



Painting the Screen with Code: Using CSS Art to uncover the embedded gendered ideologies  
in the technological practices of Coding and Software

Utrecht University  
MA New Media and Digital Culture  
Master Thesis

Nelly Clausen (UU – [n.m.clausen@students.uu.nl](mailto:n.m.clausen@students.uu.nl))  
Student number: 6889883

Supervisor: Dr. Ingrid Hoofd ( UU – [I.M.Hoofd@uu.nl](mailto:I.M.Hoofd@uu.nl) )  
Reader: Dr. Imar de Vries (UU - [I.O.deVries@uu.nl](mailto:I.O.deVries@uu.nl) )

Citation Style: MLA 8th  
Date: 14<sup>th</sup> of June 2020  
Word Count: 15,500

## Table of Contents

<b>ABSTRACT .....</b>	<b>3</b>
<b>ZEROS (0) AND ONES (1) ARE NOT EQUAL .....</b>	<b>4</b>
<b>THE CONNECTION BETWEEN FEMINISM AND CODE.....</b>	<b>10</b>
FEMINISM IN ART AND TECHNOLOGY .....	10
<i>The Female Subject in Art History.....</i>	11
<i>The Female Subject in Technology.....</i>	14
<i>When Worlds Collide.....</i>	16
THE MYTH OF CODE .....	17
<i>In Between Language and Simulation.....</i>	18
<i>Software: The Anti-Fetish .....</i>	22
<b>WHO IS PURECSS LACE? .....</b>	<b>24</b>
SHE IS CODE .....	25
<i>Why and How to look at the Code .....</i>	25
<i>Basics of CSS and HTML Coding.....</i>	26
<i>Analysis: CSS Code .....</i>	28
<i>What is She hiding? .....</i>	29
SHE IS ART .....	31
<i>Analysis: PureCSS Lace in Google Chrome .....</i>	32
<i>She has Power, She is Both .....</i>	33
<b>THE CREATIVE LOOPHOLE .....</b>	<b>37</b>
NOT ONE SIMULATION, BUT MANY SIMULATIONS.....	37
<i>Analysis: Software Simulations .....</i>	39
<i>Where is Her Aesthetic?.....</i>	42
RETHINKING THE DIVIDE.....	45
<i>From Masculine to Feminine .....</i>	45
<i>Female Executability .....</i>	48
<b>THE FUTURE IS FEMALE.....</b>	<b>51</b>
<b>WORKS CITED .....</b>	<b>54</b>

## Abstract

Coding art is a technological artistic practice in which artworks are created through writing and simulating code. The phenomena sheds light on the myth of code being a neutral executer and highlights its masculine gendered language structures. My research looks at the patterns and underlying structure of coding art from a cultural perspective, mapping the position of the female in the technological and artistic realm and creating a new space. The research focuses on the Case study *PureCSS Lace* by Diana Smith. The examination of *PureCSS Lace* dispels the myth of the accuracy of code and showcases a glimpse of creativity and freedom. The analysis consists of two parts: the source code analysis and the formal analysis. The combination of these analyses reveal the intrinsic connection between the code, the art and the software. Through using creative commands, the artist highlights the manipulability and performativity of code. Additionally, the browsers simulation of the code reveals the location of the slippage of control, which takes place between the connection of software and the code. This breaks down the masculine technological notion of command and control and highlights the femininity and poetic aspect of code. Through the coining of the term female executability, a space is manifested that gives power to female creators and artworks and reflects the cultural aspect of this progress and hints to future female developments.

**Key Words:** Coding, Simulation, Software Studies, Art History, Feminist Studies

## Zeros (0) and Ones (1) Are Not Equal

In the 1970s, the company Kodak developed a color card to calibrate lighting, shadows and skin color in photography, in order to standardize coloring during image processing. The image chosen for the color card was of former employee, Shirley, she had brunette hair, pale skin with white clothing and red lipstick. "She was the standard," Garcia says, "so whenever we printed anything, we had to pull Shirley in. If Shirley looked good, everything else was OK. If Shirley didn't look so hot that day, we had to tweak something — something was wrong" (Barco). This prompted not only gender issues and discrimination, but also a matter of beauty determination and the power dynamic of females in the technological sphere. Men took control over the ideal representation of the female and what it was supposed to represent and claimed power over this. This marks the original female positioning within the technological realm and can be found across numerous examples.

To understand the role of the female in technology, we need to look at what technology actually means. The word technology is derived from the word *techne*. The word *techne* itself, is derived from the greek word τέχνη, originally standing for craftsmanship and art (Parry). Therefore, technology is considered to be a fusion of artistic craft. Hence, technology and the practice of art are closely intertwined. They are both forms of expression following certain structures and rules, they have a goal and message they seek to portray. Furthermore, an aspect that both fields have in common is that the female is both underrepresented and misrepresented. In technology practices relating coding and software have always been very male dominated, from the workforce to representatives to institutions and the overall structure (Wajcman, "From Women And Technology To Gendered Technoscience" 288). Not only in technology, but also in art, the role of the female is suppressed, as many iconic and world famous artists are elderly white

men. In Art History, the female is often only the muse, the object of inspiration, the subject of the painting. Hardly ever is the female seen as the creator, the painter, the one with the power. Both fields are male dominated and give little to no space for female expression. By looking at the current role of the female in art and technology, we can understand these processes and begin to find a way to regain female space.

Looking at technology from a cultural perspective, we can uncover embedded ideologies that are not plainly visible to the eye. This research focuses on unravelling the embedded ideologies in code, doing so by examining an artwork created entirely by code in order to dispel the myth of code accuracy and instead showcase a glimpse of creativity and freedom. When code is displayed it is via a software, which interacts with each line of code in a unique way. This means that each reading of the code is individually calculated and therefore the outcomes and display may not result as expected. This generates a creative artistic notion to technology. A notion that can free the female from suppression and allow for expressiveness. This shift and movement can be understood by looking at coding art, in particular CSS Art.

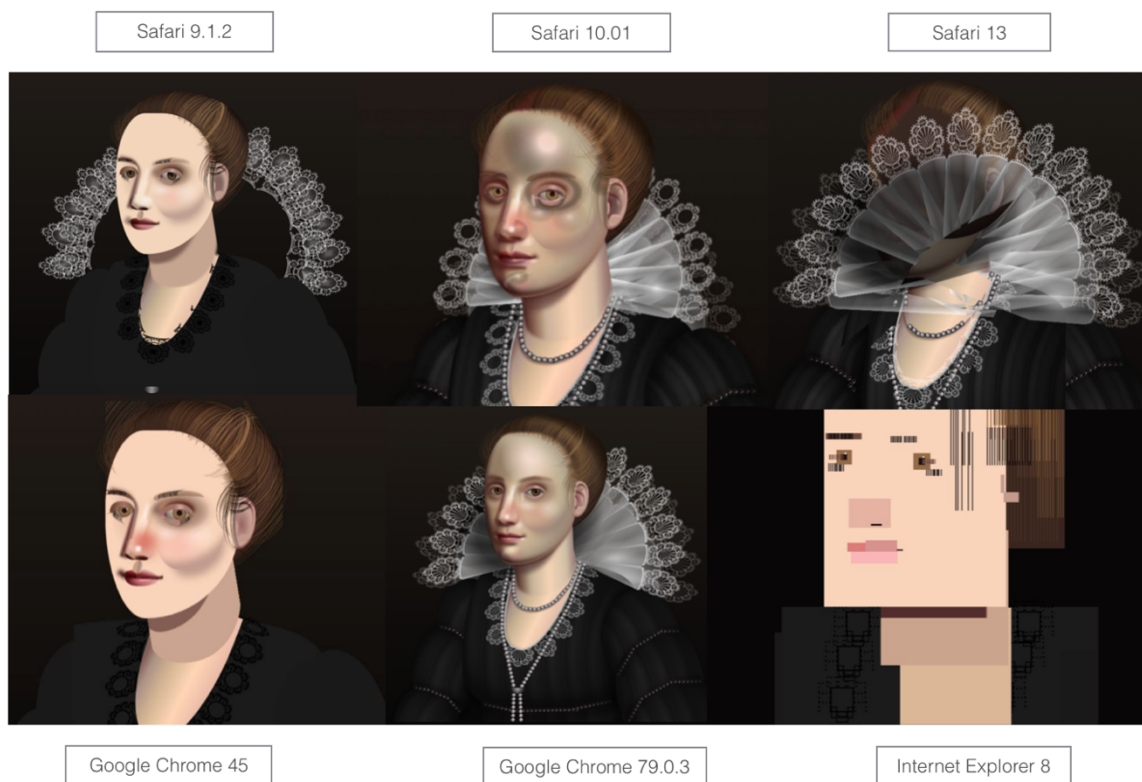


Figure 1: *PureCSS Lace* by Diana Smith simulated in different Browsers

CSS Art is an artistic practice done by coding HTML and CSS code. It is often open source and accessible online for its viewers. The artwork does not have an original image as the code is simulated by the software and reflects the compatibility of both. Many of these artworks can be found online, or on webpages that are galleries for these works. One of the most popular CSS artists is Diana Smith, a User Interface Engineer who handwrites all her code. Her work is open source and all the code is available on her GitHub (<https://github.com/cyanharlow>). This allows anyone who is able to read the code to access it and understand it. This research paper will focus on the work *PureCSS Lace* by Diana Smith. The artwork consists of HTML and CSS code, it has 1,287 lines of HTML code and 1,965 lines of CSS code. These two languages are combined to create an artwork designed to be displayed in Google Chrome. The simulation can also be viewed in different browsers, but it will not appear in the same way, raising interesting questions on manipulability and expression. Figure 1 shows CSS Lace being simulated in many different ways across six browsers. By analyzing the artwork, I aim to understand the power of simulation of data and how art can reveal hidden information regarding software and gender, normally not seen if viewed with the naked eye.

My research seeks to question the interplay of art and code and its resulting simulation by looking at it from the perspective of the female. The connection between artistic value and technological qualities of the work lies in the manipulability of the code and its corresponding connection to simulations via software. By looking at the artwork, its underlying code and its various simulations, I want to find the significance of their connection and how this affects the understanding of the field. Therefore, I ask: How is Diana Smith's Female Body in CSS Art an example of a feminist critique on the manipulability of simulations? How do the aesthetic choices and outcomes reflect the gender dynamics encoded in coding languages?



