The Daisy Vase Project - A Unique Vase for Everybody

Heiko Drewes
LMU Munich
Munich, Germany
heiko.drewes@ifi.lmu.de

Figure 1: Random generated vases from the Daisy Vase Project

Abstract

The Daisy Vase Project is somewhere between science, commerce, and art. It aims at randomly creating vases that are printable with a 3D printer. The program written for the project outputs vases meshes and some of the vases are beautiful and have an unseen design. Besides creating beautiful designs, the goal of the Daisy Vase Project is also to program a generator that can generate so many different vase designs that every person in this world can have a unique vase. The presented vase generator has this capability. Developing the algorithms and the user interface is a scientific task while the question of what is beauty belongs to art. Presenting computer-generated suggestions would provide a new shopping experience if used for commercial purposes.

CHI EA '25, Yokohama, Japan

ACM ISBN 979-8-4007-1395-8/25/04
https://doi.org/10.1145/3706599.3716224

CCS Concepts

• **Human-centered computing** → **Empirical studies in interaction design**; • **Applied computing** → **Marketing**; *Computer-aided manufacturing*; **Consumer products**.

Keywords

random vases, computer-generated design, computer-controlled manufacturing, 3D printing

ACM Reference Format:
Heiko Drewes. 2025. The Daisy Vase Project - A Unique Vase for Everybody. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA '25), April 26–May 01, 2025, Yokohama, Japan.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3706599.3716224

1 Introduction

Since the invention of mass production, we can produce large quantities of items for low costs. This created wealth for the masses as before only privileged persons could afford the unique and expensive item. The price for this is stereotypes, which means we all own the same items. Nowadays, with computer-aided manufacturing, we can produce unique and individual items with the same efficiency as mass production.

The problem, however, is who will design these items. There are not enough designers to create unique designs for everybody and the customers typically cannot create their personal designs. One possible solution for this problem introduced here is the concept of *Inspired Creativity* where the customer chooses from computer-generated suggestions. Applications of *Inspired Creativity* are the creation of textiles, shoes, houses, etc.

With the rise of 3D printers, a new output media is available and the project presented here as an example of *Inspired Creativity* is the generation of vases printable with a 3D printer. Vases are part of the cultural heritage of nearly all cultures and have existed since ancient times. They can also be artwork. However, most people cannot operate a 3D editor and therefore can not create a 3D vase model themselves, but all people have a taste and can tell which vase they like.

A scientific evaluation of the concept of *Inspired Creativity* is difficult as any study primarily evaluates the quality of the generator and not the concept. For this reason, the paper does not present a study but discusses the topic, presents insights gained during the project realization, and raises open questions for future research.

The Daisy Vase Project is somewhere between art, science, and commerce. The project is art because it is a creative approach to exploring design spaces and finding novel designs and beauty. The project is also science, in particular computer science, as it deals with generative algorithms and human-computer interaction. Finally, the project has relevance for commerce as it provides new shopping experiences and demonstrates how to create and present customized products.

2 The Concept of Inspired Creativity

2.1 Motivation

The problem of stereotypes does not only exist in the production of physical objects but also in the creation of digital documents. For many letters, it is possible to tell which text processor was used and the same is true for presentation slides. The reason for this is not a lack of options to individualize the document but the *power of the default settings*. Most software applications provide a large amount of powerful options that can overwhelm the users. For experts using the application daily, this might not be a problem, however, for novice users and people who occasionally use the software this is a problem. As computer-aided manufacturing requires a digital description of the physical object to produce the solution lies in the creative software tool.

The concept of *Inspired Creativity* allows for creating data describing an item, either physical or virtual. The vases created by the project are first created as mesh data and then printed to get a physical object. However, it is possible to apply the concept to purely virtual objects such as objects for a computer game or a presentation layout.

The main problem for creative work is getting over the empty sheet of paper or blank canvas. Users who do not have a clear vision of how the result of their creation should look or can not imagine the effect of the available options typically start experimenting with the options and choosing them by chance. The concept of *Inspired Creativity* automatizes the process of experimenting with the available options and provides the user with suggestions. The generation of suggestions by the computer is much quicker than manual creation, leading to a higher efficiency in terms of needed time. Additionally, the quick generation allows for inspecting more suggestions and making it very likely to get a better result.

