PAPER

Integrating Language-Based into Deep Learning Models for Informing Architectural Design Revision

SHERMEEN YOUSIF
Florida Atlantic University

Keywords: artificial intelligence, language-based models, generative deep learning

Artificial intelligence models are moving design exploration beyond the deterministic rule-based parametric systems by offering new possibilities and expanding the design space, which has become more flexible and adaptive to change. Yet, the fact that AI models are independently learning on their own, raises issues with designers' control over the process. More recently, models that bridge natural language processing and computer-vision such as Contrastive Language-Image Pre-Training (CLIP) have been integrated into generative deep learning models such as StyleGAN, combining the generative and classification functionalities. This way, to some degree, a certain level of designer's agency can be attained when using text prompts to modify the generative process, which was the motivation of this work. We investigate here the issue of prototyping a new design system with employing language-based models and deep learning models into an expanded design space towards informing design revision and modification. Our methodology involves experimenting with the targeted deep learning models, prototyping a new framework with language-based models are integrated into the generative process, and testing the prototype by applying the proposed system to a design case. As a result of experimentation, the generative model was modified using a set of text-prompts that describe the intended design alteration. Overall, the results show successful approaches to guiding the generative process and informing design revision, and offer insights into associated potentials and limitations, as discussed in the paper.

1.INTRODUCTION

Parametric modeling systems represent the first generation of generative systems that involve rule-based and/or performance-driven algorithms to evolve design as a product of parametric exploration, with specific parameters and constraints.[1] Such a deterministic system requires designers to input parameters and constraints to generate a viable search space where design options are pre-programmed.

More recently, the integration of artificial intelligence (AI) methods into generative systems has offered new possibilities and defined new design systems. AI techniques, such as deep learning, are referred to as "Learning Systems" since they learn directly from raw data and provide "unexpected" answers.[2] The capabilities of AI to generate more artistic and creative outputs can be attributed to its non-linear and unknown synthesis and lack of rule determination in the learning process.[3] As such, a new second generation of generative systems is emerging, benefiting from the broad use of AI, and signaling a change in design towards an infinite exploration of the design space, which occasionally becomes a "hyper-dimensional" space, such as the latent space of StyleGAN models.[4] AI models may now define their own parameters independently based on information in their input datasets.[5] In particular, Generative Adversarial Networks (GANs), and their characteristics, can expand the lateral thinking restrictions in existing generative systems, and broaden the possibilities for investigating the complexity of the design issue within a wider, more representative search space. When examining a portion of the space, the designers' decision-making may influence how it contracts or expands.[6]

The complexity of AI methods with potential mis-alignment between design intentions and design outcomes allows a new area of exploration for unexpected design to emerge. With their potential to guide new design processes, deep learning strategies raise multiple issues of consideration. One issue is the question of what constitutes the "optimum" human-machine interaction mode that leads to innovative or "successful" design processes in order to generate new design solutions (compared to predetermined solutions in parametric systems). Embedded within the human-machine interaction mode is the issue of agency. This new collaborative model with the machine requires an active role for designers in design generation and evaluation.[3] This human+machine approach leads to a redefining of designers' role since we now share agency and collaborate with the machine. As such, this collaborative mode requires research on when and where the designer should intervene in the process, and when it is ideal to let the machine train and generate outcomes.

The recent introduction of neural language models with text-to-image synthesis such as CLIP and DALL-E, made by OpenAI in January 2021, opened a different research are on AI models and architecture. For example, the Contrastive Language-Image Pre-Training (CLIP) model can match text prompts with corresponding visual representations. The advantage of those language-based models is that their direct learning from text can be employed as a source of control or supervision.[7] Also, such pre-trained models can facilitate employing computationally expensive machine-vision processing in design systems. Yet, in architectural design, an exact relationship between text and visual feature representation becomes difficult and raises issues of "ambiguity" in text-image associations. An image of architecture cannot be simply described in simple text. More importantly, it is unreasonable to ask the language model to "design" using text descriptions.

This issue of ambiguity in text-image translation can be solved by establishing an intermediary medium for delivering user expectations.[8] Language-based models can be applied to tackle specific design aspects and not the holistic complexity of the design activity. Generative deep learning models, such as StyleGAN, can be used for different design tasks, and a language model (i.e., CLIP) for certain tasks within the design workflow. This hybrid process can overcome CLIP's shortcomings by defining a generative system with specific architectural datasets for CLIP to be useful.

To address the issue of agency and facilitate designers' control of the generated outcome within deep learning models, we propose here a new design system that involves employing language-based and deep learning models. The language-based model will allow encoding designers' intents and intervening in the generation of the design space, the "latent" space, and enable a level of designer control. Our methodology involves experimenting with deep learning models, prototyping a new system framework with language-based and DL models integrated, and testing the prototype by applying the proposed system to a design case.

