ANALYSING THE USAGE FO AI ART TOOLS FOR ARCHITECTURE

Joern Ploennigs[1], Markus Berger[1]
[1]AI for Sustainable Construction, University of Rostock, Germany

Abstract

The recent advancements of commercial text-to-image and image-to-image generation platforms have created a surge in interest in many creative disciplines, including architecture. In this paper we analyze 58 million publicly available queries from the platform Midjourney to find architectural use cases. We utilize various statistical and NLP methods to extract quantitatively how users utilize and collaborate on the platform. For some of the most popular queries we discuss qualitatively the generated results in their applicability to architecture. Our results show that while there are still limitations in image generation models, they are already widely adopted for architectural use cases.

Introduction and State of the Art

Recently, the output quality of generative machine learning models has improved to a degree that new avenues of use have opened up. This increase in quality has led to the appearance of commercial generation platforms, in which users can create arbitrary text and image prompts in order to quickly generate large amounts of images. These images are sometimes used as a finished creative results and sometimes as a basis for further manual editing or design ideation.

Various traditional visualization methods from manual sketches to image editors and 3D renderings are used in architectural design on a daily basis. It did not take long for architects to take an interest in generative methods, as reflected by a special edition of the AEC Magazine (2022). The new technology is discussed widely in public, from its specific use cases to the ethics of how it has been developed and what changes it will inflict. In this paper, we want to use the open nature of the Midjourney platform to analyze current use cases and capabilities for architecture in a quantitative way. We analyze 58 million queries through several methods, including NLP methods like word2vec. We consider the relevant parts of the technology behind these models and will look into how they could benefit working architects now and in the future.

The current technological basis for image generation models are so-called *diffusion* methods. First introduced in Sohl-Dickstein et al. (2015), *forward diffusion* destroys the structured information in an image step-by-step, while *reverse diffusion* tries to regenerate the lost information. However, because the original image information has been destroyed, the reverse diffusion is working at least partially off of random noise. The end result of this back-and-forth will therefore be a completely new image, which depending on the amount of forward diffusion will only bear slight resemblance to the original in style and composition.

This basic idea of diffusion was then used to create neural network architectures that are capable of using reverse diffusion (denoising) to create high-quality image outputs from noisy inputs (Ho et al., 2020). Configuring these models to create the desired outputs used to be a process that required expert knowledge. More recent architecture variants like OpenAI's GLIDE model (Nichol et al., 2022) contain an encoder, which can take an arbitrary text prompt by a user and create a valid text encoding that can be fed into the connected diffusion model. This architecture also includes a second model, which upsamples the result of the diffusion model.

Most current commercial models use the CLIP (Contrastive Language Image Pre-training) architecture presented in Radford et al. (2021) and Ramesh et al. (2021). CLIP is responsible for training the encoder and determining how the text encodings are linked to image parts in the diffusion model. From this point, every platform contains slight differences in model and encoder architecture. Midjourney does not specify their exact architecture, but likely operates on similar principles to DALL·E 2. The specific encoder used there is called unClip, which includes an image encoder and encodes both text and image inputs into a joint representation space from which the diffusion model can create an image (Ramesh et al., 2022).

This architecture allows users to combine both image-to-image (img2img) as well as text-to-image (txt2img) prompts. This is ideal for design ideation, as we can combine textual direction with reference images. These reference images could include desired composition, colors, content and more.

Midjourneys public interface is based on this exact concept. Users write text-to-image and image-to-image prompts and post them into private or public channels on the communication app Discord. These channels are read by a discord bot, which inserts the prompt into the model and responds with the resulting image.

The main difference to other model providers is Midjourneys multi-stage interactive upscaling process. Several low-resolution image variants are generated that can be selectively up-scaled and refinement in a user-centered process. This refinement process includes different kinds of upscaling models as well as a remaster-model which drastically changes the nature of an image according to parameters pre-configured by Midjourney.

It is not known which exact data Midjourney's models are trained on, but, it are likely large image databases in the internet with textual descriptions. For example, Stable Diffusion used the LAION-5B dataset (Rombach et al. (2022)), which was created from large amounts of images and accompanying text. Depending on the training dataset, each AI model learns its own style. In Midjourneys case, the model appears fine tuned for artful composition and vibrant color palettes and often creates evocative images even from a very plain (or nonsensical) text prompt.

While differences are apparent to anyone who uses these models often enough, the quality is difficult to assess. One attempt at quantitative assessment based on human faces is made in Borji (2022). A more qualitative approach with a focus on urban planning can be found in Seneviratne et al. (2022), where thousands of images were automatically generated by the DALL·E network, based on variations of words constrained by a systematic grammar.

