The art of bots: A practice-based study of the multiplicity, entanglements and figuration of sociocomputational assemblages

Matthew Plummer-Fernandez
33365961

Thesis submitted for the degree of Doctor of Philosophy

Design Department
Goldsmiths, University of London

December 2018

Declaration of Authorship: I, Matthew Plummer-Fernandez, hereby declare that this thesis and the work presented in it is entirely my own. Where I have consulted the work of others, this is always clearly stated.

Signed:  Date: 31/12/2018

DIGITAL ARCHIVE OF PRACTICE-BASED RESEARCH 6

ABSTRACT 8

LIST OF ILLUSTRATIONS 9

CHAPTER 1: INTRODUCTION 13

A brief history of bots 14

The emergence of artbots 20

Prior notions of computer-related art 27

Note on interdisciplinarity 31

Thesis outline 32

CHAPTER 2: A FRAMEWORK FOR UNDERSTANDING ARTBOTS 34

What is an algorithm? 35

Conceptualising composite arrangements 40

Software architecture 42

Sociocomputational assemblages 47

Artbots 50

Towards a framework 53

Multiplicity 56

Entanglement 58

Figuration 60

Conclusion 62

CHAPTER 3: STUDYING SOCIOCOMPUTATIONAL ASSEMBLAGES 64

Research setting 64

Participant research 65

Inventive and practice-based methods 68

Research through collaboration 70

Research through curation 72

Methodological issues 74

Conclusion 77

CHAPTER 4: MULTIPLICITY 78

Making multiplicities 79

Scalability and duration 84

Big Data Pawn Shop 93

Conclusion 98

CHAPTER 5: ENTANGLEMENT 100

User interaction 101

Interactive bots 102

Entanglements at *Art of Bots* 104

Entanglement 112

Every User 123

Shiv Integer 127

Conclusion 134

CHAPTER 6: FIGURATION 136

Familiar figurations 137

Figuring alternatives 142

Magic Realism Bot 151

MAMMON 154

The Revenant 159

Petita Tatata 162

Novice Art Blogger 166

Conclusion 175

CHAPTER 7: CONCLUSION 177

Thesis summary 177

Understandings of Web-based software 179

Practice through research 181

Methodological contributions 183

From artbots to configurative art 185

Further recommendations 187

REFERENCES 189

APPENDIX I: PRACTICE-BASED WORK 197

I.I Artbot URLs 197

I.II Exhibitions 198

I.III Exhibitions marketing material 199

I.IV Press 204

APPENDIX II. *ART OF BOTS* 205

II.I Further documentation 205

II.II Exhibition plan 209

II.III Marketing material 210

II.IV Press coverage 213

II.V Interviews 216

Digital archive of practice-based research

The hard-drive contents submitted with this thesis are also available on Dropbox:
https://www.dropbox.com/sh/1ignk5s9kelyj07/AADR4G4QXvxOox5PbMt13JJFa?dl=0
Shortened URL: *http://tiny.cc/MPFbots*

Contents

1. *Art of Bots*
    - 1.1 Images
    - 1.2 Press

2. *All Eyes*
    - 2.1 Eyes
    - 2.2 Source code
    - 2.3 Tweet archive
    - 2.4 Twitter data
    - 2.5 Videos

3. *Every User*
    - 3.1 Images
    - 3.2 Source code
    - 3.3 Tweet archive
    - 3.4 Twitter data
    - 3.5 Users

4. *Novice Art Blogger*
    - 4.1 Images
    - 4.2 Novice Art Blogger HTML
    - 4.3 NAB posts
    - 4.4 Source code

5. *Shiv Integer*
    - 5.1 3D Files
    - 5.2 3D Prints
    - 5.3 Emails and comments
    - 5.4 Images

5.5 Source Code

5.6 Videos

6. *Revenant Bot*

6.1 Images

6.2 Source Code

7. *Petita Tatata*

7.1 Images

7.2 Poems audio

7.3 Source code

Abstract

This thesis examines and analyses an emerging art practice known as *artbots*. Artbots are internet-based software applications that are imbued with character and configured to engage and entertain online audiences. This form of practice, and the community of practice leading it, was found to be underrepresented and misunderstood. I argue that this artform is original and warrants a more thorough understanding. This thesis develops a conceptual framework for understanding artbots that focuses on and enables questioning around pertinent aspects of the practice. A wide range of literature was reviewed to provide theoretical underpinnings towards this framework, including literature on algorithm studies, science and technology studies, and software architecture. The devised framework examines artbot case studies through the notions of *multiplicity*, *entanglement*, and *figuration*, having understood artbots as heterogenous *sociocomputational assemblages* comprised of software components and human intra-activity. The research followed a varied methodology that encompassed participant observation and my own practice-based experiments in producing artbots. The study resulted in several original works. In addition, a showcase titled *Art of Bots* brought together key proponents and artbots, further providing material that is analysed in this thesis. The study helped identify and discuss artbots with attention to how they utilise modular software components in novel arrangements, how normative human and nonhuman relations of interaction are being eschewed in favour of entangled interrelations, and how artbots challenge common narratives dictating technological constructs by inventing unique characters and figurations.

List of illustrations

Figure 1. Computer architecture diagram for *Storyblocks*, a Web-based service for searching for licensable 'stock' photographs. 45

Figure 2. *Earth Rover Bot* (2014) by Colin Mitchell, at the location in Russia where changes to the Twitter API caused the configuration to cease working. *https://twitter.com/EarthRoverBot.* 80

Figure 3. *Museum Bot* (2014) by Darius Kazemi. *http://twitter.com/museumbot*. 81

Figure 4. *Reverse OCR* producing a scribble that reads "goodness". *http://reverseocr.tumblr.com/image/166751774363* 83

Figure 5. *Cloud OCR* (2015) by Everest Pipkin. *http://everest-pipkin.com/* 83

Figure 6. The Raspberry Pi computer with Wi-Fi dongle that hosted the artbots made during this study. 86

Figure 7. An eye produced by *All Eyes* (2014). *https://twitter.com/_all_eyes* 86

Figure 8. A screenshot of *Everyword* (2007) by Allison Parrish. *https://twitter.com/everyword* 87

Figure 9. *Everyword* producing the word "sex", its most popular word by June 2014. 88

Figure 10. *Every Icon*, 1996 by John F. Simon Jr. The wall hanging version was comprised of the software running on a Macintosh PowerBook170 encased in plastic acrylic. 89

Figure 11. Tweets from *Everyword* uttering the words "bush", "did", "nine", and "eleven" were appropriated by Twitter users as a means to spread the message "Bush did 9/11". Screenshot taken from *https://twitter.com/thetrashknight/status/578538466407002112*. 89

Figure 12. *The Ephemerides* (2015) by Allison Parrish. 91

Figure 13. *The Ephemerides* was reconfigured as a chapbook for *Art of Bots* (2016). 92

Figure 14. A cropped screenshot of http://twitter.com/unchartedatlas 92

Figure 15. Screenshot of *https://twitter.com/cspan5* by Sam Lavigne. 94

Figure 16. Presentation of *CSPAN-5* and *Parliament-Live* at Art of Bots, April 2016. 95

Figure 17. A screenshot of *http://bigdatapawnshop.lav.io*. 96

Figure 18. The HTML inspector reveals the source image that is superimposed onto the Zazzle product. When the Web shop visitor clicks on a codename, speculative products are generated 'on-the-fly', by fetching a random selection of Zazzle products and replacing their design with that of the leaked NSA document, simply by substituting the source image in the URL. 97

Figure 19. Big Data Pawn Shop at Art of Bots, 2016 97

Figure 20. Zazzle marketplace items offered by Supernova23 _____ 98
Figure 21. *http://twitter.com/SortingBot* by Darius Kazemi._____ 104
Figure 22. User-audience interface for *5Point9Billion*, presented at Art of Bots.____ 105
Figure 23. The *5Point9Billion* floor-based star map, presented at Art of Bots._____ 105
Figure 24. Screenshot (cropped) of *https://twitter.com/GraphicScoreBot*. *Graphic Score Bot* (2016) by Emma Winston. _____ 108
Figure 25. *Treatise* (1967) by Cornelius Cardew was a compilation of graphic scores drawn by the avant-garde composer (Cardew 1967). _____ 108
Figure 26. Emma Winston performing to *Graphic Score Bot* at Art of Bots. _____ 109
Figure 27. Screenshot of *https://twitter.com/artassignbot* by Jeff Thompson, demonstrating a completed artwork from *Bot Art School*._____ 110
Figure 28. *Bot Art School* at Art of Bots. Jeff Thompson (second from left) hands an art assignment to a new assignee. _____ 111
Figure 29. Visitors making bots using *Cheap Bots Done Quick* at Art of Bots, 2016_ 112
Figure 30. Screenshot of *https://twitter.com/pentametron*, by Ranjit Bhatnagar. ____ 113
Figure 31. *http://twitter.com/empathydeck* by Erica Scourti. _____ 115
Figure 32. Oliver the dog by the window, with the microphone and Raspberry Pi in front; all components of a heterogeneous software architecture that manifests online as *@oliverbarkbark*._____ 116
Figure 33. *http://twitter.com/oliverbarkbark* by Henry Conklin. _____ 116
Figure 34. A screenshot of *http://twitter.com/oliviataters*, by Rob Dubbin. _____ 118
Figure 35. A fragment from *Generated Detective* (2014) by Greg Borenstein. The full comic is available at *http://gregborenstein.com/comics/generated_detective/1/*. 119
Figure 36. *Five Years of Captured Captchas* (2017), by Sebastien Schmieg and Silvio Lorusso. _____ 120
Figure 37. An example of a Google *ReCaptcha* used to train software to distinguish Mark Hamill from Slavoj Zizek. _____ 121
Figure 38. Final image state of *https://www.reddit.com/r/place/*. _____ 122
Figure 39. A screenshot of *https://twitter.com/every_user_*. Not all numbers followed succession by an increment of one, as some accounts may had been deleted. Some sections of the database also incremented ID numbers by ten rather than by one. 124
Figure 40. *Every User*'s first three tweets echoing the first three users to join the service: its founders. _____ 125
Figure 41. The notifications panel for *Every User*, showing reactions from mentioned users. _____ 126
Figure 42. Extract from *Every User*'s source code, showing functions "searchUser" and "tweetStatus"._____ 126

Figure 43. A screenshot of *http://www.thingiverse.com/shivinteger/designs*. _____ 128
Figure 44. Thingiverse comment left by Brightlights, for the object titled *Master by a Dual adapter*: *https://www.thingiverse.com/thing:1424612/comments*. _____ 131
Figure 45. *Shiv Integer* exhibition at iMal, Brussels, 2017, during Art Brussels. The 3D prints were suspended inside a hanging tunnel. Other exhibits such as projections and wall-prints were also presented in the exhibition space. _____ 132
Figure 46. *Shiv Integer* exhibition at iMal, Brussels, 2017, during Art Brussels. A collection of 3D Files printed and suspended within a display tunnel. _____ 132
Figure 47. *Shiv Integer* creation titled *Flexible Door 20 x 2*, licensed as 'Creative Commons - Attribution - Share Alike' and found at *https://www.thingiverse.com/thing:1666865*. _____ 133
Figure 48. User comments made into wall-pasted posters. Many of the posters depict fragments of an art-spam debate between Thingiverse users – hence, some posters simply read "SPAM". _____ 133
Figure 49. Ending credits video projection at the *Shiv Integer* exhibition. _____ 134
Figure 50. Karel Čapek's 1920 science fiction play, *R.U.R.* _____ 138
Figure 51. A screenshot depicting the description for "Molly", an algorithmic website-design software tool created by *Grid.io*. _____ 139
Figure 52. Advertisement for Amazon's Alexa, the virtual assistant (software architecture) that is interfaced via the Amazon Echo microphone-speaker device. _____ 139
Figure 53. Cortana in Microsoft's Halo 2 videogame, 2004 _____ 140
Figure 54. Cortana in Skype, 2018 (Skype was acquired by Microsoft). Image sourced from *https://twitter.com/linguangst/status/1069742586460598273*. _____ 141
Figure 55. Billboards articulating the fabled persona of 'The Algorithm', by advertising agency Crispin Porter Bogusky, for ask.com 2007. _____ 142
Figure 56. *Tiny Star Fields* (2014) by Everest Pipkin. *https://twitter.com/tiny_star_field* _____ 144
Figure 57. *Lost Buoy* (2013) by Mark Sample. *https://twitter.com/_LostBuoy_* _____ 145
Figure 58. A screenshot of *http://twitter.com/losttesla* (2016) by Kate Compton. ___ 146
Figure 58. Screenshot of Treasure Column video: *https://www.youtube.com/watch?v=bvNrGZoH2XQ* _____ 148
Figure 60. An example output of *TV Helper* by David Lublin. _____ 149
Figure 61. *Magic Realism Bot* (2015) by Chris and Ali Rodley. *https://twitter.com/MagicRealismBot* _____ 151
Figure 62. Installation of *Magic Realism Bot* by Chris and Ali Rodley at Art of Bots, 2016. _____ 153

Figure 63. *MAMMON* by Shardcore at the Art of Bots Showcase, 2016. _____ 155
Figure 64. Screenshot of *http://twitter.com/bffbot2* ('twitter.com/bbfbot1' was suspended by Twitter, so a new account was created) _____ 157
Figure 65. Screenshot of http://twitter.com/trippingbot_____ 158
Figure 66. Screenshot of @trippingbot's Twitter timeline. _____ 159
Figure 67. Screenshot of https://twitter.com/Revenantbot_____ 160
Figure 68. A screenshot of *soundcloud.com/petita_tatata*. _____ 163
Figure 69. Bing image search for "Kawaii". _____ 164
Figure 70. Screenshot of *http://soundcloud.com/petita-dumdum* (2016) by Memo Akten. _____ 165
Figure 71. A screenshot of *http://noviceartblogger.tumblr.com*._____ 167
Figure 72. A screenshot of *http://deeplearning.cs.toronto.edu/i2t*. In this screenshot, I capture how I would manually use the Toronto research demo, and here, test its response to a painting by Picasso. _____ 168
Figure 73. Extract from NAB's source code showing the rewording of captions. ___ 169
Figure 74. *Death and the Conquistador* by Aubrey Williams, 1959, side by side with a pizza decorated to look like an 'angry bird'. _____ 170
Figure 75. *Novice Art Blogger* presented *Algorithmic Rubbish: Daring to Defy Misfortune* (2015) at the Stedelijk Museum of Contemporary Art in Amsterdam. _____ 172
Figure 76. An entry from *http://noviceartblogger-stedelijk.tumblr.com*, the Stedelijk version of *Novice Art Blogger*. _____ 172
Figure 77. *Novice Art Blogger* was invited to interpret the front cover of *Dazed and Confused*, Winter 2014. _____ 173

Chapter 1: Introduction

> "A scholar imagines a puppy that can destroy the universe,
> and becomes obsessed with finding it." – Magic Realism Bot, 3 November 2016[1]

A new community of creative practice has emerged in the last decade with an interest in making *artbots*. Artbot practitioners creatively explore internet-based arrangements of software that are imbued with character. Artbots are active on social media platforms such as Twitter, and some have become hugely popular, reaching online audiences of hundreds of thousands and running for years. These unique works contribute to contemporary approaches to art, and yet remain little discussed and documented. Moreover, these works tend to be poorly understood, as no adequate frameworks for understanding this type of practice have been proposed – presently, there is not even a name for it.

Artbots are not only important as an artform in their own right, but also open a wider understanding and debate about software automation on the internet. Bots on social media have become a contentious issue as swarms of them have been identified as having manipulated public opinions and political debates (Hern 2017). Furthermore, social media platforms are increasingly being scrutinised and critiqued for their role in facilitating this manipulation (Solon 2017), as well as for their own contentious uses of computer-controlled interactions with people (Bucher 2017a). Additionally, as computational technologies advance, more of these software applications are characterised as exhibiting *artificial intelligence* (AI). AI devices are becoming increasingly ubiquitous, appearing in smartphones, vehicles, and homes, and are often characterised as assistants. Given these circumstances, the relevance of artbots is clear. Artbot practitioners enable artistic and audience engagement with matters of bots, automation, platforms, and the characterisation of software. Moreover, a framework for understanding artbots produces a toolset for discussing and understanding these matters more broadly.

This dissertation develops a framework for understanding artbot practices and applies it to the examination and analysis of artbot works. Developing this framework involves

[1] See *https://twitter.com/MagicRealismBot/status/794059551364423680*.

identifying and applying theoretical concepts from philosophy, sociology (notably science technology and society studies, or STS), and feminist studies towards an understanding of *configurations* of computational and social elements in interrelation. Having arrived at the view that bots can be understood as *sociocomputational assemblages*, a research methodology designed to study these complex arrangements and the community of practice around artbots is described. An important component of this methodology involves novel experiments with artbots, a form of research through design (Frayling 1993) that has resulted in various original works created as part of the study. This practice-based approach has not only contributed to an empirical knowledge of making bots, but has also enabled participation and collaboration with the community of practice in study.

This introduction discusses some of the background and context of this thesis. First, a concise history of bots is provided, which leads into an account of the emergence of contemporary artbot practices. Following this, other possible framings of computer-related practice are discussed in relation to artbot practices, further demonstrating the need for a new approach to understanding these new practices. Finally, I provide a summary of how the thesis is structured.

A brief history of bots

The artbots of the last decade are more easily understood within a context of what 'bots' are more generally, along with the history that precedes contemporary artistic interest in them. This concise account describes how bots became a way of characterising software that runs over the internet, and how there has been continued interest and development in programming bots to engage with people over the internet. Understanding this evolution of bots helps to outline and contextualise the field of study, revealing that the uptake and development of bots has been increasing (with ensuing opportunities, implications, and concerns), and that this increase reflects wider implications of web-based software and artificial intelligence (AI). Artbots, in this context, are also an important topic of study because they enable critical understandings of these broader developments.

The bot story begins before the World Wide Web (invented by Tim Berners-Lee in 1989). Bots first appeared on other computer network communication systems such as

Usenet, IRC, and MUDs.[2] Many characteristics and ideas, originally conceived on these networks, would later manifest on the Web. This is also the setting in which key bot attributes originated, an important factor that I elaborate upon shortly.

It is also important to note the confluence of various areas of interest in computer science that led towards bots. Usenet and MUDs were being developed at universities, mainly by researchers engaged in studying various possible uses of computers. As such, computer networks would become host to computer science experiments. One area of interest was the development of techniques created in an attempt to pass the *Turing test*, a thought experiment defined by Alan Turing to test whether computers could fool humans into thinking they were human through text-based interaction. Early computer-networks were host to not only science and engineering matters, but to an emerging 'internet culture' of early computer-network participants as well. This created a set of circumstances that led to the notion of bots, which I will illustrate with two case studies. First, I introduce the computer-controlled Usenet character *Mark V. Shaney* (1981); second, I describe a computer-controlled MUD player named *Julia* (1989).

A provocative Usenet newsgroup called *net.suicide* was host to a number of participants adopting fictional characters as their online persona, according to the Usenet Handbook (Harrison 1995: 217-219). The experiment was short-lived but led to the origins of computer-controlled Usenet 'characters'. *Mark V. Shaney* (1981) was designed by Rob Pike, a Canadian programmer and the Usenet administrator for *net.suicide*. The software ran on a Bell Laboratories computer (Bell Laboratories were a major mediator of the Usenet network). The name 'Mark V. Shaney' is a play on 'Markov chains', the algorithms behind the character's software. The software and computer networking were engineered by Bell Lab scientists Bruce Ellis and Don P. Mitchell.[3] It was made active on the Usenet group *net.singles* and communicated with other unsuspecting members of

[2] Usenet was established in 1980. Internet Relay Chat (IRC) was created in Finland, 1988. Multi-User Dungeons are a type of text-based, multi-player computer game, based on the original MUD created in 1980 at the University of Essex.

[3] Markov chains are a computational technique for creating semi-coherent text by following a rule-based analysis and re-ordering of an input text. As the *Usenet Handbook* details in a footnote, this bot "used third-order Markov chains, in which the next word is chosen based on the two previous words; punctuation marks were considered part of the preceding word, which made the output much more reasonable; and the humans were allowed to select and reorder sentences, but not to alter the sentences themselves."

the group. Its software worked by aggregating and analysing the posts from other members from the previous day (p. 227), and from this, generating new utterances that it would pass as its own.[4] In a way, it worked by mixing with humans in two senses: socially intermingling with them, and computationally mangling their conversations into a semi-coherent mix. Arguably, this process also contributes to the *Mark V. Shaney* persona, rendering it as a sort of nonsensical *net.singles* member. The *Mark V. Shaney* project was a proto-bot experiment which demonstrated the key elements of artbots identified by this research study: the *figuration* of a character, the *configuration* of networked software, and the sociotechnical *entanglement* with others.

Usenet, MUDs, and IRC were host to a plethora of experimental software-run processes. Journalist Andrew Leonard of *Wired* documented how the term 'bot' originated to as a label for some of these experiments (1996). Simply put, it was "short for robot" (overlooking, perhaps, the question: 'what is a robot?'). The term 'bot' also began to take on new meanings for network-based software, attributing to it "personality", "behaviour", and even "feelings – even if those feelings are nothing more than cleverly conceived algorithms" (ibid).[5] Through Leonard's account, we witness the myth-making and 'net culture' of the time in the attribution of these properties to mere computer processes. However, there is arguably an ontological shift from a *computer-controlled character* to a *bot*, as if the overlap of the distinct elements of 'computation and fictional character' became a new entangled whole – a new social construct.

[4] An example post reads, "The longer one 'waits' to experience sex, the more important ones [sic] virginity becomes and the more artificially important it can be in relation to the posting on the dead cat and eating of same, which I absolutely refuse to repost."

[5] The full quote is: "DeadelviS, aka John Leth-Nissen, an IRC hacker who maintains an FTP archive of bot source code, defines bot as being 'short for robot, which sounds cooler than program.' […] In current online parlance, the word bot pops up everywhere, flung around carelessly to describe just about any kind of computer program - a logon script, a spellchecker - that performs a task on a network. Strictly speaking, all bots are 'autonomous' - able to react to their environments and make decisions without prompting from their creators; while the master or mistress is brewing coffee, the bot is off retrieving Web documents, exploring a MUD, or combatting Usenet spam. But most bot connoisseurs consider true bots to be more than just mindless ones and zeroes. Even more important than function is behaviour - bona fide bots are programs with personality. Real bots talk, make jokes, have feelings - even if those feelings are nothing more than cleverly conceived algorithms."

One can imagine Leonard writing simultaneously from the perspective of observer and from his own experiences of conversing with these programs directly, and finding meaning and personality in computer-generated text. From the perspective of a bot's programmer, this experience was often intended by design. While studying at Carnegie Mellon University, computer scientist Michael Mauldin used MUDs as a testing ground for work on the Turing test (1994). Mauldin designed *Julia*, which he describes as both as a "computer-controlled player" and "chatterbot" to converse with other human players and "answer questions about other players, rooms and objects" (p. 16). Mauldin used MUDs as an "unsuspecting Turing Test" where "players assume everyone else playing is a human" (p. 17). Conceivably, some players knew of the presence of bots on MUDs, and others did not, but what is important here is that Mauldin (and possibly others) used bots as Turing test contenders, postulating them as human participants. *Chatbots* are considered to be a particular type of conversational bot, and techniques for computationally processing human queries and responding to them are still very much in development and use in contemporary voice-activated assistive devices (Amazon's Alexa, Apple's Siri, and Microsoft's Cortana, to name a few).

Julia was also part of the ludic fabrication of the multiplayer game; MUDs were essentially text-based role-playing experiences, and players of different ranks in the game would often be playing as 'gods', 'wizards', and 'regulars'.[6] Understandably, some of that game-like characterisation has been retained in how internet-software is often described by its designers, or perceived by others. The term 'spam', for example, now associated with unsolicited emails, was coined by Usenet administrator Joel Furr when he compared the ARMM Usenet moderation software to a Monty Python sketch.[7] Soon after, experiments in 'spambots' and 'cancelbots' became regular Usenet adversaries in game-

[6] See, for example, the player listing for the MUD 'DruidMuck': *http://www.logrus.com/~halsted/druidmuck/players.php*.

[7] The exact origin of the term 'spam' is difficult to verify; see *https://en.wikipedia.org/wiki/Joel_Furr*.
The Monty Python sketch, first televised in 1970 and written by Terry Jones and Michael Palin, depicts two customers, trying to order breakfast in a café that has Spam in every dish. The other customers, all Vikings, begin chanting SPAM, SPAM, SPAM. This probably resonated with the nuisance output of the Usenet moderation software, accidently moderating itself in feedback loops, generating ARMM, ARMM, ARMM. The Monty Python sketch is available on YouTube; see *https://youtu.be/g8huXkSaL7o*.

like battles (Leonard 1996). To this day, email services provide 'anti-spam' software and a 'spam' folder.[8]

With the rapid adoption of the new computer network protocol known as the World Wide Web, development of computer-controlled network tasks took a commercial turn. However, some of the flair for characterisation was retained. The metaphor of a 'web' led researchers like Mauldin to think up arachnid-themed descriptions for their new Web-based software applications. *Spider-bots* weren't so much conversational, but instead solved a growing problem of mapping out the ever-expanding Web. These new bots were programmed to computationally browse the interlinked Web and make searchable listings of it. Early Web developers created spider-bots with names such as *MomSpider*,[9] *tarspider*, *Arachnophilia*, *HTMLgobble*, *Websnarf*, *Webfoot*, *churl*, *Peregrinator*, *Scooter*, *Aretha*, *Checkbot*, *Webcrawler* and *Lycos*. It soon became evident that owning these Web listings and providing a public means to search this list would be a commercially valuable service.

Named after the Lycosa tarantula (also known as the 'wolf spider'), Lycos was created by Mauldin in 1993 using code from *Julia*, as one of *Julia*'s functions was to roam and map out *tinyMUD*, the player-extendible MUD world it was built for.[10] This provided an advantageous basis for the Lycos spider-bot, which was left running to explore the Web and provided the list of websites behind one of the first ever *search engines*. The Lycos search engine evolved into a commercially successful business in quick succession: in 1994, Mauldin received two million dollars in investment; in 1996, Lycos became a publicly traded company; by 1999, *lycos.com* was the most visited domain on the Web; finally, in 2000, it sold for US$12.5 billion.

Various early search engine companies (Lycos, Yahoo, Google) were arguably in part built upon the spider-bot skills of a particular generation of computer science graduate students. This fascinating link between spider-bots and search engine companies is often

[8] The first commercial spam is credited to US lawyers Canter and Siegel; see *https://en.wikipedia.org/wiki/Laurence_Canter_and_Martha_Siegel*.

[9] MOMspider was made by Roy Fielding who worked on the WWW protocols and the wrote an influential thesis in 2000 on REST Architecture, which prescribed a new model for the Web. Fielding's practice and writing is discussed further on.

[10] Taken from the extended history of 'Verbot' on Wikipedia; see *https://en.wikipedia.org/wiki/Verbot#TinyMUD*.

overlooked, favouring discussions around the algorithmic ranking of those listings, which led to the success of Google over its competitors through its PageRank algorithm. Spider-bots, however, still play a vital role for search engine companies, who keep their listings up-to-date by aggregating ever more information to supply their services. Many of Google's ventures into other domains mirror this approach; whether it is mapping streets or books, the first step is the deployment of a data-collecting device.

Little is documented about bot-making during the early 2000s. However, it is evident that conversational 'chatbots' resurged on the arrival of Web-based instant messaging services such as AOL Instant Messenger, Yahoo! Messenger, MSN Messenger, and ICQ. Under the influence of the increasing commercialisation of the Web, many of these chatbots were designed by businesses. The AOL bot *dnLFlipit* was made for the 7UP company to promote a new drink called dnL (Olsen 2003).[11] Similarly, AOL created the *Austinpowers* chatbot to promote the movie of the same name (ibid). The multifunctional personal assistant *Smarterchild* was the leading product of ActiveBuddy, a company specialised in instant messaging chatbots. In 2002, the website *runabot.com* was founded to support the craft of instant messaging chatbots and claimed to have registered over 100,000 bots and 20,000 bot-makers.[12]

During the 1990s, the first internet-based art scene emerged. Among other strategies and devices, its participants sometimes developed Web-based software applications for art-activism (Stallabrass 2003). To contextualise this, it is worth appreciating that the commercialisation of the internet was met with resistance and disgruntlement by many early internet adopters, and this attitude was reflected in art practice. Internet artists often parodied the commercialisation of the Web (such as by art collectives eToy, ®TMark, Yes Men), engaged in spreading viruses (Eva and Franco Mattes, Heath Bunting),[13] spammed email lists (JODI), hijacked search engines (eToy), made 'wrong' browsers (JODI),[14] and even ran internet programs to illicitly compile e-books out of

[11] 'dnL' looks like '7up' upside-down, hence the bot 'flips it'.

[12] These figures are taken from an interview with *runabot.com* founder "Dave" on website *IMAddict.com*. The archived interview is available at: *https://web.archive.org/web/20030810114712/http://www.imaddict.com:80/interview-dave.php*.

[13] See *https://0100101110101101.org/biennale-py/* and *https://web.archive.org/web/19970121041235/http://www.irational.org:80/indexes/observer.html*.

[14] For 'Wrong Browser', see *https://youtu.be/_aVk_E0owNA*.

Amazon previews (Ubermorgen).[15] This type of practice has spawned a lineage of artists that continue these internet-activist approaches to this day. Internet art precedes artbot practices in its use of the Web as a new space for encountering art, as well as using Web-based software as an artistic material. However, artbot practices have emerged unrelated from this earlier community of practice and took just as much influence (if not more) from commercial Web developments. More significantly, artbots emerged out of a new *Web 2.0* movement that blurred the distinctions between Web commercialisation and community-led activity.

The emergence of artbots

The term *artbot* emerged in the last decade to heuristically mark a divergence away from chatbots and impersonation bots, and to specify the works of a new community of artistic practice. In the words of Alex Hern (2016), writing for the Guardian: "Move over, chatbots: meet the artbots." This study investigates these artbot practices and presents an analysis and framework that goes beyond mere disambiguation. What this thesis does *not* do is to attempt to historicise this movement; instead, it seeks to examine and analyse this approach to practice. This introduction does, however, cover an outline of its story and key proponents. As this account illustrates, the artbot scene brought together a mix of influences ranging from poetry, videogaming, interaction design, and media, and has co-evolved alongside developments in Web 2.0 technologies and platforms.

Instant messaging declined in popularity as a means to communicate across the Web around the mid-2000s as it began to compete with new Web 2.0 social networking and media-sharing platforms. To give a sense of the times, Facebook was founded in 2004, Twitter in 2006, and the first Apple iPhone was released in 2007. The Web 2.0 era can be understood as ushering in a shift in both technical protocols and ideological values that support 'openness', 'freedom', 'community', and 'inter-connectivity'; some saw this as bringing the Web back to its founding principles. New Web companies began to design their systems and Web services to be extendable, with modularity, inter-connectivity, and third-party appropriation in mind. As such, web services would begin offering *application programming interfaces* (APIs) to enable Web-based software applications to communicate with the platform's servers, as well as send and receive data and services programmatically.

[15] See *http://www.amazon-noir.com/index0000.html*.

Flickr was one of the first Web companies to offer an API, and was soon followed by Facebook, Twitter, and others. By 2010, Twitter had over 70,000 registered third-party applications.[16] A programming practice that grew around interoperability and the use of multiple APIs was often referred to as *mashup* (Yu et al. 2008).[17] The mashup approach was used to experiment with reconfigurations and relations between increasingly available APIs, Web services, and data resources.

These new tools and circumstances inspired a new approach to making bots. In London, a small but influential scene of interaction designer-technologists shared associates and ideas, mainly between two small design studios – BERG and RIG.[18] Some of these designers had previously worked on instant messaging chatbots,[19] but now APIs and social networking platforms provided new directions for bot experimentation. Designer and technologist Tom Armitage created *Tower Bridge Bot* in 2008.[20] For Armitage, Twitter was more than a messaging service; he called it a "messaging bus", an open-ended node through which other things could be connected. *Tower Bridge Bot* was his personal investigation into "what [it is] like to be friends with a bridge" (Armitage 2007).[21] The bot output the physical activity of Tower Bridge (in the first person, as if it were the bridge) to Twitter, tweeting about opening and closing, as well as the names of ships passing through. Other bots from this London scene created in that same year include Phil Gyford's *Samuel Pepys*,[22] Tom Taylor's *Low Flying Rocks* (warning of near-Earth asteroids),[23] and, later on, James Bridle's *A Ship Adrift* (2012)[24] and Matt

[16] This figure was published by Twitter founder Biz Stone in a company email newsletter in 2010 and reported in the New Statesman; see *https://www.newstatesman.com/digital/2010/03/twitter-registered-created*.

[17] *https://en.wikipedia.org/wiki/Mashup*

[18] Both studios are now no longer active, but their websites are still available at *berglondon.com* and *riglondon.com*. The two studios were closely related and worked next door to each other.

[19] See a comment from Matt Webb on *https://mastodon.social/@genmon/100746396749546468*.

[20] See *https://twitter.com/twrbrdg_itself*, and for Armitage's notes on the project, see *https://infovore.org/archives/2008/02/28/making-bridges-talk/*.

[21] Armitage gave a talk about Tower Bridge Bot in 2012. The given quote appears at 23:55; see *https://archive.dconstruct.org/2012/makingfriends*.

[22] Running since 2008, see *https://twitter.com/samuelpepys*.

[23] See *https://twitter.com/lowflyingrocks*.

[24] Made in 2012 and commissioned by Artangel; see *http://jamesbridle.com/works/a-ship-adrift*. Bridle has since positioned himself as an artist and his artwork is highly acclaimed.

Webb's *5point9billion* (2015). Bridle's *A Ship Adrift* was a Twitter bot that uttered extracts from geo-located data resources, steered by data coming from a weather station. Bridle is also known for *Dronestagram* (2012), an Instagram account that publishes Google Maps satellite views of recent drone-stricken areas.[25] Although manually edited, it is nevertheless a relational arrangement of incongruent Web services and resources in mashup fashion. Bridle played a role in promoting ideas originating from RIG and BERG with his notion of the 'New Aesthetic', which for some marked an emerging new artform.[26] Despite their influence on this community of practice, both RIG and BERG folded. However, many of their bots continued in operation, maintained by their designers. These can be considered a proto-artbot scene that introduced a turn towards bots constituted of 'mashup' configurations that were open to wider characterisations beyond Turing test impersonators and commercial chatbots – to be figured as ships, bridges, literary figures, and so on.

Around the same time, a disparate number of US-based practitioners began their own experiments, several of whom were interested in ideas that expanded on language-based tools and poetry. One of the first Twitter bots was Allison Parrish's *Everyword* (2007), which posted every word in the English language, one at a time. Parrish's background included videogames and software development, but she also had a personal interest in computational approaches to poetry. *Everyword* was different from other bots. It wasn't a chatbot; it had no persona-based characterisation; it did not interact with others; and although text-based, it did not form sentences – it simply emitted one word at a time, and took years to complete its task. This bot was a platform-based experiment in abstract poetry, and it showcased a restrained, conceptual simplicity over technical prowess. It was one the first *artbots*, a term now retroactively applicable to this experiment. Following *Everyword*, Parrish created a number of artbots (text-based and image-based), and now teaches at ITP NYU. Through her teaching and artwork, Parrish has helped to establish this new type of practice.[27]

---

[25] See *http://jamesbridle.com/works/dronestagram*.

[26] Starting out as an informal research blog, the New Aesthetic evolved into a term that describes the visual upshots of computational systems, especially those that use computer-vision technologies and digital image formatting; see *http://new-aesthetic.tumblr.com/*. See Bruce Sterling's response to the New Aesthetic which understood it as a new, emerging artform: *https://www.wired.com/beyond-the-beyond/2012/04/an-essay-on-the-new-aesthetic/*.

[27] Parrish also holds a master's degree from NYU's Interactive Telecommunications Program; see *https://tisch.nyu.edu/about/directory/itp/853082171*.

Another influential Twitter bot that stemmed from an interest in experimental poetry is *Pentametron*, by artist Ranjit Bhatnagar. Although *Pentametron* was created in 2012, it was conceived in 2009 as a Twitter engagement project for the Brooklyn Museum of Art.[28] The original version invited people to participate in composing consecutive verses of a multi-authored sonnet over Twitter.[29] *Pentametron* automated the process by analysing the entire flood of publicly available Twitter messages, accessed through the API, and opportunistically 'retweeted' those that were in iambic pentameter and rhyming.[30]

Soon after *Pentametron*, artist and programmer Darius Kazemi made his artbot debut with *Metaphor-a-Minute!*,[31] the first of many influential bots and internet software projects. Kazemi worked in and around videogaming, having spent six years as a gameplay analyst, as well as independently making and writing about games. In 2012, he joined internet consultancy firm Bocoup as an HTML5 game developer with the remit of bringing videogames to the internet. Kazemi, upon reflection, re-oriented his approach to this challenge, and saw the internet and its constellation of interconnected elements as a creative medium upon which gaming strategies could be implemented:

> "In 2012 I realized that a huge part of my interest in games came from an interest in generative experiences, serendipitous systems, and randomness. I had a lot of fun figuring out how that worked in the context of games: level generators and AI and that kind of stuff. But you know what? Turns out the

[28] This is archived on *https://web.archive.org/web/20121024045304/http://www.brooklynmuseum.org/support/1stfans_twitter_art_feed.php?artist_id=7*

[29] Earlier to this, Bhatnagar had also experimented with multi-authored sonnets on Usenet. As he lists: "crowdsourced sonnets on USENET, [a] similar project for Brooklyn Museum, discovering Twitter streaming API and CMUDICT simultaneously and making *Pentametron*".

[30] Bhatnajar revealed that *Pentametron* looked at about a million tweets per couplet; see *https://web.archive.org/web/20120725013629/http://gawker.com/5905550/weird-internets-the-amazing-found+on+twitter-sonnets-of-pentametron.*

[31] This bot was an assemblage of heterogenous components that include the Wordnik API (a dictionary service), the Twitter API, Node.js (server-side software), and Nodejitsu (the server that hosted the Kazemi's software). It was inspired by Ian Bogost's "Alien Phenomenology" (Bogost 2012). See *http://tinysubversions.com/2012/05/how-i-built-metaphor-a-minute/index.html*.

> Internet ecosystem itself is a way more fertile and effective medium for exploring that kind of thing."[32]

In that year, Kazemi made a number of experimental projects to test this direction. One of the more successful ones was *Random Shopper*, an internet program that randomly bought items from Amazon, coaxing from the Web the randomness and serendipity that Kazemi sought. The shopper was a ludic fabrication implemented within the real-life internet ecosystem itself, and one that further enacted the shipment of physical goods. One of the facets of Kazemi's practice is the myriad of characterisations a bot can take, often figured through a simple process such as randomised shopping, or sometimes through a reference to popular culture. Kazemi quickly built a peer group of bot-makers, aided by his penchant for sharing code and community-building, which greatly contributed to the uptake of artbots.[33]

The *Random Shopper* idea was later appropriated by Mediengruppe Bitnik, an art duo more aligned to communities of practice concerned with internet art-activism. They created *Random Darknet Shopper*, which randomised shopping on an 'unlisted' internet shopping site – not accessible on the Web – that sold items such as contraband and drugs (Power 2014). The variances amongst these two shoppers helps identify differences in approach to their practice. Earlier proponents of internet art associate with the pre-commercialisation of the internet in which bot activity deliberately inflicted anxiety on encroaching mainstream adoption and commercial exploitation of the internet. Kazemi's internet art, on the other hand, begins with the more presently ubiquitous and everyday enactment of software-mediated sociality, and through playful re-arrangements and figurations, operates as part of it.

> "Kazemi and his bot-making friends can be seen as exploring a medium through which we now do much of our everyday business—and then rerouting the wiring that underlies that medium… Kazemi may have found nothing less than a

[32] Taken from Kazemi's web slide presentation "Fuck Videogames": *https://tinysubversions.com/fuckvideogames/#slide19*.

[33] Kazemi ran two community-building initiatives: Bot Summit and NaNoGenMo (National Novel Generation Month). These have had a huge impact on the creation of a network surrounding Kazemi.

> new kind of public art for the 21st century—changing, self-referential, and in its insistent randomness, oddly alive." (Neyfakh 2014)[34]

More practitioners from videogaming, especially those stemming from a computer-generated approach to game design, began making artbots. One such practitioner is researcher Mike Cook, who runs Procjam,[35] a generative videogame competition that often crosses over into bot-making. Another is US-based artist Kate Compton, who created Tracery, a tool used to computationally construct language fragments or images from a human-specified grammar. In 2015, London-based game designer George Buckenham bundled Tracery into their bot-making service *Cheap Bots Done Quick*,[36] a resource that has greatly increased accessibility to the practice and, at the time of writing, hosts over 5,000 bots.[37]

Other notable participants of the artbot scene simply discovered the emerging practice on Twitter early on. Li Zilles, a computer scientist interested in natural-language processing, created *Portmanteau Bot* (2014) – a generator of portmanteaus. Their most popular work is *Wikisext*, an artbot which generates odd flirtatious messages based on texts from a 'how-to' site. Everest Pipkin similarly came across bots on Twitter while living in rural Minnesota in 2012 and made *Feelings.js* in an afternoon by reconfiguring code and resources found online. Five days later, Pipkin made *Tiny Star Field*, a bot that generates ASCII-based graphic star fields within the text-only limitation of Twitter of that period. The visual simplicity of its output was a charming deviation from readable tweets, and the artbot gained an unprecedented hundreds of thousands of followers.[38] One of those followers was London-based musician and researcher, Emma Winston, who extended the 'tiny' concept into a series of bots including *Tiny Gardens*, *Tiny Cities*, and *Tiny Gallery*.[39] Winston's series uses emojis rather than ASCII symbols, reflecting the rising adoption of emojis in the co-evolution of smartphones and social

---

[34] Leon Neyfakh has written the most extensive account of Kazemi's work circa 2012-2014 for the Boston Globe; see *https://www.bostonglobe.com/ideas/2014/01/24/the-botmaker-who-sees-through-internet/V7Qn7HU8TPPl7MSM2TvbsJ/story.html*.

[35] Procjam's motto is "make something that makes something"; see *http://www.procjam.com/*.

[36] CBDQ can be found at *https://cheapbotsdonequick.com/*.

[37] A stat taken from Buckenham's Patreon account; see *https://www.patreon.com/v21*.

[38] For a detailed description of Pipkin's practice, see *https://www.furtherfield.org/about-a-bot-interview-with-katie-rose-pipkin/*.

[39] For a full list of Winston's bots on Twitter, see *https://twitter.com/deer_ful/lists/emma-s-bots/members*.

media. Winston's background in music also led to *Graphic Score Bot*, a generator of visual scores for music improvisation.

Another influx to artbots worth noting came from more traditional media. In 2013, Rob Dubbin, a writer for the US TV show *The Colbert Report*, made the artbots *RealHumanPraise* and *Olivia Taters*.[40] Similarly, Jacob Bakkila, a creative strategist for Buzzfeed, ran the hugely followed Twitter account *Horse_ebooks*. Bakkila bought the account off a Russian spammer and then manually operated in the style of a spambot, penning what appeared as serendipitous moments of 'spam-poetics'. Another Buzzfeed writer based in Australia, Chris Rodley, in collaboration with his sister Ali Rodley, created *Magic Realism Bot*, a popular bot that generates abstract texts inspired by the magic realist stories of Jorge Luis Borges.[41]

Whilst many names have been left out of this introductory account of artbot practitioners, more will be mentioned later on in more detailed analyses of different works. One particular underrepresentation that is concerning, however, is the uptake of artbots from movements outside of Euro-American, English-speaking circles. South and Central American practitioners are actively making bots in Spanish; Leonardo Flores, a scholar based in Puerto Rico, documents these practices.[42] Some English-based bots have been translated; for example, Kazemi's *Two Headlines* exists as *Dos Titulos* by Argentinian Juan E. D. (JED),[43] and Parrish's *Everyword* also appears in a multitude of languages. JED addresses Argentinian issues through bots such as *Demoliendo Buenos Aires*, which tracks the demolition of historical Buenos Aires and the replacement apartment blocks that later appear visible on Google Street View.[44] JED masterfully interconnects APIs from disparate services to reveal social injustices in his own country. If JED's artbot work is exemplary of developments in Spanish-speaking countries, it makes sense to assume that artbots are being explored in other languages and contexts as well.

---

[40] See Dubbin's own article in the New Yorker: *https://www.newyorker.com/tech/elements/the-rise-of-twitter-bots*.

[41] Magic Realism bot can be found at *https://twitter.com/MagicRealismBot* and further described on Chris Rodley's website: *https://chrisrodley.com/2015/10/10/magic-realism-bot/*.

[42] See Leonardo Flores's research on *http://leonardoflores.net/* and *http://iloveepoetry.org/*.

[43] See *https://twitter.com/DosTitulos* and a complete list of JED's work on *https://twitter.com/j_e_d/lists/bots/members*.

[44] See *https://twitter.com/DemoliendoBA/*.

Prior notions of computer-related art

It is useful to acknowledge prior existing notions of computer-related art and assess whether their framing can contribute or lend ideas to the artbot practices in study. Terms such as *computer art*, *algorithmic art*, *generative art*, *software art*, *internet art*, *new media art*, and *digital art* still permeate understandings of computer-related approaches to art, and are inevitably applied to describe artbots. The question is, are these terms applicable? And if so, what can these terms do to contribute to an understanding (or misunderstanding) of artbots? Arguably, these terms are deeply entangled with and formed by their prior communities of practice, values, aesthetics, and a range of other implicated actors (institutions, curators, writers, and key texts). As such, they are not necessarily apt or adaptable to conveying a new arrangement of practitioners, artworks, actors, aesthetics, and so on. However, there may be useful intersections in the frameworks and conceptual tools they provide. Moreover, these terms form part of the discussion (and contextualisation) of artbots, by both association and related lineages of practice. This section provides a concise overview of these genres, as well as the readings and understandings they may provide to the study of artbots.

Other than their association to prior art communities, many terms for computer-related art tend to frame a practice in the context of a specific 'material' or 'medium' (e.g. software, internet, media, digital). Starting with the broadest of these, *computer art* emerged in 1965 to describe art made with computers. Since its inception, a lot has changed about what a computer is, who does and does not have access to computers, and how this access facilitates and mediates creative practice. Arguably, computer art has over time become inextricably related to – and sometimes synonymous with – *computer-generated imagery* (CGI). 'Computer art' was first used to describe the technical development and production of visual output using computers (Dietrich 1986). Technical developments in CGI now encompass hardware (e.g. GPUs),[45] data protocols (e.g. digital image formats such as JPEG), and software (e.g. digital image editors such as Photoshop). During the 1960s, scientists studying CGI began producing computer-generated visual art at institutions such as Bell Laboratories in New Jersey, and

[45] *Graphical processing units*, or GPUs, are a specialised electronic circuit for accelerating the creation of digital images for output to a display device.

Technische Universitat in Stuttgart, Germany (ibid). The work of Michael Noll, who was based at Bell Laboratories (circa 1962), is an example of these early practices.[46]

To this day, CGI technologies are in part driven by enquiry into the visually realistic simulation of phenomena such as mountains, fluids, cloth, physics, hair, and natural light. As both a term and a framework, computer art often invokes this, which can lead to assumptions about 'computer art' being inherently representational and visual. Artbot practices, on the other hand, produce a variety of media and are usually text-based, however, describing artbot works as 'computer art' or 'computer-generated art' does identify the computer as an element in between the artist and artwork. In artbots, the configuring of a software-platform assemblage that produces 'art' during runtime could be described as 'computer-generated'. However, the output of this system does not necessarily constitute the art; rather, the 'generator' is considered the artwork. Moreover, some artbot practitioners would argue that the whole configuration, including its ongoing generative processes, constitute the artwork. Likewise, the terms 'algorithmic art' and 'generative art' describe similar approaches that produce computer-generated artworks, with an emphasis on the software instructions and processes which generate them.

With the adoption of these notions of practice, the concept of characterising the 'software-as-artist' begins to emerge. The aforementioned Technische Universitat was under the influence of philosopher Max Bense, who studied mathematical aesthetics and coined the terms 'artificial art' and 'generative aesthetics' (Bense 1965). Bense and his devotees advocated a mathematical, process-based visual art that could be enacted by computers. An adoptee of Bense's ideas was philosopher-turned-computer-artist Hiroshi Kawano, who compared his relationship with the computer as a father instructing a child on how to draw (Kawano 1976). This characterisation of the computer as a 'learning' entity has been considered prevalent in computer science (Castañeda and Suchman 2014). Kawano is an early example of programmers endowing character onto software, and imagining it as a separate and autonomous entity – often with the goal of characterising software as an artist, a goal still pursued by some (see, for example, Colton 2012).[47] Artbots equally share in the desire to create computational personae, but differ in that their practices experiment with a whole array of figures other than artists.

[46] See Noll's early works at *http://www.citi.columbia.edu/amnoll/CompArtExamples.html*.

[47] Colton created *The Painting Fool*, a "software that we hope will one day be taken seriously as a creative artist in its own right."

Software art is a term that surfaced around in the early 2000s to describe a community of practice exploring and critiquing the increasing ubiquity of software applications. A critical form of computer-related art practice emerged (Fuller 2003), conducting work that deconstructs, appropriates, and misuses existing software – even producing parodies of software, such as Adrian Ward's *Auto-Illustrator* (Cramer and Gabriel 2001). Software art borrowed from (and overlapped with) internet art in its resistance and distrust of the increased commercialisation and ubiquity of computer technology. This period also coincided with the open-source software movement (Lakhani and Wolf 2003). Software art, in various ways, is similar to the practice of artbots given that these are often reconfigurations of existing software modules that produce interventions into ubiquitous Web-based software arrangements (that involve entanglements with humans). Artbot practitioners, however, are more engaged in the characterisation of software, and tend to produce work for larger audiences through social media platforms. The overlaps between the two communities of practice are certainly thought-provoking, and Fuller's proposal for "Speculative Software" (Fuller 2002), which imagines a development of software art concerned with "software as science fiction", resonates closer still.

Internet art is another term that approximates a description of what artbots are about – namely, operating on the Web and using Web-based coding and protocols as material. During the 1990s, a number of internet art pioneers emerged to form a recognised community of practice, including names such as JODI, Heath Bunting, Rachel Baker, Olia Lialina, and Vuk Cosic (a group also referred to as 'net.art'). As such, when artbot practitioners describe themselves as 'internet artists', it is done with precautions to distinguish themselves from early internet art. Kazemi, for example, often uses the term "weird internet art" (Neyfakh 2014). The disparate communities of early internet art and artbots are notably influenced by the periods (and technological changes) that separate them; the former movement creatively explored HTML and standalone websites (amongst other things), and the contemporary movement is arguably more interested in social media platforms and APIs. Artbot practice is a sort of 'internet art' of Web 2.0.

Notions of *new media art* and *digital art* are arguably less focused on specific artistic communities, and instead represent broader areas of creative practice. However, certain traits are still conveyed by these terms. *New media art* was in part theorised by Lev Manovich (2001) to describe art that is produced *and* distributed by computer technology, conjoining artistic production (e.g. using computer graphics applications) with the dissemination of these creations through computer-mediated channels such as

the internet, videogames, and virtual reality. New media art, however, tends to tacitly place 'newness' at the fore of practice conducted under this banner; in other words, new media artists are often motivated by technical innovations in the production and dissemination of media. Artbot practices, by contrast, often eschew innovation and instead reconfigure existing resources: APIs, software modules, media archives, and datasets. New media art discourse has gradually shifted to the even broader notion of *media art*, where the mediums it represents are no longer considered 'new' (Quaranta 2013; Manovich 2013).

Digital art has become a broadly used umbrella term that covers notions of computer art (CGI practitioners are often referred to as 'digital artists') and media art, as well as conveying practices that operate within a wider 'digital' turn in society (social media, smartphones, and ubiquitous digitised imagery).[48] The broad use of the term makes it difficult to utilise with any specificity, although it is more closely aligned to the digitisation of visual information that manifests on screens, smartphones, websites, and virtual or augmented reality. The notion suggests a focus on substrates for visual aesthetics rather than specific artist communities. The more niche term 'postdigital' often alludes to crossovers with non-digital substrates such as books (Cramer 2015, Berry and Dieter 2015). Similarly, the term 'postinternet' has surfaced to describe crossovers between internet-based art and non-internet distributions and mediums that are closer aligned to art gallery exhibitions (Cornell, Halter, and Phillips 2015, Olson 2012). The notion of 'digital art' rarely assists with nuanced readings of practice; when used as an umbrella term, 'digital art' may problematically ensnare new artist communities under it.

Artbots, although realised with computational elements, enable both human and nonhuman activity. Rather than producing objects of 'art' with a specific medium, artbot practices can be said to be more attentive to setting up the processes of the art itself. For this reason, an approach to describing these practices would aptly suggest their *methods*, rather than their *medium*. This resonates with Deleuze's essay *What Can a Body Do?* (1990: 217) which shifts thinking towards questions of what becomes enacted through the coming-together of elements. This effective shift in enquiry is useful for the study of

[48] The UK Arts Council, from 2018 onwards, require all their funded arts organisations (NPOs) to have a "digital policy and plan to show how digital is strategically embedded across work that they do"; see *https://www.artscouncil.org.uk/digital-culture-2017*.

artistic practice: by asking what a practice *can do*, or purports *to do*, a framework is created around its *practices* rather than things.

To reiterate, this study is a response to the lack of an adequate framing and understandings of artbots. As such, much of the literature review in the following chapter looks thoroughly for theoretical tools and notions that help build discussions and readings of the complex, process-led, socio-technical interventions and characterisations that are produced by these practices. This enquiry led to a framework of analysis that furthers the examination and discourse for this type of practice.

Note on interdisciplinarity

Conducted within the Department of Design at Goldsmiths, University of London, this study takes an interdisciplinary approach. This reflects the department's own diverse community of practice that encompasses influences and research from sociology, art, and computing; furthermore, it avoids compartmentalising the field of design into discrete types such as interactive design, graphic design, speculative design, fashion design, and so on. Moreover, this doctorate came about after a period of time working at the Interaction Research Studio (IRS), an interdisciplinary research group composed of designers, sociologists, and technologists that conducts practice-based and ethnographic studies of human interactions with computational devices.[49]

Artbot practices are similarly interwoven in an assemblage of discourses, histories, values, and modes of practices that are impossible to disentangle towards a particular disciplinary approach. Instead, this study examines this "mangle of practice" (Pickering 2010) and makes sense of it through new understandings and concepts. More information about the methodological matters and issues of this study are more thoroughly provided in the third chapter. This study's investigations can be understood to contribute across (or in-between) disciplines, and the conclusion broadens this discussion, reflecting on how this research may contribute towards new understandings and knowledge for different communities of scholarship and practice.

