UNIVERSITY OF CALIFORNIA
SANTA CRUZ

SLIMEART

A thesis submitted in partial satisfaction of the
requirements for the degree of

Master of Science

in

COMPUTATIONAL MEDIA

by

Montana Fowler

September 2021

The Thesis of Montana Fowler
is approved:

Angus G. Forbes, Chair

Leila Takayama

Peter Beihl
Vice Provost and Dean of Graduate Studies

Table of Contents

List of Figures

Abstract

SlimeArt

by

Montana Fowler

SlimeArt is a Casual Creator generative art tool that allows users to create dynamic, particle designs with the Monte Carlo Physarum model using a particle brush and a deposit brush. Casual Creators are creativity support tools that value the feeling of enjoyment, the support of exploration, and usability of the interface. Through a user study (N=15), SlimeArt was evaluated with the Creativity Support Index showing it supported exploration, immersion, and enjoyment above expressiveness, results worth effort, and collaboration as hypothesized beforehand. By observing users' questions, suggestions, and behaviors in the tool, we were able to identify insights as to how we should redesign the presentation of the particle deposit strength slider to help users better understand how the particles and deposit interact. In addition to the usability study, the SlimeArt Encouragement Study placed users into two conditions to evaluate if the presence of an encouraging virtual agent, SlimeBot, affected users' creative self-efficacy, perception of ownership over their art, or level of exploration. While SlimeBot did not affect users' creative self-efficacy and perception of ownership, it did show a significant effect on how users reported SlimeArt's support of exploration in the Creativity Support Index. A thorough investigation into the usability of SlimeArt and the effects of SlimeBot presented the issues with the first design which led us to identify the

ways SlimeArt's redesign could better support users to create evolving network designs modeled after slime mold.

dedicated to my adventurous, goofy family

Acknowledgments

I would first like to thank my advisor, Prof. Angus Forbes, for his mentorship and support, as well as, Prof. Leila Takayama for her help with this project. I would also like to thank my four undergraduate research assistants, Keenan Camacho (lead RA), Laura Jasmin Gavia, Tanisha Mandal, and Azzaya Munkhbat for their work conducting user studies, transcribing recordings, and making it all more fun for me. Additionally, I want to thank the Computational Media students, faculty, and staff for providing so much support during a challenging time.

Chapter 1

Introduction

When creating with a medium, an artist takes the physical limitations of that particular medium and works with it until they can express themselves using it, such as a painter working with the transparency of their paint to create layers. Working with digital materials is much the same, but instead of working with physical properties of a material, the properties are designed with software. The constraints of a digital space determine the artwork that can be created, and the interface determines how the artist can work with the digital materials. In Jennifer Jacob's Dynamic Brushes, users designed brushes that generated patterns [14]. They programmed their brushes to form intricate designs. They controlled their medium with visual programming and manual manipulation with a stylus. SlimeArt leverages the computational creativity of the Monte Carlo Physarum Machine (MCPM) [10] so artists can create generative art out of the emergent networks the particles create. In Elek's work MCPM modeled astrological data to create the first visualization of the cosmic web, and SlimeArt leverages the model to create generative art. Particles create designs by looking for deposit. They sample the deposit texture from directly ahead of them and from another random angle within their field of view. The Monte Carlo Physarum Machine extends Jones' Physarum model by adding an additional random sampling of deposit [15]. In the Jones model, particles follow the deposit by sampling two directions with the same angle between them each time. The Monte CarloPhysarum Machine produces networks that are less tight with an element of randomness. Artists can use the particle brush to add agents to the scene with their own custom parameters: their visibility distance, speed, field of view, and deposit strength. Artists can also use the deposit brush to add structure to their dynamic design. In SlimeArt, the designs the particles form can be more or less controlled by the user. By having the particles follow each other as they emit deposit with a small field of view, they form lines and bend into circles before exploding into lines again. In this state of emitting deposit themselves, the designs are difficult to control, and the user must embrace that lack of control over the medium as if they are splattering paint on a canvas. If the artist wants more control over their SlimeArt design, they can set the particle deposit strength to zero so the particles no longer follow each other, and instead they will form networks among deposit structures designed by the user. One of the main design flaws of SlimeArt is how punishing it is for the novice user [23] because it does not have a way for an artist to partially remove their design. There is only a clear canvas button that clears the deposit and particle canvases completely. Six users requested either undo/redo functionality or an eraser. This would make creating with SlimeArt more forgiving and more in line with the Casual Creators definition [7]. In addition to the usability of the design affecting users' perceived ownership over the generative art they created with SlimeArt, we also tested the effect of an encouraging virtual agent being in the system. Given how difficult it can be to control generative art, we were curious to see the effect of an encouraging agent on users' perception of ownership, creative self-efficacy, and level of exploration. For users in the SlimeBot condition, SlimeBot would provide encouragement and instructional tips when the user interacted with certain UI components. We tested the effect of SlimeBot on users' creative self-efficacy [20] and balanced the measurement with their Creative Mindset Scale to ensure we knew their growth/fixed creativity mindset when analyzing their task-specific creative self-efficacy measurement [16]. To measure their level of exploration, we analyzed the click data in the conditions with and without SlimeBot to see if users explored more of the UI components with SlimeBot's presence. However, we did not discover a significant relationship until we analyzed how the conditions affected the components of the Creativity Support Index. We discovered that users reported SlimeArt supported exploration better when they were in the SlimeBot experimental condition.

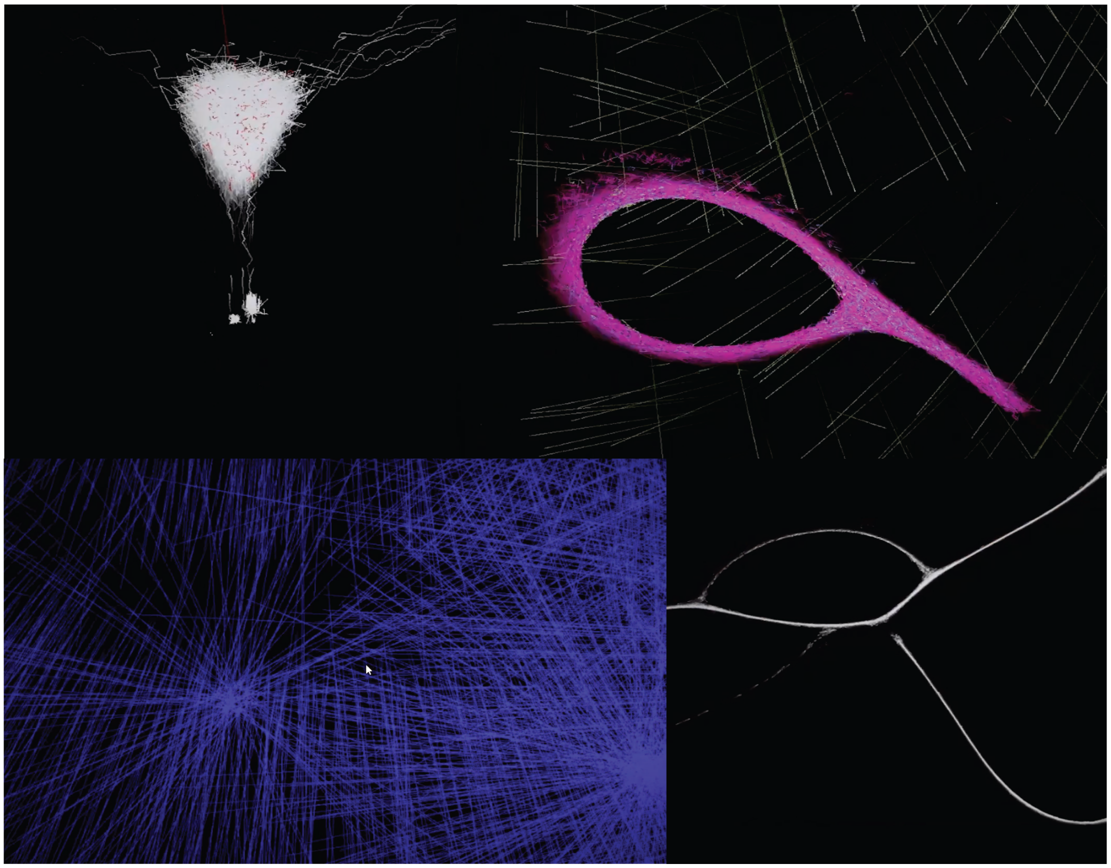

Figure 1.1: Top left: A participant attempted to drag down a clump of particles by adding more below them. Top right: A participant scattered particles with a low visibility distance around their emergent network so they look like rain drops in the background. Bottom left: Particles scatter leaving a long trace because they leave no deposit. Bottom right: A view of the deposit texture to show users the deposit particles are emitting as they create their own networks during the introductory task.

Chapter 2

Related Work

2.1 Generative Art and Computational Creativity

Computational Creativity and Generative Art, while related, have distinct definitions. Generative art's definition focuses on an artist's use of a computational system to create a piece of work [13], whereas Computational Creativity focuses on the creativity of the system itself [6]. Developing systems where both the artist can create with a computationally creative system requires studying mixed-initiative co-creativity, since both the creativity of the system and the artist are involved[19]. In SlimeArt, the algorithm itself can be used in a passive manner, allowing it to evolve and create on its own, or the user can choose to control the behavior of the agents more with more structure in the design. According to Boden, evaluating creativity requires an evaluation of both novelty and value, and, therefore, when evaluating computational creativity we must apply these metrics [2]. Boden acknowledges that novelty and value are subjective given the context of the evaluation, and therefore, we need to consider this context when evaluating the novelty through separating creative evaluation from historical (H-creative) and psychological (P-creative) contexts [2]. To analyze the H-creative context for SlimeArt, we must investigate how physarum-inspired models have already been used creatively. Physarum polycephalum, also known as slime mold, can solve shortest path problems [3]. The algorithm built into SlimeArt, Monte Carlo Physarum Machine (MCPM) [10], was adapted from Jones' Physarum Model [15], and it has already been used for solving astrological problems [4]. Existing computational artists use physarum as a basis for their art. Sage Jenson is an artist based in Berlin who builds greyscale simulations that are based on biological algorithms, some of which use Jones' Physarum Model. Jenson does not share specific details of their implementation on their website, but they do build their own systems (sagejenson.com). Given these historically novel contexts for physarum-based algorithms, SlimeArt's H-creative component is defined by its slider driven interface and particle-specific attributes. When modeling scientific data MCPM parameters are defined globally for the agents: their move/turn angles, move distance, sense distance, etc. When the purpose of the agents is to discover a scientifically accurate network among data points, the parameters being tuned globally ensures the agents behave uniformly when interacting with the data. This gives the most accurate shortest-path discovery. However, for SlimeArt, the agents are not scientifically fitting data, but are instead creating engaging visuals, so artists need to be able to create particles with different attributes in the same simulation.