In this paper we analyze the current state of generative AI art models in architecture in the following ways:

- A discussion of the technology and interface of Midjourney as an example AI art platform
- An quantitative NLP analysis of how Midjourney is used for architecture today
- An qualitative analysis of the most popular prompts

Midjourney overview

Because Midjourney allows users to prompt their bot in a collection of public channels on their discord server, we were able to monitor these channels and extract the queries with a web crawler. This is a massive data set—Midjourney's image generation is fast enough that users often iterate queries rapidly with different prompt configurations in parallel, hoping to find a fitting result. Resulting in a new image appearing in mean every 3 seconds on popular channels. This opens complex hierarchies of changing prompts, which are hidden in dozens of successive images created by multiple users sharing the same channel.

Formulating these prompts is not that easy and requires some experience and skill in finding the right phrases by adding and removing keywords and finding their right order. The resulting phrases are usually not full sentences but convoluted keyword collections that steer the image generator in the right directions. We will analyze these keywords within our study and classify them based on their typical intent to learn what is commonly used and why.

As multiple users share public channels in Midjourney it is not uncommon that users pick up interesting queries from other users and start deriving new variants of these images. This leads to social network effect, where some queries start spreading across larger user groups. This is encouraged by Midjourney by providing topic channels where top queries belonging to an area are selected and highlighted.

Another option in Midjourney to derive new images in a certain style is by using image-to-image approach by uploading reference images to discord. They are posted as link in the prompt and are then applied as the base for the diffusion process. How they influence the result is a game of luck. In some cases the generator copies the style of the original image, in some cases the arrangement, and in some cases the object in focus.

A unique selling point of Midjourney compared to many competitors is the combination of multiple different generation and up-sampling steps into one workflow. After every iteration step, multiple choices are offered for further processing the result. Because every option takes the image into a different stylistic direction, this setup tends to create large branching trees of queries and refinements, in which one user is working on multiple branches at once. The most drastic change is usually brought by the *Remastering* model, which tries to increase coherence and realism of generated images (with often mixed success). Figure 1 shows the basic idea behind the workflow and apart from some minor changes applies to both model version 3 and 4. The general pattern is to write a prompt, generate variants until one or multiple variants appeal to the user and then to refine those variants.

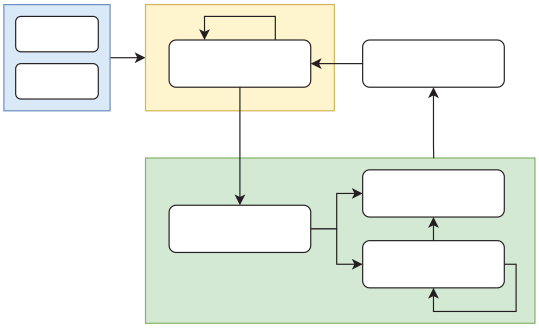


Figure 1: The image generation process in Midjourney.

Analyzing queries

Setup

To understand how users utilize Midjourney for architectural designs we collected the queries from the public channels on Midjourneys Discord server and analyzed them. We collected in total 58 million queries across ca. 30 channels over a Year from January 30th 2022 to 2023. Figure 2 shows in blue the message frequency per day of the queries we analyzed. The first messages are dated to January 30th 2022 when Midjourney was still in closed alpha. It became available as a closed beta on March 21th 2022 with 7222 queries that we observed. It moved to a friend invite schema in April and scaled up their server availability in May. From here on the usage starts to increase. Its hype started in the summer months when DALL·E released its version 2, which resulted in huge press coverage. In contrast to DALL·E's restricted access in the beginning, Midjourney went open beta on July 13th

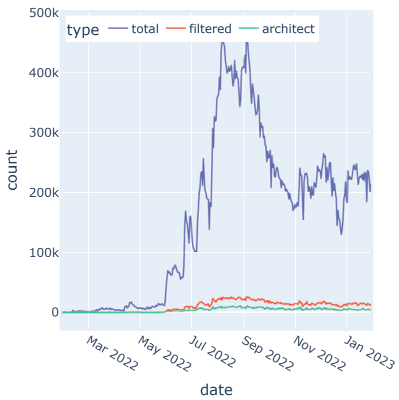


Figure 2: Frequency of queries on Midjourney per day.

with 256k observable queries on the first day. The hype peaked first on August 7th with 463.000 queries in a single day and again on September 4th with 474.000 queries. It then slowed down to about 200k queries per day. On November 6th Midjourney released the new version V4, which significantly improved image quality and resulted in another peak with 241k queries. All these query number are what we could see from the open accessible channels. As is it also possible to send the Midjourney bot private messages, the real numbers of queries is certainly higher. However, the analysis gives an idea of the amount of queries the AI is processing in minimum and how it scaled along the hype.