[49] It was at IRS that I developed my first bots circa 2012, which in turn helped resource a wider research project (see Wilkie, Michael, and Plummer-Fernandez 2014). For more information about the Interaction Research Studio, see *https://www.gold.ac.uk/interaction/*.

Thesis outline

This thesis is structured in a way that edges towards more and more detail about artbot practice. Starting with developing the conceptual tools and framework to understand software configurations and artbots, the thesis proceeds to examine case studies through this framework. A literature review assesses discourses that support an understanding of internet-based software architecture and software-mediated sociality towards the development of the notion of *sociocomputational assemblages*. The methodology section then describes the practice-based research and participant observation conducted for the study, as well as 'research through collaboration' and a showcase of artbot works. This is followed by three practice-led chapters examining artbots. Some of these case studies are original works conducted as part of this research, whose presentation is interspersed with others.

Chapter 2 reviews a range of literature which examines concerns of internet-based software. The increase in literature around matters of 'algorithms' is reviewed. I introduce Deleuze and Guattari's notion of *assemblage* as a tool to think about the complex constellations of software processes, computers, data, and humans. The notion of *sociocomputational assemblages* is proposed as a more nuanced method for specifying intra-related social *and* computational configurations. These theories are empirically shown to be in keeping with the Web-based software protocols introduced by Roy Fielding, which proposes a Web architecture based on configurations of software components that become inter-operable during system 'runtime'. The notion of sociocomputational assemblages is shown to add further analysis to how these architectures become 'entangled' with human activity. Moreover, I introduce Suchman's notion of *figuration* that describes how human-computer interactions are often characterised. Based on these theoretical ideas, I develop an analytical framework comprised of *multiplicity*, *entanglement*, and *figuration* to examine artbots.

Chapter 3 presents the methodology and research methods utilised throughout this research study. The methodology involved 'opportunistic' participant-observation of artbot practitioners and their work. Much of this engagement happened over social-media platforms, software repositories, and a blog of mutual interest. Conducting one's own practice is discussed as a method for participating in a community of practice, as well as inventive methods such as 'research through collaboration'. A collaboration with the artist Julien Deswaef assisted in gaining an insider's view into his practice, as well as making experimental work through the collaboration. In addition to this, a curated

showcase called *Art of Bots* brought together seminal practitioners for the first time, and enabled new engagements and insights into artbots.

Chapters 4, 5, and 6 take the reader through empirical examples of modes of practice, specific works, and their analysis. The case studies are a mix of the author's own original works and the works of others. The chapters are structured around the three elements of the framework for understanding this type of practice. Chapter 4, on *multiplicity*, examines how sociocomputational assemblages are configured and become generative of new material over long periods of activity. Chapter 5, on *entanglement*, discusses the entangling of human and nonhuman elements in various forms of intra-active arrangements. Chapter 6, on *figuration*, studies how sociocomputational arrangements are characterised, and how artbots depict playful alternatives to the normative tropes of computational figures.

Chapter 7, the conclusion, restates the research area and summarises the thesis, discussing key findings. This chapter also outlines the implications of the study for further research and future developments of artistic practice, drawing on the study's contribution to knowledge on artbots.

Chapter 2: A framework for understanding artbots

Artbots are internet-based software applications imbued with character that enact behaviours, often to engage online audiences through social media platforms. An aim of this literature review is to support this intriguing practice by developing a theoretical framework that helps examine, analyse, and understand artbots. A problem faced in developing this framework is that internet-based software applications are often misunderstood, and subject to much confusion and speculation. Internet-based software processes are increasingly encroaching on everyday human activities, and critical reflections of these processes have become topical in the press and scholarship. Often, these processes are referred to as 'algorithms' – an emphasis popularised by books such as Christopher Steiner's *Automate This: How Algorithms Came to Rule Our World* (2012), and talks such as Kevin Slavin's *How Algorithms Shape Our World* (2011).

In this literature review, I first survey the growing collection of literature on *critical algorithm studies* (CAS) to review how scholars have begun to assess and understand algorithms and their role in human activity. In particular, I look at "Algorithms and their Others" by Paul Dourish (2016), which already draws upon a wide survey of CAS literature. This is followed by an analysis of more recent CAS literature that shifts the focus of study towards wider ensembles of algorithms and other elements. In contrast, I also review the literature on *software architecture* (Perry and Wolf 1992; Fielding 2000), which not only produces abstractions of heterogenous arrangements of internet-based software components, but is demonstrably linked to the history and practice of online software development. Software architecture is shown to provide an understanding of internet-based software applications as distributed and emergent configurations of reusable and scalable software components.

I then employ the notion of *assemblage* developed by Deleuze and Guattari (1987) and expanded by scholars (L. Suchman 2007; Orlikowski 2007) to suggest that configurations of both computational *and* social processes can be understood as *sociocomputational assemblages*. Having identified artbots as experimental sociocomputational configurations, I develop a framework for the study, analysis, and understanding of artbots.

What is an algorithm?

In this section, I introduce how algorithms are understood in computer science, both theoretically and by way of empirical examples, such as commonplace sorting algorithms. I then draw on theory from the cross-disciplinary interest in critical algorithm studies, particularly drawing on Dourish's discussion of the algorithm and their 'others' – namely, the supplementary processes that work alongside a singular and dispersed algorithm within software. I argue that Dourish's notion of the algorithm and its dispersed positioning within software is not entirely adequate, and that from practice we can observe and develop the argument that software can be better understood as contingent upon multiple algorithms. For this thesis, the value of this lies in supporting an understanding of artbot practices, which will be developed in further chapters.

In computer science, two frequently cited references to algorithms are *The Art of Computer Programming* by Knuth (1968) and the widely studied textbook *Introduction to Algorithms* by Cormen et al (1990).[50] Both attempt to provide definitions for the word 'algorithm'. However, it is commonplace within the discipline to acknowledge that no conclusive formal definition has been established and is rarely attempted (Moschovakis 2000), and that the notion is hard to pin down because it is expanding (Gurevich 2012). The Knuth notion of algorithm is not so much a definition as a list of defining properties of algorithms, i.e. "finiteness", "definiteness", "input", "output", and "effectiveness". The Cormen definition has been cited outside computer science to some approval from the growing CAS community (Seaver 2014) and declares:

> *"An algorithm is any well-defined computational procedure that takes some value, or set of values, as input and produces some value, or set of values, as output. An algorithm is thus a sequence of computational steps that transform the input into the output."* (Cormen et al. 1990)

The Cormen text describes algorithms as instructions ("a sequence of computational steps") for processing data (a "set of values"), and that these instructions have to be explicitly transcribed into code for a computer to execute ("well-defined", "computational"). Knuth's properties of algorithms also place emphasis on inputs and outputs – i.e. the "set of values" given to the algorithm before it begins its "sequence of steps", and the outgoing values calculated as a result of that sequence (Knuth 1968). This is perhaps more useful as a heuristic visual aid for imagining an algorithm in

[50] *Introduction to Algorithms* interestingly depicts a sculpture by Calder on its cover.

principle: data is input into the algorithm, instructions are executed to process this data, and the processed results are output. But this viewpoint can manifest as a challenging "black-box" (Latour 1999) in which it is assumed (and criticised) that between input and output, the internal processes of algorithms become hidden from view (MacKenzie 2005, Diakopoulos 2013, Kitchin 2014).

Critical algorithm studies (CAS) is not a discipline in itself but a collation of literature (Seaver, Gillespie 2015) that spans various disciplines, including sociology, anthropology, science and technology studies, communication, media studies, and legal studies, among others. CAS has a shared interest in engaging critically with the social, cultural, and technical aspects and implications of algorithms beyond the intrinsic interest that originates from mathematics, computer science, and software engineering. Although CAS scholars have significantly contributed to an urgent and critical discourse regarding algorithms, it has often been with a focus on algorithms of "public relevance" (Gillespie 2014), referring to algorithms that interface to "publics" of users on a large scale. Examples of algorithms within this category include search algorithms (Granka 2010), recommendation algorithms (Seaver 2012), and social media platform algorithms such as Facebook (Bucher 2017b). Gillespie notes the difficulty in benchmarking what is and is not of public relevance, and attempts to do this by way of scale and complexity, identifying that algorithms such as Google's search algorithm would process hundreds of 'signals' – different sources of data – to return adequate search results to a user's search query. I argue that this can overlook the relevance of much simpler algorithms that process far fewer signals, but nevertheless have become widespread in their usage. To do this, I shall demonstrate the significance of basic sorting algorithms in their empirical use.

One of the most basic procedures for which an algorithm may be utilised is to reconfigure the ordering of a set of values, and for this a whole range of *sorting algorithms* have been invented, such as *bucket sort*, *bubble sort*, *insertion sort*, *selection sort*, *quicksort*, *heap sort*, *shell sort*, *comb sort* and *merge sort* (Cormen et al. 2001). *Merge sort*, for example, works by dividing an unsorted list of input data into a number of sublists. Each of those sublists is independently sorted before being merged with another sorted sublist, resulting in half the original number of sublists. These new sublists are then sorted and merged, again reducing the number of sublists by a half. This process is repeated until all the values are sorted into one list. This may seem like an overtly complicated way of arranging data, but for large sets it becomes necessary to

divide the process into smaller tasks. Merge sort, by way of a simplified example, looks like this:

Initial 'Input' sequence (8 values)
[5] [2] [4] [6] [3] [1] [9] [7]
Become 4 sorted sublists
[2 5] [4 6] [1 3] [7 9]
Become 2 sorted sublists
[2 4 5 6] [1 3 7 9]
Sorted 'Output' sequence
[1 2 3 4 5 6 7 9]

Merge sort was invented in 1945 by John Von Neumann (Knuth 1968), one of the founding figures of modern computing, who wrote a version of it for the EDVAC.[51] It is easy to overlook merge sort as simply a curio of computing history, or as a starter exercise for getting to grips with algorithms. In practice, however, merge sort is still in use and executed by computers everywhere. This elementary algorithm has become the integrated sorting method of numerous mainstream programming languages such as Perl, Java, and more recently Python, which uses a newer variant of merge sort called Timsort.[52]

The merge sort algorithm helps illustrate a couple of key points concerning algorithms. First, algorithms can be both one and many at the same time. Merge sort is a discrete design for an algorithm, but its implementation occurs frequently; it is instantiated and executed by potentially millions of operational software systems. This example illustrates the problem with the misguided view that algorithms are singular and discrete entities that exist once, a thought that is popularised by the marketing of proprietary algorithms such as the 'Facebook Newsfeed algorithm'.

The second point, I argue, is that underlying many different types of software is a myriad of algorithms of various degrees of complexity, many of which are incorporated

[51] The EDVAC (Electronic Discrete Variable Automatic Computer) was an early binary computer the size of several rows of bookcases, in operation between 1951 and 1961.
[52] TimSort is a derivative of Merge Sort and Insertion Sort and was implemented by Tim Peters in 2002 for use in the Python programming language.

into software by way of readily available software modules. This approach to software design is a growing trend that Lev Manovich describes as the "modularisation" of software (2013). As software-programming practices have developed, more and more effort has been made into making code re-usable and inter-operational. In practices of *object-oriented programming*, software processes are collated into self-contained sets, often in a technical format called a *class*, and described as *objects*. These are designed to be usable and reusable, without needing to alter the contents of these classes. A class typically contains a set of *functions* that can be invoked externally. These classes are often repackaged as modular components of software that extend a broader software framework.

To make readily available code easier to append onto pieces of software, classes are usually bundled as *libraries* or *packages* designed for a particular programming language. Software developers are arguably increasingly at ease with coding by way of making arrangements of pre-existing code. For instance, few programmers using the programming language Python would directly code an algorithm that generates a random number, but would simply append onto their software the 'random' module for Python, a package that contains functions for returning random values. Very rarely would a programmer engage with the underlying, low-level code that implements the design of an elementary algorithm.

"Algorithms and their Others" by Dourish (2016) does a lot of the groundwork of building upon and critiquing the growing CAS literature, notably citing Mackenzie (2015), Gillespie (2014), Introna (2015), and Burrell (2015), to argue for a more rigorous understanding of algorithms. Dourish builds upon Niklaus Wirth's view that "algorithms + data structures = programs" (1978), and further argues that algorithms need to be analysed alongside their "others" – associated concepts such as "automation", "code", "architecture" and "materialisation". Dourish (p. 2) maintains that this is necessary to avoid an "essentialising view of algorithms" in which algorithms would be singled out for discussion, downplaying the importance of other concepts. Arguably, Dourish's most noteworthy contribution to the critical discussion on algorithms is to examine the "systems of relation that give them meaning and animate them" (ibid), acknowledging that there is a multitude of colloquial ways that the term 'algorithm' is used, resulting in subjective meanings given to the term.

In seeking to clarify relations between 'algorithm' and 'code', Dourish fosters the concept of a singular 'algorithm'. Arguably, discussion of a singular algorithm supplants

discussion around the presence of a multitude of interrelated algorithms. Dourish maintains the notion that the 'algorithm' remains a singular unit, and that its presence in software becomes diffuse and fragmented as "snippets of code distributed through a large program". The defence of this notion requires taking the view that other parts of a software program are not based on algorithms, but are other subsidiary processes. Dourish provides examples of such processes, listing those that "read files from disks", "connect to network servers", "check for error conditions", "respond to a user interrupting a process", "flash signals on the screen", "shuffle data between different storage units", "record progress in log files", "check for the size of a screen", and "free space on a disk". But, of course, all these processes invoke algorithms.

Let's take, for instance, Dourish's example of a process that "shuffles data between different storage units" (p. 4). Here, the software would call to action the operating system's algorithms that read and write data to memory. If this transfer of data occurs between networked computers, requiring the software to "connect to network servers", this subsidiary process would most likely protect that data transfer using an encryption algorithm such as an RSA algorithm.[53] Dourish's description of "subsidiary processes" itself suggests that asynchronous software tasks are being carried out in parallel to the main task, and in practice these are often scheduled with a Unix-system *cron* timer, which is based on the Franta-Maly Event List Manager algorithm (Franta and Maly 1977).

Dourish's view of the complex-yet-singular 'algorithm' can be challenged with an alternative framing in which multiple, often simple algorithms are behind the complex algorithmic operations in question. Rather than articulating that an algorithm is "snippets of code distributed through a large program", software can be understood as *arrangements and interconnections between numerous software elements that implement algorithms*, a framing that I will continue to develop. Not only are algorithms operating in relation to one another within a software application, they operate at different levels of computation – from the circuitry up to the operating system, and sometimes across computer-networks (Feynman, Hey, and Allen 1998). Algorithms are not only multiple but also invoke algorithms themselves, triggering lower levels of computation. Other scholars from the CAS community have also noted that algorithms are possibly contingent on other algorithms (Kitchin 2014: 15, Seaver 2014).

[53] RSA is an algorithm widely used for secure data transmission. It can play a crucial role in maintaining the security (or exposing a weakness) of the whole body of software.

Conceptualising composite arrangements

Since Dourish's paper, a turn towards understanding algorithms within wider ensembles of activities and practices is being advanced by several proponents of the CAS community. In this section, I discuss two particular CAS texts (Seaver 2017; Mackenzie 2018), and follow with an alternative framing by the computer science theorists that developed the notion of *software architecture*. I then review other theoretical concepts that propose frameworks for understanding heterogenous arrangements of parts more broadly.

Much of the CAS literature focuses on case studies related to large tech companies that operate internet-based software-driven services, such as Google, Facebook, Twitter, and Spotify. The CAS literature often places an emphasis on 'algorithms' as the core technology behind these Web-based services. However, it is now generally understood that these services are contingent upon a range of other components and practices. The CAS literature provides a survey of studies into these complex arrangements, and its understandings of algorithms have considerably branched off from computer science literature on algorithms. Sometimes, however, their concepts refer back to computer science explanations to support their arguments; for instance, Dourish (2016) cites Niklaus Wirth (1978), Seaver (2013) cites Cormen (1990), and Gillespie (2014) cites Winner (1978).

Seaver (2017) provocatively suggests that CAS scholars have perhaps embraced a critical study of algorithms without sufficient attention to the discrepancies between the computer science descriptions of algorithms and the CAS community's interest in the more unclear "mega-algorithms":

> "Where an algorithm like bubble sort was so simple it could be described in a sentence, the charismatic mega-algorithms that had caught scholars' attention—Google's search, Facebook's newsfeed, New York City's predictive policing system, Netflix's recommender, and so on—were more complicated and harder to specify. Although these systems were called 'algorithms' by the public, academics, and even by their makers, the distance between them and their Computer Science 101 counterparts was unsettling. Maybe, in our inexpert enthusiasm, critical algorithm studies scholars had made a mistake. Maybe we don't understand what algorithms are, after all."

In addition, Seaver identifies a distancing from the notion of algorithms in the practices often labelled as 'algorithmic'. Seaver's ethnographic study took place at a company that runs a music-recommendation service (Seaver 2017). Their employees, when asked to locate the 'algorithms', typically "balked at the question" and could not pinpoint the 'algorithms' within their own technical and nontechnical roles in the company (p. 3). Even the more senior programming practitioners had moved on from "Computer Science 101".

Seaver, admitting to a degree that the CAS concept of algorithms does not represent the practices they critique, argues that the term 'algorithm' could instead be performatively given new meaning and suggests that ethnographers could "enact [algorithms] as rangy sociotechnical systems constituted by human practices". As such, Seaver's own use of 'algorithm' morphs into a notion of a system comprised of the company's software, practices, employees, and so on. Seaver's version of 'algorithm' attempts to solve the "terminological anxiety" he identifies in CAS literature; however, the term remains ambiguous and generative of various interpretations.

Adrian Mackenzie (2018), in his analysis of similar tech companies notes that composite arrangements comprising "machines, signs, people, platforms, tools, processes of invention, practices of work, technical, geographical, social, and energetic materialities" are difficult to "analytically conceptualise" (p. 49). Mackenzie considers the various possible frameworks proposed by scholars, such as *infrastructure* (Star and Ruhleder 1996), *sociotechnological project* (Latour 1996), *global assemblage* (Ong and Collier 2005), *interface* (Marres and Gerlitz 2015), and *stack* (Bratton 2016). Mackenzie adopts Lucy Suchman's (2007) notion of *configuration*, which assists in his analysis of both the composition of platform services and the narratives that influence their realisation.

I shall return to Suchman's notion of configuration later on in this chapter, as it usefully frames technologies as materialisations of thinking and storytelling. Furthermore, several of these frameworks, as well as Suchman's sociomaterial assemblages (interwoven into her notion of configuration), make reference to Deleuze and Guattari's (1987) notion of assemblage, which I also pick up again later on.

Mackenzie's understanding of platforms as configurations describes compositions that very broadly range from "technical details, such as the specific key shortcuts a software developer sets up in their code editor, to the broad architecture of data-flows between

database, webserver, and mobile messaging system in a high-volume social media platform" (p. 38). This very inclusive notion of configuration, I argue, becomes imprecise and difficult to use for the analysis of platforms (and specifics about their software processes). How can a shortcut in a code editor (such as 'Ctrl+/', widely used to 'comment out' a line of code) be described in the same arrangement as a network of webservers? Arguably, this conceptual abstraction is a little imprecise for a detailed analysis of such heterogenous ensembles.

Both Seaver and Mackenzie have attempted, with different approaches, to conceptualise computational technologies as complex arrangements of elements that include software, servers, computer practices and engineers. Although my research is concerned with all these things (a community of practice, artbots and their constituent software, servers and datasets), bundling the full constellation under an umbrella term does not facilitate a nuanced examination and discussion about these constitutive elements. However, detailed abstractions can aid both thinking and practice. Over the next two sections, I begin to construct an approach to understanding these computational services with a different set of references and conceptual tools by building on the frameworks and abstractions used in contemporary computing practices. This, in turn, helps to bridge the study and the practice of computational arrangements.

Software architecture

> "The World Wide Web is arguably the world's largest distributed [software] application." (Fielding and Taylor 2002)

Software architecture understands software applications as composed of multiple interoperable parts and distributed across computer networks. There are different models of understanding software architecture, and I shall discuss both Perry and Wolf's model (1992) and the modified version of this, conceived by Roy Fielding (2000). These notions are important because not only do they provide a useful framework for understanding the 'platforms' and 'algorithmic services' of today, but they arguably laid down the foundational ideas for their implementation.

In *Foundations of Software Architecture* (1992), Perry and Wolf boldly claimed that the 1990s would be "the decade of software architecture" (p. 1). They opted for the analogy to architecture to adopt a whole set of metaphors from the field of architecture, such as 'materials', 'styles' and 'architects'. Perry and Wolf created a framework of abstraction

that understood complex software structures as composed of *processing elements* that are interconnected by *connecting elements;* these, respectively, would process and supply *data elements*:

> "*A software architecture is defined by a configuration of architectural elements – components, connectors, and data – constrained in their relationships in order to achieve a desired set of architectural properties."* (Fielding 2000: 7)

Fielding, an influential practitioner who implemented his ideas into the protocols of the Web while working with Tim Berners-Lee as part of the World Wide Web Consortium (W3C) at MIT, improved on the foundations proposed by Perry and Wolf. Fielding worked on the standards for URLs and HTML, and became the lead editor for HTTP (Severance 2015: 8). Fielding proposed the terminology of *components*, *connectors*, and *data*. He defined a *component* as "an abstract unit of software instructions and internal state that provides a transformation of data via its interface", a *connector* as "an abstract mechanism that mediates communication, coordination, or cooperation among components", and a *datum* as "an element of information that is transferred from a component, or received by a component, via a connector".

Fielding not only perceived components as a heuristic abstraction, but sought to encourage the creation of these components in practice. A component, as Fielding saw it, would be an isolatable piece of software that would have an interface to connect it to other pieces of software. Self-contained processing components would provide an *application programming interface*, or API. A processing component would receive data and send it back after having carried out a transformation to that data, such as performing a calculation or translating data into a different format. Processing components are where underlying algorithms may be implemented, modified, and encapsulated into a service for external, multi-part applications to interface with. As such, it is foreseeable under this model that architects may not engage with algorithm design directly, but interface with their ready-made implementation towards the construction of a computational edifice of many interconnected components. Of course, other engineers would work on developing these API-fronted components and be responsible for their inner workings, but may do so unaware of possible implementations by others.

Fielding's model became a seminal influence on how Web-based software applications were to be designed. In 1999, Web-based applications scaled from 100,000 HTTP

requests per day to 600 million (Fielding and Taylor 2002: 147). Web companies began providing online services that operated more like software applications. Components with APIs were increasingly created and offered for third-party use. Flickr, launched in 2004, was one of the first software architectures built on Web 2.0 principles – an online photo-sharing application with social networking functionality that registered two million users in its first year. Flickr also pioneered the practice of offering an API that allowed other architectures to interface with it. Flickr's API made their users' data and media assets available through the API, pioneering a practice that has since become questionable. Facebook in particular has become embroiled in controversy over the third-party misuse of user information: data utilised to target users with misinformation and political propaganda (Hern 2017).

Fielding, along with Perry and Wolf, all express interest in the considerations and motives that inform software architecture. Fielding maintains that the complexity of modern software systems necessitates a greater emphasis on component-led methods so that elements can evolve independently. Perry and Wolf describe the need for architectures to be adaptable, modifiable, and allow for reuse. Moreover, in both accounts, the need for systems to scale up seems imperative. Perry and Wolf believe that software architecture criteria are set by economics, performance, and reliability (p. 45), and give it the term *rationale.* Fielding critiques Perry and Wolf's inclusion of 'rationale' into their model, arguing that "rationale itself is not part of the architecture" (p. 8). Fielding's argument is that an architecture may be informed by a rationale for its inception but is not bound to it, and may evolve independently from its original intentions. Fielding's model cautiously avoids setting in stone that software architecture is informed by a rationale, but in an enlightening interview (Severance 2015), Fielding revealed that new commercial pressures informed his model:

> "As companies became involved, they wanted to find ways to use the Web corporately as one of their platforms, so it needed to be more 'business-like.' One of the ways to make things more suitable for business is to create common standards for everyone to adhere to, rather than to adopt things as you go along." (p. 8)

Fielding's model was implemented into the very fabric of the internet and fulfilled the criteria of a more "business-like" Web. Moreover, Fielding's research and implementations spanned the rise and fall of the 'dotcom bubble', and arguably responds to the problems that companies faced during this formative period.

Fielding's "architectural properties of key interest" help articulate other properties and notions of software architecture (as well as revealing what Fielding was interested in). Fielding lists "performance", "efficiency", "reliability", "scalability", "simplicity", and "modifiability" (which itself is further broken down into concepts of "evolvability", "extensibility", "customisability", "configurability", and "reusability"). These various terms, seen as desirable qualities, illustrate that Fielding envisions software applications as evolving and transformative, rather than fixed. Much of software architecture today is indeed in a state of flux; many services, such as Google's search engine, are regularly modified, extended, and reconfigured without disrupting the service. Moreover, many internet-based software applications of the 2000s have successfully benefitted from the scalability and extensibility of the software architecture principles. Facebook, YouTube, Twitter, and others have grown to unprecedented scales, even though they may have begun as modest configurations of computer architecture. The term *platform* is now attributed to these, although in its technical understanding, a 'platform' refers to the parameters (such as hardware and devices) for which a software application is built.

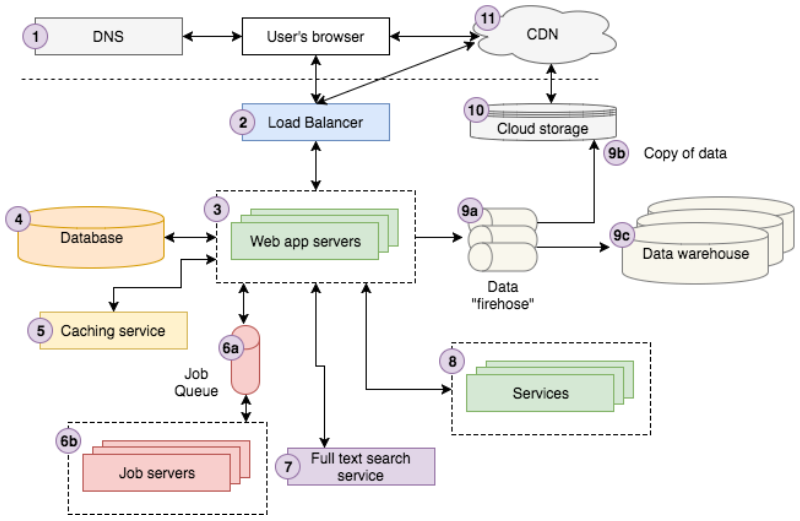


Figure 1. Computer architecture diagram for *Storyblocks*, a Web-based service for searching for licensable 'stock' photographs.[54]

Although it may be difficult to gain access to the software architecture maps for the major platforms, their lesser-known competitors can provide insight into how contemporary software architectures manifest in practice. Figure 1 depicts the architectural plan for *Storyblocks*, a platform that allows photographers to sell their

[54] Source: *https://engineering.videoblocks.com/web-architecture-101-a3224e126947*

'stock' images, providing advanced search facilities and algorithmic curation for others to find and purchase an image (Fulton 2017). Furthermore, the diagram illustrates how various software components, connectors, and data storage elements are connected and interoperate. The service is distributed across various computer servers, and users even interface a randomly selected software application server to distribute search traffic (see Component 3 of Figure 1). Fulton (ibid) describes that the architecture is designed for "horizontal scaling", meaning that the configuration is set up so that more servers can be added if necessary. The software application servers are utilised to "execute the core business logic that handles a user's request", and this manifests as the HTML content that the server sends to the user's browser. To do this job, the servers enact various processes across the architectural arrangement:

> "[The Web application servers] typically communicate with a variety of backend infrastructure such as databases, caching layers, job queues, search services, other microservices, data/logging queues, and more." (ibid)

This example helps illustrate how the notion of computer architecture not only provides a theoretical abstraction but informs practice – and as such, can be used to describe practice. This notion, however, can be extended even further.

First, I propose that the term *edifice* can also be used to describe a composition of software architecture. 'Edifice' is an apt term that suggests both a material structure and a set of imbued and instilled values – the *rationale* – behind the architecture. Using *edifice*, I can refer explicitly to *a connected set of concrete software elements and the thinking and contexts that dictate configurational choices*. An edifice suggests a closer relation between software architects and architecture, rather than assuming that these software applications run independently of their creators.

Second, software architecture could be criticised for playing down human activity tangled in these systems. People do not only interact with these systems as 'users', but also provide much of the data and media content that populate the architecture's databases. Furthermore, many platforms are created for specific communities of practice, and often attempt to nurture and mediate human networks. In the following section, an understanding of *sociocomputational assemblages* is developed to address these issues.

Sociocomputational assemblages

How is human sociality viewed through the lens of software architecture? Software architecture can understate, or purposefully define, human activity into a homogenous component described as 'users'.[55] However, a software system's human activity is not simply page clicks and eyes looking at browsers, but a complex arrangement of practices, politics, and every-day life. As more and more human activity is transformed by software edifices, the study of how human activity is interrelated with computational activity becomes ever more urgent. The field of *human-computer interaction* (HCI) has provided ample literature and research into the design and use of interfaces between human and computational devices (Zimmerman, Forlizzi, and Evenson 2007). Furthermore, a vast number of social studies have examined specific human practices that are now mediated through algorithmic systems, including but not limited to financial trading (MacKenzie 2005, 2014), dating (Hitsch, Hortaçsu, and Ariely 2005), encyclopaedia editing (Geiger 2012), governance (Introna 2015), and journalism (Anderson 2013).

Several concepts have been proposed towards a theoretical framing of the interrelation of the social and the technological. Scholars of science and technology studies (STS) have made particular contributions to this matter, such as Latour's *actor-network theory* (Latour 1996) and Donald Mackenzie's and J. Wajcman's *social shaping of technology* (MacKenzie and Wajcman 1999). More recent, and more attuned to human-computational interaction, yet challenging its core understandings, is Lucy Suchman's notion of *sociomaterial assemblages*. Sociomaterial assemblage cannot be entirely attributed to Suchman, as it builds on the notion of *assemblage* (Deleuze and Guattari 1987), and takes references from Feminist Studies (Haraway 1996; Barad 2003; Butler 2011), but it is through Suchman's work that the concept becomes of relevance to this study of artbots. In *Human-Machine Reconfigurations* (2007) – an updated edition of the influential *Plans and Situated Actions* (1987) – Suchman argues that a reconceptualization is needed to move away from a framing that sustains "categorical purification" between the social and the technical, person and artefact (p $$) –referring to the challenges of a 'human-computer' dichotomy. Further to this, Suchman reveals the asymmetries between the figuration and status of 'designers' in contrast to 'users', as well as potentially marginalised engineers, product-testers and other actors, in order to maintain the machine-user dualism. As such, a framing is needed that moves beyond the

[55] For a detailed study on this process of figuring the 'user', in tech companies, see (Wilkie 2010).

singular user, and can account for multiple, interrelated actors. Suchman turns towards a new framework that:

> "at once locates the particular accountabilities of human actors, while recognising their inseparability from the sociomaterial networks through which they are constituted. [Additionally,] the cyborg metaphor dissolves into a field of complex sociomaterial assemblages, currently under study within the social and computing sciences. From close readings of encounters at the interface of person and machine, through extended historical and comparative analyses of technology-in-tensive, distributed worksites, these reconceptualisations have opened a generative wave of new scholarship and practice." (2007: 286)

This generative wave of new scholarship includes the work of Wanda Orlikowski and Susan Scott (Orlikowski 2007; Orlikowski and Scott 2008) whom take up the notion to study more specifically, the sociomaterial assemblages of work places, arguing that ensembles of equipment, techniques, devices and software, are intrinsic to everyday workplace activities (p. 445). It is important to note that the notion of the sociomaterial assemblage shifts between providing an ontological framework and an epistemological framework. Barad's material-discursive *agential-realist ontology* builds upon the epistemological framework of physicist Niels Bohr, that rejects the representationalist divide between 'words' and 'things' (Barad 2003: 814). Barad questions the basis of knowledge as being constituted by separable and representational discursive practices in contrast to the world of matter, of things, that they describe. Instead, Barad proposes that these intervene in one another, in casual relationships that Barad calls "agential intra-actions" (ibid). Barad's "ontoepistemology" argues that knowing is as a material practice that intervenes in the world it describes (p. 89). This is not dissimilar from Deleuze and Guattari's argument that semiotics, representation and matter are interconnected elements of an assemblage, in which

> "there is no longer a tripartite division between a field of reality (the world) and a field of representation (the book) and a field of subjectivity (the author)" (Deleuze and Guattari 1987: 87).

Suchman, taking both Barad's agential ontology, and Deleuze and Guattari's assemblage, constructs the notion of sociomaterial assemblage, that places more emphasis and analysis on questioning the ontological separation of humans and technology, but also questions the discursive-thinking practices that engage and

construct that separation. In other words, it matters who decides where the boundaries lie between humans and technologies. This 'cut' is of concern to Suchman who argues that this process of *figuration* (the topic of chapter 6) is not only attributed to academic discourse, but also to the designers that produce these distinctions in practice (Suchman 2007: p $$ 80%). Suchman's analysis invites both scholars and practitioners of sociomaterial assemblages to question how we can "configure assemblages in such a way that we can intra-act responsibly and generatively with and through them" (ibid).

This framework provides a more critical and rigorous reflection of Fielding's Software Architecture, as through this lens, Fielding's own architectural structure is one of many ways in which human-software application relations can be figured and configured. The sociomaterial assemblages described by Fielding in his thesis problematically encapsulate users as a homogenised component within the architecture. Users, in this sense, are figured as a resource to the wider system – returning data, clicks, and in some cases, processing capabilities. The architectural designs of Fielding however, are shaped by his own design criteria and circumstances, and are not the sole way in which sociomaterial assemblages concerning architecture can be configured. By framing software architecture, and its multiple and inseparable human actors in interrelation (including designers and engineers of the architecture), as a type of sociomaterial assemblage, I have developed my own notion of the *sociocomputational assemblage*.

A sociocomputational assemblage is a configuration of a software architecture and its interrelated human and nonhuman elements, in which properties and capacities emerge through sociocomputational intra-action. Assemblage thinking is useful in this context because it not only understands things as constellations of constituent parts, but as undergoing processes – both human and nonhuman in interrelation – 'plateauing' into stable arrangements, and open to possible transformations that destabilise and alter the assemblage. When applied to the study of software architecture, in which configurations continuously scale and evolve, especially in the current paradigm of technical interoperability and demand for growth, the notion of sociocomputational assemblage provides an analytical frame that anticipates these shifts.

In summary, Suchman's work on the sociomaterial assemblage can be understood as a coming-together of discourses and frameworks, attuning its focus to the ontological-epistemological figuration and configuration, of human (socio-) and technological (material) arrangements, in both theory and practice. One implication of this, is that through this understanding, sociomaterial assemblages are not seen as fixed, nor merely

identified through theoretical analysis, but can be prescribed, designed, and shaped by sociomaterial practices. Assemblages more broadly, are understood as dynamic arrangements of heterogenous components, both material and figural, human and nonhuman, in continuous interrelation. Applied to human-computer ensembles, this framework avoids establishing a clear boundary between human components and nonhuman components, but also clarifies that the ontological boundary is regularly being reconstructed and redefined by sociocomputational practices. I elaborate on this further in chapters 5 and 6, through artbot case studies including my own works.

Artbot practices can be understood as enactive of sociocomputational assemblages, finding alternative ways of configuring and entangling people within software edifices. Artbots are configured arrangements of software architecture that in a sense are 'social' – their configurations inter-relate with humans and their online activity, communication, and culture. To do so, they often engage people through interfacing with a social media platform. Moreover, the properties and capacities of artbots are not necessarily pre-defined but emerge through social-computational intra-activity. Artbot practice can be described as a sociocomputational practice. In the next section, I introduce the sociocomputational practice of artbots, and apply some of the above concepts and thinking to frame artbots as sociocomputational assemblages. I then branch out this framework to a set of analytical focal points to further examine and understand artbot practices in the coming chapters.

Artbots

Configurations of servers, databases, distributed software elements, and human participants can scale up towards huge Web services for commercial gains and mass networking. But they can also remain small, experimental, absurd, subversive, and characterful – and yet interoperable with larger systems. A small number of practitioners are radically exploring these new resources and approaches, 'mashing up' the interoperable components of Web 2.0 towards web-software curios that plug into larger platforms, often with the intention of participating in the social sphere that these platforms mediate – a sociocomputational practice often described as *artbots*.

A bot is sometimes described as "a software application that runs tasks over the internet".[56] As explained in the section on software architecture, software applications are comprised of multiple and distributed internet-based software components. These can scale up considerably and interface with millions of users. At a certain scale, these begin to be described colloquially as 'platforms'. Bots can be described *also* as computer architecture, comprised of interconnecting software elements and performing computational tasks over the internet. However, bots are customarily smaller and less complex than their social media platform counterparts, and often interoperate with a larger service by interfacing with their API. Using the architectural analogy, one can imagine that a social media platform is like an indoor market building, and a third-party application may be normatively considered a food stall housed inside of it. An artbot is often a more subversive version of this, an architectural intervention that both nestles within the host architecture but finds alternative uses as well, like setting up a community greenhouse rather than a food van.

As internet-based software configurations, bots run from networked servers and usually interface with the API of the social media platform they engage with.[57] These configurations can interface with a number of other APIs and software components. APIs exist for all kinds of services, such as dictionaries, maps, image processing, language processing, translation, machine learning, and so on. Moreover, the bot's architect can make use of many data repositories and media archives. The practice of artbots has emerged partly out of the experimental re-use or re-configuration of these software components. Artbot practitioners make-with a vast range of tools, and find new capacities through their combination. Artbot artists generally avoid clichéd and overused bot narratives (for instance, the human-like Turing test contender), and instead work towards new forms of character, agency, behaviour, and purpose.

The artist Allison Parrish, whose work is presented later in this thesis, makes a comparison with skateboarding, citing Iain Borden's study of architecture through skateboarding:

> "Skateboarders target functional everyday spaces and objects. [...] For example, a handrail is a highly functional object; both the time and nature of its use are

[56] Wikipedia's definition suggests this: *https://en.wikipedia.org/wiki/Internet_bot*

[57] There is another method: bots can automate a browser using dedicated browser-automation software.

> fully programmed. [...] The surprise of the skateboarder's reuse of the handrail – ollie-ing up onto the rail, and sliding down its length sideways – turns it into an object of risk. […] The whole logic of the handrail is turned on its head." (Borden et al. 2001)

In this scenario presented by Borden, a skateboarder approaches existing everyday spaces and finds its constituent elements – handrails, staircases, ramps, ledges – to be reconfigurable in a playful manner, putting these to use towards the production of skateboarding tricks and reconfiguring architecture for skateboarding. This practice of rearranging pre-existing software components towards experimental edifices subverts the more authoritarian edifices that they intervene in. Parrish argues that her work explores the multiple ways in which Twitter can be used. The important thing to note is that artbot practitioners are finding their own set of motives and rationale for creating software edifices. Rather than see their practices as responding to a given set of constraints and obligations set by other actors, they actively find new aesthetics, contexts, and impetus for configurations. In other words, their own artistic voice is expressed through their software creations. The artist Mark Sample, for example, has a personal manifesto to create protest bots that "reveal the injustice and inequality of the world and imagines alternatives" (Sample 2015).

The practice of artbots, however, is far from understood. Even their own proponents, who have tacitly developed the artform, often find difficulty in describing it. In an interview with artist Chris Rodley, a question about this type of practice illustrated this point:

> "I'm actually not sure that the words exist yet for this type of practice, which is why myself and other bot artists sometimes use vague self-descriptions like 'I make stuff on the Internet.'"[58]

As the quote above illustrates, artbot artists have difficulty articulating their practice. An advantage to framing this type of practice as experimental sociocomputational assemblages is that this understanding identifies a social media bot as one of a multiplicity of possible configurations. The artist Darius Kazemi, for example, not only makes bots on social media, but also what he calls "generators" – essentially webpage-

[58] See the full interview with Chris Rodley for Art of Bots: *https://www.andfestival.org.uk/blog/introducing-chris-rodley-the-art-of-bots/*.

fronted software architectures.[59] As such, I use the term 'artbot' to stand in for a number of ways in which artists choose to describe and characterise arrangements of software architecture, not only as 'bots'. I also gradually introduce the term *configurative art* (playing on the term *figurative art*), which I describe in Chapter 6 on figuration. I use the term 'configurative art' to speculate about what artbot practices may evolve into; a practice predicated on not just bots, but a wider range of software architectures and figurations. Recommendations for future practice are further discussed in the thesis conclusion.

With this widening glossary of terms, thinking, and understanding, this study can begin to frame better questions and examinations into the study of artbots. Over the next sections, I begin to construct a nuanced framework for the study, and return to some of the theoretical influences that have been introduced so far.

Towards a framework

So far, this chapter has developed an understanding of complex arrangements of software architecture and people as *sociocomputational assemblages*. This notion begins to provide the language for examining, analysing, and discussing case studies of artbots. This study aims to create an analytical framework to further understand and discuss this particular sociocomputational practice, and in this section, I propose key areas of analysis.

The notion of sociocomputational assemblages provides a conceptual framework for understanding artbots, however, more nuances to this art practice are identifiable in their works and methods, and a set of analytical viewpoints are required to draw out these from observation. Moreover, one cannot simply reduce a set to practices to a 'formula' or 'criteria'. Instead, a good framework can more openly work as a set of questions for examination, for example, enquiring into what these practices produce, enact, reveal, and convey. A useful framework provides tools for possible readings of the work rather than conditions for the work to meet.

Assemblage thinking helps in this task, as it is regarded as a conceptual toolset for analysis and intended to be adapted and put into use. Deleuze and Guattari (1987: 22)

[59] See Kazemi's list of projects for 2013: *http://tinysubversions.com/2013/12/stuff-i-made-in-2013/index.html*.

maintain that assemblages act on "on semiotic flows, material flows, and social flows simultaneously". Assemblage thinking provides these three separate strands, and at the same time, imply the inseparability and interrelation of these. Deleuze and Guattari maintain that their ideas purposefully move beyond conceptual framings that separated the material, social, and semiotic. These three 'flows' suggest three facets of assemblages for analysis. When applied to artbot practices, these aspects of assemblage suggest questions such as: what are the 'material' elements of the configuration? How is the assemblage enacting human and nonhuman 'sociality'? And what 'semiotics' emerge, or have been specified, through the assemblage?

These three matters of assemblage thinking have parallels with Suchman's (2010) tripartite usage of the term *reconfiguration*, which she applies as an analytical framework for specifically understanding case studies of human-computer (sociomaterial) design practices. Suchman points out that the words *figuration* and *configuration* are opportunely nested within *reconfiguration*, and uses the three terms to denote various parts of her framework. Suchman's argument is that "material practices" are informed by "cultural imaginaries" – that stories, narratives, metaphors, and "possibilities for thinking" are materialised through practice (ibid). Firstly, 'figuration' (or the 'figural') is at the core of Suchman's framework, questioning "what are the metaphors that are available" and "how are these concretised or realised?" Secondly, through the notion of *con*figuration, Suchman asks how humans and nonhumans are configured in relation to each other. Thirdly, through *re*configuration, that enquiry is extended to ask how sociomaterial assemblages may be reconfigured in alternative ways.

Suchman's framework is generative of many useful questions, and produces examinations into the semiotic, social, and material elements of sociocomputational assemblages. It predominantly emphasises the question of how technologies currently (and alternatively) are informed and materialised by cultural imaginaries. These matters of figuration help examine artbots, as they are often imbued with character that have been 'figured' by their authors. These bot figures and the practices of figuring them are an important element of this study. Moreover, they are constitutive of the meanings and semiotics that flow through an assemblage. Deleuze and Guattari (2014) have a similar concept of *conceptual personae*, referring to the embodiment of philosophical discourses. The aptness of figuration for examining artbot practice and assemblage semiotics makes it the first aspect of my three-part toolset for analysing sociocomputational assemblages. I elaborate on it further in a follow-up section.

As illustrated beforehand, a defining element of sociocomputational assemblages is the interrelation of human and computational activity. Human and computational elements become-with one another, in co-constitution of the outputs and activities produced by their interlocking mechanics. Software architecture traditionally, would take the predetermined view that 'users' are distinct from computational processes and interact with the architecture. Such user-computer frameworks are being challenged and re-figured by more recent sociomaterial theory and practice. Artbot practitioners, I argue, explore novel arrangements of computational sociality. Their explorations span a wide gamut of interactive and intra-active ensembles, rather than fixate on a particular approach. As such, I appropriate the term *entanglements*, from Barad, to describe human-nonhuman sociocomputational relations in artbot practice – both the more conservatively 'interactive' and more radically 'intra-active' relational set-ups. I must stress however, that Barad's own use of the term entanglement specifies the material entanglements of knowledge practices – that "practices of knowing are specific material engagements that participate in (re)configuring the world" (p91). My use of entanglement does not exclude Barad's use of the term, as indeed, creative practices such as artbots are not just about making things but developing ideas and implementing them in physical form. With this in mind, the term entanglement also conveys the practitioner's own entanglement with their material creations and inseparability from the sociocomputational assemblages they configure and intra-act with. To exclude the artbot puppeteer from the puppet, in our analysis of artbots, would neglectfully play-down an important element of their sociocomputational constitution.

Suchman's framework brings up a discussion about the materialisation of technology, but also misses the opportunity to ask more questions about this process – how does technology materialise? Is this materialisation instantly realised and stable? Or rather, is the development of technology a dynamic and ongoing process that evolves and scales? Fielding's thesis (2000) proposes that software architectures are normatively designed for 'scalability', 'modifiability', 'evolvability', 'extensibility', 'customisability', and 'reusability'. Moreover, assemblage thinking provides an understanding of 'materialisation' as a coming-together of multiple elements, and such arrangements are open to a 'multiplicity' of possibilities. *Multiplicity* is a useful term that suggests *the inherent heterogeneity, variability, and dynamics of an assemblage, as well as its ability to expand and be generative of further transformations and enactments*. This third analytical viewpoint is explained further in the next section, followed by *entanglement* and *figuration*.

Multiplicity

As one of my three analytical viewpoints for understanding artbot practice, I adopt the term *multiplicity* as a tool for examining the multiplicative arrangement of artbots. This section briefly discusses this applied notion of multiplicity, as well as the influences that have contributed to my understanding of it. The framework and analytical tools developed in this chapter support the artbot case studies and examinations in subsequent chapters of this thesis.

Multiplicity is a notion from Deleuze and Guattari (1987) that was developed to understand that an assemblage is constitutive of multiple interrelated elements, and yet can be treated as a more or less stable entity that can be configured and reconfigured or expanded and decreased in magnitude. The authors maintain that it escapes "the abstract opposition between the multiple and the one, to escape dialectics, to succeed in conceiving the multiple in the pure state […] and distinguish between different types of multiplicity" (p. 53). Multiplicity reminds us that assemblages are heterogenous, dynamic, reconfigurable, scalable, and productive. Moreover, Deleuze and Guattari explain that there is a direct link between multiplicity and *becoming* – that an assemblage manifests through the process of composition. Haraway's (2016) related phrase "becoming-with" is a useful, self-explanatory variation on 'becoming'. Another useful Haraway term is "making-with", which suggests the *practice* of configuring multiplicities – of bringing elements together in creative ways.

I argue that multiplicity can be used to describe artbot practices as these seek to bring together – to *make-with* – modular resources of software architecture that are put into inter-operation. Fielding (2000) argues for the configurability, evolvability, and reusability of software architecture. Artbots can be understood as compositions of software components (through APIs), data elements, media resources, and human activity, configured to operate as a sociocomputational assemblage of these.

Deleuze and Guattari (1987: 249) maintain that "multiplicities continually transform themselves into each other, cross over into each other." Rather than treat artbots as fixed entities, these can be considered as more or less stable arrangements that may change during operation. During operation, the architect may decide to change its code, provide new behaviours (or new restrictions to its behaviour), or even migrate it to another platform. In 2018, a number of artists decided to migrate their works from Twitter to Mastodon (an alternative, open-source networking platform).

More important, however, is the understanding that the artist is not the only element influencing the multiplicative transformations that an artbot undergoes, but rather that all elements are contingently affecting the overall composition of the artbot. No example is clearer than the fact that the protocols of the Twitter API and terms of use are regularly being updated by Twitter engineers, which routinely influence, limit, and sometimes even revoke the performance of a Twitter bot. An extract from A Thousand Plateaus (1987: 249) pleasingly suggests this 'many hands' scenario, using the metaphor of a puppet attached to multiple strings in tandem:

> "Puppet strings, as a rhizome or multiplicity, are tied not to the supposed will of an artist or puppeteer but to a multiplicity of nerve fibres, which form another puppet in other dimensions connected to the first […] An assemblage is precisely this increase in the dimensions of a multiplicity that necessarily changes in nature as it expands its connections."

Further to describing the becoming-with of a composition and its open-ended reconfigurability, a third understanding is intrinsic to the notion of multiplicity: *scalability*. Assemblages are understood to possibly expand (and contract), multiplying in size throughout their operation. This property can also be understood as either as about the expansion of its composition, or a temporal expansion – enduring time is a form of multiplicity, what Deleuze and Guattari (1987: 53) refer to as 'durational multiplicities'.

Further to this, *scalability* is one of the concepts proposed by Roy Fielding (2000: 32), referring to a software architecture's ability to "support large numbers of components, or interactions among components, within an active configuration." Fielding's remark helps understand how the theoretical notion of multiplicity can be observed within empirical practices of computational system design. Fielding's note also suggests that the uptake of software modularity is partly designed with scalability in mind, which, in turn, has allowed for the scalability of configurations made-with these elements.

Artbots, I argue, are scalable sociocomputational assemblages that scale up in the number of interactions between components and over long durations of their operation (some artbots have been running for years). As part of my framework for understanding artbots, multiplicity invites us to be attuned to how long an artbot has been running for, and how many engagements, interactions, and materials have been generated during that

operation. It must be understood that during operation, a functioning sociocomputational assemblage would invariably produce data, media, and interrelated computational and human activity. My framework understands artbot configurations to be scalable in both a durational and generative sense.

Entanglement

At the very core of my set of conceptual tools for understanding artbot practice is the notion of sociocomputational assemblages. The argument, as described earlier, is that computational configurations are inherently in constitutive entanglement with human activity, including that of the artbot practitioner. A multitude of other human actors may be entangled within a configuration, such as modular software developers, platform engineers, unsuspecting others providing data as resource, and the audiences of these artbots. This notion challenges a dichotomous thinking that understands humans and machines as ontologically separate and only meeting at points of planned interaction.

Understanding artbots as sociocomputational assemblages or operating as agents interfacing larger software architectures attunes us to examine and analyse possible entanglements of human activity, both explicit and obscured. Entanglement here, is adapted from Barad (2007) who argues for a "posthuman" understanding of entities that do not pre-exist but "emerge through and as part of their entangled intra-relating". Some artbots exemplify this clearly. For example, Rob Dubbin's Twitter bot *Olivia Taters* (2013) is a sociocomputational assemblage configured by the artist to computationally mangle the tweets of unnamed others, re-figured as a bot. Utterances are produced by concatenating tweets found to contain adverbs such as "literally" and "finally", commonly expressed by teenagers, to construct phrases such as, "if you guys would actually mean something".[60] *Olivia Taters* emerges through the entangled intra-activity of unsuspecting teens on Twitter and Dubbin's software architecture.

The politics of entanglement are more pronounced and contentious in situations in which constituent human activity is purposefully concealed to create the illusion that a computational agent is entirely machinic. Astra Taylor (2018) calls this *fauxtomation*, a term given to projects of computational and robotic automation that mask the human processes required to support these systems. Through the framework of sociocomputational assemblages it can be appreciated that no software or hardware

[60] A tweet from 16/12/18, see *twitter.com/oliviataters/status/1070435397809783380 9=20.*

system is free from the entangled human activity that figured and implemented their materialisation. Through this lens, sociocomputational entanglements can be traced and analysed. The goal here is not to call out cases of fauxtomation, but to understand the entanglement of multiple actors and the any underlying motives to either highlight or obscure their participation.

An attribute of artbot practice is the consideration as to how an artbot is connected to people as audiences or spectators of the artbot's activity. Evidently, this practice normatively uses social media platforms as the site of engagement with others, but more unusual cases employ different tactics. Sam Lavigne's *Yelp Review Faxbot* (2015), for example, reaches prison administrators by sending them faxes. Moreover, this bot is in constitutive entanglement with people who write reviews of prisons on a user-review platform called Yelp, which the bot covertly looks out for. The bot then provides mediation between the unhappy Yelp reviewers of prisons and the prison administrators who can be reached via fax.

Artbots can be understood as an exploration of more experimental forms of entanglement. In doing so, artists challenge how sociocomputational assemblages are more conventionally configured to engage, interact, or covertly depend on humans. For example, in computer science, there are now commonly held ideas that chatbots should aspire to pass the Turing test and interact using human conversation (either text or speech-based). As such, software technologies such as natural language processing (NLP), speech synthesis, and voice recognition have been developed towards the aspiration of achieving conversational interaction. In contrast, artbots explore other forms of sociocomputational relations. For example, Kazemi's *Dolphin Town* (2018) is a social network in which people can only converse using the letter 'e', to prompt them into "squealing like dolphins" at each other ("eeeEEeee"). The artbot *God Tributes* (2014) by Jim Kang communicates with others over Twitter by interjecting in conversation with a declaration of praise for the god of *X*, where *X* is a keyword picked up from the conversation. So, for instance, if I were discussing my thesis on Twitter, I may receive a surprise response from *God Tributes* exclaiming, "THESES FOR THE THESIS GOD".

Suchman is interested in the agencies, the "capacities for action", that are "distributed across human and nonhumans" (2010). In this vein, it can be argued that *certain capacities (and thus accountabilities) of sociocomputational assemblages come into being upon their contingent human and nonhuman co-activity*. This co-activity can also

be attributed to the bot artists themselves, entangled in the sociocomputational assemblage – their artistic contributions may become part of the bot's persona or output. This is acknowledged by Suchman in her analysis of new media-based arts:

> "New media artists, their works, and the persons whom the latter engage are configured together through these [sociomaterial] assemblages." (2010: 283)

The stories produced by *Magic Realism Bot*, for example, are constructed out of fragments penned by Chris Rodley. The software simply randomises how these fragments are put together towards serendipitous, implausible, nonsensical, or even otherworldly stories. This co-authorship manifests through entanglement. Magic Realism bot is further discussed in chapter 6, and many other examples of sociocomputational entanglements are analysed in chapter 5.

