Designing Co-Creative AI for Virtual Environments

Josh Urban Davis
Dartmouth College, Hanover, NH, USA
josh.u.davis.gr@dartmouth.edu

Fraser Anderson
Autodesk Research, Toronto, ON, Canada
fraser.anderson@autodesk.com

Merten Stroetzel
Autodesk Research, Toronto, ON, Canada
merten.stroetzel@autodesk.com

Tovi Grossman
Autodesk Research, University of Toronto, Toronto, ON, Canada
tovi@dgp.toronto.edu

George Fitzmaurice
Autodesk Research, Toronto, ON, Canada
George.fitzmaurice@autodesk.com

Figure 1: Conceptual Illustration of the Calliope system featuring a collection of objects generated during design walkthrough

ABSTRACT

Co-creative AI tools provide a method of creative collaboration between a user and machine. One form of co-creative AI called generative design requires the user to input design parameters and wait substantial periods of time while the system computes design solutions. We explore this interaction dynamic by providing an embodied experience in VR. Calliope is a virtual reality (VR) system that enables users to explore and manipulate generative design solutions in real time. Calliope accounts for the typical idle times in the generative design process by using a virtual environment to encourage parallelized and embodied data-exploration and synthesis, while maintaining a tight human-in-the-loop collaboration with the underlying algorithms. In this paper we discuss design considerations informed by formative studies with generative designers and artists and provide design guidelines to aid others in the development of co-creative AI systems in virtual environments.

CCS CONCEPTS

• **Human-centered computing > Human computer interaction (HCI) > Interaction paradigms > Virtual reality**; • **Computing methodologies > Machine learning > Learning paradigms > Reinforcement learning > Adversarial learning**; • **Human-centered computing > Interaction design > Interaction design process and methods > Interface design prototyping**; • **Computing methodologies > Artificial intelligence**;

KEYWORDS

VR/AR/XR, human-ai collaboration, creativity support tools

ACM Reference Format:

Josh Urban Davis, Fraser Anderson, Merten Stroetzel, Tovi Grossman, and George Fitzmaurice. 2021. Designing Co-Creative AI for Virtual Environments. In *Creativity and Cognition (C&C '21), June 22, 23, 2021, Virtual Event, Italy.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3450741.3465260

C&C '21, June 22, 23, 2021, Virtual Event, Italy

ACM ISBN 978-1-4503-8376-9/21/06…$15.00
https://doi.org/10.1145/3450741.3465260

1 INTRODUCTION

Generative design [1, 3, 4, 24, 27] presents a promising opportunity for co-creative AI that leverages the computational power of design simulation, parametric design, optimization techniques and artificial intelligence, enabling computational tools to play an active, participatory role in the design process. Generally, this technique enables designers to explore a greater number of design solutions than traditional 3D modeling processes, or generate solutions that would be difficult or non-obvious for humans to create alone [9, 18, 25, 32, 33]. However, the typical workflow of these interfaces is antithetical to the mental model of traditional design tools. Often, generative design software obligates the user to parameterize the problem and solution space up-front and then requires the user to wait a substantial period of time while generating candidate solutions. Aside from being inefficient, this workflow is counter-intuitive to the mental-model of the creative process, which is iterative, extemporaneous, and playful, often employing a divergent problem-solving process, as opposed to the convergent process of generative design. Furthermore, by encouraging the user to specify all constraints of their problem up-front, exploratory and serendipitous browsing are eliminated as potential interaction methods, thus resulting in the user primitively eliminating many potential and inspiring solutions from their design space before fully exploring the problem [13]. While prior work has explored embodying the design process to locate the desired design solution (e.g. get the right design) [29, 34, 35] our work embodies the design exploration process in order to create the right design [22].

Prior work in this domain has largely focused on 2D interfaces for procuring design solutions. DreamSketch, for example, supported greater design creativity in generative design workflows by enabling a user to input design constraints via sketching [36]. However, this workflow may not be optimal when working in 3D, since traditional generative design systems for 3D objects rely on 2D graphic interfaces [18, 32, 33]. Virtual Reality (VR) offers a promising platform that allows designers to sculpt and model 3D objects using methods more attune to traditional sculpting tools, as well as supporting embodied exploration of data. Furthermore, VR offers a more engaging experience, which can enrich the synthesis experience as well as mitigate inactivity due to algorithmic idle times. However, this modality has largely gone unexplored for generative design.

In this paper, we present Calliope (named for the Grecian muse of architecture and epic poetry) an interactive model synthesis and exploration tool that helps designers explore divergent solutions to a given design problem within a virtual environment. The system allows users to inspect the visual appearance of geometric models, directly edit their geometry, and receive design input from a generative adversarial neural network (GAN). By leveraging the spatial exploratory potential of virtual reality, our system keeps the user engaged in synthesizing and evaluating other geometric models while the GAN computes potential solutions to the designer's problem. This approach facilitates an iterative design process, while encouraging a collaborative dialogue between the machine and designer. Similar ideas have been explored in the realm of literature by Borges [5], however, Calliope comprises the first human-AI creative collaboration interface for VR. The contributions of this work include: (1) an exploration of the design considerations implicit in creating VR interfaces for generative design; (2) A series of interaction techniques to allowing designers to create, combine, and explore divergent solutions to a given design problem within a virtual environment, working in close parallel with the underlying algorithms. (3) a proof-of-concept system, Calliope, which enables designers to work closely with underlying generative algorithms to creative exploration within an engaging VR environment; (4) insights for designing human-AI creative collaboration interfaces in VR extracted from multiple design validation sessions.

2 RELATED WORK

This work builds upon creative authoring platforms in virtual reality, co-creative and mixed-initiative design, as well as rapid-ideation and collaborative AI interfaces.

2.1 Creativity Support in VR

Prior work in VR literature investigates the potential for immersive environments supporting 3D design tasks such as sketching and modeling [19, 29, 37]. More recent inquiries explore the mechanics of sketching and modeling in mid-air, emphasizing the integrity and quality of the input gestures [30, 34] or suggesting improvements and corrections to the design [2], as well as enabling more advanced workflows. Similarly, DreamRooms, allows a user to interact with a procedural system to develop 3D room layouts for VR [35]. Many initiatives exist in mixed/augmented (MR/AR) reality as well. MixFab, for example, allows users to 3D model objects for fabrication using an AR gestural approach [28]. Most closely related to our interests is Mix&Match, which enables users to sample and view artifacts from Thingiverse in AR within the context of their physical environment [21]. Our work distinguishes itself from these in that it allows user to work directly with generative design algorithms in a collaborative process and enables creative exploration earlier in the design process to find the right design.

