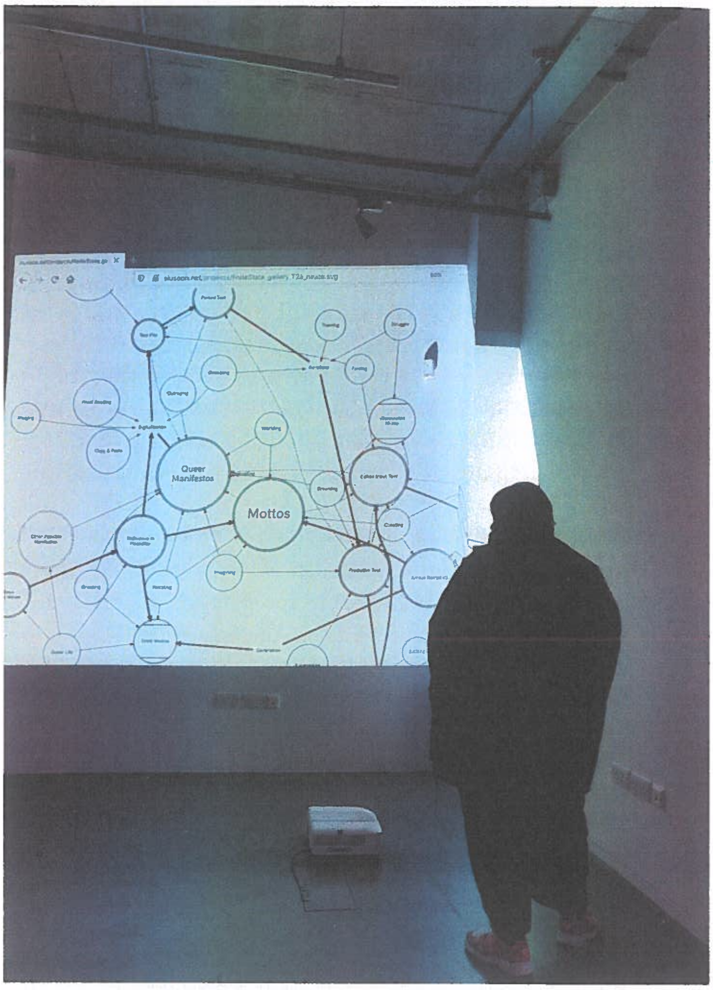
Winnie Soon



5 Recurrent Queer Imaginaries, Exhibition Research Lab, Liverpool John Moores University

Your work is truly transdisciplinary, bridging and mixing programming and computer science, social sciences and digital art. How did you end up here?

