Formeta:3D: Posthuman Participant Historian

Daniel Eisinger
Ball State University

Steven Putt
Design Collaborative

Interactive, Real-Time, Recursive Form Generation and Realization

1

ABSTRACT

Formeta:3D is a project that engages the posthuman through the development of a machine that translates inputs from its surroundings into physical form in real-time. By responding to interaction with the inhabitants of its environs and incorporating the detected activity in the inflections of the produced form, it has an impact on the activity in the space, resulting in a recursive feedback loop that incorporates the digital, the physical, and the experiential. This paper presents the development of this project in detail, providing a methodology and toolchain for implementing real-time interaction with additive physical form derived from digital inputs and examining the results of an interactive installation set up to test the implementation.

1 Detail of one of the forms generated by Formeta:3D in real-time in response to activity observed by a webcam.

2 Previous Formeta:2D project developed by the authors as a means to explore real-time translation of inputs to a physical output.

3 One of the drawings produced by the Formeta:2D drawing machine.

INTRODUCTION

In a posthuman landscape, how is form generated? If form is autonomously designed, what are the inputs to the machine, when is the design realized, and how can the form continue to evolve? Formeta:3D is a project that explores these questions through the development of a machine that translates inputs from its surroundings into physical form in real-time, responding to interaction with the inhabitants of its environs and incorporating the detected activity in the inflections of the produced form. This in turn has an impact on the activity in the space, resulting in a recursive feedback loop that incorporates the digital, the physical, and the experiential.

Formeta:3D operates by translating motion detected in live video to 3D print toolpaths for immediate printing. In so doing, the system produces artifacts that capture in physical form the spatial activity at a given place and time. These objects are not merely records of activity, however, since they influence the human activity in a space by the fact of their evolving presence within it. The machine operates as both historian and collaborator, establishing itself as a player in human events.

This paper presents the development of this project in detail, providing a methodology and toolchain for implementing real-time interaction with additive physical form derived from digital inputs. It also examines the results of prints produced as a response to real-time inputs collected while the machine was set up as an interactive installation.

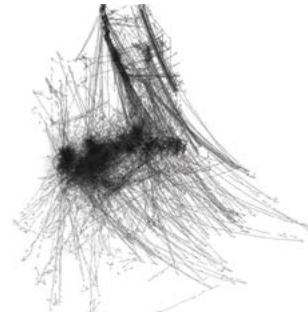
BACKGROUND

This work builds on a rich history of drawing machines that have been developed over the past half century as part of experimentation in art and architecture. In contrast to early examples in the 1950s and 1960s—including the creations of Jean Tinguely, which some reviewers would say regarded human and machine as diametrically opposed—more recent drawing machines have conceived of the machine as a creative partner, enabling a feedback loop between human and machine. In this view, "Where humans and machines are viewed as sharing patterns of organization, there can emerge the 'posthuman,' the splicing of human with the machine" (Hugunin 2000). Formeta:3D operates in this vein, machine and human acting both as observer and as participant in an iterative, non-deterministic process of form generation.

Precedents

One early project that sought to explore recursion or feedback within the CAM process was a project called "Shorting the Automation Circuit," developed by the sixteen*(makers) group in 2000. They created an instrument that would collect environmental data—light detected as the instrument was moved