2.2 Co-Creative AI and Generative Design

User-driven generative design tools enable a user to specify high-level design intents, and systematically produce candidate design solutions using generative algorithms. Previous systems have applied this approach to optimizing office building layouts [40], fabricating airplane partitions [38], and designing furniture [39]. The pipeline in these systems automate parts of the design process by enabling users to specify abstract design goals and high-level constraints to the system which are used to produce candidate design solutions [8]. Tightly-looped interactivity between the user and the underlying algorithms in these systems is often difficult for a variety of reasons, such as the high latency of the algorithms themselves. Typically, after specifying high-level design goals, the designer must wait hours or days before viewing the results of the algorithmic generation process. Previous efforts have attempted to address this issue such as eifForm, a generative design tool combining structural model generations with a traditional modeling approach [20]. Martinez et al. enables designers to input sample patterns and obtain a design optimized for structural integrity and aesthetic similarity to the input [17]. While promising, these prior works do not permit users to directly manipulate system output and provide feedback on desired solutions, nor keep the user engaged while the system produces candidate designs. Furthermore, even after waiting substantial time to view results, the designer then has to manually sort through all the design candidates generated by the algorithm. While these algorithms are effective at producing a multitude of solutions, sorting through these solutions for ideal designs still requires substantial work on behalf of the designer. To account for these limitations, recent work has investigated generative design interfaces which explore and visualize large sets of data and design solutions [41], as well as systems which interprets sketches created by the user as problem representations [8, 36]. Most closely related to our work is Forte, which allows a user to iteratively design 3D printable objects using interaction techniques designed for topology optimization [8]. Early explorations of embodying the collaboration between human and machine seem promising [47–49] but have yet to extend this domain of inquiry to the embodied system of VR.

Our work distinguishes itself from these predecessors by leveraging the spatial properties of VR to enable users to interactively explore design artifacts and specify desired design features to the underlying algorithms. Calliope enables designers to interact with the generative process in a tightly looped human-ai interaction by encouraging users to explore candidate designs in parallel. Additionally, Calliope employs the interactive modalities of VR to encourage embodied exploration and synthesis.

2.3 Collaborative AI Interfaces

Mixed-initiative interfaces support collaboration between an intelligent system and a designer, allowing both the user and the intelligent agent to "do what they do best" by assuming different, but complementary roles in the design process [11]. Within the scope of design tools, user-driven suggestive interfaces are mixed-initiative interfaces which enables users to control and drive the design task. While the user controls this process, the intelligent system observes, analyzes and suggests improvements or alternatives to improve the design. An early work in this field is Chateau, a 3D sketching tool which predicts what a designer will draw next and suggests alternative completions to the drawing based on observations of user input [42]. Other examples include the work of Umetani et al. who presented a furniture design tool which indicates unstable structures while a user edits their design, and offers alternative suggestions for solving these instability problems [25]. Additionally, Tsang et al. developed a system which takes in target design images as input, generates 3D curves, observes users' input strokes, and suggests relevant geometry accordingly [23]. In contrast to this previous research, Calliope enables designers to perform direct mesh manipulation on generated 3D artifacts in a virtual environment and provides design candidates based on features specified this way. Further, Calliope leverages a generative approach to explore a multitude of design candidates within a VR environment.

3 FORMATIVE INTERVIEWS

To better understand the role that embodied experience could play in dealing with delays, maintaining engagement, and exploring design variations we conducted a series of semi-structured interviews with end-users of generative design applications.

3.1 Participants and Methodology

We recruited 6 participants (3 female identifying, 3 male identifying) between ages of 22 and 43 for a remote semi-structured interview. Participants were familiar with generative design through their creative practice or professional development experience. 5 participants were familiar with VR/AR in some capacity and 2 of these had VR development experience. The interview consisted of demographic questions and 11 open-ended interview questions discussing prior generative design, human-AI collaboration, and experiences designing in VR/AR as well as their overall experience with these mediums.

3.2 Design Considerations and Goals

Guided by prior literature and the results of our formative interviews, we performed an iterative qualitative analysis of the transcribed participant interviews informed by prior ground-theory research best-practices [21]. Following the interviews, an author coded the transcribed data for need-finding design considerations and participant expectations for interaction with such a system. From this, we extracted a series of design goals and considerations for our system outlined below.

3.2.1 Tightly-Looped Human-AI Collaboration (D1). The traditional workflow of generative design is a linear, solution driven process that requires the designer to input design requirements up-front, and then wait substantial periods of time to generate solutions. These solutions then must be manually explored which is an additional time-consuming and tedious process. P6: "It can be tedious. . .you want to do a thorough job and not miss anything cool but you also get tired after looking at [so many] things". Any generative design system should expedite this process, allowing a user to efficiently explore many design decisions, and support a creative dialogue between the user and the algorithm. This allows a user to freely explore and test as many design options as possible, assisting in getting "the design right and the right design" [22, 43].

3.2.2 System Transparency (D2). All of our participants discussed an underlying uncertainty with underlying generative algorithms. P3: "It can be exciting to not know what the machine will dream-up. . .but it can also be frustrating. . .I sometimes feel the machine knows more about [my design] than me." Furthermore, many of our participants mentioned potential distrust when using such systems P1: "I've crashed [the generative system] so many times I've lost count. . .I never understood what decisions will break the software." Evidence suggests that explainability of the underlying AI system is crucial for fostering trust with users [10, 26]. We believe that an interface for these systems should mitigate these concerns by providing as much transparency as possible.

3.2.3 Creativity vs Creation (D3). When discussing current limitations to creativity in VR, we found a distinction between creation (the physical acts of creating an object) and creativity (scaling the act of creation to allow exploration of many ideas). Many participants indicated that they believed quality of ideas came from quantity, and thus creativity encouraged the generation of as many design ideas as possible, sorted and explored to find quality candidates. P2: "My old boss used to call it the 'blah blah blah gold' philosophy. You have to create a lot of blah to get gold". However,

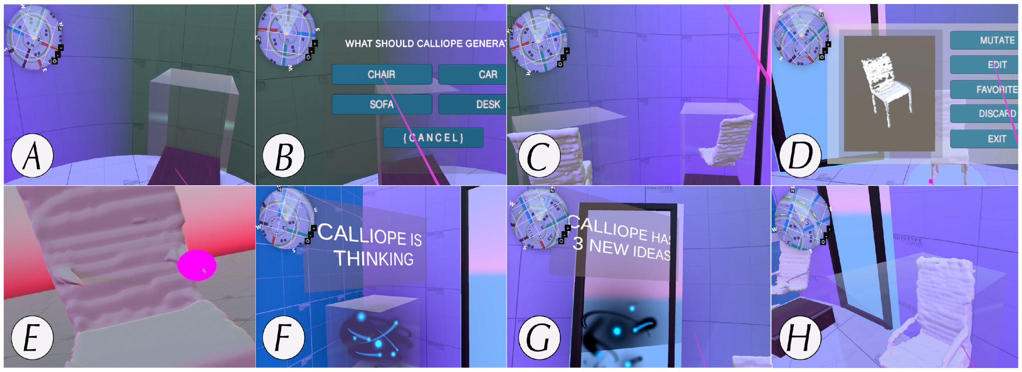

Figure 2: Interaction Overview of the Calliope System: A. An empty vitrine; B. Object generation menu; C. Filled vitrines with genreated objects; D: Editing menu; E: Mesh editing; F: Progress lights signal mesh manipulation interpretation; G: Computation complete; H: Results of mesh manipulation interploation displaying in preceeding room.

current VR sculpting practices are limited by the difficulty of creation in VR. P5: "It just takes forever, and by the time I'm halfway done [sculpting] something real basic I start to feel a little sick". Thus, we believe it is imperative that generative systems support rapid creation and ideation to support creativity within VR.

