and so on.

All three functions may be augmented using an *instrument*: a handkerchief for the semiotic good-bye gesture, a turn-table for the ergotic shape-up gesture of pottery, or a dedicated artefact to explore the world (for example, a retro-active system such as the pantograph [Ramstein 94] to sense the invisible).

2. Gesture in HCI: Computer Input Devices are "tricky instruments"

In Human Computer Interaction, gesture has been primarily exploited for its ergotic function: typing on a keyboard, moving a mouse, and clicking buttons. The epistemic role of gesture has emerged effectively

from pen computing and virtual reality: ergotic gestures applied to an electronic pen, to a data-glove or to a body-suit are transformed into meaningful expressions for the computer system. Special purpose interaction languages have been defined, typically 2-D pen gestures as in the Apple Newton, or 3-D hand gestures to navigate in virtual spaces or to control objects remotely [Baudel 93].

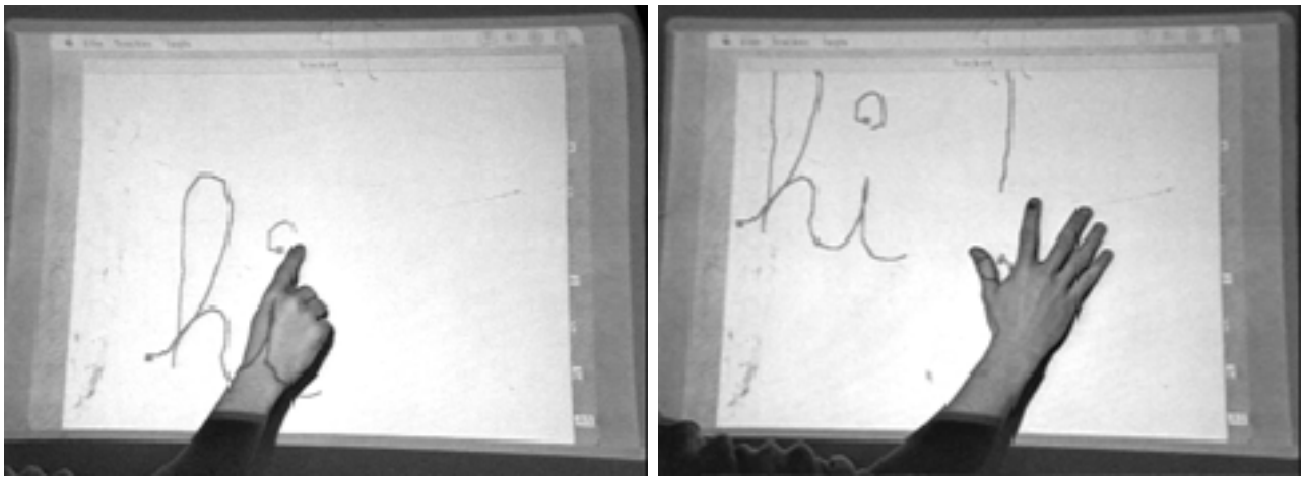
With the exception of the electronic pen and the keyboard which both have their non-computerized counterparts, mices, data-gloves, and body-suits are “artificial add-on’s” that wire the user down to the computer. They are not real end-user instruments (as a hammer would be), but convenient tricks for computer scientists to sense human gesture.

We claim that computer vision can transform ordinary artefacts and even body parts into effective input devices. Krueger’s seminal work on the videoplace [Krueger 91], followed recently by Wellner’s concept of digital desk [Wellner 93] show that the camera can be used as a non-intrusive sensor for human gesture. However, to be effective the processing behind the camera must be fast and robust. The techniques used by Krueger and Wellner are simple concept demonstrations. They are fast but fragile and work only within highly constrained environments.

We are exploring advanced computer vision techniques to non-intrusively observe human gesture in a fast and robust manner. In the next section, we present FingerPaint, an experiment in the use of cross-correlation as a means of tracking *natural pointing devices* for a digital desk. By “natural pointing device”, we mean a bare finger or any real world artefact such as a pen or an eraser.

3. FingerPaint

FingerPaint runs on an Apple Quadra AV/840. It uses a work-space projected with an overhead projector using a liquid-crystal display "data-show". A CCD camera with an 18mm lens observes this workspace and provides visual input. “Finger down” and “finger up” events are simulated using the space bar of the keyboard but they could be sensed using a microphone attached to the surface of the desk. As illustrated in Figure 1, any “natural pointing device” such as a finger can be used to draw pictures and letters, or to move a drawing.



a b **F i**

gure 1. Drawing and placing with "Fingerpaint".

The robustness of the tracking system is reasonable but, as discussed below, its performance is inadequate with respect to Fitt's law. Preliminary experiments with local users indicate however that the current performance is acceptable for investigation purposes. In addition, the widespread availability of image acquisition and processing hardware adequate for real time correlation should alleviate our current performance problem.

