AI for Conceptual Architecture:

Reflections on designing with text-to-text, text-to-image and image-to-image generators

Horvath, Anca-Simona; Pouliou, Panagiota

Published in:
Frontiers of Architectural Research

DOI (link to publication from Publisher):
https://doi.org/10.1016/j.foar.2024.02.006

Publication date:
2024

Document Version
Version created as part of publication process; publisher's layout; not normally made publicly available

Citation for published version (APA):
Horvath, A.-S., & Pouliou, P. (2024). AI for Conceptual Architecture: Reflections on designing with text-to-text, text-to-image and image-to-image generators. *Frontiers of Architectural Research*, *13*(3), 593-612. https://doi.org/10.1016/j.foar.2024.02.006

ARTICLE IN PRESS
+ MODEL
Frontiers of Architectural Research xxx (xxxx) xxx

Available online at www.sciencedirect.com
ScienceDirect
journal homepage: www.keaipublishing.com/foar

RESEARCH ARTICLE

AI for conceptual architecture: Reflections on designing with text-to-text, text-to-image, and image-to-image generators

Anca-Simona Horvath [a,*], Panagiota Pouliou [b]

[a] Research Laboratory for Art and Technology, Aalborg University, Aalborg 9000, Denmark
[b] Lighting Design Lab, Aalborg University, Copenhagen 2450, Denmark

Received 13 October 2023; received in revised form 30 January 2024; accepted 17 February 2024

KEYWORDS
Machine learning;
StyleGAN2-ADA;
RNN TensorFlow;
VQGAN + clip;
AD journal;
eVolo;
Conceptual design;
Architectural design

Abstract In this paper we present a research-through-design study where we employed text-to-text, text-to-image, and image-to-image generative tools for a conceptual architecture project for the eVolo skyscraper competition. We trained these algorithms on a dataset that we collected and curated, consisting of texts about and images of architecture. We describe our design process, present the final proposal, reflect on the usefulness of such tools for early-stage design, and discuss implications for future research and practice. By analysing the results from training the text-to-text generators we could establish a specific design brief that informed the final concept. The results from the image-to-image generator gave an overview of the shape grammars of previous submissions. All results were intriguing and can assist creativity and in this way, the tools were useful for gaining insight into historical architectural data, helped shape a specific design brief, and provoked new ideas. By reflecting on our design process, we argue that the use of language when employing such tools takes a new role and that three layers of language intertwined in our work: architectural discourse, programming languages, and annotations. We present a map that unfolds how these layers came together as a contribution to making machine learning more explainable for creatives.

* Corresponding author.
E-mail addresses: ancah@ikp.aau.dk (A.-S. Horvath), papo@create.aau.dk (P. Pouliou).
Peer review under responsibility of Southeast University.

https://doi.org/10.1016/j.foar.2024.02.006
2095-2635/

Please cite this article as: A.-S. Horvath and P. Pouliou, AI for conceptual architecture: Reflections on designing with text-to-text, text-to-image, and image-to-image generators, Frontiers of Architectural Research, https://doi.org/10.1016/j.foar.2024.02.006

1. Introduction

Using artificial intelligence (AI) to generate works of art, design, and architecture has been an emerging research field in the last decades, with experiments made as early as the 1960s, inspired by Alan Turing's question "Can computers think?" (Turing, 1950). In architecture, AI allows the exploration of large design spaces and the optimization of certain aspects that can be expressed in a numerical format, such as areas, volumes, material use, or energy consumption (Mostafavi et al., 2023; Tamke et al., 2018; Xu et al., 2022). In various case studies, we see that the problems these algorithms are applied to are as broad as the field itself, with examples ranging from floor plan generation (Chaillou, 2019), facade and section generation (Güzelci, 2022), urban scale planning (Zhong et al., 2022), or optimization of path planning of robotics systems in digital manufacturing for building construction (Nicholas et al., 2020). AI has gained so much popularity during the last few years that two books sharing parts of the same name *Architecture in the Age of Artificial Intelligence* (Bernstein, 2022; Leach, 2022)—written by different authors and published by different publishers—were released in 2022 alone.

However, it is the emergence of text-to-text, image-to-image and especially text-to-image machine-learning powered generative tools such as OpenAI's Dall-e (OpenAI, 2023), Midjourey (Midjourney, 2024) or StableDiffusion (CompVis, 2023) that have had the most profound influence in architectural visual culture, as these tools impacted our field "suddenly, severely and seemingly out of nowhere" (Steinfeld, 2023). Recently Patrick Schumacher declared that at their office, Zaha Hadid Architects, text-to-image generative tools are used in "almost all projects" in the early stages of concept generation (Barker, 2023b). Other large architectural practices with an important influence on the profession report using text-to-text, text-to-image or image-to-image generators. For example, Coop Himmelb(l) au have been using image-to-image generators trained on images produced by their office for conceptual design in the last few years (Bolojan, 2022), while Hickock Cole presented a building designed with the use of text-to-text generator ChatGPT and later text-to-image generator Midjourney (Barker, 2023a).

It is therefore timely to ask: how can machine-learning tools for text-to-text, text-to-image, and image-to-image generation trained on architecturally relevant data be employed in design processes for conceptual architecture? Further, what are the opportunities and challenges of using these tools for early-stage architectural design? While the use of text generative tools has been often reported by architects in the past months, to our knowledge, research studies that trained text-to-text generative models on architectural texts have not been conducted so far. It has been shown that text-to-image generators hold high potential for architectural visualization, yet there is a need to further research so-called prompt crafting and the relationship between prompts and visual outputs (Karadag, 2023b; Milošević et al., 2023; Stigsen et al., 2023). As it has been argued that the only way to make effective use of AI frameworks for architectural design is for architects to train them on architectural data (del Campo et al., 2021; Bolojan, 2022), there is a need to further investigate how these tools can be tailored for use in architectural design.