3.2.4 Leverage Spatial Interaction for Engagement (D4). One common obstacle to tightly-looped human-AI interaction involves the high-latency computation cost, and thus the resulting speed inhibition, of generative design [8]. P1: "I'll send samples off for processing which can sometimes take hours and the results that come back are garbage." We believe a system that utilizes VR must leverage the engaging properties of the virtual modality to accommodate for this limitation. Furthermore, many generative design systems are intended to create 3D objects, yet rely on 2D means of interaction, which is counter-intuitive for users. P6: "Sometimes while modeling I get frustrated looking for the right tool. . . all I want to do is reach into the screen and pinch sculpt the mesh". The spatial nature of VR not only provides affordances to keep a user engaged during computation idle times, but also supports a more intuitive process for 3D modeling [44]. Thus, any such system should aim to keep the user engaged exploring and synthesizing additional design solutions while the system computes solutions in parallel.

4 CALLIOPE

To account for the above design considerations, we developed Calliope: a user-driven interface for tight-looped human-AI creative collaboration within a virtual environment [Figure 2]. While system development was largely guided by our formative interviews and literature review, we acknowledge that this is only one possible instantiation of the higher-level concepts and design goals. This system can be considered an initial exploration to guide future development of co-creative AI in VR.

4.1 Virtual Environment

The structure of our virtual environment drew inspiration from radial data visualization and is progressively populated as it is explored by the user. Given that the designs produced by Calliope lend themselves to organization that is both hierarchical and categorical, the results are well suited for representation in a pseudo radial data visualization [15]. Furthermore, this representation translates easily to VR, and mimics the structure of the labyrinth from Borges' *Las Ruinas Circulares,* which helped inspired this work [5]. Users are required to develop a certain number of objects within each room before additional rooms are spawned. This was designed to encourage

the user to explore additional design options before continuing (D3). The placement of the room is dependent upon the kind of modification and generation performed by the user in their current room. This methodology also allows the user to construct a visual representation of their generated artifact's lineage as they proceed through the system. The rooms themselves are color-coded in order to provide the user a sense of variation in room design, assist with a user's memory of the environment, as well as further support the visualization of their design process [15]. Each room, once spawned, is populated with a number of vitrines determined by the actions in the previous room, which serve as the central locus for providing input and receiving output from Calliope [Figure 5]. Vitrines were chosen as an interactive locus in order to evoke a metaphor between the synthesis of data and the creation of a museum or gallery exhibition. This way, the user develops an exhibition of iteratively designed artifacts while procedurally constructing a visualization of their design interests.

5 SYSTEM WALKTHROUGH: DESIGNING A CHAIR

Here, we provide a walkthrough demonstrating the design process using our human-AI collaborative system. In this example, the designer would like to create a custom chair for a client that meets specifications such as height, material, and presence of certain features such as armrests and head rests.

5.1 Initializing Objects

The structure of the virtual environment drew inspiration from radial data visualization and is progressively populated as it is explored by the user [Figure 3]. The designer begins by selecting an empty vitrine [Figure 5] to begin the generation process. A menu appears with several options for generating an object [Figure 2B]. The user can select "sculpt" to sculpt a new mesh from scratch using a brush-like interaction or select "generate". By selecting generate, the generation menu appears with a variety of object classes from which the user can select. The user selects "chair", and a chair appears inside the previously selected empty vitrine [Figure 2C]. The user then proceeds to repeat this process with each of the vitrines in the room. Once a user has filled all the empty vitrines in the current room, a new room is spawned, containing new vitrine corresponding to the operation performed in the previous room. Each new room will contain a minimum of 2 empty vitrines to allow generation of new artifacts. If the user performed direct mesh-manipulation in the previous room, 4 additional vitrines will be spawned in the new room containing the results of mesh-manipulation interpretation. Similarly, if the user performed mutation in the previous room, 2 additional vitrines will be spawned in the new room containing the results of the mutation.

5.2 Direct Mesh Manipulation

The designer selects an artifact containing visual features which they find interesting, but wishes they had other specific characteristics such as armrests (D1). To communicate this to Calliope, the designer selects the target object, and then chooses "edit" from the menu [Figure 2E]. The designer then uses the direct mesh editing raycasting tools to quickly sculpt an impression of armrests on their target object. These techniques are performed using a traditional ray-casting point-and-click interaction common to other sculpting platforms such as TiltBrush [45]. Once the user has completed their mesh manipulation, a progress light appears, indicating that Calliope is performing a computational operation. Calliope will take the user's manipulated object as input, and attempt to create similar designs to the user's custom input. During this time, a user may venture to other rooms and conduct parallel design processes on other meshes or generate new objects. When Calliope has completed the computation, results are displayed to the user in the adjoining room.

5.3 Mutation

The user favors certain characteristics of one artifact, and certain characteristics of a different artifact (D2). Wishing to combine the two artifacts in order to render a single object containing a mix of the favored features, the designer uses the mutate function of Calliope. The designer selects a generated object inside a vitrine and selects "Mutate" from the menu [Figure 4]. When "Mutate" is selected, the user is prompted to select how many sampled objects they wish Calliope to generate (D3). The upper bound of generated objects was determined through our stress test in Section 7.3 to not overwhelm the system and induce frame skipping. The user indicates they desire 4 objects, and the walls of the environment are then lowered to display all objects generated during the session, as well as the 4 randomly sampled inspiration objects generated by Calliope. The designer then selects the second object containing favorable features, and the environment walls are raised. A progress light displays inside the selected vitrine. After completing

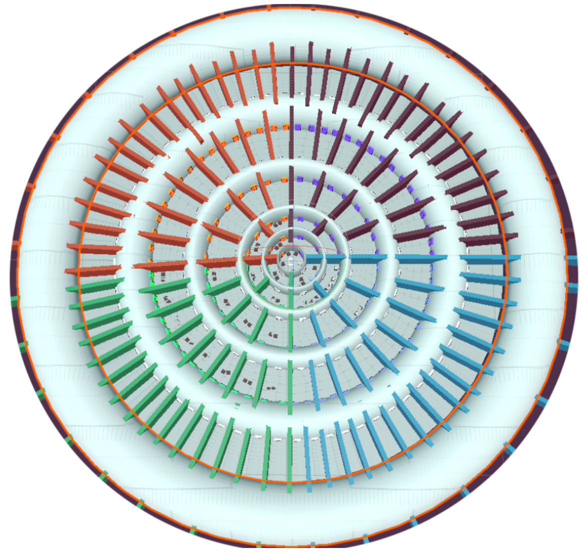

Figure 3: The Calliope Virtual Enviornment Design Was Informed By Radial Data Visualization.

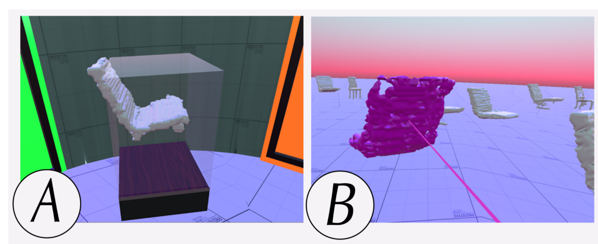

Figure 4: Micro vs Macro Creation Viewing; A. User views a single design; B. User lowers walls to view all designs from this session.

the requested interpolation, Calliope displays 4 mutated objects in an adjoining room [Figure 2H].