2.2 Applications

One example of *Inspired Creativity* is the creation of pictures supporting artwork and design. Some years ago this was done with random algorithms. The internet offers many Random Art projects[1]. Nowadays generative AI has become popular for creating pictures[2].

It is also possible to write suggestion generators for presentation software templates, letter layouts, or business cards and generative AI can even write texts as suggestions[3].

Another field of application is the generation of 3D objects. Developers of 3D computer games and creators of animated movies have to create many 3D objects such as the terrain, buildings, NPCs (non-playable characters), and more. The objects must differ from each other. The NPCs should not all look the same and there should not be only one type of house in a village. Editing all the 3D objects from scratch is a lot of work and *Inspired Creativity* can help to reduce this work. It is worth mentioning that one of the very few related work papers suggests creating the scene lightning and also animations with random suggestions (see Section 3).

A promising application of *Inspired Creativity* is online shops. Some shops offer customizable products, for example, sports shoes[4]. While functional customization, such as customization of a laptop by choosing processor and memory, seems to be not suitable, design customization seems to be perfectly suited. Customers of an online shop typically do not have an education as a designer. With *Inspired Creativity* it is possible to offer a customer suggestions until she or he finds something matching the personal taste. At the moment companies produce a wide range of designs to offer customers a choice which finally leads to overproduction. With *Inspired Creativity* and production-on-demand it is possible to offer even more choices without overproduction and this makes production more sustainable.

2.3 Generation of Designs

For the concept of *Inspired Creativity*, the quality of the suggestion generator is crucial. This raises the question of how to judge the quality of a suggestion generator. For the answer, it is necessary to look closer at design spaces.

[1]https://www.random-art.org/, https://www.behance.net/gallery/70048085/Random-Art, https://www.infimum.dk/HTML/randomArt.html, and more
[2]https://www.midjourney.com/, https://openai.com/index/dall-e-3/, and more
[3]https://openai.com/chatgpt/, https://copilot.microsoft.com/
[4]https://www.nike.com/nike-by-you

2.3.1 Formal, Meaningful, and Generatable Design Spaces. In the context of machine generation randomness plays a central role. However, randomness alone is not sufficient. The attempt to create bitmaps by dicing out the color for every pixel will create colored versions of pepper and salt, although theoretically, this approach can produce every possible bitmap.

The first demand on a random generator is the output of valid data. When we create something on the computer, a text document, presentations, graphics, or 3D objects we can save the result of our creation, which finally means that we created a file. All possible valid files define a formal design space (fDS). In general, such a design space has an enormous amount of members. The amount of members of the formal design space which makes sense, which are the files that have some meaning for us, is also very large and is called meaningful design space (mDS) here. While it is possible to define the fDS exactly (by syntax), the mDS does not have a sharp definition as the concept of being meaningful is vague (semantics). All possible randomly generated designs form the generatable design space (gDS). See Figure 2 for a visualization of these spaces.

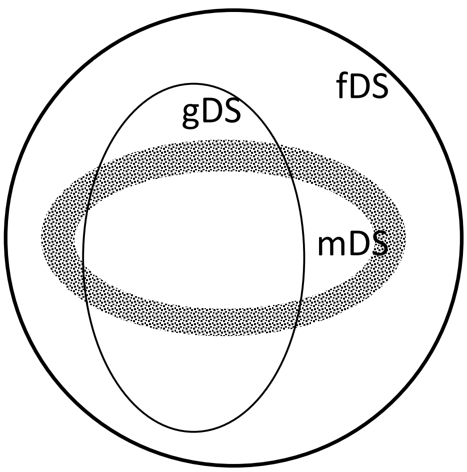

Figure 2: The formal design space fDS, the meaningful design space mDS and the generatable design space gDS.