2.2 Casual Creators and Creativity Support Tools

Casual Creators [7] are a subcategory of Creativity Support Tools (CSTs) [23] that prioritize enjoyment over productivity in an interactive intelligent system. Casual Creators support the enjoyment of fast paced AI-co-creation. According to Compton, users of casual creators will experience "the fast, confident, and pleasurable exploration of a possibility space, resulting in the creation or discovery of surprising new artifacts that bring feelings of pride, ownership, and creativity to the users that make them" [7]. In having this focus on enjoyment over productivity, Casual Creators are designed so that the user has enough control to feel ownership over the artwork they create, but the tool provides enough support to lower the amount of cognitive load to keep the enjoyment and flow [8] of the user's experience. The combination of fast exploration and being surprised by outcomes of the interactive tool help the user feel more pride and ownership over their Casual Creator artifacts. When referring back to Boden's requirement that we evaluate computational creativity we must consider both novelty and value contextually [2]. Compton argues that the value in the Casual Creator context is placed more on the enjoyment than the productivity in contrast to professional design tools like Photoshop. When evaluating SlimeArt as a Casual Creator CST, we used the Creativity Support Index which measures a tool's level of collaboration, enjoyment, exploration, expressiveness, immersion, and results worth effort by a combined pair-wise evaluation and a 10-scale rating of the tool in the different categories [5]. We hypothesized that SlimeArt would receive a low score for collaboration, but higher scores for enjoyment, exploration and immersion.

2.3 Exploration

The main design challenge for SlimeArt is deciding how to present the possibility space of actions for an algorithm that user's should not need a full understanding of in order to create engaging results. To minimize the need to understand the Physarum model, the user is only presented with a limited number of parameters to tweak and two types of brushes to divide the possibility space. According to Schneiderman's theory of Direct Manipulation [22], users can learn a tool quickly by getting immediate feedback from their actions and are able to reverse these actions so that they are able to make mistakes during exploration. Since SlimeArt does not have erasing or undo/redo implemented yet, our initial user testing of SlimeArt found evidence supporting that not having reversal functionality was frustrating to novice users [9]. This harmed their situated creativity by taking them out of the flow of creating with the tool. To alleviate the stress of having made a mistake, an encouraging virtual agent could add a presence that influences the creative self-efficacy, or the creative confidence of the user as they perform fast exploration of the tool. SlimeBot's tips for how to use the sliders in SlimeArt could support distributed creativity, where the CST helps alleviate the tool-specific knowledge required for a task [9]. In order to test the flow users could achieve when using SlimeArt, they are provided an exploratory task where there are no goals set for the user. During this task, the user can engage in embodied creativity, where they gain creative insights through interacting with the environment presented by SlimeArt through epistemic actions [9, 18].

2.4 Social Agent Coaching and Feedback

Social agent coaching has been shown to provide instructional content to users through social interaction, tracking user behavior, and providing feedback on the user's task performance. Relational agents are used to develop long-term social-emotional relationships with their users [1]. In addition to these initial studies, Fasola et al. discovered that participants prefered receiving praise in addition to feedback from their robot for motivation to motivate physical exercise [12]. Since our SlimeArt application prioritizes the enjoyment of the creation of the artwork as a Casual Creator tool, in this study we explored the effect of a virtual agent providing encouragement to the user in this relational way would increase their creative self-efficacy, perception of ownership, and level of exploration. According to Maedche et al., acceptance of a user assistance system, or in our case, a virtual agent, the user must perceive a high level of intelligence of the AI and it must have a high level of interaction for the user [21]. The assistant behaving in ways that are unexpected take the user out of the task and make them wonder about the system. In order to prevent this from happening with our users, SlimeBot needed to be both specific enough to be perceived as intelligent [21] without saying anything out of context that could be interpreted incorrectly. One specific example is that SlimeBot does not begin providing encouraging messages until the user has started drawing. In Silva-Coira et al., their virtual assistant for a gamified environment had natural language processing and reactions to the player's game data in its reactions [24]. Since SlimeArt already is a fast moving Casual Creator system, we did not want the user being interrupted in their creation to type into a chat with a chatbot, so the interaction the user had with SlimeBot was through using SlimeArt. SlimeBot would react to the UI that the user decided to change so that the tips were not entirely out of context and could provide necessary information in the moment the user needed it.

2.5 Creative Self-Efficacy and Creative Mindset

Creative self-efficacy is defined as the creative confidence a person has for a particular task [20]. Since SlimeArt may be a unique experience to users who have not tried generative art before, we wanted to keep our creativity measurement to one that is task specific and focused on the immediate moment rather than a more permanent belief about creativity. To additionally take into account the user's personal creative mindset when analyzing their creative self-efficacy scale, we used the Creative Mindset Scale [16], which tests the participants' fixed/growth mindset when it comes to both little-c creativity and big-C creativity [17]. This measurement is only taken during the pre-survey as it is unlikely a session with SlimeArt would affect the user's creative mindset. This measurement can help us understand how much of a growth mindset a user has when analyzing how their creative self-efficacy changed during the study. According to Karwowski et al., there are four main recommendations when designing questions to measure creative self-efficacy for a particular task: *future orientation, perceptions of confidence, key features or levels of task performance,* and *broader-ranging response scales* [11]. We used all four of these recommendations when crafting our questions in the pre and post survey in both of our conditions for SlimeArt, with and without SlimeBot.

Chapter 3

SlimeArt Usability and Design

3.1 Method

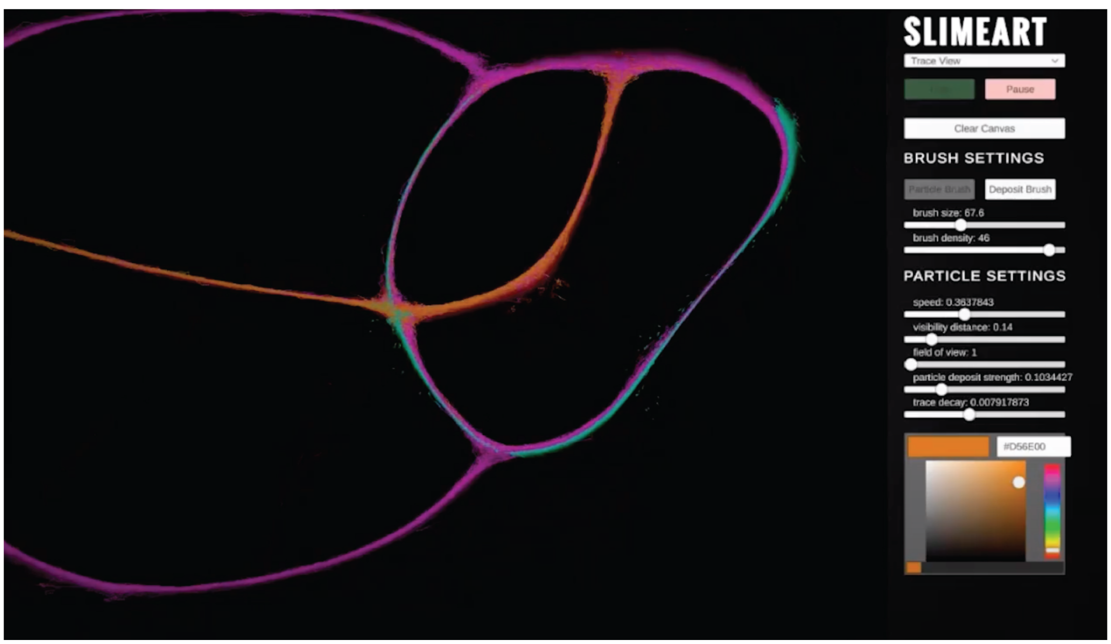

Figure 3.1: This is a view of the interface of SlimeArt without SlimeBot for the user studies. The user currently has the particle brush selected and is viewing the particle settings.