5.4 Parallel Design ideation

Rather than wait for Calliope to complete the computation, the user instead turns their attention to other empty vitrines inside the virtual environment (D4). They select the nearest door, and proceed to explore other rooms, selecting empty vitrines to generate new artifacts for consideration. Some of these artifacts contain desirable physical features which the designer favorites from the menu, saving these objects for future mutations. A light appears on the mini-map, indicating that Calliope has completed its computation. The user selects the newly generated room from the mini-map, teleporting the user to the room.

5.5 View Results

The user examines each result generated by Calliope during the mutation process. They decide to discard one result and favorite the other. The designer can then continue the design iteration process by generating and manipulating additional objects. After

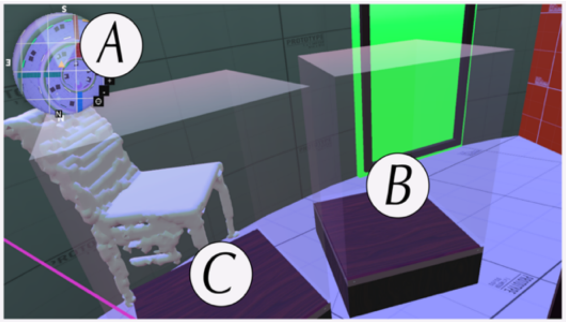

Figure 5: Overview of the Calliope User Interface: A. Mini-map with favorited object highlighted; B. Vitrine prior to object generation; C. Vitrine containing generated object

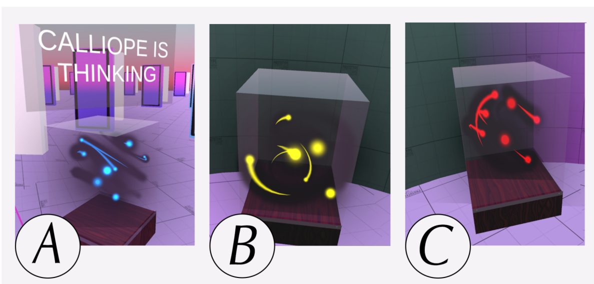

Figure 6: Color-Coded System Indication

several iterations of generation, manipulation and interpolation, the designer returns to the central room and lowers the walls of the environment in order to survey all the objects created during the current session. During this time, the designer favorites their final design candidates which are saved to the working system as obj files for future use, fabrication, and reference. The design can be seen in Figure 8A.

5.6 Additional Interaction Techniques

In order to support creative collaboration between the designer and the generative algorithm within the virtual environment, we developed the following additional features and interaction techniques.

5.6.1 Micro vs Macro Creation Viewing. In order to give the user a macro-view of all objects created, the walls of the environment can be lowered at any time, allowing the designer to view all objects they have created during the session (D3). The walls can then be raised in order to give the user a micro-view of the objects in their current room, allowing the designer to focus on current objects they are designing [Figure 4].

5.6.2 Color-Coded System Indication. While Calliope is performing a given computation (either generation or mutation) we provide feedback on general progress as well as system status to the user in the form of a color-coded light system. Once the user has specified a given function for Calliope to perform, the vitrine is filled with a particle effect light, which glows brighter and moves quicker to indicate the progress of the given computation. These lights are also color coded to reflect the current GPU-memory consumption, since this is an adequate indicator of computational cost demanded of the system. If the GPU has currently consumed less than 40% of volatile memory, the lights remain blue, 40%-60% turns these lights yellow, and above 60% turns these lights red. These values were chosen because 60% and above memory consumption may affect system performance. This color-coded system subtly informs the user of system status and allows them to make informed decisions with subsequent interactions (D2).

5.6.3 Mini-Map and Teleportation. In order to assist the user in navigating the system, a mini-map presents the user with a top-down view of the virtual environment, updated as the user progresses through the procedurally generated environment [Figure 5A]. In addition, the mini-map informs the user of new rooms to explore, status of current Calliope computations, and location of favorited designs (D4). The user can also teleport to a specific room within the environment by clicking on a room within the mini-map.

6 EXAMPLE OBJECTS

We present the following sample design task walkthroughs in order to demonstrate the creative potential of Calliope. Our walkthrough tasks were completed by members of the research team and chosen in order to demonstrate the variability of design approaches supported by Calliope, as well as the application of our design considerations mentioned previously.

6.1 Car Chassis

Similar to the described workflow, Calliope was used in the design of a car body which exhibited a series of unexpected visual characteristics [Figure 8B]. In this design session, the designer used a "breadth-first" design approach, asking Calliope to generate many sample designs. This process occupied half of the virtual environment, with one section dedicated to generating car designs, and another dedicated to generating other vehicle designs as inspiration. The designer then selects their favorite designs containing interesting visual characteristics and allocated an entire section of the virtual environment to interpolating these favorited objects with each other. Throughout this process, the designer didn't feel the need to edit any of the objects directly, instead relying on Calliope to sample candidate designs and then mutate designs containing favorable visual properties.

6.2 Novel Furniture Form-Factor: Sofa-Desk

In this workflow, a user experiments with inter-class interpolation to explore candidate designs of novel furniture form-factors. The designer generates and discards many sample design candidates for sofas in a single vitrine until they find one exhibiting desirable qualities. In the empty vitrine next to it, the designer similarly generates and discards many sample candidates of desks.

After identifying two ideal candidates from two different classes, the designer mutates these two objects into the adjoining room. While the designer waits for this mutation to occur, they turn their attention to generating sample candidates in a different room, repeating the above process with different object classes. The designer repeats this process with many classes until they have mutated several quality candidates exhibiting desirable characteristics and