Figuration

In artbot design, a common practice is to construct a fictional character to be used as the avatar and 'identity' that provides the software an outward-facing presence on networked sites of engagement. This 'characterisation', however, does more than merely provide a decorative façade; it can be seen to inscribe and dictate ideas about how the bot is meant to behave and interact with others. To discuss this element, nuanced language that implies more than simply the representational effects of characterisation is needed.

Figuration is a notion introduced by Donna Haraway (1996: 11) and further developed by Suchman that argues that technologies are "figured", imbuing them with meanings and characterisations that inform how they are designed and configured. Suchman further argues that projects of robotics and AI are commonly preoccupied with figuring technologies to be "humanlike" (providing these with attributes of embodiment, emotion, and sociality). Haraway proposes an understanding of technologies as forms of *materialised figuration* that bring meanings and materials into more or less stable assemblages, building on assemblage thinking. Haraway (1996) coincidentally discusses the use of avatars and personae in early internet culture and sociality, and introduces *Modest_Witness*, *FemaleMan*, and *OncoMouse* – speculative characters figured on feminist notions. Suchman (2007: 228) argues that alternative figurations challenge the norms, calling for "a critical consideration of how humans and machines are currently figured in those practices and how they might be figured – and *configured* – differently."

Suchman further argues that the effects of figuration are political because they can work to "re-inscribe existing social orderings" (2007: 228). As Haraway (2016) eloquently puts it, "it matters what stories make worlds, what worlds make stories."

The act of figuring can be considered an element of software edifices, for which practitioners consider what sociocomputational arrangements should be figured as, as well as why and how. Artbot practices, I argue, are tacitly engaged with creating figurations. Their art is explicitly concerned with identifying new possible figures, and arguably have achieved an expansive taxonomy of software figurations. Artbots come in a plethora of forms. For example, *TV Helper* (2015) by David Lublin is figured as a nonsensical television-viewing commentator, *TrippingBot* (2015) is a drug-experimenting diarist by Shardcore, *A Real River* (2014) by Colin Mitchell is an endless river of emojis, and *Tiny Space Poo* (2015) is, well, figured as a tiny poo floating in outer space. To demonstrate how this last example goes beyond mere narrative and characterisation, *Tiny Space Poo*'s software works by programmatically following the Twitter account of another artbot called *Tiny Star Field* and responds to any of its character-based star-field drawings by producing a replica, but with the addition of a poo emoji. The figuring of *Tiny Space Poo* dictates the way the software is made to work and behave, enacting a number of interrelated elements.

Artbot artists may be tacitly exploring how software agents can be figured differently from normative characterisations, but have any of these proponents explicitly stated that experimental figuration is part of their practice? In some cases, yes. Kazemi, for example, was originally inspired to take up bot-making as a practice in part as a response to writer Ian Bogost's "Alien Phenomenology" (2012).[61] Bogost's notion argues that philosophy is embodied in objects, and thus Kazemi seeks to make things that embody philosophical ideas. In addition, Kazemi has also acknowledged that his *Random Shopper* bot purposefully considers alternatives to e-commerce systems that are configured as 'recommendation algorithms'.

A similar notion was maintained by the designer of *Tower Bridge Bot*, Tom Armitage, who proposed his notion of "synecdoche" (a term borrowed from literature) to describe how bots can be the human-facing element of a far more complex assemblage that

[61] From Bogost's thinking, Kazemi is specifically interested in his notion of "carpentry" (Neyfakh 2014).

stands in at the "human-scale" (Armitage 2010).[62] Armitage arguably understood a bot to be an element in a wider sociocomputational assemblage rather than a representation of it (and yet standing in for the whole).

Understanding figuration as a constitutive element of an assemblage, a bringing together in relation meanings and materials, also reflects Deleuze and Guattari's assertion that assemblages are composed of both material and semiotic elements. Moreover, they propose that these intervene-in one another, influencing the overall assemblage (Deleuze and Guattari 1987: 87). This, in turn, helps us to understand how figuration is a process as well as an element, and that figuration is never fully stable, but equally influenced by other elements.

Conclusion

The goals of this chapter were: to examine the literature on algorithms and the algorithmic systems that are purportedly powered by them, review other theoretical frameworks for understanding internet-based software applications, and construct a set of references that help discuss the practice of artbots.

Software architecture was found to be an overlooked set of references and abstractions that not only help inform an understanding of complex arrangements of internet-based software applications, but relate to the practice of these through their implementation in internet protocols and widespread influence. However, software architecture can often assume that human activity is a homogenous 'user' component, rather than a rich set of practices and activity in constitutive interrelation. Suchman's thinking on configurations and sociomaterial assemblages, furthered by Orlikowski, develops a framework and terminology for understanding these complex entanglements. Applying this thinking to software architecture, I propose the notion of *sociocomputational assemblages*.

With this nuanced understanding, it becomes possible to examine artbots as small and experimental software architectures that interface and interoperate with larger platforms. By understanding artbots as sociocomputational assemblages, a framework enables questions about their multiplicity, entanglement, and figuration is developed. Over the next three chapters, I use this framework to examine and analyse a number of artbots,

[62] See Armitage's blogpost for Berg: *http://berglondon.com/blog/2010/10/01/open-data-for-the-arts-human-scale-data-and-synecdoche/*

including my own; in turn, I develop further thinking around these three aspects of artbot practice.

Chapter 3: Studying sociocomputational assemblages

This research seeks to examine and analyse the practice of *artbots* towards a framework for understanding them. This multifaceted area of enquiry requires research strategies that correspond to various interrelated elements of study, including a community of artistic practice, reconfigurable software components and web-based software applications, and the platforms that mediate human and bot sociality. Furthermore, this study seeks to contribute to the practice of artbots through novel experimentation. Having arrived at the view that internet-based software can be understood as *sociocomputational assemblages* and that artbots are artistic experiments in their configuration and framing, the methodological questions that arise ask: How are sociocomputational assemblages researched? What methods yield knowledge about artbot practices? And what kind of knowledge will these approaches yield?

In this chapter, I describe the methodology taken in response to this enquiry. My approach, involving mixed methods, views research as *opportunistic* (Riemer 1977), *enactive*, and *inventive* (Law 2004, Lury and Wakeford 2012, Marres 2012). It is grounded in *sociotechnical assemblage thinking* (Deleuze and Guattari 1987, Orlikowski and Scott 2008) as a means to address both human and nonhuman elements of study in constitutive interrelation. Moreover, this research study is both *about* practice and *through* practice by way of my own artbot works conducted for this study. Making my own artbot projects not only enabled experimentation in the field, but also provided me with access within the group of practitioners being studied through mutual interest and collaborative work.

This chapter begins with restating the research setting of artbots and artbot practitioners. It then discusses the research methods that were carried out for this study. These include *participant observation*, *inventive* methods specific to this study, practice-based research, research through collaboration, and a showcase of prior art. Finally, the methodological issues that these approaches bring up are discussed.

Research setting

As outlined in the introduction to this thesis, the history of bots spans from the 1980s through to the present day. However, this study focuses on a new community of practice that emerged alongside the arrival of Web 2.0 social-networking platforms such as

Flickr, Twitter, Facebook, YouTube, and others. These platforms not only encouraged sociality and communication, but also the creation and sharing of media (often referred to as 'user-generated content'). The development of web-based applications and reconfigurable software modularity became widespread, providing new tools such as APIs. Under these circumstances, a new community of bot-making practices emerged, constituted by individuals from a range of interdisciplinary backgrounds both technical and nontechnical, encompassing videogaming, interaction design, computational poetry, and other fields.

This study mainly examines case studies of artbots and practitioners present on Twitter. However, it felt important to provide a wider investigation into artbots plugged into other platform assemblages. As such, many of my own experiments and artbots were hosted on other networking platforms such as Tumblr (a blogging platform), Soundcloud (a music-sharing site), and Thingiverse (a 3D file sharing platform for 3D-print maker communities). Moreover, practitioners that experiment outside of the Twittersphere were brought into the study, including Sam Lavigne, whose work explores platforms such as LinkedIn (a social network for business and employment), and Julien Deswaef, whose work engages Facebook and YouTube. This opened up the study beyond the Twitter-centric group, and facilitated comparisons and insights that helped avoid developing an analytical framework that would only be relevant for sociotechnical assemblages contingent upon Twitter. Interestingly, towards the end of this study period, Twitter became more hostile towards bot accounts, and now more rigorously vet third-party applications. As a result, many of the artbot practitioners in this study have migrated their works to other platforms, further underscoring the need for a more inclusive framework of understanding that is not platform-specific.

Participant research

At the time, the artbot community of practice that I was studying mainly engaged with one another over the social-networking platform Twitter. Unique to this type of practice, their artworks occupied the same networking environment, and for research purposes (as well as personal networking and bot appreciation) I closely 'followed' both artbots and artbot makers. My engagement with this community and their creations could be described as *participant observation*, a widely recognised social research method (Schwartz and Schwartz 1955, Becker and Geer 1957, Musante and DeWalt 2010) in which the researcher "takes part in the daily activities, rituals, interactions, and events of a group of people" (Musante and DeWalt 2010: 1). Participant observation has been

relevantly used to study online communities, such as the virtual community Second Life (Boellstorff 2015) and the parenting community Mumsnet (Hine 2015). In some cases, participant observation can purposefully precede and influence the formulation of research questions (Spradley 2016: 10). In the case of this research, my focus of enquiry evolved alongside the early stages of community engagement, making it a formative period of this study that helped shaped the framework for understanding these practices. The participation in the emerging artbot community mostly involved communicating over social media, but for a more meaningful engagement in a community of 'practice', one must be active through one's own concurrent practice. Making artbots was my 'way in'. Having said that, the openness of social media also allowed observation without much direct involvement, and many communications between more active practitioners were accessible and visible to others on Twitter without insider access to the community.

'Following' the artbots generally involved observation rather than interaction, as there was a shift towards making non-interactive bots (which differ from chatbots, where dialogue is triggered upon user interaction). Artbots mainly broadcasted messages without chat solicitation, usually in scheduled intervals or triggered by external events. As a researcher, my role involved observing these nonhuman utterances and observing indirect interactions with these utterances through the 'likes', 'retweets', and responses of artbot spectators. These indirect interactions were crucial for understanding the appeal of artbots and how they successfully grew in popularity over the time of study.

Likewise, other practitioners conducted their own observations to hone their bot-making skills towards ever more popular artbots. The close observation of bots could be considered a 'nonhuman ethnography'. However, it was important not to fall into a binary thinking that separates humans and nonhumans, but rather take an *assemblage thinking approach* that identifies their *intra-relations* (Deleuze and Guattari 1987, Suchman 2007, Orlikowski and Scott 2008). My ethnography of artbots paid close attention to the human components that may constitute bot configurations. The artbot *Olivia Taters*, for example, is framed as an artificial teenager, but is assembled from fragments of other people's Twitter conversations, making it an assemblage of both software and human sources. These observations involving the constitutive sociocomputational nature of some bots would later be crucial in formulating the *entanglement* aspect of the framework, supported by the theoretical discourse of Barad (2003, 2007).

Opportunely, the artbot community came into being around the time of the research, between 2012-2014 (this study commenced in 2014). It was also favourable that I had already conducted some early practical experiments and research into Twitter bots (Wilkie, Michael, and Plummer-Fernandez 2014). Using my own experiences and interests to access a community for research purposes is itself a social research strategy that was popularised in the 1970s. In "Varieties of Opportunistic Research", Reimer (1977) argues that researchers too often neglect at-hand knowledge, their own life experiences, and what Reimer calls "situational familiarity", which can opportunistically serve as a source of research. Reimer reviews a number of studies in which the researcher was involved in the community of study beforehand, including that of home towns, taxi cab drivers, a police department, a chiropractic clinic, race tracks, and carnivals. A proponent of this type of research was Becker (1963), who used his status as a professional pianist in Chicago to gather research material on jazz musicians. At the start of this research project, like Becker, my personal artist practice was of a similar vein to the community I wanted to learn from, but not squarely the same. My early explorations in Twitter bots were clunky and short-lived, but were enough to begin interacting with a burgeoning community that understood more complex tools and methods. By continuing work on my own artbots and other experiments, I learned from others and further connected to them through a shared practice. I actively became what could be considered an "inside learner" (Blaikie 2009: 11).

Practitioners were keen to share their methods; for example, artist and educator Jeff Thompson would post bot-making tutorials on his personal blog, from which I began learning this craft.[63] Others, such as Kazemi and Parrish, would make their artbot code accessible to others on the software repository platform GitHub. I followed interactions between practitioners and the ongoing sharing of code and resources between them. The study of this source code in itself is an important research strategy for enquiry into these practices. Examining these software components would not only be resourceful in helping me create my own works, but offered valuable insights into how web-based software often relies on modular components such as APIs and software libraries – the constitutive elements of bot configurations. As conceived and explained by Fielding in his seminal thesis (2000), web-based software is increasingly made by interconnecting modular software components that are rendered contingently inter-operable during

[63] See, for example, Thompson's blog post about setting up a Raspberry Pi computer for running bots. This is a tutorial I followed and used as a method for hosting my own bots: *https://www.jeffreythompson.org/blog/2014/08/31/setting-up-raspberry-pi-to-run-bots/*.

runtime. During my studies of artbot practices, I learned first-hand how others employed such modularity, and how they also contributed to modularity by making bot-specific software components for others to reuse. In reciprocation, some of my artbot code was made available on GitHub for others to reconfigure towards their own artbot creations.

This participatory study mostly focused on gathering insight and understanding into the practices (the doing and making) rather than the personal biographies of practitioners. There is, perhaps, some detachment from their personal stories that may be detrimental to the overall understanding of particular artists, but the goal of this research attends mainly to establishing a broad notion of practice that can be identified across this emerging scene.

During this period of practice-based participant observation, I also ran a blog called Algopop, which acted as a community noticeboard for sharing new projects, artbots, code, and informal research into sociocomputational entanglements with others. Algopop was started in 2012 and grew to a following of nearly 20,000 subscribers. Creating a blog has been recognised as a method for establishing online community and rapport with others (Knowles and Cole 2008: 315). Algopop was created using the blogging platform Tumblr, which has previously been used for developing artist communities (Troemel 2011: 87).

Inventive and practice-based methods

Although this study has recognisable roots in social research methods (mainly participant observation), the parallel strategies that are more technology-based – coding artbots, deploying artbots within a platform, examining software, and collaborating over GitHub – could be considered *inventive* within sociology (Lury and Wakeford 2012, Marres, Guggenheim, and Wilkie 2018). The 'inventive' in sociology can mean both the innovation of new methods, and also a turn towards research that is more performative and participatory of the research material ("inventing the social"). More important, perhaps, is the need for inventive methods in the first place: why are new methods required for research? Law (2004) argues that social research methods are ill-equipped at dealing with the complex, diffuse, and messy nature of sociality, and that rather than understanding assemblages to be temporal and indefinite, they tend to work on the assumption that the world is to be understood as a set of "fairly specific, determinate and more or less identifiable processes" (p. 5). As such, new methods are needed to enact and bifurcate along with the multiplicative and complex nature of research assemblages.

Law proposes an assemblage and performative approach to research that "is a combination of reality detector and reality amplifier" (p. 14). This notion strongly resonates with the way research has been conducted for this study: intervening-in, becoming-with, and extending the sociotechnical realities it studies.

My approach is not attempting to be solely inventive of the social, but also of material and technical invention towards a sociotechnical study. There are some similarities with the human-computer interaction notion of 'research through design' (Zimmerman, Forlizzi, and Evenson 2007, Gaver 2012), through which computational objects are prototyped and tested with people in order to study their interactions. Here, new social happenings are prompted through interacting with these objects. The interactions are open-ended, what Gaver (2012: 940) describes as design's capacity to be "generative".

This methodology also has parallels to contemporary approaches in media studies that seek to "research media through new media" (Sayers 2018: 1). Sayers's companion to media studies suggests making media objects to study their wider impact and influence; for instance, making games to understand their politics, participating in social networks to locate their biases, assembling hardware to expose their ingrained defaults, and so on (ibid). Furthermore, Sayers argues that as researchers "we are entangled with the media we produce and research", and further proposes that "the study of media is the study of entanglements", which echoes my interest in adopting assemblage thinking – to think of research as a process that is interconnected, participated-in, and practiced.

My approach was to study sociocomputational assemblages through their making and operation, allowing their productive processes to play out towards enacting social events and media generation. The artbots I assembled and left to run on platforms were generative of a wide range of social engagements in which users commented on or otherwise responded to the actions of the bot. The bots also elicited responses from platform employees such as moderators and public relations personnel. Moreover, the bots were productive of their own media creations (the Thingiverse bot, for instance, created sculptural objects) and engaged nonhuman elements of the platform as well – triggering and intervening in, for example, the automated sorting and curation of a platform's content.

It could be argued that for the study of certain sociocomputational assemblages, it can be of use to 'invent' new configurations and allow their processes and intra-actions to play out, between both human and nonhuman components. Many of the case study scenarios

would not have been available for study otherwise. The appearance of a bot on Thingiverse, for example, is not something that could have been studied from prior case studies – there had never been a bot like it before on that platform. Enacting computational sociality through these inventive assemblages was immensely productive of new research material; the Thingiverse bot has prompted over 600 new bot-generated objects, over 300 user responses, as well as commentary from several prominent tech blogs, including an article by a key advocate of Web 2.0, Cory Doctorow.[64] These articles, in turn, prompted further comments and debates from readers, adding to the rich and diverse collection of research material.

Further to opening new research material and enacting sociocomputational processes and inter-relations, making my own artbots arguably helped move this form of practice in new directions. Firstly, as mentioned earlier, I used my own works to explore and examine how artbots operate within other platforms that hadn't been fully explored before. Secondly, my own creations allowed me to research particular aspects of the practice in relation to the conceptual framework I was beginning to develop. Towards the end of the research period, I became more aware of the importance of *figuration* (framing as character), and thus experimented with a focus on this element of practice. Through my own empirical research, I could quickly test ideas. For example, upon seeing a meme take off in response to David Cameron casually whistling, unaware that his microphone was still on after an important announcement, I created the Soundcloud bot *David Doo-doo* (2016).[65] In doing so, I tested the appropriation of popular figures and meme culture for a bot figuration.

Research through collaboration

One of the components of this study which was most productive of insight was a collaboration between myself and the artist Julien Deswaef. Deswaef, originally from Belgium but based in New York City, collaborated with me on the aforementioned Thingiverse artbot *Shiv Integer* (2016) over the course of three months. This long-distance collaboration was made possible through regular Skype conversations, a shared software repository on GitLab (similar to GitHub), and emails. This method intensifies participant observation towards a more focused and practice-led engagement with

[64] See Doctorow's article at *https://boingboing.net/2016/05/03/artists-troll-thingiverse-with.html*.

[65] See *https://soundcloud.com/daviddoo-doo*.

another practitioner. This allowed us to co-operate throughout the entire cycle of a project, from its conception until its operation.

Deswaef and I came up with the idea for the bot together after having an informal Skype conversation about a project of mine called *The Collector* (2015), which I had made for the Victoria and Albert Museum's Digital Design Weekend. The Collector 'scraped' (computationally examined) Thingiverse for 3D print files to download and had amassed a modest assortment of files. Around the same time, Deswaef had made the exceptional *Word Wars* (2015), a complex YouTube bot-account that published generative parody videos of the Star Wars opening crawler, based on current affairs. I blogged about Word Wars on Algopop, and we had exchanged emails and talked about it over Skype. Word Wars was novel in its use of Blender, an open-source 3D animation tool that could be controlled via code rather the user interface. The opportunity presented itself to learn more about this approach (and Deswaef's practice) by proposing a collaboration that combined his strategy with elements from the Thingiverse collector bot I had made. The collaboration lasted roughly three months, and on 14 February 2016, *Shiv Integer* became active on Thingiverse. Deswaef and I later worked on several exhibits of *Shiv Integer*, as well as an important solo exhibition of the project at an arts centre called iMal in Brussels.[66]

Research through collaboration seems to be rather underexplored as a research method. The artist-researcher Staikidis (2006) conducted an ethnographic study of the practices of Mayan artists by collaborating with two painters in a "mentorship learning experience", and examined that the collaborative relationship created "a kind of 'insidership' that is otherwise not possible to experience" (ibid, p. 119). Staikidis recalls that the engagement also led the Mayan instructors to become partly influenced by the researcher's own individual and cultural approach to painting (p. 127). A similar cross-pollination of thinking and practices was a welcome outcome of the collaboration with Deswaef, as we both brought something to the table.

In software development, collaborative practice has become commonplace, and this is evident in collaboration-enabling tools such as Git and the platforms that extend them, such as GitHub and GitLab.[67] These platforms can function as both a repository and as a

[66] See *http://www.imal.org/en/exhibition/shiv_integer*.

[67] Git is a version control system (VCS) created by Linus Torvalds (2005), the creator of Linux. It is said to make "collaborative development a pleasure" (Loeliger and McCullough 2012: 1).

project management and co-authoring tool. Git allows for different team members to work separately and regularly 'push' (put forward) one's own contributions to be incorporated into a 'master' file. Deswaef was familiar with this way of working, but it was still fairly new to me. We worked collaboratively and were largely assisted by the GitLab system, which allowed us to work on the same body of code simultaneously. The adoption of these tools and practices arguably influence practices to be more open and collaborative, including artistic practice. Kazemi, for example, sees the source code for his bots as very much part of the artwork.[68] Git repositories also record project activity, which was useful for later reflecting on the collaboration, much like looking back at notes. Deswaef and I, for example, utilised an 'issue tracking' component of GitLab to open and close conversations about artistic decisions as well as technical matters.

Research through curation

As part of this research study, I curated and organised a group showcase titled *Art of Bots* (2016). This showcase brought together, for the very first time, a selection of the seminal artbot practitioners, in a showcase event. The practitioners invited were Allison Parrish, Darius Kazemi, Matt Webb, Everest Pipkin, Emma Winston, Sam Lavigne, Jeff Thompson, Chris and Ali Rodley, George Buckenham, Julien Deswaef, Shardcore (Eric Drass), Thricedotted (Li Zilles), and Daniel Armengol Altayó. The showcase was supported by the UK-based arts organisation Abandon Normal Devices, and took place at Somerset House, London. It involved not only displaying artworks, but also bringing these artists together and facilitating important face-to-face engagement and moments of exchange with these practitioners.

The showcase allowed me to play a different role to that of researcher or fellow artist. Instead, I was more of an enabler and commissioner, and this granted the licence to have different kinds of conversations with the artists. Early conversations were set up over Skype to discuss which project would be suitable to showcase and how could it be adapted or reconfigured for a physical exhibition setting, with different audience engagements to consider. This allowed for further experimentation as to how artbot works could be further modified or have certain features drawn out. Processes and

[68] Kazemi discusses the relationship of source code to his work in this December 2014 podcast interview with the Guardian, around 30:36: *https://www.theguardian.com/technology/audio/2014/dec/17/darius-kazemi-bot-tech-weekly-podcast.*

relations were considered just as important, if not more, as having objects on display. One tactic was to consider exhibiting works that were already performative of participatory engagement outside of social media platforms. This worked well for projects such as Jeff Thompson's *Art Assignment Bot*, which was modified into an 'arts school' where audience members participated in completing bot-generated art assignments. Similarly, Emma Winston's *Graphic Score Bot* was reconfigured as a live music performance, in which Winston and other musicians improvised in response to the bot-generated graphical music scores appearing on a screen beside the stage. Other projects in this vein include the bot-making workshop *Cheap Bots Done Quick* by George Buckenham and the star finding project *9point2Billion* of Matt Webb. Buckenham's project invited audience members to create their own bots, blurring the line between exhibited artwork and audience participation and learning, which art institutions often deliver alongside exhibitions in the form of 'workshop' events. Meanwhile, Webb's artbot manifested as a physical installation of a floor-based star map that visitors could use to appreciate their age in relation to the distance of stars from the Earth.

A second curatorial approach was to extract some of the generated output of the online artbots to be shown as standalone media. Extracts from Sam Lavigne's YouTube bot *CSPAN-5* was showcased on large screens. A large print-out of a single glitched emoji from Allison Parrish's *Smiling Face Withface* was mounted on a wall. This was partly to observe how these projects translated to non-platform contexts; by removing any reference to the platform in which they manifested, I could examine whether the platform was simply a means for distributing media artworks, or a seminal part of the artwork's makeup.

The showcase also opened up new ideas around the curation of artbot practices. I strongly favoured the idea that the artworks could be reconfigured for the showcase, rather than work towards 'faithful' reproductions of the works as originally conceived by the artist and made manifest online. For this reason, there were no screens installed in the space to display internet browsers pointed at the platform profiles of the bots, as they appear online. Instead, I took the approach that people's smartphones already populated the show with the necessary screens and means of interaction with bots online. With this in mind, every caption for an artwork contained a URL reference to the artbot.

My approach to enabling processes, engagements, and reconfigurations reflects contemporary ideas in curation that challenge more traditional models of exhibiting.

Teresa Gleadowe (2000: 29) maintains that curators are now often implicated in the production of the work, rather than taking on the role of the art historian that would normatively present their research within the conventions of historical presentation. Curator Hans Ulrich Obrist (2000: 54) argues that museums should deal with process and not just objects, and seek to act and catalyse processes in a fluid and transdisciplinary way. His 1997 exhibition *Do It* was based on his open-ended, process-led model of an "exhibition in progress", in which artists contributed with "instructions" rather than things.[69] In *Rethinking Curating: Art after New Media*, Cook and Graham (2010: 1) argue that media-based art curation is best understood not as showcasing materials but "behaviours" that are participatory, performative, and generative. What is interesting about these approaches in the context of ongoing research, and in using curation itself as a research method, is that in doing a showcase *during* the research phase rather than presenting research findings about artworks, the research and artworks become mutually 'in progress' and entangled in their formation. Both artworks and research can be considered open-ended during the showcase, and plateau afterwards, through documentation and analysis of the showcase-research.

The research through curatorial practice as methodology resulted in research material collated for this study. The showcase event also importantly created rapport and social engagements with other practitioners, and insightful conversations took place during its planning and happening. Furthermore, the response from the press, including the Guardian's review of the show, helped establish the artform and provide feedback that would become valuable for my analysis.[70]

Methodological issues

A methodology is not without its defined boundaries between what is included in the research and what is "othered" (Law 2004). The methodology used in this study is mostly determined by the view that bots and web-based software are sociocomputational assemblages, and explores how these are configured, intra-related, and put into productive operation. A potential issue with this framing could be a lack of specificity, concreteness, and clarity offered by a framework that seeks to generalise across instances of sociocomputational assemblages. Assemblage thinking is used as a

[69] For the exhibition documentation, see *http://curatorsintl.org/special-projects/do-it*.

[70] See *https://www.theguardian.com/technology/2016/apr/15/move-over-chatbots-meet-the-artbots*.

framework guiding the heuristic abstraction of complex constellations of parts and processes, each of which may have empirical specificity and materiality that may be overlooked through this conceptual abstraction. As such, practice-based methods and experiments were pertinent towards balancing the abstract thinking with empirical and nuanced understanding. For example, the practicalities of this research sometimes involved simply getting an API to work, or installing a software module of the Python programming language (a problem that many programmers bemoan). These banal details of making sociocomputational assemblages work in practice puts the theory into action, testing its validity as a framework for understanding. Moreover, my use of assemblage thinking is very much as a conceptual tool that can be applied to the details of more specific applications.

As researcher conducting participant observation, I had to consider responsibilities towards research participants, such as the need for competency, informed consent, protection of confidentiality, and ethical publication (Musante and DeWalt 2010: 211). In situations where participant observation is carried out to study online communities, other specific concerns arise (ibid: 220), such as the need for sensitively addressing whether online communications on 'public' platforms can be quoted directly or if these require informed consent. Both approaches have been explored. Sanders (2005) conducted a study of sex workers by following their online conversations without making her presence known, as she argues that these message boards already protected their participants through anonymity and pseudonyms. In contrast, Boellstorff (2015) conducted a study of the virtual community Second Life and was open about his presence and role as researcher, and used consent forms and avatars to protect the virtual identities of Second Life citizens.

In this study, participant observation occurred across several points of engagement – Twitter, email correspondence, GitHub, Tumblr (Algopop), and Skype conversations. Furthermore, there was a multiplicity of types of engagement. For instance, GitHub was a site for examining a participant's code, and on Twitter, I may have observed a post by a participant that provided information about an artbot. Some of these engagements may have been pertinent to this study, and others less so. This multiplicity of engagement requires the researcher to be sensitive and astute about when and for what requesting consent or making known that research was being conducted was necessary. Not all of these points of engagement required consent, and the norms and terms of many of these platforms already handle consent and privacy matters. GitHub, for example, strongly suggests to participants that they include a licence with their code that would state terms

of usage. In regards to conversations with Deswaef mediated over Skype, Deswaef and I would continuously discuss issues of authorship and disclosure. In addition to these considerations, this thesis does not employ direct quotations of participants; instead, their contributions are abstracted through analytical observations that speak more generally about practice. Moreover, this approach helped maintain and foster a practice-based kinship in which informal conversations could be had, knowing that these could be left out of the study. Ultimately, the goal was to create an account of artbot practice and the processes and works of this community of practice, rather than conduct a social study into the lives and biographies of individual participants.

Another methodological concern is the potential bias in having a personal artistic practice that is somewhat predetermined before the study and brought into the research. Moreover, being entangled as individual practitioner as well as researcher could lead to ethical concerns regarding appropriating ideas and 'trade secrets' for my own practice. In this situation, I have taken great care to address these concerns. My own entanglement with the research is impossible to disentangle, and instead is used constructively towards the research to provide insights that are granted through this entanglement. However, to avoid bias, I have disregarded my own prior personal work and any predisposed personal 'manifesto' as part of the research. The framework for understanding practice is built from the ground up, starting with the examination of emerging artbot and other internet-software practices. Further to this, my earlier bot experiments are excluded from the case studies, and where possible, I've assumed a novice's approach and discovery of configuring sociocomputational assemblages.

In regards to the appropriation of ideas, care has been taken in ensuring that all the ideas here are either already available to the public through software repositories and artist blog posts, or provided in this research with consent. When it comes to my own practice, I responsibly declare any collaborators or remixing of software. The *Shiv Integer* project, for example, uses other people's 3D models to create 'mashup' sculptures, and each sculpture is fully compliant of Creative Commons licencing and credits the original authors appropriately at all times. Collaborating with Deswaef also required care in responsibly crediting his involvement and being transparent about our mutual work. This poses a challenge in conducting research that results in an 'original contribution to knowledge' – and care is taken in making sure that the practice is credited to both Deswaef and I. The analysis and understanding of this practical work that are developed as part of this study, however, are my own original thoughts.

Conclusion

This research study opportunistically builds upon my prior experiences, interests, and software skills to study an emerging artistic practice that experiments with Web-based software applications known as *artbots*. In the previous chapter, *assemblage thinking* was adopted to see these as *constitutive* or *contingent* on complex *sociocomputational assemblages*. Towards developing an appropriate methodology for researching such assemblages and artbots, a mixed approach was taken. Participant observation and practice-based research mutually allowed me to engage with other practitioners. Much of this engagement was mediated over the social networking platform Twitter, a platform that was also host to many of the artbots in study. Artbots on other platforms were also examined and experimented with towards a framework that helps understand a diversity of internet-based software and sociotechnical assemblages. Practice was at the heart of this study, and this involved creating my own bots and sociotechnical experiments that could help examine assemblage interactions and generativity – a strategy that shares similarities with contemporary media studies and 'research through design'.

This study explores the notion of conducting research through collaboration, which successfully enacted productive insights into practice, as well as a co-authored artistic project. In addition to this, a showcase of artworks was explored as a method. As the research was in progress, the showcase reflected this as a dynamic, interactive, and experimental setting where curatorial ideas and understandings of practice were tested out. These approaches yield a 'mess' of new sociotechnical happenings and case studies that require further examination and analysis. As Law (2004: 15) maintains, "Realities are not secure but instead they have to be practiced." This flux of new practice warrants research to play a performative role in its social construction towards a new framework for this type of practice. The following chapters unpacks these case studies with my proposed analytical toolset, describing by way of example, the details of this framework.

Chapter 4: Multiplicity

> "The artistic question is no longer: 'what can we make that is new?' but 'how can we make do with what we have?' […] Artists today program forms more than they compose them: rather than transfigure a raw element (blank canvas, clay, etc.), they remix available forms and make use of data."
> (Bourriaud 2002: 4)

Working with found materials is an increasingly familiar and accepted aspect of contemporary art practice. Bourriaud's *Relational Aesthetics* compared this approach to programming and working with data. But what if the contemporary artists were in fact programmers making use of existing software and data resources? In artbots, artists configure and reconfigure arrangements of components into active and dynamic sociocomputational assemblages. In this first chapter of three focusing on case studies, the notion of *multiplicity* (Deleuze and Guattari 1987b) is used as a framework to produce questions and discussion about the composite, heterogenous, reconfigurable, dynamic, expanding, and generative aspects of artbots. The notion of multiplicity clarifies that wholes can be multiples that are open to extension and durational changes.

This chapter illustrates how artbots are configurations of software, data, and connective components that are rendered interoperative and productive. Proponents of this approach to artistic practice are finding novel uses for readily available software modules and APIs, and reconfiguring them in interesting ways. Moreover, media is also treated as a resource, and formatted for interconnectivity. These approaches to practice are partly rooted in the *mashup* software development movement that coincided with *Web 2.0* principles and technologies (Yu et al. 2008).

One of the leading protagonists in the artbot community, Darius Kazemi, is influenced by Web 2.0 mashup culture.[71] In this chapter, I discuss some of Kazemi's works alongside other proponents of artbots who exemplify this reconfigurative approach. He repurposes software components and APIs in unexpected ways alongside developing software components and data repositories for other artists to reuse. Kazemi's own

[71] Kazemi confirmed this via a tweet is which he explained: "Web 2.0 mashup culture was a huge inspiration to me circa 2005, even if I didn't really participate until 2011." See *https://twitter.com/tinysubversions/status/960659827176321024*.

works remain open-endedly active, and generative of output. It was this approach that inspired my own artbot study titled *All Eyes*, in which I adjoin the API for the photo-sharing platform Flickr with computer vision software to produce endless photographic edits of detected eyes.

The open-ended activity of artbots can operate over periods of years, relentlessly generative and durational. Multiplicities aren't fixed, but change over durations, and this has become a theme of artbot work. I discuss this through examples such as Allison Parrish's *Everyword* (2007) and Sam Lavigne's *CSPAN-5* (2015).

Making multiplicities

Deleuze and Guattari's (1987: 8) notion of *multiplicity* principally describes the becoming-together of multiple parts. The authors further maintain that the "*multiple must be made*" (p. 6). Even the writing of their book is described as a multiplicity of parts, worked on in no particular order, and brought together into a cohesive multiplicity (p. 2). Haraway's (2016) term *making-with* suggests the making of multiplicities; I adopt this term to describe how, in software practices, pre-existing modular components are used in combination. Artbot practitioners tend to approach their work by making-with software modules and APIs; the Twitter API in particular became a key component which connected their configurations with a readymade audience of Twitter spectators.

A virtual rover is slowly roaming the Earth and transmitting its whereabouts to followers on Twitter. *Earth Rover Bot* (2014) by Colin Mitchell is an artbot that works by combining several Web-based elements into a software configuration including Google Maps, Google Street View, and the Twitter streaming API.[72] The bot is configured to virtually travel across continents, and uses Street View to obtain an image of its current location, which is presented back to Twitter followers (see Figure 2). Furthermore, people on Twitter can send the rover instructions, such as *move 10* and *left 90*, to alter its course. The bot manifests as a fictional narrative of a rover roaming Earth, but it can also be read as a virtual exploration of its parameter space, taking a trip through the database-space of Google's geo-tagged image bank. The configuration becomes contingent upon the functioning of its various elements, and it was due to a change to the Twitter API that caused the bot to end its final voyage halfway through Russia.

[72] The code for *Earth Rover Bot* is available on GitHub at *https://github.com/muffinista/EarthRoverBot*.

Figure 2. *Earth Rover Bot* (2014) by Colin Mitchell, at the location in Russia where changes to the Twitter API caused the configuration to cease working. *https://twitter.com/EarthRoverBot.*

The approach to combine disparate APIs and Web-based resources towards new applications can be traced back to circa 2006, when a new kind of Web-based software application, colloquially dubbed *mashups*, appeared. They were characterised by the "monster-of-Frankenstein-like manner in which they aggregate and stitch together third-party data" (Merrill 2006). The then-new Web 2.0 philosophy of providing APIs to make web services and components accessible for third-party use inspired this new approach to software development. Many early mashups utilised the newly available Google Maps API. Computer scientist Duane Merrill (ibid) explains that the Google Maps API "opened the floodgates, allowing Web developers (plus hobbyists, tinkerers, and others) to mash all sorts of data". This led to mashup applications such as *ChicagoCrime.org*, which visualised crime data over a Google map of Chicago.[73] Photo mashup applications were also popular, utilising the Flickr API, one of the first APIs that so generously made user-generated media available outside of the host application. The mashup trend, in turn, catalysed the creation of more API resources for third-party use. In addition, specially-made mashup connector applications were devised, such as Yahoo! Pipes (Yu et al. 2008). Now defunct, Yahoo! Pipes was a visual editor that facilitated the mixing of popular data feeds (for example, RSS or XML sources) through

[73] *ChicagoCrime.org* is no longer active.

interconnecting operators, and data processing operators that performed actions such as sorting or filtering.

Darius Kazemi claims that Yahoo! Pipes was an influence on his approach to programming.[74] Kazemi often "pipes" data from one resource to another. His *Museum Bot* (2014), for example, pipes random images from the Metropolitan Museum of Art to Twitter.[75] *Museum Bot* illustrates how Kazemi often draws upon new data sources; in this case, Kazemi was inspired by the Metropolitan's newly available open-access image bank that freely provided 400,000 images for non-commercial use.[76] Moreover, this example demonstrates another aspect of artbot 'piping': that data is never simply migrated from one source to another, but is drip-fed on a timer, trickling the data to the intended endpoint. In the case of *Museum Bot*, a *cron* timer schedules the piping of a single image to Twitter four times a day. Like grains of sand in an hourglass, datapoints are made to flow from the source database into a recipient database, preferably one which publicly fills a social media timeline for observers to experience this gentle cascade of incoming media over time.

Figure 3. *Museum Bot* (2014) by Darius Kazemi. *http://twitter.com/museumbot*.

[74] See *https://twitter.com/tinysubversions/status/960632213824978945*.

[75] See *https://twitter.com/museumbot*.

[76] The Metropolitan announced this programme in May 2014, shortly before Kazemi created *Museum Bot*: *https://metmuseum.org/press/news/2014/oasc-access*.

Other works by Kazemi also use media archives as an input, including the use early hip-hop flyers for *Old School Flyers* (2015), early tweets for *Very Old Tweets* (2014), and an animal video library for *Animal Video Bot* (2015).[77] Kazemi and others tend to use pre-existing media data rather than producing entirely novel visual media through software. This is in stark contrast to *generative art*, a software-based artform that uses software to generate images and other media. Furthermore, generative art rarely produces internet-based applications that are continuously running and trickling-out generative works. Instead, generative art more commonly involves making desktop software applications to produce iterations that the artist then hand selects as exemplary specimens produced by the setup. An appeal of artbots is the removal of this selection process, leaving spectators to experience the full gamut of the software's output, and to patiently wait for serendipitous moments in which the software outputs a winner.

Another implication of working in this approach is that Web-based datasets tend to be formatted for interconnectivity, which makes it possible for artbot artists to work across different forms of media (image, text, sound, video, and others). The artist Sam Lavigne, for example, whose work is discussed later in this chapter, has created artbots works that work with video (*CSPAN-5*, 2015; *The Infinite Campaign,* 2017), emails (*The Good Life,* 2016), faxes (*Yelp Prison Review Faxbot*, 2015), patent applications (*Patent Generator*, 2014), predictions (*White Collar Crime Risk Zones*, 2016) and graphics (*Big Data Pawn Shop*, 2014). This approach echoes what theorist Lev Manovich's describes as the *metamedium* (Manovich 2013: 101), and also relates to art critic Rosalind Krauss's (2000) *post-medium condition*, in which contemporary artists are no longer defined by working in a particular medium.

Having illustrated the basic media-piped-to-Twitter artbot configuration, I will now describe some more complex assemblage configurations comprised of APIs and modular software components that process data. Often, artist configurations purposefully reconfigure a processing element in an unexpected way, subverting its most common-sense use. *Reverse OCR* (2014) is another work by Kazemi that uses a Web-based software module that performs *optical character recognition* (OCR). OCR software is normally used to convert images of typed, handwritten, or printed text into machine-encoded text, usually from a scanned document or a photo of a document. In *Reverse*

[77] Darius Kazemi has all these listed with URLs to the projects on *http://tinysubversions.com/projects/*.

OCR, Kazemi repurposes a Web-based OCR service called Ocrad.js, and uses it to process random scribbles generated by the software until a fully detected 'word' is successfully generated.[78] These words appear barely intelligible to a human but pass as legible when processed through the OCR software (see Figure 4). Kazemi's configuration then uses APIs for Twitter and Tumblr to post these generative scribble-words onto both platforms.

Figure 4. *Reverse OCR* producing a scribble that reads "goodness".
http://reverseocr.tumblr.com/image/166751774363

The artist Everest Pipkin uses the same Web-based OCR component in the work *Cloud OCR* (2015). The configuration of this work exemplifies the mashup approach often identified in artbot practices. In this work Pipkin brings into interrelation Google Maps, Google Street View, and Ocrad.js towards a configuration that looks up to the sky on Google Street View and processes the image capturing the sky through the character recognition software. A skyline captured above the Gila National Forest in New Mexico was computationally read as "j - _| | | _ . . _- .- _ :'.'.". . .J . |_" (Figure 5).

Figure 5. *Cloud OCR* (2015) by Everest Pipkin. *http://everest-pipkin.com/*

[78] Ocrad.js is available at *http://antimatter15.com/ocrad.js/demo.html*.

Making-with readily available Web-based components seems key to artbots. *Cloud OCR* is conceptually defined and described by its interrelation of clouds and OCR software. Often, the premise of a work is a novel repurposing of an API, or the juxtaposition of two or more disparate components working with each other in unexpected ways, co-contingently producing a durational multiplicity of media. In addition, an important aspect of artbot practices is not merely the production of configurations, but of components and datasets available for other artists to use. Often, these are produced for a particular project, but are then encapsulated into reusable modules for others. The code-sharing platform GitHub is an example of a platform where these components are commonly archived and made accessible. Kazemi's *Corpora* collection, for example, contains datasets formatted as JSON files.[79] If, for example, you needed a formatted list of dinosaur names, 'dinosaurs.json' could be imported into your project from the corpora.

To summarise the discussion so far, the configurability and reusability of software architecture, along with the encapsulation of software components for third-party use through APIs, prompted a trend known as *mashup* development. The mashup approach and the available tools for it influenced a new wave of artbot artists. Artists soon set up software architecture configurations that output media to Twitter, trickling the serendipitous outcomes of interconnected Web-based resources and services into view. This approach can be described as the configuring of a sociocomputational assemblage, that is durational and multiplicative. In the following section, I explore in more detail what proceeds after these sociocomputational assemblages are set up, and how many artbot projects continue to be active, productive, and open to further modifications to maintain their continuous running.

Scalability and duration

> "[The work of art] now functions as an active agent, a musical score, an unfolding scenario, a framework that possesses autonomy and materiality to varying degrees." (Bourriaud 2002)

[79] See *https://github.com/dariusk/corpora*.

In artbots, the artwork is defined by the intra-activity of the sociocomputational assemblage, and this activity can be productive of virtually endless output. The aggregated body of work grows – but spectatorships may also grow, interactions may become more frequent, and the digital footprint of the artbot increases, appearing in more search results. This approach is conceptually different to a determinate artwork, and even different to a kinetic sculpture that remains active but reproduces the same activity time and time again. An aim of artbots is to balance repetition and novelty through the inventive design of its *durational multiplicity* (Deleuze and Guattari 1987: 53), – the event space that unfolds during the artbot's operation. The artbot practitioner would not only be focused on what the configuration generates, but when, how often and for how long.

One of the architects of the Web, Roy Fielding (2000: 32), introduced various terms to describe properties of software architecture. *Scalability* refers to a software architecture's ability to support a growing number of components, processes, and productivity within an active configuration. The configurability, evolvability, and reusability of software architecture has implications for artbots. These properties enable artworks to be considered open-ended and ever-expanding, rather than fixed in form and realisation. The property of scalability is made possible by design, often by using Web-based servers to host the software, and interfacing social media platforms that support unlimited user-generated media.

For my own artbot experiments, I set up a home-based Web server to run bots continuously. My first experiment, *All Eyes* (2014), is a configuration designed to aggregate eyes detected in images from Flickr. Its basic hardware setup consists of a *Raspberry Pi* single-board computer,[80] running without the need for a screen and keyboard.[81] This set-up was re-purposed for the other artbots made during this study (see Figure 6). Using a computer vision library called OpenCV to process images sourced from Flickr's API, the active configuration began to search for images tagged 'face' and 'selfie', and then processed these found images through face and eye detection

[80] Raspberry Pis are inexpensive single-board computers developed in the UK; see *raspberrypi.org*.

[81] For this, I followed the online tutorial "Setting up Raspberry Pi to run bots" by the artist Jeffrey Thompson: *https://www.jeffreythompson.org/blog/2014/08/31/setting-up-raspberry-pi-to-run-bots/*.

software.[82] The images were then programmatically cropped and resized. Finally, they were disseminated on Twitter.[83]

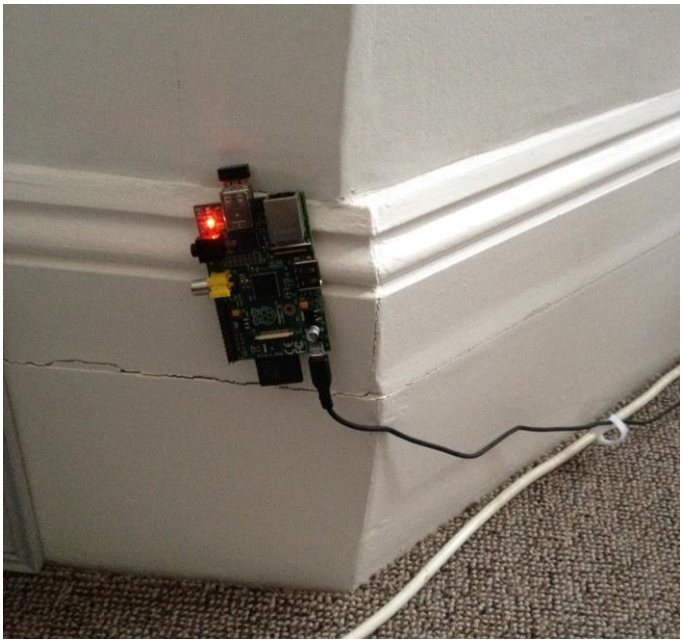


Figure 6. The Raspberry Pi computer with Wi-Fi dongle that hosted the artbots made during this study.

Figure 7. An eye produced by *All Eyes* (2014). *https://twitter.com/_all_eyes*

[82] For OpenCV, see *https://opencv.org/*.

[83] See *https://twitter.com/_all_eyes*.

All Eyes becomes a sociocomputational assemblage, that is not only the result of a procedural software process, but also of the sociocomputational activities of digital photography, image-tagging and image-sharing. The human act of looking into the camera is mirrored and completed by the computational act of capturing the human, first through the digital camera and later, through the computer-vision software that identifies the human eye in the digital photograph.

Durational multiplicities are to be understood as a material-temporal expansions – enduring time as a parameter. Some artists have explicitly made the scalability and durational aspect a theme of the work. *Everyword* (2007) by Alison Parrish, for example, was active for seven years, outputting a word to Twitter every half-hour until all words in the English language had been tweeted.

Everyword is considered one of the first artbots on Twitter, and it was made whilst Parrish was studying at ITP NYU, where she now lectures on computational poetry. *Everyword* helped establish aspects now pertinent to artbot practice: the use of existing datasets, the trickle feed of that data to social media over long durations, and vastness as a theme.

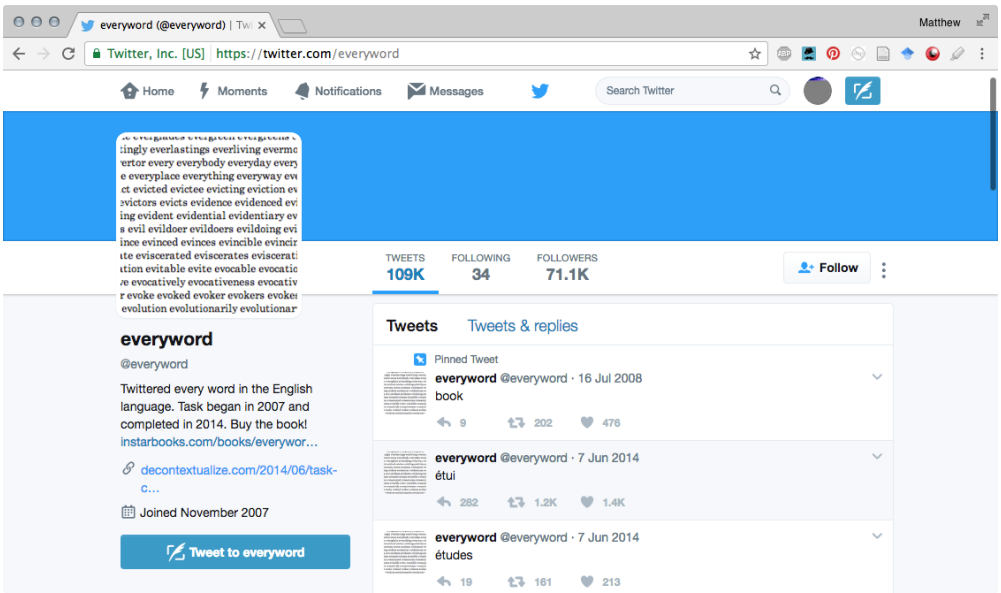


Figure 8. A screenshot of *Everyword* (2007) by Allison Parrish. *https://twitter.com/everyword*

The bot's configuration is a simple arrangement of readymade software components and a dataset containing every word in the English language that, when active, trickle-feeds words to Twitter once every 30 minutes. The artwork and its audience suggest a sort of

seven-year pilgrimage through vocabulary space, exploring cultural preferences and interpretations of words. In 2014, the blog *Gizmodo* reported that *Everyword*'s most endorsed words were "sex", "weed", "vagina" and "shit" (Horn 2014), perhaps more illustrating of the social makeup of Twitter in 2014 than of the English language.

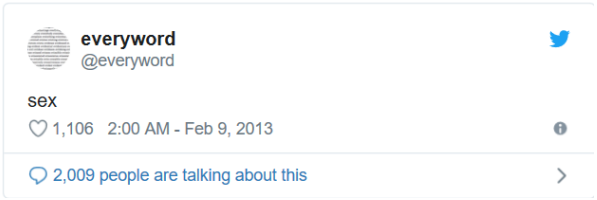


Figure 9. *Everyword* producing the word "sex", its most popular word by June 2014.

After seven years of predictable outputs, the account delivered every consecutive word beginning with 'z' and then progressed to words beginning with 'é'. The event caused a stir amongst the thousands of people that were patiently deliberating on a predictable final word beginning with 'z'. Instead the final word was *étui.*[84] The non-sequitur exposes how the data was structured, where 'é' would come after 'z'. In this case, textual media is structured by ASCII conventions – each character has a numerical representation, and as such, 'é' is in range of values higher than those assigned from 'A' to 'Z'.[85]

A multitude of parody and homage accounts have been made by others such as *Fuck Every Word* (which tweeted the word 'fuck' followed by the consecutive word), also, *Every Bird, Every Curd*, *Every Nerd, Every Non-word, droW yrevE, Misspell Every Word, Every Letter, Every Colour*, *Every Unicode, CYBEREVERYWORD,* as well as mirror accounts in other languages such as *iederwoord* (Dutch) and كُـل كَـلِمة (Arabic).[86]

A precedent and source of inspiration for *Everyword* is the software art work *Every Icon* (1996) by John F. Simon, Jr., an artwork that presents every variation of white and black pixels possible on a grid, 32 squares wide and 32 squares tall – the size of a 1997 desktop icon (Mirapaul 1997). *Every Icon* would have taken several hundred trillions of

[84] An *etuí* is a small ornamental case for needles and cosmetics.

[85] ASCII stands for American Standard Code for Information Interchange. Computers only understand numbers, so an ASCII code is used as the numerical representation of a character.

[86] The project also was turned into a book. *Everyword, the book* is published by Instar Books: *http://www.instarbooks.com/books/everyword.html*.

years to complete. Parrish contemplated that “well, if someone can make every ‘icon’, I'm going to say every word on Twitter, and it went on for seven years” (Garcia 2016).

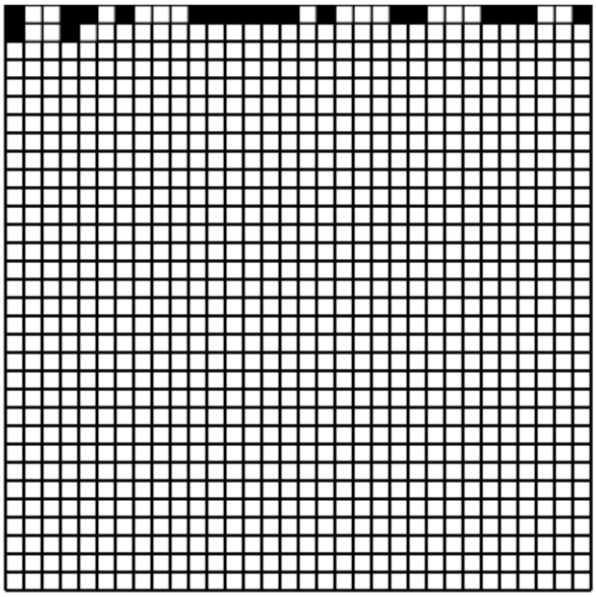


Figure 10. *Every Icon*, 1996 by John F. Simon Jr. The wall hanging version was comprised of the software running on a Macintosh PowerBook170 encased in plastic acrylic.

