TLA Based Face Tracking

Matthew Turk, Changbo Hu, Rogerio Feris, Farshid Lashkari, Andy Beall‡
Computer Science Department ‡Psychology Department
University of California, Santa Barbara, CA 93106

Abstract

Human face tracking (HFT) is one of several technologies useful in vision-based interaction (VBI), which is one of several technologies useful in the broader area of perceptual user interfaces (PUI). In this paper we motivate our interests in PUI and VBI, and describe our recent efforts in various aspects of face tracking in the Interaction Lab at UCSB. The HFT methods (GWN, EHT, and CFD), in the context of VBI and PUI, are part of an overall "TLA approach" to face tracking.

> **TLA** /T-L-A/ n. [Three-Letter Acronym] 1. Self-describing abbreviation for a species with which computing terminology is infested. 2. Any confusing acronym…. (From the Jargon File v. 4.3.1)

1 Introduction: Perceptual interfaces

The interface between people and computers has progressed over the years from the early days of switches and LEDs to punched cards, interactive command-line interfaces, and the direct manipulation style of graphical user interfaces. The "desktop metaphor" of graphical user interfaces, a.k.a. WIMP interfaces (for Windows, Icons, Menus, and Pointing devices), has been the standard interface between people and computers for many years. Of course, software and technology for human-computer interaction (HCI) is not isolated from other aspects of computing. Computers have changed enormously over their short history, increasing their speed and capacity, and decreasing component size, at an astounding rate. The size of computers is shrinking, and there are now a plethora of computer devices of various sizes and functionality. In addition, there are many non-GUI (or "post-WIMP") technologies, such as virtual reality, speech recognition, computer vision, haptics, and spatial sound, that promise to change the status quo in computer-human interaction.

One can view human-computer interaction as a hierarchy of goals, tasks, semantics, and syntax, as shown in Figure 1. The goal level describes what a person wants to do, independent of the technology – talk with a friend, for example. Tasks are the particular actions that are required to attain the goal – e.g., locate a telephone, dial a number, talk into the headset. The semantics level maps the tasks onto achievable interactions with the technology, while the syntax level specifies the particular actions (such as double clicking an icon) that accomplish a subtask.

One may view user interfaces as a necessary evil, because they imply a separation between what one wants the computer to do and the act of doing it [11], i.e., a division between the goal level and the task, semantics and syntax levels. This separation imposes a cognitive load upon the user that is in direct proportion to the difficulty and awkwardness that the user experiences. Poor design, to be sure, exacerbates the problem, giving rise to the all-too-common experience of frustration when interacting with computers.

This frustrating user experience can certainly be improved upon in many ways, and there are many ideas, initiatives, and techniques intended to help – such as user-centered design, 3D user interfaces, conversational interfaces, intelligent agents, virtual environments, and so on.

One point of view is that direct manipulation interfaces, such as the GUI/WIMP model, where users manipulate visual representations of objects and actions, and "information appliances" [8], which are devices built to do one particular task well, will alleviate many of the problems and limitations of current computer interfaces. Although this is very likely true – and such devices may well be commercial successes – it is not clear that this interface style will scale with the changing landscape of form factors and uses of computers in the future.

To complicate things, it is no longer obvious just what "the computer" is; the largely stand-alone desktop PC is no longer the singly dominant device. Rapid changes in form factor, connectivity, and mobility, as well as the continuing effects of Moore's Law, are significantly altering the computing landscape. More and more, computers are embedded in objects and systems that people already know how to interact with (such as a telephone or a child's toy) apart from their experience with stand-alone computers.

So what might replace, or at least complement, the current HCI paradigm? In recent years, some have argued