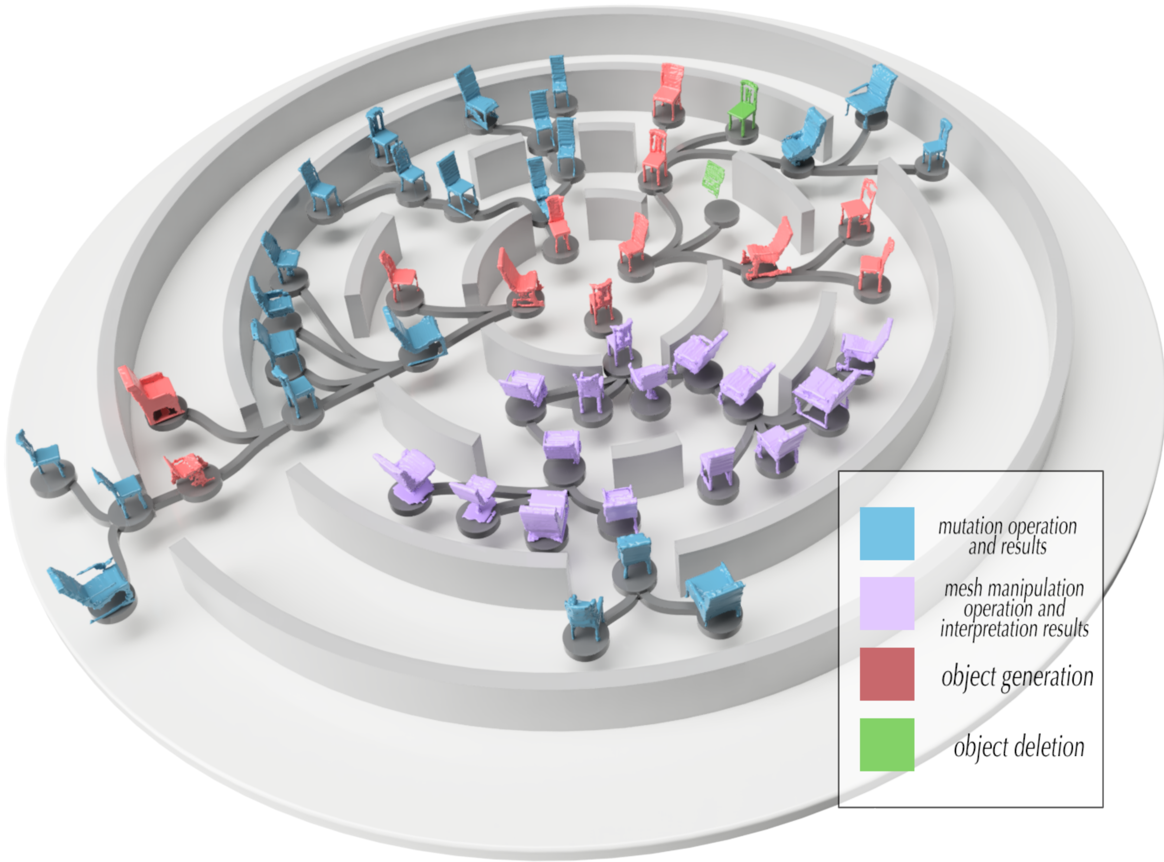

Figure 7: Results of Design Session Indicating Breadth and Depth of User Creations

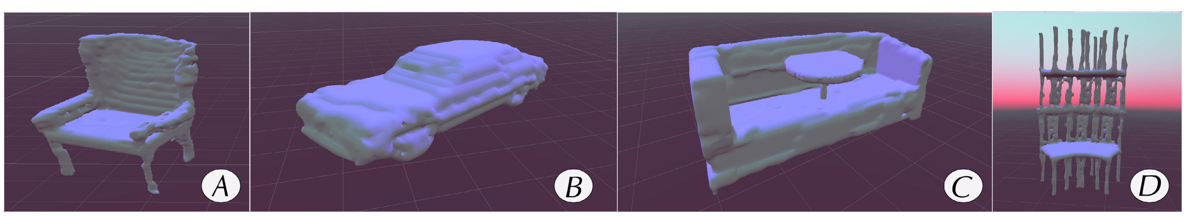

Figure 8: Example Objects Generated Using Calliope: A. Custom chair; B. Car Body; C. Sofa Desk; D. Artwork

novel form factors. The designer then performs mesh manipulation on these final candidates, in order to accentuate various qualities of the inter-class mutated candidates. This allows the designer to fix any immediate structural flaws with the design candidate, such as missing legs or incomplete surfaces. Afterwards, the designer favorites their desired candidates for export as obj files [Figure 8C].

6.3 Sculpture

This session was significantly more unconstrained because the user embraced a highly experimental workflow. During this session, the user followed no discernable structure to their design process, instead choosing to follow their intuition completely. The user generates objects of various design classes and mutates them with each-other, performing mesh manipulation out of aesthetic experiment to "see what happens" more than functional directing the generative algorithm. Often, the user's decisions are motivated more by a curiosity of the algorithms reaction to their decision, than any functional quality of the objects generated as a result of these decisions. The session concluded when the user arrived at an experimental result they deemed interesting, favoriting the result for export for possible fabrication [Figure 8D].

6.4 Video Game Scenery Suite

Generative systems often provide a narrow selection of solutions and produce a single design candidate. This is not ideal should a user require a collection of similar aesthetic objects. In this session, a designer wants to create an entire suite of scenery for a 3D video game. They allocate separate sections of the virtual environment for each type of scenery object they are modeling. After generating several samples of each individual class of objects, the designer selects favorites objects containing their desired visual aesthetics. The designer then mutates each favorited object of a given class with the objects favorited from other classes. This ensures that the final design candidates will contain visual characteristics similar to each other across different object class types. The designer then repeats this process with each remaining class of object until they have created a collection of objects bearing similar aesthetic characteristics.

7 IMPLEMENTATION

In order to ensure replication of our system, we provide the following details on implementation and technical contribution. Our virtual environment was developed in Unity 2019.3.0f6 using OpenGL 4.5 on Ubuntu 18.04, and tested on Vive Pro VR headset. Our generative algorithm was tested on Pytorch 1.5 with CUDA 10.2 using python 3.8-dev on an NVIDIA 1080Ti GPU. Details on our generative algorithm are outlined below.

7.1 3D Generative Adversarial Network Terminology

Generative Adversarial Networks are appealing designers due to their promising ability to persistently generate novel objects [31]. Goodfellow et al. proposed the Generative Adversarial Network (GAN) which comprised two networks: a generator and a discriminator. The generator network synthesizes convincing objects in order to fool the discriminator [12]. Meanwhile, the discriminator attempts to distinguish between ground-truth objects (3D models taken from ShapeNet [7]) and objects synthesized by the generator. Training consists of the generator learning how to create 3D objects by adjusting weights corresponding to object features. Once trained, the resulting generator is able to produce, and interpolate between, the selected domain of objects taken from ShapeNet. We follow the architecture described by Wu et al. to produce a generator which creates a representation of a 3D object by randomly sampling a z vector from a probabilistic latent space. This 200-dimensioinal latent z vector maps to a $64 \times 64 \times 64$ voxel cube, representing an object in 3D voxel space. The probabilistic latent space, in this case, refers to the solution space of possible objects generated by the system. Therefore, each z vector sampled from the latent space represents a novel object resulting from an interpolation of the 200 dimensions of the latent space. Each dimension of the latent space, in this case, represent a different geometrical aspect of the object. Generated objects are then rendered from voxel arrays to meshes using Marching Cubes and then refined using Laplacian smoothing before being imported into the virtual environment [16].