In general, it is easy to program a random generator where gDS is equal to fDS. However, such a random generator is not very useful, as the ratio of the number of meaningful designs and the formally possible designs is very small. This means such a random generator will mostly create garbage suggestions. To specify the quality of a random generator for designs it is helpful to define two criteria, the coverage c and the hit rate h:

$$c = |gDS \cap mDS|/|mDS| \tag{1}$$

$$h = |gDS \cap mDS|/|gDS| \tag{2}$$

The coverage c is the percentage of meaningful suggestions that the random generator can create. The formula for the hit rate is valid under the assumption of equal probability for each member of the gDS. The hit rate h is the percentage of meaningful designs in the output of the random generator. In the case of an unequal probability distribution, the hit rate is the probability of a suggestion from the mDS.

A useful random generator should have a hit rate $h > 0.01$. In other words, among 100 generated designs there should be at least one meaningful design. Otherwise, the experience for the user becomes frustrating and the user will give up soon. Of course, it is possible to demand that the hit rate $h > 0.1$, which means one useful design within 10 suggestions. In general, it is always good to have a high hit rate and the minimum value depends on the task. The bigger the effort of creating the design manually, the more suggestions a user is willing to inspect.

Although there is no sharp definition for the mDS and therefore it is difficult to calculate the hit rate, it is easy to measure the hit rate h in an experiment for an existing random generator. In contrast, the coverage c is difficult to estimate and only good for qualitative statements.

2.3.2 Demands for Suggestion Generators. There are some demands for suggestion generators:

- sufficient number of different suggestions.
- acceptable hit rate.
- obeying basic constraints.
- good coverage of the meaningful design space.

The first demand is essential as a suggestion generator that can create only a few suggestions will repeat itself and can not keep the promise of unique designs. The number of possible suggestions should be close to infinity. If a suggestion generator promises a unique design for everybody, it means that the number of suggestions should be much bigger than the world population.

An acceptable hit rate is mandatory. The hit rate determines how many suggestions a user has to inspect until she or he finds a usable suggestion. Users do not have endless patience and give up if the chance of finding something is low. The number of suggestions a user is willing to inspect depends on how much editing work can be saved, how important it is to find something, and also on the personality of the user.

A suggestion generator should obey constraints because an offered suggestion violating constraints is not usable and lowers the hit rate. However, a user can recognize a constraints violation and consequently do not choose this suggestion. A good coverage of the meaningful design space is good for the users' acceptance of the generator. If certain suggestions will never be suggested the generator's usefulness is limited.

2.4 The Human in the Loop

A shopping website will most probably be more successful with a good suggestion generator. However, it is not clear how to program a good suggestion generator. The demand for the generator is that it creates a suggestion suiting the personal taste of a user and, as users have different tastes, there is no optimal suggestion, in this case, no optimal vase. This is where the human aspect comes in and this is why programming suggestion generators is a field of HCI.

Random-based machine generation has the problem of not always generating good output but also generating garbage. For this reason, it needs a human in the loop to decide which design is garbage and which one is beautiful. Consequently, the demand for a generator is not to create only beautiful designs but to create designs where everybody can find a design satisfying their personal taste. However, although the concept of beauty is subjective, there is also an objective aspect of beauty. This becomes obvious when the generator creates ugly suggestions that nobody will like. This leads to a demand for design generators: the ratio of beautiful design suggestions and the total number of suggestions, the hit rate, should not be to small. In other words, a generator should create enough beautiful suggestions so that a user can find her or his design within a reasonable number of generations.

At the moment generative artificial intelligence (AI) is very popular. Generative AI works by providing a random seed to a neural network. The outcome depends strongly on the training data set used to train the neural network and can be seen as associations based on the training data set. The alternative for writing a design generator is a constructive approach using parameterized models with random values for the parameters. Typically, such an approach uses many parameters that guarantee by the power of combinatorics that the design output space is large enough. The advantage of the constructive approach is that it does not need a training data set and as there is no data set for vases, the Daisy Vase Project uses the constructive approach.

3 Related Work

Programmers are creative people and creating a program that creates something is an appealing idea for them. Having access to output media like screens, printers, plotters, and speakers inspired some programmers to use it for art already in the very beginning of computer technology[5] [5, 6, 13].