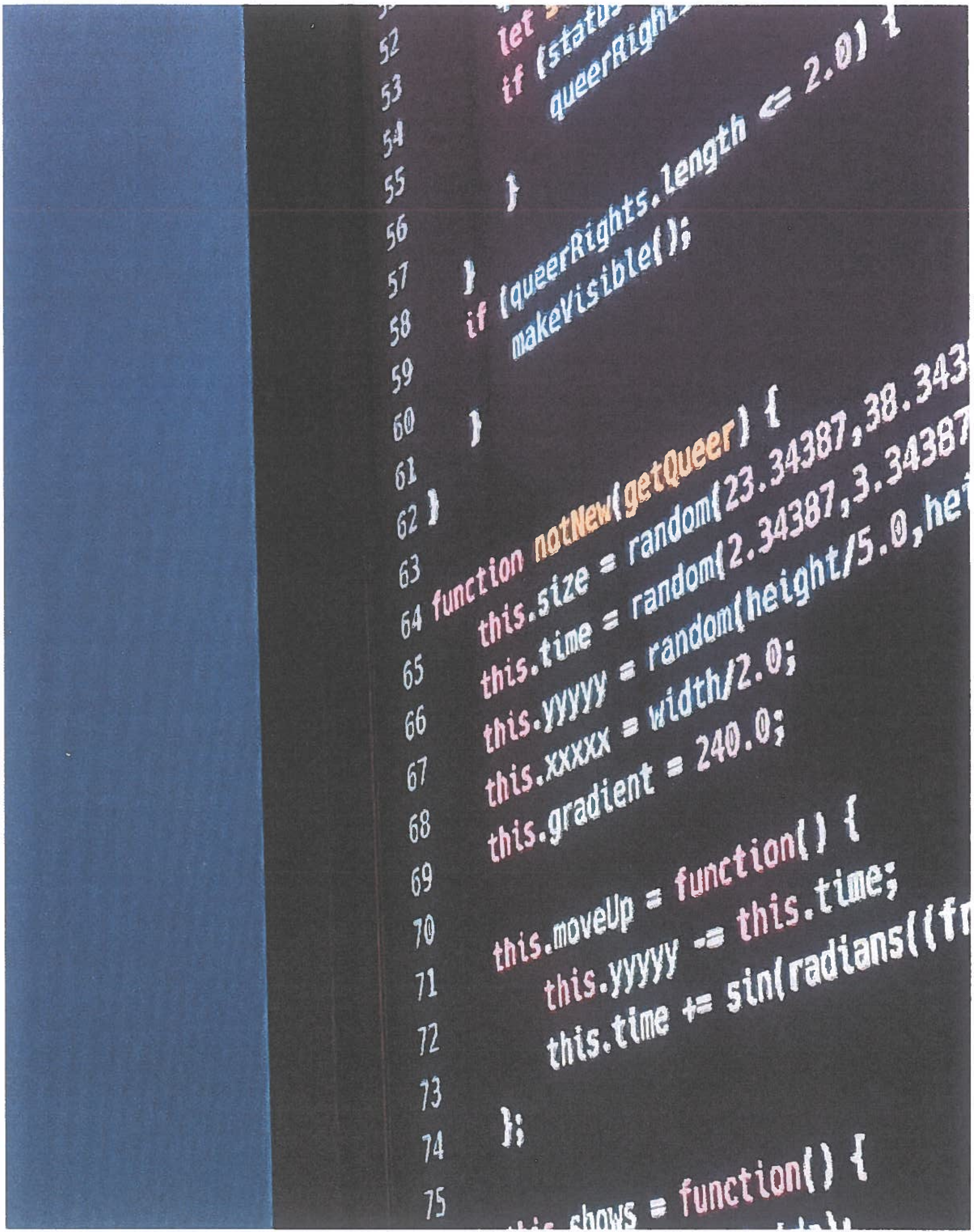
I have been interested in programming and computer science since I was young. I guess I was influenced by my Aunt, Christine Cheung, who raised me in Hong Kong. She bought a computer home in the early 90s, the time with the super slow dial-up Internet and I was exposed to telnet, BBS, LOGO programming, web browsers such as Netscape. What's impressive about her was that she broke apart the computer to fix hardware issues, add ram & a video card, and formatted the computer when things went wrong in the command-line DOS environment by herself. She wasn't highly educated, she simply she loves computers and subscribes to the spirit of do-it-yourself. As such, I developed a different relationship with computers.

Though I have an education background in information systems and computing, I felt I could do more with it creatively. The closest area was web design in the late 90s in Hong Kong. I then went on to study design but quickly found that wasn't quite right still. At that time there wasn't such a thing as 'media art' and it took me a long time to figure out my interest was at the intersection of art and technology in a broad sense.

The turning point was in my masters study in Media Cultures, Digital Art and Technology at the School of Creative Media in City University of Hong Kong and iDAT in University of Plymouth respectively (after years working in the commercial and tech industry). I was exposed to various forms of 'software art' and critical theory, and discovered that there is an area that further connects to cultural studies and politics. The book titled Software Studies: A Lexicon, a collection of essays edited by Matthew Fuller, was especially eye-opening to me, because it opens up ways of thinking on technical materials and terms in humanistic ways. I started to rethink the cultural meaning of technical materials, and what that means. For example, what does it mean for to work on the launch of one of the first Asian surveillance systems on mobile devices in 2014?

It's a pretty long answer but these are some of the key milestones that shape why I end up having a transdisciplinary background.