In the next section, we present the principles of tracking based on correlation. Details can be found in [Berard 94].

4. Correlation as a Tracking Technique for the digital desk

The visual processes required for a digital desk application like FingerPaint are relatively simple. The basic operation is tracking some pointing device, such as a finger, a pencil or an eraser.

4.1. Principles of cross-correlation

The tracking problem can be expressed as: "Given an observation of an object at time t, determine the most likely location of the same object at time t+ΔT". We have experimented with two different approaches to tracking pointing devices: correlation tracking and active contours (snakes)[Berard 94]. The active contour model [Kass 87] presented problems which we believe can be resolved, but which will require additional experiments. Our current demonstration uses cross-correlation.

The pointing device is modelled as a reference template. The reference template is a small neighbourhood, i.e., a window w(n, m) of a picture p(i, j) obtained at some prior time, t. The reference template is compared to the image by computing a sum of squared differences (SSD) at each pixel with a MxM search region :

$$SSD(i, j) = \sum_{m=0,N} \sum_{n=0,N} (p_k(i+m,j+n) - w(m, n))^2$$

A perfect match gives a value of 0. Implementing cross-correlation by SSD requires solving practical problems such as determining the sizes of the reference template and of the search region, triggering and breaking tracking, and updating the reference template.

4.2. The size of the reference template

The size of the reference template must be determined such that it includes a sufficiently large portion of the device to be tracked and a minimum of the background. If the template window is too large, correlation may be corrupted by the background. On the other extreme, if the template is composed only of the interior of the pointing device then the reference template will be relatively uniform, and a high correlation peak will be obtained with any uniform region of the image, including other parts of the pointing device. For a reasonable correlation peak, the reference template size should be just large enough to include the boundary of the pointing device, which contains the information used for detection and localisation.

Our workspace is of size 40 cm by 32 cm. This surface is mapped onto an image of 192 x 144 pixels, giving pixel sizes of 2 mm by 2.2 mm. At this resolution, a finger gives a correlation template of size 8 by 8 pixels or 16mm by 18mm, as shown in figure 2.

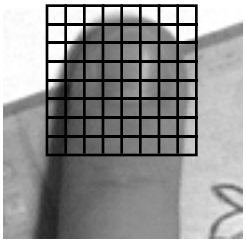


Figure 2 : Reference template for a finger.

4.3. The size of the search region

Clearly, the size M of the search region depends on the speed of the pointing device. Considerations based on Fitts law [Card 83] indicated expected tracking speeds of up to 180 cm/sec. To verify this, we performed an experiment in which a finger was filmed making typical pointing movements in our workspace. The maximum speed observed in this experiment was V_m = 139 cm/sec. Expressed in pixels this gives V_m = *695 pixels/sec*.

Given an image processing cycle time of ΔT seconds per cross-correlation, and a maximum pointer speed of V_m pixels/sec, it is possible to specify that the pointing device will be found within a radius of $M = \Delta T$ V_m pixels of its position in the previous frame. For images of 192 x 144 pixels, our built-in digitizer permits us to register images at a maximum frame rate of 24 frames per second, giving a cycle time of ΔT_{max} = 41.7 msec. This represents an upper limit on image acquisition speed which is attainable only if image tracking were to take no computation time.

The computational cost of cross-correlation is directly proportional to the number of pixels in the search region. Reducing the number of pixels will decrease the time needed for the inner loop of correlation by the same amount. This, in turn, increases the number of times that correlation can be operated within a unit time, further decreasing the region over which the search must be performed. Thus there is an inverse relation between the width of the search region, M, and the maximum tracking speed, V_m. The smaller the search region, the faster the finger movement that can be tracked, up to a limit set by the digitizing hardware.

The fastest tracking movement can be expected at a relatively small search region. This is confirmed by experiments. To verify the inverse relation between M and V_m, we systematically varied the size of the search region from M = 10 to 46 pixels and measured the cycle time that was obtained. The maximum speed of 126 pixels/sec is obtained with M=26. Although this is 5.5 times less than the maximum desirable speed (i.e., 695 pixels/sec), the system is quite usable to perform drawing and placements in a "natural" way.

4.4. Triggering and breaking tracking

When tracking is not active, the system monitors an 8 by 8 pixel "tracking trigger" located in the lower right corner of the workspace. As each image is acquired at time k, the contents of this tracking trigger are subtracted from the contents at the previous image k-1. This creates a difference image. The energy of the difference image is used to detect the trigger. When a pointing device enters the tracking trigger, the energy rises above a threshold. In order to assure that the tracking device is adequately positioned, the system waits until the difference energy drops back below the threshold before acquiring the reference template. At that point, the contents of the tracking trigger are saved as a reference image, and the tracking process is initiated.