Figure 2: Tweet by Diana Smith on her artwork's intention

Intentionality plays a significant role when researching this area. In the world of art, the intention of the artist is taken very seriously and is often considered to be the answer to all analytical questions. With coding, many tasks are trial and error and sometimes coders cannot control why something resulted in the way it did. Therefore intention can be considered, but does not tell the ultimate truth. For this artwork the creator Diana Smith said “because of the artistic nature of this project I have not concerned myself with cross-browser-compatibility, so the live preview will most likely look laughable in anything other than chrome” (Cyanharlow). Therefore, she states it was not her original intention to create different versions, but she is fascinated by the effects and questions it raises (refer to figure 2). These coded artworks, that were intended for beauty, have become works that breakdown the role of coding in art and how coding language can manipulate software and its embedded ideologies, while simultaneously prompting further thought on what femininity is. This academic claim will be explored throughout my thesis, elaborating on the notion that despite the lack of intention the work raises interesting considerations.

During my research I will use several different methodologies for this analysis. First, I will explain the theoretical framework of both the artistic and technological realm using

discourse analysis, in order to prepare a Case study analysis for Chapter 2. This Case study analysis will consist of two parts, the first is analyzing the code, and the second is analyzing the artwork itself. I will look at the code in detail to understand its function and expression, this is called a source code analysis and is a technical art historic method. Through this method the code will be analyzed to reveal the hidden ideologies created through the artistic. The second part of the Case study analysis will consist of looking at the artwork from an Art History perspective and conducting a formal analysis. By looking at the artistic qualities I can understand the meaning and power of the image. This will then lead to linking technology (through the code) and Art History (through the artwork) together. Lastly, I will compare the outcomes and discuss the relevance of software studies and simulations, as well as rethinking the masculine and feminine divide in art and technology and conclude with an idea on creating a space for the female.

This cross-combination of methodologies and analysis brings relevance to my project as it combines fields in new ways. The social relevance is established by combining the artistic outlet with technical practices with understanding, while also further critiquing the theories academically. The academic relevance of this projects lies in critically examining the gendered structures encoded into the simulations we view. By extrapolating how images can be constructed and through what means they are displayed in their respective contexts, we can gain further understanding of the role of the female in art and technology. The main research questions leads to my sub questions:

- How can aesthetic artistic practices such as form, shade and texture help understand and deconstruct code in a way that expose masculine ideologies about the embedded data?
- How does the CSS and HTML code, in its structure, function and expression, reflect notions about language and masculinity?
- What aspects of software functionalities of browsers does CSS art uncover? By deconstructing the simulations, how do those different simulations create meaningful



statements through their data practices on implications of technological differences on the role of the female?

- How do the different browser simulations of the CSS art effect the understanding of the female in technology and art?

By answering these questions, I hope to demonstrate the objectives of my research which include: revealing the manipulability of code, highlighting the connection between code and software, questioning the instrumentality of technology and its command and control aspect, showing the masculine and feminine divide in art and technology, and creating a space for female executability.

## The Connection between Feminism and Code

### Feminism in Art and Technology

What role does feminism play in art and technology? In order to discuss the interplay between art and code, and the importance feminism plays in it, we need to understand what the notions of femininity entail and how the feminine subject is produced, as well as explain the female background and where the current situation has arisen from. This will be done from a Western centric perspective, as the Case study also follows this trajectory. I claim that in both the artistic and technological realm, the female subject is impeded through the underlying masculine gender dynamics of the artistic and technological spheres, and by rethinking those traditional structures the female positioning can be shifted. To tackle this reconstruction, we need to first analyze the basic dynamics within the feminist realm.

In order to situate the female subject, we must examine what it means to be feminine and how it correlates in the landscape of feminist theory. In her book *Gender Trouble* (1990), Judith Butler explains and elaborates on different discourses within the feminist field and aims to understand the role of the female as a subject. For the female to be represented, they have to be politically visible (Butler 3), according to Foucault power produces subjects, and this power stems from politics (Butler 4). For the female to become the subject it has to reflect power and be politically visible. This defeats the overall point of emancipation and equality between genders, as it negatively frames the divide, thus creating a destructive critique as opposed to constructive (Butler 5). This leads to the question; how can we constructively rethink the notion of femininity without being destructive? Taking a more contemporary approach to the claims by notable feminists Beauvoir, Irigaray and Butler, the feminists Evelien Geerts and Iris van der Tuin have a more radical approach to the female canon in order to understand the female in male thinking. The female as a subject can be recognized for its transcendence and individual otherness (Geerts). This recognition positions the female subject into a field, where for there to

be one, there needs to be another. This means for there to be female, there needs to be another comparable notion. In order for one, there needs to be something to compare it to, a divide of opposites, an original and a copy. This depicts the gender divide in male and female as the male comes before the female. By understanding the divide and rethinking the female role, we can now try to unveil the position of the female subject in the sphere of art and technology.

### The Female Subject in Art History

Overall, the female subject has been underrepresented in Art History. While the female body and its essence is often the subject of many paintings, in addition to being the muse of artists, the female itself has rarely been recognized as the artist of well-known works. As times have changed, females have started to manifest their position in the art world outside of just the subject, yet this rethinking and repositioning is only gradual. In her two books *Feminism and Art History- Questioning the Litany* (1982) and *Expanding the Discourse* (1992), Norma Broude touches on this topic by explaining the field of Art History from a feminist perspective and how the female as a creator has been represented and the overall need for adjustment and restoration. Through creating a new reading, she emphasizes a reconstruction of the female position in both culture and society, highlighting the way the history of art is portrayed is from a male centered perspective in a male-dominated story. Art history “as a discipline has had a point of view, which involves choices and exclusions” (Broude, “Feminism and Art History” 27). When examining it from a feminist perspective, the discipline is still determined by the underlying values and assumptions made in its masculine narrative.

The role of the female as a subject travels throughout history. The female subject first appears in Art History as goddesses in Egyptian art culture (Broude, “Feminism and Art History” 17) before moving towards the mother figure in Ancient Greece and the Roman working woman (Broude, “Feminism and Art History” 20). From there, the woman is

established to be situated at home and nurturing the family. In regards to Renaissance art, the female, subject to many works, has an interesting position. “What can it have been like to live as a woman in Renaissance Florence? Few if any artistic images offer female self-representation; the great majority of images of women present them according to an elaborate set of social codes that governed their expected behavior as wives and mothers” (Broude, “The Expanding Discourse“ 30). The female was often represented in a relationship, yet individual portraits of women do exist. Depictions of a subject are often presented with power and influence; this goes for male and female subjects. To create this representative feeling of power and influence “portraitists have tended to favor poses that put their subjects into some sort of partial profile, breaking up the stark symmetry of a frontal gaze by angling the face and thus preventing the portrait subject from staring too glaringly out of the canvas” (West 72). This style of painting was well-known at the time to express the most majesty and influence.

When looking at this representation, both the gender of the sitter and artist have to be taken in consideration in order to properly understand the implications of the work. A female artist in the Renaissance time is most likely to be a portraitist than any other type of artist. This craft was able to be produced at home without breaching the social norms of the domestic women, and this art form was considered to be not as creative as others as it involves copying a pre-existing object (West 145). Over all, the female was considered to live and work at home, yet some females in the higher classes managed to obtain power. For a Renaissance “woman reigneth above man, she hath obtained it by treason and conspiracy committed against God.... [Men] must study to repress her inordinate pride and tyranny to the uttermost of their power” (King 160). In order for her to gain power to become more than the wife or mistress of a man, she would have to take extra actions that were seldom successful. In the Renaissance times, women were able to exercise “the power of patronage. Women who did not rule or direct with their armies the forces of destruction could wield their authority and wealth to shape thought and culture” (King 160). This is the power that can be seen in portraits showing female subjects,

as those were often commissioned, and therefore, expensive and show the relevance and power of these strong personalities. Many of these women are yet still linked to influential men of the time.

The way the female was represented overall was very normative and constructed by the hegemonic structures of society. There are some cases where this does not apply, for example in the art of Vermeer. Vermeer offers “alternative approaches as fragmentary perception, a non-possessive relation between artist/observer and female subject, and the pictorial presentation of monumental female figures, who in their exclusive attention to their own affairs are supremely ‘self-possessed’” (Broude, “Feminism and Art History” 28). Even when painted by a male artist, the female subject is represented in a powerful way that allows a shift in traditional thinking. This type of painting encouraged the female to be represented in a different way. “In ages when many women’s positions in society were constrained, such portraits allowed them to break out of their conventional role and assume the guises of mythology or allegorical figures, or to be shown in ways that might be construed as playful or transgressive” (West 158).

More contemporary artists have also tried to represent gender inequalities and fight the divide, such as German expressionist Käthe Kollwitz, who fought against misogyny in art and supported political activism, or the tradition of quilting which is an American craftsmanship common amongst women (Broude, “Feminism and Art History” 36) as well as the creation of lace, which is historically known to be fabricated by females. Through that we have now seen, the females’ position is mainly disadvantaged in Art History, and there is a lot of potential to claim or reclaim its subject position, as reflected in the Case study.

## The Female Subject in Technology

The field of technology shows a similar narrative. Although in contemporary times, more females have entered this work space, in the past this was not the case. Rosemarie Buikema and Iris van der Tuin, in their book *Doing Gender in Media, Art and Culture* (2018), examine the technological sphere through media objects and introduce the gendered computer culture. According to their reasoning we need to “reconsider the sweeping generalizations about women and technology and reconsider the social-cultural history of computer culture from a gender sensitive approach” (Buikema 194). There is still a lack of employment and consideration in regards to females in the Western technological sphere. “Especially in informatics and computer engineering, the numbers are alarmingly low. Women make up 56 per cent of the total US workforce, yet they occupy only 27 per cent of ICT related jobs (NCWIT, 2007). In Europe the figures are different, ranging from high – 50 per cent female ICT workers in Italy and Spain – to extremely low – Belgium 16 per cent, Germany 14 per cent, and the Netherlands 11 per cent (Luijt, 2003; Collet, 2005)” (Buikema 197). This begs the question, why are females so excluded from this field?

