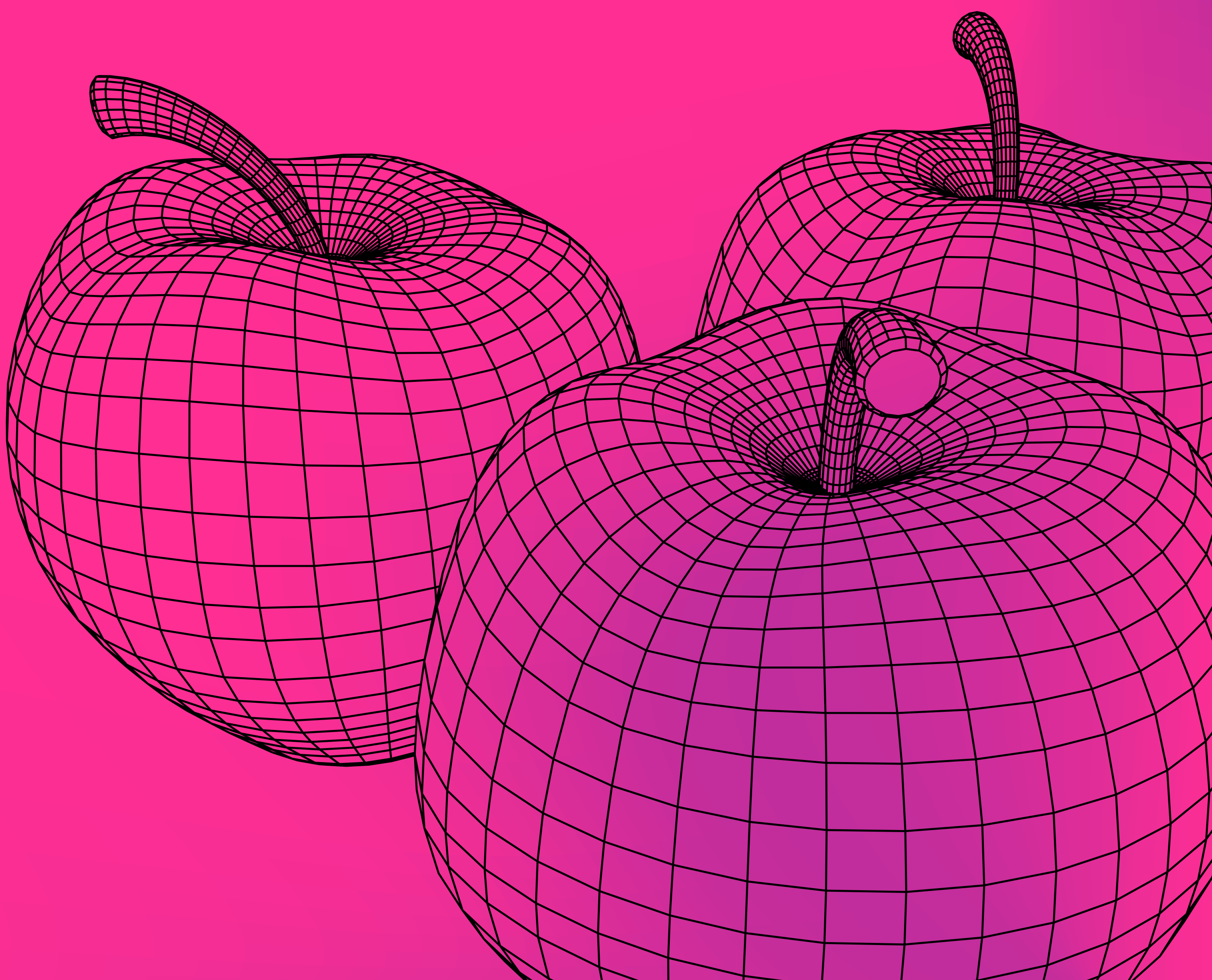
adam

a novel design tool harnessing
generative machine learning
for autotelic cad creation
& physical making

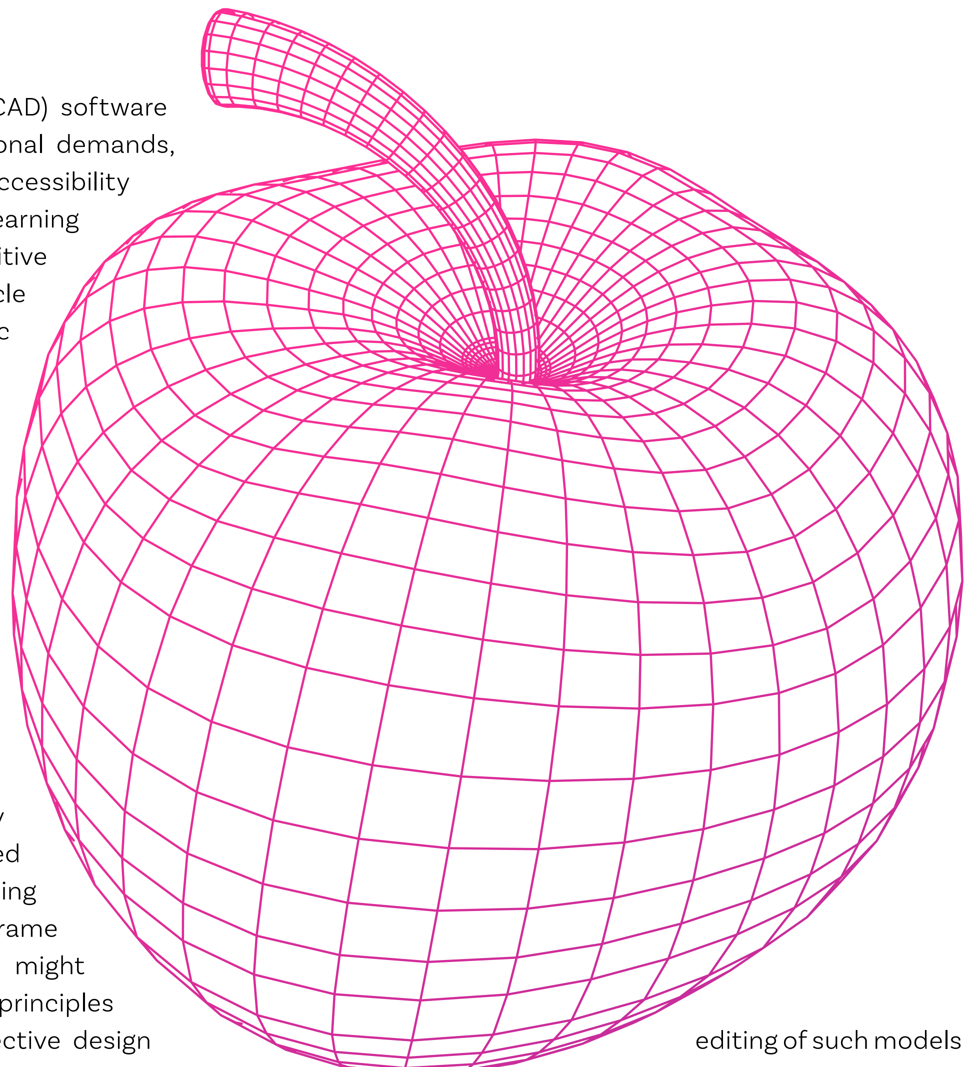
by **zach dive**



brief

This thesis explores the integration of generative machine learning into computer-aided design (CAD), specifically through large-language models, to simplify CAD interfaces. It introduces a natural language interface with easily adjustable parameters, making CAD intuitive for casual or novice designers to design 3D printable objects. The research aims to address the dichotomy in CAD between complex functionalities and user-friendliness, aligning CAD with human thought processes and communication styles, thereby fostering amplified creativity in the design process.

abstract



The realm of computer-aided design (CAD) software has predominantly catered to professional demands, often at the expense of broader accessibility and user enjoyment. The significant learning curve, coupled with often unintuitive interfaces, poses a fundamental obstacle to innovation. The recent rise of autotelic tools, as illustrated by the term ‘casual creators’ (Compton & Mateus, 2015), redirects our focus towards a blend of creativity and pleasure, contrasting the traditional emphasis on mere productivity. While some innovations have ventured into combining generative interfaces with existing CAD software, a holistic approach catering to everyday creators with little prior knowledge still remains elusive.