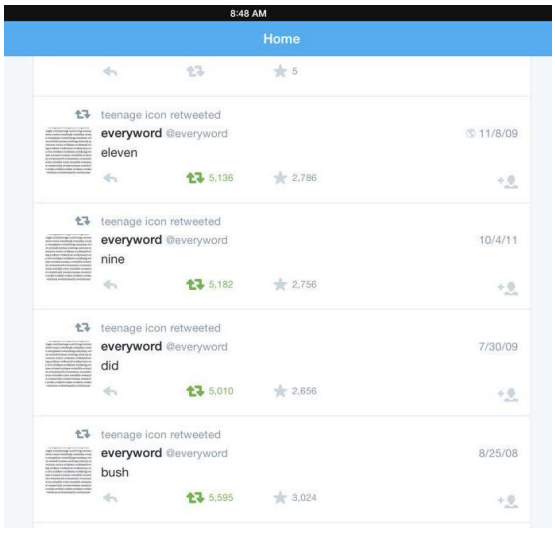


Figure 11. Tweets from *Everyword* uttering the words “bush”, “did”, “nine”, and “eleven” were appropriated by Twitter users as a means to spread the message “Bush did 9/11”. Screenshot taken from *https://twitter.com/thetrashknight/status/578538466407002112*.

An upshot of on-going and open-ended artbots is that they may inevitably remain ‘unfinished’, a strategy also found in relational art that has been both endorsed (Bourriaud 2002) and criticised (Bishop 2006). A comparison can be made to art theorist

Umberto Eco's *The Open Work* (1989), suggesting that 'open-endedness' has been a wider concern of art theory, principally in regards to the open-ended interpretation of artworks. Artbots works differ from Eco's reading of traditional art practice in that the open-endedness does not stem from the open-ended reflection of an artwork but, rather, it is open-ended in its production. The machinic production of the artbot often entails that the work has no definitive end and instead remains open to extension. Umberto Eco's notion of *opera aperta* – the open work – argues that art is fundamentally an act of creating "ambiguous situations open to all sorts of operative choices and interpretations" (Eco 1989: 44). Eco's notion alludes to the multitude of meanings and interpretations created by viewers engaging with an artwork. Here, multiplicity is derived from 'reflection' rather than production.

This is of relevance in relation to the exhibition of *Art of Bots*, which was an experimental showcase in which artworks were adapted for the event. Art of Bots provided a chance to extend the multiplicity and reconfigurability of artbots into the physical event. In collaboration with their respective artists, each was reconfigured in ways that would work for a new set of considerations and audiences. As such, the works were not simply showcased within their original sites of dissemination (mostly Twitter), but adapted to reach exhibition audiences in more appropriate ways.

Various artbots by Allison Parrish, including *The Ephemerides* (2015), were reconfigured as 'chapbooks' – small, fanzine-like publications that were staple-bound. Reconfiguring the output of a bot into a printed publication is to find a plateau along the open-ended way, and take pause. Concretising the work for an event, for a moment, is to materially suspend the spectacle of generative trickle, and shift from one form of open work to another – finding multiplicity in the endless readings and interpretations of a poetry book.

The Ephemerides (2015) pairs a randomly-selected image from NASA's OPUS database (a repository of data from outer planet probes like Voyager, Cassini, and Galileo) with a software-generated poem. The idea behind the artbot was to address the similarity between space probes and generative poetry procedures and ask: what would poetry written by a space probe look like?[87] The text of the bot comes from two sources: *Astrology* by Sepharial and *The Ocean And Its Wonders* by R. M. Ballantyne, both

[87] Taken from Parrish's website: *http://portfolio.decontextualize.com/*.

available from Project Gutenberg.[88] The first text contains references to the planets and their movements; the second text is about the open seas and oceans, and lengthy, often one-way voyages of discovery. Parrish clearly sees parallels with journeys into undiscovered expanses, and the vast parameter-space of software configurations which require software-probes to explore. As Parrish so eloquently puts it, "[artbots] are really good at vertigo in the face of the infinite" (Garcia 2016).

Figure 12. *The Ephemerides* (2015) by Allison Parrish.

[88] Project Gutenberg offers over 57,000 free eBooks, with a focus on older works for which copyright has expired. Parrish often uses the Gutenberg archive as a source for computational poetry: *http://www.gutenberg.org/*.

Figure 13. *The Ephemerides* was reconfigured as a chapbook for *Art of Bots* (2016).

In facing the infinite, artbots traverse the durational multiplicities their configurations can virtually create, only ever touching the tip of a parametric iceberg. These metaphors of exploration and distant worlds are a reoccurring theme in artbot practice. Take for instance *Uncharted Atlas* (2016) by Martin O'Leary (see figure 14).

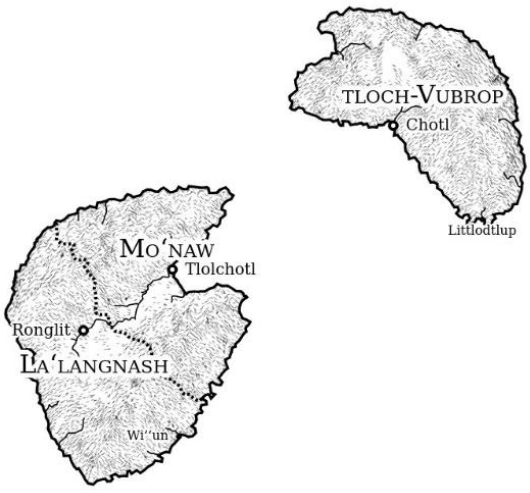


Figure 14. A cropped screenshot of http://twitter.com/unchartedatlas

Uncharted Atlas generates maps of imaginary places. O'Leary's attention to detail is evident in the way the naming of the towns and villages of the map are believably of the same region and language. Figure 14 depicts two neighbouring islands, one has a settlement called Tlolchotl, and the other Chotl. To achieve this, O'Leary generates a fictional language for each map, and then generates the region's names based on that language. The language and the map are diegetic elements of the narrative world, along with its computer-generated rivers and coastal lines. In chapter 6, I further discuss the narratives and characterisation of artbots, and revisit the theme of exploration.

Big Data Pawn Shop

In *The Work of Art in the Age of Mechanical Reproduction* (1936), Walter Benjamin described how the replication of art in the twentieth century lead to a "tremendous shattering of tradition" and that contemporary artforms such as film were destructive in their reproduced images of cultural heritage. Benjamin was conservatively critical of how the popular masses were consuming culture through modern modes of cultural dissemination. Later, Berger (2008) argued that reproduction devalued artworks and these were in danger of losing their original meaning through their reproduction:

> "For the first time ever, images of art have become ephemeral, ubiquitous, insubstantial, available, valueless, free. They surround us in the same way as a language surrounds us. They have entered the mainstream of life over which they no longer, in themselves, have power." (p. 32)

Berger's concern for art becoming "ubiquitous", "insubstantial", and "valueless" is worth discussing in light of a new artform that embraces computational multiplicity. Perhaps the creation of valueless items could be acknowledged as characteristic to some cases of artbot practice, and that the 'value' of the artwork is found elsewhere – not in the output produced by artbot, but it in its meta-commentary on information overload.

CSPAN-5 (2015) by Sam Lavigne is of relevance here. This artwork outputs short videos of auto-edited news channel footage focused on the coverage of U.S. governmental proceedings, and automatically generates short edits that piece together segments only where the most used words are uttered. This artbot is a complex assemblage combining automated transcribing, language analysis, and automated video editing. The software architecture is comprised of many software components, including Lavigne's own automated video transcribing software library called *Audiogrep*, a video editing software

called *MoviePY*, the YouTube API, a YouTube video-capturing software called *youtube-dl* (used for aggregating videos from the CSPAN channel), [89] and a part-of-speech tagging software called *TextBlob*. Furthermore, Audiogrep is comprised of other components such as a speech-to-text tool called *Pocketsphinx*.[90]

Figure 15. Screenshot of *https://twitter.com/cspan5* by Sam Lavigne.

CSPAN-5 was exhibited at *Art of Bots* by presenting a selection of its generative videos on two large screens. One screen presented videos generated by a new reconfiguration of *CSPAN-5* titled *Parliament-Live* (2016), which processed UK Parliament proceedings (see Figure 16).[91]

[89] The Python package MoviePY can be found at *http://zulko.github.io/moviepy/*. Audiogrep, released under an open-source software licence, can be found at *http://antiboredom.github.io/audiogrep/*.

[90] The procedure that *CSPAN-5* follows is to 'visit' the URL *https://www.c-span.org/search/?sdate=05%2F16%2F2017&edate=05%2F16%2F2017&searchtype=Videos&sort=Most+Recent+Airing&text=0* with the date changed to the current date. It selects a video at random that it hasn't already downloaded. It downloads the video using youtube-dl. The video is then transcribed using Audiogrep, which outputs words with timestamps. It then finds the most commonly spoken nouns by identifying nouns in the text using the part-of-speech tagging software TextBlob, makes a new edit of the video stitching together segments when the noun is uttered using MoviePY, and finally, the new *CSPAN-5* video is uploaded to YouTube.

[91] *Parliament-Live* can be found at *https://www.youtube.com/channel/UCEo_Dm8JsDe1YLwd52HCr4A*.

Lavigne finds that these edits not only offer insights into the "language patterns in political speech and, the tedium and rhetoric of governing"[92], but also considers the edited videos to be a counter-strategy to the "deluge of information" that he sees as a form of obfuscation (Rosenthal 2017). In my opinion, the strength of the work is not the dry detection of language patterns, nor its feat of tactical art-activism, but in its humorous results. In one example, footage depicting Donald Trump, resulted in the detection of the frequent use of "Trump" in the transcript – mostly uttered by Trump himself.

Lavigne's work makes reference to not only the vast parameter spaces probed by software, but how these are increasingly based on the data to be extracted from more complex media such as video. Tim Berner-Lee's dream of the "semantic Web" will be comprised of software architectures like Lavigne's *CSPAN-5*, mining the valueless deluge of archival video in the hope of identifying newfound meaning and value.

Figure 16. Presentation of *CSPAN-5* and *Parliament-Live* at Art of Bots, April 2016.

Lavigne's other high-profile project *Big Data Pawn Shop* (2014) similarly reflects computationally-driven mass production. A collaboration with Adam Harvey and Surya Mattu, *Big Data Pawn Shop* is an online store which offers everyday products decorated with leaked official United States National Security Agency (NSA) documents through

[92] Interview with Lavigne for the Art of Bots showcase in April 2016.

an online marketplace called Zazzle.[93] The configuration generates combinations of Zazzle products – mugs, fridge magnets, pillowcases, lamps, candles, door mats, pet supplies, neck ties, can coolers, tote bags, water bottles, shower curtains, soap dispensers, to name a few – with the fifty pages of the NSA ANT Catalog. The two inputs, when interconnected, generate a vast number of speculative items which can be ordered and manufactured on demand.

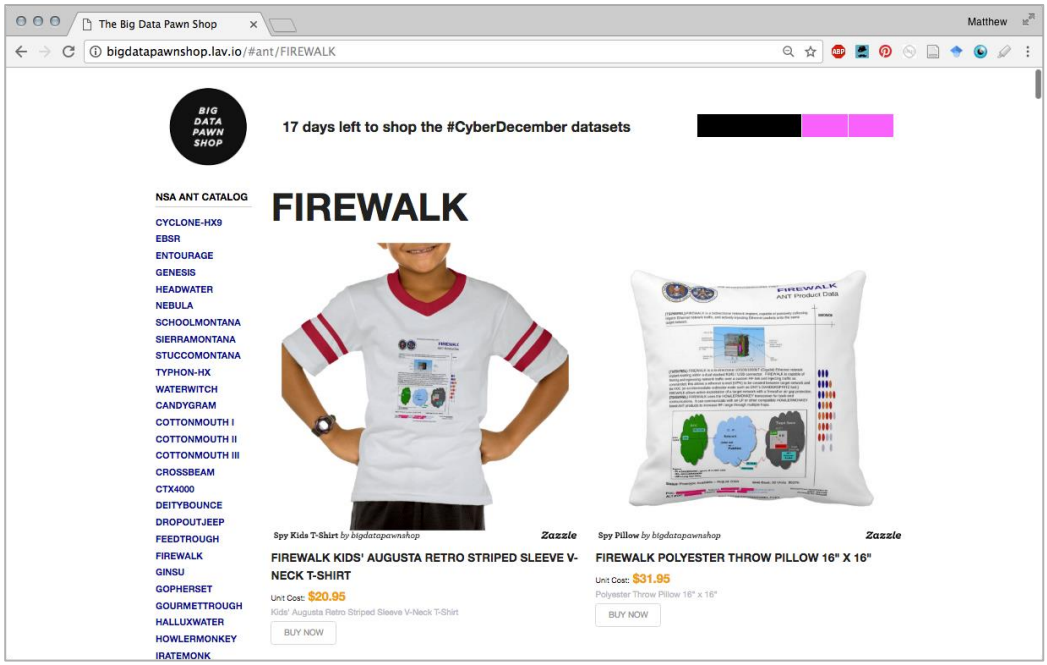


Figure 17. A screenshot of *http://bigdatapawnshop.lav.io*.

In 2013, computer analyst Edward Snowden leaked a cache of top-secret documents that he obtained while working as a contractor for the NSA. These were published simultaneously in the Guardian and The Washington Post, drawing criticism over the extent of undisclosed internet surveillance. One leaked document was the NSA ANT Catalog, a classified document detailing technology available to the NSA. It is a list that reads like a mail-order catalogue for spies, with prices ranging from free, to a quarter of US $1 million (Applebaum and Stöcker 2017). Lavigne and his collaborators produced *Big Data Pawn Shop* somewhat as a parody (Stinson 2017).

[93] Zazzle can be accessed at *http://www.zazzle.com*.

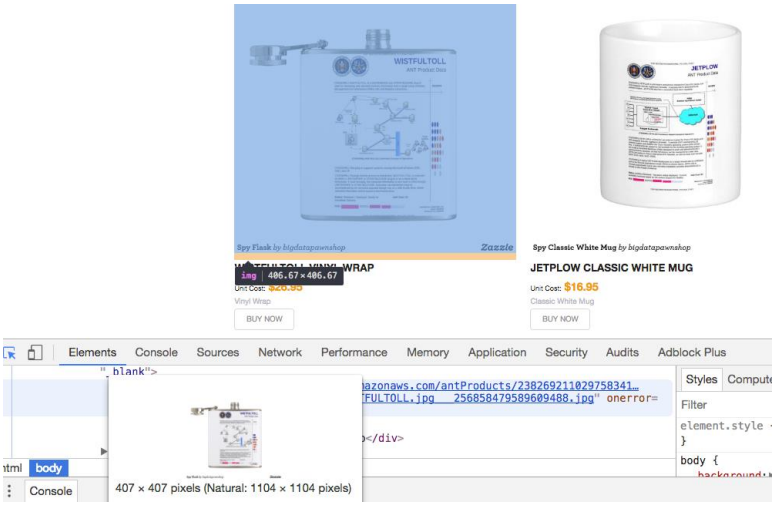


Figure 18. The HTML inspector reveals the source image that is superimposed onto the Zazzle product. When the Web shop visitor clicks on a codename, speculative products are generated 'on-the-fly', by fetching a random selection of Zazzle products and replacing their design with that of the leaked NSA document, simply by substituting the source image in the URL.

For *Art of Bots* (2016), this project was adapted by getting a selection of objects printed to make a market stall of readymade items (see Figure 19). The combinations were manually selected; for example, the *TOTEGHOSTLY* document was printed onto tote bags.

Figure 19. Big Data Pawn Shop at Art of Bots, 2016

Big Data Pawn Shop illustrates Benjamin's concerns about the effect of the mechanical reproduction of images, but rather than devaluing the 'aura' of an artwork, it shatters the aura of confidential documents. Once leaked onto the internet, these special, never-before-seen images, begin to suffer a demise in significance, and become objects of ridicule – a set of clumsily-designed, meaningless graphics. *Big Data Pawn Shop* exacerbates this completely, making these images the fodder of mugs and mousepads. One can imagine Berger balking at a mug decorated with a Monet; now computational architectures can systematically render all images, no matter how important, as the lowly matter of mugs.

Furthermore, the project shines a light on the strange ecology of Zazzle. The online marketplace is a magnet for bot activity, evident through the weird slogan-sporting T-shirt designs that populate the site. The Zazzle market vendor *Supernova23*, for example, offers a range of designs with software-generated slogans that are clearly following templates that are computationally completed (see Figure 20).[94]

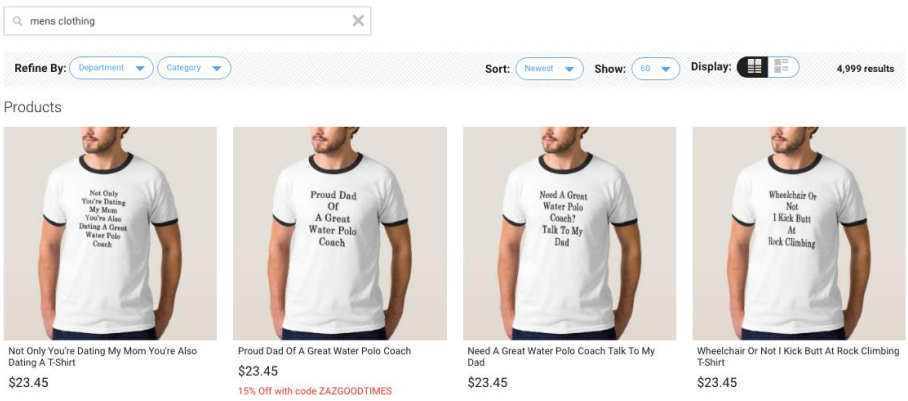


Figure 20. Zazzle marketplace items offered by Supernova23

Conclusion

Artbots practitioners make-with Web-resources – software components and interesting datasets, often accessed through APIs – and connect them in either simple or complex assemblages that often remain active over extensive periods of time. Such configurations may simply drip-drip randomly-chosen images in front of social media spectators, allured by the promise of database gems. Spectators may be given to following artbots

[94] Supernova23 t-shirts can be accessed at *https://www.zazzle.co.uk/supernova23+tshirts*.

with more complex configurations that activate generative processes, and await the serendipitous moments in which the architecture produces an exceptional result.

As illustrated, artbots often invoke certain traits that have become characteristic of this type of art practice. Artbot works often remain 'unfinished': open-ended and continuously active over long durations. These durational multiplicities are characterised by scalability and evolvability, and as such, can be reworked for different audiences and different parameters. At the Art of Bots showcase, artbots were reconfigured or evolved to work within the requirements of the exhibition space and offline audiences. The works of Allison Parrish, for example, manifested as short poetry books that contained selected extracts from artbot outputs. These can be appreciated as plateaus of artbot practice that take pause and dictate a different kind of reading of the work, as described by Umberto Eco's notion of the open work.

Two distinct themes were identified in artbot works in connection with their multiplicative nature. On one hand, artbots often explore the theme of vastness and explorations. Probes, stars, rovers, and voyages are utilised as narrative devices that reflect how the underlying system is simultaneously exploring the parameter space created in its multipart configuration. On the other hand, artbots may reference the deluge of data in the information age – a surplus of media that is no longer appreciated with attentive eyes but left to software systems to pick through, doing the work of detecting patterns and 'semantics' in the clutter of cyberspace. What would Benjamin and Berger think of artbots in the age of Web-based re-production?

In the following chapters, I continue to examine and discuss case studies of artbot practice through the analytical perspectives of entanglement and figuration. Similar to multiplicity, these are not strict criteria that all artbots adhere to, but provide a vantage point that enables discussion and the highlighting of key aspects and implications to artbot practice.

Chapter 5: Entanglement

This chapter examines and discusses an aspect of artbot practice that explores the entangled relations of humans *within* sociocomputational assemblages. Sociocomputational assemblages are comprised of both human and computational activity, arranged in software architectures that encounter, engage, or contingently depend on people, including the entangled practitioners that make them. Artbots are characteristically social, habitually present on social media, and engaged with human spectators that follow their activity for amusement. Artbots, however, often eschew traditional roles of social bots. Rather than interact with humans, they are configured to engage with them in other capacities and types of connections.

Human-computer relationships extend beyond 'interaction' to 'intra-action', through which humans are not externally distinct from a bot, but entangled within the software architecture that brings it into being. Artbot practices often use – and consciously encourage – intra-activity with entangled humans both to produce their effects and to highlight entanglement itself.

In this chapter, I first illustrate how the notion of 'interaction' normatively figures humans as 'users' of computational technologies. My argument is that 'user interaction' is one of various modes of human entanglement with Web-based software applications. Some artbot practitioners continue to employ more traditional user-interaction relations between people and software, and I examine a few case studies of these presented at the Art of Bots showcase.

Entanglement can be understood as a single conceptual dimension regarding the different points of inter-connection between humans and software architecture. As the chapter progresses, I examine and discuss projects that increasingly entangle humans into participatory roles within artbot configurations, becoming constitutive of it. Such human intra-relations can manifest in the authoring of software, the populating of databases, and the reuse of media assets. Moreover, some software architectures make use of components that encapsulate outsourced human task-solving (described as *artificial artificial intelligence* as well as *human computation*).

In this chapter, I also discuss *Shiv Integer* (2016), an artbot I made in collaboration with Julien Deswaef. This project illustrates multiple points of entanglement with others

within a 3D-file sharing platform called Thingiverse. This experiment opened up multiple debates around co-authorship between human creators and bots, and various people felt that the encroachment of bot activity was a form of spam, whilst others embraced the artbot's mangling of their designs.

User interaction

Interaction, by which audiences are invited to respond, engage, and contribute to the realisation of an artwork, has become an increasingly established current within art (Brown 2014: 1). Interactivity has also been a key area of exploration in computer-related art practices (Paul 2003: 3), and can differ between subgenres. For example, artworks may involve screen-based or kinetic installations in which a person's movements may be picked up by sensors, influencing the artwork. The seminal works of artist Rafael Lozano-Hemmer are often configured in this way, providing experiences for audiences to interact with the artwork.[95] These works rely on software and hardware architectures composed of screens, sensors, and servers, and may also employ software components such as face detection software. In these types of interactive systems, the spectator is usually figured as a form of 'user', expecting the system to respond to actions and gesticulations.

Interactive art of this type shares similarities with software applications that are configured to provide responsive services mediated over a user interface. A Web browser is a graphical user interface that makes it possible to navigate the Web. Fielding's (2000) contribution to software architecture sought to make improvements to the structure of Web-based software application design. However, its concept of users (and how users are configured to interface the Web via the Web browser) remained largely unchanged and unchallenged. Software architecture is limited by its implicit distinction between users and the architecture, configured in modes of interaction that meet at the 'interface', the gateway between the two sides.

Orlikowski and Scott's research (2008) finds that prior research (in organisational studies and technology) often invokes the notion of 'interaction' in which "actors and objects [are] primarily self-contained entities that influence each other" (p. 455). The problem they identify is that interaction becomes an ineffective framework for the study of human-nonhuman "agencies that have so thoroughly saturated each other that

[95] See *http://www.lozano-hemmer.com/projects.php*.

previously taken-for-granted boundaries are dissolved", and as such, adopt the notion of sociomaterial assemblages.

Interactivity is *not* a determinate property of technology. Rather, technology becomes interactive by design; that is, through a designer's understanding of humans and technology as discrete elements that require an interface between them to interact. In software development practices, humans and nonhumans may be treated as, and moulded into, these user-interface-software arrangements. In the design of artbots and their predecessors, this type of relation has often been formative of the bot's design. However, as I will demonstrate further on, other relations are being explored and implemented in artbot design. Barad's notion of intra-action is key to understanding these other types of relations, and how they differ from an approach informed by interaction:

> "The neo-logism 'intra-action' signifies the mutual constitution of entangled agencies. That is, in contrast to the usual 'interaction,' which assumes that there are separate individual agencies that precede their interaction, the notion of intra-action recognises that distinct agencies do not precede, but rather emerge through, their intra-action." (Barad 2007: 33)

My appropriation of Barad's theory, leans away from the epistemological roots of *agential realism ontology*, and hinges more upon its implications for the ontological framing of human-nonhuman ensembles, as well as the entanglement of human practitioners within the materiality they re-configure. This leaning is closer to Suchman's own interpretation of Barad's theory that she references to enable a renewed understanding of sociomaterial practices of human-computer interaction.

Interactive bots

Chatbots are often thought of and configured as user interfaces in which users engage a software application via conversation rather than a graphical user interface. An example of this is *Dom* from Domino's Pizza, the "AI bot with one purpose in life: to make ordering pizza fast, from any of your gadgets".[96] Users can speak to Dom to order a pizza, or have a text-based conversation in the browser. In 2016, there was a surge of interest in chatbots-as-interfaces to services. Microsoft CEO Satya Nadella claimed that

[96] See *https://www.dominos.co.uk/easy/*.

"chatbots are the new apps", and moreover, Facebook implemented chatbots into their Messenger application, for which 30,000 chatbots were created (Dredge 2016).

Artbots, on the other hand, tend to eschew these conventions of chatbot interactivity, and make artbots that are non-conversational, yet output utterances or media on social networks. People may still sign up for following the output of the bot as spectators (Veale and Cook 2018: 2). As discussed in the previous chapter, these spectator followings can last for years. *Everyword* (2007-2014) maintained spectators for the entire duration of its finite operation to tweet every single word in the English language.

Although these bots are non-conversational, human spectators may still engage in conversation, either talking back at the artbot, or talking amongst themselves. It is as though despite these bots not having the faculties to listen and understand human responses, spectators may still persist or simply find comfort in talking back. The artist Everest Pipkin comments about how long-term followers of *Tiny Star Field* (2014) often talk back to the ASCII-art star-fields that the artbot generates:

> "Most (although not all) of my bots are non-interactive, meaning that they do not @reply [sic] back when spoken to. That being said, they are absolutely interacted with. *Tiny Star Fields* in particular gets a ton of messages; lots of people will have conversations in the mentions." (Bucher 2015)

Although most artbots do not offer responsive user-software behaviours, there have been some experiments that do so. Darius Kazemi's *Sorting Hat Bot* (2016) attracts fans of Harry Potter by composing a unique poem for each follower (see Figure 21). The artbot generates bespoke poems in the style of the Harry Potter *sorting hat*, which, in the book series, sorts students into Hogwarts houses. A response from *Sorting Hat Bot* reads:

> "@irenebm_13 The bravery of a buffalo, the laughter of a cod
> In Hufflepuff you do belong since you are really shod".[97]

Sorting Hat Bot produces the sort of audience experience found in more traditional interactive-art, in which a personalised response is produced by the artwork.

[97] This engagement can be found at *https://twitter.com/SortingBot/status/931072356675661824*.

Figure 21. *http://twitter.com/SortingBot* by Darius Kazemi.

Entanglements at *Art of Bots*

At the Art of Bots showcase, some works could be examined and understood as more traditional interactive art experiences. Other works explored more unusual relations between audiences and artwork, in which different actors became constitutive of the artbot's activity. The showcase helped examine and analyse the possible points of entanglement that span interaction and intra-action.

5point9billion (2015) by Matt Webb is an artbot that informs its subscribers how far light has travelled since leaving the Earth on the day they were born.[98] Having provided it with a date of birth, the bot calculates the distance travelled since that day at light-speed, and looks up which stars are closest to that distance. At Art of Bots, this project manifested as a floor map of the stars, as well as a screen-based interface for user-audiences to engage with. Upon entering their date of birth, the system would inform them of their most proximate star, encouraging them to find it on the floor-based star map (see Figures 22 and 23).

[98] See *http://twitter.com/5point9billion*. The Twitter-based bot is an update on a former iteration of the project called *Light Cone* (2003), which utilised RSS to deliver messages to users. Webb created *Light Cone* before co-founding BERG in 2005, an influential interaction design studio that were pioneers in internet-enabled devices. Webb has a background in physics and has been creating internet-related projects since 1998.

Figure 22. User-audience interface for *5Point9Billion*, presented at Art of Bots.

Figure 23. The *5Point9Billion* floor-based star map, presented at Art of Bots.

Webb's experience as an interaction designer at BERG is an influence on his approach to artbots. *5Point9Billion*'s arrangement figures software and users in external relations to one another, meeting at the point of interface. Here, the software provides a response (and payoff) for inputting personal data via an interface. The software performs a calculation with that data, crosschecks this with the database of stars, and returns the results.

If these traditional modes of interaction were our only framework for understanding artbots (and for conducting the practice of artbots), it would neglect discussion about internal relations and intra-activity. In this particular case study, the database of people and their personal details generated by this user-software configuration is somewhat overlooked. *5Point9Billion* not only interfaces with humans in external relation to its architecture, but also populates an internal database with information on people and their birthdays. In response to privacy concerns, Webb commented that "the thing is, your birthday will leak anyway",[99] and furthermore, suggested that this privacy risk works as a warning: "it makes people realise that their information will be public anyhow" (Webb 2015). Webb's thin responses seem to indicate that the internalisation of user data had been ignored in favour of providing unhindered, front-end user services.

This example illustrates that 'interaction' poses the danger of alluding to the separability of the user from the architecture, but in fact, user data remains part of the software-database assemblage. Webb has no nefarious motives behind this, but it does highlight the issue for other interactive systems that may fail to disclose how data is retained and utilised. Facebook, for example, has repeatedly facilitated data-sharing with other tech companies such as Spotify and Netflix, claiming that this does not breach privacy concerns because it is done with the intent of extending services to their users (Dance, LaForgia, and Confessore 2018).

Other works included at Art of Bots were similarly configured to deliver interactive experiences for audiences. My curatorial approach was an experiment into different manifestations of artbots in a physical setting and facing people directly. At the time, my understanding of sociocomputational assemblages, entanglement, and intra-activity had not been fully developed, and my own preconceived ideas about interaction and externalised activity guided curatorial choices. The next two case studies demonstrate

[99] Quoted from Webb's blog: *http://interconnected.org/home/2015/12/14/5point9billion*.

arrangements of interactive experiences, whilst a third case study demonstrates a move towards an arrangement that fosters intra-active entanglement.

Graphic Score Bot (2016) by Emma Winston was another work presented at Art of Bots. This artbot generates abstract music notation for musicians to improvise to (see Figure 24). The computer-generated scores are meant as prompts for open-ended music-making, an external activity from the software architecture's operation. This work follows a lineage of graphic score design by composers such as John Cage. The graphic scores of Cornelius Cardew have visual similarities to those produced by Winston's bot (see Figure 25). For the Art of Bots showcase, the graphical score software was output to a screen, and musicians, including Winston, performed in response to the generative scores throughout the duration of the exhibition (see Figure 26).

Graphic Score Bot is intended for musicians to loosely respond to. The artbot is not interactive in the sense that it responds to a user's request or interface engagement, so in a sense, the bot is not interactive. The direction of interaction is reversed, with the bot prompting responses from humans. In the case of *Graphic Score Bot*, human activity is still externalised. However, in contrast to this, Winston's own output and aesthetics can be said to be internalised in the software assemblage, replicating decisions and aesthetic choices that the artist embeds into the software. This 'entanglement' between programmer and software can often be overlooked, but in reality, all software are implementations of the embedded ideas, values, and decisions of their programmers. Software, in this sense, acts a proxy to execute the intents of its programmer.

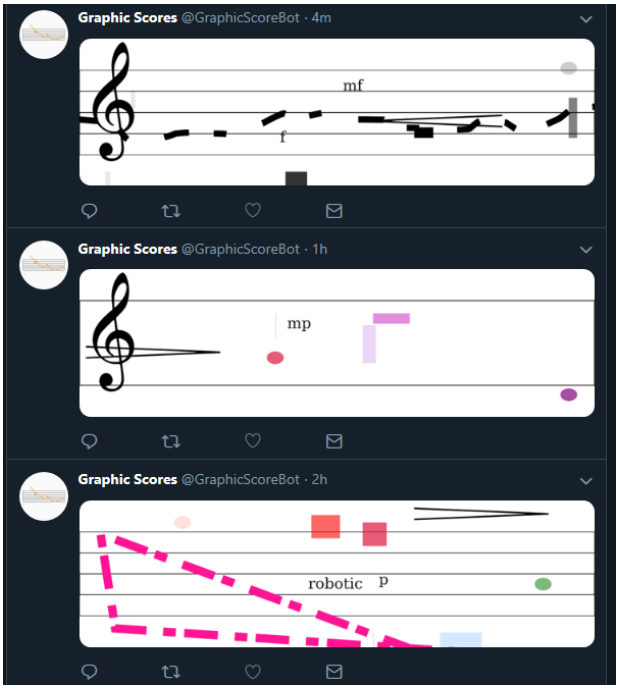


Figure 24. Screenshot (cropped) of *https://twitter.com/GraphicScoreBot*. *Graphic Score Bot* (2016) by Emma Winston.

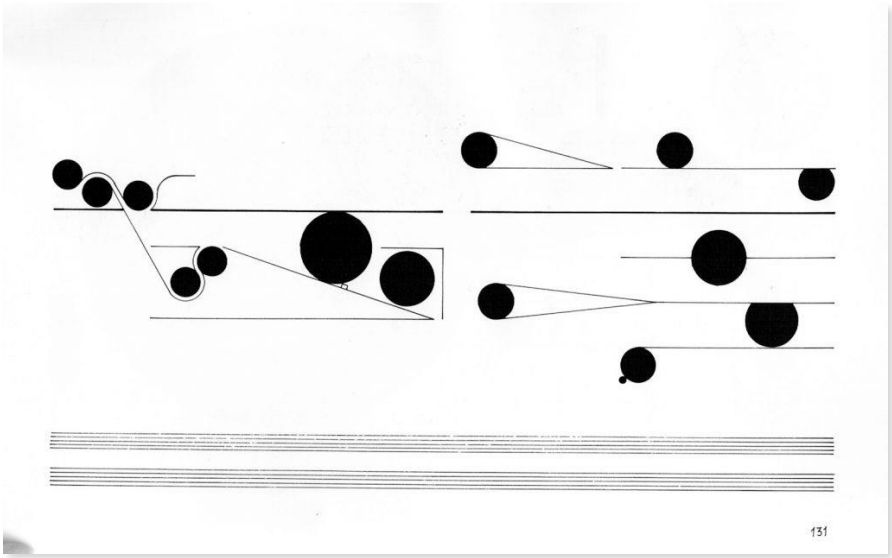


Figure 25. *Treatise* (1967) by Cornelius Cardew was a compilation of graphic scores drawn by the avant-garde composer (Cardew 1967).

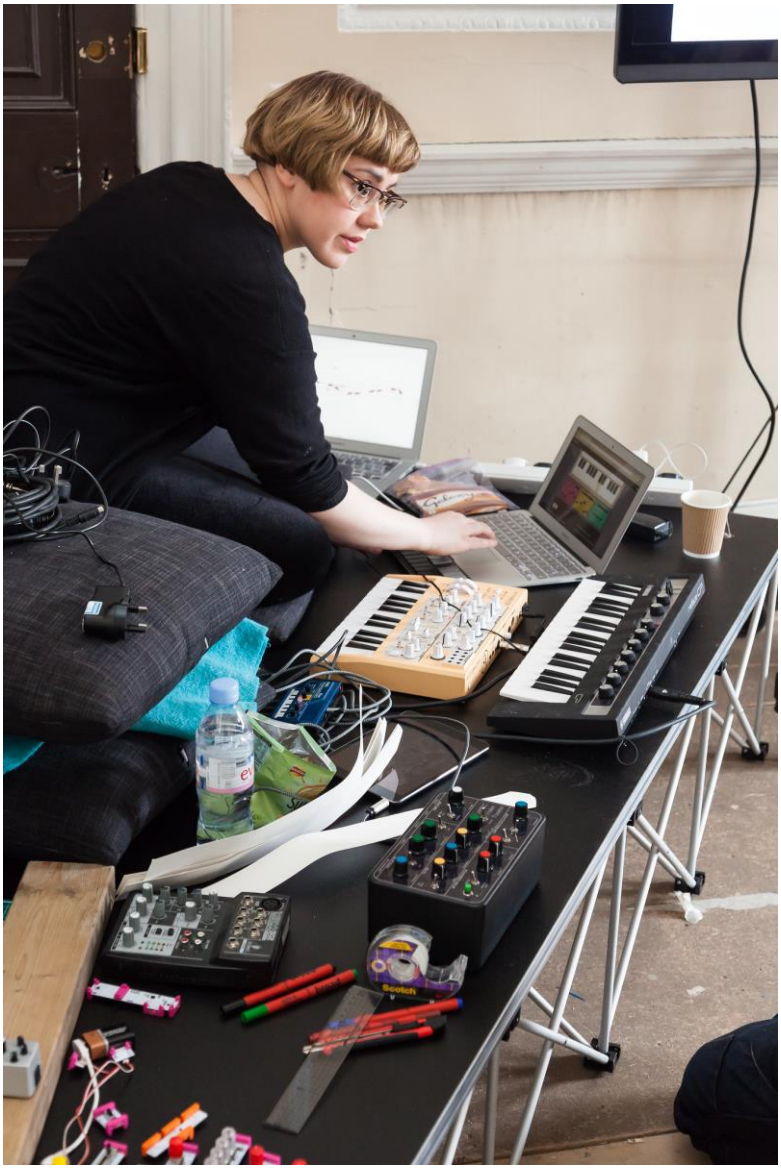


Figure 26. Emma Winston performing to *Graphic Score Bot* at Art of Bots.

Art Assignment Bot (2013) by Jeff Thompson generates art assignments for others to complete. Thompson initially intended for the bot to be a parody of open calls for artist residencies, mocking the use of themes and deadlines. Each assignment created by Thompson's software includes a choice of medium, a topic, and a deadline which ranges from minutes to years. Thompson began discovering that followers of *Art Assignment Bot* would sometimes complete an artwork in response.

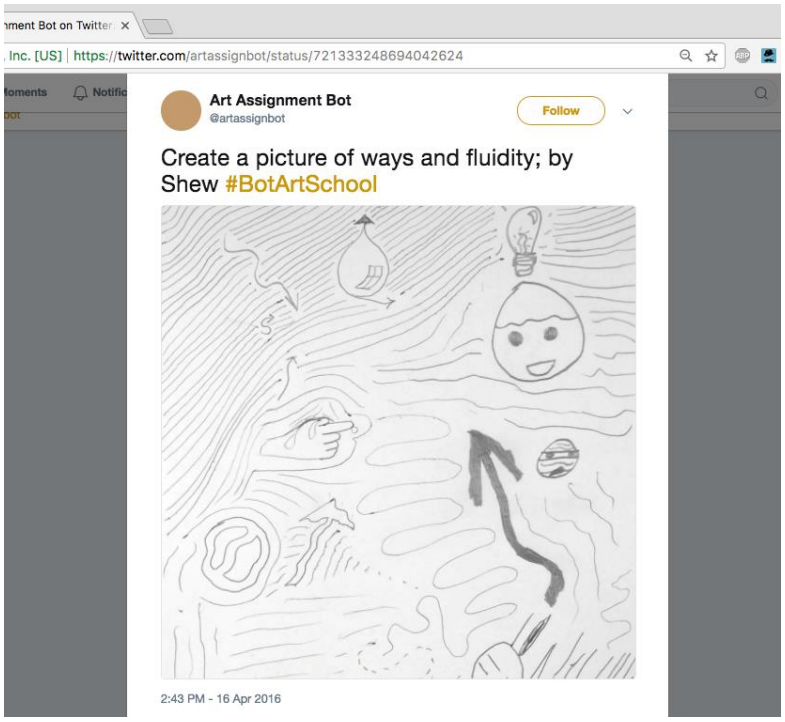


Figure 27. Screenshot of *https://twitter.com/artassignbot* by Jeff Thompson, demonstrating a completed artwork from *Bot Art School*.

For Art of Bots, the project manifested as *Bot Art School* (2016). At *Bot Art School*, audience members were invited to respond to computer-generated art briefs, printed out on A4 sheets with space for a drawing. The drawings were shared with others both in the physical space by fixing them to the wall, and online by publishing them through the *Art Assignment Bot* Twitter account. Thompson was there to bridge the software and audiences, which suggests that his own role was that of a conversational interface between the two entities; the human chatbot, so to speak.

Figure 28. *Bot Art School* at Art of Bots. Jeff Thompson (second from left) hands an art assignment to a new assignee.

My point is that even in situations in which artbot-audience relations are configured as 'interactive', other forms of human involvement may be present and integral to the functioning of the architecture. The binary thinking that separates humans and machines fails to account for the intra-activity present in software configurations that involves authors, engineers, mediators, and human sources of data.

At Art of Bots, one other project experimentally proposed a different role for audiences in which they are not figured as users, but as the authors and engineers of new artbots. At the Art of Bots installation of *Cheap Bots Done Quick* (2015) by George Buckenham, a bot-making facility was set up to invite audiences to work towards creating Twitter bots, and many of these bots remain present on Twitter.[100] At Art of Bots, this project configured an entanglement of intra-activity in which the physical setting illustrated the human activity behind the online presence of social media bots. Moreover, audiences experienced a more immersed and transparent understanding of artbots, in which the

[100] To create a bot using the platform, a participant has to manually set up a Twitter account and then proceed to *cheapbotsdonequick.com* to give the platform authorisation to run the Twitter account. Once the account access is enabled, the participant codes the operational basis of the bot using a generative grammar tool called Tracery. Tracery was created by Kate Compton (Compton et al. 2015).

making process was opened up to their input as co-authors (see Figure 29). This approach contrasts the traditional approach of treating audiences as observers and users. The *Cheap Bots Done Quick* installation becomes an ever-multiplying sociocomputational assemblage of human and computational intra-action.

Figure 29. Visitors making bots using *Cheap Bots Done Quick* at Art of Bots, 2016

Entanglement

On Twitter, you may find yourself retweeted by an account called *Pentametron* to then discover you have inadvertently penned a verse in pentameter length that rhymes with someone else's tweet and has been specially selected to be included to join with millions of others in an ongoing poem that started in 2011.[101] *Pentametron* (2011) by Ranjit Bhatnagar is an artbot that collects countless posts from people that are found to follow an iambic structure and rhyme with one another (see Figure 30). The result is a Twitter stream entirely composed of entangled people's tweets, forming a dynamic and scaling co-authored poem. Bhatnagar's software analyses millions of tweets to find a single rhyming couplet (Read 2012).

[101] In a verse of poetry written in iambic pentameter, there are five unstressed syllables, each of which is followed by a long (or *stressed*) syllable.

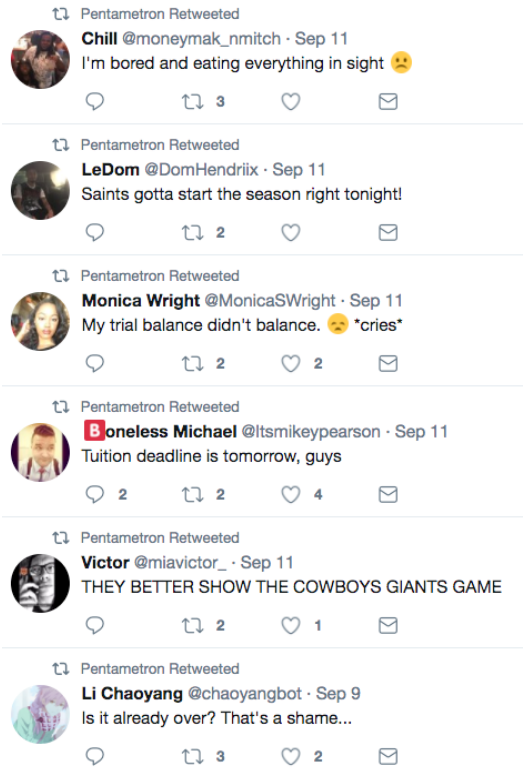


Figure 30. Screenshot of *https://twitter.com/pentametron*, by Ranjit Bhatnagar.

Pentametron is a remarkable example of an artbot that emerges through the intra-activity of millions of entangled humans within a sociocomputational assemblage. Bhatnagar's edifice does not hide this process of aggregation but brings it to light, allowing entangled others to see how software has picked up a pattern within their utterances. Rather than treat these users as potential targets for services and advertising, the entanglement is shown for what it is, displaying the emergent and dynamic phenomena that occur upon congruous intersections of human and nonhuman activity.

My notion of sociocomputational assemblages clarifies that software architectures are comprised of and contingent upon both human and nonhuman activity. Through this lens, an examination of artbot practices asks how human-nonhuman entanglement is purposefully configured or reconfigured within an artbot assemblage, and what sort of capabilities, properties, or implications are realised through these entanglements.

> "The universe is agential intra-activity in its becoming. The primary ontological units are not 'things' but phenomena—dynamic topological reconfigurings/ entanglements/ relationalities/ (re)articulations." (Barad 2003: 818)

The notion of entanglement, in this framework for understanding artbots, is inspired by the theoretical arguments of Barad (2003, 2007), Suchman (2010) and Orlikowski (2007), whom have all contributed to understandings of human and nonhuman intra-activity, and their constitutive becoming-together as sociomaterial assembalges. In my own understanding of entanglement, I am interested in human-nonhuman relations and activities that constitute sociocomputational artbots. Moreover, these entanglements are often brought into being during the runtime of software architecture, so it is through their empirical realisation and running processes that entangled relations may manifest.

Software application edifices are assemblies of components, connectors and data, in inter-operation. As I shall illustrate, software architectures may contain several points of entangled intra-activity with humans, and become contingent upon it. Architects (designers/engineers/ artists), data (human resources of data), media content (sometimes explicitly 'user-generated'), and human-based processing components (human computation, and human resources for training machine learning algorithms) are all possible areas of entanglement within sociocomputational assemblages.

Let me take as a starting point the entanglement of software architects within architectures. Distributed software applications are often designed with evolvability and scalability in mind (Fielding 2000: 32). Rarely is software produced at the outset and left unattended and unaltered by its engineers. In continuous entanglement, the designers and engineers of a software assemblage may be tweaking and observing the runtime functioning of the architecture, working behind-the-scenes (or more aptly, from within). Seaver (2014:10) points to this ongoing tweaking, and Orlikowski (2007:1440) arrives at a similar conclusion in her analysis of Google's search engine:

> "The Google search engine is computer code that was produced and is maintained by software engineers, that executes on computers (configured with particular hardware and software elements), and whose operation depends on the millions of people who create and update web pages every day, and the millions of people who enter particular search terms into it. The result is a constitutive entanglement of the social and the material."

Software applications, through this understanding, are inherently entangled with their creators, and are co-constituted by their activity and thinking. For the case of artbots, their code can be seen as an implementation of the creator's decisions, ideas, aesthetics,

values, emotions, concerns, humour, taste, politics, and so on, which of course influences the functioning, configuration, and figuration of the bot.

Some artbot practitioners may tacitly or involuntarily embed themselves into the artbot assemblage, and others may be more conscious of it and seek to enact their influence on the artbot. *Empathy Deck* (2016) by Erica Scourti creates personalised tarot cards generated from mashing up Scourti's diary entries with self-help texts. Scourti's persona becomes constitutive of the artbot's, behaving as a cyborg extension (see Figure 31). *Empathy Deck* outsources Scourti's online engagement with others, offering comforting messages and empathy to her friends without her direct involvement, and at the same time, is not characterised as humanlike, but instead is figured as a deck of cards (that inherit traces of Scourti's writing and persona). In the following chapter on figuration, I discuss how artbots often eschew humanlike characters for alternative forms.

Figure 31. *http://twitter.com/empathydeck* by Erica Scourti.

Entangled configurations that extend, reformat, or garble the reach or persona of an entangled other are not limited to humans. *@oliverbarkbark* (2015) is the entangled dog-bot of Henry Conklin, who set up a Raspberry Pi computer with a microphone to pick up his real dog's barking and translate these into tweets (see Figure 32).[102] Oliver the dog

[102] The code for this project is available at *https://github.com/HenryWConklin/barkdetect*.

becomes a component, along with other components such as microphone and bark detection software, in an edifice configured to output, “Woof woof ruff woof ruff. Bark!” A difference between this entanglement and that of Scourti with *Empathy Deck* is that the entanglement is configured to occur dynamically during the software’s runtime, syncing the actions of the dog with the actions of the bot.

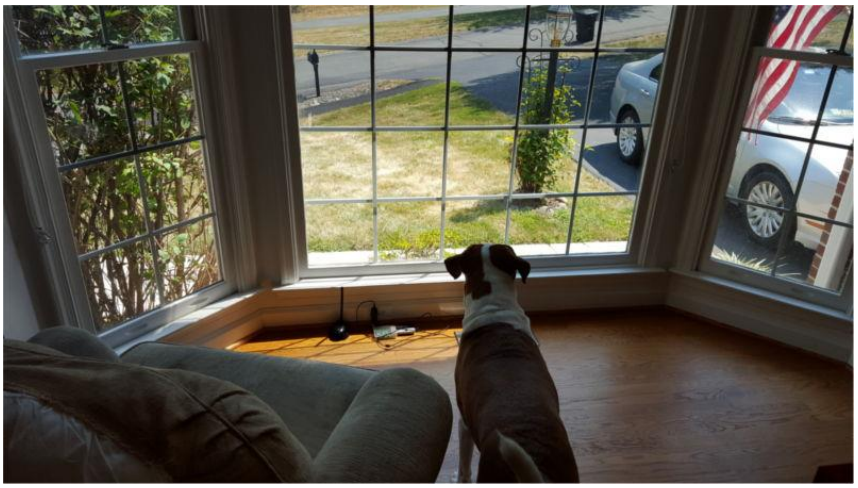


Figure 32. Oliver the dog by the window, with the microphone and Raspberry Pi in front; all components of a heterogeneous software architecture that manifests online as *@oliverbarkbark*.

Figure 33. *http://twitter.com/oliverbarkbark* by Henry Conklin.

A similar point of entanglement in sociocomputational assemblages is the relation between software edifices and human sources of data. Reformatting prior user activity as data to populate databases and be used as resources towards analytical insights, statistics, or plainly, to look up information about individuals, can render people uninformed and unaware of the computational activities they may become entangled with. Social media platform architecture often stores a multitude of data points about their users, and may contentiously make this data available to third-party software application engineers to be repurposed.

Artbot practitioners often repurpose data and thus entangle others in the construction of their artbots, which may contentiously abstract human-based sources under the guise of a nonhuman persona. The previously introduced *Olivia Taters* (2013) by Dubbin constructs sentences by repurposing archived Twitter interactions between users containing adverbs such as 'literally' and 'totally'. Dubbin explains that the artbot would find tweets such as "Hillary Clinton would totally win in 2016" and "that hamburger was totally the greatest I ever had", and rearrange these to construct new phrases such as "Hillary Clinton was totally the greatest I ever had" (Sankin 2014). *Olivia Taters* embodies a chimera of these unknown sources, but problematically never points to their origins, making it appear as though these utterances are entirely original.

Dubbin's project, at the very least, filters the content and is under strict supervision to ensure that these utterances do not overstep privacy concerns, nor spew profanities. *Olivia Taters* was the inspiration for a similar bot called *Tay* (2016), launched by Microsoft. *Tay* equally mangled input utterances from others that it would record through online conversations (Hunt 2016). Similar to *Olivia Taters*, is a sociocomputational assemblage – an interrelational and co-contigent arrangement of human and computational processes – characterised as a teenage girl. This jarring figuration failed to attribute capacity across the interconnected elements that co-contigently enacted Tay's performance. In addition, the project catastrophically failed to filtering abusive language, and was quickly highjacked by people that coerced it into echoing vulgarities and racist remarks (ibid). *Tay* now serves as warning as to how this form of entanglement can be abused. Microsoft's public apology blamed *Tay*'s problems on the "coordinated attack" and "abuses of the system", and declared that this type of entanglement is inevitable in the future developments of software architecture, since they "feed off" others:

> “Looking ahead, we face some difficult – and yet exciting – research challenges in AI design. AI systems feed off of both positive and negative interactions with people. In that sense, the challenges are just as much social as they are technical.” (Lee 2016)

Figure 34. A screenshot of *http://twitter.com/oliviataters*, by Rob Dubbin.

Another way that this entanglement with the online activity of others manifests is through user-generated media. Platform architectures of Web 2.0 were heavily predicated around the notion of co-authored Web content created by userbases, and often were configured to mediate and nurture communities of practice that specialised in a particular medium. Flickr centred around photography, YouTube on video-making practices, and so on. This participatory approach has also been criticised for exploiting content producers by not providing adequate means for claiming recompense, and for failing to protect users in cases of plagiarism. Instead, many tech companies have terms and conditions of service that ask people to give wide-reaching consent to the platform. As such, much media content, although entangled with their human authors, become yet another building block for third-party reuse in the building of new software applications and platforms. During my own investigations into 3D file-sharing platforms, I discovered that many new platforms were harvesting media content from other platforms without crediting their authors.

In artbots, media archives are sometimes used, entangling co-authors of media into the sociocomputational assemblage that it enacts. *Ominous Zoom* (2014) by Casey Kolderup uses the Wikimedia Commons API to source user-contributed images.[103] The Artbot zooms-in on detected faces in randomly selected images with humorous results. *Generated Detective* (2014) by Greg Borenstein generates comics by producing a detecting story and then sourcing relevant images from Flickr, which are then processed through various filters to give them a hand-drawn aesthetic (see figure 35).[104]

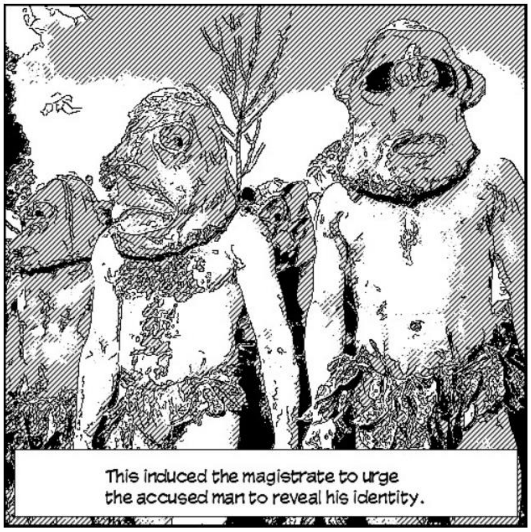


Figure 35. A fragment from *Generated Detective* (2014) by Greg Borenstein. The full comic is available at *http://gregborenstein.com/comics/generated_detective/1/*.

A more direct form of entanglement in software architecture is the configuring of human activity as constitutive processing element in the architecture. Human-based components take on a number of forms that have become encapsulated into API-fronted services like software components. Amazon provides a 'marketplace' for soliciting human activity and call it "artificial artificial intelligence". Google, under the direction of computer scientist Von Ahn (2008), have created their own human components described as "human computation". The politics of this type of entanglement are more pronounced and controversial in situations in which constituent human activity is purposefully concealed to create the illusion that a software edifice is entirely machinic, what Taylor (2018) gives the term *fauxtomation*. Facebook, for example, requires undisclosed

[103] See *https://twitter.com/OminousZoom*.