Tracking continues as long as the correlation remains above a relatively low threshold. However, it can happen that the tracker locks on to a pattern on the digital desk (for example a photo of the pointing device!). To cover this eventually, if the tracked location of the pointer stops moving for more than a few seconds (say 10), the system begins again to observe the difference energy in the tracking trigger. If the trigger energy rises above threshold, the tracker will break the previous track and re-initialise the reference pattern with the new contents of the tracking trigger.

4.5. Updating the reference mask

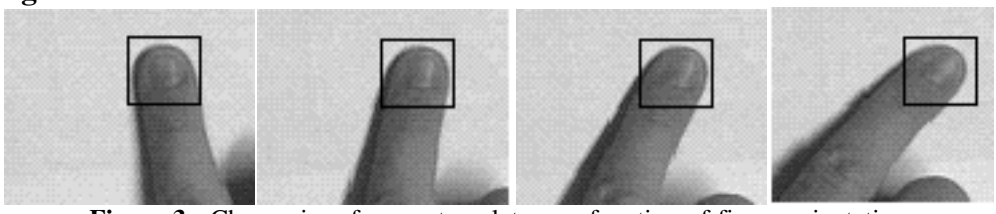


Figure 3 : Change in reference template as a function of finger orientation.

As the user moves the pointing device around the workspace, there is a natural tendency for the device to rotate, as shown in figure 3. This, in turn, may cause loss of tracking. In order to avoid loss of tracking, the smallest SSD value from each search is compared to a threshold. If the smallest SSD rises above this threshold, then the reference template is updated using the contents of the image at time k-1 at the detected position.

4. Conclusion

In this paper we have presented some preliminary results with tracking of fingers as a pointing device. We have also experimented with eyetracking using snakes and other techniques based on signal processing, as well as face interpretation using EigenFaces [Turk 91]. It is increasingly apparent that computer vision has the potential to provide a rich new source of interaction modes for computer human interaction. Vision can make the machine "aware" of the user, his movements and environment, in ways that are yet to be invented.

Bibliography

[Baudel 94]
T. Baudel, M. Beaudouin-Lafon, “Charade: Remote Control of Objects Using Free-Hand Gestures”, Communications of the ACM, Vol.36 No.7, July 1993, pp. 28-35.

[Berard 94]
F. Berard, "Vision par Ordinateur pour la Réalité Augmenté: Application au Bureau Numrérique", Mémoire du D.E.A. en Informatique, Univeristé Joseph Fourier, Juin 94.

[Cadoz 94]
C. Cadoz, “Les réalités virtuelles”, Dominos, Flammarion, 1994.

[Kass 87]
M. Kass, A. Witkin, D. Terzopoulos. “Snakes : Active Contour Models”. Proc. 1st International Conf. on Computer Vision, pp. 259-268, 1987.

[Krueger 93]
M. Krueger, “Artifical Reality II”, Addison Wesley, 191.

[Ramstein 94]
C. Ramstein, “The Pantograph: A Large Workspace Haptic Device for Multimodal Human Computer Interaction”, CHI’94 Interactive Experience, Adjunct Proceedings, pp. 57-58.

[Turk 91]
M. Turk and A. Pentland. “Eigenfaces for Recognition” Journal of Cognitive Neuroscience, 3(1):71-86, 1991.

[Wellner 93]
P. Wellner. “Interacting with paper on the DigitalDesk”. Communications of the ACM, Vol.36 No.7, July 1993, pp. 86-95.

Jöelle Coutaz

Research activities

Joëlle Coutaz is professor at the University Joseph Fourier of Grenoble where she has been teaching the foundations for operating systems since 1973 and Man-Machine Interaction since 1986. She is the founder and director of the User Interface group at LGI-IMAG (Laboratoire de Génie Informatique of IMAG). This group is primarily concerned with the software engineering aspects of Computer Human Interaction and aims at transfering HCI design principles and theories into software design modelling.

In particular, the group has developed PAC, a conceptual architecture model that captures software engineering issues such as modularity and modifiability, as well as user-centred concerns such as multithreading and multimodal interaction. The group has developed demonstrators and software tools based on the PAC model: a fusion engine has been implemented as a reusable mechanism for supporting multimodal input interaction; PAC-Expert, an expert system based on heuristic rules, demonstrates the automatic generation of a PAC global architecture from the description of the system behavior as perceived by the user; an environment based on PAC, has been developed for Renault, the French car company, to specify and generate embedded user interfaces; NEIMO, a usability lab based on PAC, is under development in collaboration with CNET, the French telephone company.

Joëlle Coutaz is the author of two books : “Developing Software for the User Interface (co-author Len Bass), Addison Wesley, 1991; and “Interfaces Homme-Ordinateur : Conception et Réalisation”, Dunod, 1990.