As we are interested solely in the queries that are related to architectural topics we filtered the collected queries. Manually identifying the real intent behind a query or an image with this number of queries is impossible. So we needed an automated approach. We do not know the background of users and also did not try to analyze user related information in respect for their privacy. As most queries are collections of keywords without syntax it is hard to build a classifier. Instead we decided to filter out queries based on specific keywords. We distinguish three types of keywords: (i) the explicit use of the term “architect”, “interior” or “exterior” design in the query; (ii) the implicit use of a keyword semantically related to these terms. The reason is that we observed that not all users that create architectural designs will explicitly use that term in their query. They may instead use a query like “award winning building at a lake”. We identified this list of implicit keywords by correlating the terms from architectural glossaries [1] and [2] in their usage within the explit queries. For example, is an architectural term like “window” frequently used in queries that contain “architect” then we added it to the implicit keyword list. As cutoff we defined that each term had to at least occur in 10 % of all explicit cases containing either “architect”, “interior” or “exterior”. The (iii) class of keywords is a list of 941 famous architects from Wikipedia[3]. We included them as users often refer to the style of those architects. Here the full name needs to be used in the query.

By filtering all 58 million queries by these three keyword types we selected 3.81 million queries (6.6 %) that are using at least one keyword types. They include 1.54 million queries (2.6 %) that explicitly contain “architect”, “interior” or “exterior” design; and 419,487 (0.72 %) that are referring to one of the famous architects.

We will in the rest of the analysis distinguish between queries belonging to the *filtered* and *architect* set. The *filtered* query set contains all 3.81 million queries containing any of the three keyword types (explicit, implicit, or architect name). The *architect* set contains only the 1.54 million queries using the explicit keywords “architect”, “interior” or “exterior” design. This implies that *architect* queries also belong to the *filtered* set.

In Figure 2 we also show the frequency of all filtered (red) and architect (green) queries. It is to note that it stays rather constant across the time and does not follow the hype. A reason may be, that most of the people that test out Midjourney run other types of queries, while architects or architecture enthusiasts use the system regularly with in stable numbers.

Word Frequency

We first analyse the most common words used in the filtered queries using any of the explicit, implicit keywords or architect names. Figure 3 presents the frequency of the top 25 words without stopwords. The blue bar is the frequency across all 58 million queries to give an idea of how frequent that word in general is. The red bar is the frequency within the filtered queries and green within the explicit queries. As all three classes are mutually inclusive, is the green bar a subset of the red bar which subsets the blue bar. It is first to note that the top 10 of words has a similar frequency across all three classes. Many of those refer to Midjourney style commands like “detailed”, “realistic”, “cinematic”, “render”. Some terms like “black”, “full” or “portrait” have high overall frequency, but are only used with low frequency in the architectural context. Other terms like “architecture”, “interior”, “house”, and “building” do only occur exclusively within our filtered results, as they are part of our keyword list.

Figure 4 lists the frequency of our explicit and implicit keywords. As these words are part of our keyword list on which we filter, their total frequency is identical and not displayed. It is of note that “architecture” and “interior” keywords are the most and third frequently used words within all the filtered queries. Other important keywords are “house”, “building”, “window”, “floor”, “concrete”, “pool” and “cathedral” to complete the top 10.

Next we analyze in Figure 5 who from the list of famous architects on Wikipedia are the one most frequently queried. The winner is: “Zaha Hadid”. Her organic architecture style is well recognizable and obviously very popu-

[1] `https://www.heritage.nf.ca/articles/society/architectural-terms.php`

[2] `https://en.wikipedia.org/wiki/Glossary_of_architecture`

[3] `https://en.wikipedia.org/wiki/List_of_architects`

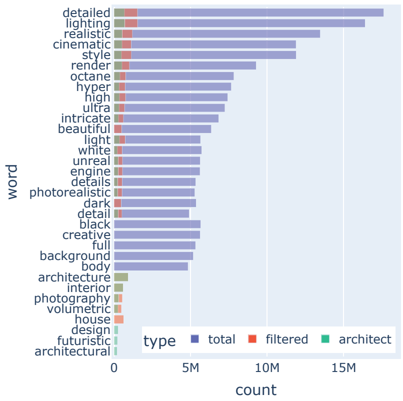


Figure 3: Most frequent words in the filtered queries.