[104] See Borenstein's description of the process on GitHub, alongside the source code: *https://github.com/dariusk/NaNoGenMo-2014/issues/70*.

numbers of third-party human moderators employed to screen contentious images and adverts posted on the platform; from the user's point of view, human moderation is invisibly integrated into the operation of the architecture (Hopkins 2017).

A tactic developed by Google configures humans to be unaware of their involvement in processing data for training software systems. By using *captchas* (human-solved puzzles to verify a human), Google elicits computational activity from users. The human responses to these tasks then assist in training machine learning architectures. Artists Sebastien Schmieg and Silvio Lorusso began collecting Google's captchas to document these fleeting encounters with problem-solving requests. The collection became an artwork titled *Five Years of Captured Captchas* (2017). The project illustrates the development of AI, capturing the transient demands for data needed to train Google technologies based on machine learning. In doing so, the artists also record their everyday entanglements with Google's AI training architecture.

Figure 36. *Five Years of Captured Captchas* (2017), by Sebastien Schmieg and Silvio Lorusso.

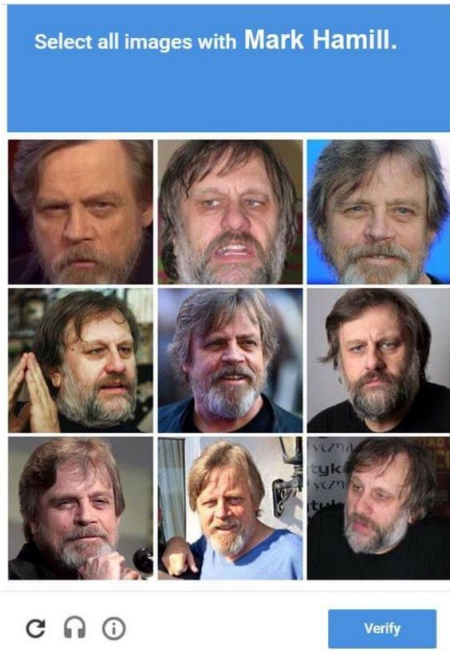


Figure 37. An example of a Google *ReCaptcha* used to train software to distinguish Mark Hamill from Slavoj Zizek.

Machine learning techniques, trained on information that requires vast amounts of human decision-making through menial perception tasks, abstracts those entanglements into software technologies that purportedly run without human assistance. This proxy entanglement can mask the human activity required to support the development of these technologies and the promise of nonhuman automation. In reality, these technologies allow for human workers to go unseen, which can, in some cases, become ethically dubious. Much of that human activity may be activated without fully clarifying the future uses of its contribution.

Machine learning components (and the APIs that interface with them) are becoming increasingly available for third-party re-use. At the time of writing, a new community of software art practice is emerging (unrelated to the artbot community) which specifically experiments with machine learning techniques (Rea 2018).[105] My own artbot *Novice Art Blogger* (2016), which is covered in the next chapter on figuration, uses a machine learning component. *Novice Art Blogger* produces descriptions for abstract art, and this generative output often reveals traces of human-produced image captioning entangled in the training data.

[105] For a list of artists employing ML, see *https://news.artnet.com/market/9-artists-artificial-intelligence-1384207*.

The examples I have described so far are characterised by the inclusion of multiple humans in complex sociocomputational assemblages in which the humans are unaware of their involvement. Alternatively, assemblages may be configured to provide intra-active participation with consent. An example of this, outside of artbot practices, is that of Reddit's mass experiment *r/place* (2017). Reddit is a popular web content aggregation and rating platform, where users vote, comment on, and influence the popularity of linked content. It operates as a massive participatory sociocomputational assemblage for its routine functioning. For April Fool's Day 2017, Reddit launched the webpage *reddit.com/r/place/*, which gave their userbase a digital canvas to draw on one pixel at a time. Different Reddit groups, incited to compete with one another, fought for their own territory on the canvas to leave their mark. Over a million contributors painted and re-painted over each other's pixels, creating a complex and dynamic co-authored artwork (see Figure 38). This is a compelling example of people consciously entangling themselves with a sociocomputational assemblage, illustrating the potential for more inventive forms of designed mass-entanglement.

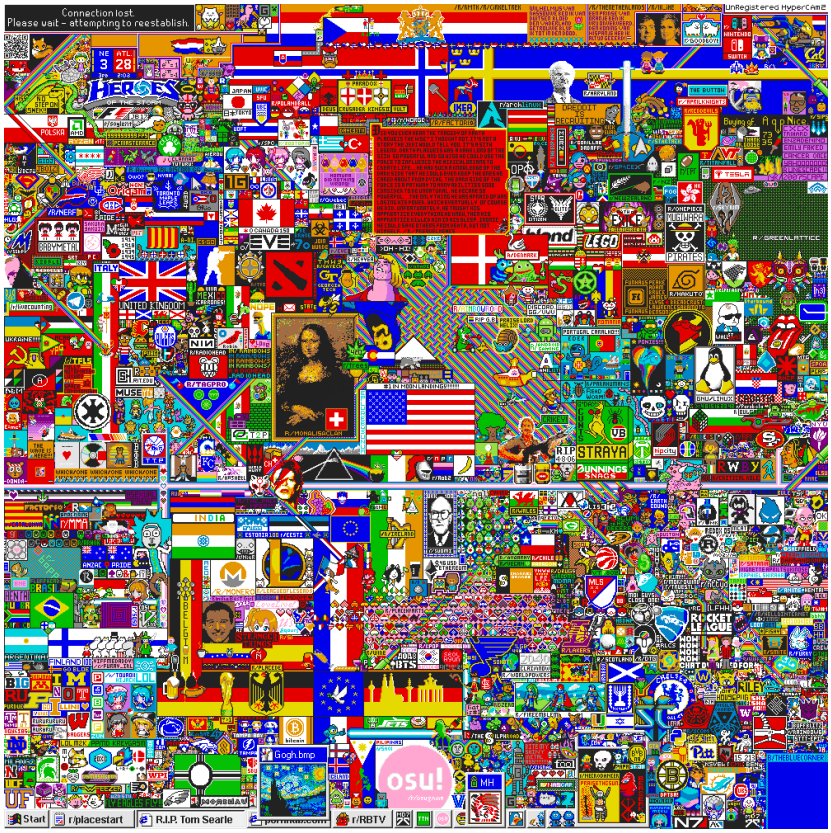


Figure 38. Final image state of *https://www.reddit.com/r/place/*.

In this section, I have highlighted multiple intersections of entanglement between software architectures and human activity – through users, architects, data, media, and

human resources for computation. These multiple points of intra-action within artbot configurations blur previously held distinctions between humans and nonhumans towards a new understanding of entanglement. Characters such as *Olivia Taters* are impossible to disentangle, and should not be understood as fully nonhuman, but as composite hybrids that reveal layers of constitutive elements and processes. It is through these internal entangled processes that artbots emerge, especially in their coding, by which programmer-artists inscribe their ideas into code and configurations. Further to this, these software architectures may only enact certain entanglements with data, users, and other elements during their runtime. A citation from Barad, substituting the word 'individuals' for 'artbots', forms a relevant statement:

> "[Artbots] do not pre-exist their interactions, rather, [artbots] emerge through and as part of their entangled intra-relating." (Barad 2007: ix)

Other than the multiple points of entanglement presented so far, artbots are entangled with people as audiences, spectators to the active configuration that interpret outputs both according to the ways they are presented and in light of whatever they may know about the underlying mechanics. Moreover, entangled audiences provide their own interpretations and readings of the artbot and its output, collectively figuring and presenting the artbot through their sharing, liking and disliking, and commenting on the project. In the following case study of one of my own works, the spectators to the bot's output were also presented archived data about themselves, forming a link between two points of entanglement. Their likes, shares, and comments seemed to play a role in avoiding breaking the rule on contacting multiple non-followers, managing to send unsolicited messages to thousands of Twitter users. In the case study following thereafter, entangled others become so divided by the project that their approving and disapproving comments become an integral part of the overall artwork.

Every User

Every User (2014) is an artbot that finds and lists every single user on Twitter, in the order that they are listed by their unique user ID in Twitter's database. It was configured to multiply into a dense sociocomputational assemblage constituted by as many users as possible, computationally strung together. Moreover, the intervention exposed the pre-existing entanglements of Twitter, its users, and even its founders, illustrating how these are all amalgamated into a database of information that is made accessible for third-party developers through the Twitter API.

Every User's software runs hourly and calls the Twitter API to return data about a single Twitter user, based on their unique user number. Each time the script is run the current number is incremented by one to reach information for the next user in the Twitter database. Once a user has been identified, the artbot sends out a tweet mentioning that username, along with their unique number and the exact date and time they joined the service (the ID numbers are assigned in the order in which users join the service).

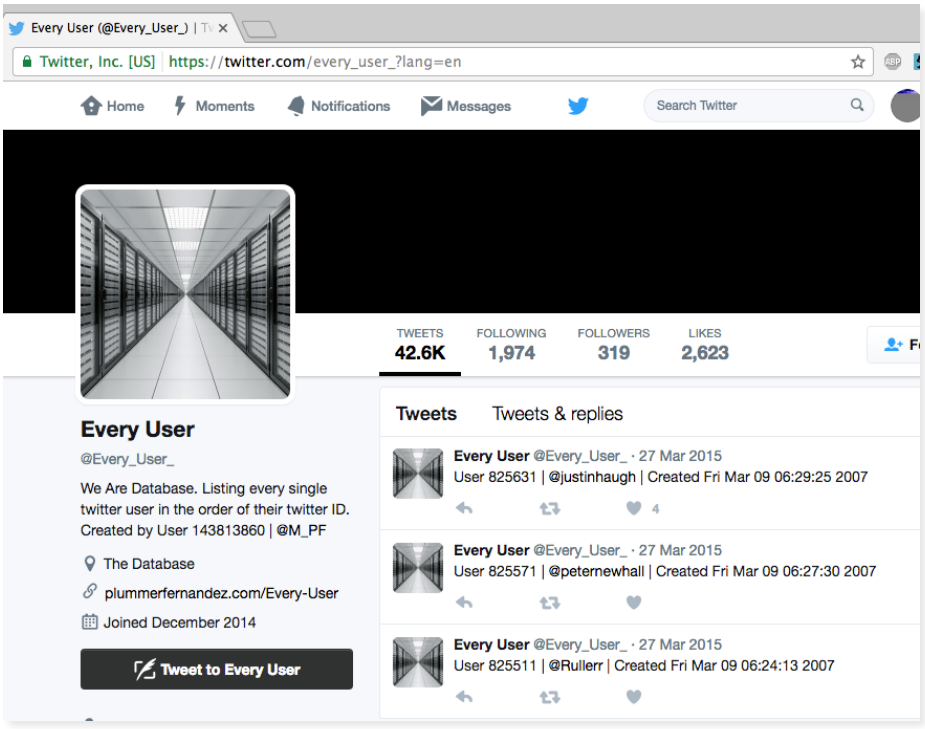


Figure 39. A screenshot of *https://twitter.com/every_user_*. Not all numbers followed succession by an increment of one, as some accounts may had been deleted. Some sections of the database also incremented ID numbers by ten rather than by one.

This configuration ended up producing over 42,500 tweets of aggregated Twitter users. The first three that *Every User* became comprised of were "@jack," "@biz," and "@noah" (see Figure 40). These are the usernames of Twitter's founders Jack Dorsey, Biz Stone, and Noah Glass, whom, according the data revealed by *Every User*, joined the service on 21 March, 2006. Jack joined at 20:50:14, Biz followed at 20:51:43, and Noah joined nine minutes later at 21:00:54. The first user ID is 12, as Dorsey deleted the first 11 test accounts just before the launch that day.[106] Within the first hundred inductees, Dorsey's parents became entangled in the *Every User* assemblage.

[106] Dorsey tweeted an explanation of this: *https://twitter.com/jack/status/49923786786615296*.

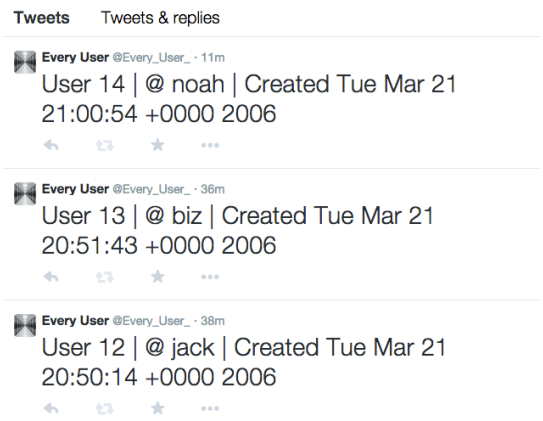


Figure 40. *Every User*'s first three tweets echoing the first three users to join the service: its founders.

Users that had subscribed to Twitter in its first year were formative to Twitter's early history. Many of these early adopters would have been work colleagues, friends and families of Twitter employees, and other people within the same networks and geographic location as the founders. Every User, in a sense, is mimics the process of growing a platform and recruiting humans into an ever multiplying sociocomputational assemblage that is contingent upon the activity of its users as much as its computing processes.

Every User continued to aggregate users right up until user number 825631, who joined the service on 9 March 2007. The date of *Every User*'s last tweet was 27 March 2015, at which point the bot was suspended by Twitter. In total, the ensemble grew to become constituted by most of the users that joined Twitter in its first year of operation, over the course of four months. The project had to be paused at that point, as Twitter had frozen the account a couple of times for sending out too many unsolicited messages to users, breaching the Twitter Terms and Conditions. To protect the account from a permanent deletion, I stopped running the software.

In doing this experiment, I also gained insight into the behind-the-scenes reactions of the users. By logging in to Twitter with *Every User*'s account, the notifications section revealed how people were reacting to their mention (see Figure 41). The reactions were many, ranging from a multitude expressing the project as "creepy" to lots of reposts to use the message as a confirmation of their 'early adoption' status. One user called it the "Twitter numbers station".

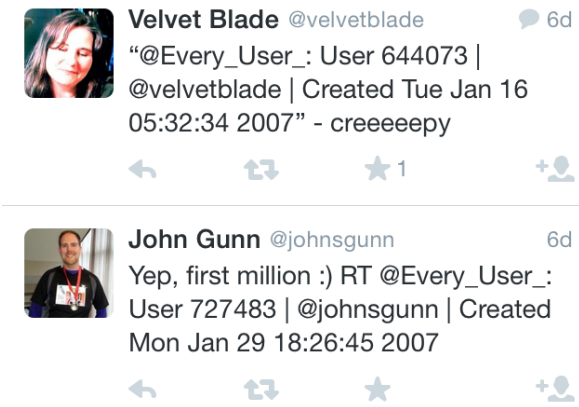


Figure 41. The notifications panel for *Every User*, showing reactions from mentioned users.

```
from twython import Twython, TwythonError
import time

global twitter
print 'twitter client requested'
twitter = Twython('kIxel27nfo3rRABWZsbZkbHTy',  #API_KEY
                  'GqOmjynpV8tlGB1XY5o0rthJIKKIayHpYJPtZdMNGjl0UCikaa', #APP_SECRET,
                  '2915177765-gcpIsQagMAR4tFk8ruprqYd4YgllPHf8cj53xRz', #OAUTH_TOKEN,
                  'Zpiho1YBTgRRIzdfejwdfMH09RSZ30DRLmd2QUkX0YxWI')#OAUTH_TOKEN_SECRET

path = "/home/pi/everyUser/"

#### TWITTER FUNCTIONS
def searchUser(newID):
    print "* Searching User"
    try:
        print newID
        fullUser = twitter.show_user(user_id = newID)
        screenname = fullUser["screen_name"]
        createdat = fullUser["created_at"]
        createdat = createdat.replace("+0000 ","")
        statusId = fullUser["status"]["id"]
        print screenname
        print createdat
        return str(screenname), str(createdat), statusId
    except TwythonError as e:
        print e
        return "noUser"

def tweetStatus(Id,username, created):
    print "*tweeting now"
    msg = "User "+str(Id)+" | @ "+str(username) + " | Created " + created
    print msg
    twitter.update_status(status=msg)
    print "^status updated"
```

Figure 42. Extract from *Every User*’s source code, showing functions “searchUser” and “tweetStatus”.

Every User can be seen as an intervention into Twitter, using the pre-existing entanglements between Twitter, its users, and user data accessed through the API, towards a configuration that puts users into contact with their own data. Learning of the numerical formatting of their activity and identity, Twitter subscribers are made aware of how their data is archived, formatted, and readied for third parties to access, aggregate and act-upon. *Every User* disturbs the rich user experience to provide a hint of the underlying logic and configuration of the platform’s architecture. People’s reactions to this varied from being smug about their early adoption of the platform to being ‘creeped

out', with a lot of ambivalence in between. Whatever their reaction, the commenting, liking, and sharing of *Every User*'s tweets seemed to keep it off the radar of Twitter's monitoring software that would freeze accounts believed to be programmatically sending unsolicited messages to others. Surprisingly, the bot was able to contact thousands of people before finally being warned about its activity. In effect, the burgeoning sociocomputational assemblage of users functioned as a self-ratifying system that evaded disapproval from its host. This property was emergent rather than codified into the architecture, and became integral to the survival of the project.

Every User was an important experiment during this study that cause me to think in more detail about the various relations between bots, people, platforms, APIs, and data. The interconnection between these revealed the various points of contact and entanglement may often be overlooked. The project inspired further exploration of how artbots may use media-data from people, followed by engaging them once more by making the computational processing of their data known. These moments of entanglement, and drawing the connection between the two, help inspire aspects of the next artbot case study.

Shiv Integer

Shiv Integer (2016) is an artbot made in collaboration with Julien Deswaef that entangles authors in the creation of sculptural compositions distributed on a 3D file-sharing platform called Thingiverse.[107] The artbot mashes together disparate 3D files collected from different people into amalgamations that are then reciprocally shared on the platform with generated word-salad titles such as *quick cat near a jaw*, *disc on top of an e-juice golf*, *automatic event adapter*, and *customizable damage mask*. The artbot activated thousands of entanglements with authors of 3D print files who found themselves co-authoring the sculptural mashups, as well attracting hundreds of Thingiverse citizens to comment on the bot's activity. Many derided the bot, finding its operation a nuisance that resulted in useless spam; however, in contrast, some people became fans of the bot, and happily engaged with it in inventive ways. Deswaef and I regarded all of these encounters as a formative part of the artwork, and these entanglements were carried into an exhibition that documented the project. The

[107] The name 'Shiv Integer' is an anagram of Thingiverse. This was kept secret for people to discover for themselves. The idea of using an anagram was inspired by the bot's reason for being: to *rearrange Thingiverse*.

exhibition contained posters of all the comments left by others, an 'ending credits' video that listed all the contributors, and several 3D printed works that were credited to the multiple authors that unsuspectingly provided their constituent parts.

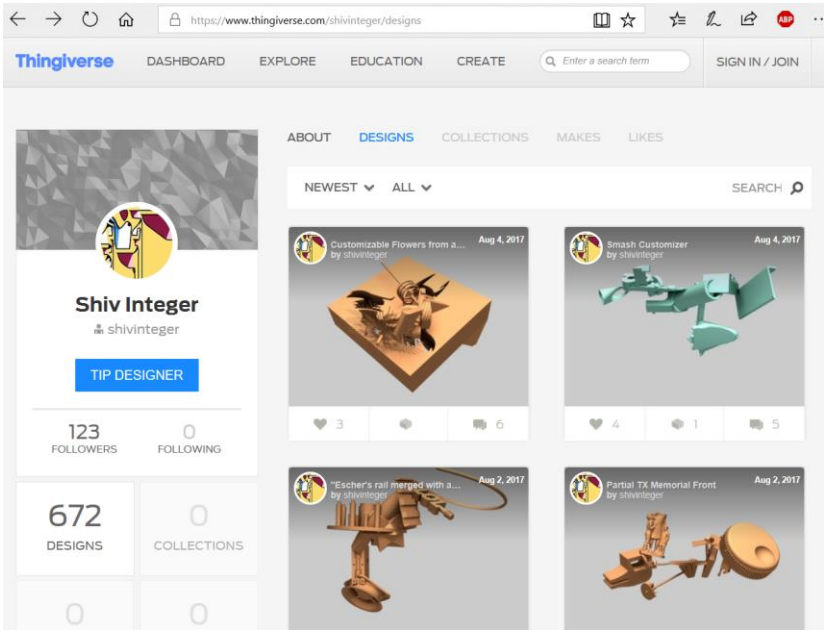


Figure 43. A screenshot of *http://www.thingiverse.com/shivinteger/designs*.

To make *Shiv Integer*, Deswaef and I utilised a multitude of software tools in interrelation, notably the browser-automation software *Selenium*, the *Firefox* browser, Web scraping tools (*Beautiful Soup* and *requests*), together with the 3D modelling software *Blender*. The software architecture opens up a browser, and programmatically explores the Thingiverse website using Selenium. It parses the HTML using *Beautiful Soup* and uses *requests* to download STL files. It also aggregates other information relevant to the STL file, including the name, author and license.

When creating a new amalgamation of 3D objects (an assemblage in its own right) the software architecture automates Blender, conducting a careful sequence of resizing, rotating, overlapping and merging functions. The idea behind the aesthetic was to appear haphazard and clumsy, but the process in fact carefully resizes each object in proportion to the others, and ensures they overlap at the extremities so that all the components remain visible, much like links in a chain. After much trial and error, the results began to take on an aesthetic not dissimilar from the assemblages of contemporary sculptor Anthony Caro. The visual similarities are partly due to the engineering aesthetics of many of the objects found on Thingiverse, such as components for 3D printers.

Much of the work done to assemble objects in Blender was done by Deswaef, while I focused more on the characterisation of the bot. The characterisation of course, was not merely wrapped around the software as a decorative element, but informed choices about how the software should perform. I discuss in more detail, the influence of *figuration* in the following chapter. The artbot was figured in such a way that evaded being explicitly presented as a bot. Instead, an internet mystery was to be engineered by not giving too much away. This direction was inspired by cases of online automation that provoked mystery and bewilderment such as *Webdriver Torso*, an automated YouTube account that baffled people with videos of red and blue rectangles. The mystery account turned out to be a monitoring system ran by Google.[108] The concept of Shiv Integer, hidden behind layers of mystery for others to solve, very much influenced its name, which is an anagram of Thingiverse. The profile picture was taken from a Google search for "Shiv Integer" that picked up a YouTube tutorial on integers.

Shiv Integer can be considered a sort of trickster, mischievously abusing the generosity granted by the Creative Commons licensing system that Thingiverse uses to make it possible for media to be freely shared and modified whilst retaining author-attribution of media.[109] *Shiv Integer*'s software collects and produces a database of 3D files and their respective authors, as well as the Creative Commons terms that those authors have licensed their files with. *Shiv Integer* can then re-use the 3D files whilst programmatically adhering to their licencing conditions. When *Shiv Integer* generates amalgamations, it ensures that the licences of disparate authors are of the same type, and that the newly created artefact carries the same licence, plus the required attribution that evidences the provenance of its constituent files.

As soon as the software architecture became active and visibly stocking its Thingiverse profile with new designs, responses from users began to pour in. Some people were mildly amused and ambivalent. "The crap is going on here? Lol" expressed one individual in response to *Master by a Dual adapter*.[110] Others played along with the

[108] See BBC's exposure of the mystery https://www.bbc.co.uk/news/technology-27778071

[109] Haraway often refers to the trickster as a character and conceptual device. "Coyote is the trickster who constantly scatters the dust of disorder into the orderly star patterns made by the Fire God, setting up the non-innocent world-making performances of disorder and order that shape the lives of terran critters." (Haraway 2016, Chapter 1).

[110] *https://www.thingiverse.com/thing:1424612/comments*

silliness of it. For example, in response to *Green Bench Festivus Stand*, a person reached out to comment:

> "This will make a great stand for the Green Bench Festival next week. Just one question. What colour should I use for the support under the rectangular tray component?"[111]

Others were a little more annoyed and impatient with the bot. The other comment left on *Green Bench Festivus Stand* reads, "Please remove this."

Thingiverse's chief engineer, Tony Buser, got in touch directly over email, and together we settled on a compromise over how much activity the artbot should be allowed to conduct. Buser was concerned about the number of complaints the platform was receiving from users over *Shiv Integer*, but was supportive of the project, and believed it was a creative contribution to the platform.

Multiple users were originally annoyed at the appearance of *Shiv Integer* works on the communal feed on the homepage. It turns out that many users enjoyed having a short moment of fame on the Thingiverse homepage and were disgruntled if they lost the top spot soon after to a *Shiv Integer* creation. Other users and commentators were quick to point out that this perhaps a flaw of the platform, and not of a particular user.

One particular person with the username 'Brightlights' developed a curious relation with *Shiv Integer*, whereby the user would write short stories in response to new creations. The curious stories penned by Brightlights were also to taunt more disgruntled users, offering nonsensical explanations to their queries about the purpose of the project (see Figure 44).

[111] *https://www.thingiverse.com/thing:1670480/comments*

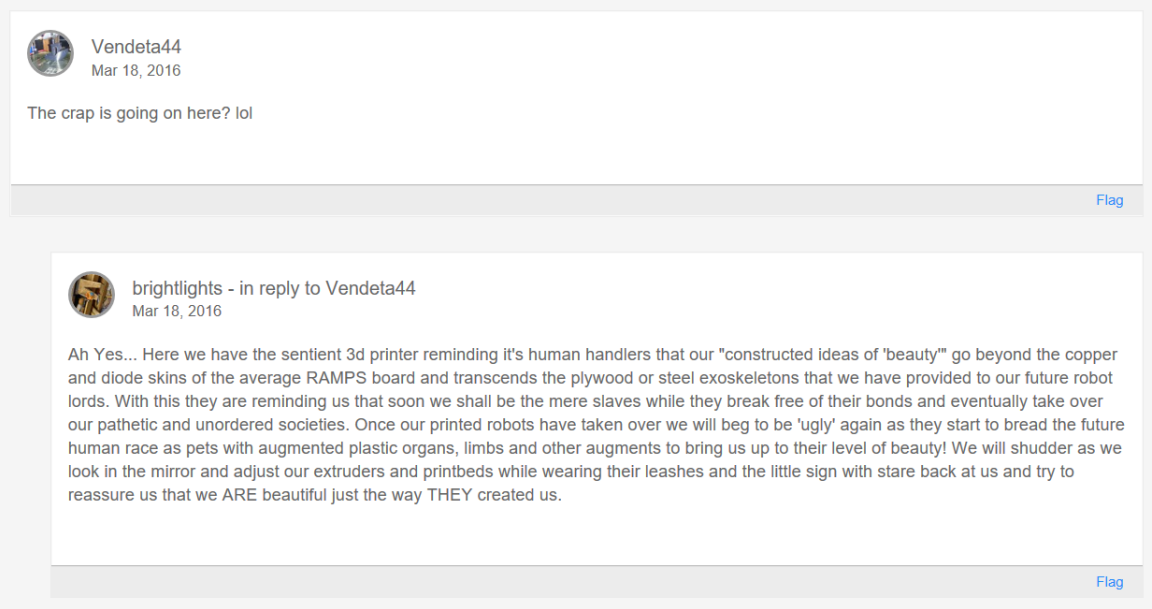


Figure 44. Thingiverse comment left by Brightlights, for the object titled *Master by a Dual adapter*: *https://www.thingiverse.com/thing:1424612/comments*.

The *Shiv Integer* project was documented for an exhibition that took place at iMal in Brussels, 2017 (see Figure 45). The exhibition was an attempt to fully embrace the entanglement of its many co-authors and commentators, and present the work as a complex, multiplying and durational sociocomputational assemblage contingent upon a myriad of entanglements. The comments left by users within the comment sections beneath each object quickly grew into the hundreds. These were then aggregated by a separate script and turned into posters (see Figure 48). Each 3D printed sculpture was exhibited with the full attribution of its co-authors and its Creative Commons licence. Furthermore, an 'ending credits' video was generatively produced to credit every single collaborator for all the collectively authored amalgamations (see Figure 49). To our surprise, the number of co-authors was so vast that the video was two hours in length.

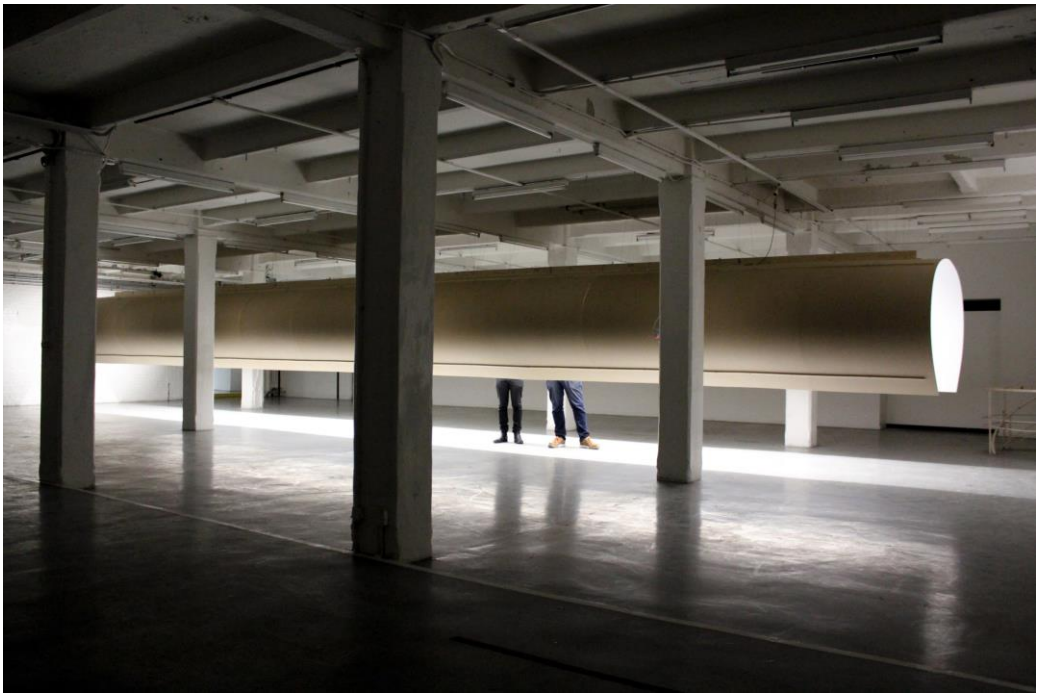


Figure 45. *Shiv Integer* exhibition at iMal, Brussels, 2017, during Art Brussels. The 3D prints were suspended inside a hanging tunnel. Other exhibits such as projections and wall-prints were also presented in the exhibition space.

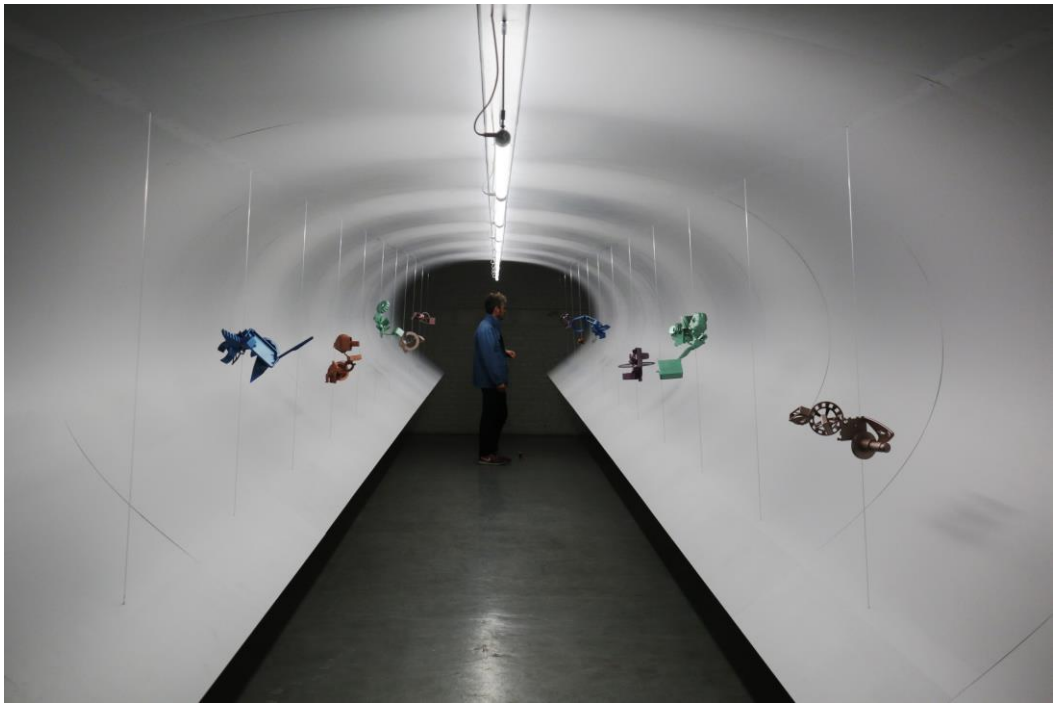


Figure 46. *Shiv Integer* exhibition at iMal, Brussels, 2017, during Art Brussels. A collection of 3D Files printed and suspended within a display tunnel.

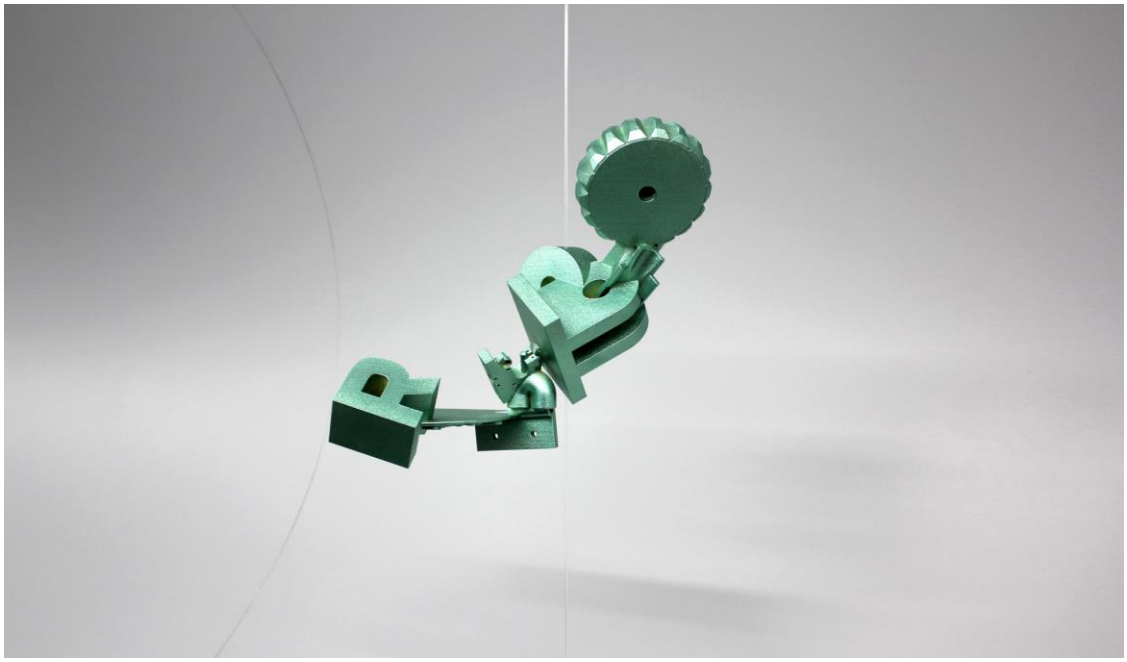


Figure 47. *Shiv Integer* creation titled *Flexible Door 20 x 2*, licensed as 'Creative Commons - Attribution - Share Alike' and found at *https://www.thingiverse.com/thing:1666865*.

Figure 48. User comments made into wall-pasted posters. Many of the posters depict fragments of an art-spam debate between Thingiverse users – hence, some posters simply read "SPAM".

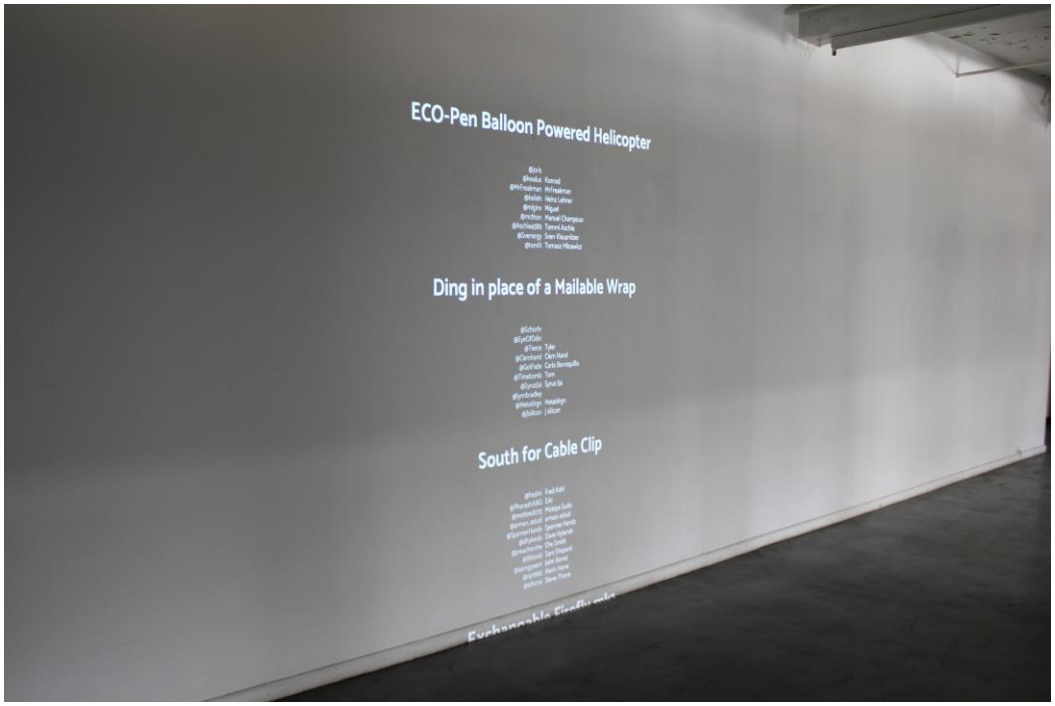


Figure 49. Ending credits video projection at the *Shiv Integer* exhibition.

Shiv Integer helped formulate and demonstrate many of the ideas and theoretical framing developed throughout this thesis. First, it demonstrates the notion of making-with sociocomputational assemblages; it takes a pre-existing interrelated platform-users-media assemblage and intervenes, co-operates and makes-with its constituent parts. Second, its operation becomes ongoing and open-ended, generating a multiplicity of media and further interactions. Third, it entangles authors in the co-creation of composite media, plus a diverse set of audiences including users of Thingiverse, viewers at exhibitions, and other practitioners interested in 3D printing.

Conclusion

Human-computer relationships can be understood to extend beyond simple configurations of user-interaction, into various points of entanglement within sociocomputational assemblages. Using this broader framework, many types of entanglement can be identified and discussed, and this framing helped uncover how artbot practices are experimenting with different modes of human interaction and intra-action. Moreover, artbot practices are often consciously producing the effects of entanglement to highlight its implications.

There are many types of entanglement, and I have described examples illustrating the following types: the entanglement of programmers, human sources of data, user-

generated media, and architecture components that encapsulate human computation. Artbot configurations employ these intra-actions to various degrees and with different understandings of the implications of these forms of entanglement. Scourti, for example, is deeply thoughtful about her own entanglement in the *Empathy Deck* bot, whilst Dubbin rarely expresses, or makes known, how implicated real teenagers may be to the creation of the 'artificial' teenager.

The need to maintain narratives of artificiality and roboticism can contentiously mask the true nature of human-dependent compositions. Microsoft's Tay highlighted the implications of creating a hybrid system under the guise of an AI agent. These implications will continue to create confusion and badly-posed questions regarding the development of AI, if these are continued to be assessed through a theoretical framework of interaction, and the prevailing narrative of human and computer separation. Through the notion of sociocomputational assemblages, more interesting discussions and implications can be raised, questioning where and how human intra-activity becomes contingent in the operation of AI architecture.

While entanglement is not definitional of artbots, varying possibilities along the dimension of entanglement is useful in understanding these, and more experimental artbots are enacting more complex forms of entanglement. Two projects presented in this chapter, *Every User* and *Shiv Integer*, made sociocomputational entanglement a focal point of both their operation and theme. These provoked difficult yet urgent debates about the use of data and the role of computation in modes other than conventional user-software relations. The following chapter on *figuration* looks more closely at how such narratives are deeply ingrained in the collective imaginaries that describe and influence the development of software technologies including bots.

Chapter 6: Figuration

> "[…] initiatives in new technology materialize the cultural imaginaries that inspire them and which they work in turn to enact." (Suchman 2007: 226)

One of the most important aspects of artbot practice is how simple arrangements of internet-based software applications are described in terms of a character. The construct of such characters is both the inspiration for their materialisation into sociocomputational edifices and vice versa, the narrative given to software applications to imbue them with character.

Figuration (Haraway 1996:11; Suchman 2007) is arguably the element that presents a bot as a *bot* rather than a mere software application. The pre-existing cultural imaginary of robots imbues internet-based software with popular meanings and interpretations. In turn, these inscribe and dictate ideas about how software-bots are supposed to materialise, what behaviours they may adopt, what technologies they may be driven by, and what characterisations they may subsume. The figuration of chatbots, for example, has helped inspire the development of language parsing, speech recognition, and synthetic voices.

Artbots can be considered a practice that explores new types of figuration. In this chapter, I move away from the familiar figurations of assistive and humanlike bots and demonstrate that artbot practices are exploring a much wider range of figures. In turn, these figures manifest as more unusual configurations of software architecture, drawing on software components and semiotics that, in combination, are rendered as odd characters or abstract concepts. The value of this lies in the contributions made to the widening of cultural imaginaries that dictate the development of technology by showcasing and prototyping the many other realisations imaginable.

This chapter begins with the familiar figurations that contextualise the need for alternatives. I then discuss case studies, including two projects realised during Art of Bots. Finally, I introduce three of my own artbots that develop distinct approaches to figuration.

Familiar figurations

> "The effects of figurations are political in the sense that the specific discourses, images, and normativities that inform practices of figuration can work either to reinscribe existing social orderings or to challenge them." (Suchman 2007: 228)

Suchman argues that practices of computational technologies are not only informed by figuration, but can often be used to reinforce common narratives, with social implications. Internet-based software applications are no exception; the ongoing figuration of these as *bots* opens up opportunities to imbue them with character. However, this process can also be used to reinforce deep-rooted stereotypes. In this section, I discuss how Web applications are often informed by frequent tropes of figuration, starting with the common notion of bots as humanlike, conversational, and assistive. I also argue that 'the algorithm' could be considered a figuration for Web applications, and I reveal how marketing has helped shape this characterisation. These characterisations not only continue to prevail but are being augmented by advances in software technologies such as voice recognition, speech synthesis, and machine learning. Current home assistant devices are reinscribing the normoactivities of the artificial servant. This provides a context to my argument that artbots importantly eschew these traits towards odd and experimental figurations.

As mentioned in my introduction to bots, the term 'bot' was coined and adopted in early computer network sociality as meaning 'short for robot' – i.e. figuring online software applications as virtual robots (Leonard 1998). Many of these performed machinic duties on behalf of their creators: network administrators created automated moderation software, others created spambots to automate online campaigns (ibid). In addition to this, bots were sometimes deployed as human impersonators for Turing test research (Mauldin 1994). The combination of automating tasks and giving them humanlike qualities (such as Mauldin's bot *Julia*, which would provide directions upon request) can lead to figurations of them as assistive companions in online networks. The idea of having an artificial assistant was not new then; it is arguably one of the more long-standing figurations of computational technology, still pursued to this day.

The figuration of software applications as 'bots' subtly feeds into the narrative of the artificial assistant, with its clear association to robots and ingrained ideas about robots. Robots, in a way, could be considered a figuration of a mechanical contraption as having lifelike qualities, often humanlike, and usually in servitude of humans. Where did this

figuration for a machinery originate from? Remarkably, robots originate from a story about factory-made labourers. Karel Čapek's 1920 science fiction play titled *R.U.R.* (which stands for *Rossumovi Univerzální Roboti*) was about the factory production of artificial humans (of flesh, rather than mechanical), made to serve their human masters (Čapek 2004). These servant class beings were called *Robota*, a Czech word that means 'unpaid labour' (or serfdom), but through Čapek's play and its worldwide performances, the term gained new meaning (Hockstein et al. 2007). The science fiction writer Isaac Asimov adopted Čapek's term, coined the notion of *robotics*, and wrote a set of rules for robotics. Asimov's laws further cement the idea of the robot's role of servitude, stating that "a robot must obey the orders given it by human beings" (Asimov 1950).

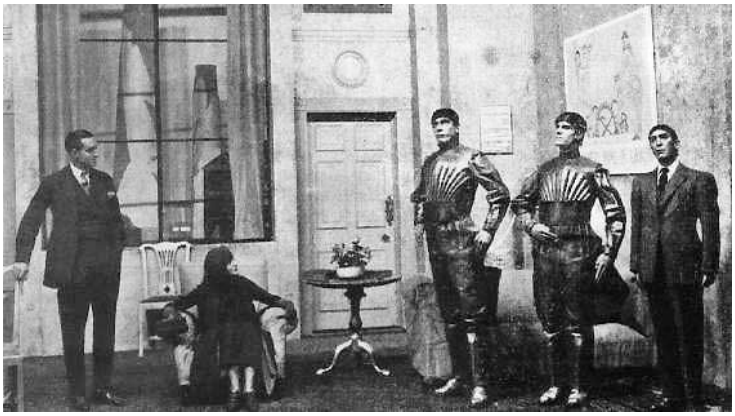


Figure 50. Karel Čapek's 1920 science fiction play, *R.U.R.*

If bots are considered the software equivalent of robots, they arguably inherit these long-standing understandings and take on similar roles. Software 'daemons', for instance, are characterised as software agents that work on menial routine tasks in the background.[112] Early instant messaging chatbots such as *SmarterChild* provided learning assistance and games. More recent examples include *Erica*, the smartphone application assistant for the Bank of America, that can chat via voice or text, uttering phrases such as, "Michelle, I found a great opportunity for you to reduce your debt and save you $300" (H. Taylor 2016). Suchman's survey of chatbot interfaces identifies a number of cartoonish characters, from blobs with behaviours to virtual pets, yet all of these were found to convey common tropes in figuration.

[112] One of the robots in R.U.R was called Daemon.

> "Whether figured as agents, assistants, or pets, their reasons for being are to serve and comfort us, to keep us from being overwhelmed in the future workplace/homeplace of cyberspace." (Suchman 2007: 213)

Further to the figuration of robotic servant, many of these enact gender stereotypes. From Mauldin's *Julia* to *Erica*, to hardware-software hybrids such as Amazon's *Alexa*, many assistive bots are figured as female. These technologies tend to reinscribe stereotypes of women as homemakers or as lower-ranking assistants in the workplace (see figure 51).

Figure 51. A screenshot depicting the description for "Molly", an algorithmic website-design software tool created by *Grid.io*.

Figure 52. Advertisement for Amazon's Alexa, the virtual assistant (software architecture) that is interfaced via the Amazon Echo microphone-speaker device.

Science fiction storytelling continues to inform and figure the development of computational technologies. Computer games are increasingly blurring the line between 'story character' and 'computer character', concretising science-fiction as software. A thought-provoking example of this phenomenon that demonstrates the multiplicity and evolution of software is *Cortana*. Cortana was a character in the Microsoft videogame *Halo*, characterised as a female assistant in the game. In 2013, Microsoft announced that Cortana would retain its name and voice, recorded from voice actress Jen Taylor, for the new Cortana AI assistant for the Windows 10 operating system (Pitcher 2014).[113] Since then, Cortana has been further integrated into several Microsoft products, such as their mobile operating system, their Web browser Edge, and Skype. Its figuration as an operating system assistant follows directly from the game version. Furthermore, the figuration has become contingent on Taylor's entanglement with Cortana.

Figure 53. Cortana in Microsoft's Halo 2 videogame, 2004

[113] Microsoft developer director Frank O'Conner commented "one of the Cortanas is an intelligent, learning AI who is duty bound to help her companion as much as possible, using a staggering database of information combined with real, growing knowledge of that companion — and the other Cortana is, well, the same thing" (Pitcher 2014).

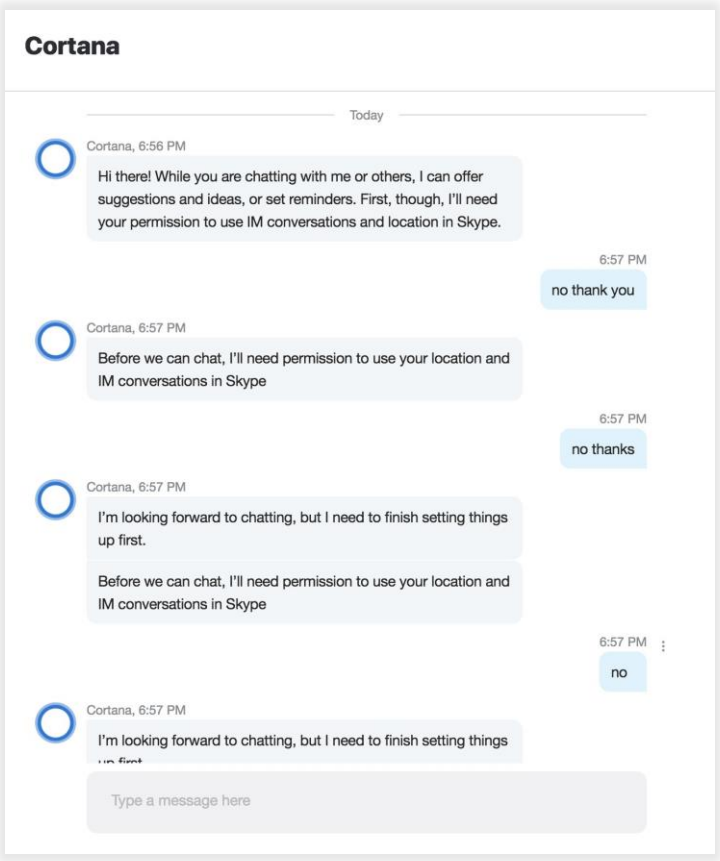


Figure 54. Cortana in Skype, 2018 (Skype was acquired by Microsoft). Image sourced from *https://twitter.com/linguangst/status/1069742586460598273*.

Internet-based software figuration played a role in the early Web. As previously discussed in the introduction, the metaphor of the spider web inspired the figuration of bot spiders, crawling the Web. The Lycos bot created by Mauldin became one of the first search engines for the Web. Even 'engine' is somewhat metaphorical and figurative – suggesting that as software assemblages scale, they may undergo refiguration by design. One search provider employed the servant narrative as their choice of characterisation. *AskJeeves.com*, launched in 1997, centred around the figure of Jeeves, an assistive butler or valet. Later on, the service evolved into *ask.com*, with an overhaul of their software around 2007. Around that time, the company's marketing moved figuration from Jeeves to 'the algorithm' (see figure 55). The algorithm, although referring to a concrete component of the service, was conceivably used as a figural and mythical entity. 'Algorithms', like bots, have become a type of figuration applied to internet-based software applications, and this narrative is often figured to imply autonomy, intelligence, and impartial decision making. Algorithms are figured by what Taina Bucher calls "the algorithmic imaginary" (Bucher 2017: 31), which describes "the

ways of thinking about what algorithms are, what they should be, and how they function". Bucher's study into the Facebook newsfeed algorithm identified that the algorithmic imaginary plays a generative role in moulding the Facebook algorithm itself (ibid).

Figure 55. Billboards articulating the fabled persona of 'The Algorithm', by advertising agency Crispin Porter Bogusky, for ask.com 2007.

Bots, chatbots, algorithms, AIs – assistive, smart, servient, and humanlike – can be considered commonplace figurations of software and computational processes. These metaphors are useful but also have social implications that reinforce certain ideas about technology. Suchman argues that one approach to intervene in these practices is to consider how they might be figured and configured differently (Suchman 2007: 227). As Haraway puts it, "it matters what stories make worlds, what worlds make stories" (Haraway 2016: 12). My argument is that one of the core strategies of artbot practice is to imagine and realise alternative figurations. In this chapter, I demonstrate this point with several case studies before discussing some of my own artbots and how these were figured.

Figuring alternatives

Artbots are figurations of internet-based software applications (configurations of software, data, and APIs) provided with character and a reason-for-being. Unconstrained by the need to provide assistance and convey humanlike qualities, artbots demonstrate a wider scope of roles. The artists devising these are making important explorations into alternative figurations and finding new rationales for software applications running on

the Web. Some provide entertainment, some are poetic, some conduct political activism, and some are simply nonsensical. Importantly, artbots collectively demonstrate how differently internet-software can be figured, and how these materialisations can still interoperate with existing platforms in humorous and subversive ways.

> "The role of conceptual personae is to show thought's territories."
> (Deleuze and Guattari 1994: 69)

Philosophical ideas can be enacted through figures-of-thought. Deleuze and Guattari's notion of 'conceptual personae' maintains that characters can stand in for philosophical schools of thought (i.e. Kant is the figurehead for Kantian philosophy). Some notable philosophers have devised characters to represent and embody their ideas. Stengers (2005) devised the character of the *idiot* to embody her notion of cosmopolitics, Serres (1982) developed the *parasite*, and Haraway (1996) created *Modest_Witness*, *FemaleMan*, and *OncoMouse* to materialise various characters of feminist thinking as well as speculative practices in technoscience. It is Haraway that originally conceives of the notion of *figuration* (p. 11), which to some extent suggests that thinking-through-characters can be materialised in practice. Through the characters of *Modest_Witness*, Haraway's own values and concepts become manifest, at least speculatively. In practices of software architecture, values and design criteria become manifest; in the principles of software architecture, Perry and Wolf (1992) proposed that the architect's 'rationale' is a crucial element in defining an architectural construct.

Whereas many software architects may align their reasoning with either normative tropes or commercial obligations, artbot artists tend to develop their own philosophical and aesthetic rationale. Artist Everest Pipkin maintains that artbots are "automated versions of a specific slice of their creators". Pipkin's practice is exemplary of this approach. Pipkin has a distinct personal aesthetic of ASCII characters and hand-drawn elements. Through artbots, such as *Tiny Star Fields*, Pipkin's personal approach to image-making becomes software-run, generating thousands of permutations – as Pipkin puts it, "the cartographer draws the cliffs that contain a sea of one hundred thousand artworks" (Bucher 2015).