7.2 Mesh Manipulation and Mutation

Once the user has completed their mesh manipulation, the resulting mesh is voxelated by Calliope, and returned to the neural network for z-vector extraction. This is done by freezing the weights of the GAN and optimizing for the voxel matrix representing the designers manipulated mesh. By doing this, Calliope is able to identify a vector within the latent space that best represents the designer's manipulated mesh. Once this z-vector has been located, the resulting voxel object is returned to the designer inside the virtual environment using the Marching Cubes approach, combined with a series of mesh cleaning and repairing functionalities to ensure watertightness. This process not only allows Calliope to perform mutation on a user-manipulated object, but also automates several aspects of the sculpting process by cleaning the designers manipulated mesh, thus mitigating the time required for careful mesh manipulation by the user and enabling a more rapid sculpting process. Mutation is performed by locating the two Z-vectors of the two input objects within the latent space and fixing a Euclidian-distance line between the two. Calliope then samples the z-vectors of 2 objects evenly spaced along this line between the two input object z-vectors, renders the meshes of these 2 objects, and returns these objects to the user within the virtual environment. Once returned to the user, a new room is spawned containing these objects to be inspected by the user for desired features. In order to allow the user to interpolate between directly manipulated objects and GAN generated objects, we devised a technique for locating a given object geometry within the latent space. Using the method of [14], we froze the weights of the network, vectorized the manipulated mesh, and optimized a latent z vector which best represented the features of the manipulated mesh within the latent space. Since multiple candidates may meet this requirement with similar probability, 4 results are sampled and displayed to the user. While this technique is present in many GAN-based approaches, it has not been implemented for a 3D voxel-based GAN architecture such as ours. This technique is crucial to our interaction pipeline because it enables a closed-loop interaction between the user and machine.

7.3 Computational Idle Time Costs

Due to the heavy computational cost of performing generative design coupled with direct mesh editing within VR, it is necessary to examine performance demands of multiplexing these tasks. To examine this, we stress-tested our system. It is evident from this that mesh interpretation is the most computationally expensive operation, followed by mutation, and finally generation. While none of these tasks incur a detrimentally long wait time, they do provide a substantial period during which the user could perform other design tasks. Wait time increases rapidly with the addition of each simultaneous parallel computation process, especially interpretation of mesh editing. We should note that after 4 parallel operations, performing direct mesh-editing incurred significant performance latency, which could cause motion sickness. However, performing 4 tasks necessary to request 4 parallel computations from Calliope before the first requested task has completed is difficult. Thus, it seems unlikely that this performance issue is critical to the use of the system. These results are, of course, hardware dependent and should be explored further to better understand the computation demands of performing multiple optimization tasks in parallel with one another and a complex VR authoring system. Details of this analysis and evaluation can be found in Appendix A.

8 DESIGN RECOMMENDATIONS

In this section, we outline principal lessons learned through the process of designing and implementing Calliope. These results are not exhaustive but are important considerations for developing future co-creative AI systems. We also compare VR vs non-VR co-creative systems where appropriate and discuss the benefits and weakness of an embodied approach.

8.1 Finding the Goldilocks Sample Number

As mentioned in section 3.2.3, participants expressed a need to see a multitude of options in order to inspire creativity, but not so many that they are overwhelmed. Generative design finds as many solutions as possible that fulfill the input parameters, and subsequently produce an overwhelming number of options. Constraining this solution space is key for constructing an authoring system that is useful and beneficial for target end-users. Given the spatial nature of VR, user's cognitive load is affected differently than 2D GUIs, and thus must be constrained appropriately [49]. Over-constraining this sampling, however, would defeat the purpose of using such a system for inspiration and creativity. In this work, we took a greedy approach, and displayed as many samples as we could to our user without over-burdening the system, causing severe performance lag and thus inducing simulation sickness (See Section 7.4). Striking a balance between number of samples to present to the user is a significant challenge that hinges on constraints afforded by hardware and user bandwidth.

8.2 Democratizing Design

Interfaces to generative design which employ intuitive metaphors for creation could potentially lower the barrier for interacting with these powerful design algorithms. Our system used a radial-data visualization inspired environment in order to provide a structural guide for a user to iteratively develop design solutions. Given the spatial affordances of VR, an embodied approach could better service a wider variety of users by enabling an intuitive exploration of a virtual space versus a 2D visual representation [15]. The benefit of this metaphor-based approach could provide an accessible platform for novice users of generative design and democratize the design process as a whole by providing an intuitive interface for novice users to sample from generative solution spaces, and guide them through the iterative design process by encouraging gradual solution space constraints.

8.3 Embracing Unpredictability

Often creativity support tools are employed and evaluated for their reliability of producing useful designs. However, the benefit of co-creative AI is the unexpected solutions produced by the system. In this way, systems that wish to encourage collaborative machine creativity should embrace the unpredictable nature of these systems as a source of inspiration and unexpected ideas. Similar results have been previously demonstrated by systems such as FoldIt, which leveraged a co-creative approach demonstrate the benefit of unpredictability in these systems for inspiring and engaging users [47, 50]. In a virtual environment, however, the displayed results are more easily inspected visually, but may be more difficult to group or explore based on parametric characteristics evident in 2D GUI interfaces for generative design [36, 41]. Thus, Calliope better affords more granular visual inspection of individual generative results, relying on this inspection to curate and produce additional artifacts, whereas traditional 2D interfaces better afford exploration of the solution space through parametric manipulation. Investigating VR as a modality for parametric exploration of a solution space remains a fruitful topic for further inquiry.

8.4 Guided Sampling of Infinity

The power of generative design lies in its ability to produce ample solutions within a design space. Interfaces to such systems must take into account that this large space of potential solutions can be extremely daunting and overwhelming for users, especially when these solutions are 3D and being observed in VR. Therefore, interfaces to such algorithms must account for this by gently guiding the user through a process of iterative constraint and need-finding. In our approach, the user moved from room to room, specifying and constraining the solution space with each iteration, then viewing their results by lowering the walls. This spatial affordance is a key benefit of VR that could be leveraged beyond design of 3D objects to carefully guide a user through the iterative design process, only viewing all possible design candidates when the user desires. This micro and macro perspectives of design candidates is a key benefit of these authoring platforms in VR.

9 FUTURE WORK AND CONCLUSION

While this initial work is promising for using the medium of VR as an interface for generative design collaboration, many opportunities exist for further investigation. While our system was able to generate and mutate between 4 different classes, additional object classes can be incorporated given adequate training of the network on a sufficient corpus of voxel object data. Creating a large latent space of such objects in the spirit of BigGAN would be the fruitful subject of future work [6]. While our work focused on virtual reality, future work could expand this embodied data synthesis and exploration paradigm to augmented reality by allowing users to generate and manipulate objects in their physical environment. In this way, the spatial nature of VR which we leverage for engagement could easily be transferred to the user's current physical environment. Finally, while we focused on interfacing to generative adversarial networks in this paper, future work could extend using VR as an interface to other methods of generative design such as topological optimization. Adapting the affordances of VR to more viscerally direct the optimization process for generative algorithms is a promising avenue for further examination. Virtual reality presents a promising platform for 3D design authoring tasks in close collaboration with generative algorithms. This initial exploratory work examined interaction possibilities of using generative adversarial networks as an active collaborator in the design process. Users are able to generate, sculpt, and delete objects, as well as mutate objects with others created during the design session. The spatial nature of VR allows the user to remain engaged in the design process exploring and designing other objects during the idle time incurred by generative computation.