As the implications of *Inspired Creativity* are not widely researched yet there is only little directly related work. The related work that suits here best is Pepperell's paper from 2002 on Computer Aided Creativity [12]. His paper supports the random approach used here. Pepperell's paper introduced the term 'Computer Aided Creativity'. At that time the adjective 'aided' was still popular and appeared in terms of CAD (computer-aided design) and CAM (computer-aided manufacturing). The popularity of the adjective 'aided' went down and nowadays the term *Inspired Creativity* seems to suit better. Another term in this context is 'gallery' which focuses more on the presentation of the suggestions than on its creation. Marks et al. [11] used the term 'Design Galleries' already in 1997. Lee et al. presented in 2010 an approach for designing web pages with interactive example galleries, where they stated that "examples may benefit novices more than experienced designers" [9].

The principle of *Inspired Creativity* is not only applicable to products but works in most cases where the users have to adjust settings in a huge parameter space. Marks et al. showed this for setting lights in a 3D computer graphics scene and also for the creation of animations [11]. Marks et al. called their paper "A General Approach to Setting Parameters"which is exactly what is proposed here and together with the early date of their publication they deserve the honor of being the inventors of the concept of *Inspired Creativity*.

[5]http://www.herbert-w-franke.de/WsFr5Korr.htm

However, there is a lot of literature that is somehow related to the topic. There is a literature review by Shi et al [16] with 93 papers from 2007 to 2022. Most literature in this review is restricted to the relation of designers and artificial intelligence only. The approach presented here, however, aims to users without design education, and the concept of *Inspired Creativity* is not restricted to generative artificial intelligence.

The early generating approaches used random values for the parameters. The recent breakthrough in machine learning opens the door for generative AI that creates texts[6], pictures[7], music, and videos. Attempts to create interactive applications of generative models where the users can control generation parameters, for example by a set of sliders [3], lead to the same problems identified by Marks et al [11].

The combination of online shops and computer-aided manufacturing allows for offering customizable products. The concept of *Inspired Creativity* can help customers designing their customized products. The fact that customizable products generate immense value for customers has been proven by Brock already in 1968 where he pointed out that both the scarcity of products and the effort put into obtaining products influence perceived product value [2]. In 2004 Franke et al. found that customers were willing to pay more than twice the price of an off-the-shelf watch for a self-designed watch with the same technical specifications [7]. The customization toolkit used in the studies focused on visual customization such as selecting watch faces and straps [7].

In 2006 Schreier et al. [14] found an average value increment of 134% compared with non-customized alternatives. They also derived perceived benefits for the customer when engaging with product customization toolkits. Two of these benefits are "perceived uniqueness" [14, p. 323], meaning the customer has a feeling of designing and buying a unique product no one else can buy, and "pride of authorship" [14, p. 324], meaning the customers create a special relation to the product because it is the result of their creativity. This aligns with the assumption that the desire to express one's individuality is a big factor that drives design customization. Another benefit is the "process benefit" [14, p. 324], stating that using the customization toolkit itself is a fun experience for users and is part of what increases the perceived product value. The "pride of authorship" benefit has been examined in the paper "The 'I Designed It Myself' Effect in Mass Customization" by Nikolaus Franke et al. [8] in 2010. With the rise of generative AI, the question of authorship of computer-generated suggestions became an important research question [4, 10].

An example of an interactive system for customized clothing design was given in 2020 by Zhu et al. [18].

The presentation of suggestions is similar to recommender systems. Sivaramakrishnan et al. [17] presented a mathematical approach to a recommender for customizable products in 2015.

4 Construction of Vases

There are many constraints for objects that can be told a vase. A vase must have a stable stand on a surface, must have a closed form without holes to keep water, and must have an opening to put

[6]https://openai.com/chatgpt/
[7]https://openai.com/index/dall-e-3/

flowers inside. Typically, vases have a form that is higher than wide as otherwise it would be a bowl. There are additional constraints that come from the production process. The 3D printer used for this project adds material in a layer on top of the layer below. It cannot print in mid-air and the slope of the vase wall can not exceed a certain limit.