3.1.1 Implementation

SlimeArt is a Unity Application that runs the physarum model through compute shaders which are only available on Windows operating systems. The Physarum model was implemented in 2D with buffers holding the different parameter settings for the different particles at the time they are drawn on the canvas. SlimeArt provides two different types of brushes to the user: a particle brush and a deposit brush. The particles' movement is determined by parameters like their speed, field of view, and visibility distance as they look for deposit in the deposit texture. The deposit brush allows the user to draw static deposit emitters into the deposit texture to provide structure to the emergent networks the particles create. Particles can also create their own emergent networks by emitting small levels of deposit themselves. This helps them follow each other to create paths in the network. The particle's appearance is determined by their color and trace decay. The trace decay variable controls the trail particles leave behind as they travel. A permanent trace leaves a scribbly network of colors, showing every location state for each particle. By having a medium trace, the networks appear more blended as the decaying tails have a blurry effect on the appearance of the network. (Figure 3.2)

3.1.1.1 Space Management and Accessibility

When a user initially begins the application, they are prompted to choose a level of quality of graphics for their hardware. This decision needs to be made immediately to set up the length of the buffers based on the number of particles the user's

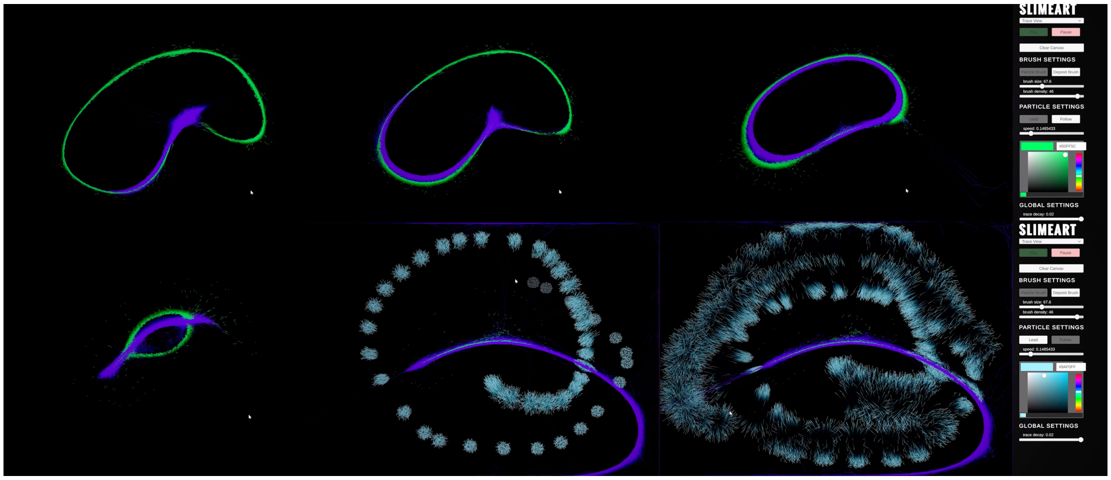

Figure 3.2: From left to right along the rows you can see a sequence of a SlimeArt drawing evolving. In the first four stages, only particles emitting deposit themselves are on the canvas. They follow each other and collapse into a line. In this piece, the purple particles are moving much faster than the green particles and they quickly loop around the green structure. Then in the last two frames, the particle deposit strength is zero for the blue particles. They wander aimlessly until they find the deposit emitted by the purple and green particles.

GPU will be able to handle. The amount of particles a user can have on the screen at one time is limited by $\frac{1}{4}$ of the size of the buffers allocated with the initial graphics quality selection. There were several design options explored to decide how to manage this space limitation. As a design choice, I wanted the user to not be obviously visually limited by the amount of space they had. When a user ran out of space and their particles overwrote the oldest particles on the screen, it was obvious that the user was running out of particles. When the user had the lowest quality graphics setting, it was even more inhibiting since the user would overwrite their own particles quickly. To hide the overwriting of the particles in the buffer data, particles are given a random index when they are created so that they randomly overwrite particles so that the user does not see large groups of particles expire all at once, which was the effect of overwriting the oldest particles on the screen to add new ones. Additionally, to improve the experience of users using the lowest level of graphics quality, the level of brush density is managed so that users with less space have fewer particles in each brush stroke. This extended the particle space for users with the lowest graphics quality to last longer.

3.1.1.2 Data Collection

To collect the click data from users, every click is saved with a time-stamp and the values associated with the UI at the time of the click. The data was saved into a csv file in the user's AppData folder which is uploaded in the post-survey for the researchers to analyze. The zoom call of the user study was recorded so that the participant's screen with SlimeArt was visible, but in post processing their image was cut out of the video, and the audio was transcribed into subtitles and a transcript so the audio could be deleted from the video. All data was stored on the UCSC google drive.

3.1.2 User Study Design

Fifteen UCSC affiliated participants who were over 18 years old were recruited through word of mouth and UCSC online groups. The studies were conducted on zoom by a member of our team: **Keenan Camacho** (lead Research Assistant), **Laura Jasmin Gavia**, **Tanisha Mandal**, and **Azzaya Munkhbat**. The researcher for the study session walked the user through a pre-written script to ensure that the same instructions were given from each member of the team.

3.1.2.1 Pre-survey

To participate in the study, users took a pre-survey to understand their level of interest and experience in digital art. Optionally, users could consent to upload example pieces of their work for us to compare with their SlimeArt output. During the pre-survey, users were also asked the creative self-efficacy measurement questions to evaluate the effect of using SlimeArt with their perception of their own confidence in accomplishing tasks within a generative art system [20]. Additionally, the pre-survey measured the user's fixed-growth mindset as it pertains to creativity using the Creative Mindset Scale [16]. While SlimeArt would not attempt to change the creative mindset of its users, the Creative Mindset Scale helped in evaluating how the user perceived their own growth and learning in answering the creative self-efficacy questions, as their answers are also influenced by their creative mindset as a whole.

3.1.2.2 Tasks

After choosing the level of graphics quality best suited for the user's windows computer, the researcher running the study would walk them through the *introductory tasks* to both familiarize the user with the tool while also evaluating how intuitive the interface design was at first glance. At first the user is shown the most uncontrollable state of the tool when the particles emit deposit themselves and they follow each other creating their own moving networks. Then the user is asked to look at the deposit view to see the deposit those particles are emitting before clearing the canvas and drawing particles that do not emit any deposit on the screen. They notice that these particles travel aimlessly and do not follow each other since they are not emitting deposit. Then the user is asked to pause the simulation in order to introduce a strategy for design within SlimeArt: by pausing the simulation it is easier to have time to conceptualize a design without the simulation changing. With a paused simulation, the user is asked to use the deposit brush to put deposit on the canvas. Once they have done so, they are asked to release the simulation by pressing play. They then notice the particles that were moving aimlessly are now rushing towards the deposit they drew. Finally, the user is walked through using the speed slider and the trace decay slider. The main goal of the introductory task is to familiarize the user with how the particles and deposit interact in order for them to best set goals in the future for what they would like to create. After the introductory task, the users complete a two minute *exploratory task* where the researcher observes the user explore the tool without any specific goals in mind. During this task, the researcher identifies trends in instinctual exploratory behavior by the user. Without a goal in mind, the users would not have as many expectations for themselves with what they create, so the researcher can note what they are drawn to trying. Following the exploratory task, the researcher would ask the participant to pause and conceptualize a plan for a design they would like to create with the tool. During this *planning task* users would describe something they would like to make. During this task we saw after a novice level of exploration what ideas the user would come up with to create with SlimeArt. After the planning task, the user would be encouraged to attempt to execute their planned design. Some users were more willing to adapt their goal as they discovered new artifacts in the tool while others stuck more strictly to their original idea. During the *execution task*, since users were working with a more specific goal, we could discover what functionality they wished they had to accomplish their planned design. Unlike in the exploratory task, the execution task revealed more insight into the design breakdowns in how the user believed the tool worked and how the tool actually behaved. By covering basic usability evaluations in the introductory task, exploration without instruction, planning, and execution of planning we could evaluate SlimeArt's design to novice users.

3.1.3 Evaluation

In order to evaluate SlimeArt to a CST standard, we collected the user's evaluation of the tool according to the Creativity Support Index during the post-survey. The Creativity Support Index evaluates a CST's level of *collaboration, enjoyment, exploration, expressiveness, immersion*, and *results worth effort* by a combined pair-wise evaluation and a 10-scale rating of the tool in the different categories [5]. We expected the Creativity Support Index of SlimeArt to have a low score in collaboration, but a higher score in enjoyment, exploration, and immersion. In addition to the Creativity Support Index, we also evaluated the transcripts of the study sessions to analyze the moments of confusion and the overall reactions to the behavior of SlimeArt during the four tasks. During the introductory tasks, we analyzed the videos and transcripts to see which aspects of the introductory tasks were the most confusing or the most surprising based on the user's reactions. During the exploratory task, we observed which UI components were most common to explore and trends in behavior during this exploration task that could tell us what was most inviting to try initially after the introductory tasks. During the planning task, we observed what designs users thought they would be able to create with SlimeArt given their novice skill level. During the execution task, we observed how users went about creating their design and we could see where their points of confusion were in how the tool worked. In addition to our own analysis and observations, we gained insight into the usability of SlimeArt based on how users said they would describe the tool to someone and their own suggestions for what to change.

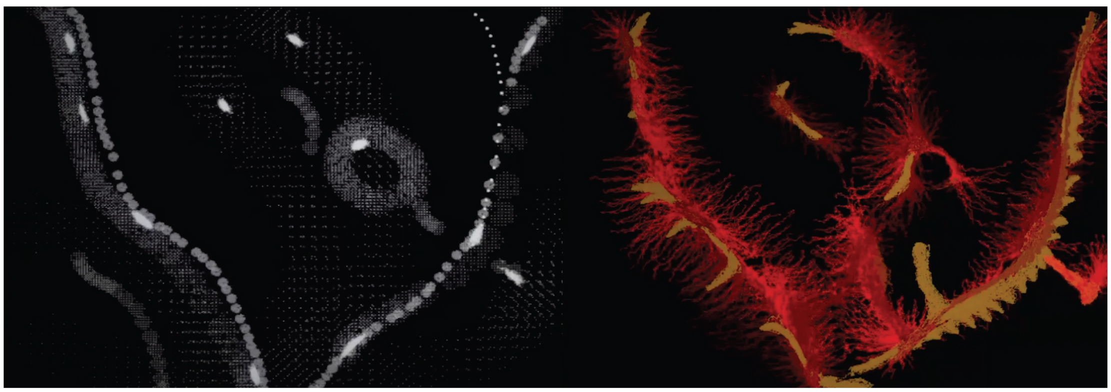

Figure 3.3: Here is a comparison between the deposit view and the particle view of a participant's design.