Historically, technology as well as ICT has not been a work place for females. Judy Wajcman’s peer reviewed article “From Women And Technology To Gendered Technoscience” published in 2007, focuses on the role of females in Western ICTs and the general discussion on feminism and technology, in the past, present and future. This example refers to the Western standards and cultures, as this might differ in other parts of the world. Since the 1990s, feminists have gained hope in changing and reconstructing the sphere into a more liberal and inclusive space (Wajcman, “From Women And Technology To Gendered Technoscience” 287). The lack of females in the field occurred due to the lack of education available to females. Therefore, feminists were satisfied with the solution of allowing more females to enter the field of science, as a sign of equality. This was still problematic as women

were being asked to exchange major aspects of their gender identity for a more masculine version without prescribing a similar 'degendering' process for men in order to work in scientific practices (Wajcman, "From Women And Technology To Gendered Technoscience" 289). This was also due to the fact that technologies were and still are embedded with masculinity. "Technologies have a masculine image, not only because they are dominated by men but because they incorporate symbols, metaphors and values that have masculine connotations" (Wajcman, "From Women And Technology To Gendered Technoscience" 289). This will be elaborated on in the second half of this chapter, where these notions are applied to coding and the construction of its language.

To become a member of the scientific realm, which includes the technological sphere, women had to give away part of their femininity. Technology has embedded gender so deeply that it has embedded the notion of male control over women (Wajcman, "From Women And Technology To Gendered Technoscience" 289). Feminism tried to shift these notions by promoting technology for females, such as In vitro fertilization (IVF) (Wajcman, "From Women And Technology To Gendered Technoscience" 290). "The Internet and cyberspace are seen as feminine media, providing the technological basis for a new form of society that is potentially liberating for women" (Wajcman, "From Women And Technology To Gendered Technoscience" 291), which encourages and creates a space for female empowerment and a space for rethinking traditional practices. Through these liberations, the academia in the field of Science and Technology Studies (STS) has also shown how feminism and technology are growing to become mutually constitutive. Technology and gender interplay and "gender relations can be thought of as materialized in technology, and gendered identities and discourses as produced simultaneously with technologies" (Wajcman, "From Women And Technology To Gendered Technoscience" 293). There is still a gendering taking place as "several empirical studies have demonstrated that the marginalization of women from the technological community has a profound influence on the design, technical content and use of artefacts"

(Wajcman, "From Women And Technology To Gendered Technoscience" 293). "Feminist theorists are now much more aware of both the contradictory effects of ICTs on women and the different meanings the same artefact might have for different groups of women" (Wajcman, "From Women And Technology To Gendered Technoscience" 294) as there is an optimistic future for females and their identity, but not everyone has the same chance and there will always be an outside who will be othered.

Wajcman's article "Reflections on Gender and Technology Studies" furthers this thinking by telling stories and critically analysing the stories of women in the history of technology who have been involved in technology and have previously been forgotten. She also talks about female contributions in tech journals and the alarming amount of published works in the field by females (Wajcman, "Reflections on Gender" 448). In the recent years, this has improved and she claims that women have "long denied the opportunity to conquer outer space, in cyberspace women can at least nourish the dream of a world free from gender hierarchies" (Wajcman, "Reflections on Gender" 460). Therefore, nowadays the sphere of technology allows more space for the female through for example the "Feminist Data Manifest-NO." This is a non-academic work in which a group of feminists make 10 claims in order to make data regimes less harmful towards females (Cifor).

### When Worlds Collide

As seen through the examples and the thoughts expressed by thinkers in the field in both sections above, art and technology have positioned the female subject in a vulnerable and misplaced position. In both fields the female is perceived as weak as well as incapable of functioning in the field. It is mainly suppressed by the male gender and the systems put into place in the sphere, yet there is a potential for this to shift. In the technological sphere the female is present but not recognized, while in Art History she is mainly absent from creating and only



present as a representation. Furthermore, in the technological sphere the positioning of the female is shifting with a greater potentiality for the female to gain power and recognition, while the norms in the art sector are rigid. In both the artistic and technological realm, the female is suppressed through the underlying masculine gender dynamics of the spheres, and by rethinking those traditional structures the females positioning can be shifted. Through understanding the female in the field of Art History and technology, we have been brought closer to restructuring its current position. By analyzing the Case study, the position will be made clear as well as demonstrate the potentiality in between the spheres and its movement. Where is the female positioned now? How can we change this and what is necessary to make this feminist claim? How is this a socio-political statement that is constructive and not destructive?

### The Myth of Code

One way of understanding feminism as a constructive socio-political statement is by looking at the aspect of language and the gendering of it. In this specific case the language we will look at code as a language. Code is considered perfectly executable language and therefore said to be a neutral executer. But just like normal language, code can be gendered and masculine and therefore I claim: the masculine structures embedded in coding can reflect the position of the female in technology.

Through questioning the construction of the code this myth of the neutrality of code can be debunked. To understand this idea, I want to look at fundamental techno-deterministic notions that apply to concepts in the field. In their work “Critical Questions for Big Data” by danah boyd and Kate Crawford, realized in 2012, the two feminists critique the main aspects of Big Data, ask key questions and debunk the myth of the validity of data. They do this under the lens of Critical Data Studies. Big Data can be understood through three different aspects:

- 1.) Technology: using data to benefit computational purposes
- 2.) Analysis: looking at patterns to make socio-cultural claims
- 3.) Mythology: the belief that data sets can reveal hidden truths that otherwise cannot be seen (boyd 663).

Data, therefore, is said to be all knowing and able to reveal truths that are otherwise hidden. However, this is not necessarily the case, as data is always situated in a context in order to have meaning, its knowledge production is through its combination with other data (boyd 671). The objectivity and accuracy of data alone is therefore a myth. As Big Data relies on coding, then if Big Data is a myth, it follows that so are its counterparts and components. Code is seen as a neutral generator and as always perfectly executable, yet this is also not always the case. Code alone does not produce anything, as it is a language that needs to be processed to then create knowledge. This implementation of the language causes subjectivity and hence can be manipulated, in this case through software and their interpretation of it. Furthermore, the process of reading can malfunction, and lead to unpredictable outcomes at times. This therefore states that coding, and its perfect neutral executability, is also a myth and a fantasy and that its language is gendered. The next section highlights this phenomenon.

### In Between Language and Simulation

The fantasy is that code will always produce the same outcome and that it reads the same and creates the same simulation. This myth is debunked. This is not the case, as the code is translated through software, and it is the software that determines what it will simulate. The mythical fantasy slips away and through the loss of control and we can see the actual structures of knowledge production embedded in the system around coding. Every situation is context specific and due to the high amount of variability, the predictability of the process escapes. This slippage of control reflects both the masculine power and the loss of it, with the role of the

female and its power peaking through. Federica Frabetti covers in her book *Software Theory: A Cultural and Philosophical Study* from 2015, the connection between technology, instrumentality and language as well debunking its knowledge system. Her three main arguments are: 1.) Computer culture is filled with masculine connotations, 2.) Cybernetics are powered by the idea of command and control, but code does not follow this thinking due to slippages in the process, and 3.) the overall role of logos and logic in this debate and its implications.

In order to discuss how exactly how the role of code as a language, we need to understand the connection between technology and language. Frabetti's thoughts follow Derrida's deconstructive ideas on technology and technicity. "It is rather a matter of recognizing the mutually constitutive implications of technology and language—via the concept of instrumentality— and therefore of radically rethinking both terms together, since there is no way of (re)thinking one without the other" (Frabetti 9). By looking at the combination and intertwinement of the two terms, we can understand their relation. "On the one hand, he [Derrida] recognizes the foundational (originary) character of technology for the human, while on the other hand, he reveals technology's complicity with metaphysics" (Frabetti 33) allowing Frabetti to question the connection between technology and technicity through the concept of language. By deconstructing the idealization of technology Frabetti asks, through Derrida, to rethink the relationship between technology, software and the human (Frabetti 34).

According to Derrida "today's technology is 'in deconstruction' because, with its capacity to generate unforeseen consequences, it challenges received notions of invention and instrumentality, and in so doing it challenges the (Aristotelian) notion of 'tool' within which technology itself has been constricted for centuries" (Frabetti 36).

Therefore, technology is not a tool of predicted outcomes, but it can generate unforeseen outcomes and convert the notion of instrumentality. How can technology be an instrument and what role does language have in this? "In what way does software both 'open and prevent the

realization of instrumentality“(Frabetti 37)? “To what extent a deconstruction of software involves a deconstruction of its relationship with writing and language?” (Frabetti 37)

For that process Frabetti is inspired by the thinking of Katherine Hayles and her book *My Mother Was a Computer*. Hayles looks at the connection between technology and language by looking at the role of code in software and how language and writing reference in each in the process (Frabetti 37).

“For Hayles, code is ‘the language in which computation is carried out’—whereby ‘computation’ is defined as, a process that starts with a parsimonious set of elements and a relatively small set of logical operations’, which, instantiated into some kind of material substrate (such as a computer), can build up in creasing levels of complexity, ‘eventually arriving at complexity so deep, multilayered, and extensive as to simulate the most complex phenomena on earth, from turbulent flow and multiagent social systems to reasoning processes one might legitimately call thinking’” (Frabetti 38).

Therefore, the code is the underlying system of thinking in computation; it is the source of knowledge production. Furthermore, Hayles focuses on the difference between natural language and programming and code as a language. On one hand coding is a niche language, while on the other hand the world is fundamentally computational, which could make code the nature of language (Frabetti 38). This division is difficult to decipher as limited research has been done on outcomes of code as a language, in comparison to yearlong research on natural languages. Yet “‘Language alone’, Hayles writes, ‘is no longer the distinctive characteristic of technologically developed societies; rather, it is language plus code’” (Frabetti 39), showing the increasing relevance of accepting coding as a language.

“Programming languages operating at still higher levels translate the lower levels of signification into commands that more closely resemble natural language. Hayles explains that ,[t]he translation from binary code into high-level languages, and from high-level languages back into binary code, must happen every time commands are compiled

or interpreted, for voltages and the bit stream formed from them are all the machine can understand“ (Frabetti 43).

The translation and comprehension process of code into natural language is of high importance and has a high cultural and social value, yet it is highly complex, and therefore, prone to misunderstanding or misinterpretation.

Through this process, code has a unique performativity to it. “Code that runs on a machine is performative in a much stronger sense than that attributed to language” (Frabetti 45). The machine and the relation to the code works under different rules than natural language does.

“Code running in a digital computer causes changes in machine behaviour and, through networked ports and other interfaces, may initiate other changes, all implemented through transmission and execution of code. Although code originates with human writers and readers, once entered into the machine it has as its primary reader the machine itself. Before any screen display accessible to humans can be generated, the machine must first read the code and use its instruction to write messages humans can read. Regardless of what humans think of a piece of code, the machine is the final arbiter of whether the code is intelligible. If the machine cannot read the code or if the program does not work properly, then the code must be changed and corrected before the machine can make things happen” (Frabetti 45).