In the Daisy Vases application, the construction of a vase starts with the random choice of the base area's shape, which means a circle, a triangle, a square, a star shape, etc., with a vertical wall. The next step is the application of a random number of spatial transformations, such as twisting, shearing, applying decorative elements, etc., all chosen by chance (see Figure 3). The transformations include the joining of two vases which gives the construction description of a tree-like structure. The final step is adding a foot and a head.

The implementation of the vase generator was done with Visual Studio using MFC, OpenGL for displaying on the desktop, and WebGL for displaying on the internet. The source has more than 10.000 lines of code representing some years of work. Figure 6 shows a screenshot of the self-developed software.

The details of the creation algorithm are very complex. It uses two different coordinate systems, polar and Cartesian coordinates, downward and upward recursion, and over twenty geometrical transformation groups with up to four integer and four floating point parameters. The challenges besides programming skills and extensive mathematical calculations were obeying the constraints mentioned above and balancing the probabilities with which a certain transformation occurs. Especially balancing the probabilities is challenging. With badly chosen probabilities for the occurrence of functions and the values of its parameters, the generator produces either very similar or many ugly vases. Testing the effect of changed probabilities requires the generation and inspection of many vases to exclude good or bad luck.

The vase generator outputs 3D meshes as OBJ files which is the input for a slicer application, in this case CURA[8], which outputs g-code to control the 3D printer (see Figure 7).

5 Results

The vase creator creates a new suggestion in much less than a second with just a mouse click. On the internet (http://daisyvases.com) it takes a little bit longer as transferring the mesh data takes additional time. Compared with the author's previous random art projects, for example, generating 2D images, the ratio of usable designs to unacceptable designs of vases is pretty good. The exact value of this ratio depends on how picky the selecting person is, but roughly one of ten suggestions is acceptable. Figure 4 shows some selected examples.

The number of possibly generated designs is enormous. Many created designs are surprising and it is very unlikely that these designs could have been created in another way. Additionally, many vases show floral patterns when looking at them from the top (see Figure 5). This property of the vase generator's algorithms was not programmed by intention but suited perfectly to vases.

3D printers are slow and therefore the printed vases are only 6 cm high. Even in this small size, a vase print took between 3 and 14 hours. A vase with 6 cm height is only good for daisies and this is where the name of the project comes from (see Figure 8). Despite the project name, the Daisy Vase generator can create vases at any size and resolution as the vase definition is based on mathematical functions. Currently, there are nearly 200 printed vases that cost more than 1000 hours of printing time. There are several thousand saved vase files that are waiting to be printed.

6 Random Generation versus Generative AI

Throwing dice for creating parameter values is one option for generating the suggestions needed for *Inspired Creativity*. Generative AI which became very popular in the last years is another option. Generative AI also uses random values as a seed but not directly as parameter values and the results are quite different from the results of random generation. This raises the question of which method suits best for *Inspired Creativity*. The answer depends on what should be generated.

Random generation typically suffers from low hit rates while generative AI seems to produce better hit rates but needs a huge training data set. Additionally, the data set has to be in 'production parameters'. For the example of a customized sports shoe mentioned above a nice bitmap is not sufficient - it needs the data to control the manufacturing machine and to get these data from a picture is challenging. The situation for other examples is similar. In the case of layout generation it does not only need a preview bitmap how the layout will look like but also the corresponding file for the layout software. Using generative AI for *Inspired Creativity* means to creating a training data set in 'production parameters' to train the neural network and software that creates a preview for the user from the 'production parameters'.

There are many applications for face creation as there is a big demand for generated faces as avatars. Some approaches use generative AI [9] while other approaches use random generation [15]. For realistic faces the generative AI may provide better results but mostly as bitmaps and not as 3D meshes.

In some situations, such as the task of setting lights in a 3D computer graphics scene [11] mentioned in Section 3, the generative AI approach does not work because there is no training data set available. Creating such a set would be too much effort and the members of the set could be used to pick suggestions which means that the neural network is not necessary.