Applying methodologies that employ generative machine learning, especially in 3D shape generation, prompt-based renderings, and natural language processing (NLP), offers a promising avenue to reframe the CAD environment. This exploration might not only restructure the foundational principles of CAD but also establish it as an effective design collaborator (Steinfeld, 2017).

adam—the system proposed here—builds upon an existing open-source, script-based CAD framework to develop a natural language interface that in turn lowers its barrier to entry. By reimagining existing tools, the goal is to design and prototype a CAD interface for physical making that appeals more broadly, especially to those unfamiliar with traditional CAD systems.

This research predominantly concentrates on the conceptualization and preliminary prototyping tailored for the casual creator demographic, deliberately focussing on two key features: the generation of 3D printable CAD models through typed text prompts, and the parametric editing of such models through a subset of easily manipulable variables.

By simplifying CAD methodologies, this research envisions a shift toward a computational design paradigm that is not only time-effective but also intuitively accessible, promoting a democratized approach to design.

While the current research is focused on conceptualization and preliminary prototyping for everyday creators, future endeavors could delve into extensive usability studies, refinement of the integrated AI for broader application domains, and potential to explore avenues of industrial-scale software deployment within the broader CAD ecosystem in more production-ready use cases.

contents

1	history and prior art	5
2	motivation	8
3	method	9
4	final design	11
5	discussion	14
6	future works	15
7	conclusion	16

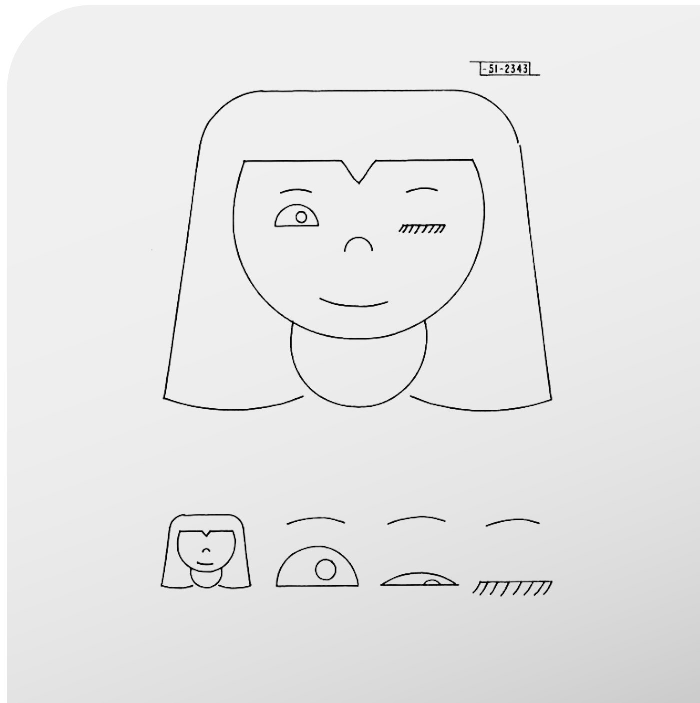
history and prior art

Six decades have passed since Dr. Patrick Hanratty, widely considered the "Father of CAD," created the first numerical control system. This seminal work laid the foundational framework for what would eventually evolve into computer-aided design (CAD). From those early beginnings, CAD technology has undergone multiple phases of innovation and expansion, deeply embedding itself into various industries such as engineering, architecture, and manufacturing (Fateminasab, 2021). With each technological leap, CAD has become increasingly powerful, offering precision, versatility, and adaptability, in turn revolutionizing the design landscape.

In the late 1950s, General Motors (GM), the leading automobile manufacturer of the time, partnered with IBM to create one of the initial substantial systems for consumer product design, initially known as "Digital Design." Released in 1964, IBM's Design Automated by Computer (DAC-1) software represented a pioneering venture in this field (Prince & Wesley, 1971). GM used DAC-1 extensively for car development until the late 1960s, but a change in presidential leadership led to the abandonment of the program and the disbanding of the teams involved. DAC-1 operated on an IBM 7090 mainframe computer, equipped with a specialized graphics console, allowing users to draw, rotate, and manipulate images using a light pencil. This display system is often regarded as the first commercially available CAD program (Krull, 1994).

Soon after came the most defining moment in CAD's early evolutionary timeline which was the Sketchpad system. Developed by Ivan Sutherland in 1963 for his MIT doctoral thesis, Sketchpad's true innovation was rooted in its graphical user interface (GUI), an unprecedented feature that enabled designers to interact directly with the computer via visual elements. For the first time, users could draw and manipulate geometric figures on a CRT monitor, fundamentally altering human-computer interactions in CAD systems (Sutherland, 2003). Sketchpad paved the way for a host of design and drafting software, becoming the most celebrated among early drawing tools. Primarily utilized for engineering sketches, it also had artistic uses, notably in a well-known depiction of Nefertiti that featured basic animation capabilities.

In Sutherland's PhD thesis, he noted the particular advantages of Sketchpad for drawings requiring motion or high accuracy, highlighting its efficiency over at-the-time conventional techniques. Moreover, the graphical interface was not just a technological advance; it was a conceptual one too. It embodied the paradigm shift; repositioning the computer from a mere computational tool to an interactive partner in the design process.



Winking Girl "Nefertite"

While the history of CAD closely parallels the history of the computer, this pivotal innovation, whilst transformative, has iteratively grappled with issues related to user accessibility and intuitive use. Each new layer of capability has often added an extra layer of complexity, limiting its initial creative potential.

For instance, the 1980s witnessed the emergence of AutoCAD and the inception of parametric as well as

AutoCad AutoDesk Software

variational modeling. AutoCAD, functional on personal computers (PCs), transformed the drafting sector, emerging as the leading drafting software for PCs. Its success played a key role in establishing Microsoft Windows as the predominant platform in industrial design and architecture, despite Apple's established presence in design industries and visual arts. 40 years in the future and AutoCAD is still the most widely used CAD solution in the world. Furthermore almost every object we use is developed with computer-aided design (CAD).

Whilst consequently expanding design capabilities, this has also escalated the learning curve for users of these systems. Specifically this progression has also brought with it a phenomenon commonly referred to as "feature bloat," where the continuous addition of new features leads to overly complex and less user-friendly interfaces. This complexity can deter casual and novice users, stifling the very creativity that these tools are intended to support. George Stiny's critique of CAD systems as tools for supporting creativity is particularly relevant in this context (Stiny, 1989). He emphasizes that while CAD systems have become increasingly powerful, their complexity often overshadows their potential as creative tools. The idea of 'casualness' (Compton, 2015) and 'everyday creativity' in software design is crucial, as it speaks to the accessibility and intuitive use of these tools by a broader range of users. This harkens back to the core purpose of CAD: to act as a facilitator in the design process, empowering users to translate their creative visions into tangible designs with ease and efficiency.

As recent researchers have noted, modern CAD tools often stifle the user's innate creativity. In Lee & Jan's 2016 study, they observed the impact of current 3D CAD interfaces on user ideation. The results were clear; after modeling, sketches became simpler, with fewer organic shapes and details. Moreover, modeling success was a key predictor of whether complexity increased or decreased in subsequent sketching. Clearly, the complex interfaces of modern CAD software have various relative disadvantages, including restricting creativity (Lee & Jan, 2016).