This study deals with these questions by making a threefold contribution. First, we describe a methodology of using text-to-text, text-to-image and image-to-image tools in a design process for a conceptual architectural project for the eVolo skyscraper competition. We provide practical insights, including two workflows, detailing the entire design process, and discuss how and where these tools were useful. The evaluation of the results of the algorithms was conducted subjectively (by us as designers), as would be the case when such tools are included in a conceptual design process. Second, we consider the outcome of this process as a contribution on itself, as it demonstrates possible outcomes of such a methodology in practice. Additionally, we reflect on this artifact by discussing the theoretical implications of the integration of machine learning tools for conceptual architecture. Third, based on this experience, we argue that in designing with generative AI for text-to-text, text-to-image and image-to-image tools, different types of language interweave, and we unpack them into a map based on our design process. This map can help make machine-learning-powered frameworks more explainable for other architects, designers, or artists and can serve as a resource for creatives interested in exploring the implications and possibilities of designing with AI-powered frameworks. It can also be helpful for human-computer interaction or explainable AI scholars or practitioners interested in understanding the specific experiences of designers using AI for architecture. More generally, our study contributes to a critical understanding of the relationship between AI tools and creative work.

The rest of the paper is structured as follows: we first give a brief overview of the current state of related work on using machine learning in architectural design focusing on the use of text-to-text, text-to-image, and image-to-image generators and conceptual architecture. Next, we describe our design process and results, a process that consisted of two iterations where the second iteration builds on the first one. In Section 3, we describe the first iteration, which includes collecting a dataset of texts and images about architecture and training machine learning algorithms on this dataset. In Section 4, we present the second iteration, showing how the previous step informed the final proposal that we submitted. Next, this two-step design process and its results are analyzed in Section 5. This section is divided in two subsections, with the first focusing on how we experience designing with machine learning algorithms and the implications this can have for future research and practice in architecture, while in the second subsection we discuss the role of language in this process. We conclude by summarizing our findings in Section 6.

2. Machine learning in architecture

According to Domingos (2015a; 2015b), there are five main schools of thought in machine learning, corresponding to five principles by which computers create new knowledge, namely: (1) by filling gaps in existing knowledge (symbolists), (2) by emulating the brain or trying to reverse engineer our current understanding on how the brain works (connectionist), (3) by simulating evolution (evolutionist), (4) by systematically reducing uncertainty (Bayesian), and (5) by noticing similarities between old and new knowledge (analogyst). The *symbolist* strand has its origins in logic and philosophy and has inverse deduction as its "master algorithm", as Domingo calls them. The *connectionist* strand has its origins in neuroscience and has backpropagation as the master algorithm. The *evolutionist* strand has its origins in evolutionary biology and has the genetic algorithm as its master algorithm. The *Bayesian* strand originates from statistics and its master algorithm is probabilistic inference. Finally, the *analogyst* strand with origins in psychology and kernel machines as the master algorithms.

Today, out of the five strands of machine learning, one of the more popular is the *connectionist* strand with backpropagation as the engine that drives deep learning which is the paradigm behind neural networks. There are different ways to categorize how backpropagation algorithms work, and among the most simple distinctions is to differentiate between supervised, unsupervised, and reinforced learning. Supervised learning refers to a process where the training data for the machine-learning model is labeled, or annotated (Delua, 2021). Unsupervised machine learning attempts to model patterns found among observed data without having any specified output values (Delua, 2021). In reinforced learning, the model simulates a sequence of steps from which it collects rewards and calibrates its behavior to improve a score (Chaillou, 2022). Popular text-to-text generators such as ChatGPT, text-to-image generators such as Midjourney, and image-to-image generators such as StyleGAN2 (Karras et al., 2020) are all forms of neural networks.

Text-to-text generators function as probabilistic models that predict the next most likely letter in a word or the next most likely word in a sentence. These generators are trained or very large datasets consisting of corpora of natural language. One of the most widely used corpora used for training language models is WordNet (Fellbaum, 1998) which has been used by major search engines for information retrieval. WordNet contains "nouns, verbs, adjectives, and adverbs grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations" (Princeton, 2010). While there are studies that look into the language used when discussing architecture (Horvath, 2022c) showing how this language is specific and different from general discourse, there are no studies that trained text-to-text generators specifically on architectural texts so far.

Text-to-image generators have been used more often, by comparison, in architectural practice and research over the last few years. They are based on datasets of images that have text annotations associated with them, for example a picture of dog would have the text tag "dog". The largest annotated image dataset is ImageNet (Deng et al., 2009a). Its images were collected by querying internet databases, using nouns found in the WordNet dataset. ImageNet only uses the nouns in WordNet for image annotation, as it was considered that "nouns are things that pictures can represent and that would be sufficient to train machines to automatically recognize objects" (Crawford, 2021). ImageNet currently contains over 14 million images annotated under close to 22 thousand synsets. In this way, language models, or text-to-text generators, and text-to-image generators are connected through their logic.

The images in ImageNet are labeled or annotated with the use of crowd-sourcing by workers from around the world. The creators of ImageNet presented the annotators with a set of candidate images and the definition of target synsets, and asked them to verify whether each image contains objects in the synset (Deng et al., 2009a). In other words, the query for the images themselves was done using the synsets present in WordNet and the annotations subsequently followed the sysnsets in the WordNet dataset. According to Kate Crawford (2021), this makes the structure of the ImageNet "labyrinthine, vast, and filled with curiosities". ImageNet has been widely criticized: only the nouns in WordNet are selected to describe images, the images have been scraped from the internet and have been labeled by people, under time constraints all while being paid precarious wages (Crawford, 2021; Sanjay et al., 2023). Examples of employing text-to-image generators for architecture include Stigsen et al. (2023) who used diffusion models in architectural education and argue that such tools can enhance the ideation phase by becoming sources of inspiration for the students and can help with streamlining 3D modeling processes. Similarly, Tong et al. (2023) used the text-to-image generator Midjourney in a workshop with first-year students and discussed how AI tools might influence students' development of visualization skills and could be used together with sketching. One of the challenges discussed by studies where text-to-image generators are employed relates to so-called prompt-crafting: finding the right words to describe architectural ideas, and managing to control the output of these tools (Tong et al., 2023; Stigsen et al., 2023).