Generative AI typically expects a prompt describing the desired result which means that the user of generative AI normally has a result in the mind. This is the opposite of the concept of *Inspired Creativity* where the user seeks inspiration. However, the difference between both approaches is fuzzy. The prompt for the Generative AI could be very vague and therefore generate suggestions. On the other hand, the random generation could offer restrictions to certain themes and require some imagination in advance.

7 Conclusions

The vase generator seems to be powerful enough to keep the promise to supply every person on this planet with an individual design. The fascinating aspect, however, is that the vase generator

[8] https://ultimaker.com/software/ultimaker-cura/

[9] https://this-person-does-not-exist.com/en

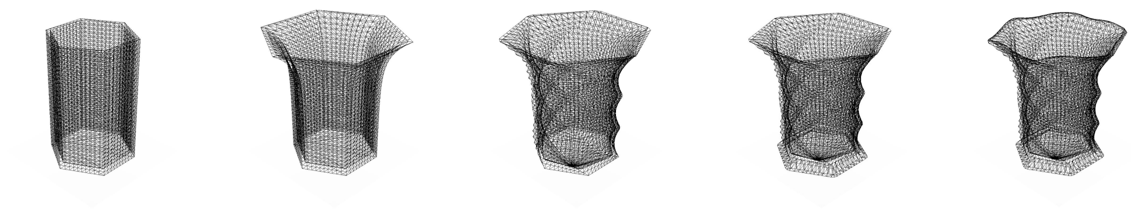

Figure 3: Construction of a vase - choosing a hexagonal base area - transform exponential over height - twist - add a foot - add a head

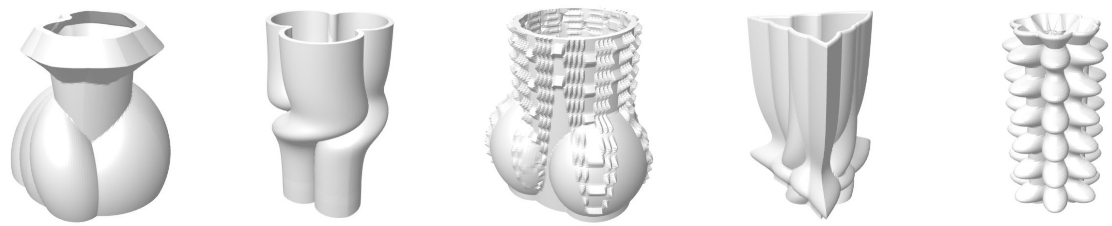

Figure 4: Examples of randomly generated vases

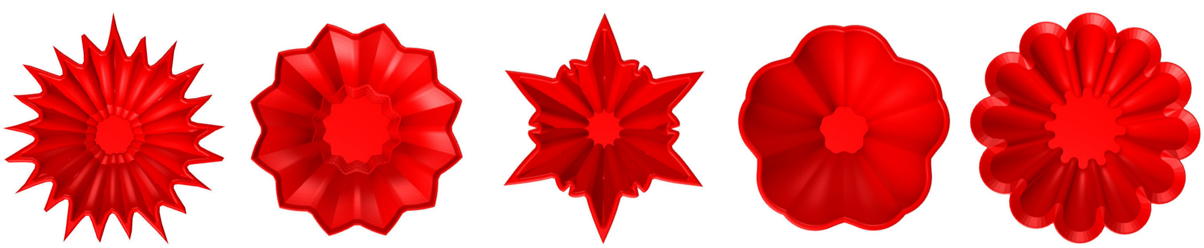

Figure 5: View from the top for some vases. As a surprise the random generator creates floral patterns.