Fast forward to today, and we find ourselves at a similar transformative juncture as at MIT in the late 1960s. Just as the introduction of the GUI transformed CAD by broadening its appeal and making it more intuitive, the incorporation of natural language interfaces, through the revolution of large language models (LLMs), is redefining the way humans interact with computers in a wide range of programs. This new frontier seeks to address the need for CAD tools to be more accessible and intuitive for a broader audience, including casual or novice designers. Such an evolution reflects a response to unmet needs for simpler, more approachable interfaces in design technology.

Steve Jobs' famous analogy of the computer as a "bicycle for the mind" provides a compelling vision for the future of CAD. In this view, computers serve to augment and extend our cognitive capacities. The current state of CAD interfaces, however, seems to diverge from this vision. Their complexity often acts as a barrier rather than a facilitator, inhibiting the user's ability to fully engage and be creatively uninhibited within the software. The proposed integration of natural language interfaces aims to align CAD with Jobs' vision, effectively transforming it into a tool that amplifies rather than restricts cognitive productivity.

Another noteworthy venture into such human-computer interaction was the "Put-That-There" system developed in 1979 at MIT. This system combined voice commands and gesture recognition to create an intuitive interface for spatial data management. While not a CAD tool per se, the "Put-That-There" system provides essential clues for our current investigation into intuitive, natural language interfaces for CAD. The system made spatial manipulation significantly more user-friendly through natural language and gestures. This represents a key conceptual bridge to the multimodal, natural language CAD interfaces, that form the core focus of our research.

In recent years, there have been several innovative projects that have sought to incorporate more intuitive, 'human' forms of input into CAD systems, enhancing their accessibility and user-friendliness. For instance, a team from the University of Hong Kong developed a 'Natural Voice-Enabled CAD' system (Xue et al., 2009) that allows users to employ everyday language to carry out CAD modeling tasks. Additionally, the MIT Media Lab introduced

"A New Kind of Bicycle"

"Put-That-There"

'The EmotiveModeler' project, which uniquely leverages descriptive adjectives to assist designers in crafting objects that embody specific emotional attributes. This tool is designed with both seasoned professionals and beginners in mind, facilitating the manipulation of 3D forms through its user interface. These advancements exemplify the ongoing efforts to democratize CAD technology, making it more approachable for a diverse range of users.

By revisiting such seminal works, we appreciate how groundbreaking innovations in natural language and novel interactions have incrementally made computing more user-friendly and efficient. As we integrate natural language capabilities into existing CAD tools, we are in essence continuing this legacy, aiming to create an interface that is not only technically advanced but also intrinsically aligned with human thought processes and communication styles.

motivation

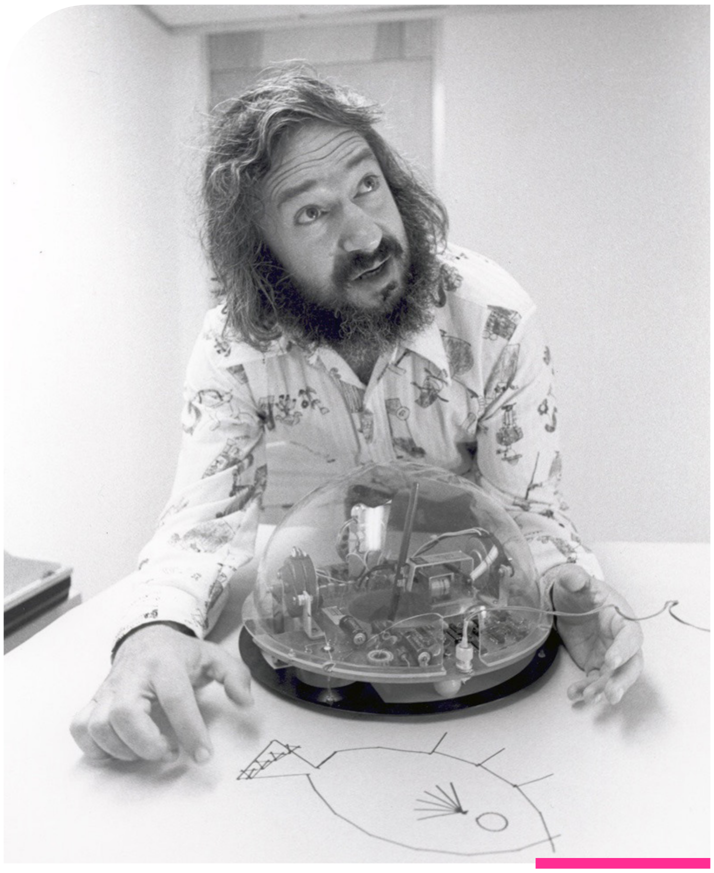
The journey of CAD reflects a continuous endeavor to create tools that extend our capacity to think and design; a quest to make technology speak the language of human creativity. However, as we have seen most traditional software including CAD, deploys a highly abstract and low-level syntax that is designed to be easily understood by a computer but not by a person. This makes computer-aided creation inaccessible and inefficient (Ebbecke, 2022).

In order to enable computers to be true creative collaborators, human-machine interfaces should ideally use a language that human designers are familiar with. Consequently In integrating a natural language interface into CAD, we are hoping to build on Seymour Papert's vision of "tools to think with," where the design process becomes an active construction of knowledge (Papert, 1980).

Seymour Papert, founder of the MIT Artificial Intelligence Lab and later co-founder of the Media Lab, was one of the early pioneers of artificial intelligence and proponents of using software tools to supercharge learning. Papert's constructivist approach heavily influences the creation of **adam**, advocating that we learn best by doing (Harel & Papert, 1991).

Papert recognized that computers could be used as more than just an information delivery system, but had the potential to empower people, particularly to explore, experiment, and express themselves. **adam** as an autotelic tool aims to do precisely this; using esoteric high-level language to convey an equally large amount of design decisions the way humans naturally do. This consequently allows people with limited technical knowledge to utilize their rich natural language abilities to iteratively design complex 3D models with unprecedented speed in a playful manner. Furthermore, by empowering users with the ability to build and iterate with the ease of simple text prompts, users are able to effectively accumulate 3D CAD knowledge. This is as per the central tenet of 'Constructionist' learning theory, as users are actively engaged in constructing 'things' in the world. Traditional design software, conversely, cannot interpret such language as it instead operates with low-level interactions that can take years to master, resulting in great inaccessibility and expense. For instance, anyone with a concept for a 3D form would have to work with a trained mechanical engineer or industrial designer to staunchly translate their descriptions into cartesian coordinates of digital vertices, normal maps of cyber materials, or integers that determine the appearance of a given mesh (Ebbecke, 2022). The ability to express an idea through an artificial 3D model displayed on a computer is therefore reserved for an elite minority who possess the resources required to learn and practice computer-aided design.

Furthermore, the complex nature of how humans experience and communicate their own reality underscores the need for more intuitive design interfaces. The theory of "enaction," as proposed by Davis et al. (2015), suggests that human creativity is not a static attribute but emerges through continuous and improvised interactions with the environment as well as its agents (Varela et al. 1991; Stewart et al. 2010). Therefore, the ideal creative software tools are those that are able to function in collaboration with human users, by simulating the natural dialogue that occurs between humans in creative realms (Biles, 2003; Lubart, 2005; Hoffman & Weinberg, 2010; Zook et al. 2011; Davis et al. 2014).