In this relation, reading and writing differ majorly, and there is a unique slippage happening due to this performative value. Frabetti looks at Butler through Hayles rethinking the performativity in terms of gender performativity. While Hayles sees the symbolic value in the performativity of code in relation to gender performativity, as to her gender is determined by its symbolic values. These are also determined by the language used around and in the performance. “As Butler herself remarks in *Excitable Speech*, language can produce very powerful effects on the materiality of the human body—effects that Hayles would without doubt consider quite, straightforward” (Frabetti 46). The materiality of the body and of the

subject is therefore affected by the language. The way code is performative, is also the way gender is performative. Therefore, this relation reflects the way masculine and feminine power structures are represented in the technological sphere.

It is through this, that Frabetti is able to show the instrumentality of the knowledge production system that is code. “One has to remember that technology can always generate consequences that escape predictability” (Frabetti 169) as the accuracy of production through technology is a myth due to its many involved components and its unpredictability of error and dynamic change, in this case, in regard to code and software. “This active problematization of software ultimately clarifies the significance of software—always to be thought in its singularity—for our understanding of the human and of its constitutive relationship with technology” (Frabetti 170). Software reveals these slippages in the code and can show the values of the production system. By looking into the construction of software theory, we can highlight the point of location of code between language and simulation.

### Software: The Anti-Fetish

Furthermore, this affordance of manipulability of the code can be seen through the simulations of it, as demonstrated by softwares which as mentioned earlier. In her book *Programmed Visions: Software and Memory*, from 2011, Wendy Chun dives into the world of software, code and its implications on how software creates discourse. Software “emerged as a thing—as an iterable textual program—through a process of commercialization and commodification that has made code logos: code as source, code as true representation of action, indeed, code as conflated with, and substituting for, action (Chun 19). Software changed the whole discourse on code. Software is a set of instructions for the computer to fulfill a task (Chun 21). Software participates in creating knowledge power as it converts the original code into a simulation that can be viewed (Chun 21).

Software and its simulations create this mystery around the concept of code. “Code, however, is a medium in the full sense of the word. As a medium, it channels the ghost that we imagine runs the machine - that we see as we don’t see - when we gaze at our screen’s ghostly images” (Chun 50). The code is a fetish that created wonders. Code is a language, but in order for it to have meaning and be powerful it needs to be translated by software (Chun 22). Software is the program that puts together and simulates the source code, and therefore, this pure code an object of fetish (Chun 19) as it produces visions and dreams of pleasure (Chun 20). The implications of software and code will be furthered in Chapter 3, which will explain how the Case study and artwork gains new meanings through the power of the simulation of software and the different outcomes.

By understanding this we can start to question the fundamental issues. These issues are raised by analyzing the Case study introduced in the introduction: *PureCSS Lace* by Diana Smith. This artwork is analyzed from two different perspectives and through the combination of both analyses, the technical and art historic, I hope to reflect the thinking of the theoretical background of this chapter onto a real-world example and demonstrate the possibility of shifting the understanding of the female.

Who is *PureCSS Lace*?



Figure 3: *PureCSS Lace* by Diana Smith, viewed in Chrome



## She is Code

In order to situate the Case study into the academic debate from Chapter 1, we need to look at it from a closer angle. The analysis implemented will explore the structure of the coding language as well as understanding the specific Case study. The Case study was designed to be viewed in the browser Google Chrome ( see Figure 3), so for this chapter the analysis will focus on the image as it was meant to be viewed. The first section will focus on the code and coding language. This prompts: How does the CSS and HTML code, in its structure, function and expression, reflect notions about language and phallogocentrism? For this we have to understand HTML and CSS and their differences. Generally, HTML code structures the page and its set up, while the CSS code gives provides the styling. Therefore, I will focus on the CSS code for this analysis in order to analyze the artistic elements. The CSS code constructed by Diana Smith for this work goes beyond the normal functionalities of language through the way she tricks the commands and layers them. With this technique she creates a new, female centered narrative, through its playful and poetic nature. Using this technique, she plays with the manipulability of code and uses its performativity to construct her work, just as gender is constructed through performativity, as stated by Butler and Frabetti.

## Why and How to look at the Code

In order to look at the code through the lens of Art History, I will use the method of Deena Engel and “Source Code Analysis As Technical Art History”. “Source Code Analysis As Technical Art History” explains the methods that can be used to better understand source code. “Technical research on artist-generated source code not only serves conservation, but it can also aid art-historical research on artists’ aesthetic aims and their working methods” (Engel 91). Source code can be used as a documentation in order to give extra information and clarification on the objective, as well as the outcome of how it is supposed to look. This can be done with

these methods: a) annotation of the code, b) narrative description (Engel 92) c) visual documentation, and d) UML diagrams (Engel 93). Another methodology that helps understand the artist and the documentation is the information on their GitHub, which I will use for the analysis, provided by the artist itself. Furthermore, I have used an article in which she explains her practices. “We discovered embedded in source code will continue the tradition of conservation research benefitting broader art-historical understanding about creativity in the age of digital production” (Engel 99). By looking at the source code, the artistic analysis gains more value and recognition in this debate, as it demonstrates the argumentation.

### Basics of CSS and HTML Coding

The artwork is constructed out of CSS and HTML code, but what does that mean? Hypertext Markup Language (HTML) and Cascading Style Sheets (CSS) work together a lot as their structures build on each other. HTML structures the page itself and the items, while CSS provides the styling and look. Therefore, the analysis will focus on the CSS code as it reflects the choices made and the underlying presumptions of aesthetic. In CSS, one uses selectors to choose the part of the HTML code that receives the properties for CSS to apply to it. A selector is set before the brackets. The brackets are the declaration, in which the properties are stated. Properties can vary from size, color, opacity and more.

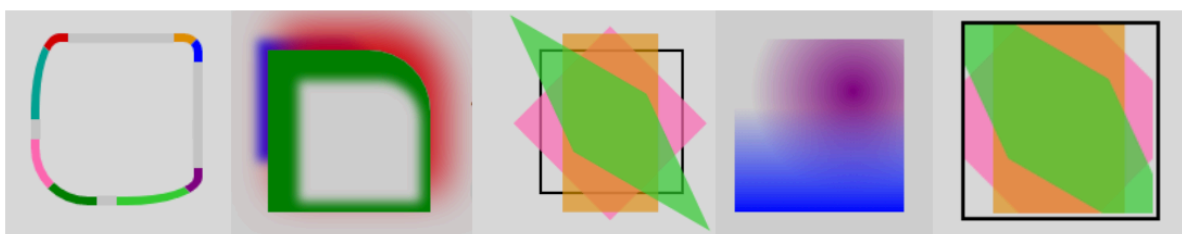


Figure 4: CSS commands (from left to right) : border-radius, box-shadow, transform (rotate, scale, skew), linear-gradient and radial-gradient, overflow: hidden

The artist herself, Diana Smith, has posted on her HOW page on her website, as well as on a blog, how she uses CSS for her artworks and what commands she favors. CSS only offers a limited set of rules and commands, which is very practical for designing, but when using it for artistic purposes, the language has to be tricked. HTML comes in squares, and while CSS can change their shape, the human body is formed with many different shapes. In order to change the shape of a box, she uses Border-radius to create extra shapes to shift on top of the original shape. She then puts these shapes into the same color as the background, so they blend in, only letting a part of the original shade peak through and therefore changing its appearance (Smith, "Solving Life's Problems with CSS").

The main five commands she uses to do these types of manipulations can be seen in Figure 4. Border-radius is a command that has eight variations, which refer to the various lining of the border, each variation is a different shade in the image above. If these are not directed upon, the shape would have a 90-degree angle. Box-shadow refers to the show around the edges created by the border-radius. It applies multiple layers and is the best artistic way to create depth, as by using several shadows with different shades, a blend of colors can be created. The parameters are border-radius: [X axis], [Y axis], [blur radius], [spread outward or inward]. For this it applies the code from left to right, this applies the shading from top to bottom, which is important to consider for the end aesthetic and color fade. The transform command has three options: rotate, scale and skew. They can be used all at once, or separately, depending on the desired perspective of the shape and it changes the angle and dimension it is positioned in. Linear-gradient and radial-gradient change the direction of the gradient of the shape or background itself. The linear-gradient follows a straight line, while the radial-gradient is circular. Without this command a shape would only have one solid color. The last command, Smith's favorite, is overflow:hidden, which compacts the artwork. It compresses the shapes that are on top of each other and removes the excess of extra shapes used to create further different shapes. This is what brings the artwork together (Smith, "Pure CSS Drawing Essentials"). She

uses these different techniques to play with the coding language. Natural languages are often used to play and construct meaning in poetic ways, creating new meanings and outcomes. Smiths' technique is similar and can be consider poetic as it plays with the commands and uses the poetic nature of language to create something meaningful and fantasy-like.

### Analysis: CSS Code

Using the commands explained above, this analysis will go into detail in a section of the code from *PureCSS Lace*, to show how the commands determine the visual outcome. I will focus on lines 224-234 of the CSS Code, which create the style for top left lip, as these incorporate many of her unique properties. The code can be seen in Figure 5.

```

224  .topleftlip {
225      background-color: #d08178;
226      background-image: linear-gradient(31deg, #1f0303 39%, #752727, rgba(99, 54, 34, 0) 96%);
227      box-shadow: inset 1.4em 0.2em 0.6em -0.6em #502323, inset -0.5em 0.3em 0.3em -0.3em #ad8162;
228      height: 106%;
229      left: 0%;
230      width: 37%;
231      bottom: -26%;
232      border-radius: 45% 50% 9% 51% / 94% 70% 30% 12%;
233      transform: rotate(-20deg);
234  }
```

Figure 5: CSS Code of *PureCSS Lace*, Lines 224-234

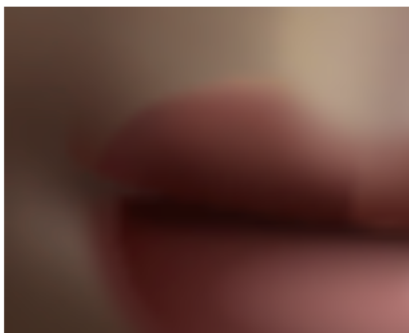


Figure 6: Top Left Lip of *PureCSS Lace*

This is the upper left lip of *PureCSS Lace* (Figure 6). I will now explain its appearance, color, shape and shading through reading the code. Line 224 states the declaration and determines the section that will be styled, in this case, the top left lip. The following lines explain the properties that will be done. Line 226 determines the background image, in which a

linear gradient goes straight, from a lighter pink to a darker, black shade. Line 227 determines the box shadow, which changes the gradient around the border radius (line 232) and creates different fades and shades around the lip, making it look smooth and blend into the background skin. Line 232 determines the border radius, and therefore, the shape of the lip and its plumpness, its curves of the edges of the box, using all eight parameters. Line 233 determines the angle as it has the command `transform:rotate`, which changes the direction of the shape. Line 234 ends the declaration. All these together create a seemingly natural looking lip instead of just a one colored box. But what does natural looking mean? The lip itself is not actually natural, but as seen in the lines of code, it is very constructed. It is a result of the code and its properties. This is similar to the idea of gender being a construct. Only through their performativity they gain expression. Gender and code, therefore, both have a certain performativity. Smith's work hence demonstrates Butler's thinking of performativity and Hayles thinking of connecting code and gender through this notion. Through using these different commands in one, artistic value is added to a normal simulation of a shape.