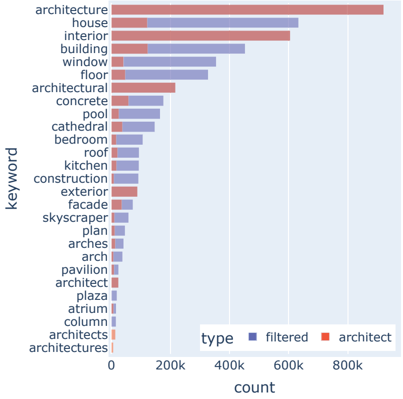


Figure 4: Frequency of keywords in queries.

lar within the Midjourney community. She also has an AI team that may contribute to these numbers with their experiments. Michelangelo is in second place, but often used as a reference for his art style and expansive engineering work and less often for his architectural contributions visible by the low red bar. We see similar behaviour for Adrian Smith and William Morris. The top 10 of architects then continues with Frank Lloyd Wright, Tadao Ando, Frank Gehry, Lebbeus Woods, Kengo Kuma, Peter Zumthor, and Antoni Gaudi that are often used in an explicitly architectural context visible by the larger red bar.

Query Frequency and Collaboration

Most of the queries are not unique as users rerun the same query to iterate trough different variants. The filtered 3.81 million queries reduce to 992 thousand unique queries meaning that each query is repeated about 3.28 times in mean. Figure 6 shows the frequency of the top 1000 unique queries including references by other users. We removed very simple queries like “architecture”, “house” and “architecture rendering” with 824.887, 590.817, and 235.688 calls, respectively. Their popularity illustrates the relevance of the topic, but, they are very short and unspecific and we consider them not representative. The query frequency follows a Pareto distribution. The top three ranking query were executed 2970, 2734, and 1606 times including extended variants. These high numbers are most likely created by scripts that users run like Seneviratne et al. (2022). The distribution is then quickly falling off toward the median of 2 executions.

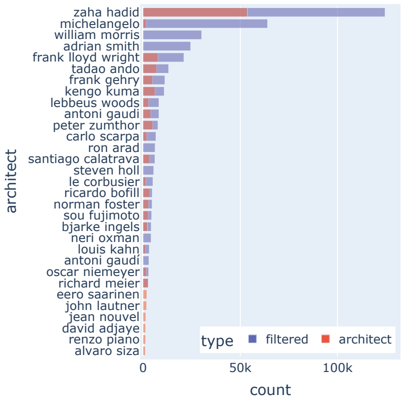


Figure 5: Most frequent of architect names in queries.

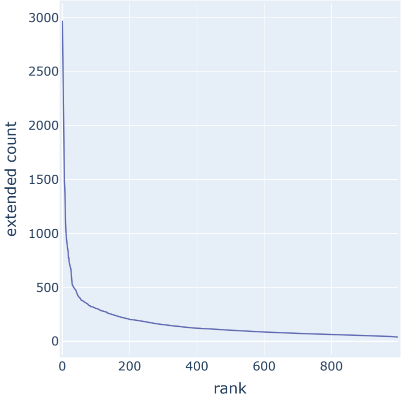


Figure 6: Frequency of the top 1000 prompts.

As these prompts are run on public channels they are visible to other users, who might pick up good ones they like. Every 11th query is reused in mean. Figure 7 shows on the x-Axis that this does not directly correlate to the number of executions (y-Axis aligned with Figure 6). A small trend exists as the top 10, 100, and 1000 queries are reused by 15, 6 and 2 users in mean, respectively. But, the highly frequent queries used by more than 30 users have between 686 and 1731 users and are often pinned as reference example on the “environment” channel in Midjourney. And, the low number number of users for queries with more than 2000 calls supports our assumption that they are scripted.

Query Length and Workflows

We show the mean length of queries in Figure 8 depending on whether they got up-scaled, remastered or left in draft mode. A draft mode image is of low image size and usually contains four variants, so users will normally up-scale or remaster the variants they like. By doing so the user may alter and potentially extend the prompt. We see that with 3 million the majority of queries are left in draft mode. 791 thousand queries are up-scaled in some way (light, medium/beta, max, remastered). The draft mode

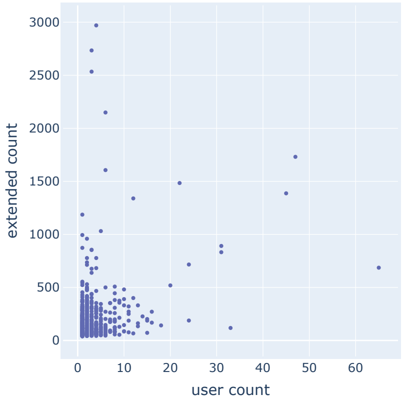


Figure 7: Number of users of the top 1000 prompts.

