Metaphor Machines: Past and Future of Human-Computer Interaction

“An interface is by nature a form of artistic imitation: a *mimesis*,” Brenda Laurel wrote in her important 1984 essay “Interface as Mimesis.” She went on to write that “if designing interfaces feels like painting on cave walls by flickering torchlight, it is only because we, the designers, have not availed ourselves of better illumination: the science of the mimetic arts, poetics” [@laurel_brenda_interface_1986, 67]. Laurel, who started her academic career in theater studies, went on to work for companies like Apple, Activision, Atari, and Sun Microsystems. Her work, cited in more than 120 technical patents, reminds us of the close link between poetics and the design of human–computer interaction. It is all the more important today, as immersive computational environments begin to structure experience beyond the merely instrumental. Computers mediate in the interface between the public and the private, between art and politics, and between forces of capital and control. The simulacrum requires that we advance a reflective “science of the mimetic arts” lest we lose a sense of what Michael Taussig has called the space ulterior to mimesis [@taussig_mimesis_1993, 129-144]. In other words, the suspension of disbelief must remain, as it was in Samuel Coleridge’s original formulation, a *willful* act, containing further a “semblance of truth” required to animate the shadows of imagination [@coleridge_biographia_1917, 6-7]. An involuntary or worse yet forced suspension of disbelief can only lead to total critical disempowerment.

Before the 1980s, the dominant paradigm of interacting with machines was the dialog [@shaw_joss:*1964; @cameron_dialog:*1967; @gaines_timesharing_1986; @martin_computerized_1970; @martin_design_1973]. Already in 1950, Alan Turing imagined a conversation between a critic and an artificially intelligent poet:

> `Interrogator:` In the first line of your sonnet which reads “Shall I compare thee to a summer’s day,” would not “a spring day” do as well or better?
> `Witness:` It wouldn’t scan.
> `Interrogator:` How about “a winter’s day,” That would scan all right.
> `Witness:` Yes, but nobody wants to be compared to a winter’s day.
> `Interrogator:` Would you say Mr. Pickwick reminded you of Christmas?
> `Witness: In a way.`Interrogator:`Yet Christmas is a winter's day, and I do not think Mr. Pickwick would mind the comparison.`Witness:‘ I don’t think you’re serious. By a winter’s day one means a typical winter’s day, rather than a special one like Christmas [@turing_computing_1950].

The goal of conversational programming was to provide a similarly naturalized dialog between the operator and the machine. Although machine understanding of informal human speech does not develop in earnest until the twenty-first century, interactive computing environments like DIALOG and JOSS attempted to abstract away from machine language towards interfaces that understood a limited number of English words. JOSS, an experimental on-line computing system created by the RAND corporation in 1963, consisted of a typewriter connected to the JOHNNIAC computer. Prior to JOSS, programmers would interact with the JOHNNIAC machine via keyboard and punch card, initially using octal number notation and then binary assembly language [@gruenberger_history_1979]. These methods of programming were time consuming and prone to error. "An octal desk calculator was nice to have," a historian of the system would later quip [@gruenberger_history_1979, 58]. The JOSS experiment was meant to introduce a kind of an interpreter that sat between the human operator and the machine, facilitating communication in a friendly, English-like language. Think of JOSS as the "user's computing aide and a single contact with the computer," the engineers wrote [@shaw_joss:*1964, 456]. Instead of feeding punch cards into the machine, the operator would now use a standard typewriter. Instead of flipping switches the operator typed words. JOSS thus referred both to the "simple language" for machine instruction and to this new "remote console" way of interacting with the computer. JOSS and the user would "take turns controlling the typewriter" in the words of the documentation. The computerized aide understood simple commands like `do`, `go`, and `type`. Given a number of predetermined commands, it would attempt to respond with a result. When confused, or when given an unfamiliar command, its default for an unspecified error mimicked human confusion: "Eh" [@shaw_joss:*1964]?

This mode of conversational interaction was captured also in one of the earliest text adventure games, the *Colossal Cave Adventure*, designed by Will Crowther in 1975 [@montfort_twisty_2003]. The following dialog illustrates the call-and-response game play typical of the genre:

```
You are standing at the end of a road before a small brick
building. Around you is a forest. A small stream flows
out of the building and down a gully.

> enter

You are inside a building, a well house for a large spring.
There are some keys on the ground here. There is a shiny
brass lamp nearby. There is food here. There is a bottle
of water here.

> get keys

OK

> get lamp

OK

> exit

You're at end of road again.
```

Although much more accessible and interactive than communication in octal or binary machine code, the conversational model posed several significant downsides. While resembling human communication, it in fact used only a limited set of words. Researchers from the U.S. Air Force academy would later write:

> The lower cost of computer access and the proliferation of on-line systems produced a new breed of users, people whose expertise was in some area other than computer technology. As their initial fascination with conversational computing wore off, users reported experiencing feelings of intense frustration and of being "manipulated" by a seemingly unyielding, rigid, intolerant dialogue partner [@walther_line_1974, 379; @gaines_timesharing_1986, 15].