2

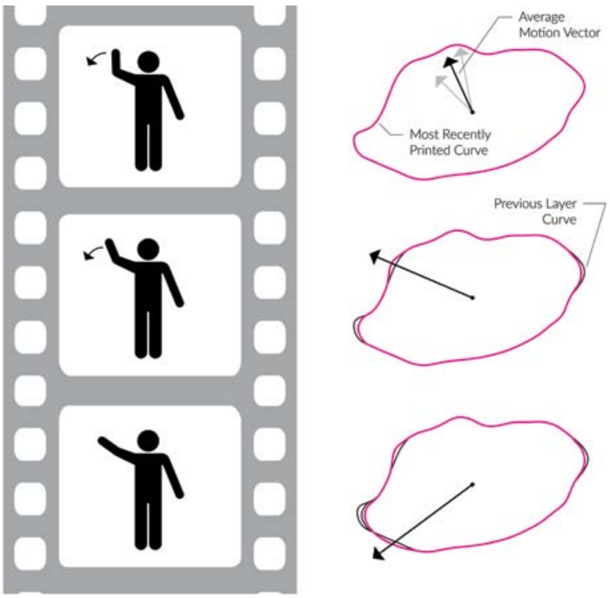


3

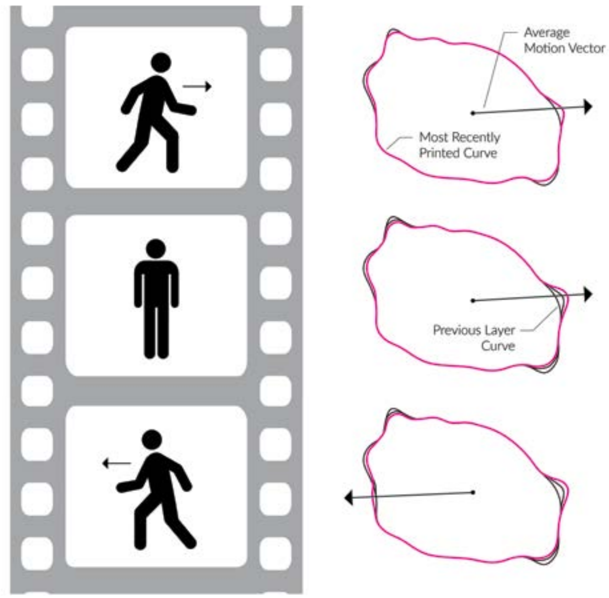
by wind currents (or as people passed by)—as well as a script that would translate this data into three-dimensional form. However, since this project was developed prior to the popular rise of 3D printing over the past decade, the group did not have direct access to a 3D printer. As a result, their system generated snapshots in time that did not reflect a real-time response to the forms as they were manufactured. The group speculated about the potential for an active feedback loop but did not have an opportunity to realize it or explore its ramifications (Ayres 2006).

Drawing machines such as the Artsbots of Leonel Moura have investigated other aspects of the machine, including artificial intelligence and swarm behaviors (Moura 2014). Moura's work frames the machine as an autonomous agent, dependent on human input primarily for its initial operating rules. While these rules may include responses to sensor input, he is more interested in the autonomous machine "taking the human out of the loop," than one that engages in a dialogue with humans.

Researchers at SCI-Arc have developed an advanced system for real-time human interaction and interfacing with the physical through robotics. Recent work is focused on bridging the digital and the physical in service of design through gesture recognition and environmental awareness enabled by 3D scanning. Live, the platform developed at SCI-Arc for real-time connectivity with their industrial robot arms, together with the various experiments developed on this platform, explore design ideation, blurred virtual and spatial communication, dynamic environments, tooling development, and autonomous decision-making in relation to a physical workpiece (Batliner, Newsum, and Rehm 2015). Many of the technologies employed and the associated toolchains are similar to those employed by Formeta:3D (in particular the Eyerobot project), and in fact many of the explorations could be framed similarly in the way that they engage the roles of the designer, the machine, and the user. Even so, Formeta:3D approaches real-time human-machine interaction from a different perspective, examining how formal and physical artifacts of human-machine interaction can passively record and yet actively influence human experiences and populate an evolving posthuman landscape.



4 Response of Formeta:3D motion analysis to ongoing movement of human participant.



5 When a human participant stops, the algorithm retains the most recent motion vector until new motion is detected. This results in greater emphasis being placed on the earlier motion.

Previous Work

The machine presented in this paper is informed by and is an evolution of previously published work begun by the authors in two dimensions with the simple pen-on-paper drawing machine shown in Figure 2 (Wit, Eisinger, and Putt 2016). This drawing machine employed two LEGO Mindstorms motors positioned at the upper corners of a vertical drawing plane, controlling the movement of a marker using monofilament spooled and unspooled by the motors. The instructions for the motors were provided by a Raspberry Pi single-board computer, which generated drawing geometry in real-time based on image data collected via an attached camera. Multiple algorithms were developed as options for translating color data from the collected images to XY coordinates. The resulting drawings reflected the response to the predominant colors observed as well as to the extremes when intense colors were detected (Figure 3).