In the test-case application, in a preliminary task, (Experiment 1), only the language model was used with two types of inputs of text descriptions and reference images (floor plans of office buildings) to investigate the model capabilities. In Experiment 2, the main test-case application of the proposed method, 2200 images of floor plans of office buildings were used for a generative deep learning model to generate new design options (through a StyleGAN training process). Next, a language model (CLIP) was coupled with the trained GAN model to modify the latent space in two test cases: (1) CLIP+StyleGAN, and (2) StyleGAN-NADA. The experimental test-case attempts to address the questions of how to alter current architectural layouts for temporary post-pandemic scenarios and how to prepare buildings for possible future pandemics. The issue is complex and manifold, but can be simplified to two questions: (1) rethinking design processes and (2) rethinking the architectural space. The research was targeted at developing a new design system that modifies and changes according to designers' interference in the generative process to enable architectural design modification, while the test-case was pursued to review and revise typical architectural floor plans of office buildings.

2.BACKGROUND

Despite its recent introduction, research on text-to-image synthesis has been exponentially increasing, signaling a significant area of inquiry. In connecting language-based models to generative deep learning, the work of Reed et. al (2016) introduced an early approach that uses a GAN model conditioned upon text embeddings.[9,10] The work proposed a revolutionary model for GAN formulation that learns discriminative text feature representations, bridging the gap between breakthroughs in text and image modeling and efficiently transferring visual concepts from letters to pixels.[10]

In 2017, StackGAN was developed to synthesize high-quality images conditioned to text descriptions.[11] In decomposing the difficult problem into more manageable sub-problems, the work offered a model with a two-stage process and different resolutions, a unique approach that supports smoothness in the latent conditioning manifold by improving the diversity of the synthesized images and stabilizing the training of the conditional-GAN model. [11] In further development, the authors presented StackGAN++ an advanced multi-stage generative adversarial network architecture, with both conditional and unconditional generative tasks.[12] The model consists of multiple generators and discriminators in a hierarchal structure.

In another approach, AttnGAN5 can synthesize fine-grained information at distinct subregions of an image by focusing on key phrases in the natural language description. In training the generator, a deep attentional multimodal similarity model is presented to compute a fine-grained image-text matching loss.[13] Reed et al. (2016)[14], Li et al. (2019)[15], and Koh et al. (2021)[16] are three other studies that focused on incorporating additional sources of supervision during training to improve image quality. Importantly, the work of (Nguyen et al. 2017)[17] introduces a conditional iterative generation of images in latent space, while (Cho et al. 2020)[18] advances language models research and proposes X-LXMERT, an extended model that is capable of generating semantically meaningful images from pieces of text.

In 2021, important language models such as DALL-E and CLIP were introduced by OpenAI. DALL-E is a neural network that generates graphics from text descriptions for a wide variety of concepts expressible in natural language. The model was trained to generate images from text, using a dataset of text–image pairs.[9] As its authors assert, DALL-E has a wide range of capabilities, including the ability to create representations, or to combine unrelated concepts in believable ways, to display text, and to make alterations to existing pictures. CLIP is another advanced language model that was used in his study and explained in detail in Sub-section 2.3.

In architecture design, research that addresses language-based models is fairly new. The contribution of Theodore Galanos (2021)[19] to training DALL-E with architectural datasets has become important. Galanos trained DALL-E on 150k floor plan layouts, and his model shows a potentially useful approach to generating designs from text prompts. In one instance, he tested the model to generate variant designs based on the text "a house with three bedrooms and two bathrooms". While the attempt to train such models with architectural datasets is important, the results also show the limitations of DALL-E when used as a generative mechanism. Some results show strange relationships between spaces, which are unsuccessful architecturally. This signals the significance of careful consideration when using language models, in regard to what design tasks and at what design phase such models can be best applied, and how they can be successfully incorporated into design processes.

In another architectural design approach by (Mistry and Escobar 2021)[20], CLIP was integrated into a VQGAN model for text-to-image generation. VQGAN is a Vector Quantized Generative Adversarial Network that combines convolutional neural networks –often used for image recognition, with transformers –traditionally used for language, and it performs well for high-resolution pictures.[21]

Based on extensive review and experimenting with existing methods, we can argue that at this stage, language models are not suitable yet to generate designs, because design is a complex activity with multiple interrelated layers and systems. Adopting systems thinking and Christopher Alexander's approach of "systems generating systems" (1968)[22], in our larger research project, we approach AI applications in a framework of multiple connected models, each designed specifically to tackle an architectural system or a design task.

From our experimentation, we identified possible applications of language-based models not in the sense of generating design, but rather in review and revision, informing one design aspect. In our method, two potential applications were addressed: (1) to query in the AI-generated space, and (2) to modify the generative model to reflect design intentions, encoded as text prompts. To achieve that, we targeted prototyping a new design system by employing language-based models and deep learning models.

