6th International Symposium on Academic Makerspaces
Image Classification and Prompt Engineering to Optimize Text-to-Image in Laboratory and Makerspace Design

ISAM 2022 Paper No.: 50

Regal Leftwich[1]
[1]Regal Leftwich; Science and Technology, CannonDesign; rleftwich@cannondesign.com

Introduction

Text-to-image synthesis and image-text contrastive learning are at the forefront of multimodal machine learning. With emerging software platforms such as DALL-E2 [1] and Imagen [2] supporting the creation of text-to-image creation, we are forming complicated relationships as designers leverage machine learning to augment the process of creation. As design, art and architecture are filtered through machine indexed matrixes and reordered into new configurations a convergence of augmented imagination emerges. Using natural language processing on trained models of images to generate renderings is a new process where requiring an understanding of command prompt engineering.

The paper documents the process of using command prompt engineering terms to guide diffusion modeling trained on open source software using Google Colab to generate images of makerspaces and laboratories. This study includes 1,128 command prompt engineering searches iteratively generated based on the analytical framework discussed in the Analysis section. Each search required a processing time over 5 minutes on a remote Google server.

An analysis of each image was conducted to evaluate and rank the effectiveness of the search terms used in relation to the output of the images for the computer generated images to demonstrate an aesthetic related to makerspaces and laboratories. These typologies are clearly recognizable and were chosen due to their dual role as utilitarian and inspirational spaces. Success in creating an image that matched the command prompt intent has been evaluated in the analysis portion.

Contribution

An increasingly complex relationship is evolving as the tools used for design begin to outpace the designers capacity for creative solutions. This work is a demonstration of the augmentation of the design process with machine learning focused on the idea of convergence of humans and machines interacting in the creative process. Using natural language processing on trained models of images to generate renderings is a new process requiring an understanding of command prompt engineering.

Command prompt engineering for diffusion modeling requires trial and error practice for how a string of words will be graphically translated by machines. Today's designers are still approaching computational design as a drawing exercise and not as a poetic endeavor that can harness the potential of imagination with large data sets of images and open the design process to anyone with an understanding for the fundamentals of command prompt engineering.

Fig.1 Text-to-Image Generated Images

Fig. 1 showcases some of the images created using this process. From top left to bottom right shows the evolution from some of the early search terms to later using command prompt engineering to generate photo realistic makerspaces.

The ability to index and have instantaneous accessibility to these data sets differentiates machine learning from human thinking, however comprehension remains in the realm of the natural language users. The idea of developing a design process for space that is based on natural language processing helps make design ideas more accessible for anyone trying to envision space. It is from this perspective that intent of this paper is to document the process of using command prompt engineering and machine learning as a tool for the creation of makerspace and laboratory design.

Methods

Text-to-image techniques require the use of search term input into a generative model that has been trained on images. There are multiple types of generative models, for the purpose of this experiment a diffusion modeling process was used due to its accessibility with open source software.

A. *Software Process*

Fig. 2 outlines multiple types of generative models such as Generative Adversarial Networks (GAN) [3], Variational Autoencoder (VAE) [4], Flow-based models [5], and Diffusion Models [6], [7], [8].

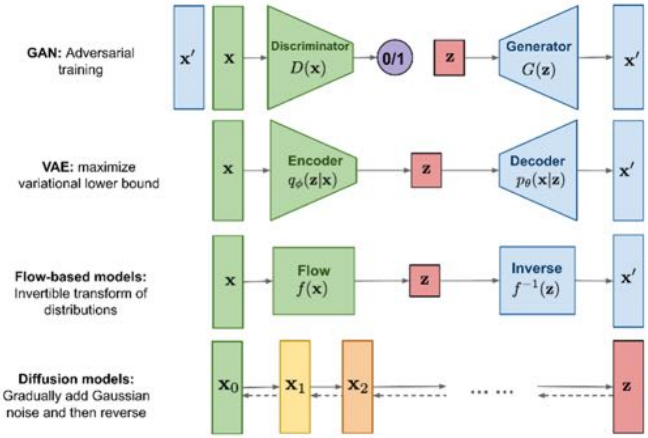

Fig.2 Different Types of Generative Models [7]

The process leverages the diffusion modeling trained on multimodal ai art models from Google Colab CLIP Guided Diffusion and VQGAN+CLIP, Latent Diffusion running on open source Disco Diffusion v5. Special thanks to Apolinario, founder of multipmodal.art for creating MindsEye beta the graphical user interface used to create the images.