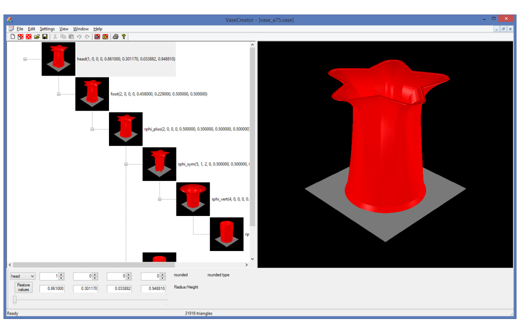

Figure 6: Screenshot of the self-developed application for vase generation

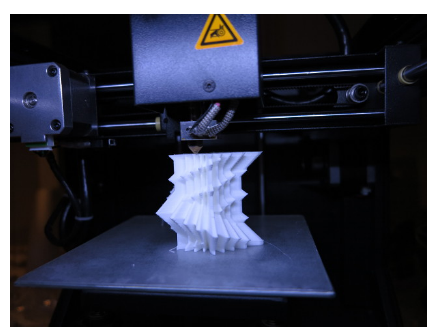

Figure 7: Printing of a vase

creates unexpected designs even beyond the programmer's imagination. The Daisy Vase Project creates unseen designs that would most probably not be created with other methods.

The vase creator software also offers the possibility to choose transformation and parameters manually (see Figure 6) and the author used this for manual creation. The manually created vases are appealing but the randomly created vases are different in character. If another programmer writes another vase creator program that generator's output will likely create vases with another character. In consequence, programming design generators depend on the programmer and can be seen as art.

The Daisy Vase Project did not only generate interesting vases but also a lot of interesting questions. One of these questions is how

Figure 8: 3D printing is time consuming and to speed up the process most vase prints done are only 6 cm high – with this size the vases are only good for daisies.

to judge the quality of a generator. Is the vase generator presented here a good one? How to find out whether another generator creates better output? The hit rate and the coverage introduced above are quality criteria but are definitely not enough to describe the quality of a suggestion generator completely. With a rising number of competing generative AIs the question of the generator's quality becomes essential, especially if the intention of programming the generator is to earn money.

Another question is how to present the suggestions. At the moment the creation software presents suggestions one by one. Most current generative AIs offer four pictures as suggestions. It would be possible to display up to 100 suggestions at once on a big screen. This would be advantageous if it is possible to spot a favored design with preattentive perception.

Another question is which interaction possibilities beyond selection should be given to the users. One possibility would be the provision of themes like 'classic' or 'fancy'. Another possibility is to offer breeding which means the user selects 'almost good' suggestions and asks for further suggestions based on that choice as known from evolutionary creative algorithms [1]. Finally, users typically ask for the possibility of post-editing.

An interesting question, especially for commercial aspects, is who holds the copyright. Possible answers are the programmer, the owner of the hardware where the creative software runs, the person who selected the suggestion, the one who claims it first, or, as chance has no owner, the whole of mankind. It seems that the general opinion on the copyright for the output of generative AI is that there is no copyright. A person who spent a lot of time looking at a large number of designs might see this differently, especially if that person did some post-editing. The question of authorship is closely related to the copyright and it is not clear whether a user of *Inspired Creativity* is an author or only a lucky finder. However, in the context of a shopping experience, both can make a user happy.

The final question that intrudes on oneself is: what is beauty? The answer seems to be as difficult as the question of what is art. This is a pity as knowing what beauty is would make it easier to program creative software that generates only beautiful designs.

References

[1] Peter J Bentley and David W. Corne (Eds.). 2002. *Creative Evolutionary Systems.* Morgan Kaufmann, San Francisco, 547–576. doi:10.1016/B978-155860673-9/50063-X

[2] Timothy C. Brock. 1968. Implications of Commodity Theory for Value Change. In *Psychological Foundations of Attitudes.* Academic Press, 243–275.

[3] Hai Dang, Lukas Mecke, and Daniel Buschek. 2022. GANSlider: How Users Control Generative Models for Images using Multiple Sliders with and without Feedforward Information. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22).* Association for Computing Machinery, New York, NY, USA, Article 569, 15 pages. doi:10.1145/3491102.3502141

[4] Fiona Draxler, Anna Werner, Florian Lehmann, Matthias Hoppe, Albrecht Schmidt, Daniel Buschek, and Robin Welsch. 2024. The AI Ghostwriter Effect: When Users do not Perceive Ownership of AI-Generated Text but Self-Declare as Authors. *ACM Trans. Comput.-Hum. Interact.* 31, 2, Article 25 (feb 2024), 40 pages. doi:10.1145/3637875

[5] Herbert W. Franke. 1984. *Computergrafik-Galerie. Bilder nach Programm - Kunst im elektronischen Zeitalter.* DuMont, Köln.