Prior to introducing our method, the most relevant methods are first explained and introduced in the following subsections. The order of the methods represents the order of their integration in our proposed prototype, involving generative adversarial networks, in particular StyleGAN, CLIP, and two hybrid models with CLIP embedded, which are StyleGAN+CLIP, and StyleGAN-NADA.

2.1. Generative Adversarial Networks (GANs)

Developed by Ian Goodfellow and his colleagues (2014), generative adversarial networks are a class of deep learning in which two neural networks compete against each other. The model learns to create new data (output) with the same statistics as the training set (input) given only a training dataset. Simply, GANs are game theoretic scenarios in which the generator network competes with an opponent. In this game, the generator network learns to create synthetic (fake) images in a representation learning process, while the discriminator network, its main rival, makes an effort to differentiate between real samples (taken from the training data) and fake ones (those taken from the generator's output). Throughout the training (learning), each network (generator and discriminator) attempts to maximize its own payoff. As a result, the discriminator tries to learn to classify samples accurately as real or false, while the generator tries to persuade the classifier that its samples are genuine. At convergence, the generator's samples become indistinguishable from the real dataset, and the discriminator outputs 1/2 (50% fake, and 50% real) everywhere.[23,24]

2.2. StyleGAN

StyleGAN is an unsupervised generative adversarial network model capable of producing highly realistic images with high resolution. Its architecture "leads to an automatically learned, unsupervised separation of high-level attributes and stochastic variation in the generated images, and it enables intuitive, scale-specific control of the synthesis".[25] Some neural units in the intermediary layers of the trained deep generative model are specialized to synthesize certain visual features. From the learnt representations of deep generative models —using layer-wise stochasticity, a highly-structured semantic feature hierarchy can arise spontaneously without any supervision, such as in the case of StyleGAN. Layer-wise stochasticity refers to the fact that the input latent codes for data synthesis are fed into all network layers of a deep neural net rather than just the first, allowing for improved interpolation and disentanglement of acquired semantic characteristics.[25]

StyleGAN can produce a wide range of high-quality picture data. It uses an auxiliary multilayer perceptron network to translate a randomly sampled latent code from an initial latent space Z to an intermediate high-dimensional space W before feeding it into the generator. The disentanglement feature of W is stronger than that of the initial space, a property that allows for scale-specific control of data synthesis, as well as improved interpolation and disentanglement of the acquired latent semantic characteristics. [26]

In further development, StyleGAN-2 with adaptive discriminator augmentation (ADA) was introduced in 2020. The new edition involves an adaptive discriminator augmentation approach that dramatically improves training stability in data-constrained environments. The model was trained on numerous datasets and show that satisfactory results can be obtained with only a few thousand training images. It can be used for both training from scratch and fine-tuning an existing GAN on a different dataset. This characteristic can open up new application domains for GANs.[27]

2.3. Contrastive Language-Image Pre-Training (CLIP) and VQGAN+CLIP

Published by OpenAI in January 2021, CLIP (Contrastive Language-Image Pre-Training) is a neural network that has been trained on a wide range of (image, text) pairings. CLIP models learn to accomplish a broad array of tasks during pre-training in order to optimize their training aim. Zero-shot learning (ZSL) is a machine learning issue in which a learner views samples from classes that were not viewed during training and must guess which class they belong to.[28] This learning is then used to enable zero-shot transfer to a variety of existing datasets via natural language prompts. At a large scale, this technique can compete with task-specific supervised models, yet there is still a lot of space for improvement.[7]

The model has been trained on a dataset of 400 million (image, text) pairings acquired from the internet, and shows that the basic pre-training job of predicting which caption goes with which picture provides an effective and scalable technique to learn state-of-the-art image representations from scratch. After pre-training, natural language is employed to refer to (or describe) learnt visual ideas, allowing for zero-shot model transfer to downstream tasks.[7] According to the authors of CLIP, employing natural language supervision for image representation learning is still new, and early research has struggled with the complexities of natural language when utilizing topic model and n-gram representations. However, advances in deep contextual representation learning indicate that we now have the skills to efficiently use this plentiful source of supervision.[29]

VQGAN+CLIP is a neural network architecture that builds on OpenAI's breakthrough CLIP architecture, which was released in January 2021. It is a text-to-image model that creates graphics of varying sizes in response to a series of text descriptions (and some other parameters). We used VQGAN+CLIP for our early experimentation to examine potentials and limitations of direct application of a language model.