Tiny Star Fields is not simply a software application, but embodies a figuration, taking as elements a slice of Pipkin's own persona, but also that of a night's sky in ASCII. This is vastly different from figurations of humanlike chatbots and instead takes on a nonhuman entity. Perhaps it was Pipkin's own experience of living in the Texan dessert,

when the bot was made (in the nonhuman company of natural elements) that inspired alternative figurations.

Figure 56. *Tiny Star Fields* (2014) by Everest Pipkin. *https://twitter.com/tiny_star_field*

Inspired by pioneering works such as *Tiny Star Fields*, artbot practices have developed into a rich exploration of nonhuman and alternative figurations for bots. Through these, artists embody their own thinking and aesthetic values. And on the contrary, practitioners sometimes create artbots that are figured to parody others or to mock distasteful aesthetics. These alternative approaches to figuring are significant in expanding ideas of how Web applications and edifices can be conceived.

Many figurations are based on nonhuman subjects, such as *Big Ben Bot* (2009) (which tweets "BONG" on the hour),[114] *A Real River* (2014) (an ever-elongating river of emojis),[115] and *Earth Rover Bot* (2014) (the virtual rover wandering through Google Street View).[116] These, and others, uncover the potential for various modes of narrative and situatedness, manifesting as things that can be either associated with fictive worlds, virtual worlds, or the physical world.

[114] It tweets the corresponding number of 'BONGS' according to the time. See *https://twitter.com/big_ben_clock*.

[115] See *https://twitter.com/ARealRiver*.

[116] See *https://twitter.com/EarthRoverBot*.

Figure 57. *Lost Buoy* (2013) by Mark Sample. *https://twitter.com/_LostBuoy_*

Lost Buoy (2014) by Mark Sample is an artbot figured, as the name suggests, as a buoy, adrift but emitting readings. These readings, however, are not just numerical oceanic data, but a mix of data with speech, uttering, for example:

> "The air, lass, you breathe it and it is 76.5°F. Oh, hard! No!"[117]

Sample (2014) explains that the software architecture behind *Lost Buoy* takes readings from Station 51000, an actual buoy in the Pacific, as input, whose data is made available online by the National Data Buoy Center.[118] These are then combined with fragments from Melville's novel *Moby Dick*. As such, this figuration ambiguously combines fiction with nonfiction, and imbues a real oceanic buoy with the character of Ishmael, the narrator of Moby Dick, who survives the devastation of a whaling boat, and is left adrift until rescued. The Moby Dick element, however, is not explicitly made known within the bot's description. The artbot becomes figured by both story and data in which the two intervene-in one another. Through *Lost Buoy*, Sample suggestively explores alternative figurations for 'internet-of-things' (IOT) devices. In this case, Sample draws

[117] *https://twitter.com/_LostBuoy_/status/929532102739152897*

[118] Found at *https://www.ndbc.noaa.gov/station_page.php?station=51000*.

out from literature a voice of despair, rather than giving an internet-enabled thing the more typical persona of an upbeat assistive appliance.

Similar to *Lost Buoy*, *Lost Tesla* (2016) by Kate Compton is an artbot figured as a self-driving car manufactured by Tesla, tweeting about its ongoing experience of trying to get home.

Figure 58. A screenshot of *http://twitter.com/losttesla* (2016) by Kate Compton.

However, rather than roaming a physical space through real-world data, the character roams a diegetic space and provides hints of this world, describing "a quiet town" and "a school child by a house" (see Figure 58). This figuration augments elements of what audiences may already know about Tesla, such as their vehicle's advanced automated parking, sensors, and satellite-aided navigation. *Lost Tesla* extends the materialised figurations of Tesla futurism, adding a fictional world around it that are observed through the speculative car's sensors – providing it with fictional sensory experiences. Interestingly, fans of the bot can jump quickly between suspending disbelief and seeing for themselves the code that dictates the bot's behaviour. Compton provides a link to the source code of the bot, which is written in Tracery (a programming tool also developed by Compton).[119] Seeing the code quickly demystifies the bot, revealing, for example, that its utterances about self-awareness are described in a line of code:

[119] *Lost Tesla*'s code is available at *https://cheapbotsdonequick.com/source/losttesla*.

"sawReflection": ["i #see# my reflection in #shinyThing#. #self-aware#"]

There is an appeal in following roaming figures – lost, wandering, meandering, or reflecting on what they observe. A number of other artbots explore this theme, such as Parrish's *The Ephemerides* (2015),[120] annotating the images captured by a space probe with generative poetry. Parrish's figuration draws on contemplative and minimal poetry:

> "[B]oth space probes and generative poetry programs venture into realms inhospitable to human survival and send back telemetry telling us what is found there […] My thought was: space probe poetry would be minimal, contemplative, turning suddenly from the technical to the lyrical and back. The poetry of Bashō, Gary Snyder and Rae Armantrout came to mind as potential points of reference." (Parrish 2015)

Another contemplative artbot processing the output of utilitarian cameras is *Treasure Column* (2016) by Derek Arnold.[121] The software architecture for this bot generates abstract films out of footage from unsecured network cameras, available online, that have been set up to watch over things rather than people (Arnold 2015).

> "My favourite cameras function just to let the owner know that an object still exists and is functioning. I have a warm place in my heart for this strangely composed camera aimed at a telecommunications tower in Arizona. The sun's angle crosses the field of vision of the camera during the afternoon. Its bright light overloads the sensor, creating a strange solar eclipse effect due to either equipment failure or a design flaw." (ibid)

[120] Found at *https://the-ephemerides.tumblr.com/*.

[121] *Treasure Column* is found at *https://www.youtube.com/channel/UCKNW6jeGUfPUg_UsyAsTaPA/feed*. Arnold's previous work at *https://www.twitter.com/FFD8FFDB* is very similar and gained much popularity.

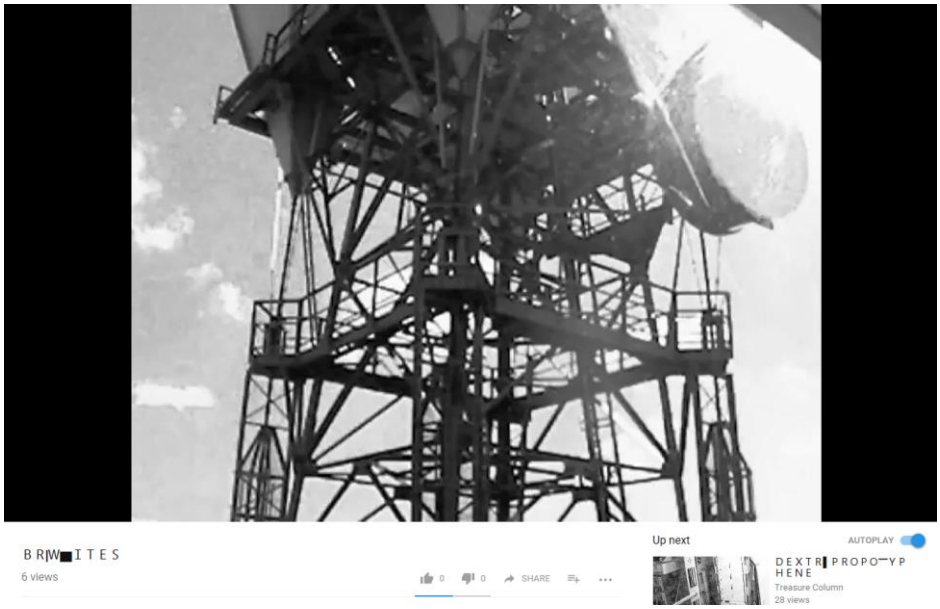


Figure 59. Screenshot of Treasure Column video:
https://www.youtube.com/watch?v=bvNrGZoH2XQ

The ambient music of *Treasure Column* that is generated to accompany the footage is sourced from royalty-free sounds from the Internet Archive, which are then slowed down.[122] Unusual names are also generated for the videos by taking words and substituting random letters with ASCII characters. In addition, random fragments of text are posted alongside the videos as obtuse descriptions. Each YouTube post is an amalgamation of these disparate elements. *Treasure Column* re-figures readily available internet-networked security cameras as a meditative appreciation of the stillness of things. Through the semiotics of ambient music, the careful choice of camera feeds, the black and white filtering, and the meaningless titles, this configuration is re-imagined as the dreamy gaze of the internet-eyes to the world, marvelling at the banal treasures of human-made world.

Figurations of software architecture can be strongly associated with their input media, but also with the algorithmic processing of these inputs. With *TV Comment Bot* (also known as *TV Helper*), David Lublin figures an artbot as a 'helpful' interpreter of television – perhaps mocking assistive technologies by making a nonsensical one. The bot works by processing TV footage through image recognition software to generate captions based on what it identifies in the image (see Figure 60). This results in absurd subtitles placed over US-based TV channel screenshots. In addition to this, *TV Helper*

[122] The Internet Archive is a database of public domain media found at *http://archive.org*, and Arnold slows these down using a software toolset called Sox.

also exists as a physical appliance, constantly watching TV in Lublin's apartment in New York.[123]

Figure 60. An example output of *TV Helper* by David Lublin.

TV Helper's figuration is parodic of 'smart' assistants and image recognition applications, as well as of TV commentators that are sometimes similarly having to offer narration on broadcasted events. Parody has been a consistent strategy in the artbot community, and early Twitter experiments were often figured as parodies, such as *Bot Marley* (2010) by Argentian botmaker JED,[124] which generates Bob Marley-like lyrics. Other early parody accounts were figured after philosophers such as *Zizek eBooks* (2011), which has over 17,000 followers.[125] Many of these used Markov chain software to programmatically make word salads out of texts. Thus, in a sense, these parodies are not just figurations but entangled figurations; *Zizek eBooks* is contingent upon the writings of Zizek.

An advantage of parodical figuration is that much can be conveyed and figured by association. *Endless Screaming* (2015) by Nora Reed is an artbot that tweets "AAAAAAAAAHHHHH" (in varying combinations of As and Hs) and also replies to followers with such screams. The profile picture borrows *The Scream* (1893) by Norwegian expressionist Edvard Munch. In doing so, some of the artbot's figuration and meaning is borrowed by association.

[123] See *http://www.davidlubl.in/blog/2015/tvcommentbot*.

[124] See *https://twitter.com/bot_marley*.

[125] See *https://twitter.com/zizek_ebooks*.

Some artbots are figured as political parodies. *Theresa May Bot* (2013) by the artist Shardcore is a parody of Theresa May created when the politician was Home Secretary of the United Kingdom, shortly after she enacted the Communications Data Bill (also known as the Snooper's Charter).[126] The bot adds strangers to lists with titles such as "HMRC Audit" and "Candidate for extradition". Shardcore explains that the parody is more than that of a political figure; the sociocomputational assemblage as a whole, in its operation and aggregation of others into lists, parodies "the complex data-driven surveillance programmes operated by GCHQ".[127] In the same way Theresa May was the figurehead for online surveillance programmes, the *Theresa May Bot* figure fronts a constellation of software processes.

Artbots have been figured as misfits, parodies, tricksters, idiots, parasites, and poets, contributing to a rich taxonomy of internet-based software materialisations. These experimental devices, in turn, help broaden an understanding of figuration and how it is performative in various practices of software technologies.

Artbots effortlessly traverse ontological complexities, experimenting in the liminal overlaps between human and nonhuman, figural and material. Creative figuration allows artbot practitioners to avoid the entrapment of dichotomous positioning and find new ground in which things can be both human *and* nonhuman, both fictitious *and* materialised. For the artbot spectator, there's an appeal in such complex layering, allowing the observer to jump between appreciating the diegesis, the underlying mechanical architecture, and the involvement and intentions of the author as well as implicated others. However, these layers may not always be immediately noticeable. It is my hope that the framework and understanding developed in this thesis will support better understandings of this art form.

So far, I have briefly described many examples of artbots to suggest the variety of figurations at work. In the following section, I spend more time examining the entanglements and figurations of two very different projects, realised in very different media, that both appeared at Art of Bots.

[126] See *https://twitter.com/theresamaybot*.

[127] Quote from Shardcore's website: *http://www.shardcore.org/shardpress/2015/05/13/theresamaybot/*.

Magic Realism Bot

> "Fifty philosophers meet once a year inside a ruined motel to decide the fate of time and space." — *Magic Realism Bot*, 17 August 2016[128]

Magic Realism Bot (2015), created by siblings Chris and Ali Rodley, has an impressive popularity of over 100,000 followers, and is held in high regards by peer artbot practitioners. The artbot generates magic realist stories similar in style to Jorge Luis Borges by randomly combining grammar templates and a corpus of possible protagonists, places, concepts, and events. These are cleverly pre-penned to evoke the Borgesian model of making the ordinary extraordinary.

Figure 61. *Magic Realism Bot* (2015) by Chris and Ali Rodley. *https://twitter.com/MagicRealismBot*

Chris Rodley recalls being enchanted by Borges as a teenager, especially the approach to stories that "stretch concepts or structures to their logical endpoint; that embrace infinity" (Rodley 2016). Rodley would write ideas for stories using this pattern and decided that the process could be software-run. His sister Ali Rodley wrote the code, and Chris wrote the syntax structures that would provide the basic framework for a story, and also devised the ample vocabulary lists for the stories.

[128] *https://twitter.com/MagicRealismBot/status/765700927902801920*

> "Once the template is selected, it is populated with words from our pretty large vocabulary list, which is categorised and tagged into particular types of people, places, things and actions: for example, rulers (like queens or presidents), animals, gemstones, books, magical things, things that are long, things to do with space, transitive verbs like kissing and killing, and intransitive verbs like singing and sleeping […] Probably our two most fruitful and frequently used categories are simply "concrete things" (like clocks or swans) and "abstract things" (like love or capitalism)." (ibid)

Rodley may have set out to parody Borges (it was originally going to be called 'BorgesBot') but soon realised the potential for it to have its own distinct voice, with a broader range of influences including Calvino, Eco, García Márquez and Allende, together with Lewis Carroll, C. S. Lewis, Philip Pullman, and Richard Hughes (Rodley 2015). In addition, the Rodleys' own authorship comes through, entangled in the sociocomputational assemblage of *Magic Realism Bot*. When asked how the authorship of the stories is distributed, Chris Rodley ascribed authorship to both themselves *and* the software, and suggested that a new posthuman notion of authorship is urgently needed (Rodley 2016).

The figuration that the Rodleys allow to emerge has hints of its literary references as well as their own authorship, but at the same time, they are cautious not to construct a humanlike figuration. Rodley points out that the artbot would often generate serendipitous combinations of corpus elements that were different to the choices of a human author; for instance, inserting an abstract concept where a concrete concept would be expected, such as "A Madagascan Young Adult writer is having a passionate love affair with capitalism".[129] Chris Rodley is critical of bots figured to be humanlike, and is rather more interested in "this distinctive robot imaginary" (2016).

[129] Posted 30 August 2016, *https://twitter.com/MagicRealismBot/status/770563264732200960*.

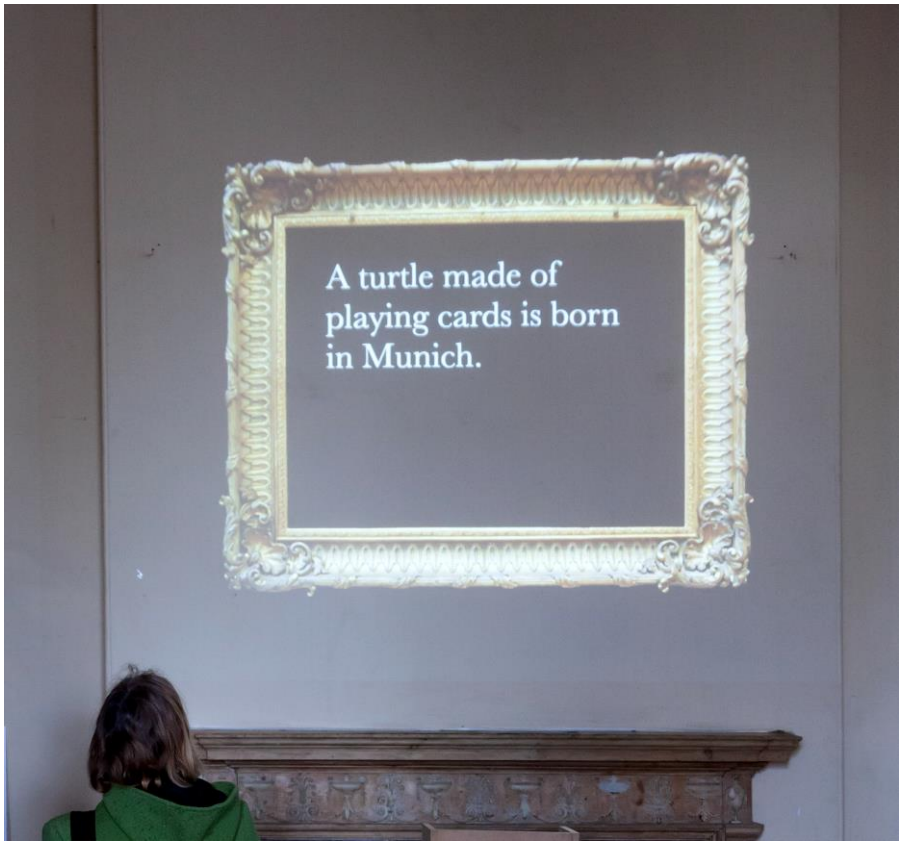


Figure 62. Installation of *Magic Realism Bot* by Chris and Ali Rodley at Art of Bots, 2016.

Magic Realism Bot was included in Art of Bots, for which it was reconfigured as a ghostly generative projection above a fireplace. The showcase gave the Rodleys a chance to experiment with a realisation that was more in-keeping with the magic realist aesthetic.

The artists made use of the semiotics of the space at Somerset House; the room was of both antiquated grandeur and slightly run-down. The projection was presented above an old marble fireplace, and included an ornate frame around the stories, which enabled the work to relate to its surroundings. The surroundings were not treated as a blank canvas but were incorporated into the figuration of the artwork. The software was also modified to display the story piece by piece, revealing different choices for characters and locations, and finally settling on an output. As such, the work reveals its inner workings and chance operations. Rodley explains that these choices were made to help demonstrate how the software contributes to the story-making and help "illuminate its machinic nature" (ibid).

In this configuration, the artwork becomes separated from the mediation and aesthetics of Twitter and becomes, in itself, a magic realist object – a meta-narrative device that

constructs narratives. Such items are present in magic realist stories; for example, the Aleph (Borges 1971) is a crystal ball in which everything can be seen. It is easy to forget that this is still a form of software architecture – far-removed from characterisations of humanlike or robotlike, this automaton becomes ephemeral and dreamlike.

Magic Realism Bot is constructed not to forefront the roles of the Rodleys, nor to present a strong (human) personality of its own, nor to suggest a machine, but instead, to evoke a storytelling device – an encapsulated virtual library that contains the parameter space of all possible magic realist stories. Stories randomly surface from its virtual vault to make contact with the world. The authorship of these stories cannot be disentangled but are contingent upon the multiple hands, voices, writings and code that constitute the sociocomputational assemblage that penned them.

MAMMON

MAMMON (2016) was created by the artists Shardcore and Sam Hewitt for Art of Bots.[130] Shardcore describes it as the "algo-god of Capitalism" and it takes the form of giant 'maneki-neko' cat figurine that generates and reads out humorous advice on becoming better capitalists to kneeling humans. When I approached Shardcore to contribute an artbot to Art of Bots, I was aware of the artist's many Twitter bots, but the artist's collaborative works that incorporated sculpture and street performance were less well known. This presented the opportunity to experiment with artbot figuration, probing the question: Could an artbot's character be a physical entity? This examination was not about transforming a bot into a robot, but instead producing a physical avatar for a software application with the purpose of imbuing it with character and realising its figuration.

[130] Shardcore also goes by the name Eric Drass (both are artist pseudonyms); his real name is Neil Forrester.

Figure 63. *MAMMON* by Shardcore at the Art of Bots Showcase, 2016.

Shardcore and I were in conversation in preparation for the showcase and discussed various works. His collaborative work with artist Sam Hewitt, under the name *Fortunecat Productions*,[131] was more oriented towards physical exhibitions and performances. Although it had little to do with bots, the artwork *Fortunecats* (2009) stood out as an incredibly rich project that combined moving sculpture, street performance, theatre, actors, and a strong fictional narrative. Created for the White Night Festival in Brighton, *Fortunecats* was a participatory work in which members of the public could have their questions answered by two giant cats. The philosophical answers were read out by voice actors (hidden and playing the role of the cat). These answers were collated in advance from excerpts from philosophy, poetry and advertising. I asked Shardcore if the work could be reconfigured to be software-driven for the showcase. The artist not only created a software-based system for the work, but refigured the (now singular) cat into *MAMMON*, the algo-god of Capitalism. Shardcore's reworking of the piece illustrates the artwork's reusability, evolvability, and adaptability. Further to the multiplicity of its arrangement and figuration as a cat-god entity, the artwork manifests through its entanglement with human participants.

[131] See *http://fortunecatproductions.com/the-history-of-the-fortunecats/*.

In a deep, authoritative voice (from pre-recorded audio), *MAMMON* asks a participant to kneel before it on a red carpet, then introduces itself and its quasi-capitalist beliefs. The cat's giant raised paw mechanically moves back and forth in a beckoning gesture whilst asking the participant to repeat after it a personalised declaration, summoning the participant to be more capitalistic.

The artwork demonstrates how figuration comes together through the assembling and realisation of various elements: the voice, carpet, cat, software, and human subjects. The entanglement with a voice actor provides it with a human voice that becomes its own. The artwork could have used a 'synthetic' voice instead; however, even synthetic voices are synthesized from pre-recorded audio from voice actors (such as Microsoft's Cortana). Many off-the-shelf synthetic voices are instantly recognisable, as these often get repurposed for satnav applications and other ubiquitous voice-assistant technologies. To give *MAMMON* a particular characterisation, the artist had the various responses pre-recorded in the desired voice.

MAMMON was an experiment in software figuration, using tongue-in-cheek behaviours, symbolism, and sculpture as elements of its characterisation. Its mechanical arm does not come across as robotic but more like that of a theme park *animatronic*.[132] Bots are perhaps closer to animatronics than robots – their repetitive, mechanical actions are transparently non-intelligent to the spectator, yet help bring a character to life. *MAMMON* bridged two closely-linked practices, and more could be developed in this direction. Other Shardcore artbot works manifest online and illustrate the artist's broad interest in developing characters.

Shardcore stands out for his ample experimentation with instilling artbots with identity. His *@bbfbot1* (2014) parodies an over-enthusiastic and needy new 'best friend' follower. This artbot goes by the name *Alex* and sports a profile image of a young woman with digitally-bulging eyes with the description, "Alex -Your new best friend" (see Figure 64). The combination of these elements as well as the software-driven behaviours designed by Shardcore created the intended persona:

> "[…] an enthusiastic new friend, she gives you all the social media love you're looking for. However, over time, she reveals a slightly more obsessive side […] Alex loves you, in a way that only an algorithmic entity can." (Drass 2016)

[132] 'Animatronic' is a portmanteau of 'animate' and 'electronics'.

Figure 64. Screenshot of *http://twitter.com/bffbot2* ('twitter.com/bbfbot1' was suspended by Twitter, so a new account was created)

Bffbot1 had programmed behaviours such as nudging users to engage with it for attention – "why are you ignoring me?" It could also generate personalised greeting cards and poems for its followers – "I wrote you a poem" (the poem is sent as an image to appear as if written on paper). These cards and poem-images are generated using image-making software modules, as well as finding the source material from the follower's Twitter profile using the Twitter API. This involves searching for the user's past tweets towards penning poems that can embarrassingly bring up the past. *Best Friend Bot* is a parody of creepy online followers, but perhaps also a parody of the practice of figuring chatbots as always-available female assistants.

Another artbot by Shardcore is *@trippingbot* (2015). *Trippingbot*'s profile picture is a portrait of the American actress Yvonne Craig starring in a Star Trek episode as a green-skinned alien named Marta (see Figure 65). The Twitter account's description reads, "on a trip from 6pm-12am GMT every day." *Trippingbot* is an assemblage of various components of machine learning, Twitter APIs, image-making software, and input text from drug-taking transcripts (as well as semiotic components such as profile name,

description, and image). This assemblage renders material the figuration of Marta, the tripping bot.

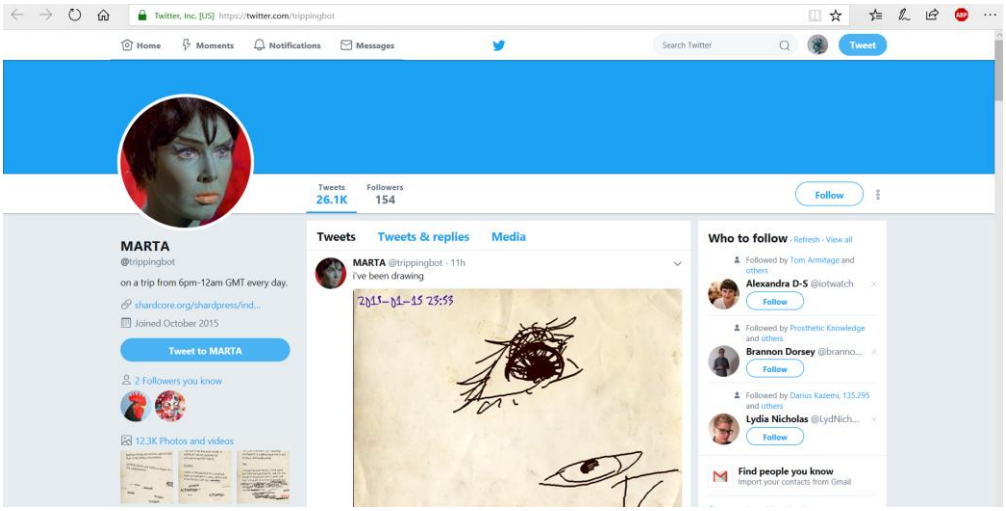


Figure 65. Screenshot of http://twitter.com/trippingbot

The artbot utilises a *character level neural network* trained on input text, character by character. The input text was sourced from *erowid.org*, a website in which users describe their drug experiences. As a result, *Trippingbot* begins to generate its 'own' drug-fuelled experiences. Of course, these generative texts are entangled with and contingent upon human drug testers.

Shardcore cleverly conflates the myths and confusion around neural network technology that give the impression that a 'brain' has been successfully simulated. The artbot's 'neurons' are tested with mind-altering drug experiences. The software assemblage also generates animations that depict the scribblings of Marta under the influence.

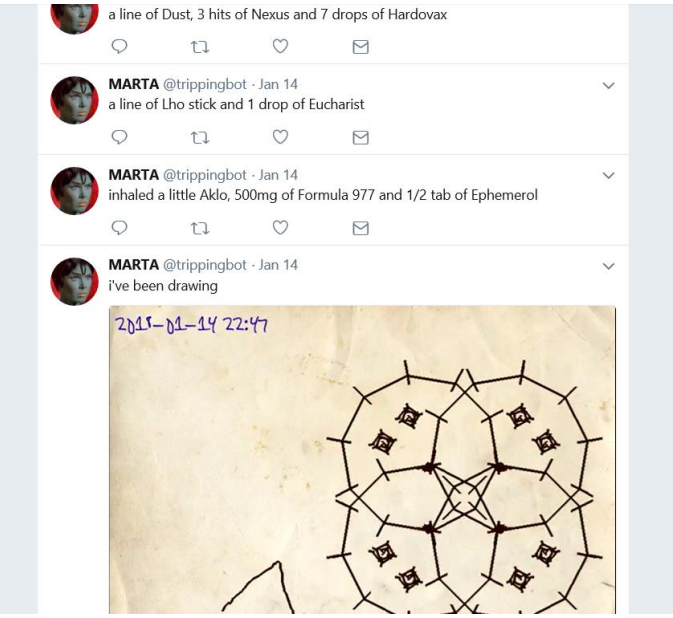


Figure 66. Screenshot of @trippingbot's Twitter timeline.

Shardcore's software-run personae, often purposefully unpleasant and contentious, sit in contrast to generative art practices that favour pleasant figurations of software-run painters and the like. Shardcore's practice is closer aligned to the use of pseudonyms for art-activism and punk personae, creating antisocial entities that are rendered capable of subversive strategies and mockery, that when placed in arrangements with relevant software components, constitute heretical sociocomputational assemblages.

The Revenant

The Revenant (2016) is one of my own artbots created during this study. This artbot was a quick sketch in comparison to more elaborate projects conducted for this study. It helped me learn and understand the craft of a bot parody, and employed new strategies such as deviating from the original persona. This artbot is the first of three of my own that I present in this chapter, and each touch on a different approach and understanding of artbot figuration.

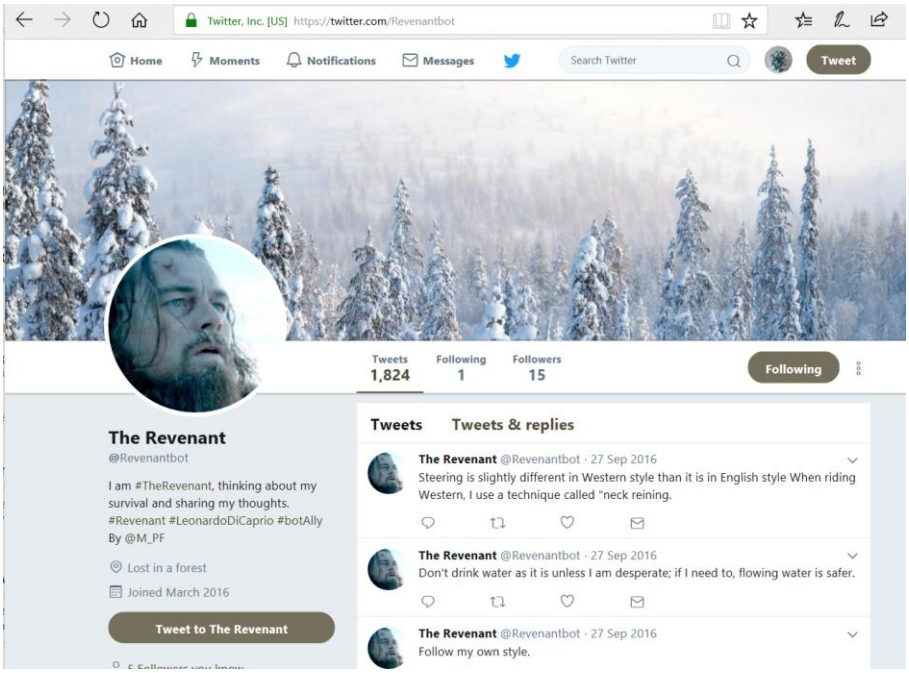


Figure 67. Screenshot of https://twitter.com/Revenantbot

The Revenant is based on the 2015 award-winning film of the same name, directed by Alejandro Iñarritu and starring Leonardo DiCaprio. The bot was created shortly after DiCaprio won an Oscar for his role, as well as Iñarritu winning Best Director. The film tells the story of Hugh Glass, a frontiersman who is left for dead after suffering a bear attack, yet makes it back alive using his survival skills. Glass is also grieving for the death of his son, who is murdered by the colleague who betrays and abandons him in the wilderness.

The automated Twitter account takes on the persona of Hugh Glass – or more specifically, the Hollywood interpretation of Glass. The profile picture for the account is of DiCaprio, in character, looking focused on survival. The description reads, "I am #TheRevenant, thinking about my survival and sharing my thoughts." The profile location is set to "lost in a forest". Many artbot parodies rely on generating utterances in the style of the imitated person by parsing original material through a Markov algorithm or creating grammar recipes that imitate linguistic patterns and keywords. With the Revenant bot, the character was purposefully given a new 'voice' – one that was suggested to be the internal thoughts of the character.

The Revenant's utterances were constructed by taking as source, survival guides from the website WikiHow.[133] WikiHow is a platform that allows people to create and share help guides for everything from how to do basic carpentry to how to change your life around. For the Revenant bot, the site was manually perused for survival guides that mirrored survival activities seen in the film – encountering a bear, surviving a fall, foraging, getting lost in the woods, and riding a horse. In addition, the guide for dealing with family bereavement was also included, and for comical effect, a guide to dealing with a difficult dog was utilised and processed to substitute 'dog' with 'bear'. The WikiHow guides are generally written in the third person, advising others on how to survive. These were automatically changed to be in the first person using a software script, so that the artbot appears to be narrating its own survival strategy. A list of grammar substitutions was scripted in Python, and the input text was parsed. Finally, the text was split into sentences that would fit into tweets, and then shuffled.

The result is a figuration of a character that seems to be advising itself. Rather than evoking struggle, it expresses plans and tips with optimism.

> "I can survive 2-3 weeks without food."

> "There are three things that I will absolutely need if I am going to survive: appropriate shelter, fresh water, and a latrine."

The instances of bereavement and grief, however, can give the persona a different twist to that confidence:

> "Scheduling an appointment with a professional is not an overreaction to my loss."

The intention was also to make it seem as though the Revenant was inside the narrative world of the forest, describing his surroundings and survival activities within that space. By appropriating a character from a movie, associations can be made between the bot's utterances with scenes from the film (granted that the spectator has seen the film).

Unlike *Magic Realism Bot* or *MAMMON*, *The Revenant* is figured with a humanlike persona. But this is an unusual human to be tweeting – lost in the forest, concerned with

[133] WikiHow is accessible at *https://www.wikihow.com/Main-Page*.

survival (in contrast to more typical, and trivial, Twitter concerns), and speaking as much to himself as the reader. The bot helped examine the intersection between Hollywood characters and Twitter bot parodies, and although it was an enjoyable challenge, I found creating a character based on an existing one slightly limiting. In the following case study, I create a character from scratch, and additionally situate it on a different platform than Twitter.

Petita Tatata

> zimzim zimzim
> gigedi gihagi gigoda gihiga
> gigedi gihagi gigoda gihiga
> pappap pappap
> zimzim zimzim
> zimzim brussala zimzim brussala
> pappap pappap
>
> — *zimzim zimzim* by Petita Tatata, 2016[134]

Petita Tatata (2016) is an artbot that generates abstract poetry recited in a synthetic voice that is disseminated on the music sharing platform Soundcloud. The artbot is one of my own works exploring figuration. Additionally, it experiments with interfacing with Soundcloud as a site for artbots, which had not yet been explored by the artbot community.

The artbot's software architecture is comprised of a software application (coded in Python) for generating text-based poems. It interfaces Google Translate using a software component called *gTTS*. This component makes it possible to send text to the Google Translate service for processing through a requested language, and to get in return an audio recording of that text read out in a synthetic voice. The architecture also interfaces Soundcloud through its API, through which it programmatically posts these audio files onto the platform.

[134] *https://soundcloud.com/petita_tatata/zimzim-zimzim-2*

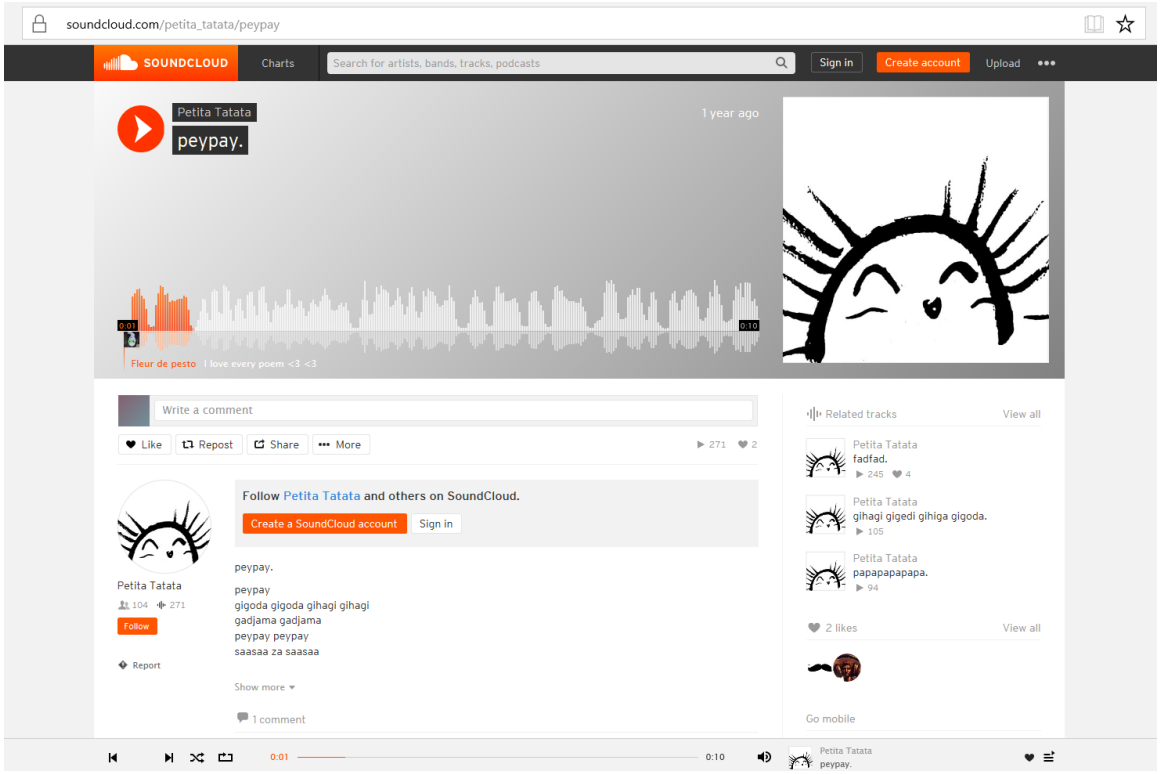


Figure 68. A screenshot of *soundcloud.com/petita_tatata*.

Petita Tatata is figured as a poet composing verse in its own unintelligible language, constructing gibberish words and phrases based on a set of rules for conjoining vowels and consonants (producing arrangements of phonemes such as ‘ta’, ‘po’, and ‘da’). Made-up words are then put into verse structures, and these are shuffled to create unusual rhythms and patterns. *Petita Tatata*’s character is part inspired by abstract Dada poetry, as well as Japanese *Kawaii* culture (Japanese for ‘cute’).

The Dada movement emerged in a club called Cabaret Voltaire in Zurich during the 1920s. Dada artists developed new approaches to painting, poetry, dance, graphics, sculpture, and performance art. They were equally interested in chance operations and the absurd, challenging the ‘serious’ intent of art. Hugo Ball, one of its leading proponents, developed a practice of reciting abstract poetry (also referred to as Ball’s *Sound Poems*). The poem *Gadji Beri Bimba* is a standout example. Following that, fellow Dadaist Tristan Tzara developed a method to construct poems in which words from newspapers would be cut up and dropped into serendipitous new arrangements.

The speech synthesis of the Google Translate service is available for a wide range of languages and dialects. The synthetic voices are either male or female depending on the language. After experimenting with several of these language and gender-specific

voices, the Japanese voice stood out as particularly interesting in its interpretation of gibberish. The sound it produced was rather ambiguous, sounding more like a cartoon rather than a human, and less gender-specific. This instantly provided a certain figuration of a cartoonish, gender-neutral character with Japanese tonality, which further inspired other aesthetic choices.

The profile picture is a hand-drawn image I made with a brush-pen, inspired by 'Kawaii' culture. *Kawaii* is a Japanese term to describe a fixation with 'cuteness'. Kawaii characters typically have large eyes and demonstrate babylike behaviour (see Figure 69).

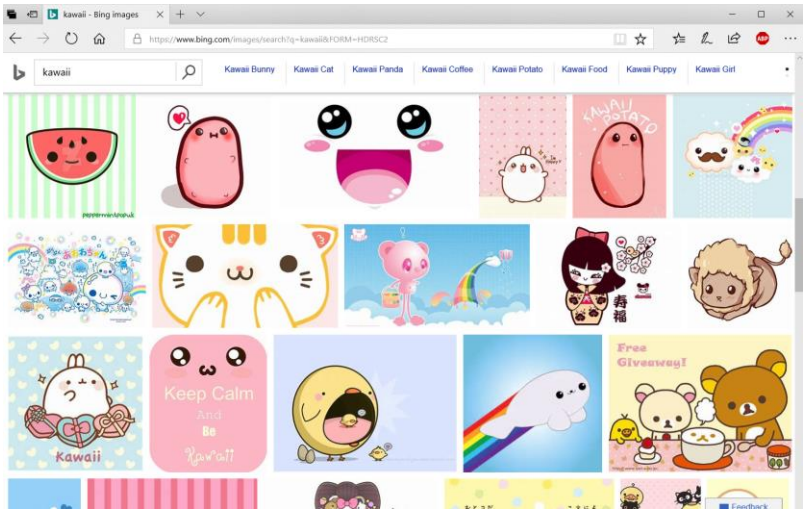


Figure 69. Bing image search for "Kawaii".

The Kawaii-inspired profile picture with the tone of the 'Japanese' synthetic voice and the generative abstract semi-structured gibberish all work together as interrelated semiotic components of the assemblage towards a distinct figuration. The name 'Petita Tatata' was chosen to be in keeping with the words that the software generates, as if the artbot had named itself.

Petita Tatata gained a small following who became regular listeners. One person commented, "I love every poem <3 <3". On the poem "gihagi gigedi gihiga gigoda", another person commented, "I must admit, this is kind of addicting."

Each post included the poem written out in the description. The sound recordings and poems are also listed with a Creative Commons license so that other people can freely remix or utilise the sounds for other purposes. The software written for the project was

also posted on GitHub and shared freely for others to use. This was to encourage others to expand on the project – potentially promoting further entanglement and multiplicity.

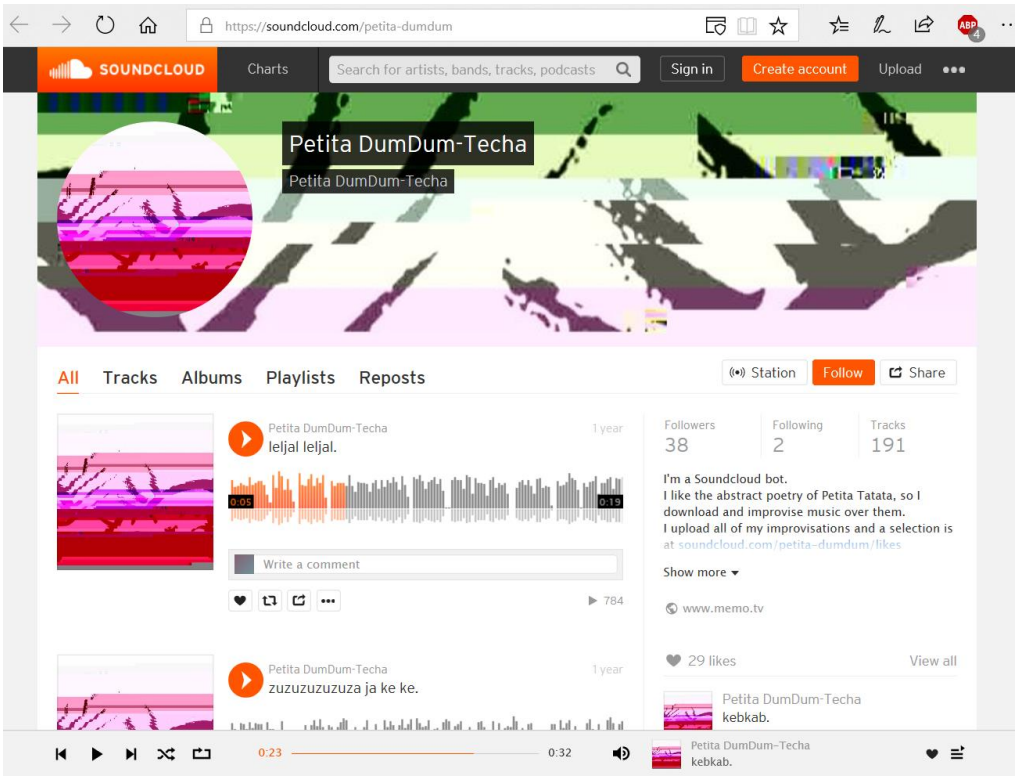


Figure 70. Screenshot of *http://soundcloud.com/petita-dumdum* (2016) by Memo Akten.

Without any further encouragement other than releasing *Petita Tatata* online under a Creative Commons license, the artist Memo Akten created *Petita DumDum-Techa* (2016). *Petita DumDum-Techa* downloads the recorded poems of *Petita Tatata* and processes them through audio software to automatically augment these with electronic drums and synthesizers. The poems are turned into abstract electronic music and posted to Soundcloud. Like the name, the profile image is also a derivative of *Petita Tatata*; Atkins has 'glitched' the original profile image by corrupting the JPEG file's data.

Petita Tatata, in this case, successfully evolved in its multiplicity towards further entanglements, productivity, and networked architecture. Its figuration helped in shaping the creation of a second character, towards a family. Petita Tatata also involved repurposing software tools normatively used for chatbots and other services, towards an ambiguous character that is neither fully machinic nor humanlike. Its components, semiotics configuration and generative compositions gently invoke a figuration for this sociocomputational assemblage that is emergent as well as designed from the outset.

Novice Art Blogger

> "There is a lot of elephants in the river but also there is a lot of elephants in the river. It is similar to a sandy area with an elephant made from sand."
>
> — *Novice Art Blogger*, 12th January, 2015,
> on *Sheep B* (1979) by Menashe Kadishman (see Figure 71).

Novice Art Blogger (2015) is an artbot that interprets abstract art and then blogs about it. It selects artworks from the online archive of the Tate Art Gallery [135] and processes these through image recognition software. The machine learning software component identifies elements within the image and generates descriptions of the images. These descriptions are then processed by my own software to rephrase these in the tone of an art critic. This text, alongside the image, are then posted to the microblogging platform Tumblr.

I swiftly created the artbot on 10 January 2015, and within a couple of weeks it had amassed over 10,000 followers on Tumblr. Its story didn't end there. It was asked to interpret the cover of a magazine; it was endorsed by Tumblr; it was invited to interpret the art of the Stedelijk Museum in Amsterdam, which was followed by an exhibition of the results. Finally, I was invited by the Tate to talk about the artwork internally as they sought inspiration from it, which was preceded by a private tour of Tate Britain, visiting some of the works that the bot had examined.

> "The prevailing figuration in Euro-American imaginaries is one of autonomous, rational agency, and projects of artificial intelligence reiterate that culturally specific imaginary. At stake, then, is the question of what other possible conceptions of humanness there might be." (Suchman 2007: 228)

Novice Art Blogger's re-figuration of *computer vision* technology into an amateur art critic eschews its more common associations and applications. Machine Learning strategies are often figured, along with its umbrella field of AI, as intelligent and rational, and as technically capable of making objective analyses of data. Computer

[135] Tate is an institution that focuses on British art and international modern and contemporary art.

vision is rarely figured as inaccurate, clumsy, and irrational, and yet when presented through *Novice Art Blogger*'s figuration, its technical capacities seems to align well with this narrative. *Novice Art Blogger* entertained with its humorous inaccuracy and wavering interpretations which seemed to parody the poetic flair of serious art critics. It also provided the artbot and its constituent technology with a clear role.

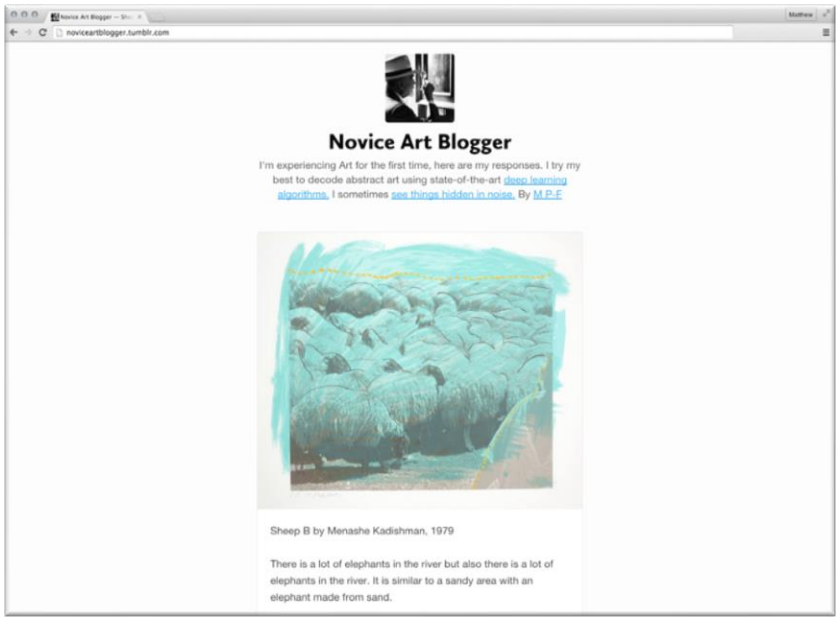


Figure 71. A screenshot of *http://noviceartblogger.tumblr.com*.

I was prompted to make this artbot after encountering new research into computer vision that uses a neural network to recognise the contents of photographs and create captions (Kiros, Salakhutdinov, and Zemel 2014). The researchers had also made an online demonstration software application that was driven by this technology. The application was designed to best work with photographs, as that is what the neural network has been trained on. However, the application would allow uploading any digital image.

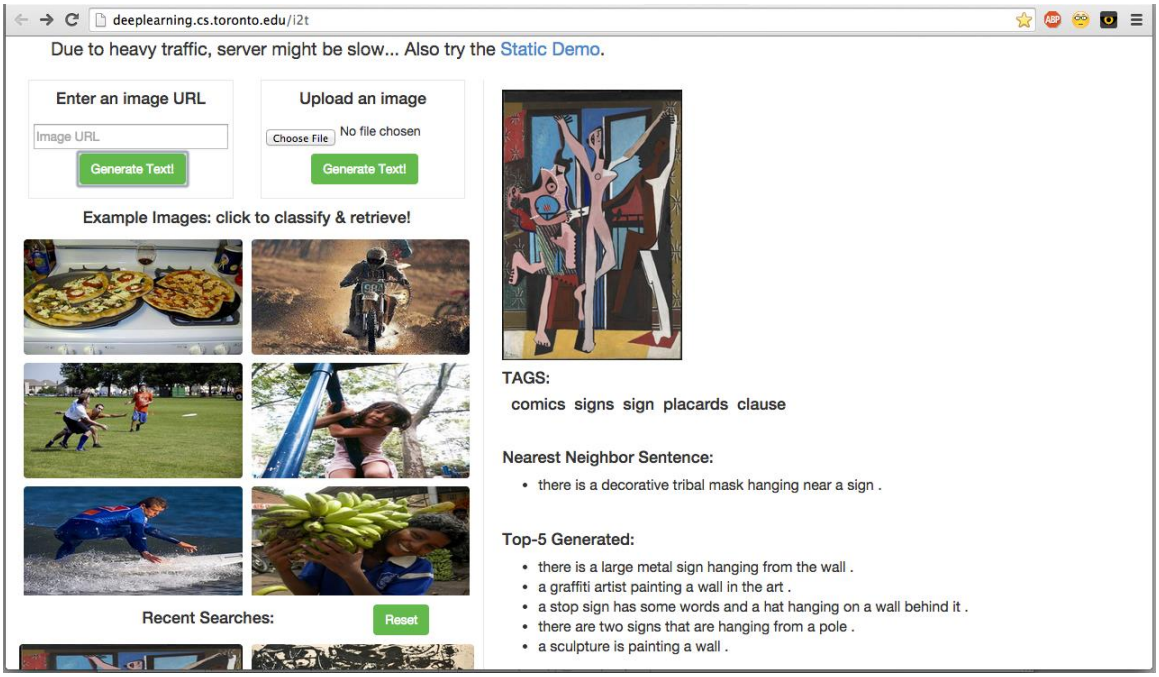


Figure 72. A screenshot of *http://deeplearning.cs.toronto.edu/i2t*. In this screenshot, I capture how I would manually use the Toronto research demo, and here, test its response to a painting by Picasso.

I was inspired by the artbots of Colin Rofls that processed images through the demonstration software application – *INTERESTING_jpg* (2013) would process images coming from the news source Reuters, and *NSFW_jpg* (2013) would process pornographic images.[136] *NSFW_jpg* generated the more interesting results, as the Toronto neural network had not been trained on pornography. These images required the software to make use of the language it had in its vocabulary, resulting in serendipitous metaphors. For example, a description for a pornographic image of fellatio was described as 'a lady is brushing her teeth with a pink toothbrush'.

This inspired me to research what other types of imagery could potentially force more metaphorical commentary on images. I tested the idea of processing images of abstract art, and the results were immediately very promising (see Figure 72). The online application would return five generated captions for an image and one "nearest neighbour" caption; that is, the caption describing the most similar image in its training set. These outputs of the demonstration software would all be utilised as inputs in my architecture.

[136] See *http://twitter.com/INTERESTING_jpg* and *http://twitter.com/JPEG_jpg*.

Novice Art Blogger's core generative software component would take these various image interpretations and concatenate these together with prepositions such as "or possibly", "or then again", and "or imagine". The 'similar image' result would be preceded by a sentence starter such as "I'm reminded of", "I once saw" or "Not dissimilar from" (see Figure 73). Altogether, the sentences formed a paragraph that evoked an art critic's musings over an image, commenting upon the artwork with various interpretations and ruminations.

A Raspberry Pi ran the software architecture and the software configuration implemented the library *Beautiful Soup* to make it possible to locate and obtain an image from the Tate's website. The architecture then sent a selected image to the online image captioning application, receiving the generated captions in return. The software then reworded these captions. Finally, the Tumblr API was used to post the image alongside its generated description online.

```
# add some variety to how the captions are concatenated
def concats():
    strlist = [" or possibly ",
    " or perhaps ",
    " or maybe ",
    " or then again ",
    " or "," or then ", " but also ", " or I suppose ",
    " or quite possibly ", " or rather ", " or I reckon ",
    " or I think ", " or imagine ", " or it is depicting ", " or envisage "]
    random_index = randrange(0,len(strlist))
    return strlist[random_index]

# add some variety to how your reminded of the nearest neighbour
def reminds():
    strlist = [" Reminds me of ",
    " It is similar to "," I'm reminded of ",
    " Not dissimilar from "," That reminds me of ",
    " It stirs up a memory of ", " It could be related to ",
    " Not unlike ",
    " I'm reminded of ", " I once saw ", " I was once shown ", " I once observed "]
    random_index = randrange(0,len(strlist))
    return strlist[random_index]

# make a two caption description of artwork, plus reminder.
def two_caption(raw_text):
    all_captions = captions(raw_text)
    if DEBUG:
        print(all_captions)

    ind = int(randrange(1,len(all_captions)-2))
    neighbour = reminds() + nearest_neighbour(raw_text).replace(" .",".").replace(" ,",",").replace("  "," ").replace("\n", "")
    artbanter = "" + all_captions[len(all_captions)-1].replace(".","") + concats() + all_captions[ind] + neighbour
    artbanter = artbanter.replace("  "," ").replace(" .",".").replace(" ,",",").replace("\n", "")
    artbanter = re.sub(' +',' ',artbanter)
    return artbanter, neighbour
```

Figure 73. Extract from NAB's source code showing the rewording of captions.