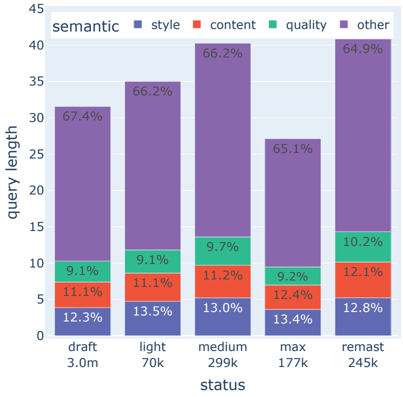


Figure 8: Query Length. x-axis labels show the # of queries.

queries are also significantly shorter with 31 terms in mean than up-scaled ones with 34 and 40 terms for light and medium/beta upscaled versions. Remastered queries contain more than 41 terms. The only exception is the maximum upscale with only 27 terms, which is not available anymore in version 4 of the model. Nonetheless, it shows that high quality queries usually contain more terms. To understand how users use the queries we classified the most frequent 150 terms into three categories: style, content, quality. It is notable that for the upscale and refined queries, the percentage of style terms increases.

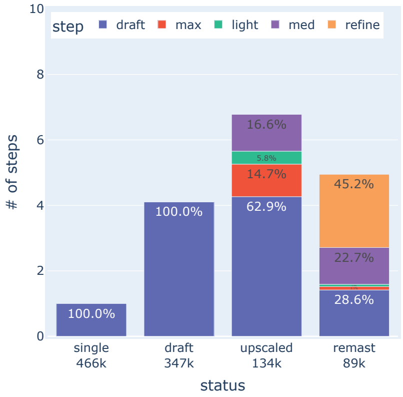


Figure 9: Query Workflows. x-axis labels show the # of flows.

We have shown in Figure 6 that users go trough multiple iterations to find a query that they finally up-scale. To analyze how the user is developing their query we investigate the history of queries modified by the same user. The discord messages do not directly contain a link to the query that a user refined, modifies or up-scales. Therefore, we analyzed the chronology of queries of a user and assumed that a query is a variant of a previous one if it either contains the same or an extended prompt within a 30 minute window. Of the 992 thousand unique queries, about 443k queries are run once (single, 45 %). Most queries are actually improved over multiple iterations as shown in Figure 9. 34.5 % of the queries remain in draft mode even if they are iterated over 4.1 steps. 13 % of the queries are good enough to be up-scaled after an average of 7.1 steps. They are up-scaled after 4.5 draft steps into different variants (light, medium/beta, max), indicating that users actually upscale multiple variants and try out different qualities. The remastered 8 % of queries have about 6.0 iterations. They quickly move after 1.8 draft mode queries into 2.5 remastering steps and 1.7 final upscale steps.

This high number of iterations shows that users usually develop queries over time and do not find the ideal image from start. Users may within this process add quality and style modifiers as discussed for Figure 8.

Word Similarity and Co-occurrence

Next we were interested in understanding which terms are used together and with similar meaning. For this we build a Word2Vec model (Mikolov et al., 2013) from all queries to extract the co-occurrence of terms.

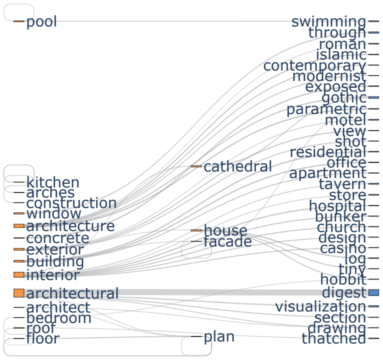


Figure 10: Keywords (left) and co-occurring terms (right).

Figure 10 shows the links between keywords and the most likely connected term. We analyzed this by predicting with the Word2Vec for each keyword on the left the most probably co-located word on the right, weighted by probability. Interesting combinations here are links between floor-plan, architecture-parametric, architecture-digest, building-facade-elevation, or pavilion-roof, swimming-pool. With this it is possible to build an auto-complete function for architectural queries.