2.4. StyleGAN+CLIP (StyleCLIP)

The concept of this approach is to combine the above-mentioned models, StyleGAN and CLIP. The objective is to use the power of CLIP to provide a text-based interface for StyleGAN image modification (Patashnik et al. 2021). To explain it simply, the model can build an image by starting with random values for the latent vectors from StyleGAN. The result, together with an arbitrary prompt, will be provided to CLIP. Next, CLIP assigns a score to this image based on how effectively it represents the content of the text prompt. This is then used to manipulate the latent vector, which will produce another picture, and this cycle will be repeated until the resulting image adequately resembles the text prompt.[30]

2.5.StyleGAN-NADA

In the attempt to develop a generative model to be taught to generate images from a certain domain based solely on a text prompt without ever seeing an image, StyleGAN-NADA, a Non-Adversarial Domain Adaptation, was formulated. The model is based on CLIP-guided domain adaptation of image generators. The approach employs two paired generators, starting both with a pre-trained model and keeping one generator constant while training the other generator occurs through requiring that the direction of their produced image in CLIP space correspond with some predetermined textual direction (Rinon Gal 2021). The model represents a text-driven strategy for shifting a generative model to new domains without collecting even a single image from those domains, using the semantic capability of large scale Contrastive-Language-Image-Pre-training (CLIP) models. With natural language prompts and a few minutes of training, the model can modify a generator across a wide range of domains with varying styles and forms.[31]

For our method development, an independent StyleGAN-2 model was first trained, and in a second phase, a StyleGAN-NADA model was employed with CLIP embedded.

3.RESEARCH METHODS

The research methodology adopts Peffers et al.'s (2007)[32] approach of "Design Science Research Methodology". In this protocol, system development follows six processes of: problem identification, objective definition, system design and development, demonstration, evaluation, and communication. Since the work is still in-progress, the methodology represents a protocol of extensive literature study for problem identification, experimentation with existing methods, prototyping the proposed system, and testing and applying the system to a design case study.

In prototyping the proposed system, a framework has been developed, following John Gero's identification of design prototypes, defined as systems that include sufficient expressive capability to capture the characteristics of the ideas that support the process (1990).[33] Gero's design prototype notion allows separating architectural design knowledge from the computational processes that operate upon that knowledge (Gero 1990). The prototype involved protocols of: (1) dataset augmentation and curation, (2) training a generative deep learning model, and (3) training a hybrid model (generative and language-based). The language-based model was integrated within as a control mechanism for guiding the shift and intended change in the generative process, inserted into two parallel models (Figure 1).

3.1.Experiment 1 (VQGAN+CLIP)

This experiment was carried out as a preliminary task to investigate the potential of language models when used on their own (without the integration of a generative deep learning model). VQGAN+CLIP was employed with the integration of reference images and floor plan designs to modify (Figure 2). The following text prompts were selected as input for running the model, with hierarchy from less to more specific and alternating between open and partitioned spaces:

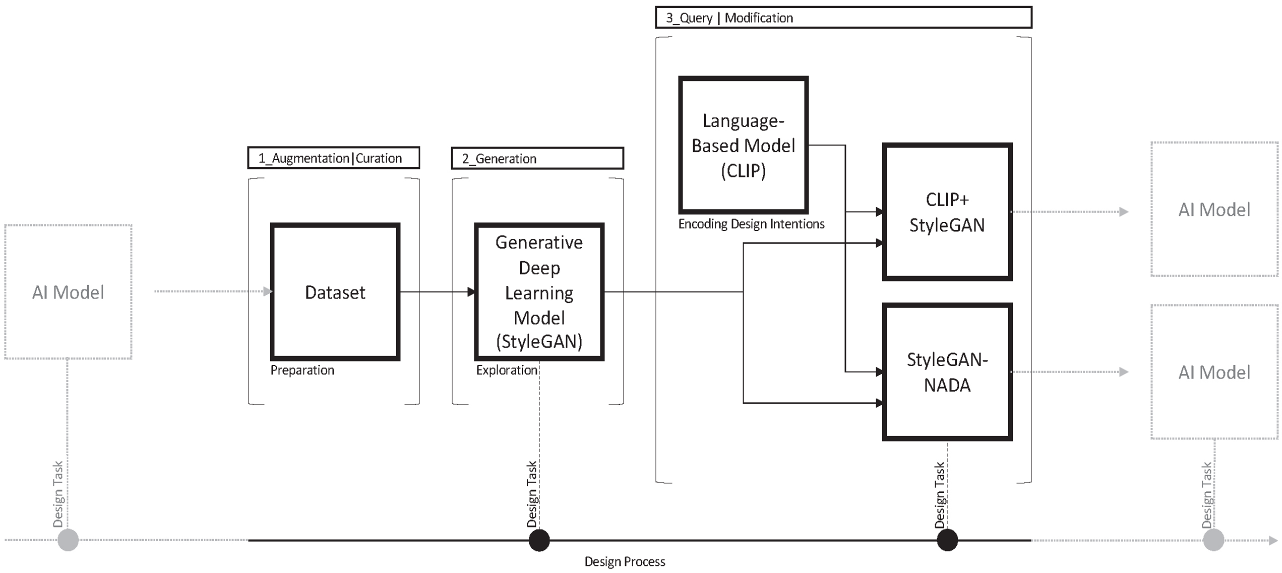

Figure 1. The proposed design system with the three main processes within the overall design process. Each model is linked to a design task, and the method proposed is part of a larger workflow with multiple connected AI models. Image credit: the author.