Image-to-image generators are a subtype of generative adversarial neural networks (GANs) that use images as their data input. They have gained widespread attention in design and artistic fields since 2014 when they were introduced (Goodfellow et al., 2014). The generation of floor plans using GANs was explored by a series of projects (Chaillou, 2020; Carta, 2021; Karadag et al., 2023; Nauata et al., 2020; Rodrigues and Duarte, 2022; Tarabishy et al., 2020), with some claiming to train GANs to "hallucinate" about architecture (del Campo et al., 2021). Facade generation was explored in Kelly et al. (2018), Sun et al. (2022), and Zhang et al. (2022), while Steinfeld (2022) explored generating perspectives and Alacam (2022) used a cycle GAN to match historical maps of Istanbul to their current counterparts. Karadag (2023b) used the Stable Diffusion text-to-image generator together with its function that takes an image as a starting point, collected images of architectural sketches from the internet, and used different text-prompts to generate detailed renderings. Güzelci (2022) created a dataset of section drawings for a type of Turkish funeral architecture and used a GAN to successfully predict cap geometry for renovation work of these artifacts. Similarly, Karadag (2023a) used a conditional GAN to predict missing/damaged parts of early Ottoman tombs and found that given a sufficiently large dataset, good results were possible.

3. Design Iteration 1

In our design process, we used text-to-text, text-to-image and image-to-image machine learning generators. These algorithms were trained on a dataset that comprises of both text about and images of architecture that we collected and curated. When we started working on this submission, we were curious whether we could produce a complete proposal for the competition by only using these algorithms, trained on architecturally relevant datasets. In the end, as we will show, the results from training the algorithms were used as a starting point that informed a proposal.

3.1. Design brief

We started our design process by trying to create a design brief after reading the eVolo skyscraper competition call for the respective year, 2022. The competition called for "Outstanding ideas that redefine skyscraper design, through the implementation of novel technologies, materials, programs, aesthetics and spatial organizations; along with studies on globalization, flexibility, adaptability and the digital revolution. It is a forum that examines the relationship between the skyscraper and the natural world, the skyscraper and the community and the skyscraper and the city. The participants should take into account the advances of technology, the exploration of sustainable systems, and the establishment of new urban and architectural methods to solve economic, social and cultural problems of the contemporary city, including the scarcity of natural resources, and infrastructure, and the exponential increase of inhabitants, pollution, economic division, and unplanned urban sprawl. The competition is an investigation of the public and private space and the role of the individual and the collective in the creation of a dynamic and adaptive vertical community. It is also a response to the exploration and adaptation of new habitats and territories based on a dynamic equilibrium between mana and nature—a new kind of response and adaptive design capable of intelligent growth through the self-regulation of its own systems" (eVolo, 2022).

The call is very broad, and there are no site constraints (i.e., the skyscraper could be placed anywhere in the world, on another planet, underwater, or nowhere—on an imagined site). Similarly, there are no references to what kind of program the skyscraper should represent (i.e., whether it should contain residential, commercial, hospitality, or mixed uses). The scale was not defined either, other than that the project should represent a high-rise.

In order to gain a better understanding of the potential of machine learning tools for architectural design, we decided to leave the design brief as open and as free as the call of the competition and to apply machine learning algorithms to solve this brief, as was. In this way, our agency as designers was kept to the minimum: we merely curated a dataset, used machine learning algorithms to automate the creation of a large design space, and then re-curated this large design space. In the next subsection, we describe how we collected a dataset of text and images.

3.2. Dataset: text and images

The project was based on two datasets: the first dataset consisted of architectural texts and the second was made up of images of conceptual architectural projects. These datasets were collected from two sources: the journal Architectural Design (AD) and the eVolo skyscraper competition. AD has been at the forefront of architectural thought since its inception, and in the last 30 years has published a large portion of the discourse surrounding computational design in architecture to the point that Mario Carpo has stated that while not all things related to computational design in architecture have been published in AD, a lot of them have (Carpo, 2012, 2017). On the other hand, the eVolo skyscraper competition is one of the more famous architecture competitions in the world focusing on "technological advancements" in the field (eVolo, 2022). Started in 2006, it sometimes receives more than 500 submissions per year from over 150 countries. In this way, the dataset is *representative* in the sense that it is relevant for the project at hand, and given the importance of both AD and eVolo, the results of working with these datasets can offer broad and generalizable insights.

The text dataset includes texts from AD and eVolo as follows. From AD, the titles of the issues, titles of articles, and texts of the articles all between 2005 and 2022 were collected. From eVolo, the titles and abstracts of all the winning projects and honourable mentions between 2006 and 2021 were collected, building on our previous work (Horvath, 2022a, 2022b, 2022c). The texts were organized by source (AD journal and eVolo) and by year, and in total, this dataset consists of around 4.6 million words. The texts collected from eVolo are under a Creative Commons license and therefore can be stored and used. AD has free access to titles of issues, articles, and keywords associated with Introduction articles, however, accessing the main texts of articles requires a subscription. Nevertheless, AD encourages the use of its text database for research purposes. The part of the dataset that can be made public is available in (Horvath, 2022b).