Seymour Papert

Inspired by the domain of AI-driven creativity support tools (Shneiderman, 2007), we therefore aim with this thesis to extend Seymour Papert's ideas of "tools to think with," towards a paradigm dubbed "tools to create with"—tooling that gives users the ability to actualize their ideas in real-time. Our efforts stand on the shoulders of the past, striving to reshape CAD into an intuitive, accessible partner for all, honoring the philosophy of software as a potential 'bicycle for the mind', thus enhancing human intellectual exploration through design.

method

Various researchers have explored different approaches to tackle the challenges in this domain, particularly in novel view synthesis and advancing generative diffusion models. In general these have shown promising results in creating realistic results and enabling controllability of generated 3D content. Specifically much of this work has come in the form of 3D generation from NeRFs (Mildenhall et al, 2020), 3D Gaussian Splatting (Kerbl et al, 2023) as well as 3D foundation models or creative tools to generate 3D models from text or semantic descriptions. For instance, DreamFusion uses natural language allowing users to use a text prompt to build a 3D mesh from scratch (Poole et al. 2022). Furthermore, user-friendly interfaces offered by Nerfstudio (Tancik et al., 2023) and Luma Labs AI have increased the adoption of these 3D representations in content creation and visual effects.

The prevailing research (Po et al., 2023) suggests that combining such generative models could lessen the dependency on zero-shot model output, with perhaps only some supplementary work to post-process such as simulation for validation. However, this thesis advocates for an alternative approach, as these methods are not currently optimized for parametric real-world modeling but rather generating digital 3D graphic content.

The methodology this thesis pursues is fine-tuning and prompting multi-modal large-language models on CAD scripts relying on a vector database of SCAD (Script-CAD) instruction embeddings to be able to execute expert level code in scriptable CAD environments. A huge variety of scriptable CAD programs exist today: ImplicitCAD, libfive Studio, OpenSCAD, CadQuery, FreeCAD, and ruckus. Moreover, Blender has a console interface and Three.js, while not CAD oriented, is also another compelling example of a 3D programming language. For the purpose of **adam** we are using an OpenSCAD environment, due to its robust open-source ecosystem, comprehensive documentation and compatibility with the STL file format. This format is optimized for physical fabrication with modern domestic 3D printers, making it a practical choice for the everyday creator.

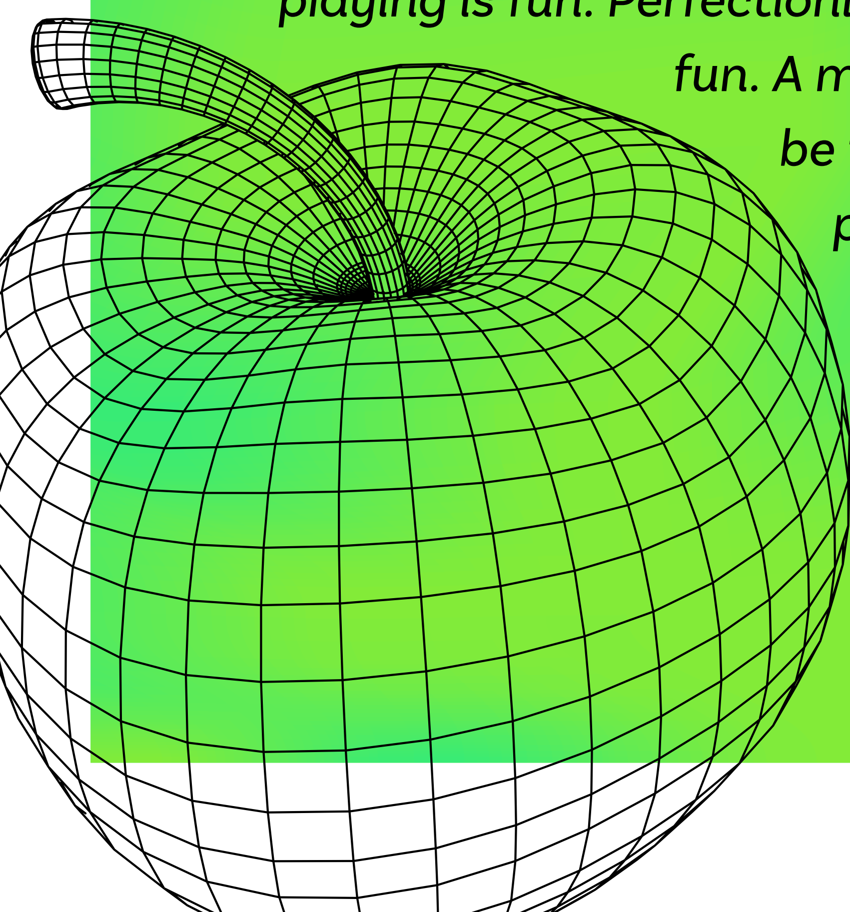
The potential of large language models in generating production-ready software in a diverse set of domains has been highlighted in recent research (Liu et al., 2023). By integrating and chaining these models with computational tools like Numpy for instance, the models' capabilities can be significantly extended. In this case, this allows the model to perform complex mathematical and statistical computations, a task which would otherwise be challenging for a model trained mostly for Q&A. The same can be said for a variety of applications including building parametric CAD models.

Moreover, with emerging multi-modal models such as LLava (Liu et al., 2023), it is possible to give our system the ability to query its own outputs much like a human would do by viewing the 3D form and iterating if the script generates a form that differs from the natural language command inputted by the user.

Overall, this process emulates a more intuitive, human-like approach to generative 3D systems, offering a potentially promising direction for future developments in CAD interfacing.

“All art is a work in progress. It’s helpful to see the piece we’re working on as an experiment. One in which we can’t predict the outcome. Whatever the result, we will receive useful information that will benefit the next experiment. If you start from the position that there is no right or wrong, no good or bad, and creativity is just free play with no rules, it’s easier to submerge yourself joyfully in the process of making things. We’re not playing to win, we’re playing to play. And ultimately, playing is fun. Perfectionism gets in the way of fun. A more skillful goal might be to find comfort in the process. To make and put out successive works with ease.”

Rick Rubin



final design

Prompt: “A teapot”

According to Kate Compton (2019) a ‘casual creator’ system is one that:

“(1) privileges enjoyment of the creative process above productivity,

(2) produces artifacts within a limited-yet-significant domain space, enabling automation and support that is both passive (encoded into the domain model and system constraints) and active (responding to user actions),

(3) supports a state of creative flow by restricting choice and preventing hard failures while allowing rapid iteration”

adam has been carefully conceived with each and every one of these core principles in mind:

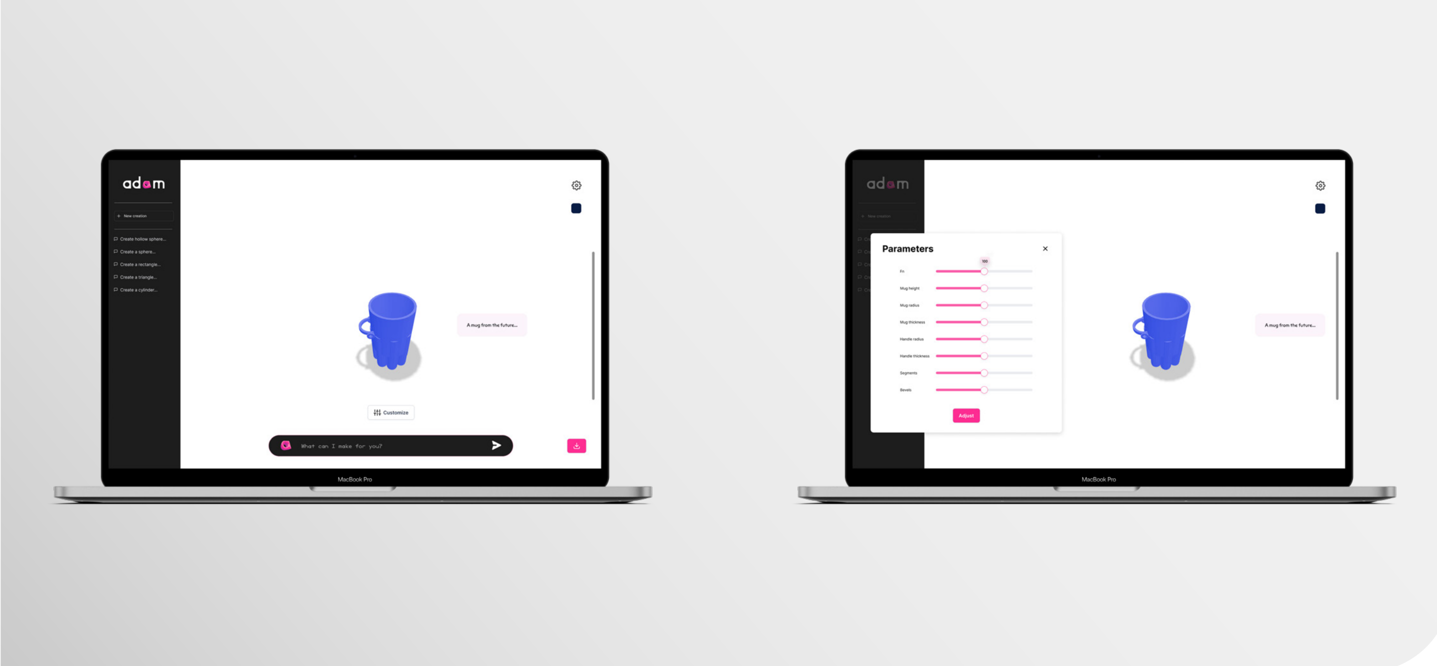
(1): **adam** is designed quite literally for the purposes of creativity “for its own sake”. Developed as a ‘sandbox’ style web-playground, the interface is deliberately equipped for exploration. Furthermore, the choice of developing an accessible web-application, rather than a native program, circumvents the process of downloading and installing heavy files, intentionally lowering friction faced by potential users and allowing them to immediately create 3D models.

Additionally the primary focus for the design of this software, is that it must be extremely easy to use. All possible learning of the tool must occur in the first few minutes, and provide a gratifying experience even if the user never spends time to become acquainted with its nuances. No documentation should be needed and core interactions intended for the software should “explain themselves” through use.

(2): **adam** is specifically optimized for the creation of 3D printable CAD models from text prompts. Users who prioritize “making” over a breadth of capabilities can be accommodated by focusing the system’s outputs on specific items, such as 3D printable plant pots or mugs. The datasets and instructions used to train **adam** have been carefully weighted towards household items and “things,” aiming to create democratized use-cases for everyday makers and crucially reducing the possibility space, compared to previously mentioned general-purpose professional CAD tools. The narrowness of the possibility space allows **adam** to provide a user-friendly experience eliminating potential bad artifacts and accelerating the process of creating good ones, at the expense of increased flexibility and versatility. Nonetheless, this loss is tolerable as these models aren’t being created in response to the strict demands of a manager or client in a professional setting, but rather because they are intrinsically fun to make.

Moreover, this approach uses inherent assumptions for every generated model such as the need for manifold geometry and a watertight mesh. In addition, given a user’s prompt, for instance “a teapot,” the model can abstract to the first principles features of this model such as substantial walls, a spout, a handle, and a suitable lid to enhance the user’s creative power within a defined scope.

In this hypothetical scenario, the user can then utilize natural language to specify key aspects like the model’s volume or iterate on the form of its key affordances, whilst still ensuring that it retains our mental model of what a ‘teapot’ is. The system then automatically adjusts the model’s dimensions, thickness, and overall form based on these inputs, streamlining the design process and bypassing the need for manual adjustments. For instance, if a user inputs a prompt to enlarge the handle or reduce



The user interface with adjustable parameters

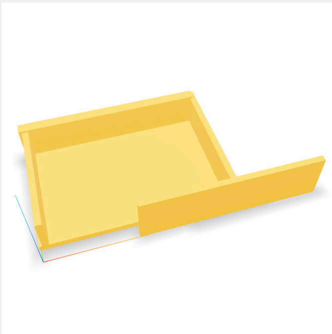
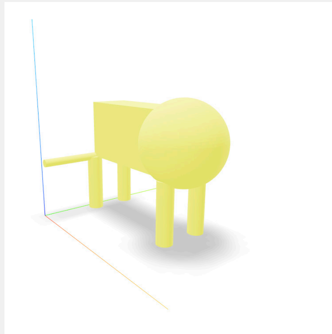
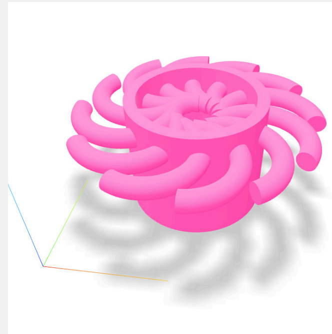
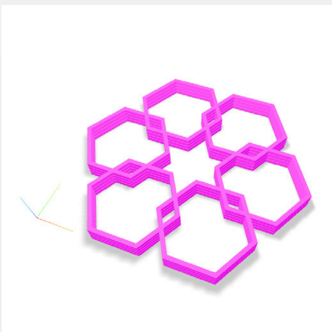
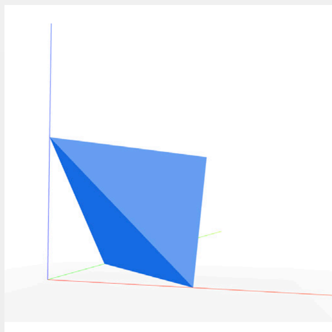
the size of the spout, the system calculates and applies these changes within practical limits.

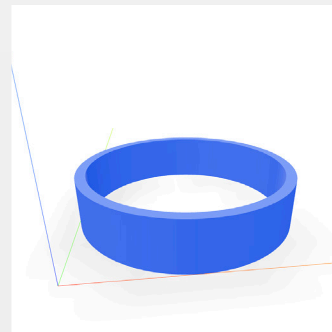
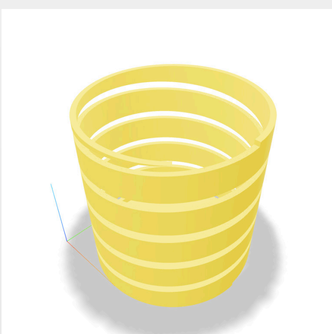
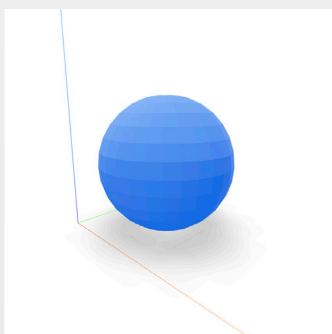
This method of guiding user prompts takes advantage of the system's generative capabilities. It directs users towards feasible and practical design choices, ensuring a balance between creative freedom and the constraints of design for 3D printing. As a result, users enjoy a sense of empowerment and creative fulfillment, with their input translating into realistic, printable 3D models without feeling overly limited by the system's boundaries.