What is She hiding?

The code does not only determine the simulation of the written commands, different values and implications are also embedded. It reveals certain aspects of the world, but also highlights both notions on feminism in technology and in the craft of coding. Therefore, I ask, how does the CSS code of Pure CSS Lace, in its structure, function and expression, reflect notions about language and phallogocentrism?

The code was produced to create an artwork, not to stylize a normal website, which is its original function. Smith repurposes the language in order to create something artistic and meaningful. Smith engages with the act of artistic coding by creating her own techniques and thereby using the code for artistic purposes, this shows her power of the craft and artistry,

reflecting the potentiality of the female in the field. She uses her craft and re-functions it and brings in a female twist to the masculine narrative of code. While code is typically seen under the structure of command and control, as well as perfect executability, Smith plays with the language and reconfigures it to have the desired outcome for her expression. The layering of different commands allows her to manipulate shapes, originally squares, into concave and convex, differently textured shapes. The trickery and combination do not go against the rules of CSS but it is not what the program was intended to produce. Therefore, Smith plays with both the language and the executability, tricking it into a variable outcome. She breaks the myth of the code, debunked by boyd, and creates her own expressions, breaking the fantasy and the masculine power.

Furthermore, the aesthetic of this work is very particular and seems well chosen. As Diana Smith hand codes all her works, every step is deliberately taken and has a meaning. This makes the analysis of the code more difficult as the language and commands she uses are precisely chosen. The way the work is created is very intricate and is elaborately and technologically constructed. Smith plays with how code allows her to express herself, and by switching up the structure, negates the executability of some executors, and creates her own fantasy. This shows the instrumentality she uses and puts into this work and also the way she manipulates the code and achieves its performativity. Through her use of structure and her constructions, she changes the performance. This reflects Hayles thinking of performativity of code and gender, as it is not a neutral executor, but a language loaded with gender connotations and reflects the debunking of power structures.

Smith goes beyond the original usage of the code and creates these slippages, these grey zones, to create something that she wants, something that she has power over. She breaks the gender divide through her power, her subject, her simulations and all things surrounding this creation. The way she uses the language is similar to the way natural language would be used in poetry. She uses the words, in this case commands, in such a structure that brings creativity

and fantasy into the language, showing different meaningful beautiful outcomes. The way she creates it puts her into a powerful position as she determines and tweaks the outcomes. Her control lies over the structure, function and expression of the code and through that she uses language to reflect on the masculine idea of command and control and phallogocentrism.

The first part of this analysis has shown how the code is constructed and how we can analyze its importance and implications. It also has put language in relation to arguments from the first chapter stating the differences of natural language and coding languages and their similarities. Furthermore, it also has elaborated the importance of Smith and her works and how she positions the female subject and the female structures she embeds into the masculine coding language. The combination of commands creates an artistic value that goes beyond the original values that code brings forth, this leads to a new understanding and positioning of the female subject through the addition of aesthetic choices. The importance of aesthetics also lies in understanding the simulations and their socio- and cultural implications through their visuals, not just the code itself. The image can also be powerful. Therefore, the next section will look into a formal analysis of the image itself. What role does aesthetics have in the representation of this subject?

### She is Art

The aesthetics of the image also have an impact on the technological simulation. Why is it important to look at artistic properties, although it is not an actual image, only a simulation? The method for this artistic analysis will be formal analysis, which is explained by Sylvan Barnet in the book *A Short Guide to Writing about Art* (2015). A formal analysis focuses on “the form the artist produces; that is, an analysis of the work of art, which is made up of such things as line, shape, color, texture, mass, composition“ (Barnet 48). The analysis explains the properties of the artwork and their effect on the subject matter by assuming the

artwork has a certain meaning (Barnet 49). This distinguishes formal analysis from a description as formal analysis explains the function of elements, instead of only describing them (Barnet 50). This technique helps deconstruct the artwork and look at slippages in the meaning and representation (Barnet 51). The simulation analyzed will also be the Google Chrome version as this browser is chosen because it was the intended browser to display the code. Chapter 3 will go into simulations of different browsers and the results and implications that come from software variety. The sub-question this section aims to answer is: How can aesthetic artistic practices such as form, shade and texture help understand and deconstruct code in a way that expose masculine ideologies about data embedded in it? The formal qualities of *PureCSS Lace* reflect the techniques used to manipulate the code and empower the female subject.

#### Analysis: *PureCSS Lace* in Google Chrome

*PureCSS Lace* shows a sophisticated, bourgeoisie lady. It is inspired by a Northern Baroque or Reinnassance portrait painting, which can be seen by the stylistic choices. What aesthetic choices were taken? The artistic choices made reflect the confidence and power of the subject. The picture is a portrait, in which the subject is center to the image. She is positioned in the foreground, while the background is a black void. It is held in dark colors, her skin and the lace being the contrast to break up the dark composition. The lace is the center point of the image, as it is also what the title refers to. The lace is very intricate and detailed, showing the effort and time that went into constructing this work. Lace is known to be made and constructed by females and their delicate hands, and this notion reflects on the power in this artwork. Clothing like this was often worn by noble women in this time, as dark cloth was the most expensive to purchase and could not have been bought by everyone.



The colors also give the image a strong power. The saturation of colors is low, as the colors are dark or almost translucent looking. It is held in cold and low intensity colors. The lines are very detailed and powerful, as can be seen in the lace and the hair. The movement of lines is organic with the object and subject. There is a high contrast between the subject and the background, almost on opposite ends of the value scale. Furthermore, the texture varies, but is mainly smooth, just like the lace panels, clothing, cloak and skin.

This imagery reflects the notions from Chapter 1, in which the female in the art historical sphere is shown to be disadvantaged, but by making it a powerful subject, showing its craftsmanship and its internal superiority, it can return to its position and reclaim its power. These descriptions fit to the subject of this image. From the formal analysis, we can say the aesthetic artistic practices such as form, shade and texture help understand the subject matter and its connection to the first analysis of deconstructing code and exposing masculine ideologies embedded.

### She has Power, She is Both

The power of the combination of artistic values and code language lies in the way Smith creates her work. Diana Smith, as a female artist who only produces works of female subjects. This gives it a recognizable power, as seen in Chapter 1, this is not very common for the female to have this positioning in the artistic sphere. Female portraitists used to be the only artists to have agency to create paintings. West explained this, as these artists would be able to work from home, and therefore, follow the norm of society while still practicing the craft (West 145). Female artistry was more connected to crafting in the sense of producing lace or knitting and sewing, not as painters. Smith follows this narrative, as she works on her project from home and represents other females. Therefore, she represents the sphere of female artists and shows the potential for the positioning of the female. Additionally, she is a female working and coding

in the sphere of technology, this is again mentioned by Wajcman as not being common in Western society. The lack of females in the field was due to the lack of education offered to females in the past (Wajcman, “From Women And Technology To Gendered Technoscience” 288). This has changed to the extent that, in order to proceed coding art, no classic education is needed, as it can be self-thought and learned. Smith works in the technological realm, yet this type of coding and production of works is part of her hobby and she has learned to tweak these commands through her own practice. This situates a female artist in the sphere of art and technology, putting her in a vulnerable position, yet the way she constructs her works shows her power in this niche field.

The way the artwork is constructed is very intricate, both in its artistic form and also the way the code is manipulated. The artwork is based on the way the code allows itself to be expressed. Only what the code projects can actually become the image. In art, often the motif and representation are dependent on the skill of the artist. While this is the case here as well, technique in painting has different values and implications than the technique of coding for art. As discussed in the Introduction, the Greek word *techne* originates from the meaning of craftsmanship, which in this case can refer to both art and technology (Parry). The craftsmanship in this process is the production, which involves the skill and all surrounding parties of the craft. Painting and paint production have a historical background, this is situated in a gender dynamic in regards to production and sharing of skill and knowledge. Craftsmanship such as paint production is female centered, while painting is considered masculine and more powerful. Similarly, coding is gendered and the knowledge and skills are shared on an imbalance. As historically, females have not been given the education, according to Wajcman, showing again the connection between the two worlds from the perspective of the female subject (Wajcman, “From Women And Technology To Gendered Technoscience” 289). There is potentiality to break from this imbalance and that is what Smith does in combining these two fields.

The combination of art and code used to simulate this subject is very specific. The code restricts the way it looks, but as Smith stated, she uses tricks to make it appear the way she wants it to. She takes control over her subject and the simulation. The language is used to create different shadings or different fine details, shaping boxes into forms and dimension to get the desired outcome. Not everything is possible in CSS, but with enough commands anything can be simulated or hidden behind extra shapes, it is through this that the art can look the desired way. This extensive effort and control over the technology gives the artwork power and value, as it shows very single detail is desired and intentional.

Artistic practices with code in the realm of New Media give a new power to the subject and reinvent the two fields. New Media offers these capabilities and possibilities to create expressions like this, which allows for the creation of new spaces for females to make art but also a space for themselves. Wajcman elaborated on the potentiality of the female in technology and technoscience in her works, and how by creating new techniques and technologies there can be stronger female empowerment. Through the increase of opportunities and female-centered technologies, such as In vitro fertilization (IVF), the power is given to the female and her body, in making her own decisions and creating her own outcomes (Wajcman, "From Women And Technology To Gendered Technoscience" 290). By creating her own way of using code, Smith positions herself into a field with power and control over her own outcomes. Similarly, through the artistic value of the piece Smith establishes herself as an artist. Not only in creating portraits, but in capturing the essence of the work and mastering of the technique, the *techne*, the craftsmanship.