This earlier Formeta investigated how the machine can serve as a translator of inputs of one kind to outputs of another. It questioned the location of design authorship in relation to the machine, given that the drawings produced by the machine were influenced by a multitude of factors—including the human activity in the space, the lighting conditions, the physical imperfections of the machine, and the variety of technologies employed—in addition to the way in which the system components were assembled and coded by the designers. As with the drawing machines described by Hugunin, this machine was a creative partner, participating in the outcomes as much as the designers and those interacting with it.

METHODS

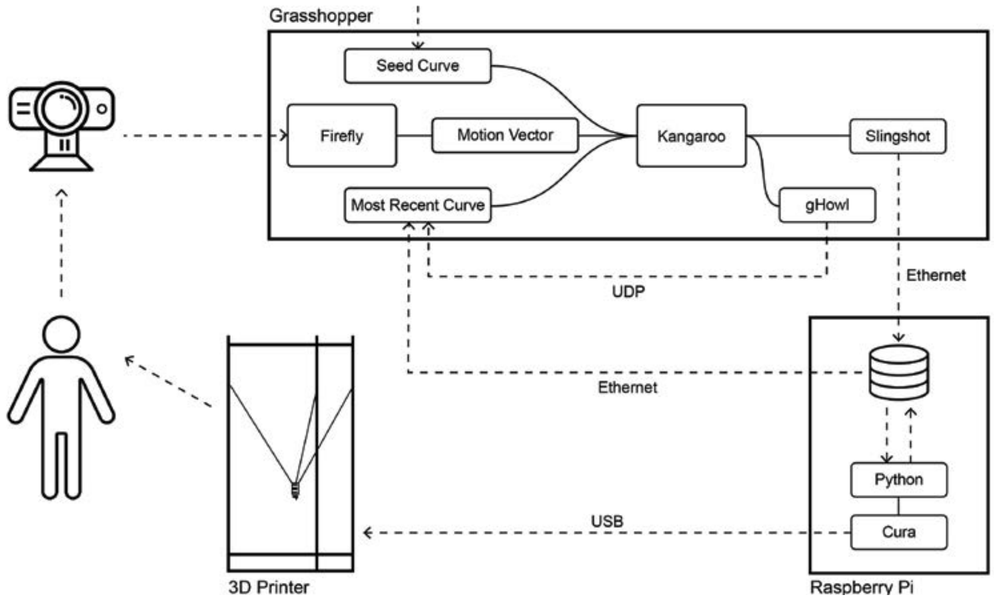
The realization of Formeta:3D required the development of a many-layered, low-latency toolchain from camera to printer as well as extensive mechanical and material calibration. This section discusses the main components and processes of the system as well as the final experiential setup.

Platform

The computational platform controlling Formeta:3D was grounded in Grasshopper, but included additional components located on intermediate and control devices. A Grasshopper definition performed the bulk of data collection and processing, relying on Firefly to obtain and pre-process the video feed and on Kangaroo to simulate forces acting on the geometry. Communication plugins, including gHowl and Slingshot, enabled recursion and the transmission of data to the intermediate control device, a Raspberry Pi. Python code on the Raspberry Pi processed coordinate data from Grasshopper and interfaced with the 3D printer, a kit-built, delta-style RepRap descendant with an Arduino/RAMPS controller and Marlin firmware, a Bowden extruder, an E3D hot end, and a heated bed.

Video Processing

The Firefly video stream feeding the Grasshopper definition for this project came from a standard HD webcam trained on the area in front of the 3D printer in order to facilitate interaction with the system and its product. Motion in that area was translated to 2D motion vectors produced by Firefly's color averaging component. These vectors, located at the XY plane origin in



6 Overview of Formeta:3D platform and data flow. Human interaction influences the form produced by the 3D printer, which in turn influences the response of the human. Human and machine are both participants as the machine records the interaction on form.

Rhino, were smoothed, filtered, and averaged to produce a single primary vector (Figure 4), as well as to determine the location of an attractor point, which in turn influenced simulated forces acting on the evolving form.

Data Throughput and Recursion

Due to Firefly's ability to process the incoming video stream very quickly, it became necessary to throttle the output of the Grasshopper script to prevent slower downstream processes from being overwhelmed. This was accomplished by tracking a limited history of data for smoothing purposes and only allowing new geometry to be emitted every two seconds.