B. *Command Prompt Engineering*

Images are generated from keywords entered in the command prompt. Some of the initial images generated were experimental in nature and different combinations of keywords were developed to create images that could be considered to align with the intent of the image terms.

The command prompt engineering experimentation was a process of trial and error that was largely guided by what terms seemed to work based on visual feedback.

The terms used for the command prompt could be short as one word or a very long string of text. The software utilized has some built in prompt enhancers that act as a sort of visual filter. Some examples of this would be "by Van Gogh," "oil on canvas," "unreal engine 4k," and "lens flare."

The prompt enhancers are capable of adding a stylistic side to the images and were largely not used, with the exception of prompt enhancers that referenced other images such as "trending on artstation" and "cgsociety."

The prompt enhancers that referenced other images allowed for a broader range of images that didn't rely on photographs, but do tend to create artifacts, which will be discussed in the analysis section.

C. *Data Tracking*

Keywords for each image search were stored with each image to allow for analysis of search terms with visual output.

Image Classification

The collection of 1,128 command prompt generated images have been organized and evaluated and classified based on four criteria: Composition, Artifacts, Aesthetic Character, Design.

A. *Composition*

Each generated image was ranked on composition, either as yes or no. Images were judged to have adequate composition if the output adequately conveyed the sense of a space that could be occupied or created in the real world.

Fig. 3 shows some examples of images that were generated that satisfied the conditions to have adequate "composition." Theses examples are strong examples of spaces that could possibly be photographed or mistaken as a photograph.

Fig.3 Examples of good "Composition"

On close inspection there are some reoccurring features such as reflections in strange locations, tables or other elements that aren't quite rectangular, lines that are not quite parallel with a vanishing point.

After some analysis of images it became clear that some of the images generated were hard to believe were real, but were deemed to have sufficient "composition" to be considered as representative of a space that could be created.

Fig. 4 is an example of images that were considered to have adequate "composition" but were not likely to exist in the real world, but were possible to imagine could exist or construct.

Fig.4 Examples of questionable "Composition"

The images of questionable "composition" were most often images that elicit the quality of science fiction scenes.

Fig. 5 is representative of images that do not have suitable "composition" because they do not convey spatial requirements of a real makerspace or laboratory composition.

Fig.5 Example images with no "Composition"

Images lacking a suitable "composition" often lacked perspectival depth and consistency. Examples in the previous figure show a lack of depth between foreground and background necessary to convey characteristics of a space.

B. *Artifacts*

Machine learning text-to-image generation often created artifacts of direct and indirect interpretation. As the images were sorted the presence of artifacts was documented as present or not present. Examples of "artifacts" are identified in the following figure.

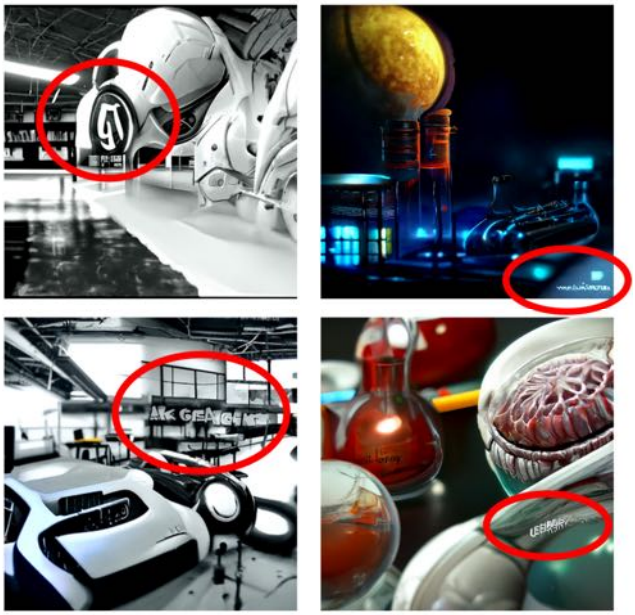

Fig.6 Example images with "Artifacts"

Fig. 6 identifies the most literal of text-to-image conversion artifacts that generated text in the images. In these examples text was added to the images based on the search terms. Starting in the top left corner, this example is from a command prompt that included "Georgia Tech" as the terms. The output added "GT" that closely resembling the university logo.