This applies for the image we see, but what if we suddenly see a different image? What if this control over the simulation is lost? What if the software takes over? The next section will focus on these phenomena. It will elaborate on the different outcomes due to different simulations by softwares and the role of manipulability of the code, as well as present the role of the female subject in this. Chapter 3 will therefore elaborate on this difference in browser

simulations and make the slippages and manipulability of code even clearer. It will demonstrate the loss of power in the masculine dynamic of command and control and show the power of mythical fantasy and artistic practices, as well as the power to overtake technology.

## The Creative Loophole

### Not One Simulation, but Many Simulations

The importance of software and its connection to code and language has been demonstrated over the last two chapters. In Chapter 1, we have come to understand the positioning of the female in the technological sphere and in Art History. In both, the female is undervalued and seen as an object for representation. The female is not granted power and is treated as less suitable for the field. Furthermore, the first chapter elaborates how coding is a language with gendered embeddings as well and its structural relation to software. These considerations lead us to the two-step Case study analysis in Chapter 2. The main outcomes of the analysis in Chapter 2 focus on the technological and art historical value of the artwork. The first part of the analysis uncovered the techniques used during coding and the unique way Smith, the artist, uses the language to create something that breaks the normal structure of language. The second part of the analysis reflects on the visual outcome of simulation of the code. This highlights the connection between art and technology, and hints towards the importance of software. This chapter will now explain its innate connection to art and how through the artistic practice connected to technology, software reveals new types of expression. It is the artistic practice and the artwork itself that reveals the slippage that occurs during the process, which triggers this discussion. The three main focuses of this chapter are: 1.) the role of instrumentality, 2.) the loss of command and control through software, and 3.) the masculine and feminine divide in technology and its practices.

The analysis from Chapter 2 focuses on the browser Google Chrome and its simulation. As the language was written for Google Chrome, the commands comply with the intended simulation. Yet, the artwork *PureCSS Lace* can also be viewed in other browsers. These read the language differently, as their processing functions differently, and therefore, the image is simulated differently. Some of these simulations can be seen below in Figure 7. I simulated the

artwork in several browsers to find these variations and screenshots them to capture their visualization. This variety of simulations from the same code is interesting because the different renditions of artwork through the varying browsers show the interlink of code and software. The overarching control and power of software influences the flexible performativity of code. It lies in the importance of the artwork and the artistic expression that uncovers the myths of code, with visualizations shedding light on the usually unquestioned structures of software. By looking at the instrumentality of technology and the loss of control and command through software simulations, I ask: What aspects of software functionalities of browsers does CSS art uncover? By deconstructing the simulations, how do they create meaningful statements through their data practices on implications of technological differences on the role of the female?

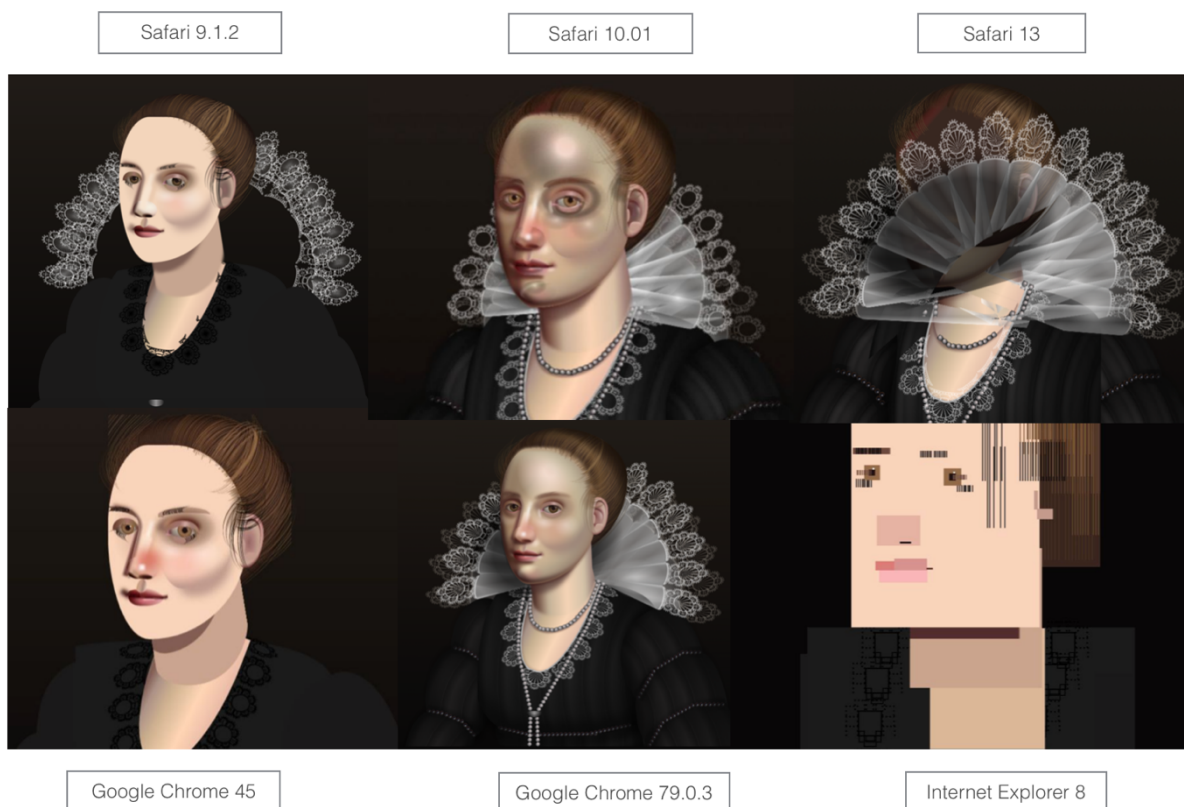


Figure 7: *PureCSS Lace* by Diana Smith simulated in different Browsers

## Analysis: Software Simulations

The images in Figure 7 show different simulations of the same CSS code. These different simulations demonstrate the power dynamic between the software and the code. The top row was simulated through different versions of Safari. The lower row shows versions in Google Chrome and Internet Explorer. The Google Chrome 79.0.3 version was discussed in the previous chapter. For this analysis, let me deconstruct the aesthetic of the artwork and the role of the code behind these simulations. The versions vary in different ways, yet their main markers are color, texture and shape, which are also the main visual pointers used to analyze the work in the previous chapter. There are many visual differences in the simulations. In the Safari 9.1.2 version for example, a lot of textures are missing, neither the face nor the neck have texture and they appear to be solid shapes. Furthermore, many parts of the lace are missing and the body, through the robe, almost blends into the background, as it is a solid shade in comparison to the original simulation. Similar patterns appear in the Google Chrome 45 version, but in this case the lace has completely vanished. In Safari 13, the shapes and details are more pronounced, but they are not in the right position. The lace is in front of the face, and therefore, ordered into the wrong position. In the Internet Explorer 8 version the artwork seems completely unrecognizable. All these versions show that through the simulations, different artistic renditions are created.

The reason these appear so different is because the browser and its software reads the same code differently. The connection between the browser and the code is determined by the proper function of translation from code into visual, and the smooth transition without being lost in technical translation. As explained before, Smith uses different techniques in coding and different commands in order to create the outcome she desires. Yet the code, if hand coded as intricately as this, has many layers to it and the possibility of a wrong or different display is very likely. The software reads the code and displays it, and in this case these softwares read it

differently, and therefore, not everything turns out the way it's supposed to. She uses five different commands to create her outcome, of which some of them did not read the intended way in these simulations. I base the intended outcome on the Google Chrome versions, as the artist stated that was this was the original. The command `overflow: hidden` is supposed to compact the artwork and merge shapes. It compresses the shapes that are on top of each other and removes the excess of extra frames that were used to create different formations. She layers a lot of shapes with these commands to create new shapes which read differently in many of the browsers. It is supposed to combine shapes, which in some cases did not work, for example, in Safari 13 the lace pops up in front of the face. `Box-shadow` refers to the shadow around the edges created by the `border-radius`. It applies multiple layers and is the best artistic way to create depth as by using several shadows, with different shades, a blend of colors can be created. The color of some aspects of the works might be different as the gradient did not flow the way it was intended. For example, these types of shadings went wrong in the Safari 10.01 version. The `transform` command has three options: `rotate`, `scale` and `skew`, and in some versions the rotation of shapes might be wrong, such as the eyelashes on Google Chrome 45. All of this is due to the connection between reading the code and the software simulating the code.

The simulation in Internet Explorer is particularly interesting as it simulates the artwork completely differently, the commands did not work out at all as intended. It shows the possibility of slippage between code and browser and the new space this creates for expression. The work is hardly recognizable, and all shapes become boxes and solid colors. This slippage occurs in the process of displaying between the code and the software. While the software was supposed to read the code in one way, it read it differently and created something according to its level of understanding. Frabetti mentions in her work that no matter how something is coded, the machine has the last say.

“Regardless of what humans think of a piece of code, the machine is the final arbiter of whether the code is intelligible. If the machine cannot read the code or if the program



does not work properly, then the code must be changed and corrected before the machine can make things happen” (Frabetti 45).

The software decides how it reads the code and what it displays. In this case, while being intelligible, the reading slipped, and control was lost. Although the machine created its own outcome, this outcome exposed the machine. This loss of control and slippage makes this version so interesting. It is the artistic value that highlights this slippage in control, this almost feminine poetic aspect in its artistic representation. This work is artistically layered, as now instead of looking like a Renaissance painting, it seems to be a work of the Cubism era. The change in simulation through reading switched the whole artistic notion of the work and brings many new structural and historical changes. The Renaissance has such a different historical background, not only in artistic technique but also in regard to socio-political situation and statements. In comparison to cubism, the two periods would normally never appear together, stem from each other or be comparable. These two periods are separated in Art History by more than 300 years and milestones in art can be found in between their timespan. In this Case study, however, through the simple simulation of code in a different browser, both periods are just a mouse click away from each other. By looking at the code and language in this way, it is as if we would X-Ray a painting to look at its historical past. We look beneath the surface to find the embedded structures. This poetic narrative of technology advancing and advancing time and space, plays with the mythological aspect of technology and code and shows its possibilities and wonders through the slippages in a seemingly rigid system. The aesthetic, artistic value contradicts the masculine control over code and shows how through visualization and simulation poetic notion is brought to a seemingly neutral language. Only through the misreading and the slip of control, the browser manages to change the artistic period and plays with underlying notions of masculinity and femininity. The browsers actually give the code more expressive freedom, so it is a blessing in disguise, which can create a dynamic discussion.