Figure 11 shows the similarity between the most frequent terms from Figure 3. This is done by doing a Principal Component Analysis (PCA) of the Word2Vec model to re-

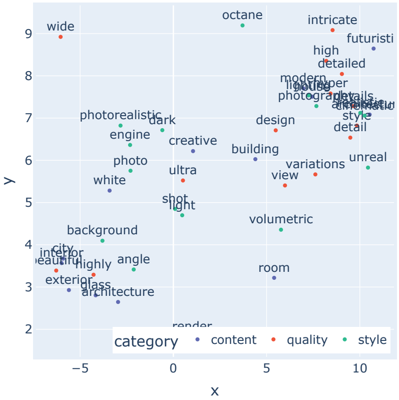


Figure 11: Word Similarity.

duce it into a 2D space, where words that are used in a similar context will be in similar positions. We color-coded the category of each term (content, quality, style) to make it easier to identify clusters. It is notable that style (green) and quality (red) terms form two clusters and content specific terms (blue) are distributed given their diversity.

Figure 12 shows the bigrams we extracted with the Word2Vec model for only the 1.54 million queries containing "architect" or "interior design". These are the most frequent terms occurring together in queries and are often referring to names. The majority of them like Anish Kapoor, Artemisia Gentileschi, Atey Ghailan, Chiharu Shiota, Didier Graffet, Eddie Mendoza, Gerhard Richter, Kar Wai, Naoto Hattori,Pino Daeni, Samson Pollen are artists that are referred to for their specific style. However, there are some architects like Coop Himmelblau, Feng Zhu, Herzog Meuron, Oscar Niemeier, Ricardo Bofill, Shigeru Ban, Sou Fujimoto, Velerio Olgiati that we not all had in the list of architects.

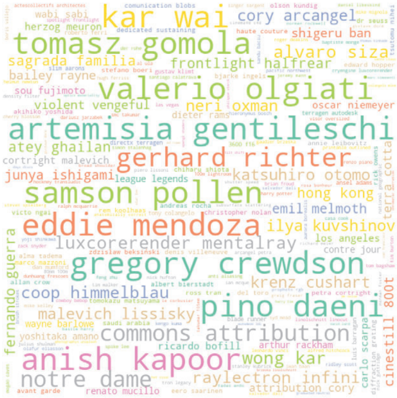


Figure 12: Bigrams from architecture prompts.

Top Architectural Prompts

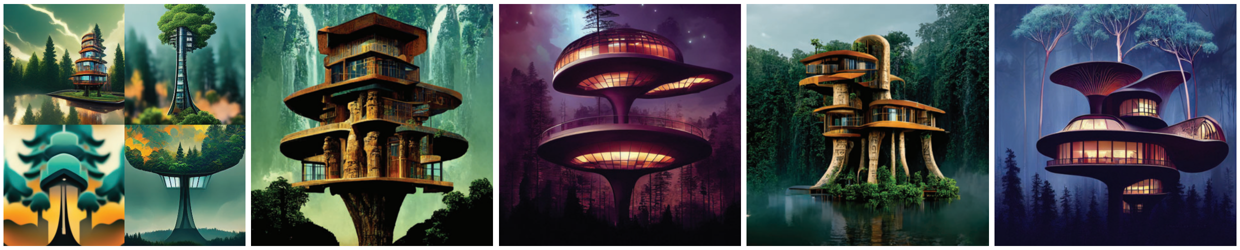
Figures 13, 14 and 15 show example results from some of the most frequently reused queries, to illustrate the development process they go through. Midjourney first responds to a prompt with an image containing four generation variants. From there users usually explore multiple different directions of iterative refinement to finally arrive at one or multiple up-scaled results. These paths can be quite different and sometimes include dozens of intermediate steps of generating new variants, generating up-scaled versions, remixing prompts and remastering images that up-scaled well. Processes that end up producing results of high quality usually contain far too many iterations to show in full in this paper.

A likely reason for the frequency of these particular prompts is that they were either executed by a bot or a small group of very dedicated users. An automated bot would make it possible to employ a brute force approach, in which thousands of images are generated, downloaded and then hand picked from an unordered collection. The advantage of such an approach is that the bot will go down generation paths that a human would dismiss, and might thus unlock possibility spaces that would have otherwise remained unexplored. Despite their likely automated origins, we will still use the most popular results from these prompts to show different ways of how to refine a prompt into a set of desired results.

Figure 13 shows the most popular architectural prompt in our data set with 2 query variants with 2970 and 2970 executions from 3 and 6 users. These results are based on Midjourney version 3 and show an attempt at generating a modern, semi-fantastical tree house. Figure 13 (a) contains a first prompt result with four very different directions. Such varied results are usually a perfect starting point to push in the correct direction. Through multiple lines of iteration we arrive at results (b), (c) and (d). It is very apparent that these results feature different materials, different painting styles and different coloration, even though they all started with the exact same prompt. This demonstrates the power of iterative refinement. A final, very high quality result can be seen in (e), showing how it is possible to explore the results of Midjourney and to uncover whole new styles in architectural design.