The conversational intermediary of the Colossal Cave Adventure understood roughly 295 commands, and knew about 1,600 words in response arranged into several hundred canned phrases.[1] Faced with an unfamiliar word, it could only repeat "That's not a verb I recognise."

By the 1980s, a new breed of metaphorical interfaces gained widespread prominence. If *Colossal Cave Adventure* epitomized the conversational model of computing, games like the early but popular *Pong*, *Space Invaders*, and *Donkey Kong* epitomized the paradigm of "direct manipulation." According to Ben Shneiderman, the researcher who coined the term in 1982, direct manipulation involved three key principles:

1. Continuous representation of the object of interest.
2. Physical actions or labelled button presses instead of complex syntax.
3. Rapid incremental reversible operations whose impact on the object of interest is immediately visible [@shneiderman_future_1982, 251; @hutchins_direct_1986, 91].

The goal of direct manipulation was therefore to achieve a perfect correspondence between "representation" and the "object of interest." Shneiderman cites Gottfried Leibniz in reference to symbolic notation, which in theory should also "express the exact nature of a thing briefly and, as it were, picture it." For Leibniz, the symbol, an exact and portable picture of an idea, could subsequently diminish the mental effort required for abstract thought, leading to a "great advantage for discovery" [@leibniz_briefwechsel_1899, 375; @shneiderman_direct_1983, 57]. The calculus of Leibniz and his symbolic notation allowed mathematicians to represent infinitely small and infinitely large

[1] Rick Adams maintains the source code for a number of early versions of the game at http://rickadams.org/adventure/e_downloads.html. I used PDP-10 Fortran version and ran `tr`, `uniq`, and the `wc` Unix utilities to isolate and count the unique words from the data file.

numbers in print—ideas that would not fit on the page or in the mind otherwise [@cajori_history_1923; @thurston_leibnizs_1973; @grabiner_is_1974]. For Shneiderman and others, a computer game like *Pong* similarly enabled the direct manipulation of complex abstractions, allowing a player to control a virtual table-tennis paddle by rotating a physical knob on a gaming console. The movement of the physical knob corresponded directly to the movement of the paddle—clockwise for up and counter-clockwise for down—thus achieving the correspondence between "operation" and the "impact on the object of interest" [@shneiderman_direct_1983, 60]. Without such a "direct" linkage, the physics of the game simulation would be too complex for players to handle effectively, if they were asked to play by writing formulas in code, for example, or to direct the paddle through dialog.

The conversational model of human–computer interaction was hindered by what researchers considered an arbitrary symbolic relationship of the sign to its signifier. In this way, when using *EMACS*, a text editor commonly found on UNIX systems of the time, one would enter the command `k` in combination with other keys to delete or to "kill" a file, while on other systems, Shneiderman complained, `k` stood for "keep a file," the opposite from killing it [@shneiderman_direct_1983, 65]. In the conversational model, the command stood in an arbitrary relationship to the intended effect, whereas in the direct manipulation model the knob and the movement of the paddle related mimetically. To be more precise, the direct manipulation paradigm advocated for an "iconic" relationship between representation and the object of interest [@norman_user_1986, 110]. Like onomatopoeic words, the movement of the knob resembles the movement of the thing it represents. Edwin Hutchins, Donald Norman, and James Hollan write: "There is an economy here in that the user's knowledge of the structure of the surface acoustical form has a non-arbitrary relationship to meaning [...] The same sort of thing can be done in the design of interface languages" [@norman_cognitive_1991, 123]. An iconic image requires no explanation; it is intuitive.[2] Instead of searching for the right command the user relies on the habituated affordances of real-world objects, like table tennis paddles and paper trashcans, to manipulate virtual objects, like computer games and file systems. The move between "conversational" and "direct" interaction models could in this way be explained as a shift from a system of symbolic signs which stand in arbitrary relationship to their referents to a system of iconic signs that stand in a mimetic relationship.[3]

Crucially, within the "direct manipulation" paradigm, the full immersion in the mimetic context of the virtual object is aimed at the suspension of disbelief [@laurel_brenda_interface_1986, 76]. Through use, the mimetic icon usurps

[2] I am using the traditional Peircian distinction between symbol, icon, and index. Peirce writes "icons are are so completely substituted for their objects as hardly to be distinguished from them" [@peirce_algebra_1992, 226].