Unerasable Images presents screenshots from Google Image search for the term "六四" ("64"), a reference to the date of the student-led Tiananmen Square Protest in Beijing in 1989. It explores the circulation of an iconic image known as 'Tank Man' depicting a protestor facing down advancing military. In 2013, a Lego reconstruction



```
        let
        if (statu
            queerRigh

        }

    }
    if (queerRights.length <= 2.0) {
        makeVisible();

    }
}

function notNew(getQueer) {
    this.size = random(23.34387,38.343
    this.time = random(2.34387,3.34387
    this.yyyyy = random(height/5.0,he
    this.xxxxx = width/2.0;
    this.gradient = 240.0;

    this.moveUp = function() {
        this.yyyyy -= this.time;
        this.time += sin(radians((fr

    };

    this.shows = function() {
```

6 Vocable Code, WRO 2019 Biennale

of the image also began to circulate online. Both images are routinely censored and blocked in China but can sometimes be found in circulation elsewhere. In 2017 you used screenshots of the image search to trace the 'geopolitics of data circulation' and Internet censorship. Are you turning the user's attention from the image itself to the infrastructures that support its dispersion and circulation?

Oh yes definitely. With the starting point of an iconic image of the Tankman, this allows me to describe the specific context and history of censorship in China, like voices that are being marginalized, and people have a strong desire to be seen even the image was being censored within 24 hours unsurprisingly. That particular image was actually uploaded to a car channel of a web portal in 2013 as part of the children's day campaign on 31 May. Among many submissions with children's cars and toys, this Lego recreation of a Tankman scene is found and eventually got censored.

But equally interesting is how the image was further picked up by Western search engines, and how the absence/presence, and particular varying positions, of the search, change the way an image is circulated and operationalized. In this way, the project is reminiscent of Hito Steyerl's notion of the 'poor image' and Harun Farocki, Ingrid Hoelzl and Remi Marie's 'operative image'. Here the image is no longer a separate unit from the political infrastructure, and the image has agencies operating through databases, formats, indexing, interface design, and various forms of governance etc.

Unerasable Images reminds me a little of Digital Methods Initiative's Censorship Explorer, an interface that allows users to identify if a particular site is censored in another geographic location using a virtual proxy. What can we learn by looking specifically at the geography of data circulation and at specific geographies of censorship?

I think the daily screenshots allow me to have a different concept of and relationship with the image via repetitive manual actions. I travelled a lot during that year to Germany, Malaysia and Hong Kong for example, and at one point my computer was broken and I needed to use an Internet cafe. All these instances allowed me to ask different questions regarding geopolitics (such as the different default search page/language, my browsing preferences, and the corresponding search results). But more interesting is that by using the same keyword search now, I could no longer find the same image even if I scrolled to the end result of the image display. This makes me think about the constant change of page indexing and tagging, and the powerful and seamless structure of search. I am unsure if my work allows others to 'learn' so to speak, but raising different questions (both political, cultural and technical) via poetic means is definitely one of the aims. Maybe what can be learnt is to try out different methods and observe the results over time to establish different relations with the materials (or subject matters) that would allow different forms of inquiry. Screenshots is one of those methods, and I have co-written articles with other artist Sarah Schorr on this subject matter. [1]

A lot of your work turns our attention towards the interfaces and infrastructures structuring digital visual cultures. Why does this fascinate you?

Wendy Chun argues that software works invisibly and ephemerally, or what she called "visibly invisible or invisibly visible". She describes computers in general as "Opaque yet transparent, incomprehensible yet logical, they reveal that the less we know the more we know (or are shown)". I am very inspired by her thinking and analysis of software, especially interested in understanding how things are rendered in(visible) and for whom and why, and what are the power dynamics in infrastructure (including the interface). Software is very interesting in terms of its specificities, regarding the linguistic and technical aspects in terms of a translation to human readable and machine executable language. This translation involves abstraction via having relations to interfaces and machines that constitute the performativity of code, and hence renders what is (in)visible and (im)perceptible.

My artworks usually address multiple layers of such performativity, from representation/interface to infrastructure, and includes also the entanglement of signs, culture and technology.

Your forthcoming book with Geoff Cox, Aesthetic Programming (forthcoming, Open Humanities Press) explores the politics and aesthetics of programming and algorithmic procedures. While there's a growing body of work on the politics of algorithms and also on methods for exploring and critiquing complex sociotechnical systems like machine learning and software, there's not so much on the aesthetics of these procedures and very little on how aesthetics and the arts can play a role in interrogating these systems. How might artists apply techniques from aesthetics to the study of algorithms?