The data throttling also served to limit the speed of recursion in the definition, which was its own challenge. Grasshopper is not designed for recursion and in fact a workaround was required to circumvent Grasshopper's loop detection. The Slingshot and gHowl plugins eventually solved the problem (where Hoopsnake and others did not), enabling curve serialization and transmission through UDP network datagrams to the beginning of the definition.

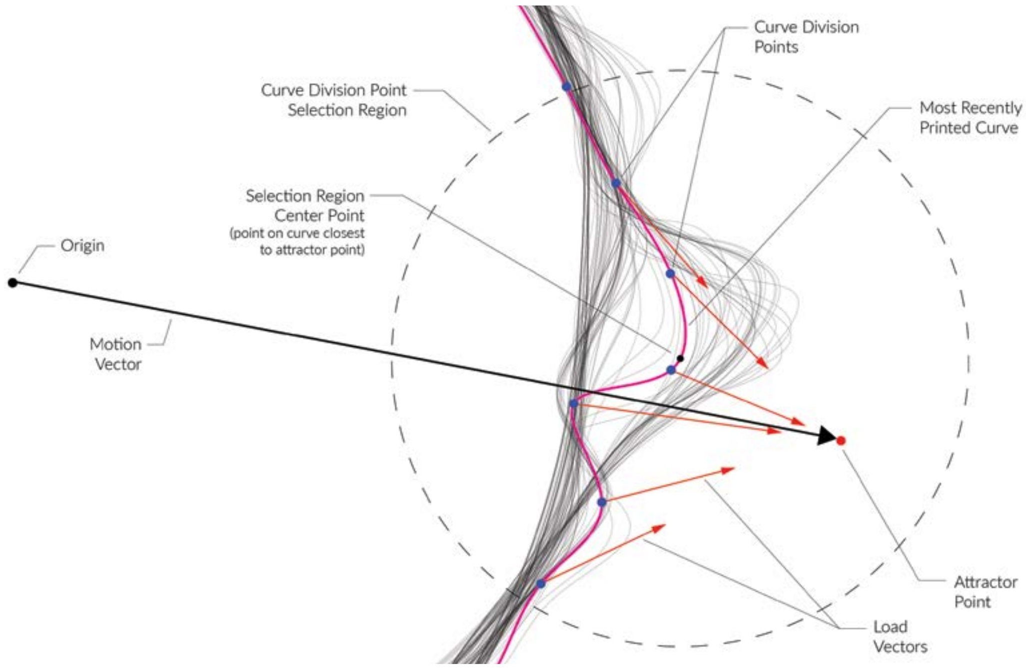
Later versions of the definition used Slingshot exclusively, sending curve data to a MySQL database on the Raspberry Pi and retrieving that data at the beginning of the definition. This enabled the Python code to indicate to Grasshopper the success or failure of printing the received geometry, creating a closed loop process that reflected the actually printed layers in the recursion (Figure 6).

Geometric Constraints and Simulation of Forces

In order to facilitate the creation of a printable form that could express the effects of interaction over time, the generated toolpath geometry was based on a circular primitive, which would be pushed and pulled based on the location of the previously generated attractor point. While precision was not a goal, constraints were necessary to allow for the evolution of form in a fashion that would enable continued printing, providing a sufficient amount of material and structure to support each new layer.

The geometric constraints employed included limits on the toolpath deviation allowed from layer to layer and limits on the effects of simulated forces on the geometry. To accomplish this, the magnitudes of the motion vectors were mapped to a limited donut-shaped region around the previous layer curve. Further constraints were implemented as forces resisting change—Kangaroo "goals" set to pull curve division points to the previous layer curve and to minimize the change in distance between division points.

The forces compelling change to the geometry were also set up as Kangaroo goals. The primary goals affecting change were set up as loads pulling curve division points near the attractor point inward or outward on a gradient based on motion vector magnitude and distance from the attractor point. The effect of this configuration was similar to the capability of some software packages (particularly Maya) to allow "soft selection," where geometric operations on a given element are translated to surrounding elements on a gradient (Figure 7).



7 Motion vector acting on toolpath. Motion detected by the webcam is translated to a vector emanating from the origin in virtual 3D space, which determines the position of an attractor point. Kangaroo forces pulling on curve division points in the vicinity pull the 3D print toolpath curve toward the attractor point to generate the next layer.