3.2 Results

3.2.1 Creativity Support Index

Our hypothesis for the Creativity Support Index was that we would receive a high score on enjoyment, exploration, and immersion with a low score for collaboration. As per the recommendations by Cherry et al., we left in the collaboration component as a check to make sure users were reading the questions carefully even though our tool and study were not designed to analyze collaboration within SlimeArt [5]. Our overall CSI score mean from all 15 participants was 68.06/100.00, with a median CSI of 66.17, and a standard deviation of 14.53. When we broke down the different categories, we found support for our hypothesis that our highest scoring subcategories were exploration (mean 46.88, median 48.00, sd 17.70), enjoyment (mean 41.06, median 34.00, sd 25.89), and immersion (mean 38.94, median 40.00, sd 18.04). Followed closely by the expressiveness category with a mean of 37.88, median of 31.00, and standard deviation of 19.15. Our second to last scoring component was results worth effort (mean 33.44, median 37.50, sd 16.41), and in last place, as predicted, was the collaboration score (mean 6.00, median 0.00, sd 9.76). We did note that one participant ranked collaboration as a 10/10 which made us reconsider their post-survey responses, just as Cherry et al. recommended. When breaking down our CSI score, we could understand which of these standard Creativity Support Tool values SlimeArt should improve on in design. Raising the results worth effort score will be a future goal of the project by redesigning the aspects of the interface that confused the most participants in the study. Having higher levels of exploration, enjoyment, immersion, and expressiveness demonstrated that while these will also likely improve with a redesign, SlimeArt is on track to being an enjoyable Casual Creator where users can immerse themselves in the MCPM parameter space.

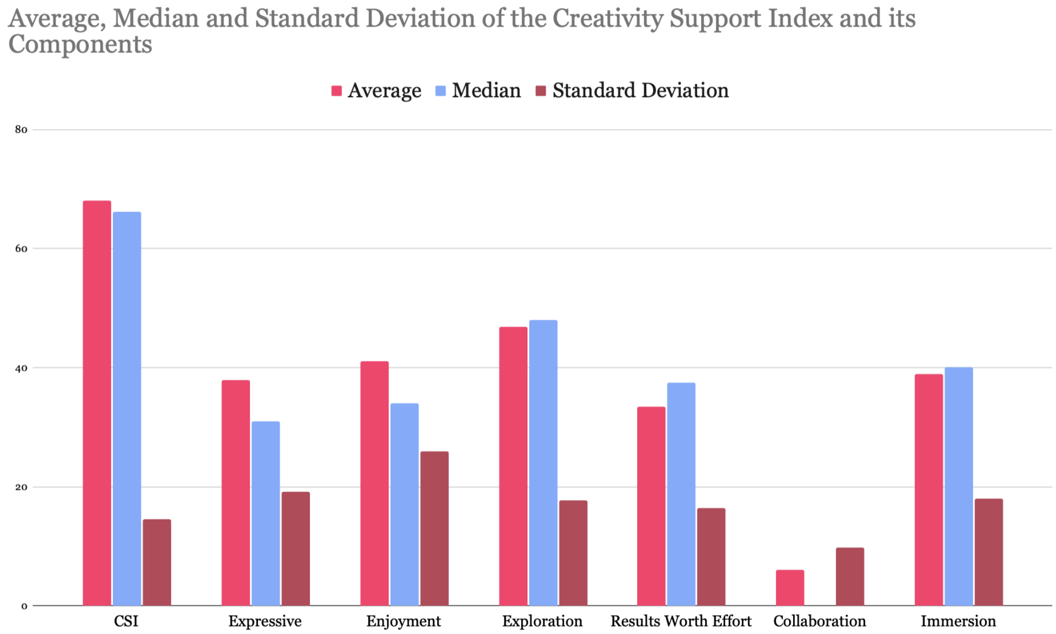

Figure 3.4: This chart breaks down the Creativity Support Index and its components' average, median and standard deviation. As expected, the highest scoring components were exploration, enjoyment, and immersion. They were followed closely by expressiveness. Results worth effort was the second lowest scoring component, and, as expected, collaboration was the lowest since SlimeArt was not designed for a collaborative context.

3.2.2 User Study Observations

During the usability studies we were able to identify the most confusing aspects of both the interface and the understanding of how SlimeArt's MCPM model worked. During the different tasks of the usability study, we were able to analyze a broad spectrum of design flaws.

3.2.2.1 Introductory Task

During the introductory task, users were walked through the initial functionality of the tool. In our first two studies, this included a list of questions asking the user to find and use the different UI components, but we discovered that these users did not get a good understanding of the most complicated aspect of the algorithm: how the particles behave when they emit deposit themselves versus when they don't emit deposit and find deposit from the deposit brush. After our initial testing and observational notes that these first two user studies did not receive the same introduction as the rest of them, we changed the introductory tasks to walk the user specifically through how to use the particle deposit strength, the view dropdown, the play/pause buttons, the deposit brush, the speed slider, the color picker, and the trace decay. The aspects of the UI that were left unexplained were the brush size and density, the field of view, and the visibility distance. Nine users asked about the field of view and three asked about the visibility distance because they had trouble understanding how these parameters were affecting the design. Four users requested that more information be provided about the settings. They never asked about the brush size and density, because these sliders' effects were apparent when the user chose to experiment with them. During the step-by-step introductory walk through we were able to observe clearly when users would get confused. One twice repeated mistake was a user making the deposit strength of the deposit brush zero instead of the agent deposit strength. If this mistake did not get corrected by the researcher, then the user had a very difficult time understanding the behavior of the particles for the rest of the study.

3.2.2.2 Exploratory Task

After the introductory task, users would enter the exploratory task and would be permitted to explore the tool without instruction for two minutes. During this task, it was common for users to ask questions about the components they didn't understand or ask for features they thought were missing like an eraser (3 participants), undo/redo functionality (3 participants), or layers (2 participants). During the exploratory task we could witness how the lack of an eraser or undo/redo would frustrate users since their only recourse to backtrack from a mistake would be to start over. It made it difficult for users to feel free to explore the tool since they would not fully understand how the action they would take would affect the design, and then they would not be able to easily correct that mistake to learn from it. One challenging aspect of the design for users was that under the particle settings heading, nearly all of the UI components did not affect particles globally, meaning, changing a slider, the speed for example, would not affect any particles already on the canvas. That new speed would only apply to newly created particles. Three users during the exploratory task revealed they believed changing the sliders would affect the particles on the canvas. This misunderstanding was likely due to the fact that the only global slider was the trace decay which affected the length of the trace the particles left behind them as they moved. When the trace decay slider is changed, it immediately would affect every particle on the canvas. There were two users who were surprised or frustrated when they noticed the trace decay affected the existing particles if they did understand that the other parameters (color, speed, field of view, and visibility distance) applied to only new particles.

3.2.2.3 Planning Task

After two minutes of exploring which had varying levels of engagement from different users, they would move into the planning task. During the planning task users were asked to describe a design, abstract or representational, to create with SlimeArt. Ten users described a representational design. Some examples were Saturn, a landscape, a river, and a chessboard. The few that described a less representational design based theirs in a network or line-based design such as modeling the Game of Life with "weird little propegaty grid things going on," "a dynamic movement...like a circulation of particles," and "a diamond shape, maybe a poseidon tri-stick."

3.2.2.4 Execution Task

During the execution task users were asked to attempt their planned design to the best of their ability. Many users either abandoned their design entirely or changed their plan, especially if they planned a more representational design that required more control over particles than a typical novice user of SlimeArt has. Users that abandoned their design to continue exploring and playing with the tool appeared to enjoy themselves more, as SlimeArt is more fun to use in an exploratory creative process rather than a planned one. Six users stuck with their original design and attempted with varying degrees of frustration and success to accomplish it. When users attempted subtasks for their design, we could see how they perceived the tool would behave. Since the introductory task had the researcher leading the actions of the participant and the exploratory task had little expectations for what would come out of it, the execution task applied more pressure to their understanding of the tool. Seven users became frustrated that their particles were clumping together instead of spreading out, that they would move and change the design too quickly, or that they did not find the deposit because they had their particle deposit strength set too high.

3.2.2.5 Debrief

During the debrief, users were asked if their design came out as they intended. Seven participants answered by avoiding the question and explained that they did not know the settings well enough or the tool would be better for designs that fit the tool better. When asked what aspects of the interface users found confusing, one frequent answer was about how the sliders affected the design and the particle view vs. trace view. When asked what people would change about SlimeArt, users suggested adding hover descriptions for what the sliders do, undo/redo, an eraser, clearing the deposit/trace views separately, providing a simple UI view with the option to view more advanced settings, transparency, layers, and saving/exporting the design.

3.3 Discussion and Redesign

After observing fifteen participants use the first version of SlimeArt, we were able to make an informed redesign to address the issues the users faced. Some users had a more positive experience of the first edition of SlimeArt if they were lucky enough