1. *A floor plan of an office building designed*
2. *A floor plan of an office building designed with partitions*
3. *A floor plan of an office building with flexible workspaces designed to integrate landscape into the building*
4. *A floor plan of an office building with partition workspaces designed to integrate landscape into the building*

Importantly, in addition to the text prompt, adding a visual reference of a floor plan design to modify was pursued. Each of the text prompts became the input for two model runs; in one instance, the selected floor plan design became an initial image, and in the other case, it was used as a target image. This has led to eight model runs, as depicted in Figure 2. For the first two text prompts, the Dominion office building by Zaha Hadid Office was selected as the reference floor plan design, while for the other two texts, the NL*A office building, Offices With Terraces, in France, was chosen.

The results show a more coherent floor plan layout when the reference image was used as an initial image, especially for Text Prompt 1 and 2. It is also noticed that including partitions in the text prompt did not necessarily yield different results, while a cubicle-like layout and small enclosed spaces can be detected. Overall, the reference floor plan design was dramatically changed, in most cases. While the results can be suggestive of certain modification scenarios, they can also be seen as too far from the original design (in the reference image), which may not be preferred when systematic change is targeted.

3.2. Experiment 2 (StyleGAN+CLIP) and (CLIP+StyleGAN-NADA)

The proposed prototype was tested in this case study, with a focus on the issue of rethinking the architectural space in post-pandemic scenarios. Problems with existing spaces can be attributed to the modernist open floor plan layouts that exist in multiple building typologies such as office buildings.[34] Speculative design solutions include a shift to new floor plan design alternatives and flexibility in defining and enclosing spaces. Tackling this issue, the aim was to modify office spaces in typical office buildings' floor plans to enhance occupancy behavior in terms of physical distancing and reduced disease transmission. The objective was to review current floor plans and generate modified, resilient futuristic design propositions.

For this test-case experiment, PyTorch® and Tensorflow® deep learning packages were used within the PyCharm environment for the training of the neural networks. The training was performed using a virtual machine in Google Colab.

3.2.1.Dataset Augmentation and Curation

One important task that precedes training a deep learning model is curating the input dataset. It is self-evident that the deep learning model's performance is contingent on the quality of the data from which it can learn and extract knowledge. The system's workflow yielded a specific dataset acquisition method. Data on floor plans of office buildings was scraped from the online architectural databases of Arch-daily and Dezeen, in addition to Pinterest. After collecting 500 images of floor plans, they were augmented using data management tools in Python programs written specifically to articulate the dataset. For post-augmentation, 2200 floor plans were retrieved and used as the input dataset for training the StyleGAN model (Figure 3).

3.2.2. Training the Generative Model (StyleGAN-2)

The main parameters for the StyleGAN-2 training experiment were: (1) dataset: 2200 images of resolution of 1024x1024 pixels, (2) training: using a pre-trained model of NVIDIA lab (3) duration: training iterations of 10,000. The model performed breeding of the dataset and generated a latent space of synthetic (fake) floor plan designs (Figure 4).

3.2.3. Training StyleGAN+CLIP

Prior to this step, a CLIP-Ascending model was used with the input of selected images from the StyleGAN dataset to extract text-prompts that the CLIP model outputs when it sees the input images. This task becomes important to understand how CLIP associates specific text prompts with the input floor plans, and to use the resulting text prompts as hints for further experimentation with the hybrid models in the next steps.

The StyleCLIP model was used primarily to query the resulting latent space of StyleGAN and find seeds (images) that satisfy the input text prompt, while modifying that seed with the intended output text prompt. The objective was to search within the latent space using a text prompt of (open floor plan layout) while using an output text of (partition floor plan layout) as an output text to modify the floor plans. Figure 5 (left) represents the outcome post using the output test.

3.2.4. Training StyleGAN-NADA

Here, the experiment was targeted to guide the generator of the StyleGAN-NADA to adapt and modify based on the input text prompts in order to generate new floor plans with encoded design intentions. The design intention was to coerce changes in the floor plans to include more partitions and enclosures. The input class text was also (open floor plan layout) and the text prompt used was (partition floor plan layout). The results were 1000 images (corresponding to the number of iterations) that represent the transition from one image (a fake floor plan) from the SyleGAN seeds to a more partitioned layout with additional rooms and enclosures (Figure 5). The order of iterations reflects the transformation process from the original fake floor plan to CLIP-informed design iterations.

Figure 2. Results of using VQGAN+CLIP with 4 text prompts, each used as an input along with reference images as initial and target images in Experiment 1. Image credit: the author.