Interface with 3D Printer and G-Code Generation

The Marlin firmware controlling the 3D printer is a standard one for open source 3D printers, but its communication capabilities are relatively primitive. It uses a basic serial connection without built-in error-checking for USB communication, requiring a fair amount of code on the client side to ensure successful data transmission and synchronization. Rather than reinvent the wheel for Formeta:3D, open source Python code from the Cura project was leveraged to handle direct communication with the printer.

Unfortunately, using Cura's slicing logic for G-Code generation was not practical for this project. Cura's slicer is located in a separate module written in C code, which would have taken a significant amount of time to appropriate for the sake of this project. An alternative approach considered was generating 3D model files of each print layer for external slicing, but this proved too problematic in terms of performance and post-processing. In the end, custom G-Code generation routines were employed, even though this meant foregoing the benefits of the advanced acceleration and extrusion algorithms embedded in the slicer.

Physical Considerations and Calibration

As a first foray into the intricacies of 3D printing, this project required a great deal of learning related to the materials in use and their response to the controls provided by the printer. The single greatest challenge was in calibrating the extrusion rate of the PLA filament to ensure consistent printing. Initial tests attempted to push far more filament through than the hot end or the extruder could handle, resulting in inconsistent extrusion and filament jams. Other factors that required calibration over many iterations included the hot end temperature, print speed, and bed leveling.

Interactive Installation

Once adequately calibrated, it was possible to set up the Formeta:3D project in a public atrium space to allow the activity in that space to inform the generated form. The installation consisted of the 3D printer with a Raspberry Pi intermediate controller, a computer running the Grasshopper definition, and a webcam mounted on a six-foot-long dowel attached to the printer to give it a wider vantage point for tracking movement. The computer's monitor was mounted in such a way to allow people to see the video stream being captured, as well as the Rhino interface showing the generated geometry and the changing motion vectors being processed. The installation was active for hours at a time over the course of three days.

RESULTS AND REFLECTION

Realized over the course of a semester, Formeta:3D was completed in time for installation prior to Ball State University students leaving for the summer. The artifacts produced by the installation represent features of a new landscape shaped by machines for and in response to humans. No longer just a tool, the machine affects the (local) course of history with its own interpretation of events expressed in physical form.

Physical Outcomes

The print pictured in Figure 8 shows the longest running print of the public installation of Formeta:3D. It exhibits a number of features that exemplify the current behavior of the form generation logic, including how motion is expressed, how transitions between layers are handled, and various material effects resulting from how these elements of the algorithm were handled.

Motion is expressed on the pictured form in the wavy character of the walls as a result of the way in which the motion vectors pushed and pulled on the layer geometry. The changes in form that occur over many layers show times when the captured motion trended in a given direction. The moments where protrusions spilled over into loops of filament are due to the inability of underlying layers to support the magnitude of the diversion from the preceding layer geometry.

The moments where spillover happens also indicate a counterintuitive outcome of the form generation algorithm as it is currently implemented, since they represent the times where the least activity was taking place. The forces simulated in the Grasshopper definition do not have a significant cumulative effect unless those forces are continuously expressed in the same general direction. Since the definition currently retains the most recent motion vector until a new vector is captured (one that differs significantly enough from the last one to overcome the smoothing process), the result is that a lack of new activity translates to the greatest cumulative effect on the geometry (see diagram, Figure 5).

Transitions between layers are handled differently in different variants of the form generation algorithm. In Figure 8, the layer transition points are indicated by little balls of melted filament that accumulated where the print head paused while it waited to receive the next layer (see also Figure 1). At the moment, the location of those points is dictated only by the start and end points of the layer geometry as generated by Kangaroo. This represents an opportunity for further development, since the location of those points could be controlled by another input to provide an additional means for indexing the activity in the space.

Another approach for handling layer transitions is pictured in Figure 9, where a series of gaps are evident. In this approach, the filament and the print head were both retracted to avoid the seepage of melted plastic. This approach also kept the nozzle from sitting immersed in melted plastic, which can lead to burnt filament and a clogged nozzle. Burnt filament is evidenced at multiple points in Figure 8 in the form of dark brown streaks.