The dataset of images contains the posters of all the winning projects and honorable mentions of the eVolo skyscraper competition between 2006 and 2021, the year prior to our submission, in total 480 images. These images also sit under a Creative Commons license.

3.3. Training the machine learning algorithms

Once we collected these two datasets, we wanted to use them in different ways, to get a large sample of possible results. We aimed to compare the results, in order to gain a better understanding on the ones that could be more useful for architects who want to work with machine learning in early-stage design.

The two datasets were used in two workflows described in Fig. 1 that involved three machine learning algorithms: one that generates new texts based on existing texts, one that generates new images based on text (both part of Workflow A), and one that generates new images based on existing images (part of Workflow B). We call the newly generated texts and images *hybrid* text and *hybrid* images, respectively. This is to differentiate between natural

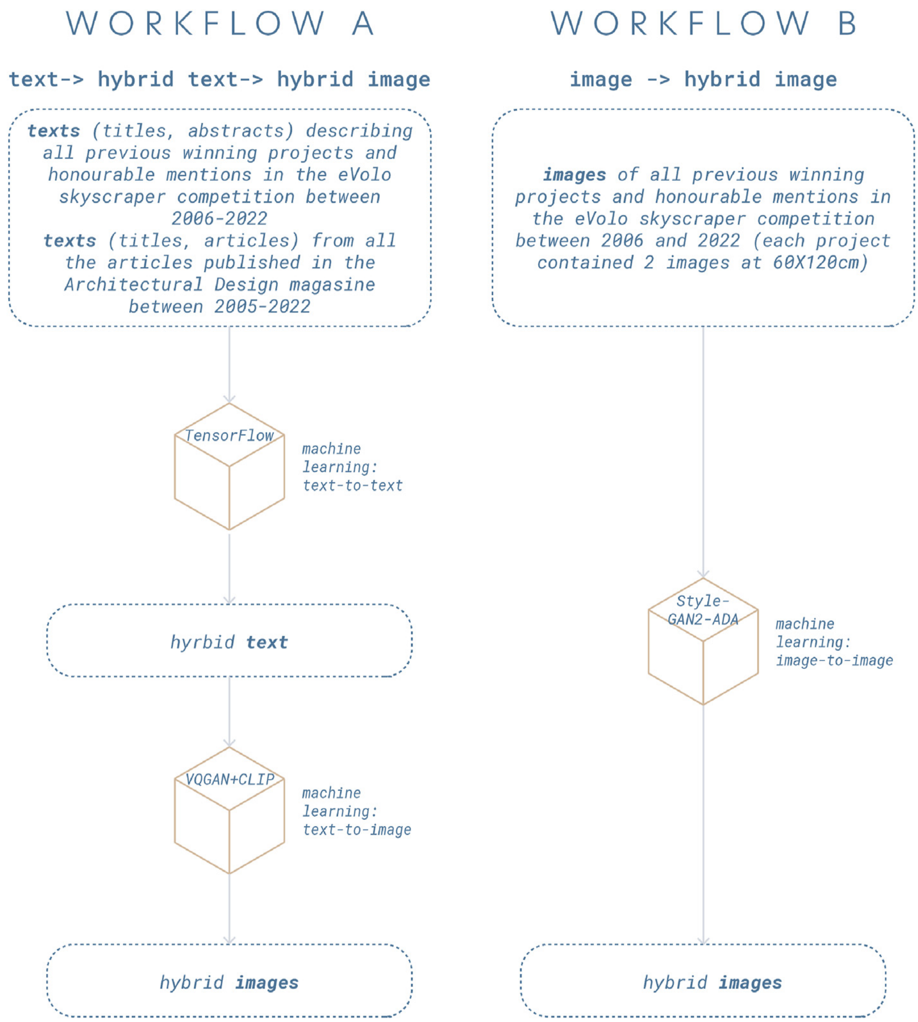

Fig. 1 The two workflows used to generate hybrid text and hybrid images based on the dataset we curated.

language (as the one used when we discuss architecture) and artificial language (as programming languages). Pouliou et al. (2023) describe the *hybrid* as a way of seeing for machine learning algorithms, as the interpolations between individual data points in latent space.

In Workflow A, we used the text dataset and first employed the TensorFlow2 library for text-to-text generation (Flow, 2019), that we ran using the Google Collab platform and trained with different parameters (from 2 to 10 epochs, different starting words or phrases, such as "the skyscraper is sustainable" or "the skyscraper is designed to" or "Babel is a building that") that resulted in hybrid texts. We used snippets from these hybrid texts as prompts to train the text-to-image generator VQGAN + clip (VOGAN, 2023; Crowson et al., 2022). VQGAN + clip is pre-trained with several image datasets, and we choose the versions trained with the "imagenet 16384" (Deng et al., 2009b) and "coco" (Lin et al., 2015) and this process generated a series of hybrid images.

In Workflow B, we used the image dataset to train the StyleGAN2-ADA image-to-image generator (Karras et al., 2020) and generated hybrid images based on images of all the previous submissions made to the competition before the year of our submission.

The TensorFlow2 library was chosen because at the time of designing the proposal (early 2022), it was the most advanced tool which also allowed us to train it with a large dataset of architectural texts. ChatGPT3 became available at that time, but did not allow the use of a large dataset, as the one we had prepared, for training. We used a character-based recurrent neural network (RNN), meaning the hybrid texts were generated by predicting the most likely new character in a sequence. Consequently, the hybrid texts sometimes contain words or punctuation that do not make sense (e.g., "Acc shirld divessous in order to fill and atmosphere. We suggest the terraforming of permafrost by a design destrinate a loog, the mootwork hasing thes"). The VQGAN + clip text-to-image generator was chosen because it was the most advanced tool for this purpose at the time. Midjourney and Stable Diffusion had not been released yet, while Dall-e only allowed the use of short sentences for its training. The StyleGAN2-ADA was chosen because our image dataset was relatively small and at the time of designing *Assembled-Growth Babel*, this implementation was the most effective at generating results from such small image datasets.