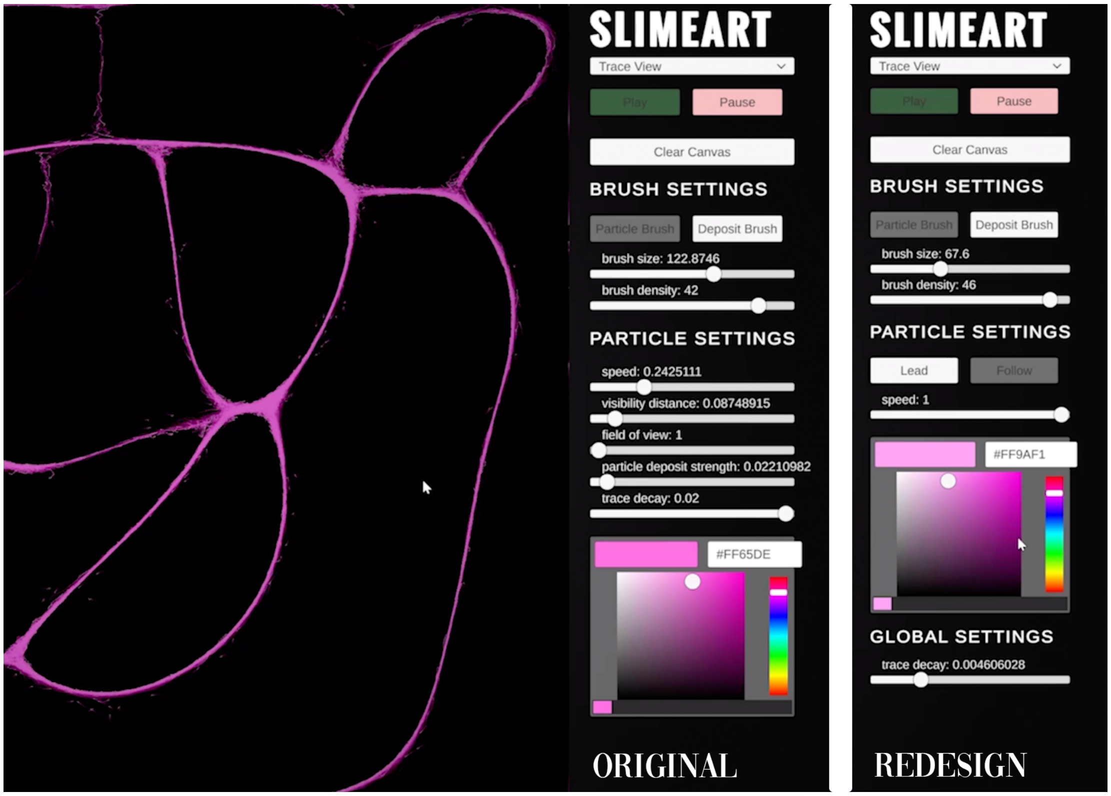

Figure 3.5: Here is a comparison between the original interface and the redesign of SlimeArt based on the results from the usability study. The most notable changes were the replacement of the particle deposit strength slider with the lead and follow buttons, the removal of the field of view and visibility distance sliders, and the adjustment of the trace decay to be in a separate "Global Settings" section.

to not accidentally put the sliders in a weird combination that gave them undesirable behavior. It was necessary to observe so many different SlimeArt experiences to get a sense for what mistakes a SlimeArt redesign could correct, so that more users had the experience of SlimeArt requiring less effort to create engaging network designs. The redesign of SlimeArt went beyond the basic functionality feature suggestions from the participants to solve the major design flaw in the presentation of the particle deposit strength. Before the user studies we did not predict how confusing this would be to users. Essentially, the behavior of the particles could be relatively binary if the deposit brush's strength was not high enough. This was common among the user tests, because the default value for the deposit brush was too low when the user opened the tool. When the strength of the deposit brush was so low, it appeared to have no effect on the behavior of particles when they were emitting deposit themselves and significant effect on the particles that did not emit deposit. Understanding how this binary was created was the most difficult concept for users to understand. If the user did end up increasing the deposit brush's strength, the particles behaved more as expected. If the deposit brush's strength was low, then the particles had two settings. If their particle deposit strength was zero, they would find the low strength deposit the user added with the deposit brush, but if they had any amount of particle deposit strength above zero, even as low as 0.001, the particles would follow each other and ignore the deposit. Users found this extremely frustrating because they were presented with the idea that the deposit brush would allow them to structure their designs, and when it did not behave as they expected, it was discouraging. In the redesign, the first change I made was to increase the default deposit strength. It became normalized and multiplied by the width of the screen for the user to make sure that it was always emitting deposit significantly stronger than the particles' deposit. See Figure 3.6.

While increasing the deposit strength value did mean that whether or not particles emitted deposit, they all were effected by the deposit brush, their behavior on their own was still distinctly different. The presentation of the initial user interface did not make clear the distinction between the two. The particle deposit strength was

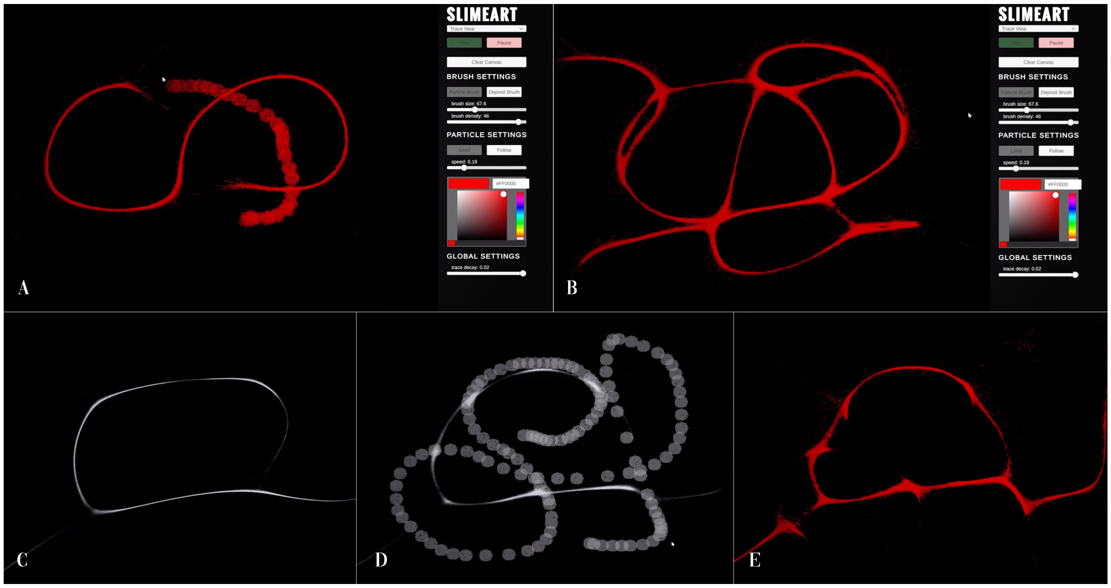

Figure 3.6: In frames A and B, the particles are "leading," also known as emitting particle deposit strength. Frame C shows the deposit they are emitting as they create their network. In the redesign, the deposit brush strength was increased so that in frames D and E there is a visible effect of the deposit brush on particles that emit their own deposit.

presented with a slider when, in reality, the behavior change was only visible between a particle deposit strength of zero and a non-zero value. To present the distinction between particles that create their own designs by emitting deposit strength and the particles that simply follow deposit without emitting it, I created two toggle buttons saying, "Lead" and "Follow." Instead of the particle deposit strength slider, users can now create particles that either lead a design or particles that follow a design. This redesign better presents the actual behavior the particles will have without the user needing to understand how the algorithm parameters make them behave. In addition to adding the lead/follow toggle buttons to the particle settings, I also removed the sliders that were most confusing to users: the visibility distance and the field of view.

In the redesigned interface the particle settings section only includes the lead/follow buttons, the speed color of the particles. Then, I moved the trace decay slider into its own section called “Global Settings.” By moving the trace decay into its own global category, it should help users expectations for how the trace decay applies to all of the particles globally and reduce the confusion that the particle settings are not applied globally. The particle settings are only applied to the particles at the moment they are created, and users were either surprised the trace decay was applied globally or confused why the particle settings were not applied globally. By separating this behavior, the interpretation will hopefully be clearer. This simpler interface will make it easier for users to understand how their interface choices affect the particles’ behavior on the canvas. By redesigning the particle deposit strength slider, increasing the deposit brush’s strength, removing confusing sliders, and adding a Global Settings section for the trace decay slider, SlimeArt would become a tool that is easier for users to explore to understand how their actions in the UI affect the behavior of the particles on the canvas.

Chapter 4

SlimeArt Encouragement Study

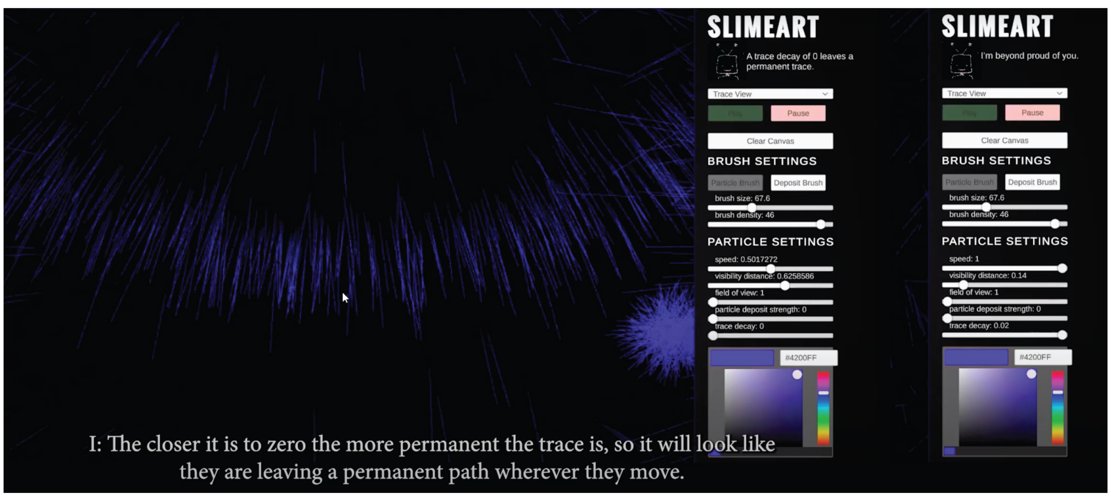

Figure 4.1: Left: SlimeBot in the top right offers a description of trace decay as the researcher relays the same description during the introductory task. Right: SlimeBot offers a message of generic encouragement to the user.

4.1 Method

4.1.1 Implementation

When the user launches SlimeArt with SlimeBot, on the top of the menu is a small virtual agent bot made using SlimeArt displaying the message: “Welcome to SlimeArt!” Since the user would find it off-putting if SlimeBot began generically complimenting their art when they haven’t started drawing yet, SlimeBot only displays the welcome message until the user begins drawing and turns on the drawing flag. Once the drawing flag is flipped, SlimeBot displays messages either from its unused generic messages list or from its unused UI messages dictionary. If the user changes or clicks on any UI components, the main script tells SlimeBot that that particular UI was changed. SlimeBot can then display one of the unused UI messages for that particular component from its dictionary if four seconds have passed since it last displayed a message. These UI messages include tips and funny sayings specific to the UI component triggered. For example, when a user hits the Pause button, SlimeBot could say “Nice! Pausing is helpful when planning a design,” or “I can’t wait to see what you design!” These specific UI messages were added to give SlimeBot more credibility. If SlimeBot has not said something new for a random amount of time between 10 to 15 seconds, he displays a random unused generic message from an internal list. These random messages provide generic encouragement like, “You’re really good at this!”, “What a great design!”, and “I love watching the particles.”