Joëlle Coutaz is member of the ESPRIT BRA AMODEUS, member of the Working Group 2.7 of IFIP, and expert or the European Commission DG III ESPRIT Programme (BR). In France, she is the founder of the research group “Multimodal man-machine communication” of the national programme PRC Man-Machine Communication. She also co-chairs the CSCW working group of PRC Man-Machine Communication and is member of the CNRS research group Cogniscience. Relationships with the industry include Renault-Recherche, EDF, CNET, Thomson.

Selected Publications

Coutaz, J. (1987) PAC: an Implementation Model for Dialog Design, *Proceedings of the Interact'87 conference*, Stuttgart, H-J. Bullinger, B. Shackel ed., North Holland pp. 431-436.

Coutaz, J. & Balbo, S. (1991) Applications: A Dimension Space for User Interface Management Systems. In *Proc. CHI'91*, ACM Publ., pp. 27-32.

Normand, V & Coutaz (1992) J. Unifying the Design and Implementation of User Interfaces through the Objectd Paradigm, ECOOP’92, 1992.

Nigay, L. & Coutaz, J. (1993) A design space for multimodal interfaces: concurrent processing and data fusion, INTERCHI’93 Proceedings, Amsterdam, May, 1993, pp. 172-178.

Nigay, L. & Coutaz, J. (1995) A Generic Platform for Addressing the Multimodal Challenge, Proceedings of CHI’95, ACM, Denver, May 1995, to appear.

Coutaz, J., Nigay, L. ,Salber, D. ,Blandford, A. , May, J. & Young, R. (1995) Four Easy Pieces for Assessing the Usability of Multimodal Interaction: The CARE Properties, Proceedings of INTERACT'95, Lillehammer, Norway, June 1995, to appear.

James L. Crowley

Research Activities

James L. Crowley holds the post of Professor at the Institut National Polytechnique de Grenoble (INPG). He teaches courses in Artificial Intelligence, Machine Vision, and Robotics at l'Ecole National Superieure d'Informatique et de Mathematiques Appliques (ENSIMAG).

Within the Laboratory LIFIA of the Institut IMAG, INPG, Professor Crowley directs the IMAG project PRIMA. PRIMA has as its goal the development of techniques for integrating perception, action and reasoning. Professor Crowley is the technical coordinator of project ESPRIT basic research project EP 7108, "Vision as Process", and serves as the coordinator for the European Computer Vision Network (ECVNet).

In the area of mobile robotics, Professor Crowley has developed systems for world modeling and navigation using computer vision and ultrasonic range sensors. Versions of these systems are used commercially by Denning Mobile Robotics, and Transitions Research Corporation. Professor Crowley has participated in the development of the navigation and perception systems for the project MITHRA (EUREKA 110), a family of surveillance robots. He is currently coordinator of the DG XII Human Capital and Mobility network SMARTnet, whose subject is the development of techniques for a mobile autonomous surveillance robot.

Professor Crowley has published two books, two special issues of journals, and over 60 articles on vision and mobile robotics.

Diplomas

Ph.D. Electrical Engineering, 1982, Carnegie-Mellon University
M.S. Electrical Engineering, 1977, Carnegie-Mellon University
B.S. Electrical Engineering, 1975, Southern Methodist University

Employment History

Since February 1985:
Professeur at the Institut National Polytechnique de Grenoble.
Researcher at LIFIA, Directeur of groupe Vision and Mobile Robotics.
Teaching activities in Artificial Intelligence, Vision and Robotics.
Doctoral Theses Directed : 10 Completed, 7 underway.

January 1982 to January 1985:
Research Scientist, The Robotics Institute,
Carnegie-Mellon University, Pittsburgh, Pa. , U.S.A.
Director of the Laboratory for Household Robotics

May 1980 to December 1981:
Research Associate, The Robotics Institute, Carnegie-Mellon University, Pittsburgh, Pa..
Research Acitivities in measurement, representation and matching of 3-D form.

Professional Activities

1) member: Conseil d'administration de l'ENSIMAG, Conseil de l'IMAG
2) member "Board of Editors" Robotics and Autonomous Systems, North Holland.
3) participant in Program commitees: IAS-II, IROS-II, E.C.C.V. 90, 7th SCIA (91), EUSIPCO-88, E.C.C.V.-92, IAS-III. ECCV '94, IROS '94, IAS IV.
4) Member of the Organising Committee :
"IEEE I.R.O.S.-II", Tsukuba, Sept. 1989, "I.A.S.-II", Amsterdam, December 1989
"European Conference on Computer Vision", Antibes, April 1990

“Second European Conference on Computer Vision”, Genoa, Italie, Mai 1992.
"I.A.S.-III", Pittsburgh, November 1992, Intelligent Autonomous Vehicles, Paris, Oc. 1994.
5) Expert for the European Commission DG III ESPRIT Programme (IPSS, MEL, CIME, BR)