Semiotic Considerations in an Artificial Intelligence-Based Art Practice

By Michael Mateas

No. 29 – 2003

Abstract

In my work I engage in a hybrid practice combining artificial intelligence (AI) research and art marking, a practice I call Expressive AI. Computers are fundamentally meaning machines - the long chains of meaningless causal processes that comprise computation can be linked to culturally meaningful signs such that computers can participate in processes of signification. AI consists of coupled rhetorical and technical strategies for structuring computational processes. Artists can consciously manipulate these rhetorical and technical strategies so as to build machines with powerful authorial affordances for crafting audience experiences.

I find myself engaged in a practice combining artificial intelligence (AI) research and art making, a practice I call Expressive AI. This paper briefly explores semiotic issues arising in an AI-based art practice.

1. Expressive AI takes as its starting point that the essence of the computer as a representational medium is not the ability to intervene in the production of three dimensional forms or visual imagery, nor the ability to interact with a participant/observer, nor the ability to control electro-mechanical systems, nor the ability to mediate signals sent from distant locations, but rather computation, that is, processes of mechanical manipulation to which observers can ascribe meaning.

2. Any art intervention can be viewed as the manipulation and modification of a network of flows - flows of signification, matter, and energy. The flows found within the institutional and organizational particularities of situations such as a busy freeway, an office in a large bureaucratic organization, a party, a riot, or, perhaps, an art gallery, are modified by the art intervention, at the same time these flows reciprocally enact the interpretation of the intervention.

3. An AI-based art work attempts to be porous to these flows, to actively engage in interpretation, modification, and generation of these flows. That is, human

participants in these flows will interpret the AI-based artwork as engaging in its own semiosis, of in some sense understanding the shared situation, and acting on this understanding. An inter-semiotic relation is established in which human subjects ascribe subject status to the AI-based work.

4. AI, and its sister discipline Artificial Life, consist of both technical strategies for the design and implementation of computational systems, and a pared, inseparable, tightly entangled collection of rhetorical and narrative strategies for talking about and thus understanding these computational systems as being intelligent (engaging in semiosis), and/or alive.

5. These rhetorical strategies enable researchers to use language such as "goal", "plan", "action", and "embodiment" to simultaneously refer to specific technical entities (pieces of program text, algorithms, electronic and mechanical devices) and make use of the systems of denotation and connotation these words have when applied to human beings. This double use of language is both a resource and a danger. When the fine line walked between the metaphoric and technical uses of the language is erased, the technical meaning becomes naturalized, doing violence to the original richness of the term, reducing its productiveness as a technical term, and contributing to rigidity in human notions of subjectivity.

6. There is an uncomfortable relationship between a purely relational (and thus literally meaningless) technical manipulation of computational material, and the interpretation of this computational material by a human observer. Simon and Newell posited the physical symbol system hypothesis as a fundamental assumption of AI. This hypothesis states that a physical system consisting of a material base which can take on various configurations (call these configurations "symbols") and a material process which manipulates these physical configurations to yield new configurations is sufficient for the production of intelligent behavior. This formulation immediately produces an interpretation problem in which an external observer is necessary in order to view the material configurations as signs such that intelligence can be observed in the material production of sign from sign. Interpretation, with all of its productive open-endedness, is thus crucial to the definition of intelligent system, but is usually pushed to the background of AI practice.

7. The necessity of rhetorical strategies of interpretation is not avoided by "subsymbolic" techniques such as neural networks or genetic algorithms, nor machine learning methods based on generalization from training data, nor behaviorist robotic techniques which link sensors to effectors through stateless combinatorial circuitry or finite state machines. All these approaches still require the interpretation of an observer in order to make sense of the input/output relationship exhibited by the system, to select the primitive categories (features) with which the inputs are structured, and to tell stories about the processes producing the

input/output relationships. These stories are essential for thinking through which technical constructions to try next, that is, for simultaneously defining a notion of progress and a collection of incremental technical constructions that make progress according to this notion.

8. The rhetorical strategies used to narrate the operation of an AI system varies depending on the technical approach, precisely because these interpretative strategies are inextricably part of the approach. Every system is doubled, consisting of both a computational and rhetorical machine.

9. The central problem of AI is often cast as the "knowledge representation" problem. This is precisely the problem of defining structures and processes that are simultaneously amenable to the uninterpreted manipulations of computational systems and to serving as signs for human subjects. This quest has driven AI to be the most promiscuous field of computer science, engaging in unexpected and ingenious couplings with numerous fields including psychology, anthropology, linguistics, physics, biology (both molecular and macro), ethnography, ethology, mathematics, logic, etc.