Taking the inspiration from "software art" or "computational art", the book explores the technical as well as cultural imaginaries of programming from its insides. We also extend the discussion beyond formal logic to its outside, emphasizing the usefulness of artistic practice for opening up more speculative, alternative, and messy imaginaries.

We use the dual elements of "RunMe" and "ReadMe" to cultivate reflection by writing code and writing about code. Creative practice is at the forefront of our approach. We do not take effective and efficient code to be a priority and sometimes twist the expected use of syntax accordingly. By introducing various set and open tasks as well as questions throughout the chapters, artists (or other readers) are encouraged to question the application of their learning in the making of artefacts. It also encourages independent learning by addressing the theme openly instead of looking for a prescribed outcome.

We take a particular interest in power relations that are relatively under-acknowledged in technical subjects, such as those concerning class and capitalism, gender and sexuality, as well as race and the legacies of colonialism. This is not only related to the politics of representation but also nonrepresentation: how power differentials are implicit in code in terms of binary logic, hierarchies, naming of attributes, and how particular worldviews are reinforced and perpetuated through computation. We feel that it is important to further explore the intersections of technical and conceptual aspects of code in order to reflect deeply on the pervasiveness of computational culture and its social and political effects — from the language of human-machine languages to the abstraction of objects, datafication and recent developments in automated machine intelligence, for example.

Artists will benefit from the book by following an applied and overtly practice-based approach to understanding the importance of programming — reading, writing and thinking with software — as a critical tool for our times. We consider programming to be a dynamic cultural practice and phenomenon, a way of thinking and doing in the world, and a means of understanding some of the complex procedures that underwrite and constitute our lived realities, in order to act upon those realities. As such, the book explores the set of relations between writing, coding and thinking to imagine, create and propose alternatives.

The book is open access and open source, and we wrote on Gitlab platform: see http://www.aesthetic-programming.net/ for updates, or the software repository here: https://gitlab.com/aesthetic-programming/book. The book is offered as a computational object open to modification and reversioning. We hope that people will fork the copy and change the content, or even add a chapter that can suit their learning.

The book introduces a reflexive practice called 'aesthetic programming' as a way to understand these systems better and also potentially rethink them. Do you think we should all be coders now? Do we need to be able to read and write code to interrogate its politics?

Your question addresses the notion of 'literacy', and if 'everyone', from ordinary people to researchers, should be a coder. The book titled Coding Literacy, written by Annette Vee, is relevant here to provide insights drawing upon the history of writing/reading literacy, and to rethink what it means by knowing, and how that knowing relates to everyday life, institutions, history and social contexts. From my point of view, there are many ways to engage with computational culture and its politics from a researcher's perspective. For example, Taina Bucher is not a coder but studies the power and politics of algorithms via sociotechnical examination and philosophical inquiry. Nanna Thylstrup studies the Politics of Mass Digitization via case studies. Each method has its own potential and limitations, and allows you to answer certain kinds of questions. The key is not only in the methods, but also in how you approach the methods (this perhaps relates to methodology). Besides, knowing how to read/write code does not mean one could engage with the politics (the analogy would be that even though one knows how to read and write, it does not mean one understands the politics of discourse). Therefore, I think there are multiple ways to interrogate politics in computation, and coding is one of those that we can investigate by zooming in to the formal logic of computation and zooming out on the cultural implications of software. As mentioned earlier, it is a way of thinking and doing in the world, and a means to understand the implications of technology.

In a world where we're increasingly concerned about algorithmic governance and its social, political and economic effects, what are some of the most effective bottom up approaches users can take to maintain agency? Is it learning to code, lobbying for greater transparency and accountability? Is it developing techniques that trouble machine surveillance such as obfuscation or adversarial attacks?

I think first is to question and reflect on the software, protocols and infrastructure that we are using, and to examine if there are other choices and what they are. What are the alternatives in the market and what's the agenda/priorities/nature of different organizations and companies, and their models of producing software/platforms? We can reflect so much on what we need to know about algorithms, what we know and what we don't know and why. This is something everyone can do regardless of having the knowledge of coding or being activists/artists or not. It is a start to think about how we might trouble machine surveillance, why we need to, and form whom we act.

Vocable Code emerged from the prompt 'can software be feminist'. I'm curious to know, what's your answer? And if software can be feminist, how?