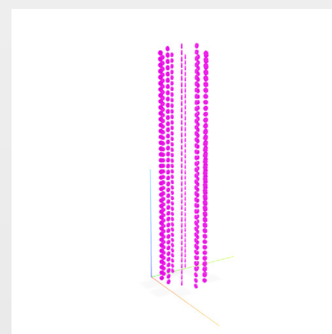
(3): As a system that supports the iterative generation of 3D models given users' text instructions to support refinement and critique. The goal is to mimic a user trying to verbally convey a visual idea to an expert CAD user. Another crucial component to the design of **adam** is the abilty to adjust a set of encoded variables easily through a set of sliders, once the model has been generated via text, giving some level of creative control to the user. This method of interaction is also pivotal in reducing the possibility space, as it confines user interactions and iterations with the CAD model strictly to text prompts, and a subset of easily manipulable variables to build the 3D form in real-time. This constraint is deliberate, fostering a focused and efficient creative process.

Historically, design and engineering in CAD programs have relied on Graphical User Interface (GUI) based interactions, a concept known as "Direct Manipulation" introduced by Ben Shneiderman in 1982. Shneiderman later expanded this concept (Shneiderman et al., 2006), characterizing it as a software category where users manipulate objects of interest directly, such as in CAD with a 3D model on their screen. This form of interaction is akin to instructing the computer on "what to do." In contrast, **adam** proposes a CAD system where the user tells the computer "what outcome they want." This in turn creates a new UI paradigm, epitomized by current generative systems in intent-based outcome specification.

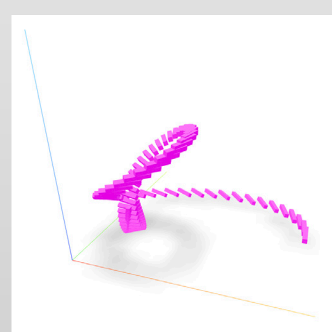
To further articulate this concept within the context of generative systems, Compton (2019) coined the term 'grok-loop.' This term encapsulates the hypothesis-action-evaluation cycle inherent to such systems. In **adam**, the grok-loop is central to the user experience, as it facilitates a continuous process of proposing ideas, witnessing their realization in the form of 3D sketches, and refining them based on the outcomes. This cycle is crucial for maintaining a state of creative flow, as it encourages rapid iteration without the risk of hard failures, thereby supporting the creative process in a seamless and user-friendly manner.




a litany of user generated models



discussion

Generative design, as (Achilopatas et al., 2018; Saadi & Yang, 2023) observed, offers groundbreaking opportunities for design teams to explore creative spaces. This innovation aligns with efforts by designers and researchers, such as those noted by Audry (2021), who are forging a symbiotic relationship between humans and AI to cultivate new ideas. Despite these advancements, there remains a lack of consensus on the optimal interface for future CAD systems.

The question of whether a natural language interface is the next evolutionary step in CAD, particularly beyond the scope of 'casual creators,' is unclear. While natural language interfaces provide an accessible 'tool to create with,' they may inadvertently impact the depth of CAD learning and mastery. This potential shift towards a more goal-oriented approach may in fact diverge from Papert's philosophy of 'tools to think with.' The concept of "mastery" in systems like **adam** can be dualistic: firstly, users might develop a predictive model of the system to effectively communicate their intentions, and secondly, they could achieve a sense of control through experimentation, exploration and even deliberate 'poking' of the systems capabilities.

For instance in such a learning process users could intentionally even submit unusual or confusing queries to the system in order to get a sense of what it can and can't do, and also to playfully exert dominance by "putting one over" an ostensibly "intelligent system." Through many hours of use the user would in theory be able to understand the system's limits, having clear intuitions on what language to use to tailor towards desired outcomes. This method of communication, now even considered a discipline in and of itself, is known as "prompt engineering" (Ziegler & Berryman, 2023).

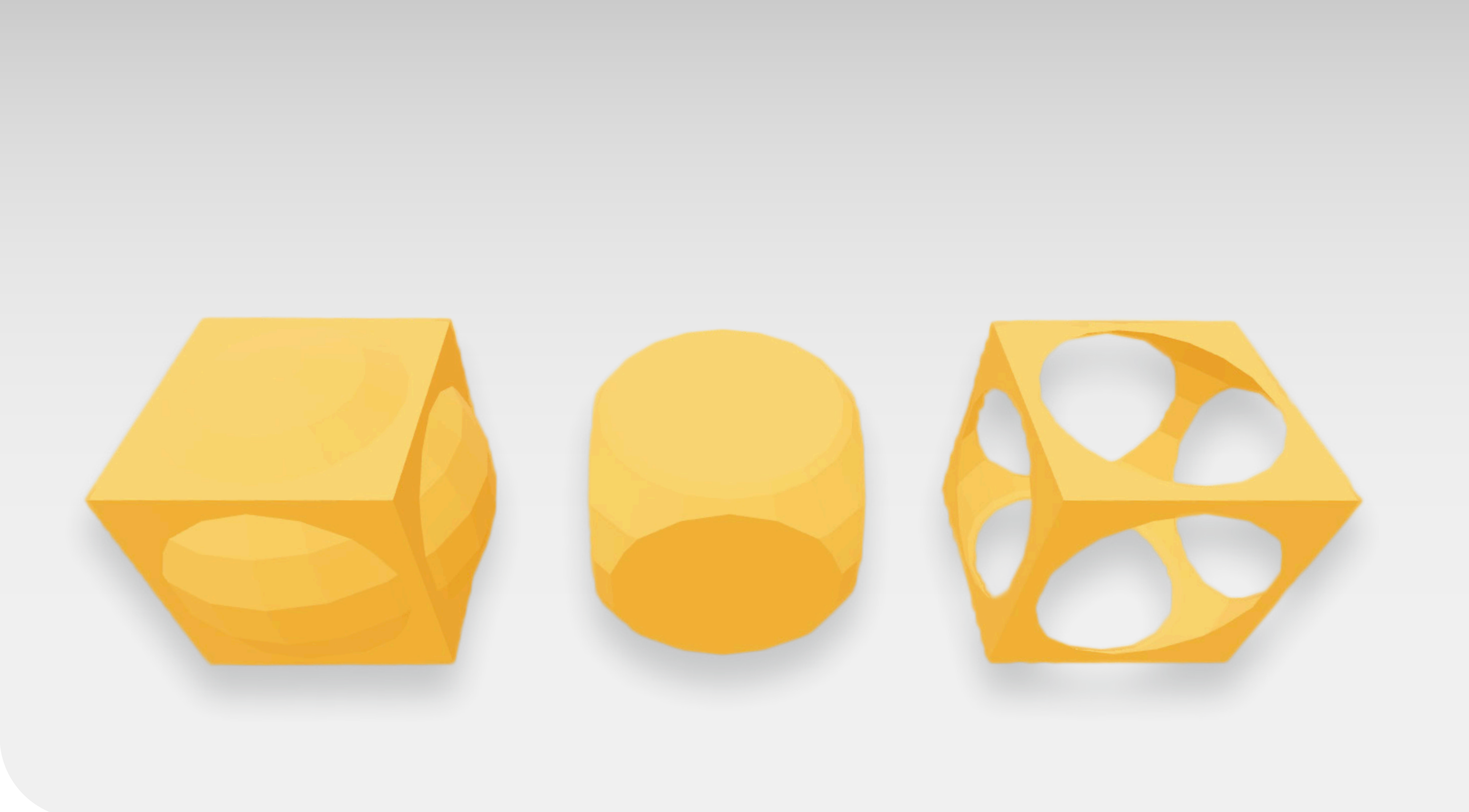
However, the potential lack of manual control in natural language-based systems could deter seasoned CAD users. Traditional GUI interactions, such as "sketch-and-extrude" mechanics, are foundational in CAD for their efficient abstraction in constructing 3D forms (Hutten, 2021). Identifying the right level of abstraction for various users and use-cases is crucial in developing effective tooling.