While the first prompt teetered the line between realistic design and fictional construction, Figure 14 is firmly set within photorealism. This query is in 2nd place with 1606 calls from 6 users and shows how Midjourney has progressed from model version 3 to model version 4. We begin with four stylistically similar building variants with very different architecture in Figure 14 (a). Instead of directly moving on to up-scaling from one of these variants, it can pay to first generate new variants that are much more focused in their content, as best seen in Figure 14 (b). The ideal variant can then be up-scaled as shown in Figure 14 (a). Figure 14 (d) and (e) show a similar progression, with less uniform variants, to show how we can use these models to generate many different finished high-quality results in very short order.

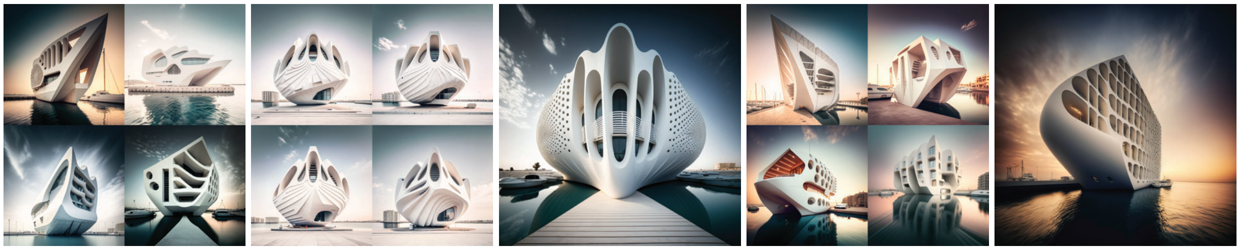
Figure 15 places 3rd with two queries with 1186 and 873 calls from a single user. It is an interior design focused example in which a certain style is set from the beginning and then the variant refinement happens over content instead of style. This could be used to plan multiple rooms



(a) Starting Variants (b) Refinement direction 1 (c) Refinement direction 2 (d) Refinement direction 3 (e) A final result

ultra modern huge treehouses between big waterfalls the deep forest art nouveau shapes with arabic ornaments futuristic architecture sphinx highly detailed ancient egyptian pylons and steles pharao ramses cinematic lightning

Figure 13: The most popular architecture-related query on the official Midjourney server. The prompt is shown with: content words in blue, style words in red, and quality words in green.



(a) Starting Variants (b) Variant refinement 1 (c) Up-scaled result 1 (d) Variant refinement 2 (e) Up-scaled result 2

award-winning architecture photography a building looks like boat limassol marina harbor luxury property cubist parametric nautilus shape structure modernist villa white renzopiano photography whitegranite futuristic cozy architectural design architectural photography archdaily architecture digest magazine 16k natural lighting soft lights atmospheric ambiance immersive environment

Figure 14: A very striking architectural query for Midjourney model version 4.

or room variants for a building, while keeping a certain style intact. Figure 15 (a) shows how multiple variants often have slight variations in style, from which the trained eye can pick out an ideal version and further refine. The room contents can then change by iterating through variants and remasters often enough. Figure 15 (b) and (d) show up-scaled results that obviously were generated from quite different variant samples. The remaster step for each of them shown in Figure 15 (c) and (e) demonstrates how remastering is another way to get variations on content, while also retaining stylistic elements.

All these approaches show one common thread: A user iterates through different steps, in order to slowly converge on one or more results of a desired style and content. The variety in results can be quite staggering. Figure 16 shows a collection of other popular query results, that illustrate that the AI art generator can create in a short time stunning architectures that combine multiple styles and presentation techniques. We expect that this will influence architectural styles in the future.