If you asked me this a couple of years ago, I wouldn't have had an answer. In recent years, I have been highly influenced by Constant, a non-profit, artist-run organisation based in Brussels since 1997. In particular, I'm drawn to the writing and works of Femke Snelting who develops projects at the intersection of design, free software and feminism. One of her projects, "Feminist Server", reminds us that technology is situated; a piece of software is fundamentally relational and embedded in social structures, practices and bodily relationships. If we think in this way, then software *can* be feminist. Software is always subjected to how it is developed, produced and maintained. The next step is to think about how we could experiment with technological services and artefacts differently beyond the normative approach. For example, p5.js is a free open source software library for creative coding, focusing on diversity and inclusivity, and the community takes these issues seriously in software development and communication. As founder Lauren McCarthy says, "thinking about community outreach and diversity is not a secondary goal of p5.js; it's the foundation on which the platform is built." As p5.js demonstrates, software is not just a tool, but also about people and politics. Artist-researcher Cornelia Sollfrank reminds us that cyber(feminism) wants to query the power structures of materials that shape reality, and to challenge power relations that are embedded in infrastructures (see her edited book titled The Beautiful Warriors). In this way, feminist software is about how one might engage with the politics of technology, people and infrastructure.[2]

Your work is both practice and research. It often seems that your practice is about revealing the hidden algorithmic infrastructures of digital cultures, while your academic research formalises and shares these aesthetic practices as a method that others can use. How do the two coalesce?

Practice and research come together mutually for me. I usually start an artwork which is driven by my interests in invisible infrastructure, digital censorship, data politics and real-time processes, and the process is usually very open-ended without a fixed research question. I explore and reflect along the way through different methods, such as coding, framing and exhibiting artworks, reading technical materials & specifications, making field notes, experimenting/hacking technologies, etc. My artistic practice allows me to explore multiple directions and in open formats to see how things come into relation, and express the poetics of these materials. Research allows me to synthesize thinking and go deeper into a selected aspect to articulate it in a different way. Practice and research go hand-in-hand, and the process of making an artwork may be inspired by certain concepts, and sometimes this will change the direction or the framing of it. On the other hand, I also like something that is concrete in which I can base upon and further articulate my thoughts in academic writing. I enjoy being in both spaces of practice and research that open up different ways of thinking and reflection.

You always make the code visible along with your projects on Github /Gitlab and describe the technical set up. Is this motivated by a desire to open source the work, to share approaches and methods with others, or is the source code part of the artwork?

There are different reasons, but yes the major one is that I subscribe to the values of free and open-source culture, and I have benefited so much from different communities of F/LOSS, and I want to contribute in one way or another (even a small one).

Additionally, since I am also a researcher I am concerned with how knowledge can be studied and built upon, and therefore I consider documentation (or even iterations) of the projects as one of the many ways to build knowledge. I think it is important to build and maintain the archive of my own works to see how things have changed and developed, but they are not necessarily part of the artwork (it depends on which one, and for example, the work Vocable code that you mentioned is something the source code is part of).

Much of your work seems to deal with temporality in different ways, as digital ephemera, digital archiving and permanence, digital time-based processes, macro and micro-temporalities. Why do time-based processes fascinate you?

Yes, I was very interested in real-time processes, or what I called "liveness" in my PhD. Our sensation of liveness is deeply reconfigured by complex technological infrastructure behind ubiquitous screens and interfaces. With advanced processors and technologies, feeds, instructions and logics are executed at imperceptible speed, in which networks of machines and the circulation of data mediate our sense of time. The thinking of images, for example, in the work of Unerasable Images is operationalized by real-time processes. Besides, the attention to micro-temporalities allows me to think of different parts of time, and the micro-decisions that are made in infrastructure, for example in network protocols & data packets transmission, signal processing via the cyclic process of fetching and execution, buffer and buffering in synchronisation, etc. (I have written about this in the article "Throbber: Executing Micro-temporal Streams"). Working with the matter of time and temporality is also a very interesting technique in artistic practice. It allows one to reveal or record the organization of tasks and events and changes over time that render what is visible and invisible. Machine-time operates at a different register to human-time, further complicated by global network infrastructures and notions of real-time computation.