4.1.2 Evaluation

4.1.2.1 Creative Self-Efficacy

In order to evaluate if SlimeBot's encouragement for the user affected the creative self-efficacy of the user, we included two questions in the pre and post surveys in both the control and experimental conditions. The four questions included *future orientation, perceptions of confidence, key features or levels of task performance*, and *broader-ranging response scales* [11]. In the pre-survey, the two questions were

Q1 "On a scale from 0-100 (with 0 being the least confident), how confident are you that you will make a variety of generative art designs using the same tool?"

Q2 "On a scale from 0-100 (with 0 being the least confident), how confident are you that you will make dynamic, intricate designs when given a generative art tool?"

Then, in the post survey, the two creative self-efficacy questions could reference SlimeArt since the user now had the experience of using the tool.

Q1 "On a scale from 0-100 (with 0 being the least confident), how confident are you that you will make a variety of generative art designs using SlimeArt?"

Q2 "On a scale from 0-100 (with 0 being the least confident), how confident are you that you will make dynamic, intricate designs with SlimeArt?"

We were then able to calculate if there were any trends in the differences between how people answered the creative self-efficacy questions in the pre-survey versus the post-survey in either our control condition when SlimeArt does not include SlimeBot and in our experimental condition with SlimeBot present.

4.1.2.2 Perception of Ownership

To evaluate the perception of ownership participants had over their art they created with SlimeArt, we asked about their opinion of ownership over generative art in the pre-survey and then compared that answer to their perception of ownership over the work they created with SlimeArt in the post-survey.

Pre-survey "How much do you agree or disagree with this statement: 'Creating art with an autonomous system (generative art/AI art) still gives a person ownership over the art they create.'"

Post-survey "How much do you agree or disagree with this statement: 'I felt I had ownership over the art I was creating with SlimeArt.'"

4.1.2.3 Level of Exploration

When measuring the level of exploration we needed to break down the different tasks in the study. We define the level of exploration as how the user explored components of the UI. If a user tended to change very few parameters when exploring the tool, we considered this a low level of exploration. Studying the level of exploration was important in determining which aspects of the interface presented to the user were the most and least intriguing, common to change, and common to leave untouched. We wanted to analyze how often users paused the simulation to design their technique, switch between the different views, and used the deposit brush. We used click data saved in a csv file collected from the user to get the click data for the UI components.

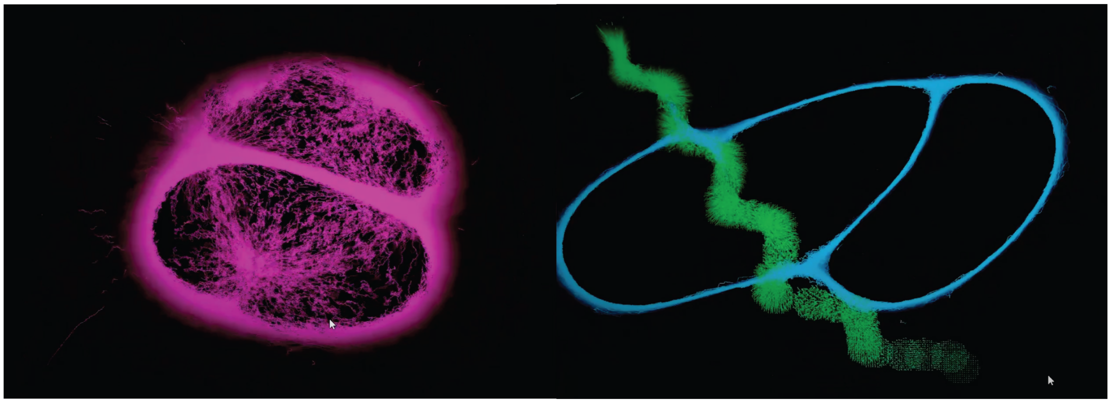

Figure 4.2: These are two examples of artwork created with SlimeArt by a participant.

4.2 Results

4.2.1 Level of Exploration

4.2.1.1 Creativity Support Index Exploration Component

Initially, the evaluation of exploration was focused on the click data from each user; however, in running our T Tests with all of the variables in the study we discovered that the only significant relationship between the experimental conditions was with the Creativity Support Index's exploration component (0.048). The exploration component of the CSI was a user-presented rating of how well SlimeArt supported exploration through a pairwise selection as well as a rating from 1-10. The mean of the control group was 41.09 with a median of 48 and a standard deviation of 14.11, while the mean of the experimental group was 59.60 with a median of 60 and a standard deviation of 19.00.

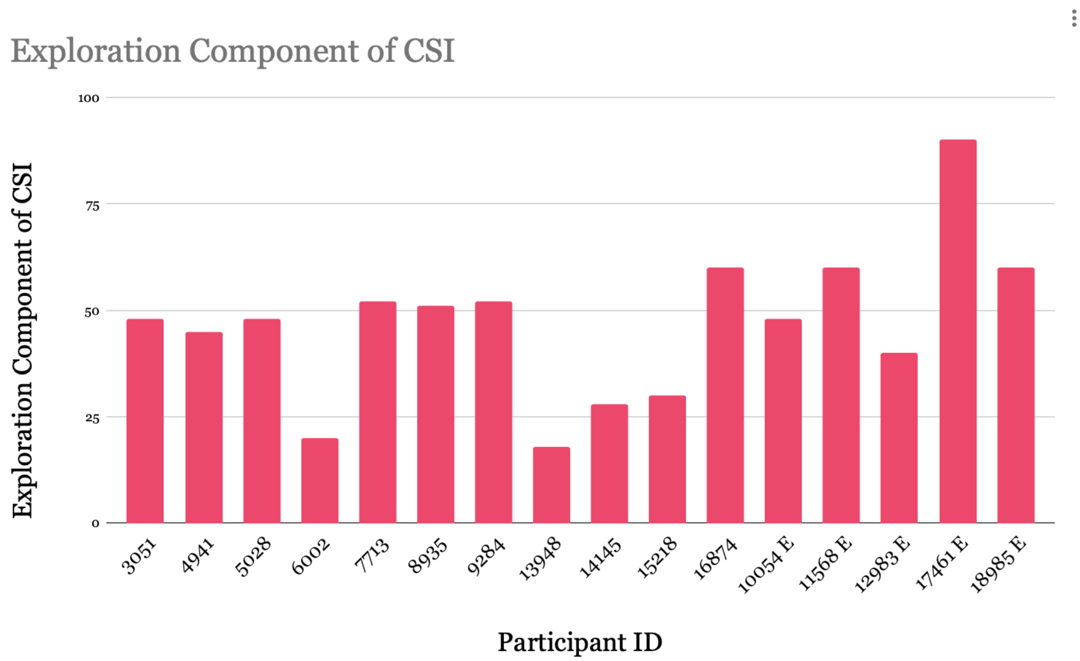

Figure 4.3: Users in the experimental condition with SlimeBot reported that SlimeArt supported exploration more than users in the control condition without SlimeBot. While the user-reported quality of SlimeArt's exploration was not the intended evaluation of how well SlimeBot increased user's exploration, it was the significant relationship between the condition and a variable (0.048). Users in the experimental condition have an E after their ID.

4.2.1.2 Click Data

We were unable to get click data from four participants in the control group and one participant in the experimental group, so our measurements using the click data include seven participants in the control group and four in the experimental group with SlimeBot. There were no significant relationships found between the condition the participant was in and their total time using the tool, their total number of interactions, or any of the specific UI components measured in the click data.

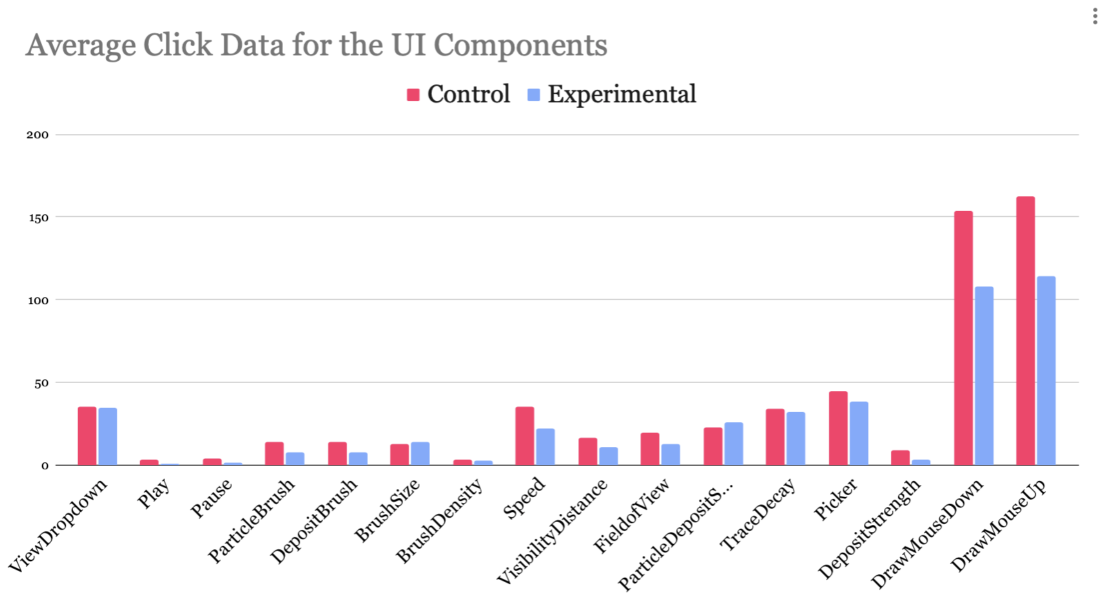

Figure 4.4: This table compares average click data for each UI component in the experimental and control conditions. The experimental condition had SlimeBot, and this data shows that on average, the experimental condition did not increase the number of interactions with the UI.