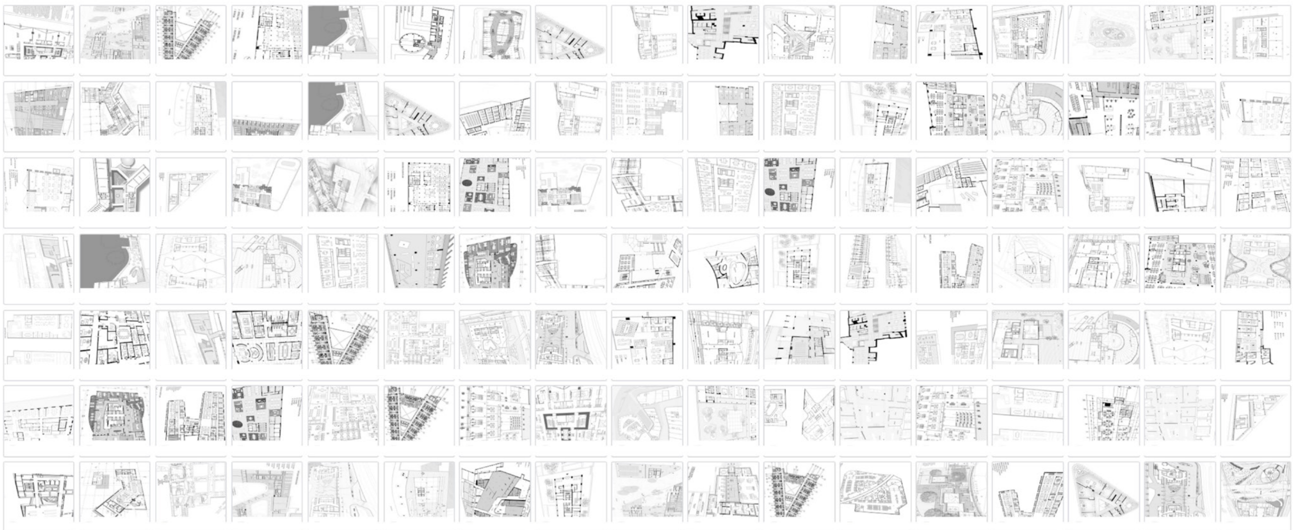

Figure 3. A sample of the input dataset (2200 floor plans of office buildings) for the StyleGAN training depicts only 119 out of the 2200 floor plans used for the model training. Image credit: the author.

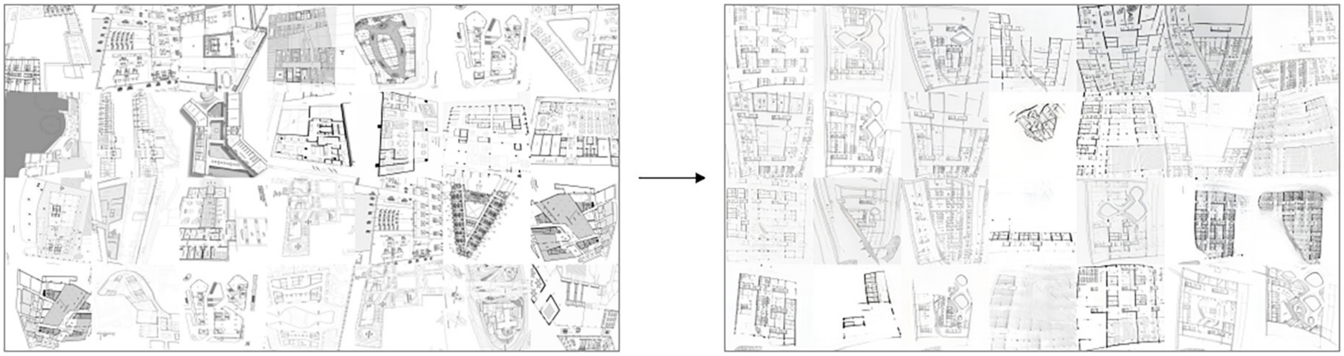

Figure 4. Left: real images representing the actual dataset prior to the StyleGAN training; right: fake images representing StyleGAN-generated floor plans when training stopped at Snapshot 60. Image credit: the author.