Perhaps using the aforementioned power of multi-modal language models there is potential for using a GUI interface for initial 3D sketching and then coupling this with natural language prompts to generate some boilerplate designs for the developer to then 'fill in.' This approach could offer a balanced abstraction for designers, allowing for subsequent refinement and customization.

It is also important to understand the limitations of the script-based CAD environments for generating 3D shapes. Fileted or curved surfaces are cumbersome to create in **adam**, particularly as SCAD environments lack obvious tools to bevel edges. GUI-based interactions are more suited for such tasks, compensating for the areas where natural language interaction may fall short. Striking a balance between GUI and generative inputs could enable a wider range of users to develop a comprehensive understanding of the software, extending its utility beyond solely autotelic use cases.

In sum, while generative design introduces exciting possibilities in CAD, the optimal interface likely lies in a hybrid model that combines the strengths of natural language and GUI-based interactions. This approach could cater to a broader spectrum of users, ranging from novices to seasoned professionals, ensuring both accessibility and depth in the CAD experience.

Prompt: "A mug from the future"—3D Printed



A series of generated cubes

future works

Overall, the ambition of this thesis and **adam** is to provide a glimpse into the future of the CAD landscape, showcasing the potential of natural language generative design solutions in a field increasingly influenced by generative machine learning systems. Moreover, where future envisionments could potentially build out a more fleshed out tool, aimed not only at everyday, casual creators, but also skilled CAD designers.

To elaborate, **adam** stands at the forefront of addressing a critical gap in the current design workflow. Professional CAD users often begin with traditional pencil-and-paper sketches, but **adam** offers a digital alternative for rapid prototyping and visualization of 3D models. This capability is invaluable, especially in the early design stages, allowing for the generation of high-fidelity forms that can be effectively communicated in digital formats, such as slideshow presentations.

Key to **adam**'s design philosophy is its alignment with the principles of generative variability. Unlike traditional discriminative systems that produce fixed outcomes, **adam** emphasizes the production of a diverse range of outputs, catering to the multifaceted needs of its users (Weisz et al., 2023). This approach supports both the creation of specific artifacts and the exploration of creative ideas, accommodating the varied aims of users in design context.

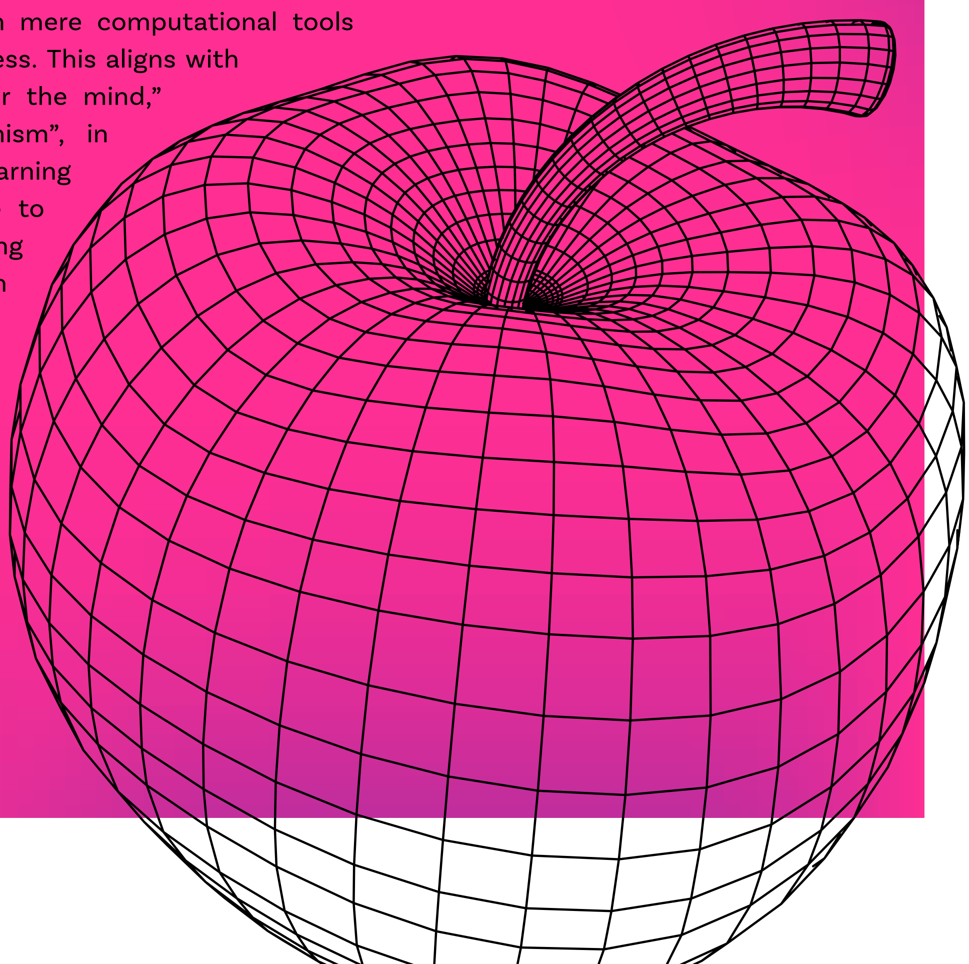
However, the inherent variability and potential imperfection in the outputs of generative models like **adam** necessitate a user-centric design approach. Users must be prepared for instances where the system's outputs may not precisely match their envisioned artifacts. To enhance user experience, future works could incorporate features that improve efficiency, comprehension, and ownership of the outcomes, staying true to the principles of human-centered AI (Weisz et al., 2023). Moreover, this would help address one the central tenets of Compton's (2019) framework for autotelic systems, whereby users should feel 'pride and ownership' over the results from such a generative system.

conclusion

Ultimately, this thesis, centered on the development of adam, a novel design tool harnessing generative machine learning for autotelic CAD creation, signifies a transformative step in the evolution of CAD interfaces. By integrating natural language processing with traditional CAD methodologies, adam presents a paradigm shift from complexity to accessibility, making CAD tools more intuitive and approachable for a broader audience, including casual or novice designers. The research underscores the potential of generative machine learning, to democratize the design process, making 3D design accessible to those without extensive CAD knowledge.

This exploration, while focused on casual creators, lays the groundwork for further advancements in CAD systems. Future studies could delve into extensive usability studies, refine AI integration for broader application domains, and explore industrial-scale software deployment within the broader CAD ecosystem. The ultimate goal is to create CAD interfaces that function as efficient design collaborators, seamlessly aligning with human thought processes and communication styles.

In essence, adam aims to be a step towards realizing a vision where CAD tools amplify cognitive productivity, transforming from mere computational tools to interactive partners in the design process. This aligns with the vision of computers as “bicycles for the mind,” and Papert’s notions of “‘Constructionism”, in turn augmenting human creativity and learning capabilities. As CAD interfaces continue to evolve, they hold the promise of fostering a more inclusive and creative design community, breaking down barriers and enabling a wider range of individuals to contribute their unique perspectives and ideas to the world of design.



bibliography

Achlioptas, P., Diamanti, O., Mitliagkas, I., & Guibas, L. (2018). Learning Representations and Generative Models for 3D Point Clouds

Audry, S. (2021). Art in the age of machine learning. ArtBreeder, MIT Press.

Biles, J. A. (2003). GenJam in perspective: a tentative taxonomy for GA music and art systems. Leonardo, 36(1), 43–45.

Bolt, R. A. (1979). “Put-That-There”: Voice and Gesture at the Graphics Interface, Architecture Machine Group, Massachusetts Institute of Technology.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. (2021). On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258.

Compton, K., & Mateas, M. (2015). Casual Creator, International Conference on Innovative Computing and Cloud Computing.

Compton, K., & Mateas, M. (2019). Casual Creators: Defining a Genre of Autotelic Creativity Support Systems. UC Santa Cruz.