8

9

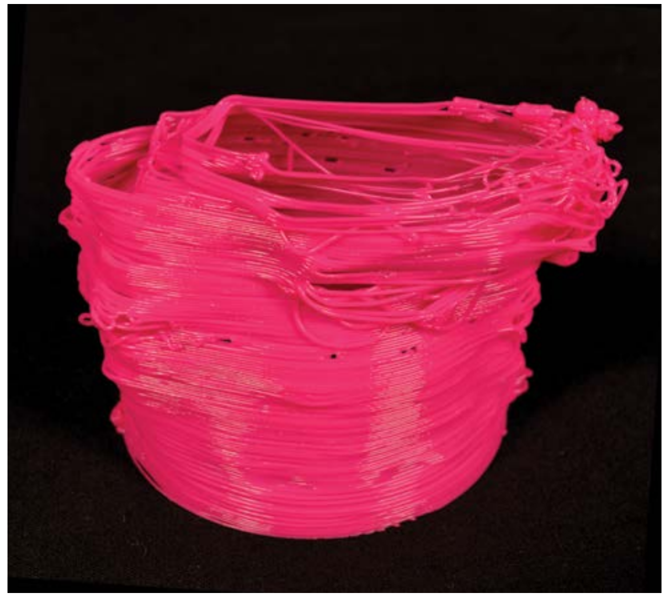
8 Extended print result. Balls of plastic on the surface indicate where the 3D printer hot end paused while waiting for the next 3D print layer to be generated.

9 An additional print result produced using an alternate layer transition approach that involved retracting the plastic filament while waiting for the next layer.

Formal Outcomes

The types of forms produced by the form generation algorithm as currently implemented are limited. As noted in the previous section, the wavy areas indicate varying motion expressed on form, but in general the form remains cylindrical without any major deviations. This is due in large part to the way in which the simulated forces behave as the geometry distorts itself to greater degrees. Beyond a certain point, Kangaroo gets overwhelmed, and the new toolpath layers lose all connection to the previous layers, gathering in a tangled mess to one side of the form and causing the printer to produce globs of material on that side (Figures 10–11).

10 Rendering of toolpaths generated by Formeta:3D that produced the print result depicted in Figure 11.



11 3D print resulting from toolpaths shown in Figure 10. The globs of plastic on the top right side are the result of the Kangaroo simulation being overwhelmed by conflicting forces.

Due to this behavior, while each form produced is unique and represents a unique set of interactions at a specific time and place, at a distance the main differentiators between individual artifacts are the height of each print and/or the result of varying the seed geometry. On close inspection, the nature of the layer transitions or the presence of pronounced protrusions also differentiates each piece.

Beyond the programmed behavior of the machine, there are unexpected outcomes that differentiate outcomes and connect the results to the space by nature of the unexpectedness of the behavior. For example, a layer may fail to adhere to the previous one due to an imperfection in the machine or in the toolchain, resulting in bridges and other formations that make the printed form significantly different from its digital counterpart (Figures 10–11). Observing this happen during a print marks the experience due to its surprising behavior.

Human-Machine Interaction and Feedback

The public installation of Formeta:3D was successful at both recording interaction on and prompting interaction with the form being generated. As noted above, the machine produced unique prints, reflecting activity captured from its surroundings at a specific place and period of time. At a basic level, the machine was simply an historian, jotting down its interpretation of what it observed. But the fact that it was doing so in the moment, in full view of those around it, and in three-dimensional form created a stimulating feedback loop that led to activity that would not otherwise have taken place.

The initial draw toward interaction with the installation was the machine itself. The delta-style 3D printer does not have the familiar box-like shape of more common 3D printers like Makerbots, which prompted discussion about its function. Once aware of both the capabilities of the machine and its purpose in responding to activity in the space, people took notice of the camera and of themselves in the video feed displayed on the accompanying screen. The real point of understanding for those interacting with the machine was when they connected their own movement to the vector display on the screen, which provided immediate, highly responsive feedback. Then, they would begin to purposefully attempt to control the more subtle behavior of the machine with hand and body motions (Figure 12). The display of previously printed artifacts from earlier interactions allowed for discussion of individual features and speculation about what caused them.

CONCLUSION

Formeta:3D presents a methodology and a toolchain for implementing real-time interaction with additive physical form via digital inputs. While currently limited by the Grasshopper-based implementation of its core elements, this toolchain could be ported to a more flexible platform that would allow for more advanced decision making and machine learning, for example to amplify trends in form development for a given space. There is also the potential for more responsiveness in the system, since the current layer-by-layer approach for realizing form lacks the immediacy provided by machines, including the earlier Formeta:2D drawing machine and the SCI-Arc robot arms.