Fig. 6 image top right reveals a group of gibberish text in the area that would normally be signed by an artist. This image was generated using the prompt enhancers that pulled from existing artwork resulting in the inclusion of a text like artifact in the lower right hand portion of the image. A similar text/signature artifact can be found on the Fig. 6 bottom right hand image, although this artifact is blending in with the texture and less of a signature distinct from the portion it overlaps.

Fig. 6 lower left image has added lettering in the text-to-image conversion that appears to mimic the command prompt term "garage" as a sign in the space. This was documented as an "artifact" due to the direct relationship with the command prompt terminology.

C. *Character*

In general, if an image created had the "character" of a makerspace or laboratory it was ranked as "low," "medium" or "high." The decision focused on if the image had makerspace or laboratory like qualities, if the image did not, it was ranked "low." If the image had makerspace or laboratory like qualities and looked like a makerspace or laboratory, then it was ranked "high." If it did not look like, but had qualities of a makerspace or laboratory then it was ranked "medium."

Fig.7 Example images with low "Character"

Fig. 7 demonstrates examples of images with low "character" because they did not have sufficient qualities that evoked makerspace or laboratory like qualities.

Fig.8 Example images with medium "Character"

Fig. 8 demonstrates examples of images with medium "character" because they did have sufficient qualities that evoked makerspace or laboratory like qualities but may not have been explicitly recognizable as a makerspace or laboratory.

Fig. 9 Example images with high "Character"

Fig. 9 demonstrates examples of images with high "character" because they were explicitly recognizable as a makerspace or laboratory.

D. *Design Intent*

The images were also classified by the "design intent" raking of low, medium and high to determine how the close the text-to-image conversion was to the command prompt intent.

Images in Fig. 5 and 7 are considered having low "design intent" to a command prompt with the intent with terms like "makerspace laboratory with tables and instruments."

Fig. 8 images have medium "design intent" to a command prompt with the same terms because they display some intent of tables, makerspace, laboratory and instruments.

Fig.10 Example images with high "Design Intent"

Fig. 10 is an example of high “design intent” to a command prompt with the same terms of “makerspace laboratory with tables and instruments in a gallery”

Fig. 11 Example images with high “Design Intent”

Fig. 11 is an example of high “design intent” to a command prompt with the terms of “makerspace building factory”

Fig. 12 Example images with high “Design Intent”

Fig. 12 is an example of high “design intent” to a command prompt with the terms of “makerspace in a candy store”

Analysis

A total of 1,128 images were generated and ranked based on the criteria of Composition, Artifact, Character and Design Intent.

Table 1 accounts for the number of images for each category that ranked as passing for Composition and Artifacts present, and “high” value for Character and Design Intent.

Table 1 Table caption

Criteria	"Pass" or "High"	
	Count	Percentage
Composition	794	70.39%
Artifacts	307	27.22%
Character	475	42.11%
Design Intent	793	70.30%

The lower instance rate for Character seemed to have the most potential to help inform which command prompt terms may be the most successful.

Table 2 lists the instances of common command prompt modifiers for Character. “Cgsociety” and “unreal engine 4k” outperformed “trending on artstaton.” Descriptive terms like “infrastructure,” “equipment,” “experimentation” and “research” preformed better than “architectural,” “experimental,” “university,” and “scientific.”

Terms that started with “rows” (as in “rows of tables” or “rows of shelves”) and “autonomous” outperformed “makerspace” and “laboratory” to generate the characteristics of makerspaces and laboratories.

Table 2 Table Character Command Prompts with “High” value

Command Prompt Terms	Total Count	Total "High" Pass	"High" Percentage
laboratory	161	90	55.90%
trending on artstation	114	68	59.65%
unreal engine 4k	112	80	71.43%
cgsociety	84	63	75.00%
infrastructure	74	56	75.68%
rows	64	43	67.19%
autonomous	60	41	68.33%
makerspace	55	26	47.27%
instrumentation	53	34	64.15%
equipment	32	23	71.88%
research	25	22	88.00%
scientific	21	8	38.10%
shop	20	12	60.00%
architectural	19	9	47.37%
experimental	11	5	45.45%
experimentation	10	9	90.00%
univeristy	8	3	37.50%
warehouse	8	6	75.00%
studio	4	2	50.00%
innovative	4	2	50.00%

Table 2 accounts for command prompt terms that generated artifacts in the images. Interestingly the command prompt enhancement terms that I had originally suspected were referencing artwork scored lower than some more generic terms like “shop,” “equipment,” and “makerspace.”