Davis, N., Hsiao, C.-P., Popov, Y., Hsiao, C. P., Zhang, D., & Magerko, B. (2014). Building artistic computer colleagues with an enactive model of creativity. In: Proceedings of the fifth international conference on computational creativity. The International Association for Computational Creativity, pp 38–45.

Ebbecke, T. (2022). Pixel Alchemist.

Haleem, A., Javaid, M., & Singh, R. P. (2022). An era of ChatGPT as a significant futuristic support tool: A study on features, abilities, and challenges.

Hoffman, G., & Weinberg, G. (2010). Shimon: an interactive improvisational robotic marimba player. In: CHI’10 extended abstracts on human factors in computing systems. ACM, New York, pp 3097–3102.

Huang, F., Schoop, E., Ha, D., Nichols, J., & Canny, J. (2021). Sketch-Based Creativity Support Tools Using Deep Learning. Artificial Intelligence for Human Computer Interaction: A Modern Approach, 379-415.

Hutten, K. (2021). “The right level of abstraction” https://learn.cadhub.xyz/blog/right-level-of-abstraction/

Jun, H., & Nichol, A. (2023). Shap-E: Generating Conditional 3D Implicit Functions.

Kerbl, B., Kopanas, G., Leimkühler, T., & Drettakis, G. (2023) 3D Gaussian Splatting for Real-Time Radiance Field Rendering.

Krull, F. (1994). The Origin of Computer Graphics within General Motors. IEEE Annals of the History of Computing, Fall 1994 (Vol. 16, No. 3).

Lee, S., & Yan, J. (2016). The impact of 3D CAD interfaces on user ideation: A comparative analysis using SketchUp and Silhouette Modeler, 120-749, Yonsei University.

Liu, C., Lu, S., Chen, W., Jiang, D., Svyatkovskiy, A., Fu, S., Sundaresan, N., & Duan, N. (2023): Code Execution with Pre-trained Language Models.

Liu, H., Li, C., Wu, Q., & Lee, Y. J. (2023): Visual Instruction Tuning.

Liu, H., Li, C., Li, Y., & Lee, Y. J. (2023): Improved Baselines with Visual Instruction Tuning.

Lubart, T. (2005). How can computers be partners in the creative process: classification and commentary on the special issue. Int J Hum Comput Stud, 63(4), 365–369.

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., & Ng, R. (2020). NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In ECCV.

Mothersill, P., & Bove, V. M. (2015). The EmotiveModeler: An Emotive Form Design CAD Tool. In Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA ‘15). Association for Computing Machinery, New York, NY, USA, 339–342. https://doi.org/10.1145/2702613.2725433.

Norman, D. A. (2013). The design of everyday things. MIT Press.

Papert, S. (1980). Mindstorms: Children, Computers, and Powerful Ideas. New York, NY: Basic Books, Inc.

Papert, S., & Harel, I. (1991). Situating Constructionism. In S. Papert & I. Harel (Eds.), Constructionism (pp. 1-11). Norwood, NJ: Ablex Publishing Corporation.

Po, R., Yifan, W., Golyanik, V., Aberman, K., Barron, J. T., Bermano, A. H., Chan, E. R., Dekel, T., Holynski, A., Kanazawa, A., Liu, C. K., Liu, L., Mildenhall, B., Nießner, M., Ommer, B., Theobalt, C., Wonka, P., & Wetzstein, G. (2023). State of the Art on Diffusion Models for Visual Computing.

Poole, B., Jain, A., Barron, J. T., & Mildenhall, B. (2022). DreamFusion: Text-to-3D using 2D Diffusion. arXiv (2022).

Prince, M. D. (1971). Interactive Graphics for Computer-Aided Design. Addison Wesley.

Rubin, R. (2023) The Creative Act: A Way of Being.

Saadi, J. I., and Yang, M. C. (February 20, 2023). "Generative Design: Reframing the Role of the Designer in Early-Stage Design Process." ASME. J. Mech. Des. April 2023; 145(4): 041411.

Samad, T. (1986). Towards a Natural Language Interface for CAD, The Kluwer international series in engineering and computer science; natural language processing and machine translation, Carnegie-Mellon University.

Shneiderman, B. (2007). Creativity support tools: Accelerating discovery and innovation. Communications of the ACM, 50(12), 20–32. https://doi.org/10.1145/1323688.1323689.

Shneiderman, B., Fischer, G., Czerwinski, M., Resnick, M., Myers, B., Candy, L., Edmonds, E., Eisenberg, M., Giaccardi, E., Hewett, T., et al. (2006). Creativity support tools: Report from a US National Science Foundation sponsored workshop. International Journal of Human-Computer Interaction, 20(2), 61–77.

Shneiderman, B. (2022). Human-Centered AI. Oxford University Press.

Soegaard, M. (2015, July 5). Affordances. Interaction Design Foundation - IxDF. https://www.interaction-design.org/literature/book/the-glossary-of-human-computer-interaction/affordances.

Steinfeld, K. (2023). Architecture after AI. The Architect's Newspaper. Retrieved from http://ksteinfe.com/hype/med-2023-arch-newspaper-austin-symposium.html

Steinfeld, K. (2017). Dreams May Come. ACADIA 2017 | DISCIPLINES + DISRUPTION. University of California, Berkeley

Stewart, J. R., Gapenne, O., & Di Paolo, E. A. (eds) (2010). Enaction: toward a new paradigm for cognitive science. MIT Press, Cambridge, MA.

Stiny, G. (1989). What Designers Do that Computers Should. In M. McCullough, W. J. Mitchell, & P. Purcell (Eds.), Proceedings of the CAAD Futures 89 Conference. (Prepublication edition.) Cambridge, MA: Harvard Graduate School of Design.

Sutherland, I. E. (1963 and 2003). Sketchpad: A man-made graphical communication system, Ph.D. Thesis. Courtesy Computer Laboratory, University of Cambridge.

Tancik, M., Weber, E., Ng, E., Li, R., Yi, B., Wang, T., Kristoffersen, A., Austin, J., Salahi, K., Ahuja, A., Mcallister, D., Kerr, J., & Kanazawa, A. (2023). Nerfstudio: A Modular Framework for Neural Radiance Field Development.

Toward a Machine with Interactive Skills, from Understanding Computers, (1986): Computer Images, Time-Life Books,

Varela, F. J., Rosch, E., & Thompson, E. (1991). The embodied mind: cognitive science and human experience. MIT Press, Cambridge, MA

Weisz, J. D., Muller, M., He, J., & Houde, S. (2023). Toward General Design Principles for Generative AI Applications. In . ACM, New York, NY, USA, 16 pages.

Xue, S., Kou, X., & Tan, S. T. (2009). Natural Voice-Enabled CAD: Modeling via Natural Discourse. Computer-Aided Design & Applications, 6, 125-136. 10.3722/cadaps.2009.125-136

photo credits

Winking Girl "Nefertite" | Ivan Sutherland; "Sketchpad, A Man-Machine Graphical Communication System", 1963 PhD thesis

"A new kind of bicycle" | Apple Computer advertisement; Wall Street Journal, August 13, 1980

"Put-That-There" | Early project by Chris Schmandt (1979); MIT Media Lab Speech Interface group video collection

Seymour Papert | Mathematics C3 SWEET Algebra 1, Text for the first two years of secondary school, first edition 2014; publisher Matematicamente.it

Special mention and thanks to all those who’ve helped me on this journey: my beloved family, my professors Eric Paulos, Kyle Steinfeld, Hugh Dubberly, Yoon Bahk, and finally, my peers in Studio, who inspire me to learn, build, invent, and have been a constant source of joy over the last one and a half years. Shoutout FBAR & The Mint.