[3] Charles Sanders Peirce, a philosopher of language whose vocabulary I have been using here, suggests Simulacra as a possible synonym for Icon, citing also Plato's *Phaedrus* in relation to the Greek or imitation [@peirce_excerpts_1998, 481; @plato_plato:_1999, 250a-b].

the "object of interest." The "directness" of direct manipulation in fact constitutes only an illusion of directness. Recall in this regard the Heideggerian insight into the nature of tool use. In *Being and Time*, Heidegger writes about the particular *handiness* of a tool, like a hammer. He writes, "the less we stare at the thing […] the more actively we use it, the more original our relation to it becomes and the more undisguisedly it is encountered as what it is, a useful thing" [@heidegger_being_1996, 65]. For Heidegger, there was no way to understand a tool like a hammer theoretically, by detached reflection. One must understand it through use, in what he called *circumspection* [*Umsicht*]—the awareness of the object ready-to-hand [*zuhanden*] [@heidegger_sein_1967, 69]. We relate to the tool in the process of using it; it comes into focus at the periphery of tool use [@heidegger_being_1996, 65]. The metaphoric device frustrates the Heideggerian vision of technology that reveals. At hand, the device takes shape as a keyboard, a mouse, or a touch screen. But in the mind, as a simulation, it dissembles to behave like a page, a folder, or a wastepaper bin. The tool—a keyboard or a touch screen—is not encountered for what it is. Nor does a simulated tool reveal anything about the material it affects. The simulation produces knowledge neither about the thing being simulated (paper or books, for example) nor about the medium giving rise to the simulation (computational device, in our example).

The simulation is rather designed to conceal the mediated nature of virtual experience. It manufactures what Laurel and others call an "interactive mimesis" and "first personness," the experience of "directly living and acting within the world established by the computer" [@laurel_brenda_interface_1986; @norman_user_1986, 490-1]. Hutchins, Hollan, and Norman write: "when an interface presents world of action rather than a language of description, manipulating a representation can have the same effects and the same feel as manipulating the thing being represented" [@hutchins_direct_1986, 98-99]. Similar to Heidegger's distinction between practice and theory, the theorists of direct manipulation differentiate between action and description. Unlike Heidegger, who thought about tools at hand, direct manipulation theory imagines the handling of representations not objects—as evidenced by the somewhat strained language of metaphor machines in which "hierarchies allow access to metaphors within the library" and "users manipulate sheets in the notebook metaphor" [@pajak_electronic_1992; @glaser_graphical_1995]. The actual instruments enabling the manipulation of image or word on the screen disappear entirely, giving rise to the immersive, interactive, first-person experience, without the feeling of mediation. "The user of a well-designed *model world* interface can willfully suspend disbelief that the objects depicted are artifacts of some program and can thereby directly engage the world of the objects," Norman et.al. write [@hutchins_direct_1986, 99]. In this view, the dialogic model of tool use encouraged by *Colossal Cave Adventure*—get keys, get lamp—is dismissed as nothing more than just "using the computer" [@laurel_brenda_interface_1986, 74]. "End users are not interested in *making* representations," Laurel writes. "They want to move around *inside* one," favoring the mimetic context over their actual, physical surroundings [@laurel_brenda_interface_1986, 75]. Verbal representations, like “get keys” or “get lamp,” puncture the first-person illusion. By contrast, the fully immersive model supports the “sensation of directness,” in which direct, that is iconic, expressions, “behave in such a way that a user can assume that they, in some sense, *are* the things they refer to” [@hutchins_direct_1986, 110].

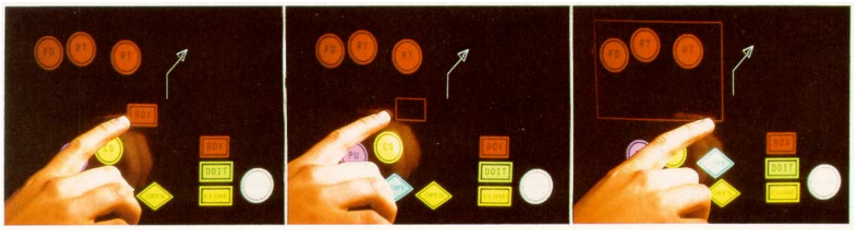

Figure 1: Direct manipulation of simulated objects, 1984. [@minsky_manipulating_1984, 199]

