Gestural Syntax HCI Systems in Coding: A Form of Somatic Coding

Neill Kaipo Shikada

ATLAS Institute, University of Colorado Boulder

Introduction

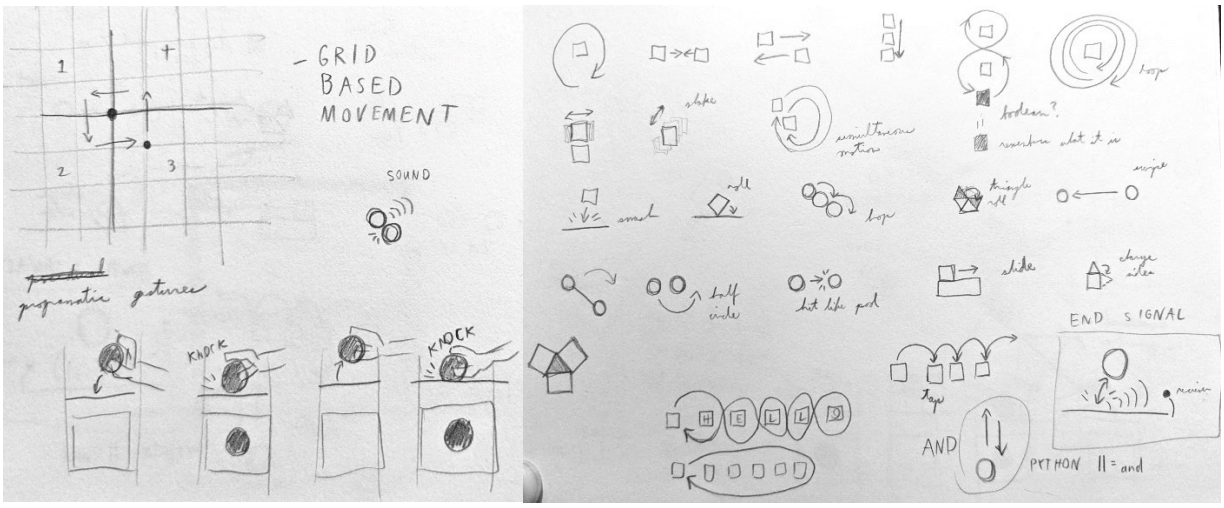
Tangible coding interfaces are becoming more ubiquitous in HCI research. As these systems evolve, it becomes important to consider the somaesthetic qualities of these systems. In Kristina Höök's manifesto "Designing with the Body: Somaesthetic Interaction Design", she coined the term "soma design—a process that reincorporates the body and movement into a design regime that has long privileged language and logic" (XVI). Many tangible interfaces throughout my research acknowledge the body, but not in motion. From the Reactable to Google's Project Bloks, these systems rely on users snapping things into place, but not actually using gesture as an input. I posit that this misses the point - designing systems that encourage movement is necessary for the development of systems that work with our bodies.

Background

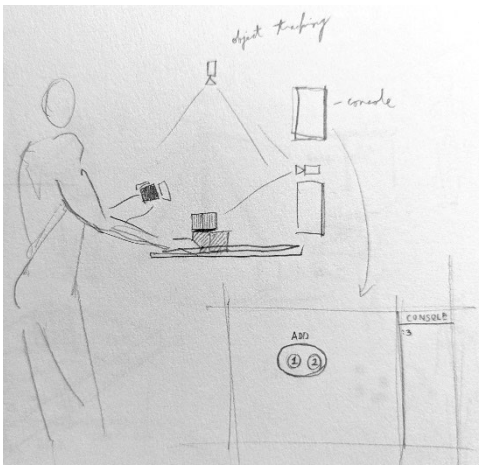
The field of contemporary HCI has begun a shift into more qualitative measures of success focused on the incorporation of the body into digital functions. This is largely driven by a disillusionment with the way our bodies interact with systems, such as the keyboard and mouse, that practically ignore the felt experience of users' physical body. This has meant that there is a "qualitative shift required, from a predominantly symbolic, language-oriented stance to an experiential, felt, aesthetic stance permeating the whole design and use cycle" (Höök, 6).

This project takes these principles and begins to apply them to a gestural interface inspired by the Reactable. Unlike the Reactable, however, this system adds gestural input as a key element to perform string manipulation and interface with ChatGPT.

Design and Implementation



Ideation



The original concept, to create a coding interface solely using gestures proved too inaccessible in the ideation phase for the short-term of this project. This focus on simplicity took me towards esoteric programming languages (esolangs) that focused on limited character usage. The esolang "Brainf*ck" proved the best form of inspiration. Its philosophy was to create a coding language with as few symbols as possible but maximize versatility in a Turing complete language. This could drive the design of a rough draft of a gestural coding syntax.

In the end, my interaction system was designed to form poetry. This proved to be an optimal use since users were mostly familiar with fridge magnet poem systems that operate similarly. The spontaneity and versatility of the ChatGPT API, which I would reference, also lent itself to this purpose, while my research into Noam Chomsky's "The Sound Pattern of English" offered some exciting insights into the ways string manipulation can provide strange quirks of the English language.