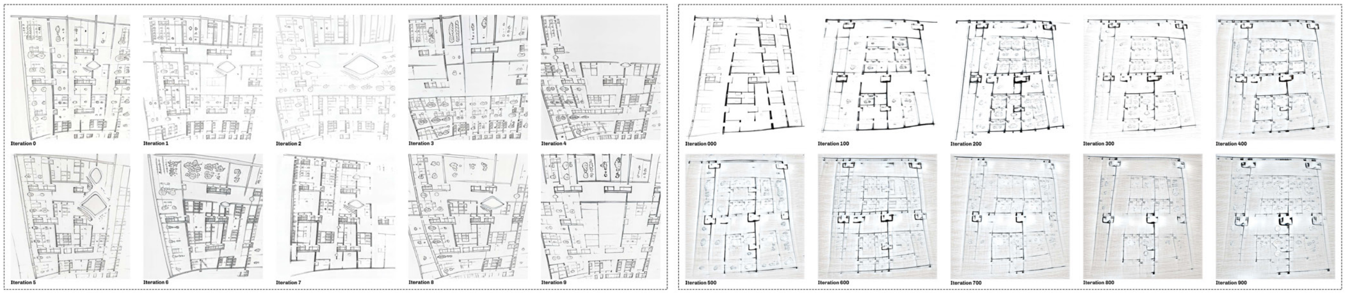

Figure 5.Left: 10 images of floor plan generated from StyleGAN+CLIP, using the text prompt: "open floor plan layout "as an input to query the latent space. Right: a selected sample of 10 images represents the transition from a floor plan design retrieved from the StyleGAN model (Iteration_000) to a series of modified floor plans (the other 9 images) informed by the text prompt (partition floor plan layout) provided in the StyleGAN-NADA experiment. Image credit:the author.

4.DISCUSSION OF RESULTS

As presented, the results show potential applications of pre-trained language-based models such as CLIP to enact designers' control and incorporate design intentions to shift and manipulate the generative process within unsupervised GAN models such as StyleGAN. In the StyleGAN+CLIP, results were less coherent since the model was performing a "query" within the latent space. On the other hand, StyleGAN-NADA showed more progressive transformation in the generative process, and the generator adapted to the desired change prompted by the input text. To evaluate the results in terms of encoded design intentions, StyleGAN-NADA proved more powerful and yielded a more promising outcome.

However, the results also show the limitations of CLIP and overall language-based models, in particular when using images of architectural designs. One issue is the randomness and ambiguity associated with the process. The text-image relationship, in particular, and the interpretation of text prompts to images become uncontrolled and can yield successful results at times yet failed experiments at other times. Another issue is the limited classes (categories) of input and targeted outputs of text prompts (i.e., dogs, cats, cars…), that do not include architecture-related classes.

It is also important to note that since CLIP performs classic classification training, it is merely concerned with the specified labels. This means, if it is effective in identifying dogs, CLIP doesn't care if it is a photo, drawing, or description of a certain breed. This aspect is problematic in architectural representation domains where drawings and renderings are two different representations. Another significant observation from CLIP's performance is its struggle with identifying fine-grained details, as it performs better at categorizing the overall composition or structure of the image than its detailed features. For architectural design, this means that CLIP can be best applied when the overall structural composition of the design is targeted, and not its details.

5.CONCLUSIONS AND FUTURE WORK

Presented in this paper is work-in-progress research that investigates incorporating language-based models into generative deep learning towards a new AI-guided design process. The proposed method was tested with application to a study of architectural design review and revision. This is a work-in-progress effort, a study within a larger research project where we target employing multiple deep learning models to address and tackle multiple design aspects, and at different phases.[35] The importance of this effort is the development of a new method that enacts designers' agency and control within a process of AI-based design generation and modification.

To overcome the limitations of the CLIP model regarding its ambiguity and inability to recognize architectural design representations, there is a need for training CLIP on architectural datasets. In terms of the test-case application, the next phase of this work involves generating new datasets retrieved from the agent-based simulation (ABM) process where multiple cases of human occupancy patterns overlapped with architectural floor plans are tested. Such an identification of human occupancy patterns will assist in informing the revision of floor plan designs to achieve intended performance. Future work also involves additional test-case applications of different design workflows where multiple AI models are employed and connected, including text-to-image synthesis, to address multiple design tasks.