The principles of direct interaction emerging in the 1980s stood against what Laurel called the “ill-formed” presence of the mediator, in the interfaces of the previous era like JOSS and the *Colossal Cave Adventure*. For those advocating the direct interaction paradigm of computer use, conversational computing prevents the unmediated “pleasure” and “catharsis” of direct engagement [@laurel_brenda_interface_1986, 75]. The intermediary takes the place of the player in a game: it swings the sword for her, takes a beating, and reports on the experience, standing in the way of immediacy. Laurel writes:

> In the file management example, the intermediary takes the form of command menus that are invoked in order to activate processes in the program that will create the desired results. The user does not have the experience of pushing files around, stowing them and grabbing them, or blowing them away. Instead, the user has the experience of communicating with the file management intermediary [@laurel_brenda_interface_1986, 75].

In combination, the ideas behind cognitive domain blending and direct manipulation gave rise to the now ubiquitous WYSIWYG (what you see is what you get) interfaces, put into mass production by Apple, Xerox, and other companies in the early- to mid-1980s. The Xerox 8010 Star workstation introduced in 1981 and the Xerox 6085 Daybreak workstation, introduced in 1985, heralded the era of accessible, metaphor-driven personal computing, characterized by the use of virtual graphical objects like windows, icons, desktops, folders, and buttons. The Star and Daybreak workstations were some of the earliest machines to put the principles of metaphoric domain blending and direct manipulation into action. The interface was meant to “reveal the structure” of the simulated objects intuitively, without training or lengthy written explanation. In Laurel's words, the mimetic interface employed "logic and aesthetics to create representations that *engage* humans in pleasurable ways" [@laurel_brenda_interface_1986, 85]. The mimetic context is simply "the experience we desire," Laurel wrote. Direct participation enabled actors to experience "the full pleasure of the mimetic form" [@laurel_brenda_interface_1986, 75].

Designers advocating for direct manipulation understood the trade-offs that came with an emphasis on such immersive and mimetic experiences [@hutchins_direct_1986, 118]. Mimesis fundamentally relies on the user's familiarity with the source domain. In this way, we understand what to do with "folder" icons because we know how folders behave in real life. Direct mimetic manipulation does not however tell us anything new about the capabilities of virtual folders. Immersion precludes critical reflection beyond the pre-defined confines of the modeled world. The user has access only to what the simulated experience allows and can provide.

Problematically, ideas behind mimetic immersion contain a kind of a logical fallacy. The literature on direct manipulation often refers to the example of driving a car to illustrate the type of an interface by which inputs and outputs correlate directly. Thus instead of giving complicated commands to a vehicle, a driver turns the wheel to the right, and the car immediately follows. A direct causal link exists between the steering wheel and the car's axle. Similarly, when operating a computer game like *Space Invaders* via a joystick, the player experiences an immediate correspondence between the movement of the controlling mechanism and the movement of the player-controlled space ship on the screen. The car is the "direct object of interest" for a driver, just like the virtual space ship is for a gamer. But unlike video game players, authors attempting to delete sensitive information are not interested in the virtual representation of their documents. The object of their interest is not mimetic. They would like to erase the document itself, located somewhere within the machine. In cases like these, inhabitants of the simulated, virtual world have an interest in objects outside of it. The paradigm of direct manipulation instead veils the object from view, suspending the rules of physical interaction in favor of the virtual. Far from being direct, the iconic representation of the document—the image of a file—usurps the physical object, the file itself. Where in the conversational paradigm the nature of the mediation was at hand, open to circumspection, in the direct manipulation paradigm the simulacrum occludes the very nature of the simulation.

Speculative Formalism

To what extent do we need to see and understand the mechanisms of the machine giving rise to the metaphor? In his 1949 paper on "The Genesis and Speed of Telegraph Codes," Frank Halstead noted that "the practical upper limits of [telegraph transmission] speed will also be limited by the ability of some human beings to operate a keyboard, until such time as electrical connection be made direct with the receiver's central nervous system" [@halstead_genesis_1949, 451]. The history of human–computer interaction began with the manipulation of physical switches, first by hand and then by removable storage media like paper tape and punch card. The next phase was dialogic and conversational in nature. Conversational computing introduced the idea of a mediating agent, an interpreter, who could translate from a limited number of natural human language commands into the specialized vocabulary of signals that could alter the machine configuration. The "direct manipulation" school of human–computer interaction has led us further towards mimesis, simulating machine states as virtual environments resembling real-world objects and their properties.