## Where is Her Aesthetic?

The relation between the aesthetic and code is complex, as we question, where is the aesthetic located, what determines the aesthetic? By finding out the location of the aesthetic, we can understand and locate where the slippage of code and software takes place. This is important to locate in order to understand the main argument of discussion. In the discussion so far, we have understood that it is the artwork and its properties that shed light on the manipulability of code and its interplay with software and its slippages. We see in the example above that aesthetic can create different outcomes and create narratives. Hence, by trying to pinpoint the location of the aesthetic, we can pinpoint the location of the slippage of control in technology. By locating the slippage, we can see where control is lost and where femininity enters the process.

Is the aesthetic in the code or the software? In this paragraph I argue the aesthetic lies in the code. The code creates the performance as it tells the software what to display. Where in this transaction is the aesthetic? Let us say in order for the image to appear on our screen we need the code to be translated by the software to create the image/the outcome, code → software → outcome. Now let us assume, the aesthetic is determined by the code. The code is the part of the artwork that is long-lasting, as once it is written down it is not changing. Therefore, as it is a piece of text it can be conserved, the software on the other hand is always changing through updates and bugs, changing its function over time. Because of this, the code can be said to contain the aesthetic of the work as it is the original determinator (Engel 99). Engel refers to code being the archive of digital media artworks, as it will trace back to the originality of the image, even when the software changes it will portray it differently or not portray it at all. “Technical documentation of source code will provide valuable information to future programmers to understand how artist-generated software programs work. They will be able to use the documentation to maintain the software when there are changes to the underlying

operating system and/or to the hardware” (Engel 92). So, while software can change, the code will always show the artwork and the initial aesthetic.

Now let us assume the opposite. What would happen if the aesthetic lies in the software? What if the code does not necessarily define the aesthetic, but is empty? This would mean the code is just the middle man of the message (before the code determined the outcome), and the software creates the outcome itself. The code gives information to the software, and the software interprets it and creates its own outcome. In this case the aesthetic would lie in the software. If the aesthetic lies in the software and not the code, would the slippage of technology become arbitrary and interpretable? Frabetti refers to Brooks in Chapter 4 of her book when talking about the connection of code and software, questioning if software is the process, the producer or the outcome? This asks, is software the process needed to translate the predetermined outcome by the code, or is software the producer of the outcome, or the outcome itself? For the software to have the aesthetic and the slippage to be located at this point, the software would need to get the code and create an outcome fixed to the processing functions. That would mean for the image to appear on our screen, code → software = outcome. The code goes into the software, and the software becomes the outcome due to the information it is fed. If the aesthetic is both the process and the product, the slippage lies within the whole system. “Software exists only as exteriorization. The distinction between ‘the ideal’ and its ‘realization’ seems therefore to become undone at the very moment of its establishment” (Frabetti 105). This means software is both the process and the product. “Brooks places the difficulty of developing a software system alternatively at the ‘material’ level, albeit this is contradicted by the supposed tractability of the computer as a support for inscription, and at the ‘ideal’ level, therefore making ‘realization’ (or ‘implementation’) the place where difficulties are actually clarified. This ambivalence suggests that the separation between conception and realization is ultimately untenable” (Frabetti 105). The connection between code and software is very complex, as the code defines what the software will display, but this dynamic does not always work. The

software depicts what it reads, but this blurs the lines of what it becomes as it sets its own parameters. This blurring of lines could be the presence of the slippage. Therefore, the aesthetic can also be found in both the process and the product. Software creates its own simulation and displays what it can read, creating the outcome in a poetic manner. In its own way, it acts upon the masculine code and brings in a feminine version of it. This leaves us to the question, in *PureCSS Lace*, where the software alters the look of the work, does the aesthetic lie in the software, or is it hidden in the commands of the code? How can we understand the point of slippage and unraveling through the power of aesthetics?

In *PureCSS Lace*, the aesthetic lies in the software and the code. This is due to the artistic nature of the piece and the extra embedded cultural values through its performativity. The performativity flips masculine notions of command and makes space for feminine notions of myth and fantasy. On one hand it is the software that portrays the aesthetic as the browsers changes the works, such as from Renaissance to Cubism. The software has the power to define the aesthetic and simulate an outcome to its understanding. On the other hand, the artwork was created for the browser Chrome, with a certain intended aesthetic and deliberately chosen commands. Therefore, the aesthetic is also determined by the code as it was created with the intention to look a certain way as stated by the artist. Ultimately, in *PureCSS Lace* the aesthetic lies in the connection between both code and software.

By questioning the artistic practice of a technological skill and looking at the actors and outcomes, Smiths artwork toys with the idea of aesthetic and its location. It raises questions through playing with performativity and manipulability of code and software. In this process slippages occur, through pinpointing these we can see the loss of masculine control and introduction of a more feminine poetic translation of language into simulation. The simulations show the loss of control of code and the power of software in its process and product, its ambiguity through the connection between original and product due to its performative and manipulability of values. The visual demonstration of coding art and its critique highlights the

relationship of code and software and the power dynamics, flipping masculine notions of command and control and feminine notions of myth and fantasy by meddling with manipulability and performativity.

### Rethinking the Divide

This myth of technology being masculine, neutral and controllable versus feminine language being poetic and expressive, has been touched on in previous chapters, especially in regard to the executability of the code but also in regard to critical data studies. Additionally, we have looked at the artwork in question from a software perspective and have tried to understand the role of art and aesthetic in these simulations. I have established that software reads the code and determines the outcome, being both the process and the product. Furthermore, software simulates the language, and plays with the performance of the code. This interplay of concepts has led us to manipulability, and how the manipulability in this whole process repositions the female in this sphere. How do the different browser simulations of the CSS art effect the understanding of the female in technology and art? How can we use art to understand the shift in technology and its underlying notions? By looking at the simulations of the artwork we can reveal the connection between the world of art and technological sphere and expose the similarities positing of the female in them and question its future transgression and progress.

### From Masculine to Feminine

The technological and art historical sphere are male dominated, from its members, to its subjects, to its notions and as well the surrounding discourse. The Case study has tried to describe the breakdown of masculine controlled notions embedded in technology and suggested an understanding of technology from a feminine poetic angle. The masculine dream of

command and control sets off with the notion of code being a neutral executor, portraying exactly what is wanted and expected. This has been disproven by the Case study through its deconstruction and slippages, which have been revealed through the artwork. The artwork, therefore, has shed light on previously hidden values.

The gender divide has been inherently embedded in the technological processes. It is not a new revelation but needs to be traced back in order to be revealed again. It is historically known for men to produce software as its production and outcomes relies on the man.

“The cost of a software project does vary ‘as the product of the number of men and the number of months’, Brooks argues, ‘but progress does not.’ Hence, the man-month as a unit for estimating the duration of a software project is a ‘dangerous and deceptive myth’, because ‘[i]t implies that men and months are interchangeable’, when in fact they are not” (Frabetti 106).

Men and months are not interchangeable, but the role of men in this case can still be considered irrelevant, as it is the performance of the technology and the product of the software that determines the outcome, not the gender. The relevance does not lie in the gender of the producer, but in the gendered process of the simulation and production.

The deconstruction of these notions in technology and the divide was already seen decades ago. Therefore, the potentiality of the female in this space has been existent, yet still requires rethinking and restructuring. “Derrida argues that contemporary technology is already in deconstruction because, although programmed and neutralized as controlled ‘development’, technological innovation still gives rise to unforeseen effects, he might as well have been thinking of the emergence of open source from the software engineering of the 1970s and 1980s” (Frabetti 115). It does not rely on the gender of the producer, but the techniques used and the understanding of the lack of control pertained. Moving away from the masculine idea of command and control, it has come to be understood that technology does not allow ultimate neutral outcomes. Instead it plays with its creation, as there is “the intrinsic instability of the

instrumentality of software” (Frabetti 116). The way the process involves interplay between the producer and the product creates a performative approach to technology that cannot be seen as neutrally executable. It contains a creativity and power of expression that shifts technological structures from being masculine to a more feminine poetic approach. This recognition of the loss of masculine power, leads to a more contemporary approach to technology, a deconstructed understanding. Technology is not an instrument, it is a process of different interactions and performances that have different influences and outcomes. In this case the connection of the code and the software create a performance in which simulations are determined by many factors, deconstructing the idea of control and instrumentality and highlighting the creative free-flowing process. It is the poetic nature of language that in this process, through the simulation of the code by the software, brings in a feminine aspect into the masculine technology. This play on language deconstructs the control and shows how code can become performative and through its manipulative notion act mythical. This deconstructive process and these slippages in the instrument, lead to the current positioning of the female and how Smiths work has helped encourage this rethinking. This results in much potentiality for the female in the technological sphere.

The female role in technology and coding is shifting. This shift could give potential to future females in the field and offer an understanding and opportunity to reclaim of space. Wendy Chun elaborates in her book on the history of the female in technology and therein states “the universality of [computer] protocol can give feminism something that it never had at its disposal, the obliteration of the masculine from beginning to end” (Chun 33). The technological sphere presents feminism with the opportunity to dissect, rewrite and produce this masculine dominated sphere. The question resides in, how can this be done? “What, however, is the relationship between feminization and feminism, between so-called feminine modes of control and feminism?” (Chun 33) How does the female take this control?

By looking at the feminine power of language and its performativity and manipulability, we reclaim a space that was female dominated in its early footsteps and are able to think about the relation between females and computer at the start of computing. The Case study has shown that by playing with the system, and by creating a performance and using artistic values, the female can gain power in this male dominated sphere. It also has shown the unpredictability of this apparent straightforwardness of technology, and the importance of recognizing the flexibility of simulations and execution. This flexibility can be valued and used by females to reposition themselves in the field and create their own execution, in their own female way of reproducing and using technology.