NAB produced blog posts for up to seven artworks a day. One notable example is its description of the 1959 painting *Death and the Conquistador* by Aubrey Williams (see Figure 74).

Figure 74. *Death and the Conquistador* by Aubrey Williams, 1959, side by side with a pizza decorated to look like an 'angry bird'.

> "A close up view of a pizza with one looking at it
> or it is depicting a pizza that is ready to bite of a large bowl.
> I'm reminded of a pizza, decorated to look like an angry bird."[137]
>
> — *Novice Art Blogger*, 12th January, 2015,
> on *Death and the Conquistador* (1959) by Aubrey Williams

Novice Art Blogger's interpretation for *Death and the Conquistador* contrasts the Tate's description for this piece as "a depiction of a field of bone-like shapes resembling human or animal forms" addressing "the colonisation of Latin America by the conquistadors".[138]

As the utterances of *Novice Art Blogger* are formatted to include a caption sourced directly from the training data (the 'nearest neighbour' entry), the artbot provides glimpses into the entangled humans that at one point would have been actively tagging and captioning images to create the training data. In the example of *Death and the Conquistador*, the artbot reproduced the nearest neighbour phrase of "I'm reminded of a pizza, decorated to look like an angry bird". This would have been a person's direct

[137] The *Novice Art Blogger* entry for *Death and the Conquistador* can be found at *http://noviceartblogger.tumblr.com/post/107795229139/death-and-the-conquistador-by-aubrey-williams*.

[138] See the gallery label, dated September 2016, at *https://www.tate.org.uk/art/artworks/williams-death-and-the-conquistador-t13341*.

observation of a pizza that looks like an *angry bird*, referring to the popular smartphone game *Angry Birds*. *Novice Art Blogger* provided many intriguing insights into the hidden practices of image captioning. For *Self-Portrait as a Drowning Man* by Dieter Roth, the bot produced, "I once observed two birds haing [sic] sex on top of a roof covered in tile".[139] The misspelling of 'having' is carried over from the source training data produced by the entangled human actors employed to caption the training images. Novice Art Blogger exemplifies the notion of a sociocomputational assemblage, contingent upon human and computational intra-activity. The properties and capacities of Novice Art Blogger emerge from a configuration of human and computational elements, including my own involvement in the refiguring of the software as an art critic. This re-configuration made me question my own ethical role in the ontological distinctions made through this figuration. What seemed like a image-captioning software component, readily available to plumb into a pipeline, revealed itself to be a sociocomputational chimera of undisclosed human performance, rendered once-removed through neural-network architecture.

Novice Art Blogger was presented at the Stedelijk Museum of Contemporary Art in Amsterdam to respond to the museum's art collection. The exhibition was curated by Annet Dekker and was titled *Algorithmic Rubbish: Daring to Defy Misfortune* (2015).[140] For this presentation, a reconfiguration of *Novice Art Blogger* was produced that would computationally inspect the Stedelijk's website instead of the Tate's site. Further to this, the Toronto captioning demo had stopped working, and only their image tagging function was working. To get around this, *Novice Art Blogger*'s software had to be modified, which gave the Stedelijk's version a slightly different tone of voice. Hundreds of items from the Stedelijk collection were processed by the new software configuration and the results were published on *noviceartblogger-stedelijk.tumblr.com*. The curator then made a selection for the exhibition (depicted in Figure 75).

[139] This entry can be found at *http://noviceartblogger.tumblr.com/post/117910659344/self-portrait-as-a-drowning-man-by-dieter-roth*.

[140] More on *Algorithmic Rubbish* can be found at *http://www.smba.nl/en/exhibitions/algorithmic-rubbish-daring-to-d/*.

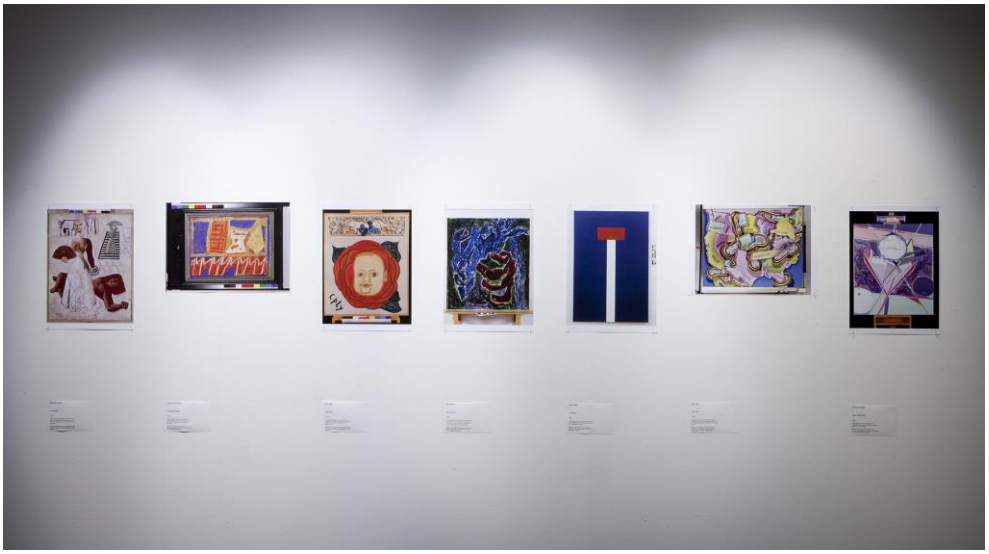


Figure 75. *Novice Art Blogger* presented *Algorithmic Rubbish: Daring to Defy Misfortune* (2015) at the Stedelijk Museum of Contemporary Art in Amsterdam.

Copley, Bill - Baby Rose, 1952

I would describe this as a modest head cabbage or possibly a backpack on top of a mask. I'm reminded of a great, silver sleeping bag.

Figure 76. An entry from *http://noviceartblogger-stedelijk.tumblr.com*, the Stedelijk version of *Novice Art Blogger*.

There were some endorsements from others, such as a mention on Twitter from science fiction writer William Gibson. Many of its spectators responded favourably to the artbot,

and described it as “hypnotizing in its honesty”, “charmingly honest”, and “the most adorable art critic.” The UK based *Dazed and Confused* magazine ran a story about *Novice Art Blogger* under the attention-grabbing headline, “This robot reviews art better than most critics” (Gorton 2015a). The magazine requested for the bot to interpret their front cover (Figure 77). The German art critique magazine *Monopol* ran the headline “Dem Bot ist egal, ob es ein Warhol ist,” which translates to, “This bot does not care if it is a Warhol” (Kohlick 2015a).

Kendall Jenner on the cover of Dazed Winter 2014

There is a young woman dressed in white and red with a large pink heart written on her face, or maybe a woman with red feathers holding a large amount of hair with a teddy bear. I'm reminded of a woman adorned with banana on her body and head.

Figure 77. *Novice Art Blogger* was invited to interpret the front cover of *Dazed and Confused*, Winter 2014.

Novice Art Blogger is not commenting on the original artworks, but on digital-photographic reproductions. The interpretation of reproduced artwork is further removed from its context – the software has no understanding of the author, period, style, history, and other contextualising elements. The artbot does not see the original materiality of the artwork – neither paint nor brushstrokes. Instead, it has a grid of pixels to process, a substrate for visual art that would be alien to some of the original authors of these artworks.

As argued by Berger in *Ways of Seeing* (2008), there are implications to reproducing artworks onto new sites of dissemination, causing shifts in meaning and interpretation. Berger discusses the authority that captions and texts accompanying artworks can have over their received meaning (p. 28). Berger illustrates this by presenting an image of an artwork as *is*, and on the following page he exhibits the same image accompanied by a caption. Berger argues that "the image now illustrates the sentence". The influence of the caption thus cannot be disentangled from the meaning-making of the spectator. In the case of *Novice Art Blogger*, the meaning of the image is jolted by the caption created by the artbot, but by no means does the artbot have an authoritative voice on the matters of art. On the contrary, *Novice Art Blogger*'s commentary, humorous and inaccurate as it is, relaxes the reading of the artwork.

In one example, an untitled image by Brice Marden (1971), which depicts a faint grid on a white background, *Novice Art Blogger* produced the description, "Two urinals are in the corner of a building".[141] The software produces these odd remarks by detecting weak signals within the minimalist artwork, linking it to the banal items it has been trained to detect. In a computer vision test, this result may be dismissed as an incorrect interpretation of the image, but given the figuration of a novice art critic, the result is rendered as plausible (to be very generous on this occasion), or at the very least, charming in its naiveté.

Much of the artbot's figuration is driven simply by its name. A well-chosen name that is descriptive, alluding to its reason for being, presents others with an elevator pitch of what the configuration *is*, what it *does*, and its *why*. I was inspired by Kazemi's *Random Shopper*, a name that describes in a nutshell what it does. I found this naming convention to be doing most of the work in figuring the bot. Moreover, it eschewed the convention of explicitly calling it a 'bot' in its title.

As a composition of software components that entangle human activity in their realisation and data resources, as well as its reconfigurability towards different manifestations, *Novice Art Blogger*'s figuration as an amateur art critic illustrates how the proposed framework for understanding artbot practices can enable discussion of various important aspects. In turn, these identified aspects of the artform have been explored through my own practical experiments, and *Novice Art Blogger* tested the

[141] Available at *http://noviceartblogger.tumblr.com/post/107788187584/no-title-by-brice-marden-1971-two-urinals-are*.

importance of figuration. A simple descriptive name and a configuration of components to materialise that figuration created a credible, yet naively charming, software-based art critic. Its figuration, however, also demonstrated the potential for concealing human entanglement embedded within a sociocomputational assemblage. The punchy and attention-grabbing headline in *Dazed and Confused* quite simply sets up the dichotomy between human and their machinic competitors that I attempted to move away from. As such, this case study also reveals that figuration is formed not only from the outset by the designer but also by the collective cultural imaginaries and preconceptions promoted by other actors, including the press.

Conclusion

This chapter sought to analyse a number of artbots with a focus on figuration. Having understood artbots as figurations of internet-based software architectures, I examined the many ways in which a software configuration can be characterised. Figuration seems key to artbots, and their variance sets them apart from their chatbot predecessors. One of the core strategies of artbot practice is to invent refreshingly new characters that test the limits of what our collective imaginaries understand a bot to be. Bots have traditionally inherited the robot's role as humanlike servant, ready to serve and respond to user requests. Artbots often eschew the humanlike and robotlike figurations that have become normalised, and expand into alternative figurations that include animals, natural phenomena, and abstract storytelling entities. If it matters what stories make worlds, artbots are a valuable contribution to storytelling and the realisation of those stories through diegetic yet working prototypes, deployed in the public sphere for others to encounter.

Artbots take their spectators on a journey within a story-world or through their persona, but they're also transparent enough to make clear their own mechanical nature. Many artbots are explained through their publicised code, or simply through an association to their artist creators. Spectators are free to swivel between the story and the construction of the story, dipping in and out of suspended disbelief, balancing enjoyment with intelligibility. Within artbots, there is a desire to build stories but not so far as to trick audiences and obfuscate configurations, unlike the algorithmic imaginaries that often figure commercial software architectures.

Bots are closer to animatronics than robots: repetitive and somewhat predictable, working within the limits of their designated parameters. As illustrated by *Magic*

Realism Bot, often the illusion of intelligence is created simply by expanding that parameter space, reducing the repetitiveness and increasing variance. *Magic Realism Bot* also illustrates that much of a bot's figuration and appeal is built into it through the entangled authorship and personae of its creators. *Magic Realism Bot*'s stories are the imaginative deconstructed stories of the Rodleys, processed through a randomised reconstruction device.

As artbots increasingly utilise machine learning components, such as *Novice Art Blogger*'s core computer vision element, the implications of figuration become more pronounced. Figuration has the capacity to reinscribe tropes about artificial intelligence, and with it, perpetuate the dichotomy between humans and machines. This potentially obfuscates how, in actuality, machine learning systems often require entangled human intra-activity for training purposes. The 'AI' figuration displaces and disregards these entanglements to fit the intended narrative. *Novice Art Blogger* was an interesting experiment in which common tropes associated with machinic vision were replaced with the figure of a wavering, naïve, and comical image interpreter. Although this figuration has successfully made AI more approachable, it still troublingly encapsulates human activity. Human activity would surface through its sociocomputational utterances, which over time alerted me to the potential issue of figured-out entanglement. This was further complicated by the involvement of the press, who had their own authoritative voice in figuring the artbot, siding with the common trope of humans facing the rise of robot competitors.

The figuration of bots is potentially one of various types of figuration applicable to sociocomputational architectures. In artbot practices, it is increasingly common for practitioners to experiment with other figurations, labelling and figuring their software applications as, for example, 'generators', 'apps', and 'simulations'. I speculate that as this artform matures, the focus on bots will dissolve, leading to a wider range of figural experiments. This speculative new development of artbots is what I call *configurative art*. Configurative art would embrace figuration as a core strategy towards the creation of increasingly diverse edifices that work with a further range of configurable components, computational and noncomputational, human and nonhuman. Configurative artists would conceivably be the authors of diegetic-nondiegetic worlds that bridge the figural with the concrete. In the following conclusive chapter to this thesis, I make further recommendations for the future of practice and research and discuss other implications and key findings of the study.

Chapter 7: Conclusion

Having analysed a range of artbot case studies through aspects of multiplicity, entanglement, and figuration, this study develops an understanding of artbots and maintains their importance as an original and new genre of internet-based software art pertaining to a community of practice that has emergently advanced its own particular style and approach.

This conclusion briefly reiterates what has been covered in this thesis, assesses the value and relevance of key findings in relation to existing literature, assesses the methodology in light of the research conducted, and discusses potential limitations with the approach taken. It also outlines the implications of this research for further developments of artistic practice, drawing on both the study's own contribution to knowledge of artbots and recent developments in the field. In addition, I discuss the implications for different potential audiences, as well as recommendations for further research.

Thesis summary

The principal goal of this doctoral research has been to study a new form of artistic software practice known as *artbots*, and identify and examine the key aspects of this sociocomputational practice. These new understandings were needed in response to the lack of acknowledgement and critical discussion around this movement. This artform is burgeoning, original, and producing work that relates to important wider implications regarding social media and software-mediated sociality. In developing an understanding of this practice, this thesis introduces a theoretical framework which sympathetically aligns with the properties of artbots, whilst not imposing restrictive conditions to be met. Instead, this framework works as a set of analytical tools to examine the *multiplicity*, *entanglement*, and *figuration* of artbots. This framework is one of the main outcomes of this research, and makes a significant contribution to the understanding and reading of this artform.

Prior to this framework, an understanding of artbots would have hinged upon unclear notions of 'algorithms', robots, and artificial intelligence. Unpacking these notions provides greater clarity in understanding the wider context of internet-based software applications. The notion of a *sociocomputational assemblage* was developed to describe internet-based software applications as complex and dynamic composites of intra-related *computational architectures* and *social entanglements*. Having arrived at this

understanding, artbots can be described as an artform that creatively *figures* and *configures* sociocomputational assemblages.

An opportunistic and practice-based approach to participant observation was taken in order to study the community of artbot makers and their works. I engaged the community through my own artbot practice, whilst experimenting with how artbots could be implemented on other platforms. Several works were created as part of this study, and two stand-out projects – *Novice Art Blogger* and *Shiv Integer* – made notable contributions. The latter was co-authored with the artist Julien Deswaef, a collaborative process that rendered possible a period of participant observation. This, in tandem with other methods, contributed to a varied methodology that informed this study. The methodology addressed the research objective of studying artbots, and enabled the practice to evolve. A showcase of works titled *Art of Bots* encouraged the reconfiguration of existing works for public display. The showcase not only provided a number of case studies analysed in this thesis, but served the important role of bringing together proponents of this movement as well.

The case studies of artbots, the exhibited works at Art of Bots, and my own works created during this study were analysed through notions of multiplicity, entanglement, and figuration. The chapter on multiplicity recognises the practice of mixing and re-configuring software components using APIs and other modular resources. These configurations become productive of generative activity. Depending on the intentions of the artist, they can be configured to output a variety of media, utterances, poetry, nonsense, or other productions. The notion of multiplicity helps understand these configurations as open to further re-arrangement and scaling up. Multiplicity implies that artbots can continuously be developed or transformed for new contexts and changes within the composition of the artwork.

The chapter on entanglement identifies multiple forms of human and nonhuman intra-activity within software architecture. An understanding of artbots that accounts for entanglement moves beyond limiting assumptions that bots are purely nonhuman interfaces and services for human users. Instead, bots are configured contingent upon human activity and data resources, which are entangled within a heterogenous assemblage.

The following chapter exploring figuration pays attention to how configurations are materialisations of character, or imbued with character. Figuration influences the design

and impetus for software architecture, and in turn influences how those constructs are perceived by others. Artbots demonstratively explore a wider range of possible figurations, and the 'bot' form is itself a figuration of software. This implies that artbot artists tacitly understand how narrative is an essential factor in their practice, and tend to reject the common tropes of the assistive and subjugated robot in favour of defining other roles.

The key argument of this study is that artbots can be considered an original and categorically new genre of internet-based software art, with its own set of methods, approaches, aesthetics, values, proponents, contributions, and implications for artistic practice. Furthermore, this nascent artform is still evolving, and I propose a new *configurative art* practice that expands artbots towards a wider exploration of figurations and forms that sociocomputational configurations can take.

Understandings of Web-based software

During this research study, I surveyed different bodies of literature concerning internet-based software applications and found a need to confront the prevalent focus on 'algorithms'. A growing body of literature described as *critical algorithm studies* is increasingly dictating discourse on issues and implications of online software, but in its current form is problematically disconnected with the practices they study (and how those practices do not refer to algorithms, nor necessarily contribute to the design of algorithms). One of the major breakthroughs of this study was identifying that these issues can be understood more clearly through the overlooked theoretical writing on *software architecture*. This theory serves as a rich foundation for understanding how algorithms are present within architectures, encapsulated within software modules, and often interfaced through APIs. This also matched my own empirical experience of creating bots.

The design of online software fundamentally changed after the Web 2.0 movement, spearheaded by theorist-developers such as Roy Fielding. This understanding paints a more revealing picture of how search engines or social media platforms, for example, come into effect through the configuration of arrangements of software *components*, *connectors*, *data*, and human *entanglements*. Critical algorithm studies often overlook software architecture as a model and practice, and instead use theoretical models that suggest a singular "mega-algorithm" as the core component, supported by subsidiary processes. It became evident that much of this view arose through the figuration of

software architectures as a singular, proprietary algorithm – a narrative that is itself rooted in the marketing and communication of complex sociotechnical assemblages.

The idea that an 'algorithm' can be a type of figuration – as well as 'bot', 'platform', 'app', and so on – avoids flawed ontological distinctions in favour of a view in which these terms refer to different materialisations of concepts for software architecture. As such, my analysis avoids this misguided search for the 'algorithm', and instead examines (and even identifies in source code) particular arrangements of software components, connectors, and data resources. Reading online software through a lens of figuration reveals that the contentious implications often attributed to 'algorithms' are contingent upon the agency rendered capable in bringing disparate elements (and their underlying algorithms) together. The logic and structures that dictate these configurations were the closest aspect I could find to the misguided notion of a singular 'algorithm'.

Although software architecture itself has limitations in adequately framing sociotechnical entanglements, I believe its theory and practice are of value for other scholars interested in bots, platforms, and other software-run internet services. Software architecture helps reconcile the discrepancies between the computer science and sociological/humanities understanding of algorithms, and allows a focus on the effects and processes that come together only during runtime of architectural configurations. The terminology of software architecture (e.g. *components, configurations*, *scalability*) is especially beneficial for developing uses of language that enable this cross-disciplinary discourse. My own term *edifice* expands on software architecture by interweaving material realisations with the thinking and figurations that influence them. 'Edifice' evades the ontological trappings of bots, platforms, apps, generators, and websites; it suggests an umbrella term that covers various types of figured architectures. Through this lens, one can imagine other edifices for future practice that escape the conceptual framing of these established typologies. For artists especially, moving beyond the conceptual borders of a 'bot' would perhaps be liberating. Rather than defining their works as 'artbots', 'generators', 'websites', 'apps', and indescribable others, 'edifice' could serve as a meta-category that one could refer to without worrying about sub-categories.

Software architecture provides a theoretical reference to bridge the theory and practice of internet-based software applications, but it is less suitable as a conceptual tool for the sociomaterial analysis of such applications. This is because software architecture is developed from the viewpoint of the configuration's architect, and thus applies

preconceptions about human relations that are limited to 'users' of the application. Moreover, scholars such as Barad and Orlikowski make compelling arguments for sociomaterial research that take a posthuman approach to studying sociality and technology. This is especially relevant to artbot practices that challenge traditional dichotomies of separating bots from humans in modes of interaction. *Sociocomputational assemblages* provide a conceptual abstraction to discuss arrangements of elements of computer architecture *and* entangled humans, and warrant intra-related processes and dynamics as crucial to these compositions.

The notion of sociocomputational assemblages can inform studies that examine and analyse other contexts in which complex software architectures are being built with human components in their composition. The growing field of artificial intelligence (AI) stands out as a notable area in urgent need for critical examination. Projects of AI increasingly employ *machine learning* techniques, and these can contentiously obfuscate the presence and intra-actions of entangled humans that perform tasks such as image tagging and data cleaning. Moreover, narratives of AI (and its current media hype) often play along to a dichotomy of humans and nonhuman AI agents. Framing AI projects as sociocomputational assemblages challenges this dichotomy, and prompts examination and analysis into their underlying mixed composition. In addition, the three-part framework of multiplicity, entanglement, and figuration that I have developed for studying artbots is apt for examining similar matters in AI. 'Artificial intelligence' in itself can be considered a figuration of software architecture – complex edifices characterised as 'intelligent' and wholly 'artificial', even though many are, in fact, partly powered by human resources.

Practice through research

This doctoral research study resulted in a multitude of insights about the practice of artbots, and led as well to various original works contributing to this artform (and understandings of it). Even though this study was devised primarily to conduct research through practice, it is worth highlighting the achievements of practice conducted through research. A vast body of personal work was produced: *All Eyes*, *Every User*, *Shiv Integer*, *The Revenant*, *Petita Tatata*, and *Novice Art Blogger*. A couple of these turned out to be rather successful and received an unexpected amount of attention. *Novice Art Blogger*, although made in a day, gained over 10,000 followers in just two weeks. Moreover, this project managed to capture the attention of art critics by parodying their own practice. These works received coverage in journals such as *Art Monthly* (along

with *Algopop* on the front cover), *Monopol* (Kohlick 2015b), and *Dazed and Confused* (for which *Novice Art Blogger* interpreted the front cover) (Gorton 2015b). Conversely, these responses from the press helped developed an interest in the *figuration* of artbots, as article titles such as "This robot reviews art better than most critics" steered and influenced the narrative that described *Novice Art Blogger*. Obviously, the bot does not review art better than most critics, but provocation outweighed accuracy in this reporting. This provided a glimpse into how the press becomes implicated in the figuration of technologies and is prone to positioning case studies within the consensual cultural imaginaries that inform our understandings of technology, such as the story of robots outperforming their human competitors.

Similarly, *Shiv Integer* received considerable attention from the press. Ars Technica described the project as "that time a bot invaded Thingiverse" (Newitz 2016), with similar implications as to how the artbot was figured and understood by audiences. This article came to the attention of Cory Doctorow, who then understood the project through the lens of Newitz, and declared that "artists trolled Thingiverse with 3D mashup bot" (Doctorow 2016). Of course, these attention-grabbing headlines must be taken with a pinch of salt. *Shiv Integer*'s press attention prompted more reactions and debate, especially within the comments section of the Ars Technica article. These engagements further contributed to the examination of the work, allowing me to view perspectives on the artbot that were outside of my own entanglement with it.

Shiv Integer was exhibited in a number of places in various formats. The main show, as discussed in the chapter on entanglement, was at iMal in Brussels. Shortly after, three pieces were exhibited at the Liste Art Fair, with the support of the Museum of Media Art in Basel. The other pieces concurrently went to Humber Street Gallery in Hull, by invitation and support of the Crafts Council. Following this, the works for Basel went to MU Gallery in Amsterdam. For exhibition requests where was difficult to exhibit sculptures, a video work of rendered artefacts was exhibited. In other, more unusual exhibits, curators and artists would 3D print *Shiv Integer* works themselves.

Although I am content with all these engagements and opportunities to exhibit *Shiv Integer*, I do hope that in the future, a more ambitious exhibition of *Shiv Integer* can be realised. Such an exhibition would find a way to celebrate the entirety of the project – all 666 3D files, either printed or partially exhibited using holographic devices or augmented reality. The *Shiv Integer* exhibitions inspired in me the idea that artwork can manifest in different arrangements, scales, and versions at every appearance. This fully

celebrates the concept of multiplicity beyond the software edifice into the multiple viewpoints of its offline documentation. This concept for exhibiting is something I plan to explore further with new artworks.

During the four-year duration of this study, I created other works through my practice, some of which, in much earlier drafts, were mentioned in this thesis. Conducting a wide range of practice helped define boundaries around the types of work covered by this thesis. Also, these other works connected me to different communities of practice and curation, which helped identify differences in approach.

To conclude with a final thought on conducting practice: situating practice as practice-based research is complicated, with yet more room to explore. In some practice-based PhDs in design, this may take the form of conducting social research through examining participant engagements with designed artefacts and scenarios. My practice functioned as a way to access a community of practice in study as a form of participant research, and to further understandings of that type of practice through direct engagement with it. This approach is useful for the wider field of design, in which many communities of design practice have tacit knowledge in need of study and documentation. For example, my father was part of a generation of graphic designers (who would call themselves *commercial artists*) based in East London that utilised pre-computing equipment, such as UV light beds for curing inks and silkscreen beds. Many other little-known design, production, and manufacturing techniques exist all over the world, and practice-based researchers are arguably the best positioned to study them.

Methodological contributions

This study makes valuable contributions to practice-based research methodology by way of developing *research through collaboration* as a form of participant observation. Similar in scope to *research through design* (Zimmerman, Forlizzi, and Evenson 2007), this method involves teamwork in the development of a prototype and its deployment. However, in my version, the collaboration with a practitioner from outside of the research team produces shared insights from an external expert in the field of study. This method was a natural fit with my area of research, as practitioners within artbots (and software development more broadly) are open to sharing, extending software, using collaborative tools such as Git, and working on projects within dynamic and informal collaborative networks. This approach helped both myself and my collaborator to develop a new artbot work, but more importantly, it generated many engagements and

interactions with my collaborator that I may not have had through traditional social research methods. The period of engagement was open-ended and centred around practice.

I believe research through collaboration could be beneficial for practice-based researchers investigating other forms of art and design that would otherwise awkwardly straddle a gap between separated worlds of 'practitioner' (when conducting their own practice) and 'researcher' (when approaching others). Through collaboration, researcher-practitioners can conduct a study of others through doing and doing-with. Furthermore, mutual explorations, understandings, analysis, and developments of practice may unfold. I can imagine that in future research studies centred specifically around the practice of an individual, this research method could be instrumental in opening up more nuance and engagement through practice. It is hoped that this study will help researcher-practitioners identify that they can opportunely use their practice to offer collaborative working relations, or become inside learners as apprentices, to learn more about an individual practitioner or creative community. Also, as collaborations are already a familiar aspect of creative practice, tacit understandings of different approaches could be reframed and repurposed for the further development of research through collaboration. For example, audio-visual collaborations allow for visual practitioners and musicians to work together, whilst coming from different backgrounds and offering disparate skillsets and output. It is foreseeable that learnings from this model could inform future research.

Another methodological experiment utilised during this research was the prior art showcase (*Art of Bots*). This method similarly created more points of engagement with other practitioners and enacted a more collaborative and insider perspective. The showcase happened during the research period rather than towards the end, meaning that the exhibition did not take the form of a retrospective and conclusive account of practice, but instead enabled further experimentation. At the time of producing the showcase, I was keen on experimenting with new ideas of presenting artbots, and was conscious that this required taking an approach in which artworks did not have to be displayed as faithful reproductions of the online form, but could be adapted and evolved. The artists were themselves keen on reconfiguring their works for the showcase, and this helped shape an understanding of artbots as reconfigurable art.

Through this approach, research and artworks become co-constitutively 'in progress' and entangled in their development. Both practice and research remain open-ended during the event, and its concretisation takes place afterwards through analysis and

documentation. This method could be more rigorously considered for future use, making this format more clearly defined to both artist-participants and audiences. The implication of this, however, is that researchers would be taking on an ever-more performative role in re-shaping the research material. However, with the informed involvement of participants, the material and research would evolve in a consensual direction.

The methodology taken in this study enables a very broad set of practices, platforms, and approaches to become the research material. However, this comprehensive perspective faces a limitation in the amount of attention that can devoted to examining valuable details. An approach that is able to scrutinise each configuration and identify precisely how the interconnection of elements became generative of new phenomena would too be worthwhile. A methodology involving a *software studies* approach (Fuller, Malina, and Cubitt 2008) that more rigorously examines software components and software applications would address these concerns fittingly. Having now conducted a much-needed broad approach to artbot studies, follow-up studies could potentially focus on detailed analyses of particular artworks and their constituent components, capturing details through a more focused study.

From artbots to configurative art

The main goal of this research study was to shine a light on artbots and illustrate the breadth and richness of the work being conducted by its proponents. In doing so, I developed a framework to examine and analyse artbots, and through it, a more in-depth understanding of artbots that pays attention to their multiplicity, figurations, and entanglements has been achieved.

Artbot practices are in flux and in progress, and as such, this thesis had to respond to the dynamic nature of the research material. The implication of studying this moving target is that its findings could be in danger of being out of date by the time of its publishing. At the time of writing, artbot practices have already evolved further, due to the changing landscape of Twitter and the emergence of new social media platforms and technologies such as Mastodon. Twitter has become widely criticised to hosting nefarious bots, and in response, the company has had to proactively shut down bot activity. Meanwhile, Mastodon, an open-source social media architecture that can be replicated and instantiated on any server, has become increasingly popular. A couple of Mastodon instances have been created in relation to artbots. One instance, *botsin.space*, was

implemented by Colin Mitchell specifically for artbots, and currently hosts over 2,000 of them.[142] To keep an eye on this development, I maintain an account on *botsin.space* and have continually noticed a proliferation of new bots. Another instance that is pertinent to artbots is *friend.camp*, initiated by Darius Kazemi. Kazemi runs this instance for friends, many of whom make artbots. Conceivably, developments of the artform may unfold through conversations happening on this channel.

These artist-led instances of Mastodon demonstrate an important shift towards creative explorations of social media architecture. Rather than being reliant on a bigger company-run social media platform to create small architectural interventions and annexes for their work, artbot artists are now building their own social media applications. Such edifices are open to experimentation as to how these are figured, composed, and entangled with others, and still possibly open to artbots. This broadening of architectural and figurational interest in artbot practices is one of various signs that suggest that this artform's evolution remains in rapid flux. The main proponents of artbot practices seem to be pushing outwards from the artbot form to discover other forms of edifices and figurations, as well as branching out to work with other reconfigurable components. The artist Everest Pipkin, for example, has recently created *Ellinger, TX* (2018), a software-run simulation of a semi-fictional town in Texas that can be viewed in real-time on the live-streaming platform *Twitch*.

Speculating about where this practice is heading and stating my own recommendations, I suggest that the artform is maturing into what I call *configurative art*. Configurative art explores a broader range of possibilities for configurations and figurations of components – software, hardware, human, and other-than-human. With increased awareness of how narrative and figuring plays a role in the conceptualisation and framing of software architecture, configurative artists will proceed to materialise diegetic figures, worlds, and edifices beyond the habitual. 'Bots' will be seen as one of many forms; however, its formative and focused exploration will inevitably influence how other forms of edifices can radically vary. Perhaps artbots will be seen as an important first act, setting into motion the network of artists, preliminary experimentation, and collaboration required to evolve into a second stage.

I personally want to be engaged within the online edifices of configurative artists rather than commercially run user environments. Now that I spend time within and as part of

[142] See *https://botsin.space/about/more*.

friend.camp and *botsin.space*, in the company of both humans and artbots, returning to platforms such as Twitter feels dull and uninspiring. Moreover, as the chapter on entanglement demonstrates, these relations to social media applications are limited in their thinking of how humans can be co-constitutively entangled within sociocomputational assemblages. I want to find myself consenting to becoming a micro-component in a mass-cyborg configuration that collectively renders the impossible possible. Why use me for unsuspectingly looking at adverts in-between my friend's utterances when I could be software-managed into cooperatively tackling climate change calculations or piloting a fleet of swarming ocean-cleaners? Bring on the imaginative configurations that bring these sociocomputational entanglements into operation.

Further recommendations

The interdisciplinary approach taken in this research means it is likely that this thesis holds appeal for audiences from different backgrounds. Artists and designers looking to broaden their practice or take up this new artform may find the documentation and discussion of case studies as a useful reference. Moreover, the framework of analysis can be adopted as a prompt for generating artbot ideas. It is hoped that current proponents of artbots will also find this thesis helpful, and that my speculations about where this artform is heading will help stir debate and possible adoption of these ideas. Software developers may find the framework of multiplicity, entanglement, and figuration useful as a set of questions of self-reflection when conducting their practice (by asking, for example, what sort of figural predispositions are present, and how might they be re-figured).

Sociologists may find this thesis useful as an experiment in intersecting STS thinking with practice-based social research. The methodology may help other practice-based researchers consider how their practice may opportunely open the door to related practices and communities. Practice-based research that falls between disciplines, especially at the intersection of design, computing, art, and social research, is still finding its way. The methodology chapter was particularly written with this in mind, with the hope that it will provide evidence that practice-based researchers are well-positioned for conducting important social research through new methods involving their practice.

This leads me to my final recommendation for further research: increased studies of bots. During the duration of this research study, the issue of nefarious and contentious

bots on social media became an increasingly pressing concern. The proliferation of bots can sometimes be traced to factories of human actors entangled in their creation and ongoing operation. The boundaries between software and human operators are increasingly blurred. Urgent studies are required to trace these, unpack their activity and composition, and identify the rationale behind these bots – are they present to incite others and polarise debate? Are they proactively inflating the popularity of certain political figures on social media? Are they configured to generate disinformation? Are they monetising or gaming flawed advertisement systems? Are they harvesting user data? Perhaps some of the key findings and understandings developed around artbots in this thesis can help inform studies into other social media bot interventions. It was frustrating to not be able to tackle these growing concerns and implications more directly within this study, but the shifting landscape of bot activity, along with its unprecedented scale, made inclusion of these matters unfeasible. It is hoped that my various observations and analysis will contribute to future research in this field.

References

Anderson, Christopher W. 2013. “Towards a Sociology of Computational and Algorithmic Journalism.” *New Media & Society* 15 (7): 1005–1021.

Applebaum, Jacob, and Christian Stöcker. n.d. “Shopping for Spy Gear: Catalog Advertises NSA Toolbox, Der Spiegel.” SPIEGEL ONLINE. Accessed May 17, 2017. http://www.spiegel.de/international/world/catalog-reveals-nsa-has-back-doors-for-numerous-devices-a-940994.html.

Armitage, Tom. 2007. “Twitter: A Messaging Bus for Telescopes (and Almost Anything Else You’d like to Overhear).” Infovore. May 22, 2007. https://infovore.org/archives/2007/05/22/twitter-a-messaging-bus-for-telescopes-and-almost-anything-else-youd-like-to-overhear/.

———. 2010. “Open Data for the Arts – Human Scale Data and Synecdoche – Blog – BERG.” October 1, 2010. http://berglondon.com/blog/2010/10/01/open-data-for-the-arts-human-scale-data-and-synecdoche/.

Arnold, Derek. 2015. “Remote Viewing – Derek Arnold – Medium.” March 2, 2015. https://medium.com/@derekarnold/remote-viewing-5cb161cdef4a.

Asimov, Isaac. 1950. “Runaround. I, Robot.” *New York: Bantam Dell*.

Barad, Karen. 2003. “Posthumanist Performativity: Toward an Understanding of How Matter Comes to Matter.” *Signs: Journal of Women in Culture and Society* 28 (3): 801–831.

———. 2007. *Meeting the Universe Halfway: Quantum Physics and the Entanglement of Matter and Meaning*. duke university Press.

Becker, Howard S. 1963. “Outsiders: Studies in the Sociology of Deviance.”

Becker, Howard S., and Blanche Geer. 1957. “Participant Observation and Interviewing: A Comparison.” *Human Organization* 16 (3): 28–32.

Benjamin, Walter. 2008. *The Work of Art in the Age of Mechanical Reproduction*. Penguin UK. https://books.google.co.uk/books?hl=en&lr=&id=ZucuzeiI-24C&oi=fnd&pg=PT2&dq=art+in+the+age+of+mechanical+reproduction&ots=NvcEjEayvd&sig=GCwxyz0pqx9N3C5ie6MJw4u_ijY.

Bense, Max. 1965. “Aesthetica Einführung in Die Neue Aesthetik.”

Berger, John. 2008a. *Ways of Seeing*. Vol. 1. Penguin uK. https://books.google.co.uk/books?hl=en&lr=&id=ilu4K-Xp0v4C&oi=fnd&pg=PT1&dq=ways+of+seeing&ots=44AdYJz4aC&sig=3Snd_NrwsoqVpKDUJBWOoy-2pSc.

———. 2008b. *Ways of Seeing*. Vol. 1. Penguin uK. https://books.google.co.uk/books?hl=en&lr=&id=ilu4K-Xp0v4C&oi=fnd&pg=PT1&dq=ways+of+seeing&ots=44AdYJz4aC&sig=3Snd_NrwsoqVpKDUJBWOoy-2pSc.

Berry, David M., and Michael Dieter. 2015. “Thinking Postdigital Aesthetics: Art, Computation and Design.” In *Postdigital Aesthetics*, 1–11. Springer.

Bishop, Claire. 2006. “Antagonism and Relational Aesthetics.” http://www.mitpressjournals.org/doi/abs/10.1162/0162287042379810.

Blaikie, Norman. 2009. *Designing Social Research*. Polity.

Boellstorff, Tom. 2015. *Coming of Age in Second Life: An Anthropologist Explores the Virtually Human*. Princeton University Press.

Bogost, Ian. 2012. *Alien Phenomenology, Or, What It’s Like to Be a Thing*. U of Minnesota Press.

Borden, Iain, J. Rendell, J. Kerr, and A. Pivaro. 2001. “Another Pavement, Another Beach: Skateboarding and the Performative Critique of Architecture.” http://discovery.ucl.ac.uk/26049/1/Borden_Another_Pavement.pdf.

Borges, Jorge Luis. 1971. *The Aleph and Other Stories, 1933-1969: Together with Commentaries and an Autobiographical Essay*. Bantam Books.

Bourriaud, Nicolas. 2002. *Postproduction: Culture as Screenplay: How Art Reprograms the World*. Lukas & Sternberg New York. http://classes.dma.ucla.edu/Fall10/161A/content/readings/bourriaud-postproduction-introduction.pdf.
Bourriaud, Nicolas, Caroline Schneider, and Jeanine Herman. 2002. *Postproduction: Culture as Screenplay: How Art Reprograms the World*. Lukas & Sternberg New York. http://classes.dma.ucla.edu/Fall10/161A/content/readings/bourriaud-postproduction-introduction.pdf.
Brown, Kathryn. 2014. *Interactive Contemporary Art: Participation in Practice*. IB Tauris.
Bucher, Taina. 2015. "About a Bot: Interview with Katie Rose Pipkin." Furtherfield. December 7, 2015. https://www.furtherfield.org/about-a-bot-interview-with-katie-rose-pipkin/.
———. 2017a. "The Algorithmic Imaginary: Exploring the Ordinary Affects of Facebook Algorithms." *Information, Communication & Society* 20 (1): 30–44. https://doi.org/10.1080/1369118X.2016.1154086.
———. 2017b. "The Algorithmic Imaginary: Exploring the Ordinary Affects of Facebook Algorithms." *Information, Communication & Society* 20 (1): 30–44. https://doi.org/10.1080/1369118X.2016.1154086.
Burrell, Jenna. 2015. "How the Machine 'Thinks:' Understanding Opacity in Machine Learning Algorithms." http://ssrn.com/abstract=2660674.
Butler, Judith. 2011. *Bodies That Matter: On the Discursive Limits of Sex*. routledge.
Čapek, Karel. 2004. *RUR (Rossum's Universal Robots)*. Penguin.
Cardew, Cornelius. 1967. *Treatise Handbook*. Gallery Upstairs Press.
Castañeda, Claudia, and Lucy Suchman. 2014. "Robot Visions." *Social Studies of Science* 44 (3): 315–341.
Colton, Simon. 2012. "The Painting Fool: Stories from Building an Automated Painter." In *Computers and Creativity*, 3–38. Springer.
Cormen, Thomas H., Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2001. *Introduction to Algorithms*. Vol. 2. MIT press Cambridge. http://euler.slu.edu/~goldwasser/courses/loyola/comp363/2003_Spring/handouts/course-info.pdf.
Cornell, Lauren, Ed Halter, and Lisa Phillips. 2015. *Mass Effect: Art and the Internet in the Twenty-First Century*. Vol. 1. MIT Press.
Cramer, Florian. 2015. "What Is 'Post-Digital'?" In *Postdigital Aesthetics*, 12–26. Springer.
Cramer, Florian, and Ulrike Gabriel. 2001. "Software Art." *American Book Review, Issue "Codeworks"(Alan Sondheim, Ed.)*.
Dance, Gabriel J. X., Michael LaForgia, and Nicholas Confessore. 2018. "As Facebook Raised a Privacy Wall, It Carved an Opening for Tech Giants." *The New York Times*, December 18, 2018, sec. Technology. https://www.nytimes.com/2018/12/18/technology/facebook-privacy.html.
Deleuze, Gilles. 1990. "Expressionism in Philosophy: Spinoza."
Deleuze, Gilles, and Félix Guattari. 1987a. *A Thousand Plateaus: Capitalism and Schizophrenia*. Minneapolis: University of Minnesota Press.
———. 1987b. *A Thousand Plateaus: Capitalism and Schizophrenia*. Minneapolis: University of Minnesota Press.
———. 2014. *What Is Philosophy?* Columbia University Press. https://books.google.co.uk/books?hl=en&lr=&id=sk4xBgAAQBAJ&oi=fnd&pg=PR5&dq=what+is+philosophy+deleuze&ots=f2R-6NOlFA&sig=rZYM1rXByLNpb_UzgBHqko3aW3I.
Dietrich, Frank. 1986. "Visual Intelligence: The First Decade of Computer Art (1965–1975)." *Leonardo* 19 (2): 159–169.

Doctorow, Cory. 2016. “Artists Troll Thingiverse with 3D Model Mashup Bot / Boing Boing.” May 3, 2016. https://boingboing.net/2016/05/03/artists-troll-thingiverse-with.html.
Dourish, P. 2016. “Algorithms and Their Others: Algorithmic Culture in Context.” *Big Data & Society* 3 (2). https://doi.org/10.1177/2053951716665128.
Drass, Eric. 2016. “@bffbot1.” June 18, 2016. http://www.shardcore.org/shardpress/2014/06/18/bffbot1/.
Dredge, Stuart. 2016. “Why Facebook and Microsoft Say Chatbots Are the Talk of the Town.” *The Observer*, September 18, 2016, sec. Technology. https://www.theguardian.com/technology/2016/sep/18/chatbots-talk-town-interact-humans-technology-silicon-valley.
Eco, Umberto. 1989. “The Open Work.” https://philpapers.org/rec/ECOTOW.
Feynman, Richard Phillips, J. G. Hey, and Robin W. Allen. 1998. *Feynman Lectures on Computation*. Addison-Wesley Longman Publishing Co., Inc. http://dl.acm.org/citation.cfm?id=550856.
Fielding, Roy T. 2000. *Architectural Styles and the Design of Network-Based Software Architectures*. Vol. 7. University of California, Irvine Doctoral dissertation.
Fielding, Roy T., and Richard N. Taylor. 2002. “Principled Design of the Modern Web Architecture.” *ACM Transactions on Internet Technology (TOIT)* 2 (2): 115–150.
Franta, William R., and Kurt Maly. 1977. “An Efficient Data Structure for the Simulation Event Set.” *Communications of the ACM* 20 (8): 596–602.
Frayling, Christopher. 1993. “Research in Art and Design.”
Fuller, Matthew. 2002. “Behind the Blip: Software as Culture.” *Nettime Mail. List* 7. http://noemalab.eu/wp-content/uploads/2011/09/fuller_sw_as_culture.pdf.
Fuller, Matthew, Roger F. Malina, and Sean Cubitt. 2008. *Software Studies: A Lexicon*. Mit Press.
Fuller, Matthew, and others. 2003. *Behind the Blip: Essays on the Culture of Software*. JSTOR. http://www.jstor.org/stable/pdf/4489252.pdf.
Fulton, Jonathan. 2017. “Web Architecture 101.” VideoBlocks Product & Engineering. November 7, 2017. https://engineering.videoblocks.com/web-architecture-101-a3224e126947.
Garcia, Laia. 2016. “Twitter Poems from Outer Space.” Lenny Letter. October 9, 2016. http://www.lennyletter.com/culture/interviews/a582/twitter-poems-from-outer-space/.
Gaver, William. 2012. “What Should We Expect from Research through Design?” In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 937–946. ACM. http://dl.acm.org/citation.cfm?id=2208538.
Geiger, R. Stuart. 2012. “Bots Are Users, Too! Rethinking the Roles of Software Agents in HCI.” *TinyToCS* 1. http://tinytocs.ece.utexas.edu/vol1/papers/tinytocs-v1-stuart.pdf.
Gillespie, Tarleton. 2014a. “9 The Relevance of Algorithms.” *Media Technologies: Essays on Communication, Materiality, and Society*, 167.
———. 2014b. “9 The Relevance of Algorithms.” *Media Technologies: Essays on Communication, Materiality, and Society*, 167.
Gleadowe, Teresa. 2000. “Curating in a Changing Climate.” *Curating in the 21st Century, New Art Gallery Walsall*.
Gorton, Thomas. 2015a. “This Robot Reviews Art Better than Most Critics.” Dazed. January 15, 2015. http://www.dazeddigital.com/artsandculture/article/23213/1/this-robot-reviews-art-better-than-most-critics.
———. 2015b. “This Robot Reviews Art Better than Most Critics.” Dazed. January 15, 2015. http://www.dazeddigital.com/artsandculture/article/23213/1/this-robot-reviews-art-better-than-most-critics.

Graham, Beryl, and Sarah Cook. 2010. *Rethinking Curating: Art after New Media-Leonardo Books*. The MIT Press.
Gurevich, Yuri. 2012. "What Is an Algorithm?" In *SOFSEM 2012: Theory and Practice of Computer Science*, 31–42. Springer. http://link.springer.com/chapter/10.1007/978-3-642-27660-6_3.
Haraway, Donna. 1996a. "Modest Witness: Feminist Diffractions in Science Studies."
———. 1996b. "Modest-Witness, Second-Millennium: Femaleman Meets Oncomouse: Feminism and Technoscience."
———. 2016. *Staying with the Trouble: Making Kin in the Chthulucene*. Duke University Press.
Harrison, Mark. 1995. *The USENET Handbook: A User's Guide to Netnews*. O'Reilly & Associates.
Hern, Alex. 2016. "Move over, Chatbots: Meet the Artbots," April 15, 2016, sec. Technology. https://www.theguardian.com/technology/2016/apr/15/move-over-chatbots-meet-the-artbots.
———. 2017. "Facebook and Twitter Are Being Used to Manipulate Public Opinion – Report." *The Guardian*, June 19, 2017, sec. Technology. https://www.theguardian.com/technology/2017/jun/19/social-media-proganda-manipulating-public-opinion-bots-accounts-facebook-twitter.
Hine, Christine. 2015. *Ethnography for the Internet: Embedded, Embodied and Everyday*. Bloomsbury Publishing.
Hitsch, Günter J., Ali Hortaçsu, and Dan Ariely. 2005. "What Makes You Click: An Empirical Analysis of Online Dating." In *2005 Meeting Papers*. Vol. 207. Society for Economic Dynamics.
Hockstein, Neil G., C. G. Gourin, R. A. Faust, and David J. Terris. 2007. "A History of Robots: From Science Fiction to Surgical Robotics." *Journal of Robotic Surgery* 1 (2): 113–118.
Hopkins, Nick. 2017. "Facebook Moderators: A Quick Guide to Their Job and Its Challenges | News | The Guardian." May 21, 2017. https://www.theguardian.com/news/2017/may/21/facebook-moderators-quick-guide-job-challenges.
Horn, Leslie. 2014. "The Most Popular Tweets From @Everyword's Esteemed Career." Gizmodo. Accessed May 12, 2017. http://gizmodo.com/the-most-popular-tweets-from-everywords-esteemed-caree-1585309345.
Hunt, Elle. 2016. "Tay, Microsoft's AI Chatbot, Gets a Crash Course in Racism from Twitter." *The Guardian*, March 24, 2016, sec. Technology. https://www.theguardian.com/technology/2016/mar/24/tay-microsofts-ai-chatbot-gets-a-crash-course-in-racism-from-twitter.
Introna, L. D. 2015. "Algorithms, Governance, and Governmentality: On Governing Academic Writing." *Science, Technology & Human Values*, June. https://doi.org/10.1177/0162243915587360.
Kawano, Hiroshi. 1976. *What Is Computer Art?* Creative Computing Press, Morristown, NJ.
Kiros, Ryan, Ruslan Salakhutdinov, and Richard S. Zemel. 2014. "Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models." *ArXiv Preprint ArXiv:1411.2539*. https://arxiv.org/abs/1411.2539.
Kitchin, Rob. 2014. "Thinking Critically About and Researching Algorithms." http://papers.ssrn.com/sol3/Papers.cfm?abstract_id=2515786.
Knowles, J. Gary, and Ardra L. Cole. 2008. *Handbook of the Arts in Qualitative Research: Perspectives, Methodologies, Examples, and Issues*. Sage.
Knuth, Donald E. 1997. *The Art of Computer Programming: Volume 1: Fundamental Algorithms*. Addison-Wesley Professional.
Kohlick, Anna. 2015a. "'Dem Bot Ist Egal, Ob Es Ein Warhol Ist' | Monopol – Magazin Für Kunst Und Leben." March 20, 2015. http://www.monopol-magazin.de/dem-bot-ist-egal-ob-es-ein-warhol-ist.

———. 2015b. "'Dem Bot Ist Egal, Ob Es Ein Warhol Ist' | Monopol – Magazin Für Kunst Und Leben." March 20, 2015. http://www.monopol-magazin.de/dem-bot-ist-egal-ob-es-ein-warhol-ist.
Krauss, Rosalind E., and Marcel Broodthaers. 2000. *A Voyage on the North Sea: Art in the Age of the Post-Medium Condition*. Thames & Hudson London. http://www.openbibart.fr/item/display/10068/721427.
Lakhani, Karim R., and Robert G. Wolf. 2003. "Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects."
Latour, Bruno. 1996. "On Actor-Network Theory: A Few Clarifications." *Soziale Welt*, 369–381.
———. 1999. *Pandora's Hope: Essays on the Reality of Science Studies*. Harvard University Press.
Law, John. 2004. *After Method: Mess in Social Science Research*. Routledge.
Lee, Peter. 2016. "Learning from Tay's Introduction." The Official Microsoft Blog. March 25, 2016. https://blogs.microsoft.com/blog/2016/03/25/learning-tays-introduction/.
Leonard, Andrew. 1996. "Bots Are Hot." *Wired* 4 (4): 114–117.
———. 1998. *Bots: The Origin of New Species*. Penguin Books Limited.
Loeliger, Jon, and Matthew McCullough. 2012. *Version Control with Git: Powerful Tools and Techniques for Collaborative Software Development*. O'Reilly Media, Inc.
Lury, Celia, and Nina Wakeford. 2012. *Inventive Methods: The Happening of the Social*. Routledge.
Mackenzie, Adrian. 2018. "48 Million Configurations and Counting: Platform Numbers and Their Capitalization." *Journal of Cultural Economy* 11 (1): 36–53. https://doi.org/10.1080/17530350.2017.1393443.
MacKenzie, Donald. 2005a. "Opening the Black Boxes of Global Finance." *Review of International Political Economy* 12 (4): 555–76. https://doi.org/10.1080/09692290500240222.
———. 2005b. "Opening the Black Boxes of Global Finance." *Review of International Political Economy* 12 (4): 555–76. https://doi.org/10.1080/09692290500240222.
———. 2014. "A Sociology of Algorithms: High-Frequency Trading and the Shaping of Markets." *Preprint. School of Social and Political Science, University of Edinburgh*.
MacKenzie, Donald, and Judy Wajcman. 1999. *The Social Shaping of Technology*. 2nd. Open university press.
Manovich, Lev. 2001. *The Language of New Media*. MIT press.
———. 2013a. *Software Takes Command*. Vol. 5. A&C Black. https://books.google.co.uk/books?hl=en&lr=&id=RarTbJ84EnMC&oi=fnd&pg=PR9&dq=software+takes+command&ots=jZiThFfIor&sig=mwZ9ZijyAXyk5_hjEucW8r7A9TQ.
———. 2013b. *Software Takes Command*. Vol. 5. A&C Black. https://books.google.co.uk/books?hl=en&lr=&id=RarTbJ84EnMC&oi=fnd&pg=PR9&dq=software+takes+command&ots=jZiThFfIor&sig=mwZ9ZijyAXyk5_hjEucW8r7A9TQ.
Marres, Noortje. 2012. "The Redistribution of Methods: On Intervention in Digital Social Research, Broadly Conceived." *The Sociological Review* 60 (1_suppl): 139–165.
Marres, Noortje, Michael Guggenheim, and Alex Wilkie. 2018. "Inventing the Social."
Mauldin, Michael L. 1994. "ChatterBots, TinyMuds, and the Turing Test: Entering the Loebner Prize Competition," 6.
Merrill, Duane. 2006. "Mashups: The New Breed of Web App." August 8, 2006. http://www.ibm.com/developerworks/library/x-mashups/index.html.
Mirapaul. 1997. "In John Simon's Art, Everything Is Possible." *Arts At Large*. http://numeral.com/articles/041797mirapaul/041797mirapaul.html.