12 The Formeta:3D interactive installation as set up in the Ball State University College of Architecture and Planning main atrium space.

Additionally, while the creation of an occupiable space was never a goal for this research, a larger scale implementation would more than likely stimulate a greater degree of interaction with the machine and might more clearly express formal and spatial variation. Another iteration might focus less on stimulating interaction and observe human activity in a wider area, perhaps using facial recognition to generate multiple simultaneous motion vectors with differing origins to influence form generation.

The implementation of Formeta:3D adds a new element to the tradition of drawing machines by bringing a digital fabrication tool previously only used in non-real-time applications into the lexicon of real-time output tools. While real-time drawing machines that produce 3D elements are not new (Hugunin 2000), Formeta:3D realizes a feedback loop that operates in the digital, the physical, and the experiential realms simultaneously. This fact situates this project uniquely in relation to the work of Moura and SCI-Arc. Where Moura eschews human influence, Formeta:3D engages both autonomous form generation and human interaction. Where SCI-Arc's spatial explorations with the Eyerobot project are constantly in flux in relation to human interaction, Formeta:3D evolves a lasting artifact that records human experience additively. This situates the machine as historian, its products akin to pictorial histories, allowing for varied interpretations, conveyed from observer to observer. Here, this project embodies the posthuman, a machine recording human history autonomously while being inextricably tied to the involvement of human interaction.

ACKNOWLEDGEMENTS

Many thanks are due to the College of Architecture and Planning (CAP) at Ball State University, for the use of facilities for the Formeta:3D installation, and to CAP Interim Dean Phil Repp, for funding the purchase of the 3D printer for this project. The referenced previous work Formeta:2D was developed by Daniel Eisinger and Steven Putt with advising from Ball State faculty members Andrew John Wit and Joshua Coggeshall.

REFERENCES

Ayres, Phil. 2006. "Constructing the Specific." In *The Architecture Co-Laboratory: GameSetandMatch II : On Computer Games, Advanced Geometries, and Digital Technologies*, edited by Kas Oosterhuis and Lukas. Feireiss. Rotterdam: Episode Publishers. 314–321.

Batliner, Curime, Michael Jake Newsum, and Casey M. Rehm. 2015. "Live: Synchronous Computing in Robot Driven Design." In *Real Time: Proceedings of the 33rd eCAADe Conference*, vol. 2, edited by B. Martens, G. Wurzer, T. Grasl, W. E. Lorenz, and R. Schaffranek. Vienna, Austria: eCAADe. 277–286.

Hugunin, James. 2000. "in::FORMATION: The Aesthetic Use of Machinic Beings." *Leonardo* 33 (4): 249–261.

Moura, Leonel, and Henrique Garcia Pereira. 2014. "A New Kind of Art [Based on Autonomous Collective Robotics]." *Footprint* 15: 25–32.

Wit, Andrew, Daniel Eisinger, and Steven Putt. 2016. "Human Interaction-Oriented Robotic Form Generation." In *Robot 2015: Second Iberian Robotics Conference: Advances in Robotics*, vol. 2, edited by L. P. Reis, A. P. Moreira, P. U. Lima, L. Montano, and V. Muñoz-Martinez. Switzerland: Springer Verlag. 353–364.

IMAGE CREDITS

All images and diagrams produced by Daniel Eisinger. Figures 4, 5 and 6 include icons from the Noun Project created by Bradley Avison, Hoach Le Dinh, Irene Hoffman, Lastspark, Arthur Shlain, and Fernando Vasconcelos.

Daniel Eisinger is an instructor at the College of Architecture and Planning (CAP) at Ball State University (Indiana), where he also manages the college's digital fabrication and robotics labs. Eisinger holds a Bachelor of Science degree in Computer Science from Taylor University and spent a number of years in software development before obtaining his Master of Architecture degree from Ball State University.

Steven Putt is a Graduate Architect at Design Collaborative in Fort Wayne, Indiana, where he specializes in digital design and fabrication. As co-founder of the in-house research entity, dc.lab, Putt focuses on implementing computational and parametric design methods in day-to-day architectural practice. Putt holds a Master of Architecture degree from Ball State University.