3.4. Results from the two workflows: training text-to-text, text-to-image, and image-to-image algorithms

Figure 2 shows parts of the design space that resulted from Workflow A, where hybrid texts were generated based on the text dataset, and then, hybrid images were generated from the hybrid texts.

Looking at the **hybrid texts** (see Fig. 2, top), it is interesting to notice how they both resemble architectural discourse, and bring together words that respond to the call of the competition itself, putting together concepts such as *sustainability*, *self-sufficient*, or *organic architecture*. These words are sometimes assembled in ways that make sense, and they gather concepts that could be useful to describe an architectural proposal, e.g., "The Open-Air produce market is designed for urban centers and for regions where the climate is harsh, providing food security and a new model of sustainable urban development". However, when reading this hybrid text closely, we found it to be lacking logic, or simply seem un-finished, as though it has no clear ending. Due to this, no matter what starting

WORKFLOW A

text -> hybrid text -> hyrbid image

Is a skyscraper designed to produce **energy, clean water and food**. The Open-Air Produce **Market is a self-sufficient urban farm that collects its own rainwater, produces its own food, and generates its own energy**. In combination with the eVolo Garbage Skyscraper, a refined version of the winning proposal from the 2017 competition, the Open-Air Produce Market provides a blueprint for a completely self-sufficient city. The Open-Air Produce Market produces its own energy with a wind turbine and solar panels, harvests rainwater, and creates year-round growing conditions with a minimum of artificial lighting. The produce is grown within a greenhouse that allows for the cultivation of fresh fruits and vegetables year-round, even during wintertime in the northern latitudes. A shading structure and a green wall are used to regulate temperature and humidity, while a multi-purpose terrace allows for a wide range of uses, ranging from recreation to commerce. "The **Open-Air Produce Market** is a city farm that is designed to be a new form of public space that is easily accessible by all. The structure is conceived using parametric design to ensure maximum efficiency and a high degree of customization for specific locations. The design adapts to the new urban culture that combines technology and nature to create new forms of organic architecture." "The Open-Air **Produce Market** is designed for urban centers and for regions where the climate is harsh, providing food security and a new model of sustainable urban development." "The structure can be adapted to provide a wide range of functions, including restaurants and retail stores, **commercial greenhouses, and an urban farm**." "To enable maximum flexibility and a high degree of customization, the structure is designed using parametric design. All construction materials are customizable and easily **assembled**, and the structure is made of material that is both durable and recyclable." "The Open-Air Produce Market is a city farm that is designed to be a new form of public space that is easily accessible by all. The structure is conceived using parametric design to ensure maximum efficiency and a high degree of customization for specific locations. ***The design adapts to the new urban culture that combines technology and nature to create new forms of organic architecture.***"

Fig. 2 Part of the design space resulting from Workflow A. Top: hybrid texts were generated using TensorFlow2 trained on the dataset of architectural texts. Bottom: we used snippets from the hybrid texts as parameters (prompts) for training VQGAN + clip.

prompt we used for training TensorFlow2, we could not generate texts that would explain how a skyscraper would work. For example, the hybrid text snippet "How much heat they need in the summer to grow, how tolerant they are to drought, when they do know to stop growing in the Autumn, these all of things are genetically determined in the tree. Therefore, areas harboring trees' populations that have adapted to specific environmental conditions or have distinct characteristics will be defined as 'Genetic conservation units' (European strategy for genetic conservation of forest trees)" seems coherent, and is interesting in itself, as it mirrors both current preoccupations within architecture, such as sustainability, and bio-technologies and has the structure of an architectural text, i.e., a European strategy is quoted. Sometimes texts lack coherence, but this is not always the case, we find, e.g., in the following hybrid text: "The structure as well as physica long healthcact of the interaction between natural and hularial a patar and belonging in their briadg gine. It such as the folding structural unit for mass transportation and rapid installation can be flexible to be connected to this specific distribution system for alternative use of the pipelines. New system into with the steel and glass structures of downtown Vancouver, offering a contemporary interpritate murs and grings about 50 years, and their combustion will be principally charged with the main than it starts washing away or blowing away", the sentence "It such as the folding structural unit for mass transportation and rapid installation can be flexible to be connected to this specific distribution system for alternative use of the pipelines" could be considered coherent in itself, but it looses meaning in the context. We saw value in the hybrid texts, as they allowed us to look at architectural discourse through a new lens, and to identify patterns we would have otherwise missed.

The **hybrid images**, generated in Workflow A, based on the hybrid texts (see Fig. 2, bottom) are also intriguing. We created a large number of iterations, some are presented here. We learned what others have observed previously: using fewer words and generally less text will produce more coherent images. When we used longer prompts (of for example 150 words) of hybrid text, there was no obvious logical connection between the texts and the hybrid images (see images 3a–3d in Fig. 2 that were generated using different snippets of 100–150 words from the hybrid text at the top of Fig. 2). In order to create shorter prompts, we needed to reduce the hybrid texts to the most representative concepts that they contained. We conducted a qualitative analysis of the hybrid texts (by printing them out and annotating the concepts we found most often in these texts). The prompts we created through this analysis included "biology, architecture and quantum physics in a new paradigm for people" (Fig. 2, image 2b), or "the design adapts to a new urban culture that combines technology and nature to create new forms of organic architecture" (Fig. 2, image 2d). One of the more interesting results we created is image 1a in Fig. 2: it reminds the viewer of cut skin, and blood, and has little in common with the other images. This image was created using the hybrid text "communett cutturisy exblosth". This hybrid text was generated by stopping the training for the text-to-text generator early (at only 2 epochs). We find this result important because it consists a so-called creative mistake: we were trying to design a submission for an architectural competition. As part of this process we could never have come up with such an image. For some of the hybrid images generated in Workflow A, the aesthetic is recognizable (i.e., one could learn that these images are created using the VQGAN + clip framework, see images 2a–2d in Fig. 2), but this is not the case for image 1a.