Some of the artists in this issue are interested in machine vision, as in how machines see through machine learning, automatic recognition systems etc. Your practice to my mind is another kind of machine vision, an exploration of how digital resolutions, files and search engines are transforming visual culture. How to get the Mao experience through internet..., for example, questions how the (live) GIF digital format reconfigures the experience of a public space and public figure such as Mao. Abstract Imaging experiments with image making processes, thinking through networked images, abstracted images, social images and computational images. Do we have to rethink aesthetics and particularly ways of seeing in the light of machine vision and computational imagery?

I am interested in computational aesthetics as it is connected to how computation operates at multiple scales, including algorithmic procedures, data processing, and abstracted modelling that are embedded in our digital culture. This is not only about the formalisation of relations, but also as Fuller and Fazi describe it, "forces of all degrees and kinds that participate in these abstractions". Some of the mentioned aesthetic aspects are abstraction and concreteness, universality, discreteness, axiomatics, numbers, limits, speeds, scale, logical equivalence, and memory. Therefore, experimenting with computational images allows us to understand and examine the impact and implications of computation beyond immediate and/or perceptible forms in computational culture.

(see Fuller, Matthew, and M. Beatrice Fazi. "Computational Aesthetics." In A Companion to Digital Art, edited by Christiane Paul. John Wiley & Sons, 2016.)

Has 2020 made you rethink your practice at all?

Not so much actually because a lot of my artworks also exist online, and sometimes in the form of net art. In my artistic practice, the outcomes takes different forms, from video & net art to software, installation and workshops.

Well, maybe also because I was born and raised in Hong Kong, something physical/tangible like space costs a lot. Therefore I am used to working with my computer and within the screen.

But maybe you are also referring to the political climate of 2020.

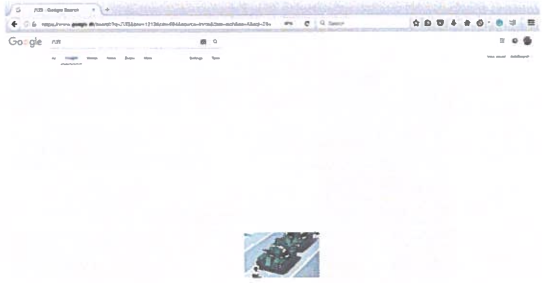
If this is what you mean, then yes to some degree. Especially questioning the privileges that I have now that I reside in Denmark. Hong Kong is in a very difficult situation this past year with many protests and major changes to laws. I have been rethinking the role of the artist, and how I can address fundamental issues such as digital censorship, freedom of speech and digital authoritarianism in a way that resonates with wider global politics.

Anything you're working on at present that you'd like to tell us about?

I am currently working with British artist Helen Pritchard on an API (Application Programming Interface) project as part of the forthcoming Transmediale site. Building upon our earlier project on "Recurrent Queer Manifestos", this project offers a queer API in which a platform/program can connect to a generative motto generator. The mottos are generated via machine learning and other generative algorithms based on historical and recent queer and feminist manifestos in text format. We want to provide an interface and software as feminist services to think through the real-time processes of generation, including asking what it means to maintain feminist services, and asking what are the limits, required conditions and resources in providing such services as a commons. What if services are provided not from the perspectives of optimization and efficiency - how might we engage with technologies otherwise?

BIO Born and raised in Hong Kong, Winnie Soon is an artist-researcher interested in queering the intersections of technical and artistic practices, with works appearing in museums, galleries, festivals, distributed networks, papers and books. Researching in the areas of software studies and computational practices, she/they is currently based in Denmark and working as Associate Professor at Aarhus University.

URL https://siusoon.net/



7 Unerasable Images

References

[1] Soon, W., & Schorr, S (forthcoming). "Screenshooting Impermanence". Geismar, H., Otto, T. and Warner, C. , eds., Introduction to Impermanence: exploring continuous change across cultures, UCL Press: London.

Schorr, S & Soon, W., 2020. "Screenshooting Life Online: Two Artworks". Annette Markam, Katrin Tiidenberg, eds., Metaphors of the Internet: ways of being in the age of ubiquity. Peter Lang Publisher.

[2] in case if you need reference on the quote: Lauren McCarthy, "P5js Diversity & Floss Panel Introduction" (2015). Video available at http://opentranscripts.org/transcript/p5js-diversity-floss-panel-introduction/.