10. Processes of interpretation play an integral role in both the interpretation of a running AI system, and in the construction of that system. Thus, an AI-based artwork must not only be a porous participant in a network of flows, interpretable as a subject in a process of inter-semiosis, but must also function as a machine within which an artist can inscribe her artistic intentionality. The doubled (computational and semiotic) internal structure of the system must allow an artist to predict the external interpretations an audience will make of the completed work. An effective internal structure offers the appropriate authorial affordances to craft the audience experience.

11. An AI-based artwork is a semiotic system which is productive of a (potentially large) number of syntagms. AI-based artworks are thus *generative*; computational processes provide the combinatoric machinery necessary to select terms out of the fields of potential terms provided by the system. It is this production of variable syntagms in different viewing situations that allows the viewer to ascribe subject status (as some kind of being capable of participating in semiosis, though not necessarily a human being) to the work.

12. The internal structure of the machine, the computer code, wires, circuits and motors out of which a work might by constructed, is itself a syntagm of the semiotic system defined by the implementation strategy employed to construct the piece. The implementation strategy is precisely the architecture, that is, the knowledge representation languages, processes, modules, and relationships between modules out of which the system is constructed. Building an AI-based work thus means constructing a semiotic system of implementation (an architecture) such that it allows the construction of a syntagm (the technical system viewed in its static form)

such that when this syntagm is executed (the technical system viewed in its dynamic, procedural form), it becomes a semiotic system autonomously productive of its own syntagms.

13. The program code, considered as a sign system, relates two planes: a plane of expression containing the space of all possible pieces of program text (the marks on a screen or page), and a plane of content containing the space of all potential executions. That is, a piece of program code (e.g. "x = 1") is a signifier signifying the mental concept of the effect of executing this code. An architecture re-articulates the relationship between the plane of expression and the plane of content, establishing novel relationships between code signs and potential executions of these code signs.

14. We may thus speak of two distinct semiotic systems: system$_1$, the sign system defined by the architecture within which the artist crafts a specific interactive work, and system$_2$, the running work, autonomously productive of its own syntagms (and possibly responsive to audience interaction). An architecture has appropriate authorial affordances when it satisfies two conditions:

1) The implementation syntagm (the code written by the artist within system$_1$) serves as a meta-language to describe the properties of the audience syntagms (the sensory display produced by system$_2$).

2) The implementation syntagm, when executed, actually implements a system$_2$ that produces the audience syntagms that the implementation syntagm purports to talk about.

15. The signs of both system$_1$ and system$_2$ are multi-articulated; their meaning arises not only from the syntagmatic and paradigmatic constraints established by their respective code systems, but also from a collection of sign systems *outside* of the code systems. This collection of external code systems is the rhetorical system. Both authors and audiences make use of the rhetorical system in narrating the operation of the system and forming intentions with respect to the system. The rhetorical system introduces an interpretive surplus to the code systems.

16. For the audience, the signs produced by system$_2$ have an interpretive surplus. The implementation syntagm never completely describes all the properties of the audience syntagm; though system$_2$ literally prescribes the possible elements and spatial and temporal relationships between elements of the audience syntagm, a portion (perhaps a large portion) of the signification is determined by external sign systems. For example, if the material realization of the audience signs consist of triggered video segments, the meaning of these video segments is determined not only by their participation within the potential fields of system$_2$, but also by the established conventions of video language. This interpretive surplus occurs because system$_2$ operationalizes a meta-language (the implementation syntagm)

for describing the audience experience. The signifieds of this meta-language are themselves signs, participating in external sign systems, which are handled by the meta-language. The crafting of these external, handled signs, becomes an irreducible problem in design and aesthetics. These handled signs must be crafted to marshal the signifying resources of these external sign systems in such a way as to match the purported meanings of the code system.

17. For the artist, the signs of system$_1$ have an interpretive surplus - we may refer to this as an architectural surplus. The architectural surplus can be understood as one or more meta-languages, in which the signs in system$_1$ (the implementation syntagm) form the content plane, and as one or more connotative systems, in which signs in the meta-language form the plane of denotation. Together, these external systems provide ways of talking about code, linking specific code constructs to ordinary language words such as "goal", "embodiment", or "symbol". This movement, from code system, into ordinary language, and back into code system, creates a circulation of signs that suggests both new ways of using the architecture and new architectural elaborations. The architecture becomes an active conceptual probe, a machine to think with.

18. Thinking with the architecture suggests new audience experiences, creating a feedback loop between authorial intention and the details of the *total* system (code + rhetoric). But establishing this interpretive framework, the planes of connotation and meta-language, takes real work. It is the outcome of a practice that simultaneously articulates novel code machines *and* ways of reading and talking about them. In contrast, a practice that views the system as a hack, as a means to an end, will likely construct systems with poor authorial affordances, lacking both the code system relationships and rich rhetorical frameworks necessary to enable richly and deeply interactive audience experiences.