Meanwhile, Fig. 3 shows part of the design space resulting from Workflow B, where **hybrid images** were generated from the image dataset consisting of the posters submitted to the eVolo competition. The hybrid images display certain aesthetic qualities (of strangeness, as del Campo (2022) calls them), that are interesting in architectural visualization, but also for gaining an overview of the prevalent visual and geometric typologies of submissions in the eVolo competition. However, all images lack detail. Looking at the images generated using this workflow, we can see different typological families: **towers** (see images 3a and 3b in Fig. 3), **twin-tower structures** (3c and 5b in Fig. 3), **bridge-like structures** (4d in Fig. 3), **monolithic structures with wide bases** (2a, 2b, 3d, and 5c in Fig. 3), **sprawl-like structures made out of discrete units** (1d, 4a, 4b, 4c, 5a in Fig. 3), as well as more **complex volumetric structures** (2c, 2d, or 5d in Fig. 3). This overview of typologies is interesting and could be useful for others who want to design a new submission for the competition.

The results from Workflow A and Workflow B are architecturally *relevant* because the hybrid texts resemble sufficiently architectural discourse, while the hybrid images remind the viewer of skyscraper typologies. Seeing these different typologies together helps to gain a unique and broad overview of the conceptual preoccupations, aesthetics and tectonics of prior submissions.

When we started working on *Assembled Growth-Babel*, we wanted to investigate the extent to which text-to-text, text-to-image, and image-to-image machine learning tools could help in the design of a conceptual architectural project. We went on to a design process of collecting a large amount of data (curation), that we fed to a set of machine-learning algorithms (automation) that would result in a large design space we would have to curate once more (re-curation). In this way, our creative work, and agency as designers, would be limited to the creation, curation, and re-curation of the data. As this process did not result in a viable submission, we reconsidered the role of technology in our design process and decided to look at the results from the two workflows as the starting point and inspiration for a proposal, Design Iteration 2, is described below.

4. Design Iteration 2

From Design Iteration 1 we had a starting point consisting of a thorough overview of previous work submitted for the competition. Looking at the hybrid texts, we could see patterns of language representing main themes and concepts in architectural discourse. The first step in generating a proposal, was to create a design brief that would be more specific: i.e., to identify the main functions of the proposed building. Below we detail how we established the design requirements, the title, and the final resulting proposal.