The trajectory from direct *physical* manipulation to direct *virtual* manipulation leads to the totalizing loss of resemblances and designations that Baudrillard warned us about. At its logical extreme, the simulacrum supplants the thing being simulated. It appears as hyper-reality, the experience of unmediated interaction without awareness or sense of the underlying referent. All "objects of interest" within such a modeled world are fabricated objects. They are thus limited to the external, artificial constraints imposed by their makers. Such manufactured experiences present us, the "users," with compelling, possibly cathartic, metaphors. The metaphor extends far beyond entertainment, subsuming all spheres of social activity mediated by computers. The manipulation of metaphors—a reflective, not merely circumspect inhabitance of virtual worlds—therefore requires not just a willing suspension of disbelief, but also poetics: hermeneutics, close reading, distant reading, deconstruction, morphology, in short precisely the legacy of interpretive practice that stretches from Aristotle to Susan Sontag and beyond. The suspension of disbelief otherwise threatens to become a permanent condition, alienating as it is pleasurable or cathartic.

At the time of writing this book, our society stands at the threshold of a new paradigm of interacting with computers and hence with each other. Direct brain to computer interfaces are common enough today to be turned into a toy.[4] Early brain–computer interfaces used either imprecise "noisy" electroencephalographic (EEG) scalp sensors or electrode implants that required invasive surgery. In 2004 a team of scientists developed a way of controlling "a one-dimensional computer cursor rapidly and accurately" using electrocorticographic (ECoG) activity recorded from the surface of the brain [@leuthardt_braincomputer_2004; @miller_spectral_2007]. And in 2015, a quadriplegic woman piloted an F-35 Joint Strike Fighter using her brain in a simulation developed by the University of Pittsburgh's Human

[4] The American toy giant Mattel makes a game called "Mindflex." The Frequently Asked Questions page includes the following prompt: "Have you ever dreamed of moving an object with the power of your mind? Mindflex Duel™ makes that dream a reality! Utilizing advanced Mindflex Duel™ technology, the wireless headset reads your brainwave activity. Concentrate…and the ball rises on a cushion of air! Relax…and the ball descends. It's literally mind over matter!" (@mindflex_mindflex:_2015)

Engineering Research Laboratories in collaboration with the Defense Advanced Research Projects Agency (DARPA) [@collinger_collaborative_2014; @prabhakar_how_2015]. On the surface, the advance of brain–computer interfaces seems to bring us closer to the vision of increasing directness: the "no interface" movement in design, interaction between people and machines without intermediary symbols, words, or images. *No Interface*, *Don't Make me Think*, and *Hooked: How to Build Habit-Forming Products* are some of the most well-written and best-selling titles advancing this argument [@krug_dont_2014; @eyal_hooked:_2014; @krishna_best_2015]. Military applications aside, we must prepare for a future in affective, mystical arts, the ultimate loss of references and resemblances, where works of art are to be experienced emotively past the senses: literature without language or representation, painting beyond media, asemiosis—messages lacking a sign.

However futuristic such possibilities may seem to us today, we should not lose sight of the mediated nature of human–computer interaction in all of its forms: paper, ink, silicone. Media intercedes. As technological dreams become reality they also grow less interesting and more mundane. We tend not to think about them; they become habit. Habituation smoothes the rough edges of irreconcilability between human and machine. The tool recedes from view, beginning to seem like a natural and direct extension of the body. I do not mean to dismiss the mastery of those engaged in creating the illusion of directness. Pleasure and catharsis are important in some contexts. In other contexts, we must privilege reflection, analysis, and interruption. Computational poetics should therefore aim to arrest the encounter: to make it seem less natural, to bring the receding media to light, and to expose the mediating systems supporting the simulation. A measure of discomfort in the fit between bodies, keyboards, and screens ensures our ability to structure the encounter on our own terms: to opt in when useful and to opt out when necessary. My concern is not with the metaphysical entailments of a possible posthuman future. The illusion of directness belies the very human mechanisms of command and control. The simulation ultimately embodies specific power structures in an economy of exchange between physical and mental resources. In this market there are those who profit and those whose virtual avatars become a commodity.

The above comments apply to computation generally. We will spend the following chapters peeling apart the layers of the literary and the bibliographic simulation in particular. Once the electronic book can be perceived as computation in kind, we can begin to examine the incongruence in the structures of meaning-making at the sites of storage and projection. If we consider "what you see is what you get" as a type of a commitment to a motivated metaphor, the following chapters will help us understand the ways of seeing and the material particulates of getting. The electronic book shall come to fore as a literary device.