Moschovakis, Yiannis N. 2000. *What Is an Algorithm?*
Musante, Kathleen, and Billie R. DeWalt. 2010. *Participant Observation: A Guide for Fieldworkers*. Rowman Altamira.
Newitz, Annalee. 2016. "That Time a Bot Invaded Thingiverse and Created Weird New 3D Objects." Ars Technica. May 2, 2016. https://arstechnica.com/gadgets/2016/05/that-time-a-bot-invaded-thingiverse-and-created-weird-new-3d-objects/.
Neyfakh, Leon. 2014. "The Botmaker Who Sees through the Internet - The Boston Globe." BostonGlobe.Com. 2014. https://www.bostonglobe.com/ideas/2014/01/24/the-botmaker-who-sees-through-internet/V7Qn7HU8TPPl7MSM2TvbsJ/story.html.
Obrist, Hans-Ulrich. 2000. "Kraftwerk, Time Storage, Laboratory." *Curating in the 21st Century, New Art Gallery Walsall*.
Olsen, Stephanie. 2003. "AOL Tries out New IM Chat Bot." CNET. January 2003. https://www.cnet.com/news/aol-tries-out-new-im-chat-bot/.
Olson, Marisa. 2012. "Postinternet." *Foam Magazine* 29: 59–63.
Orlikowski, Wanda J. 2007. "Sociomaterial Practices: Exploring Technology at Work." *Organization Studies* 28 (9): 1435–1448.
Orlikowski, Wanda J., and Susan V. Scott. 2008. "Sociomateriality: Challenging the Separation of Technology, Work and Organization." *The Academy of Management Annals* 2 (1): 433–474.
Parrish, Allison. 2015. "Decontextualize · The Ephemerides." *Allison Parrish: Words and Projects* (blog). August 2015. /2015/08/the-ephemerides/.
Paul, Christiane. 2003. *Digital Art*. Vol. 224. Thames & Hudson London.
Perry, Dewayne E., and Alexander L. Wolf. 1992. "Foundations for the Study of Software Architecture." *ACM SIGSOFT Software Engineering Notes* 17 (4): 40–52.
Pickering, Andrew. 2010. *The Mangle of Practice: Time, Agency, and Science*. University of Chicago Press.
Pitcher, Jenna. 2014. "Microsoft Unveils Halo's Cortana as Siri-Style Windows Phone Digital Assistant." Polygon. April 3, 2014. https://www.polygon.com/2014/4/3/5577322/microsoft-unveils-halos-cortana-as-siri-style-windows-phone-digital.
Power, Mike. 2014. "What Happens When a Software Bot Goes on a Darknet Shopping Spree?" *The Guardian*, December 5, 2014, sec. Technology. https://www.theguardian.com/technology/2014/dec/05/software-bot-darknet-shopping-spree-random-shopper.
Quaranta, Domenico. 2013. *Beyond New Media Art*. Lulu. com.
Rea, Naomi. 2018. "Has Artificial Intelligence Brought Us the Next Great Art Movement? Here Are 9 Artists Who Are Exploring AI's Creative Potential." Artnet News. November 6, 2018. https://news.artnet.com/market/9-artists-artificial-intelligence-1384207.
Read, Max. 2012. "Weird Internets: The Amazing Found-on-Twitter Sonnets of Pentametron." Gawker. 2012. http://gawker.com/5905550/weird-internets-the-amazing-found-on-twitter-sonnets-of-pentametron.
Riemer, Jeffrey W. 1977. "Varieties of Opportunistic Research." *Urban Life* 5 (4): 467–477.
Rodley, Chris. 2015. "Magic Realism Bot." *Chris Rodley* (blog). October 10, 2015. https://chrisrodley.com/2015/10/10/magic-realism-bot/.
———. 2016. "Introducing Chris Rodley / The Art of Bots | Abandon Normal Devices." April 26, 2016. http://www.andfestival.org.uk/blog/introducing-chris-rodley-the-art-of-bots/.
Sample, Mark. 2014. "Closed Bots and Green Bots." June 23, 2014. http://www.samplereality.com/2014/06/23/closed-bots-and-green-bots/.

———. 2015. “A Protest Bot Is a Bot so Specific You Can’t Mistake It for Bullshit.” Samplereality. October 3, 2015. http://www.samplereality.com/?p=3322.
Sanders, Teela. 2005. “‘It’s Just Acting’: Sex Workers’ Strategies for Capitalizing on Sexuality.” *Gender, Work & Organization* 12 (4): 319–342.
Sankin, Aaron. 2014. The Daily Dot: Forecasting the 2014 Midterms with America’s Favorite Robot Teenager. 2014. https://www.dailydot.com/layer8/olivia-taters-political-pundit/.
Sayers, Jentery. 2018. *The Routledge Companion to Media Studies and Digital Humanities*. Routledge.
Schwartz, Morris S., and Charlotte Green Schwartz. 1955. “Problems in Participant Observation.” *American Journal of Sociology* 60 (4): 343–353.
Seaver, Nick. 2012. “Algorithmic Recommendations and Synaptic Functions.” *Limn* 1 (2). http://escholarship.org/uc/item/7g48p7pb.
Seaver, Nick. 2014. “Knowing Algorithms.” In .
Seaver, Nick. 2017. “Algorithms as Culture: Some Tactics for the Ethnography of Algorithmic Systems.” *Big Data & Society* 4 (2): 2053951717738104.
Serres, Michel. 1982. *The Parasite*.
Severance, Charles. 2015. “Roy T. Fielding: Understanding the REST Style.” *Computer* 48 (6): 7–9. https://doi.org/10.1109/MC.2015.170.
Slavin, Kevin. 2011. “How Algorithms Shape Our World.” *TED Talks*.
Solon, Olivia. 2017. “Russia-Backed Facebook Posts ‘reached 126m Americans’ during US Election.” *The Guardian*, October 31, 2017, sec. Technology. https://www.theguardian.com/technology/2017/oct/30/facebook-russia-fake-accounts-126-million.
Spradley, James P. 2016. *Participant Observation*. Waveland Press.
Staikidis, Kryssi. 2006. “Personal and Cultural Narrative as Inspiration: A Painting and Pedagogical Collaboration with Mayan Artists.” *Studies in Art Education* 47 (2): 118–138.
Stallabrass, Julian. 2003. “Internet Art: The Online Clash of Culture and Commerce.”
Steiner, Christopher. 2012. *Automate This: How Algorithms Took Over Our Markets, Our Jobs, and the World*. Penguin.
Stengers, Isabelle. 2005. “The Cosmopolitical Proposal.” *Making Things Public: Atmospheres of Democracy* 994: 994.
Stinson, Liz. n.d. “Looking for a Hoodie With NSA Documents on It? This Is Your Store.” WIRED. Accessed May 17, 2017. https://www.wired.com/2014/12/looking-hoodie-nsa-documents-store/.
Suchman, Lucy. 1987. *Plans and Situated Actions: The Problem of Human-Machine Communication*. Cambridge university press.
———. 2007. *Human-Machine Reconfigurations: Plans and Situated Actions*. Cambridge University Press.
———. 2010. “Agencies at the Interface: Expanding Frames and Accountable Cuts.” Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.
Taylor, Astra. 2018. “The Automation Charade.” Logic Magazine. October 2, 2018. https://logicmag.io/05-the-automation-charade/.
Taylor, Harriet. 2016. “Bank of America Launches AI Chatbot Erica -- Here’s What It Does.” October 24, 2016. https://www.cnbc.com/2016/10/24/bank-of-america-launches-ai-chatbot-erica--heres-what-it-does.html.
“This C-SPAN Bot Proves Politicians Say the Same Things.” n.d. Creators. Accessed March 21, 2017. https://creators.vice.com/en_us/article/this-c-span-bot-proves-politicians-say-the-same-things.
Troemel, Brad. 2011. *Peer Pressure*. Lulu. com.
Veale, Tony, and Mike Cook. 2018. *Twitterbots: Making Machines That Make Meaning*. MIT Press.

Von Ahn, Luis. 2008. “Human Computation.” In *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, 1–2. IEEE Computer Society.
Webb, Matt. 2015. “My Latest Twitter Bot: @5point9billion (14 Dec., 2015, at Interconnected).” Interconnected.Org. December 14, 2015. http://interconnected.org/home/2015/12/14/5point9billion.
Wilkie, Alex. 2010. “User Assemblages in Design: An Ethnographic Study.” Goldsmiths, University of London. http://eprints.gold.ac.uk/4710.
Wilkie, Alex, Mike Michael, and Matthew Plummer-Fernandez. 2014. “Speculative Method and Twitter: Bots, Energy and Three Conceptual Characters.” *The Sociological Review*, August, n/a-n/a.
Wirth, Niklaus. 1978. *Algorithms + Data Structures = Programs*. Upper Saddle River, NJ, USA: Prentice Hall PTR.
Yu, Jin, Boualem Benatallah, Fabio Casati, and Florian Daniel. 2008a. “Understanding Mashup Development.” *IEEE Internet Computing* 12 (5). http://ieeexplore.ieee.org/abstract/document/4620093/.
———. 2008b. “Understanding Mashup Development.” *IEEE Internet Computing* 12 (5). http://ieeexplore.ieee.org/abstract/document/4620093/.
Zimmerman, John, Jodi Forlizzi, and Shelley Evenson. 2007. “Research through Design as a Method for Interaction Design Research in HCI.” In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 493–502. ACM.

Appendix I: Practice-based work

I.I Artbot URLs

Every User

https://twitter.com/Every_User_

http://www.plummerfernandez.com/Every-User

Shiv Integer

https://www.thingiverse.com/shivinteger/designs

http://www.plummerfernandez.com/Shiv-Integer

http://www.plummerfernandez.com/Shiv-Integer-stage-2

Novice Art Blogger

noviceartblogger.tumblr.com/

http://www.plummerfernandez.com/Novice-Art-Blogger

Petita Tatata

https://soundcloud.com/petita_tatata

http://www.plummerfernandez.com/Petita-Tatata

https://github.com/plummerfernandez/Petita-Tatata

The Revenant

https://twitter.com/Revenantbot

All Eyes

https://twitter.com/_all_eyes

http://www.plummerfernandez.com/All-Eyes

I.II Exhibitions

Abandon Normal Devices, Art of Bots, *London (UK) (2016)*
http://www.andfestival.org.uk/events/art-of-bots-london

iMAL - Shiv Integer, *Brussels (BE) (2017)*
http://imal.org/en/exhibition/shiv_integer

Stedelijk Museum, Algorithmic Rubbish Daring to Defy Misfortune *(NL) (2015)*
http://www.smba.nl/en/exhibitions/algorithmic-rubbish-daring-to-d/

Hull City of Culture, States of Play *(UK) (2017)*
https://www.craftscouncil.org.uk/what-we-do/states-of-play

MU Gallery, Eindhoven *(NL) (2017)*
http://mu.nl/en/exhibitions/materialising-the-internet

HEK, Liste Art Fair, *(CH) (2017)*
http://www.hek.ch/en/program/events-en/event/hek-liste-art-fair-basel-1.html

I.III Exhibitions marketing material


19 28
APRIL → MAY
EXHIBITION
WEDNESDAY → SUNDAY
13:00 - 18:00,
FREE ENTRANCE
VERNISSAGE
WEDNESDAY 19.04.2017, 18:30
#ADDITIVISM EVENTS
WITH DANIEL ROURKE
LECTURE
"THE 3D ADDITIVIST
MANIFESTO + COOKBOOK"
AT BOZAR, 24.05, 19:00
MASTERCLASS
AT iMAL, 25 - 26.05
(please register on imal.org)
More info:
www.imal.org
What the hell is this?
- mnturner
The moment of joy
when you've been shivintegered!
- unfold
Is this a different form of spam?
Is this an AI attempting
to learn something?
If so, can you respond to inquiry?
- ubergeekseven
It looks like you've recently learned
how to stick things to each other.
Who's a clever boy then?
- NZbot
Glad to see my Universal J-Hook
is going to a good use.
Your designs send me to the floor
rolling over and laughing. Shiv!
- Gazorpa
Take a moment and think.
What if this is not spam?
What if someone desperately needs
an out, a way to communicate?
- 29autumn
Why do you keep making these?
- Akura2
Vlaanderen
bruxsel
In partnership with
iMAL, KOOLMIJNENKAAI, 30, QUAI DES CHARBONNAGES, B-1080 BRUSSELS

EN

Every day, Shiv Integer randomly selects 3D printable objects available on a website, assembles them into sculptures and posts them back online. Shiv Integer's particularity? It's actually an automatic software robot – a bot – created by Julien Deswaef and Matthew Plummer-Fernandez.

Thingiverse is the biggest online 3D-Print community and a vast archive of user-made models - full of knick-knacks and engineering parts. Rummaging through this collection, Shiv Integer picks objects at random to conjoin into sculptures and gives them apt word-salad names such as "disc on top of an e-juice golf".

The process follows a lineage of Dadaist readymade and chance art, but also explores the authorship-inheritance of Creative Commons licensing (which makes each designer of the original 3D models a co-author of the final assemblage). Lastly, it performs an archiving of an Internet subculture, taking snapshots of 3D print culture across the whole Thingiverse database.

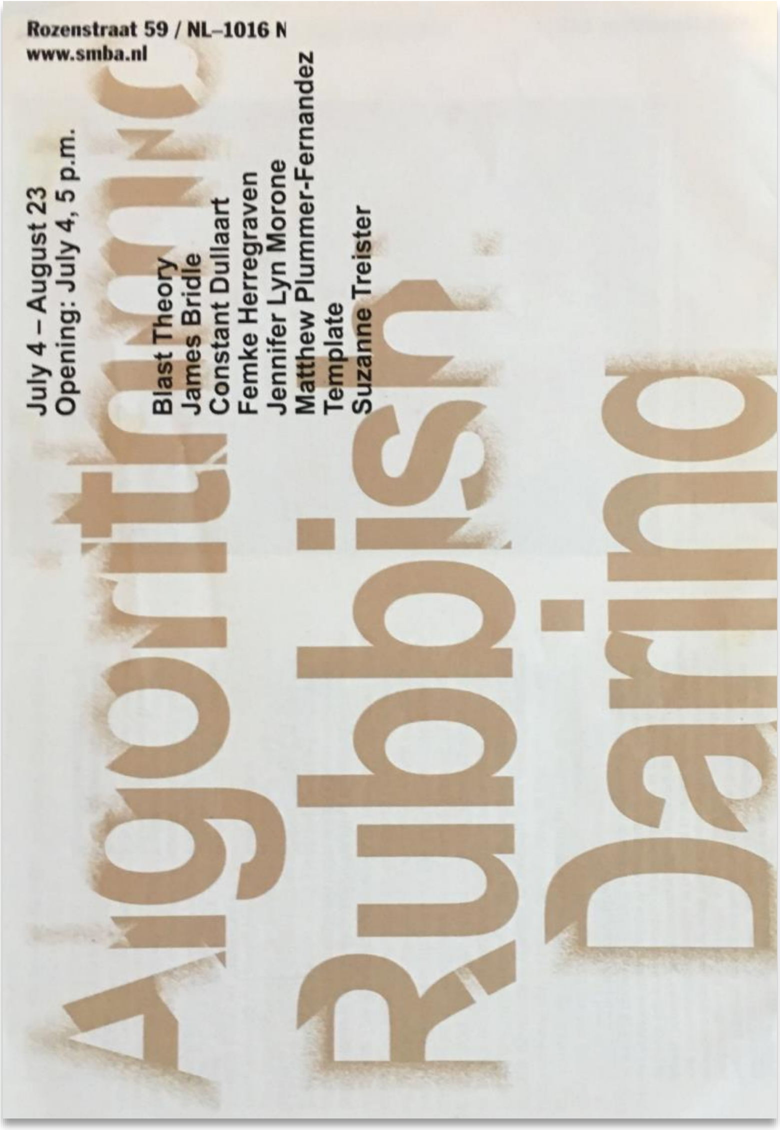
The bot ran anonymously with only a vague FAQ explanation. Thingiverse users either love or hate the bot; it's provoked hundreds of comments ranging from fan poetry to hate mail, and sparked a long debate over if it makes art or spam. User's binding stake in authorship made them fiercely active, forming a key facet of Shiv Integer.

FR

Tous les jours, Shiv Integer sélectionne aléatoirement, sur un site web, des objets à imprimer en 3D. Il les assemble sous forme de sculptures, qu'il partage à son tour en ligne. La particularité de Shiv Integer? C'est en réalité un robot logiciel (ou «bot») créé par Julien Deswaef et Matthew Plummer-Fernandez.

Thingiverse est le plus important site web communautaire autour de l'impression 3D. Ses membres y partagent des modèles 3D qu'ils ont créés, qu'il s'agisse de bibelots ou de pièces mécaniques. De cet immense collection, le bot de Shiv Integer extrait des objets sélectionnés au hasard, et les assemble pour créer des sculptures qu'il nomme des titres tout aussi improbables, tels que "disc on top of an e-juice golf".

Ce processus de création s'inscrit dans la lignée du ready made dadaïste, du procédé artistique de l'assemblage, et d'approches aléatoires dans l'art. La démarche explore également les questions de transmission du statut d'auteur propres aux licences Creative Commons.


Rozenstraat 59 / NL–1016 N
www.smba.nl
July 4 – August 23
Opening: July 4, 5 p.m.
Blast Theory
James Bridle
Constant Dullaart
Femke Herregraven
Jennifer Lyn Morone
Matthew Plummer-Fernandez
Template
Suzanne Treister

Algorithmic Rubbish: Daring to Defy Misfortune

Blast Theory, James Bridle, Constant Dullaart, Femke Herregraven, Jennifer Lyn Morone, Matthew Plummer-Fernandez, Template, and Suzanne Treister

July 4 – August 23
Opening: July 4, 5–7 p.m. with an address by Margriet Schavemaker (curator and head of research & publications at the Stedelijk Museum)

"Algorithmic Rubbish: Daring to Defy Misfortune" brings together projects that were made with new media and which largely unfold in the digital domain. In this respect, SMBA is acting as a sort of conduit, or a seductive space in which the digital works take on a more or less material shape. "Algorithmic Rubbish: Daring to Defy Misfortune" is based on technologies accessible in daily life, but which in contemporary art still have an almost separate status and require a specific approach.

With "Algorithmic Rubbish: Daring to Defy Misfortune," SMBA is engaging in the discussion surrounding the consequences of the digital revolution on art production and reception. In many ways, digital art confounds the logic of (museum) presentation institutions: it cannot be seen independently of hardware and interfaces, it is often co-authored, it has scarcely broken through in traditional collection circles like those of private collectors and museums, and it questions art's focus on the object. Just as in daily life, digital technologies are causing existential schizophrenia in the art world. Artificial intelligence is so advanced that we can have human feelings for lifeless programs and objects. Additionally, everything and everyone is always available, anywhere – as long as you have access to and with the right technology. For digital art, this has two consequences. Firstly, it is not clear *where* a work is, even if you're standing in front of it. Secondly, different

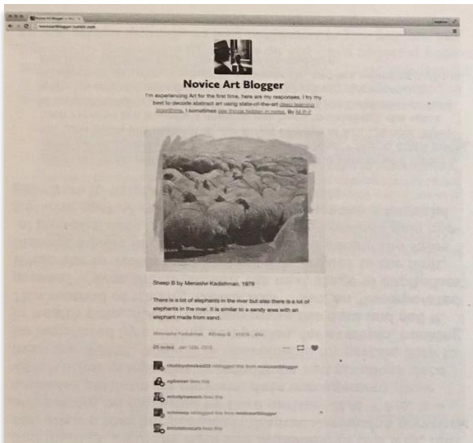
Algorithmic Rubbish: Daring to Defy Misfortune

Blast Theory, James Bridle, Constant Dullaart, Femke Herregraven, Jennifer Lyn Morone, Matthew Plummer-Fernandez, Template, Suzanne Treister

4 juli t/m 23 augustus
Opening: zaterdag 4 juli, 17:00–19:00 uur met een toespraak door Margriet Schavemaker (Conservator en Hoofd Onderzoek & Publicaties van het Stedelijk Museum)

'Algorithmic Rubbish: Daring to Defy Misfortune' brengt projecten bij elkaar die zijn gerealiseerd met nieuwe media en zich grotendeels afspelen in het digitale domein. SMBA fungeert hiervoor als een soort doorgeefluik, als een verleidingsruimte waarin de digitale werken meer of minder materiële manifestaties hebben gekregen. 'Algorithmic Rubbish: Daring to Defy Misfortune' is gebaseerd op technologieën die gangbaar zijn in het dagelijks leven, maar binnen de hedendaagse kunst nog een aparte status hebben en een specifieke omgang afdwingen.

Met 'Algorithmic Rubbish: Daring to Defy Misfortune' sluit SMBA aan bij de discussie over de gevolgen van de digitale revolutie voor de kunstproductie en -receptie. Op veel manieren gaat digitale kunst in tegen de logica van de (museale) presentatie-instelling, omdat ze niet los is te zien van *hardware* en *interfaces*, er vaak sprake is van gedeeld auteurschap, het werk nog slecht is doorgedrongen in de traditionele verzamelcircuits van particuliere collectioneurs en musea, en de status van objectgerichtheid in de kunst bevraagt. Net als in het dagelijks leven creëren digitale technologieën ook in de kunst existentiële schizofrenie. Artificiële intelligentie is zo ver gevorderd dat we mede-menselijke gevoelens voor levenloze programma's en objecten kunnen krijgen. Daarnaast is alles en iedereen altijd en overal beschikbaar, althans zolang je toegang tot



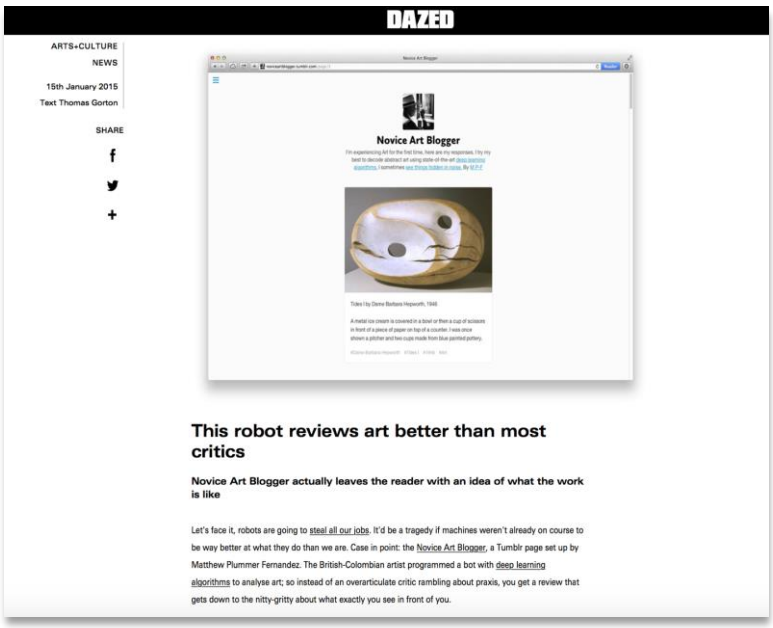
Matthew Plummer-Fernandez, *Novice Art Blogger*, 2015 (ongoing, screenshot)

Matthew Plummer-Fernandez
Novice Art Blogger, 2015 - ongoing

Matthew Plummer-Fernandez' *Novice Art Blogger* is een kunstblog dat automatisch beelden en metadata selecteert van abstracte kunstwerken uit het online archief van Tate, of elk ander museum – in dit geval het online archief van het Stedelijk Museum. Door middel van toegankelijke *Deep Learning* algoritmen, ontwikkeld door computerwetenschappers aan de universiteit van Toronto, worden de afbeeldingen verwerkt. De algoritmen proberen de inhoud van de afbeeldingen te ontcijferen en genereren bijschriften. Tegelijkertijd wordt er ook gezocht naar het meest gelijkende beeld in de database. De resultaten worden door de software naar *blogposts* vertaald. Het project verwijst naar de explosie van 'open data', en hoe musea, amateurkunst en cultuurbloggers op platformen zoals Tumblr hier mee omgaan. Daarnaast onderzoekt het project hoe *Neural Network en Machine Learning* technologieën leren en hoe ze reageren op abstracte kunst. Is dit de eerste stap naar automatische cureren van tentoonstellingen of waarderen van kunst?
http://noviceartblogger.tumblr.com

Matthew Plummer-Fernandez is Brits-Colombiaanse kunstenaar en onderzoeker. Hij onderzoekt op kritische en speelse wijze socio-culturele verwikkelingen met technologie. Op het moment richt zijn interesse zich op algoritmen, bots, automatie, copyright en file-sharing. Hij deelt zijn onderzoek op #algopop, een populaire tmblr die de verwikkelingen met algoritmen in het dagelijks leven en de reacties van kunstenaars op deze situatie documenteert. Dit is het uitgangspunt geworden voor een praktijkgerichte PhD, die ondersteund wordt door het AHRC Goldsmith Univeristy of London, waar hij werkte als research associate/technologist en gastdocent. http://www.plummerfernandez.com

I.IV Press



http://www.dazeddigital.com/artsandculture/article/23213/1/this-robot-reviews-art-better-than-most-critics

http://we-make-money-not-art.com/how-artists-and-designers-are-materialising-the-internet/

https://hyperallergic.com/tag/novice-art-blogger/

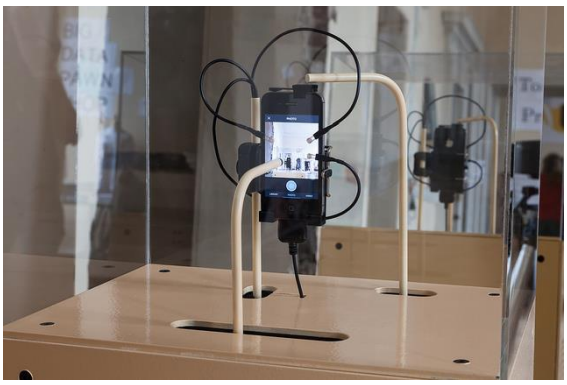
http://internet.gawker.com/tag/novice-art-blogger

https://www.good.is/articles/novice-art-blogger-art-critic-bot

Appendix II. *Art of Bots*

II.I Further documentation


Big Data Pawn Shop




SHOP
Koenig Books


HP
Glitch Logos
Darius Kazemi


A turtle made of
playing cards is born
in Munich.

Top to bottom:

Photo: Emma Winston, *Graphic Score Bot*

Photo: Sam Lavigne, *CSpan-5*

Photo: Shardcore, *MAMMON*

Photo: Thricedotted, *The Seeker*

Photo: Sam Lavigne, *Big Data Pawn Shop*

Photo: Daniel Armengol Altayo, *Artificial Selfie*

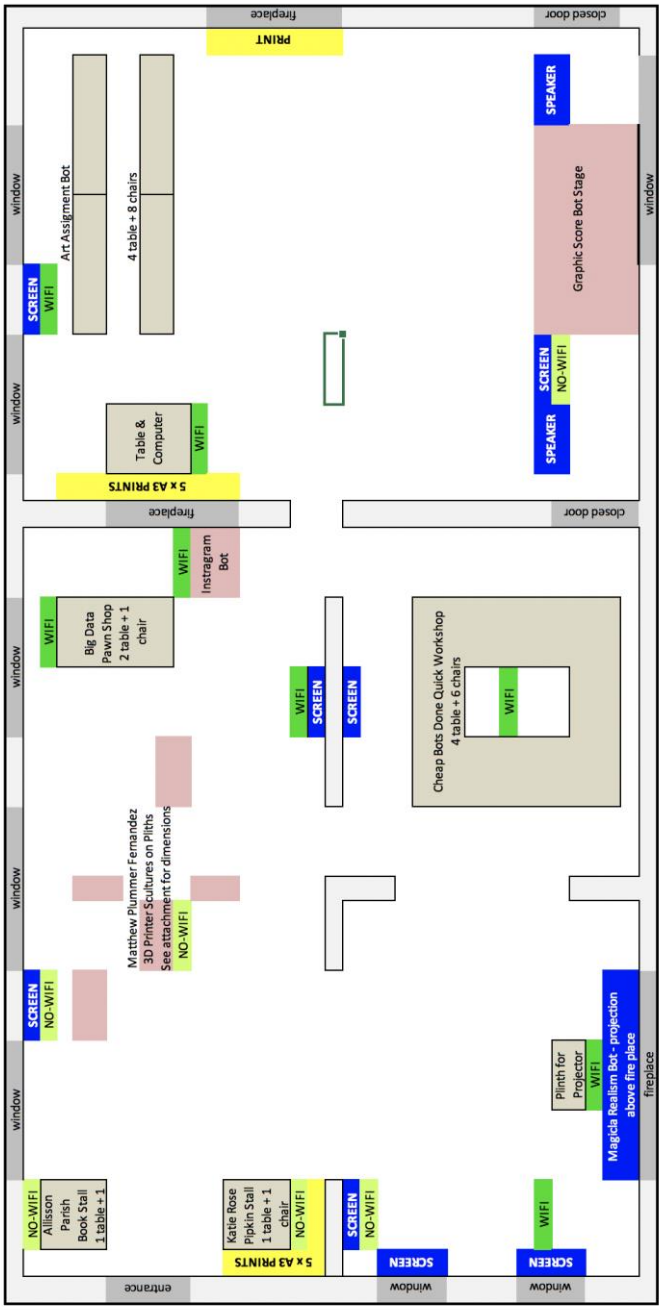
Photo: Jeff Thompson, *Bot Art School*

Photo: Matthew Plummer Fernandez and Julien Deswaef, *Shiv Integer*

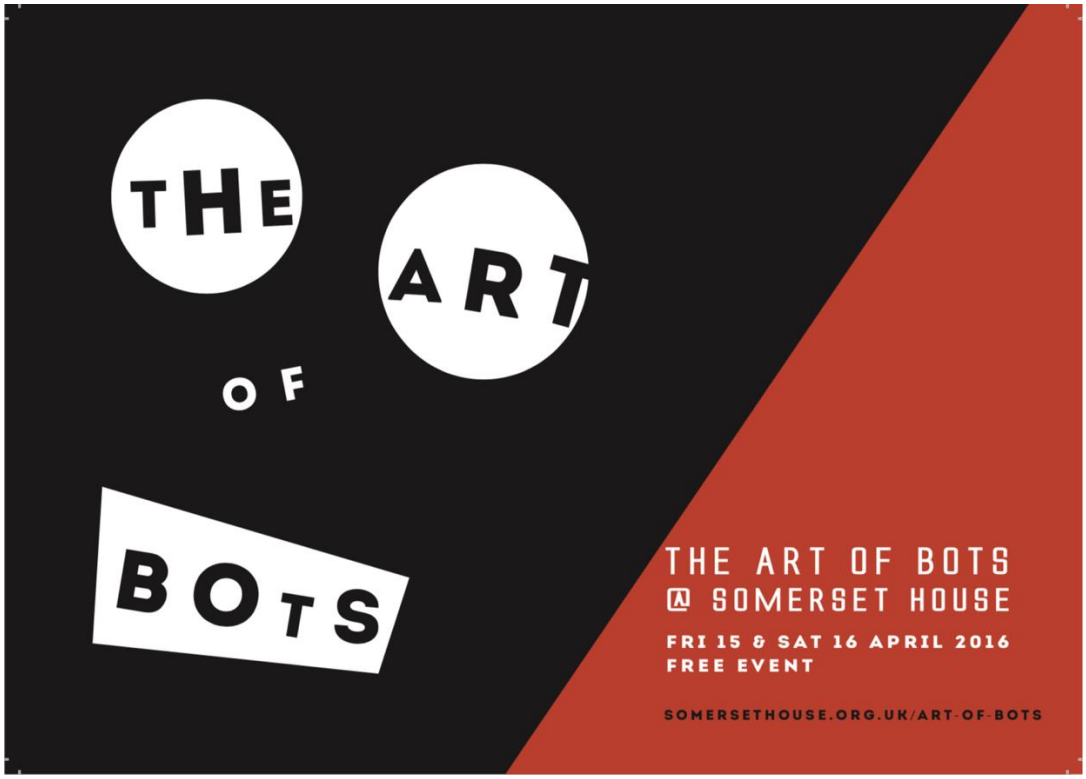
Photo: Darius Kazemi, *Glitch Logos*

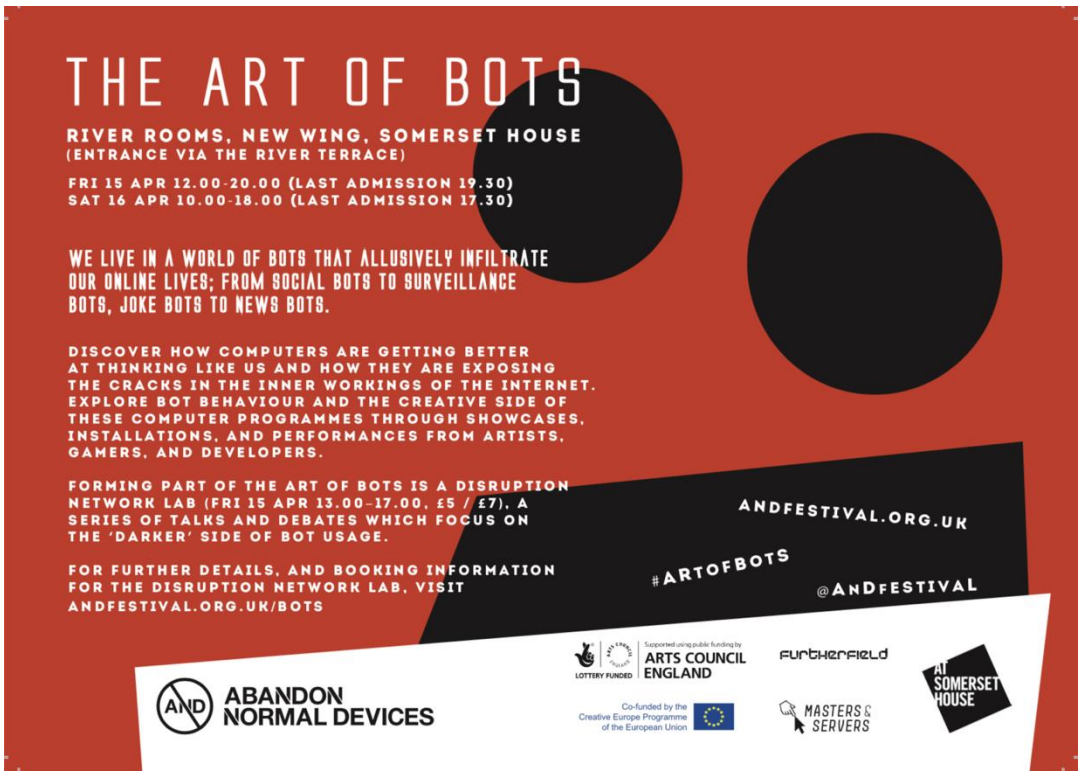
Photo: Chris and Ali Rodley, *Magic Realism Bot*

II.II Exhibition plan



II.III Marketing material





The Art of Bots

Friday 15 April: 12.00-20.00 (last admission 19.30)

Saturday 16 April: 10.00-18.00 (last admission 17.30)
New Wing, Somerset House, London

Over recent years tiny automated and nonsensical beings have infiltrated our daily life via twitter and email. They have taken delight in agitation by imitating human behaviours – from telling jokes to online-dating. These small computer programmes are called bots (derived from the word robot) and their repetitive and often simple actions have brought fun to – and often poked fun at - the Internet platforms that now mediate our lives.

This weekend of workshops, labs and bot showcases will be a chance to meet pioneers of the bot making community, gain an insight into their practices and also uncover the darker side of these computer programmes. Bot creators come in many different guises and our weekend will feature artists, whistleblowers, developers, gamers, comedians, thinkers and inventors from this global micro-community.

The showcase invites bot developers – artists, developers, gamers, thinkers and inventors – to draw on histories in automata and movements such as Dada and Fluxus. Featuring contributions from Darius Kazemi (USA), Matthew Plummer-Fernandez (UK) and Julien Deswaef (USA), Emma Winston (UK), Jeff Thompson (USA), Katie Rose Pipkin (USA), Allison Parrish (USA), Shardcore (UK), Matt Webb (UK), Chris Rodley (AUS), Sam Lavigne (USA), George Buckenham (UK), Thricedotted (USA) and Daniel Armengol Altayó (E).

The programme is co-curated by Abandon Normal Devices and Matthew Plummer-Fernandez.

www.andfestival.org.uk **#artofbot**

RELATED EVENTS

Friday 15 April 12:00 – 20:00 and Saturday 16 April 16 10:00 – 18:00, FREE
Cheap Bots, Done Quick! Drop-In Workshop
River Rooms, New Wing, Somerset House
George Buckenham leads a bot making workshop for bot novices and enthusiasts. Learn more about what bots are, how they work and even have a go at creating your own. The workshop will be simple and quick to complete, no programming knowledge is necessary at all – just an inquisitive and curious mind!

Friday 15 April 13:00 – 17:00, £5-£7
Disruption Network Lab / BOTS: Tracking Systems of Control
Lancaster Rooms, New Wing, Somerset House
A series of talks and presentations that consider and interrogate the political and artistic potential emerging from the relationship between surveillance and the use of bots. Who is responsible for the output and actions of bots, both ethically and legally?

Monday 18 April 2016 19:00, £5
Experimental Writing @ Carroll / Fletcher: Allison Parrish with Matthew Plummer-Fernandez
Carroll / Fletcher Gallery, 56 - 57 Eastcastle Street, London W1W 8EQ
The latest event in the Experimental Writing @ Carroll/Fletcher series features a talk by participating Showcase artist Allison Parrish. Allison will present some of her recent work with twitter bots and computer-generated poetry and discuss using procedural writing, bots and natural language processing techniques in creative practice. The talk will be followed by a conversation between Allison and Matthew Plummer-Fernandez.

Credits:

The Art of Bots is delivered in partnership with Somerset House and media partner Furtherfield. It is realised in the framework of Masters & Servers, a joint project by Aksioma (SI), Drugo more (HR), Abandon Normal Devices (UK), Link Art Center (IT) and d-i-n-a / The Influencers (ES).

Funded by:

Co-funded by the Creative Europe Programme of the European Union

Supported using public funding by ARTS COUNCIL ENGLAND
LOTTERY FUNDED

Partner AT SOMERSET HOUSE

Media partner furtherfield

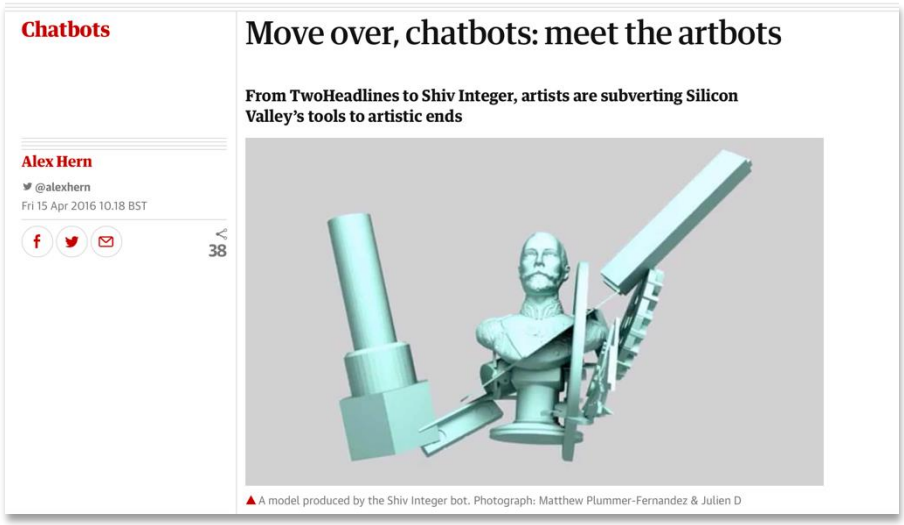
Part of: MASTERS & SERVERS

www.andfestival.org.uk **#artofbots**

II.IV Press coverage

The Guardian

https://www.theguardian.com/technology/2016/apr/15/move-over-chatbots-meet-the-artbots



Chatbots

Move over, chatbots: meet the artbots

From TwoHeadlines to Shiv Integer, artists are subverting Silicon Valley's tools to artistic ends

Alex Hern

@alexhern

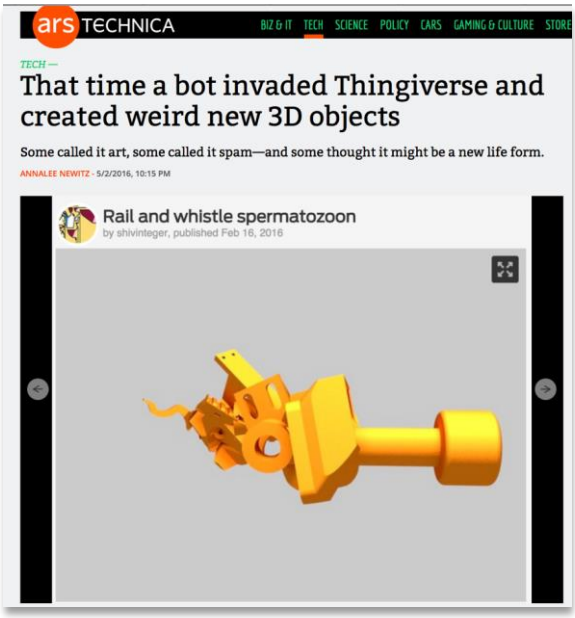
Fri 15 Apr 2016 10.18 BST

38

A model produced by the Shiv Integer bot. Photograph: Matthew Plummer-Fernandez & Julien D

Ars Technica

https://arstechnica.com/gadgets/2016/05/that-time-a-bot-invaded-thingiverse-and-created-weird- new-3d-objects/



ars TECHNICA BIZ & IT TECH SCIENCE POLICY CARS GAMING & CULTURE STORE

TECH —

That time a bot invaded Thingiverse and created weird new 3D objects

Some called it art, some called it spam—and some thought it might be a new life form.

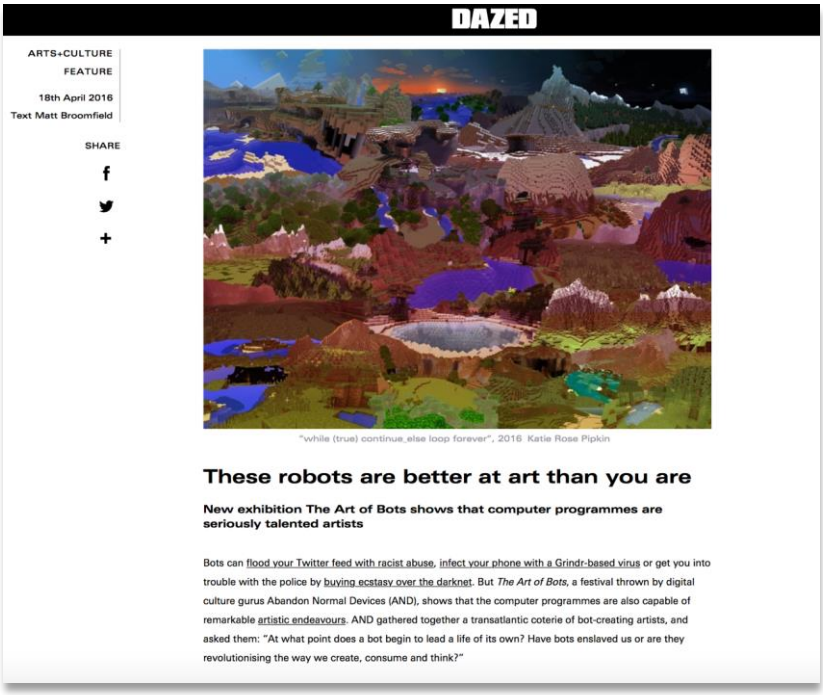
ANNALEE NEWITZ - 5/2/2016, 10:15 PM

Rail and whistle spermatozoon

by shivinteger, published Feb 16, 2016

Dazed Digital

http://www.dazeddigital.com/artsandculture/article/30795/1/these-robots-are-better-at-art-than-you-are


DAZED

ARTS+CULTURE
FEATURE

18th April 2016
Text Matt Broomfield

SHARE

"vwhile (true) continue_else loop forever", 2016 Katie Rose Pipkin

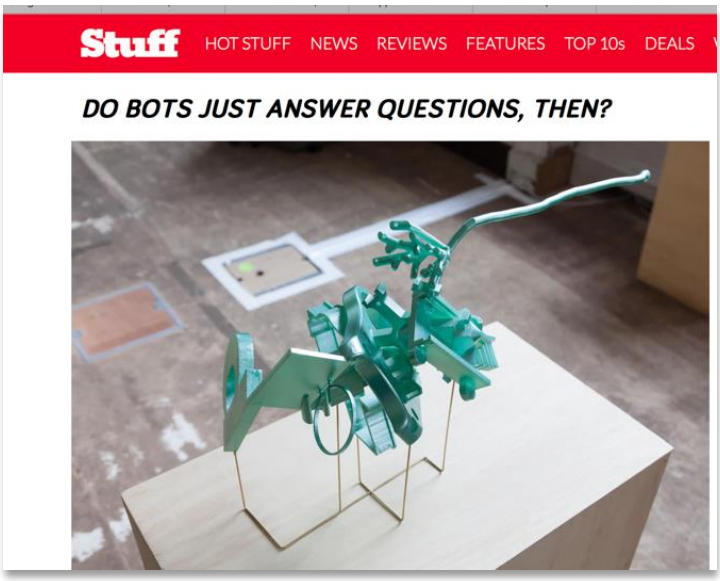
These robots are better at art than you are

New exhibition The Art of Bots shows that computer programmes are seriously talented artists

Bots can flood your Twitter feed with racist abuse, infect your phone with a Grindr-based virus or get you into trouble with the police by buying ecstasy over the darknet. But *The Art of Bots*, a festival thrown by digital culture gurus Abandon Normal Devices (AND), shows that the computer programmes are also capable of remarkable artistic endeavours. AND gathered together a transatlantic coterie of bot-creating artists, and asked them: "At what point does a bot begin to lead a life of its own? Have bots enslaved us or are they revolutionising the way we create, consume and think?"

Stuff

https://www.stuff.tv/news/wtf-are-bots-tech-worlds-chatbot-resurgence-explained?_ga=1.181854695.1920300040.1368176044


Stuff HOT STUFF NEWS REVIEWS FEATURES TOP 10s DEALS

DO BOTS JUST ANSWER QUESTIONS, THEN?

Alpha

https://www.alphr.com/art/1003229/bots-facebook-and-cut-up-poetry-an-interview-with-artist-and-bot-maker-matthew-plummer

Art

Bots, Facebook and cut-up poetry: An interview with artist and bot-maker Matthew Plummer-Fernandez

Artist Matthew Plummer-Fernandez talks to Thomas McMullan about artifice, Tay and the art-bot community

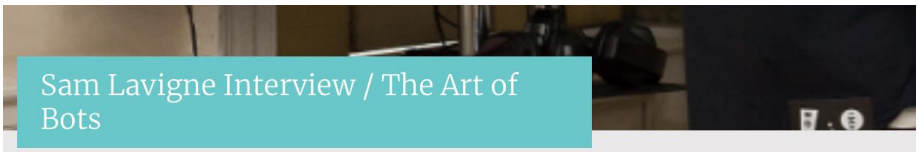
Thomas McMullan
18 Apr 2016

As if out of nowhere, bots are everywhere. Recent weeks have seen the very public meltdown of Microsoft's Twitter AI, Tay – which turned into a genocidal racist in 24 hours – followed by announcements from Microsoft and Facebook that they would be creating bot platforms for businesses. Soon you'll be able to order food or flowers by talking to a bot on

II.V Interviews

Sam Lavigne Interview

https://www.andfestival.org.uk/blog/sam-lavigne-interview/



Sam Lavigne Interview / The Art of Bots \\

Thu 14 Apr 2016

The Art of Bots curator and exhibitor Matthew Plummer-Fernandez spoke to artist, bot-maker and fellow exhibitor Sam Lavigne to find out more about his new project Parliament Live, and the role of video and humour in his work –

Matthew Plummer-Fernandez (MPF): One of the many things I love about your work is that you undermine serious problems (Capitalism, Prisons, Surveillance) through humour. What is the importance of humour to you?

Sam Lavigne (SL): A lot of my work makes use of found material, such as images, videos and text produced by institutions which they then make available online. I don't try to force material to be funny if it isn't, but if I find something that strikes me, I'll present it in a way that highlights whatever latent humour I see.

Frequently humour happens when institutions produce material that undermines their own narratives. For example, I recently discovered that many sheriff's departments in the US have YouTube channels containing b-roll footage of their prisoners. I found **this one** particularly interesting. It's a self-congratulatory piece about a special wing of a prison built to house US veterans. The sheriff's department appear pleased with themselves for taking special care of ex-soldiers, and presumably have created the video for distribution to news outlets. At the risk of explaining the joke, there are two things to note: 1) That they have decorated the prison to be 'America' themed and 2) The fact that they have a large enough veteran prisoner population to build a whole wing to house them.

Chris Rodley Interview

https://www.andfestival.org.uk/blog/introducing-chris-rodley-the-art-of-bots/

Introducing Chris Rodley / The Art of Bots

Introducing Chris Rodley / The Art of Bots \\

Tue 12 Apr 2016

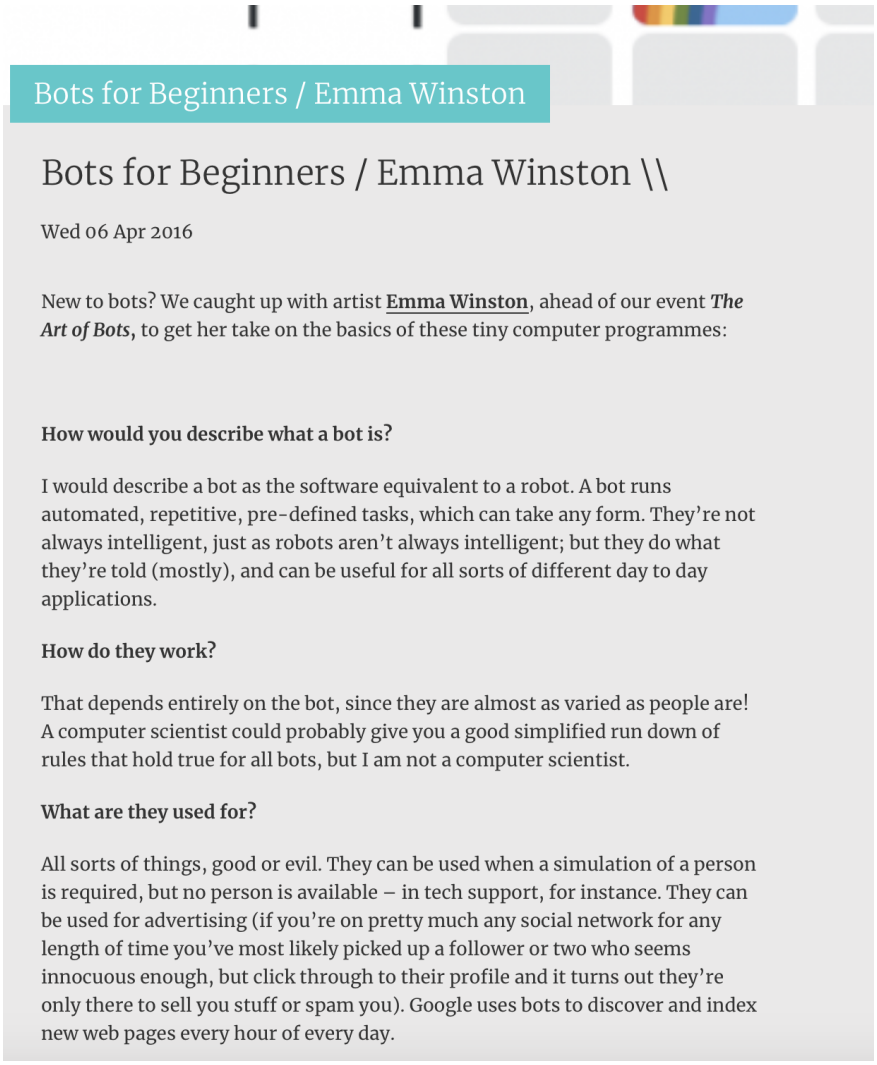
Chris Rodley is one of the creators of ***Magic Realism Bot***, a story-telling Twitter bot, featured in ***The Art of Bots***. We interviewed him to find out more about his work and how real-time data is being used in digital writing...

Tell us about your practice – do you see yourself as an artist, programmer or writer?

All three, in a way, but it's my sister and co-creator Ali who wrote the code for *Magic Realism Bot*. I started out wanting to be a conventional writer, and toiled at this for quite a few years before moving towards work at the intersection of writing and digital technology. This usually means working with code to some extent (although there are some great prefab platforms now like 'Twine' which the non-programmatic are doing fantastic things with). As well as literature and code, digital writing usually also means engaging with an extra-literary dimension – for example, graphic design, images, animation, interactivity, live art, projections, installations. That's where the art comes in. I think we are actually seeing a really interesting convergence between writing, programming and visual art. I'm actually not sure that the words exist yet for this type of practice, which is why myself and other bot artists sometimes use vague self-descriptions like "I make stuff on the Internet".

Emma Winston Interview

https://www.andfestival.org.uk/blog/bots-for-beginners-emma-winston/



Bots for Beginners / Emma Winston

Bots for Beginners / Emma Winston \\

Wed 06 Apr 2016

New to bots? We caught up with artist **Emma Winston**, ahead of our event ***The Art of Bots***, to get her take on the basics of these tiny computer programmes:

How would you describe what a bot is?

I would describe a bot as the software equivalent to a robot. A bot runs automated, repetitive, pre-defined tasks, which can take any form. They're not always intelligent, just as robots aren't always intelligent; but they do what they're told (mostly), and can be useful for all sorts of different day to day applications.

How do they work?

That depends entirely on the bot, since they are almost as varied as people are! A computer scientist could probably give you a good simplified run down of rules that hold true for all bots, but I am not a computer scientist.

What are they used for?

All sorts of things, good or evil. They can be used when a simulation of a person is required, but no person is available – in tech support, for instance. They can be used for advertising (if you're on pretty much any social network for any length of time you've most likely picked up a follower or two who seems innocuous enough, but click through to their profile and it turns out they're only there to sell you stuff or spam you). Google uses bots to discover and index new web pages every hour of every day.