4.2.2 Creative Self-Efficacy

Ultimately, there was no significant relationship between users' creative self-efficacy and their experimental condition. I performed T Tests on the condition and post survey Q1 (0.77) and the condition and post survey Q2 (0.84). Additionally, I ran T Tests on the condition and the difference between the pre/post surveys for Q1 (0.76) and the condition and difference between the pre/post surveys for Q2 (0.23). The overall averages for pre-survey Q1, post-survey Q1, pre-survey Q2, and post-survey Q2 were, respectively: 58.50, 62.56, 53.44, and 57.88. The means for the control group's post survey Q1 was 61.46 and the experimental group's post survey Q1 mean was 65.00. For the post survey Q2, the control group's mean was 56.90 and the experimental group's was 60.00.

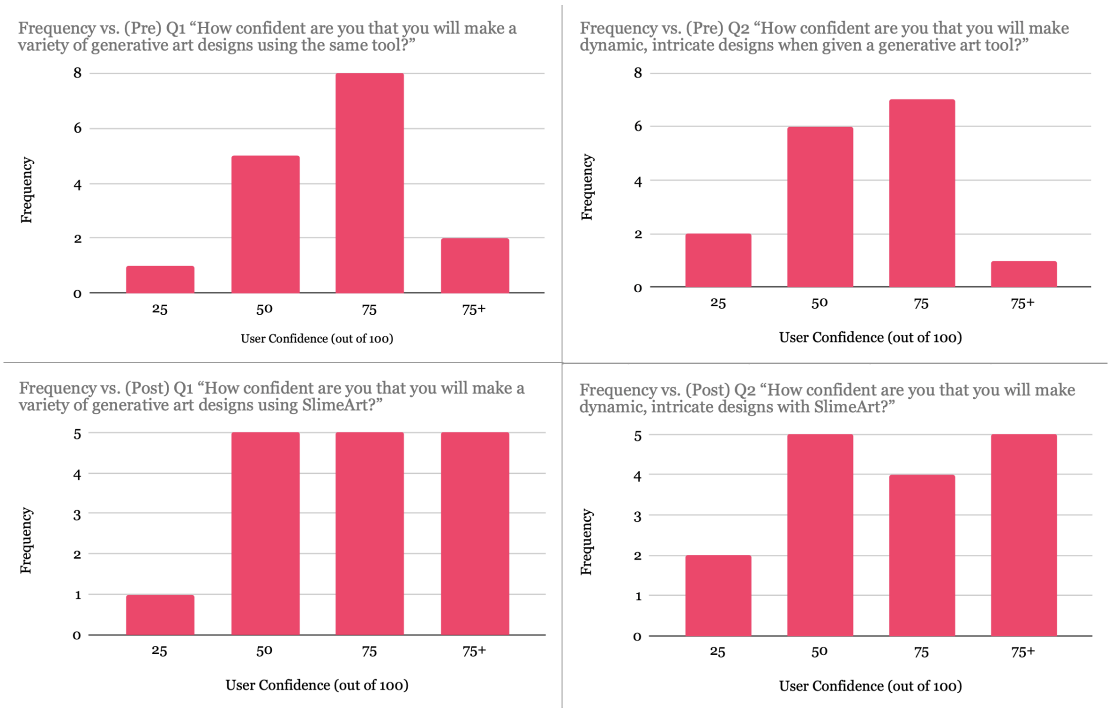

Figure 4.5: Participants' Creative Self-Efficacy was measured in both the pre and post surveys with two questions on a scale from 0-100. The first question asked the user if they were confident they would make a variety of designs with {PRE a generative art tool, POST SlimeArt}. The second question asked if the user was confident they will make dynamic, intricate designs with {PRE a generative art tool, POST SlimeArt}. The participant ID's with an E were in the experimental condition. There was no significant relationship between the Creative Self-Efficacy and whether or not the users were in the experimental condition. These charts so the frequency distribution of users' responses to these questions.

4.2.3 Perception of Ownership

The second dependent variable we were testing was the perception of ownership participants felt they had over the art they created with SlimeArt. We found no statistically significant relationship between their perception of ownership and the presence of SlimeBot. The mean, median, and standard deviation for users' perception

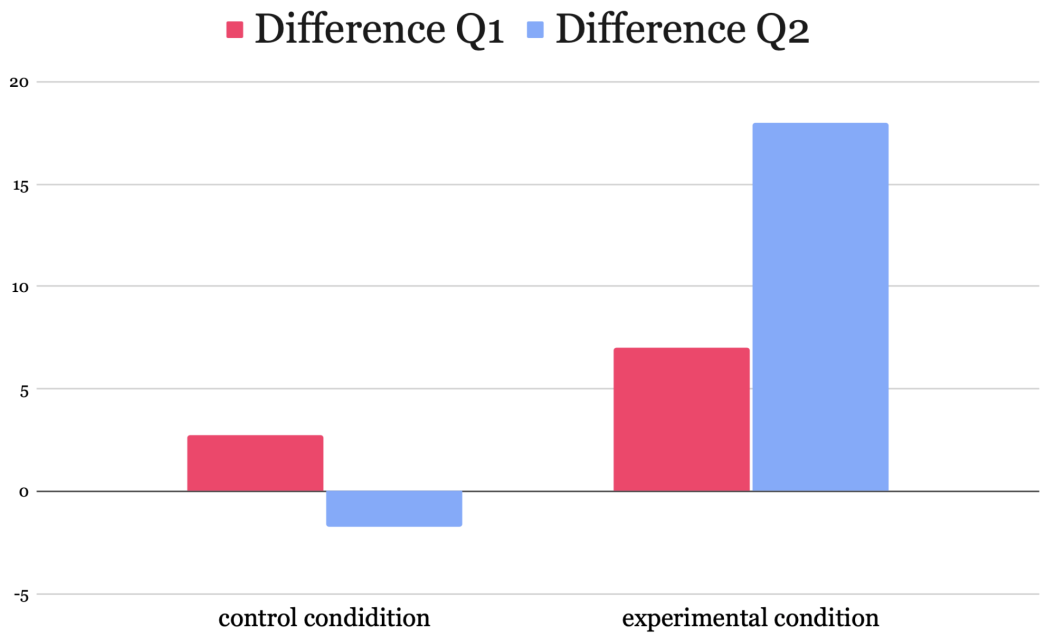

Figure 4.6: Since there were two questions in the pre and post survey to measure Creative Self Efficacy, this chart shows the average difference between the before and after score for Q1 and Q2 split between the conditions. There was no significant relationship between the Creative Self-Efficacy question differences and the condition.

of ownership in the control condition after using SlimeArt were 3.81, 4.00, and 0.87, respectively. The mean, median, and standard deviation for the experimental condition with SlimeBot were 3.80, 3.00, and 2.00.

4.3 Discussion

In the SlimeArt Encouragement Study, we investigated the effect of an encouraging virtual agent in a generative art tool on users' creative self-efficacy, their perception of ownership over their art, and their level of exploration. There were no significant relationships between the users' creative self-efficacy and their perception of

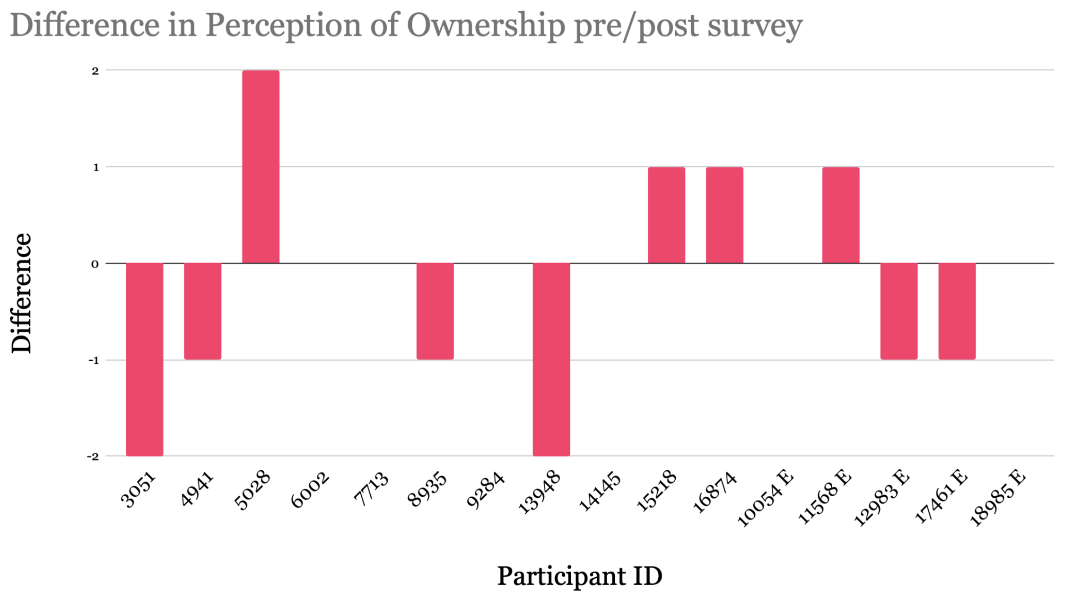

Figure 4.7: This table shows the difference between how participants initially felt about the ownership someone should feel over their generative art and how they ultimately felt about the art they created with SlimeArt. There was no significant difference between the condition with and without SlimeBot and the ownership participants felt over their art. Participants in the experimental condition have an E by their ID in the chart.

ownership and whether or not SlimeBot was present. The participants in the experimental condition primarily described SlimeBot as "cute," using phrases like "a cute little mascot" and "a cute little motivation bot." One participant said, "I like that it gives encouraging messaging and nothing really of too much content but just like wholesome and cute. I really like that." Given that no participant described SlimeBot as being intelligent, it is unsurprising that SlimeBot's presence did not have an effect on their creative self-efficacy or their perception of ownership. In future studies, SlimeBot could be made more intelligent so that users will attribute more credibility to SlimeBot and his comments. We found a statistically significant relationship between the study conditions and a variable was with the exploration component of the Creativity Support Index. Initially, the level of exploration was intended to be measured using the user's click data, but it turned out that the presence of SlimeBot had a significant effect on how high users reported SlimeArt supported exploration rather than how much the users explored the tool. Many users in the control condition requested that more information be provided about what each slider did, possibly with a hover over technique. My hypothesis for this effect SlimeBot had on users' reported exploration support is that SlimeBot partially provided this functionality. While adding a hover for more information is still important for increasing the accessibility of SlimeArt, SlimeBot still offered tool tips and explanations when users interacted with sliders. Further testing must be done to evaluate this hypothesis.