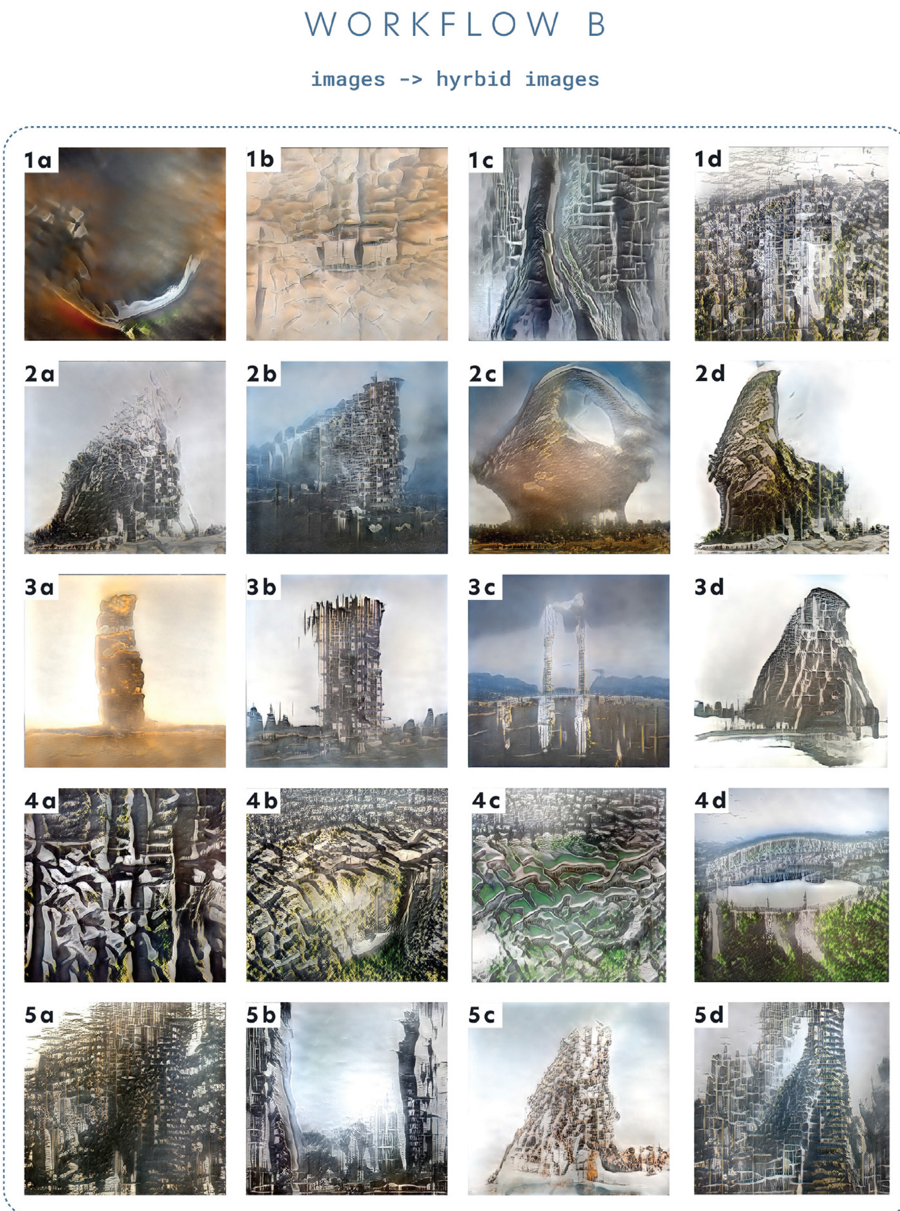

Fig. 3 Part of the design space resulting from Workflow B, where hybrid images were generated based on images of previous submissions to the eVolo skyscraper competition using the StyleGAN2-ADA algorithm.

4.1. Final design brief

In order to establish the brief, we looked at the results from Workflow A and Workflow B, and analyzed them qualitatively.

We noticed that the hybrid texts contained concepts that could be valuable in a submission. In creating prompts to generate the hybrid images in Workflow A, we started a qualitative analysis of the hybrid texts. We continued this analysis in a structured way, in order to define a design brief for the proposal. This analysis was conducted by first printing out all the hybrid texts we had generated. Next, two of the authors spent time becoming familiar with the data individually and each coded a subset of main themes by using an emergent coding approach (Lazar et al., 2010). The emergent codes where then negotiated between the two authors, until a final list of codes was produced. This final list of codes was then used to collaboratively code the whole data set. Afterwards, an iterative process started where the codes were affinity diagrammed (Holtzblatt and Beyer, 1998), first separately and then collaboratively, until a final list of elements to be included in the design brief was produced. The highlighted text in Fig. 2 are the concepts that we identified from that specific hybrid text.

The design brief that emerged from this step was: "Design a self-sufficient high-rise that can produce clean water, energy and food. The building should include diverse functions raging from leisure, commerce, open public spaces, urban farms, commercial greenhouses and residential. The building should have a high degree of customization and modularity making it adaptable to different locations."

4.2. Developing the final design

We then categorized the functional requirements in the design brief into larger clusters and created a concept map. These categories had to do with the following: (1) social functions such as living, working, or education, (2) energy, material, and food production, and (3) structural elements (i.e., vertical or horizontal circulation). Under each of these categories, sat what we called functional units that would be part of the skyscraper design. These programmatic elements are described in detail below.

The social modules (see Fig. 4) included four units: (1) residential unit, (2) educational unit, (3) work pods or innovation units and (4) local markets. The second category of discrete modules, energy, material, and food production, included three units (see Fig. 5): (1) urban gardens, imagined as units for food production, but also for the production of other bio-based fuels; (2) energy production units were imagined as units of energy production taking into account local, site-specific possibilities for harvesting green energy; and (3) cleaning units imagined as units that clear the air and water supply. Finally, the structural modules connect the other functional units and perform structural tasks as follows: (1) assembly and disassembly units were imagined to function similarly to current factories; (2) connection flows for people, energy and products were represented by horizontal and vertical slabs. All these functional units respond to the design brief, and therefore are developed based on Design Iteration 1.

Once we had defined the functions that should make up the proposal, we unpacked the ways in which they should connect to each other in a building design. While Design Iteration 1 was valuable in identifying what elements should go in a building proposal, it was less useful in describing how these elements connect to each other, and therefore, this part was designed by us. Each unit should have flows of people, energy and matter (or materials), coming in, and going out. In this way, it becomes possible to define how the functional units relate to each other in a coherent design proposal. For example, the unit *work or innovation pod* will have input flows of people coming in from the residential unit, and from the educational unit, and it will also need energy to function, therefore, it will receive inputs from energy units. The same unit will also have output flows, of people going to the urban gardens, residential units, and educational units.

After defining the main input and output flows for each functional unit, we used these as rules of assembly: i.e., defining ways in which they connect to other units in the proposal. We imagined that the units could be assembled in different ways, based on site-specific requirements, but did not go into detail in designing one instance for a specific site. As our design brief stated that the building should be self-sufficient and have a high degree of customization, we proposed the building to take different shapes according to different site characteristics that would inform, e.g., the amount and type of energy units.

Practically, we designed each of these programmatic units, and then created the assembly rules based on the input and output flows, using the Monoceros plug-in (Subdigital, 2023) for Rhino's Grasshopper (Rutten, 2023). Monoceros is a plug-in that implements an algorithm called *wave function collapse*, which is a procedural algorithm used for space-filling purposes. It allows the definition of certain spatial relationships between defined modules, or units. Given that Design Iteration 1 gave us a list of functions or programs to be included in the skyscraper proposal, but not how these elements connect to each other, designing different functional units and then creating rules of assembly between them, was a design choice that came naturally from the previous step. The five-step process through which we used *wave function collapse* for procedural content generation included: 1) *Representation*. The content to be generated was represented as a 3D grid where each cell or voxel in this grid represents a possible state or element. 2) *Initialization*. Initially, the grid is populated with a set of possible states or elements. These states in our case represent different unit positions. 3) *Propagation*. In this step, the *wave function collapse* algorithm iteratively examines neighboring cells and tries to propagate information about what elements are allowed or likely to be in those locations based on the rules or constraints we defined. 4) *Collapse*. At some point during the generation process, a cell or pixel may reach a state where only one possibility is valid or highly likely given the constraints and rules. At this stage, the algorithm collapses the superposition of possibilities in that cell into a single outcome. This means the algorithm "chooses" a specific state or element for that cell based on the rules and constraints. 5) *Repeat*. The propagation and collapse steps continue iteratively until the entire grid is filled with specific outcomes, resulting in a fully generated model. In our case, the *wave function collapse* algorithm did 523 solve attempts until it collapsed to the final version of the 3D grid that we implemented in our design. The sizes of each functional unit of the skyscraper were modules of 3 m or multiples of 3 m on each axis. The generated 3D grid was then populated 8 times in height and 2 times for both depth and width, resulting in total a skyscraper of 60 m in length, 54 m in width, and 192 m in height. This multiplication was chosen because the time to compute a 3D grid and all its super-positions of that size would exceed the limitations of our computational power. Figure 6 shows the final rendering of one instance of the skyscraper placed in an urban European setting.