REFERENCES

[1] Grégoire Allaire, François Jouve, and Anca-Maria Toader. 2002. A level-set method for shape optimization. Comptes Rendus Mathematique 334, 12: 1125–1130. http://doi.org/10.1016/S1631-073X(02)02412-3

[2] Mayra Barrera Machuca, Paul Asente, Jingwan Lu, Byungmoon Kim, and Wolfgang Stuerzlinger. 2017. Multiplanes: Assisted Freehand VR Drawing. 1–3. http://doi.org/10.1145/3131785.3131794

[3] Martin Bendsøe. 1989. Bendsoe, M.P.: Optimal Shape Design as a Material Distribution Problem. Structural Optimization 1, 193-202. Structural Optimization 1: 193–202. http://doi.org/10.1007/BF01650949

[4] Martin Philip Bendsoe and Ole Sigmund. 2004. Topology Optimization: Theory, Methods, and Applications. Springer-Verlag, Berlin Heidelberg. http://doi.org/10.1007/978-3-662-05086-6

[5] Jorge Loise Borges. (1944b). Las ruinas circulares. In Ficciones, Jorge Luis Borges: obras completas. Harmondsworth : Penguin, pp. 72–79. (1982)

[6] Andrew Brock, Jeff Donahue, and Karen Simonyan. 2019. Large Scale GAN Training for High Fidelity Natural Image Synthesis. arXiv:1809.11096 [cs, stat]. Retrieved May 5, 2020 from http://arxiv.org/abs/1809.11096

[7] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, *et al.* 2015. ShapeNet: An Information-Rich 3D Model Repository. arXiv:1512.03012 [cs]. Retrieved May 5, 2020 from http://arxiv.org/abs/1512.03012

[8] Anthony Chen, Ye Tao, Guanyun Wang, *et al.* 2018. Forte: User-Driven Generative Design. 1–12. http://doi.org/10.1145/3173574.3174070

[9] Tao Du, Adriana Schulz, Bo Zhu, Bernd Bickel, and Wojciech Matusik. 2016. Computational multicopter design. ACM Transactions on Graphics 35, 6: 1–10. http://doi.org/10.1145/2980179.2982427

[10] Upol Ehsan and Mark O Riedl. On Design and Evaluation of Human-centered Explainable AI systems. 6.

[11] George Ferguson, James Allen, and Brad Miller. 1996. TRAINS-95: Towards a Mixed-Initiative Planning Assistant. 8.

[12] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, *et al.* 2014. Generative Adversarial Networks. arXiv:1406.2661 [cs, stat]. Retrieved May 5, 2020 from http://arxiv.org/abs/1406.2661

[13] Saul Greenberg and Bill Buxton. 2008. Usability evaluation considered harmful (some of the time). Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08, ACM Press, 111. http://doi.org/10.1145/1357054.1357074

[14] Naama Hadad, Lior Wolf, and Moni Shahar. 2018. A Two-Step Disentanglement Method. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, 772–780. http://doi.org/10.1109/CVPR.2018.00087

[15] Joshua Harman. 2001. Creating a memory palace using a computer. CHI '01 Extended Abstracts on Human Factors in Computing Systems (CHI EA '01), Association for Computing Machinery, 407–408. http://doi.org/10.1145/634067.634306

[16] William E. Lorensen and Harvey E. Cline. 1987. Marching cubes: A high resolution 3D surface construction algorithm. ACM SIGGRAPH Computer Graphics 21, 4: 163–169. http://doi.org/10.1145/37402.37422

[17] Jonàs Martínez, Jérémie Dumas, Sylvain Lefebvre, and Li-Yi Wei. 2015. Structure and appearance optimization for controllable shape design. ACM Transactions on Graphics 34, 6: 1–11. http://doi.org/10.1145/2816795.2818101

[18] Yoav I. H. Parish and Pascal Müller. 2001. Procedural modeling of cities. Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH '01), Association for Computing Machinery, 301–308. http://doi.org/10.1145/383259.383292

[19] Steven Schkolne, Michael Pruett, and Peter Schröder. Surface drawing: creating organic 3D shapes with the hand and tangible tools. Proc. CHI '01, 261–268.

[20] Kristina Shea, Robert Aish, and Marina Gourtovaia. 2005. Towards integrated performance-driven generative design tools. Automation in Construction 14, 2: 253–264. http://doi.org/10.1016/j.autcon.2004.07.002

[21] Evgeny Stemasov, Tobias Wagner, Jan Gugenheimer, and Enrico Rukzio. 2020. Mix&Match: Towards Omitting Modelling Through In-Situ Alteration and Remixing of Model Repository Artifacts in Mixed Reality. arXiv:2003.09169 [cs]. http://doi.org/10.1145/3313831.3376839

[22] Maryam Tohidi, Bill Buxton, Ron Baecker, and Abigail Sellen. 2006. Getting the right design and the design right: Testing many is better than one. Retrieved May 5, 2020 from https://www.microsoft.com/en-us/research/publication/getting-the-design-right-and-the-right-design-testing-many-is-better-than-one/

[23] Steve Tsang, Ravin Balakrishnan, Karan Singh, and Abhishek Ranjan. 2004. A suggestive interface for image guided 3D sketching. 591–598. http://doi.org/10.1145/985692.985767

[24] Nurcan Gecer Ulu and Levent Burak Kara. 2015. DMS2015-33: Generative interface structure design for supporting existing objects. Journal of Visual Languages & Computing 31: 171–183. http://doi.org/10.1016/j.jvlc.2015.10.016

[25] Nobuyuki Umetani, Takeo Igarashi, and Niloy J Mitra. Guided Exploration of Physically Valid Shapes for Furniture Design. 11.

[26] Danding Wang, Qian Yang, Ashraf Abdul, and Brian Y. Lim. 2019. Designing Theory-Driven User-Centric Explainable AI. Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19, ACM Press, 1–15. http://doi.org/10.1145/3290605.3300831

[27] Michael Yu Wang, Xiaoming Wang, and Dongming Guo. 2003. A level set method for structural topology optimization. Computer Methods in Applied Mechanics and Engineering 192, 1: 227–246. http://doi.org/10.1016/S0045-7825(02)00559-5

[28] Christian Weichel, Manfred Lau, David Kim, Nicolas Villar, and Hans W. Gellersen. 2014. MixFab: a mixed-reality environment for personal fabrication. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14), Association for Computing Machinery, 3855–3864. http://doi.org/10.1145/2556288.2557090

[29] Gerold Wesche and Hans-Peter Seidel. 2001. FreeDrawer: a free-form sketching system on the responsive workbench. Proceedings of the ACM symposium on Virtual reality software and technology (VRST '01), Association for Computing Machinery, 167–174. http://doi.org/10.1145/505008.505041

[30] E. Wiese, J. H. Israel, A. Meyer, and S. Bongartz. 2010. Investigating the learnability of immersive free-hand sketching. Proceedings of the Seventh Sketch-Based Interfaces and Modeling Symposium (SBIM '10), Eurographics Association, 135–142.