Development

In the development of gestures and their corresponding functions, I made an assumption that the best design for this prototype would supplement symbolic syntax with gestural syntax instead of relying solely on a gestural syntax. Solely using gestures would require either a large library of gestures for users to memorize or incredibly long scripts to write like those found in Brainf*ck that would tire users.

The coding process included the development of UML diagrams to determine the structure of the code - prototyped in Python using object-oriented programming. This code and documentation can be found on Github at https://github.com/KaipoCraft/Thesis-SomaticCoding.

Gestures

This program included four gestures:

1. Clockwise Circle

2. Counterclockwise Circle

3. Horizontal Line

4. Vertical Line

The clockwise circle gesture ran a string manipulation function by calling the ChatGPT API to change the currently visible words on the surface to new grammatical types. For example, altering the adjective "purple" to the noun-profession "colorist". These string manipulations ideally could be done programmatically for maximum control, however the randomness of ChatGPT's davinci-3 model provides exceptionally interesting responses and spontaneity for the construction of poems.

The counterclockwise circle gesture resets the words to their original form and clears the outputs. The horizontal line gesture adds the currently visible words to the output, while the vertical line gesture runs those words through the ChatGPT API to generate a short poem.

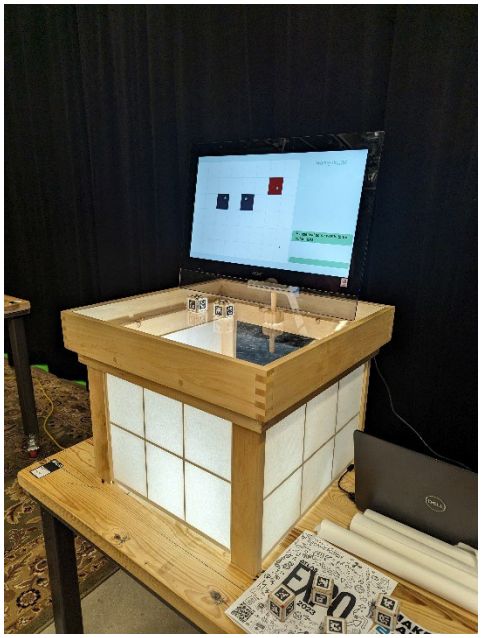
Construction

The construction process focused on using traditional wood joinery techniques to make a table interface that evokes nature. Rice paper was used on the sides to make aesthetic use of the necessary lighting fixture that would need to be placed inside the table to optimize the reading of the markers. This controlled environment would provide more consistent readings of the markers, thus improving useability. A camera was placed at the bottom of the table facing the top surface where the ArUco markers could be placed and read by the OpenCV program.

Evaluation

This interface was evaluated at the ATLAS Expo, an exposition focused on the work of students at the ATLAS institute. In this testing phase, I found many people interested in the more accessible elements of the project. Individuals would want to form their own sentence structures and poems - preferring to only use two of the program's capabilities: the left-right reading format of the program and clockwise-circular string manipulation function that would provide novel forms of words.

The "horizontal line" gesture proved to be very unresponsive. This responsiveness proved to be a challenge and in future iterations should either integrate a more dynamic call to check for gestures or a mode of user feedback that gives the users the history of their movements with a buildup bar showing how many movements left until it checks for a gesture. The memory of the program was also not accurate and adding a litany of words to the output when the "horizontal line" gesture was detected. This was caused by markers not being marked as "not visible" when they left the screen. This needs to be fixed in the code.



The limited Internet connectivity also made the API calls to ChatGPT much less efficient. The program would freeze for prolonged periods while the call was made. In future iterations, ` this needs to be optimized and perhaps the string manipulation function needs to be performed locally.

Overall, even with the UI only being composed of four gestures, it was too much for users to understand and required too much explanation. Many people lost interest or were confused, preferring to simply move blocks around and see what was happening. This type of user might benefit from more loosely coded gestures that may be able to act in spontaneous and unexpected ways to keep their interest.

Discussion and Future Work

This project's next phase will look to refine the user experience to make the interface operations more readily understandable. The gestures need to be developed to have a more fluid motion and users need more feedback on what is being seen by the software.

Scalability

This project will scale to include more versatile forms of computing. I have been in discussion with the team at Sasaki as they are developing software that enables 3D modeling using a similar interface. There are many types of usage for this device, but my main goal is to make this versatile enough to scale various programs - in essence creating a gestural coding language.

Conclusion

The lessons learned from this prototype paved the way for future development in the area of somatic coding. Incorporating gestures into coding syntax is necessary in a world where coding is the main form of work. Without somatic systems, health issues related to prolonged usage of extant HCI systems will continue to not simply encourage but necessitate sedentary forms of being.

References

Höök, Kristina. _Designing with the Body: Somaesthetic Interaction Design_. Cambridge, MA: The MIT Press, 2018.

Minsky, Marvin, Seymour Papert, and Léon Bottou. _Perceptrons: An Introduction to Computational Geometry_. [2017 edition]. Cambridge, Massachusetts: The MIT Press, 2017.

Jordà, Sergi. “The Reactable: Tabletop Tangible Interfaces for Multithreaded Musical Performance,” January 1, 2009.

Chomsky, Noam, and Morris Halle. _The Sound Pattern of English_. 1st MIT Press paperback. ed. Cambridge, Mass: MIT Press, 1991.