[6] Herbert W. Franke. 1985. *History of Computer Art.* Springer Berlin Heidelberg, Berlin, Heidelberg, 93–152. doi:10.1007/978-3-642-70259-4_2

[7] Nikolaus Franke and Frank Piller. 2004. Value Creation by Toolkits for User Innovation and Design: The Case of the Watch Market. *Journal of Product Innovation Management* 21 (11 2004), 401 – 415. doi:10.1111/j.0737-6782.2004.00094.x

[8] Nikolaus Franke, Martin Schreier, and Ulrike Kaiser. 2010. The "I Designed It Myself" Effect in Mass Customization. *Management Science* 56, 1 (2010), 125–140.

[9] Brian Lee, Savil Srivastava, Ranjitha Kumar, Ronen Brafman, and Scott R. Klemmer. 2010. Designing with Interactive Example Galleries. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) *(CHI '10).* Association for Computing Machinery, New York, NY, USA, 2257–2266.

[10] Florian Lehmann, Niklas Markert, Hai Dang, and Daniel Buschek. 2022. Suggestion Lists vs. Continuous Generation: Interaction Design for Writing with Generative Models on Mobile Devices Affect Text Length, Wording and Perceived Authorship. In *Proceedings of Mensch Und Computer 2022* (Darmstadt, Germany) *(MuC '22).* Association for Computing Machinery, New York, NY, USA, 192–208. doi:10.1145/3543758.3543947

[11] J.W. Marks, B. Andalman, P.A. Beardsley, W.T. Freeman, S.F. Frisken, J. Hodgins, T. Kang, B. Mirtich, H. Pfister, W. Ruml, K. Ryall, J. Seims, and S. Shieber. 1997. Design Galleries: A General Approach to Setting Parameters for Computer Graphics and Animation. In *ACM SIGGRAPH.* 389–400.

[12] Robert Pepperell. 2002. Computer Aided Creativity: Practical Experience and Theoretical Concerns. In *Proceedings of the 4th Conference on Creativity & Cognition* (Loughborough, UK) *(C&C '02).* Association for Computing Machinery, New York, NY, USA, 50–56.

[13] Hans Ronge. 1968. *Kunst und Kybernetik.* DuMont, Köln.

[14] Martin Schreier. 2006. The value increment of mass-customized products: an empirical assessment. *Journal of Consumer Behaviour* 5, 4 (2006), 317–327.

[15] Valentin Schwind, Katrin Wolf, and Niels Henze. 2017. *FaceMaker-A Procedural Face Generator to Foster Character Design Research.* 95–113. doi:10.1007/978-3-319-53088-8_6

[16] Yang Shi, Tian Gao, Xiaohan Jiao, and Nan Cao. 2023. Understanding Design Collaboration Between Designers and Artificial Intelligence: A Systematic Literature Review. *Proc. ACM Hum.-Comput. Interact.* 7, CSCW2, Article 368 (oct 2023), 35 pages. doi:10.1145/3610217

[17] Aravind Sivaramakrishnan, Madhusudhan Krishnamachari, and Vidhya Balasubramanian. 2015. Recommending Customizable Products: A Multiple Choice Knapsack Solution. In *Proceedings of the 5th International Conference on Web Intelligence, Mining and Semantics* (Larnaca, Cyprus) *(WIMS '15).* Association for Computing Machinery, New York, NY, USA, Article 16, 10 pages.

[18] Xinjuan Zhu, Xuefei Li, Yifan Chen, Jingwei Liu, Xueqing Zhao, and Xiaojun Wu. 2020. Interactive genetic algorithm based on typical style for clothing customization. *Journal of Engineered Fibers and Fabrics* 15 (2020), 1–9.