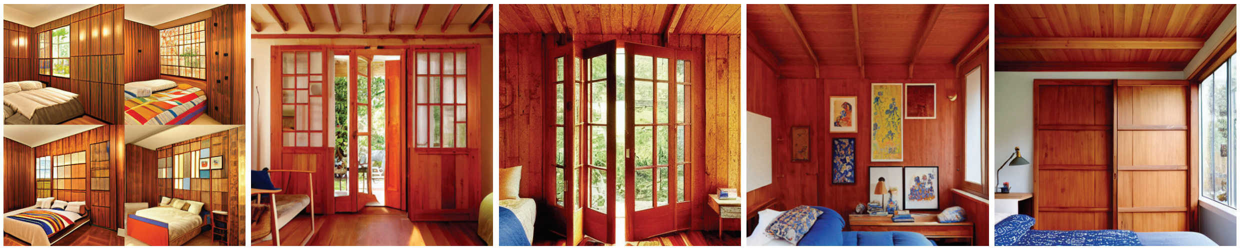
Conclusion

In this paper we investigate the usage of AI art platforms at the example of Midjourney. We analyzed quantitatively millions of publicly available user queries to extract common prompt contents, how they develop and how users cooperate. We qualitatively discussed the most popular queries in detail. A similar analysis for other tools like DALL·E or Stable Diffusion would only be possible if the creators make large query collections public.

Even within this paper, the quality difference between the two different versions of the Midjourney model is very apparent. Considering that they are just few month apart, this kind or rapid progress is likely to continue for some time. With the next generations, capabilities will evolve, new workflows will develop and new contenders will enter the market. For Midjourney in particular, editing tools that exist in other models will likely be included, like inpainting or outpainting capabilities.

However, the current generation of image generation AI models has still some distinct limitations that reduce applicability in architectural use cases. Most of the usefulness is likely in use cases like ideation, architectural collages, and the creation of building variants. Our examples show that the tools are already able today to create stunning design images that will influence future architecture, whose construction will also be enabled by new digital construction methods.

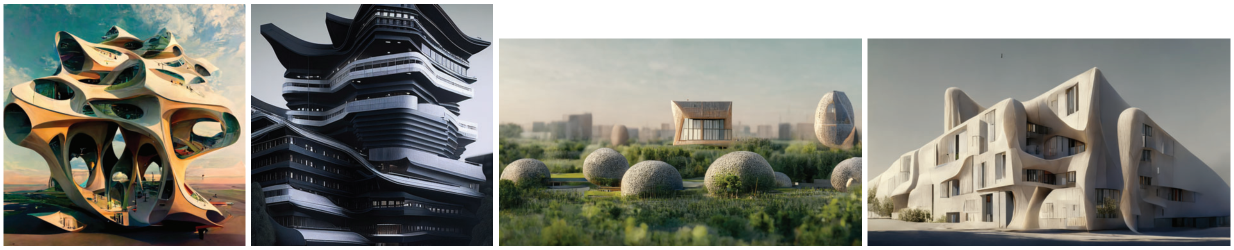
More complex use cases like generation of floor plans or even 2D or 3D models requires far more specific training



(a) Beginning variants (b) Refinement result 1 (c) Remastered result 1 (d) Refinement result 2 (e) Remastered result 2

photorealistic interior design bedroom plywood paneled ceiling reclaimed wood flooring cobalt blue wardrobe french doors opening a garden windows over midcentury bureau teak platform bed white linen bedding liberty london bedding tapestry danish wall sconces built josef frank commune design cottage style swedish variations

Figure 15: Different views of interior spaces following a specific design language, generated in model version 3.



(a) Rank 15 (b) Rank 17 (c) Rank 34 (d) Rank 46

Figure 16: A collection of popular architectural generation results.

and contextual understanding on the part of the models. The data to provide such training however already exists in the form of Building Information Models (BIM). Diffusion models are not directly applicable to them as the 3D model cannot be gradually diffused. However, we will in future research create specially trained models in the near future that work on 2D and 3D pixel representations and create powerful tools that automate workflows from the architectural design to the planning stage.

References

AEC Magazine (2022). AI special edition of AEC Magazine, volume 122. X3DMedia, London.

Borji, A. (2022). Generated faces in the wild: Quantitative comparison of stable diffusion, midjourney and DALL-E 2. arXiv preprint arXiv:2210.00586.

Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. In NIPS, volume 33, pages 6840–6851.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In NIPS, volume 26.

Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., and Chen, M. (2022). Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In ICML.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In ICML, pages 8748–8763.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. (2022). Hierarchical text-conditional image generation with CLIP latents. arXiv preprint arXiv:2204.06125.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. (2021). Zero-shot text-to-image generation. In ICML, pages 8821–8831.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In CVPR, pages 10684–10695.

Seneviratne, S., Senanayake, D., Rasnayaka, S., Vidanaarachchi, R., and Thompson, J. (2022). DALLE-URBAN: Capturing the urban design expertise of large text to image transformers. In Int. Conf. on Digital Image Computing: Techniques and Applications.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. (2015). Deep unsupervised learning using nonequilibrium thermodynamics. In ICML, pages 2256–2265.