ENDNOTES

1. Woodbury, Robert, Robert Aish, and Axel Kilian. 2007. "Some patterns for parametric modeling." Expanding Bodies: Art • Cities• Environment [Proceedings of the 27th Annual Conference of the Association for Computer Aided Design in Architecture, Halifax (Nova Scotia).
2. Hassabis, Demis. 2018. "Creativity and AI" In "The Rothschild Foundation Lecture." The Royal Academy of Arts.
3. Example of a conference proceedings paper in a book: [Author Name(s), first Wit, Andrew John, Lauren Vasey, Vera Parlac, Mara Marcu, Wassim Jabi, David Gerber, Mahesh Daas, and Mark Clayton. 2018. "Artificial intelligence and robotics in architecture: Autonomy, agency, and indeterminacy." International Journal of Architectural Computing 16 (4): 245-247. https://doi.org/10.1177/1478077118807266.
4. Gui, Jie, Zhenan Sun, Yonggang Wen, Dacheng Tao, and Jieping Ye. 2020. "A review on generative adversarial networks: Algorithms, theory, and applications." arXiv preprint arXiv:2001.06937.
5. Chaillou, Stanislas. 2019. "The advent of architectural AI." Retrieved from towards data science https://towardsdatascience. com/the-advent-of-architectural-ai-706046960140.
6. Bolojan, Daniel. 2021. "The Hitchhiker's Guide to Artificial Intelligence: AI and Architectural Design" In "https://www.digitalfutures.world/." https://www.youtube.com/digitalfuturesworld/live.
7. Radford, Alec, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, and Jack Clark. 2021. "Learning transferable visual models from natural language supervision." arXiv preprint arXiv:2103.00020.
8. Yadav, Apurwa, Aarshil Patel, and Manan Shah. 2021. "A comprehensive review on resolving ambiguities in natural language processing." AI Open 2: 85-92. https://doi.org/https://doi.org/10.1016/j.aiopen.2021.05.001
9. Ramesh, Aditya, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. "Zero-shot text-to-image generation." arXiv preprint arXiv:2102.12092.
10. Reed, Scott, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. 2016. "Generative adversarial text to image synthesis." International Conference on Machine Learning.
11. Zhang, Han, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. 2017. "Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks." Proceedings of the IEEE international conference on computer vision.
12. ---. 2018. "Stackgan++: Realistic image synthesis with stacked generative adversarial networks." IEEE transactions on pattern analysis and machine intelligence 41 (8): 1947-1962.
13. Xu, Tao, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. 2018. "Attngan: Fine-grained text to image generation with attentional generative adversarial networks." Proceedings of the IEEE conference on computer vision and pattern recognition.
14. Reed, Scott E, Zeynep Akata, Santosh Mohan, Samuel Tenka, Bernt Schiele, and Honglak Lee. 2016. "Learning what and where to draw." Advances in neural information processing systems 29: 217-225.
15. Li, Wenbo, Pengchuan Zhang, Lei Zhang, Qiuyuan Huang, Xiaodong He, Siwei Lyu, and Jianfeng Gao. 2019. "Object-driven text-to-image synthesis via adversarial training." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.
16. Koh, Jing Yu, Jason Baldridge, Honglak Lee, and Yinfei Yang. 2021. "Text-to-image generation grounded by fine-grained user attention." Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision.
17. Nguyen, Anh, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. 2017. "Plug & play generative networks: Conditional iterative generation of images in latent space." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

18. Cho, Jaemin, Jiasen Lu, Dustin Schwenk, Hannaneh Hajishirzi, and Aniruddha Kembhavi. 2020. “X-lxmert: Paint, caption and answer questions with multi-modal transformers.” arXiv preprint arXiv:2009.11278.
19. Galanos, T. DALL-E in Pytorch, github.
20. Mistry, Mayur, and Daniel Escobar. 2021. “Intro to AI for Architectural Design Explorations.” https://www.digitalfutures.world/workshops/26.html.
21. Esser, Patrick, Robin Rombach, and Bjorn Ommer. 2021. “Taming transformers for high-resolution image synthesis.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.
22. Alexander, Christopher. 1968. “Systems generating systems.” Architectural Design 38 (December): 605-610.
23. Goodfellow, Ian, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. 2016. Deep learning. Vol. 1. Vol. 2: MIT press Cambridge.
24. Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. “Generative adversarial nets.” Advances in neural information processing systems.
25. Karras, Tero, Samuli Laine, and Timo Aila. 2019. “A style-based generator architecture for generative adversarial networks.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.
26. Chen, Jielin, and Rudi Stouffs. 2021. “From Exploration to Interpretation-Adopting Deep Representation Learning Models to Latent Space Interpretation of Architectural Design Alternatives.”
27. Karras, Tero, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. 2020. “Training generative adversarial networks with limited data.” arXiv preprint arXiv:2006.06676.
28. Xian, Yongqin, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. 2018. “Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly.” IEEE transactions on pattern analysis and machine intelligence 41 (9): 2251-2265.
29. McCann, Bryan, James Bradbury, Caiming Xiong, and Richard Socher. 2017. “Learned in translation: Contextualized word vectors.” arXiv preprint arXiv:1708.00107.
30. Patashnik, Or, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. 2021. “Styleclip: Text-driven manipulation of stylegan imagery.” Proceedings of the IEEE/CVF International Conference on Computer Vision.
31. Rinon Gal, Or Patashnik, Haggai Maron, Gal Chechik, Daniel Cohen-Or. 2021. “StyleGAN-NADA: CLIP-Guided Domain Adaptation of Image Generators.” Computer Vision and Pattern Recognition. https://arxiv.org/abs/2108.00946.
32. Peffers, Ken, Tuure Tuunanen, Marcus A. Rothenberger, and Samir Chatterjee. 2007. “A Design Science Research Methodology for Information Systems Research.” Journal of Management Information Systems 24 (3): 45-77. https://doi.org/10.2753/MIS0742-1222240302. https://doi.org/10.2753/MIS0742-1222240302.
33. Gero, John S. 1990. “Design prototypes: a knowledge representation schema for design.” AI magazine 11 (4): 26-26.
34. Gibbens, Sarah. 2020. “Goodbye to open office spaces? How experts are rethinking the workplace.” National Geographic.
35. Bolojan, Daniel, Shermeen Yousif, and Emmanouil Vermisso. 2021. “Workshop:Latent Morphologies: Disentangling Design Spaces. ACADIA 2021, REALIGNMENTS: Toward Critical Computation”.