[31] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. 2016. Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling. In Advances in Neural Information Processing Systems 29, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon and R. Garnett (eds.). Curran Associates, Inc., 82–90. Retrieved May 5, 2020 from http://papers.nips.cc/paper/6096-learning-a-probabilistic-latent-space-of-object-shapes-via-3d-generative-adversarial-modeling.pdf

[32] Physics-based Generative Design | Autodesk Research. Retrieved May 5, 2020 from https://www.autodeskresearch.com/publications/gendesign

[33] ChronoFab: Fabricating Motion | Autodesk Research. Retrieved May 5, 2020 from https://www.autodeskresearch.com/publications/chronofab-fabricating-motion

[34] Experimental Evaluation of Sketching on Surfaces in VR | Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. Retrieved May 5, 2020 from https://dl.acm.org/doi/10.1145/3025453.3025474

[35] DreamRooms: Prototyping Rooms in Collaboration with a Generative Process •Graphics Interface. Graphics Interface. Retrieved May 5, 2020 from https://graphicsinterface.org/proceedings/gi2019/gi2019-19/

[36] DreamSketch: Early Stage 3D Design Explorations with Sketching and Generative Design | Autodesk Research. Retrieved May 5, 2020 from https://www.autodeskresearch.com/publications/dreamsketch-early-stage-3d-design-explorations-sketching-and-generative-design

[37] SymbiosisSketch | Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems. Retrieved May 5, 2020 from https://dl.acm.org/doi/10.1145/3173574.3173759

[38] Generative Design at Airbus | Customer Stories | Autodesk. Retrieved May 5, 2020 from https://www.autodesk.com/customer-stories/airbus

[39] Creative Computers? Here's What You Get When You Design a Chair Using Algorithms. Futurism. Retrieved May 5, 2020 from https://futurism.com/creative-computers-heres-what-you-get-when-you-design-a-chair-using-algorithms

[40] Generative Design Applied on Buildings. BIM Toolbox. Retrieved May 5, 2020 from https://autodesk.typepad.com/bimtoolbox/2017/06/generative-design-applied-on-buildings.html

[41] Dream Lens: Exploration and Visualization of Large-Scale Generative Design Datasets | Autodesk Research. Retrieved May 5, 2020 from https://www.autodeskresearch.com/publications/dreamlens

[42] A suggestive interface for 3D drawing | Proceedings of the 14th annual ACM symposium on User interface software and technology. Retrieved May 5, 2020 from https://dl.acm.org/doi/10.1145/502348.502379

[43] Sketching User Experiences: Getting the Design Right and the Right Design - 1st Edition. Retrieved May 5, 2020 from https://www.elsevier.com/books/sketching-user-experiences-getting-the-design-right-and-the-right-design/buxton/978-0-12-374037-3

[44] TutoriVR: A Video-Based Tutorial System for Design Applications in Virtual Reality. Adobe Research. Retrieved May 5, 2020 from https://research.adobe.com/publication/tutorivr-a-video-based-tutorial-system-for-design-applications-in-virtual-reality

[45] Tilt Brush by Google. Retrieved May 5, 2020 from https://www.tiltbrush.com/

[46] Christopher Andrews. 2019. An Embodied Approach to AI Art Collaboration. In Proceedings of the 2019 on Creativity and Cognition (C&C'19). Association for Computing Machinery, New York, NY, USA, 156–162. DOI: https://doi.org/10.1145/3325480.3325506

[47] Duri Long, Mikhail Jacob, and Brian Magerko. 2019. Designing Co-Creative AI for Public Spaces. In Proceedings of the 2019 on Creativity and Cognition (C&C'19). Association for Computing Machinery, New York, NY, USA, 271–284. DOI: https://doi.org/10.1145/3325480.3325504

[48] Liang Men and Nick Bryan-Kinns. 2019. LeMo: Exploring Virtual Space for Collaborative Creativity. In Proceedings of the 2019 on Creativity and Cognition (C&C'19). Association for Computing Machinery, New York, NY, USA, 71–82. DOI:https://doi.org/10.1145/3325480.3325495

[49] Elinda Ai-Lim Lee, Kok Wai Wong. 2014 Learning with desktop virtual reality: Low spatial ability learners are more positively affected. Computers & Education, Volume 79, 2014, Pages 49-58, ISSN 0360-1315 https://doi.org/10.1016/j.compedu.2014.07.010.
[50] Firas Khatib, Seth Cooper, Michael D. Tyka, Kefan Xu, Ilya Makedon, Zoran Popović, David Baker, Foldit Players. Algorithm discovery by protein folding game players. Proceedings of the National Academy of Sciences Nov 2011, 108 (47) 18949-18953; DOI: 10.1073/pnas.1115898108

APPENDIX A

In order to understand the idle time for each operation (mutation, interpretation of mesh editing, and generation) we first performed a series of each task sequentially. For each of these operations, we collected the time taken by each task. This information is reported in Figure 9. Since Calliope encourages parallel design iteration, it's necessary to understand the idle wait times of operations in parallel. To evaluate this, we performed 2, 3, and 4 operations simultaneously and recorded the idle time. The results of this investigation are reported in Figure 10. We did not evaluate generation operations in this way because the increase in latency was negligible.

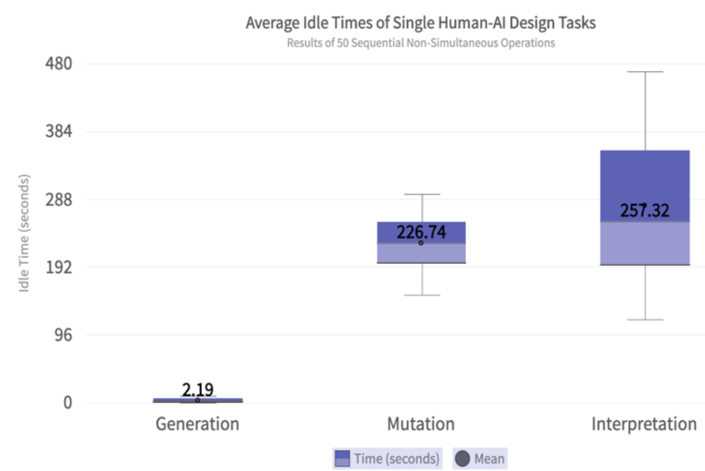

Figure 9: Average computational idle time for single tasks

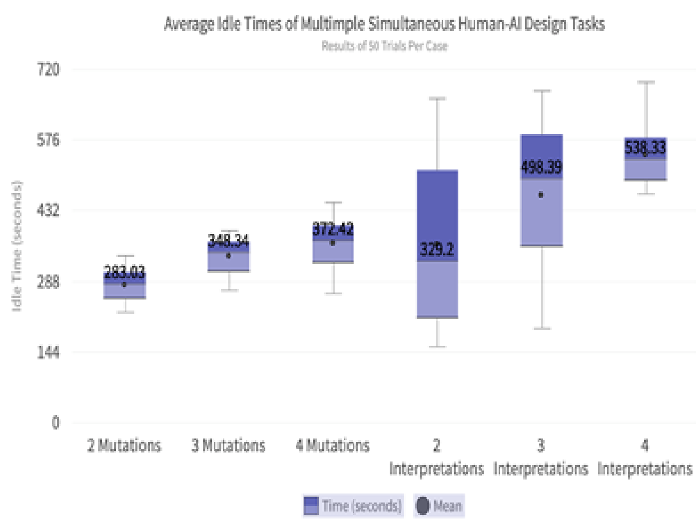

Figure 10: Average computational idle time for parallel tasks