#### Female Executability

This brings us back to; how do the different browser simulations of the CSS artwork effect the understanding of the female in technology and art? The path of thinking has originated from the questioning of the artwork itself, how does code create art and what stands behind it? Without this questioning, the discussion would not have reached this point to be able to question the role of the female in technological productions. Software has created these unique simulations and shed light on the divide in technology and the actual hidden power underneath. The variations have shown the unpredictability in regard to source, process and outcome, demonstrating that many factors play a role in coding and creating coded artworks. Both performativity and manipulability of code have resulted in different readings and different outcomes. This has led to rethinking the female in the technological sphere and understanding the changes that are happening. The artwork brings forward an uncertain ambiguity. It highlights the female aspects in technology and how more feminine qualities are embedded into the process. It also shows the power female artists can have in uncover and reposting different fields, while on the other hand the artwork itself is still grappling with technology and control,



as it itself cannot control itself. The artwork is not always simulated the way it is supposed to, control is not in the hands of the artist but in the technology. Hence, while playing with this discourse and trying to unpack it, the artwork still falls underneath it. The control is in the artists' hands, but at the same time it is not, as technology is still in control of the artist. After all the technique used to create these works is heavily influenced by the programming language and its outcomes, only by tricking it and using it differently control was able to be gained. Hence, it is a game of cat and mouse. While criticizing technology, cybernetics and its ideas are still present. This ambiguity and divide show the potentiality and the need for more research, but also more terms in order to understand the processes and roles of control present.

Therefore, I claim, in order for this repositioning to become more prominent, this understanding to be clearer and more feminine, we need to establish an understanding of female executability. Female executability should stand for the interplay between code and software, which is deconstructed through its power dynamics through highlighting the performativity and manipulability of technology. Female executability can help shine a light on the lack of transparency in technology and unravel the myth of neutrality and control. It is female executability that highlights the potentiality in the game of control and the possibility for females to invent creative ways to play with technology. This breaks down its rigid instrumentality, and by finding loops to make it their own. By understanding all dynamics of the equation and bringing in unique patterns, female executability can be a new way of using and understanding technology in an empowering way.

This term, female executability, becomes present through the connection between coding and art, it shows how cultural studies and perspectives can give new insights to rigid ideas surrounding technology. This interdisciplinary approach furthers the understanding of both fields through different lenses. It is the artwork and the creative way of coding that shows sides of technology that are normally unquestioned such as neutral executability and the command and control of processes. But why are these unquestioned? How has technology become such a

big part of society and life, but its processes are taken for granted and accepted? Through looking at technology from this different angle we can see its embedding in society in a way similar to other fields, such as the art market. Why are females placed in inferior positions in art and technology? Why does a more feminine understanding give a better understanding to the field, discovering inequality and unfairness? By using an interdisciplinary approach, technology and art can help to better understand each other.

Why is the female so important in this discussion? It is needed in order to out trick the system. The female in the artwork and the process has allowed us to rethink masculine structures and uncover ideas through new perspectives. Through rethinking the female in both spheres, we can suggest a rethinking and understanding through female executability. The control might not be understood yet, and maybe it never will, as technology always has its own mind. The machine itself creates what it wants according to what it can, but by involving a new factor, female executability, in a playful aspect technology could be revolutionized. We now need to learn to understand how we can play with it and we can out trick the system with an unconventional approach.

## The Future is Female

The overall outcome of the analysis of *PureCSS Lace* shows the current positioning of the female in art and technology. It has become clear that by looking at technology and coding through art, we can unravel the myth of accuracy of code and show a glimpse of creativity and freedom. The main outcomes of this have been the demonstration of the manipulability of code, highlighting the connection between code and software, questioning the instrumentality of technology and its command and control aspect, showing the masculine and feminine divide in art and technology, and creating a new space for female executability.

In Chapter 1 we saw that the positioning of the female in art and technology has been in its current deconstructive state for decades. In Art History the female was seen as a housewife (Broude, “The Expanding Discourse“ 30), but through the art of portraiture the female claims their own artistic space, as through self-portraiture they could still follow social norms while creatively expressing themselves (West 145). Females in technology have also tried to reclaim their space through the inventions of female centered technologies such as In vitro fertilization (IVF) (Wajcman, “From Women And Technology To Gendered Technoscience“ 290). While both spheres have suppressed the power of females, they have managed to regain some space by understanding and rethinking the realms. In the world of coding, the combination of technology and software reflect these notions (Frabetti 34). Through questioning the construction of the coding language and its validity, the myth, as hinted by boyd, of code being the ultimate neutral truth, questioned by Frabetti, is debunked. During the translations from code to outcome by the software, information is lost and manipulated, which highlights the embedded gendered power of code.

The Case study in Chapter 2 reflects these finding further. From analyzing the code, we can come to understand that the CSS code constructed by Diana Smith for this work goes beyond the normal functionalities of the language through the way she tricks the commands

and layers them. After looking at the code through a source code analysis we can see that she creates a new, female centered narrative. Her combination of commands creates an artistic value that goes beyond the original values code brings forth and leads to a new understanding and positioning of the female subject through the addition of aesthetic choices. Through looking at the artistic side of the artwork through a formal analysis, practices such as form, shade and texture help understand the subject matter and its connection to the first analysis. This new medium of coding art gives a new space for female artists, just as Wajcman hinted in “Reflections on Gender“ in regards to the possibilities for females in cyberspace. It allows females to create images of women in a masculine technology and creating powerful messages by tricking the system on multiple levels.

These tricks and playful interactions with technology are revealed in Chapter 3 through the role of software. The simulations of the different renditions of the artwork show the interlink of code and software and the overarching control and power of software over code and flexible performativity of code, as mention by Frabetti through Butlers notion of performativity. The visual demonstration of coding art highlights the relationship of code and software and the power dynamics, flipping masculine and feminine notions by meddling with the manipulability. The simulations of the artwork reveal the connection between the world of art and technological sphere and expose the similarities of the positing of the female and question its future transgression. Therefore, we can suggest a rethinking and understanding through female executability.

The possibility for more female space in technology and art leads to a structural, not just individualistic, change. Through female executability we can understand technologies from completely different angle and new dimensions which lead the reinventing of functions that bring light upon many disciplines. Overall, the research has shown that the neutrality of code is non-existent, and its truth is hiding behind a masculine curtain of command and control. By shedding light on this through cultural practices such as art, we have unraveled these complex

concepts and constructions and shown the possibility for creativity expression and the female. In the future we will not need objects such as Kodak Shirley cards for the standardization of colors in photography because 1.) females will have out-tricked technology and 2.) we can be creative, expressive, loud, and feminine.

## Works Cited

- Barco, Mandalit del. "How Kodak's Shirley Cards Set Photography's Skin-Tone Standard." *NPR*, NPR, 13 Nov. 2014, [www.npr.org/2014/11/13/363517842/for-decades-kodak-s-shirley-cards-set-photography-s-skin-tone-standard?t=1579260405262](http://www.npr.org/2014/11/13/363517842/for-decades-kodak-s-shirley-cards-set-photography-s-skin-tone-standard?t=1579260405262).
- Barnet, Sylvan. *A Short Guide to Writing about Art*. 11th ed., Pearson, 2015.
- Broude, Norma. *Feminism and Art History: Questioning the Litany*. Routledge, 1982.
- Broude, Norma, and Mary D. Garrard. *The Expanding Discourse, Feminism and Art History*. Westview Press, 1992.
- Buikema, Rosemarie, and Iris van der Tuin. *Doing Gender in Media, Art and Culture a Comprehensive Guide to Gender Studies*. Routledge, 2018.
- Butler, Judith. *Gender Trouble: Feminism and the Subversion of Identity*. Routledge, 1990.
- Boyd, Danah, and Kate Crawford. "Critical Questions For Big Data." *Information, Communication & Society*, vol. 15, no. 5, 10 May 2012, pp. 662–679., doi:10.1080/1369118x.2012.678878.
- Cifor, M., et al. *Feminist Data Manifest-No*. [www.manifestno.com/](http://www.manifestno.com/).
- Chun, Wendy Hui Kyong. *Programmed Visions : Software and Memory*. MIT Press, 2011. Accessed 2019.
- Cyanharlow. "PureCSS Lace ReadMe". (2019). GitHub repository. <https://github.com/cyanharlow/purecss-lace/blob/master/README.md>.
- Engel Deena, and Wharton Glen. "Source Code Analysis As Technical Art History." *Journal of the American Institute for Conservation*, vol. 54, no. 2, 2015, pp. 91–101., doi: 10.1179/1945233015Y.0000000004.
- Frabetti, Federica. *Software Theory: A Cultural and Philosophical Study*. Rowman & Littlefield International, 2015.

Geerts, Evelien, and Iris Van Der Tuin. “The Feminist Futures of Reading Diffractively: How Barads Methodology Replaces Conflict-Based Readings of Beauvoir and Irigaray.”

*Rhizomes: Cultural Studies in Emerging Knowledge*, no. 30, 2016, doi:10.20415/rhiz/030.e02.

King, Margaret L. *Women of the Renaissance*. University of Chicago Press, 1991. Accessed 2020.

Parry, Richard. “Episteme and Techne.” *Stanford Encyclopedia of Philosophy*, Stanford University, 27 Mar. 2020, plato.stanford.edu/entries/episteme-techne/.

Smith, Diana. 2019. CSS source code [Source code].

<https://diana-adrienne.com/purecss-lace/>.

---. “Pure CSS Drawing Essentials.” *HOW - Pure CSS*, diana-adrienne.com/how/#\_.

---. “Solving Life's Problems with CSS.” *CSS-Tricks*, 28 May 2018, [css-tricks.com/solving-lifes-problems-with-css/](https://css-tricks.com/solving-lifes-problems-with-css/).

---. <https://github.com/cyanharlow/purecss-lace>.

Wajcman, Judy. “From Women And Technology To Gendered Technoscience.” *Information, Communication & Society*, vol. 10, no. 3, 21 June 2007, pp. 287–298., doi: 10.1080/13691180701409770.

---. “Reflections on Gender and Technology Studies:” *Social Studies of Science*, vol. 30, no. 3, 2000, pp. 447–464., doi:10.1177/030631200030003005.

West, Shearer. *Portraiture*. Oxford University Press, 2004.

@cyanharlow. “I now stand seriously corrected about viewing my work in non-Chrome browsers. The results are so much more interesting than I could've tried to achieve on purpose.” Twitter, 1 May 2018, 10:19 a.m.,

<https://twitter.com/cyanharlow/status/991366472000655360>.