A term like “shop” could encompass a wide range of image types whereas a term like “warehouse” may be less

ambiguous or contain an overall image dataset with more uniform images.

Table 2 Presence of Artifacts

Command Prompt Terms	Total Count	Total "High" Pass	"High" Percentage
laboratory	161	56	34.78%
trending on artstation	114	43	37.72%
unreal engine 4k	112	39	34.82%
cgsociety	84	32	38.10%
infrastructure	74	19	25.68%
rows	64	15	23.44%
autonomous	60	15	25.00%
makerspace	55	31	56.36%
instrumentation	53	19	35.85%
equipment	32	15	46.88%
research	25	3	12.00%
scientific	21	8	38.10%
shop	20	8	40.00%
architectural	19	6	31.58%
experimental	11	2	18.18%
experimentation	10	2	20.00%
univeristy	8	3	37.50%
warehouse	8	0	0.00%
studio	4	2	50.00%
innovative	4	2	50.00%

Discussion

The data seems to indicate that to use command prompt engineering to create a makerspace it is better to use terms that describe the makerspace rather than adding the term "makerspace." Perhaps this is testament to the diversity and variety of makerspaces in the universe or conversely, a lack of images trained for makerspaces.

Continued experimentation is necessary to help inform the design process for makerspace and laboratory design and a more consistent testing between the text and image in text-to-image conversions.

A. *Artifacts*

The classification of artifacts was made on a literal text basis, however, there were also instances where the conversion was more of an "artifact of intent" in the cases in the figure below.

Fig. 13 examples reveal some of the complexities of intent and meaning with language-to-text conversions. Looking at the images may convey ideas or thoughts about the spaces that may not be consciously apparent at first glance. Clockwise from top left, "maker*space*," "Tony Stark makerspace," "Darth Vader's laboratory," and "laboratory with spiderwebs."

One could imagine a process of reversing the diffusion modeling process on images of real makerspace to create a language to track trends and inform future makerspace design.

Fig.13 Example images with "Motif Artifacts"

B. *Character*

The threshold for the different values of low, medium and high for laboratory and makerspace like qualities is subjective and limited by observer experiences and biases.

It is recommended that "characterness" aspect of image characterization be further developed with an end goal of a trained data set for machine learning automation.

C. Command Prompt Engineering Strings

As machine learning with natural language continues to evolve to recognize meaning and context in stings there is an inherent problem with analyzing the frequency of individual words. Future analysis should consider the whole structure of the command prompt sting.

References

[1] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, R. Lowe, "Training language models to follow instructions with human feedback," *arXiv, e-prints* Cornell University, 2022.

[2] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, B. K. Ayan, S. S. Mahdavi, R. G. Lopes, T. Salimans, J. Ho, D. J. Fleet, M. Norouzi, "Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding," *arXive e-prints* Cornell University, 2022.

[3] I. Goodfellow, J Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio "Generative Adversarial Nets," *Advances in Neural Information Processing Systems* 27 (NIPS), 2014.

[4] D. P. Kingma, M. Welling "Auto-Encoding Variational Bayes," *arXiv e-prints* Cornell University, 2013.

[5] D. J. Rezende, S. Mohamed "Variational Inference with Normalizing Flows," *arXiv e-prints* Cornell University, 2015.

[6] J. Sohl-Dickstein, E. A. Weiss, N. Maheswaranathan, S. Ganguli "Deep Unsupervised Learning using Nonequilibrium Thermodynamics," *arXiv e-prints* Cornell University, 2015.

[7] Y. Song, S. Ermon "Generative Modeling by Estimating Gradients of the Data Distribution," *arXiv e-prints* Cornell University, 2019.

[8] J. Ho, A. Jain, P. Abbeel "Denoising Diffusion Probabilistic Models" *arXiv e-prints* Cornell University, 2020.