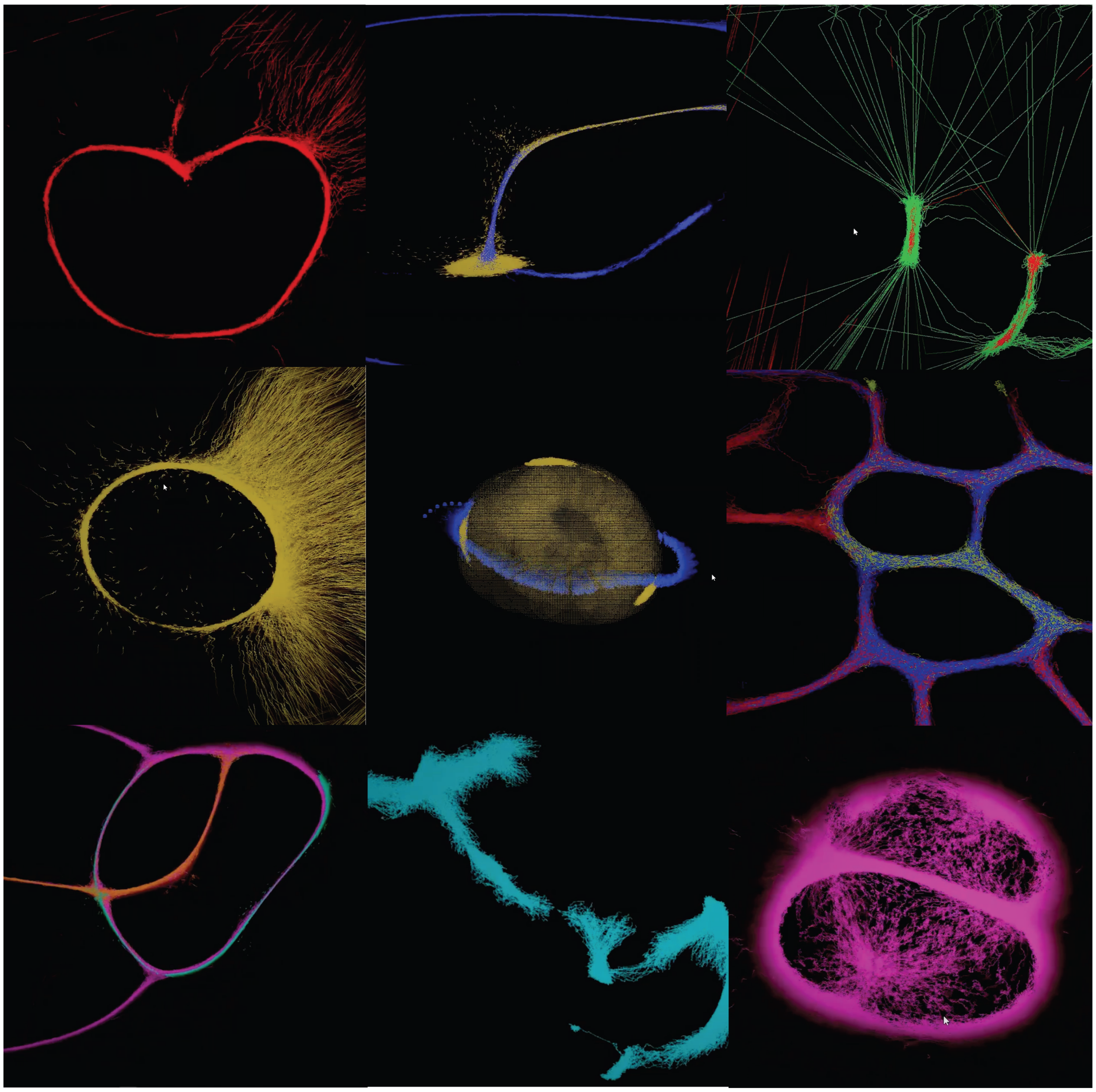

Figure 4.8: Here are nine examples of SlimeArt pieces created by participants during the study.

Chapter 5

Conclusion

As a Casual Creator, SlimeArt prioritizes users' enjoyment in exploring the parameter space the Monte Carlo Physarum Machine provides. Given the complexity of how particles behave when they emit deposit themselves compared to how they behave when they don't, SlimeArt's redesign will include replacing the particle deposit strength slider with a descriptive toggle button providing two types of categories particles could be. To clear up the confusion over particle attribute mutability, a particle library will allow for more modification and clarity about which settings belong to which particles. By adding a deposit heat map, users will have more information about the strength of the deposit they are applying to the canvas, which will help them understand their design. Additionally, by adding undo/redo functionality or an eraser, novice users will have a more forgivable environment for exploring SlimeArt. While SlimeBot's presence in SlimeArt did not affect users' creative self-efficacy, perception of ownership, or click data of exploration, it did make users report that SlimeArt supported exploration better than users did in the control condition. SlimeBot was able to provide explanations for sliders that participants in the control condition wanted to better understand the tool. SlimeArt is on its way to becoming an even more exploratory, immersive, enjoyable, and expressive tool, hopefully with even better emergent network design results that are worth the effort.

Bibliography

[1] Timothy W Bickmore and Rosalind W Picard. Establishing and maintaining long-term human-computer relationships. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 12(2):293–327, 2005.

[2] Margaret A Boden et al. *The creative mind: Myths and mechanisms*. Psychology Press, 2004.

[3] Vincenzo Bonifaci, Kurt Mehlhorn, and Girish Varma. Physarum can compute shortest paths. *Journal of theoretical biology*, 309:121–133, 2012.

[4] Joseph N Burchett, Oskar Elek, Nicolas Tejos, J Xavier Prochaska, Todd M Tripp, Rongmon Bordoloi, and Angus G Forbes. Revealing the dark threads of the cosmic web. *The Astrophysical Journal Letters*, 891(2):L35, 2020.

[5] Erin Cherry and Celine Latulipe. Quantifying the creativity support of digital tools through the creativity support index. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 21(4):1–25, 2014.

[6] Simon Colton, Geraint A Wiggins, et al. Computational creativity: The final frontier? In *Ecai*, volume 12, pages 21–26. Montpelier, 2012.

[7] K. Compton and M. Mateas. Casual Creators. In *ICCC*, 2015.

[8] Mihaly Csikszentmihalyi. *Beyond boredom and anxiety.* Jossey-Bass, 2000.

[9] Nicholas Davis, Holger Winnemöller, Mira Dontcheva, and Ellen Yi-Luen Do. Toward a cognitive theory of creativity support. In *Proceedings of the 9th ACM Conference on Creativity & Cognition*, pages 13–22, 2013.

[10] Oskar Elek, Joseph N Burchett, J Xavier Prochaska, and Angus G Forbes. Monte carlo physarum machine: An agent-based model for reconstructing complex 3d transport networks. In *Artificial Life Conference Proceedings*, pages 263–265. MIT Press, 2020.

[11] Steven M Farmer and Pamela Tierney. Considering creative self-efficacy: Its current state and ideas for future inquiry. In *The creative self*, pages 23–47. Elsevier, 2017.

[12] Juan Fasola and Maja J Mataric. Using socially assistive human–robot interaction to motivate physical exercise for older adults. *Proceedings of the IEEE*, 100(8):2512–2526, 2012.

[13] Philip Galanter. What is generative art? complexity theory as a context for art theory. In *In GA2003–6th Generative Art Conference*. Citeseer, 2003.

[14] Jennifer Jacobs, Joel Brandt, Radomír Mech, and Mitchel Resnick. Extending manual drawing practices with artist-centric programming tools. In *Proceedings of*

the 2018 CHI Conference on Human Factors in Computing Systems, pages 1–13, 2018.

[15] Jeff Jones. Characteristics of pattern formation and evolution in approximations of physarum transport networks. *Artificial life*, 16(2):127–153, 2010.

[16] Maciej Karwowski. Creative mindsets: Measurement, correlates, consequences. *Psychology of Aesthetics, Creativity, and the Arts*, 8(1):62, 2014.

[17] James C Kaufman and Ronald A Beghetto. Beyond big and little: The four c model of creativity. *Review of general psychology*, 13(1):1–12, 2009.

[18] David Kirsh and Paul Maglio. On distinguishing epistemic from pragmatic action. *Cognitive science*, 18(4):513–549, 1994.

[19] Antonios Liapis, Georgios N Yannakakis, Constantine Alexopoulos, and Phil Lopes. Can computers foster human users' creativity? theory and praxis of mixed-initiative co-creativity. 2016.

[20] Edwin A Locke. Self-efficacy: The exercise of control. *Personnel psychology*, 50(3):801, 1997.

[21] Alexander Maedche, Stefan Morana, Silvia Schacht, Dirk Werth, and Julian Krumeich. Advanced user assistance systems. *Business & Information Systems Engineering*, 58(5):367–370, 2016.

[22] Ben Shneiderman. 1.1 direct manipulation: a step beyond programming languages. *Sparks of innovation in human-computer interaction*, 17:1993, 1993.

[23] Ben Shneiderman. Creativity Support Tools: Accelerating Discovery and Innovation. *Commun. ACM*, 50(12):20–32, December 2007. Place: New York, NY, USA Publisher: Association for Computing Machinery.

[24] Fernando Silva-Coira, Alejandro Cortiñas, and Oscar Pedreira. Intelligent virtual assistant for gamified environments. In *PACIS*, page 193, 2016.