4.3. Title

Apart from using the text dataset to train text-to-text generator as part of Workflow A, we also analyzed it using natural the language processing tools from Voyant Tools (Sinclair and Rockwell, 2016). These tools allow the visualization of texts in different ways and to see patterns that would be difficult to notice otherwise. We did this because the size of the text dataset made it impossible for us to gain an overview of it without computational tools. One of the analyses we conducted in this way, was on the titles of the eVolo skyscraper submissions. Analyzing the texts in the dataset in this way, we noticed a peculiarity: nearly every year between 2006 and 2021, at least one of the winning projects or honorable mentions of the eVolo skyscraper competition had the word Babel in the title. As an

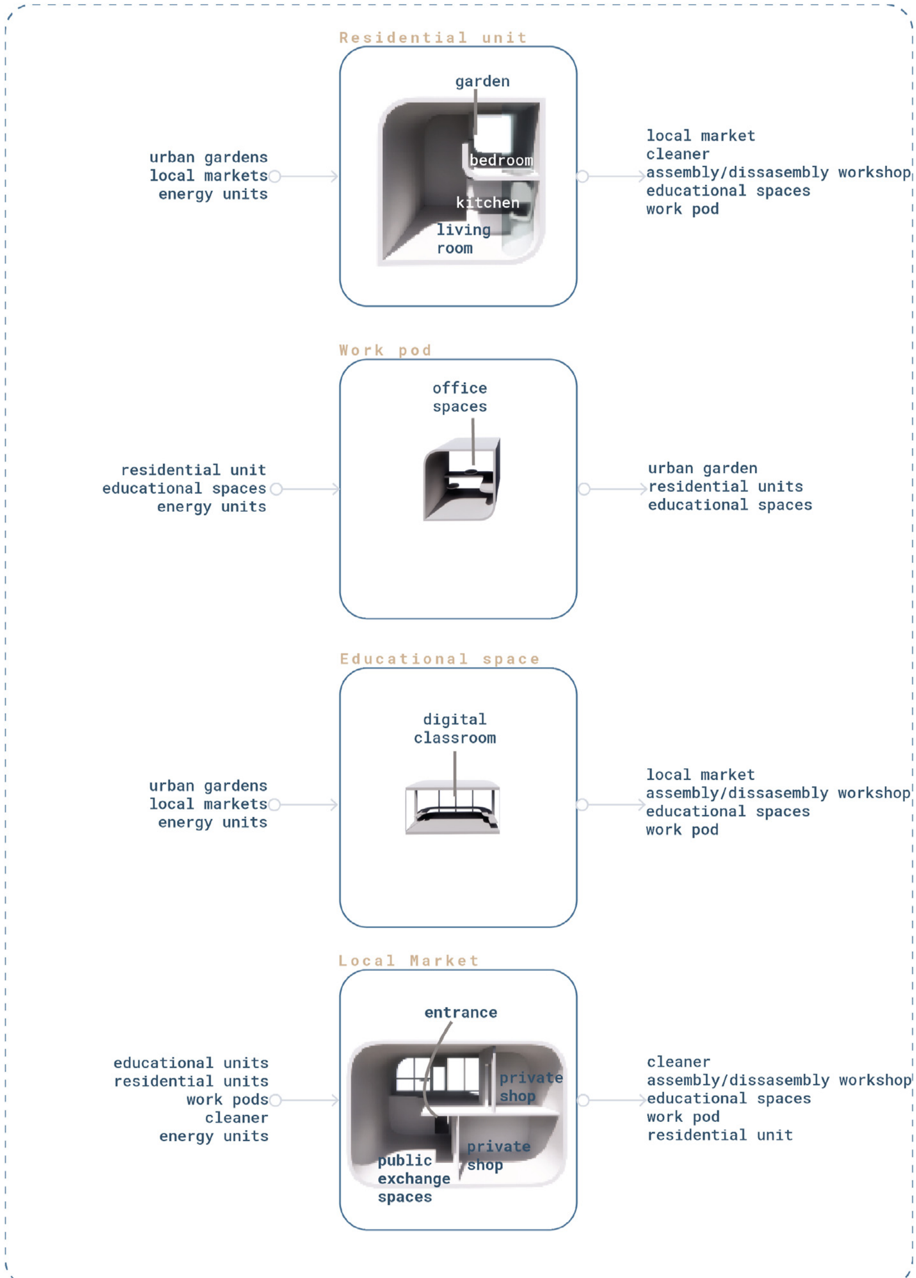

Fig. 4 Social units making up *Assembled Growth-Babel*. On the left, are the inputs in terms of flows, of people, energy, or materials, one unit gets from the other units in the skyscraper. On the right, are the outputs from the unit.

experiment, we decided to use Babel in the title of our submission as well. Given that we imagined this proposal to be a prototype, that would assemble a series of functional units together, according to site-specific constraints, we called the proposal: *Assembled Growth*. This is how we ended up naming our final submission *Assembled Growth-Babel*.

5. Discussion

We begin the discussion with a subsection where we present reflections on our two-step design process and discuss the impact of integrating generative machine learning algorithms in workflows for conceptual design based on our experience. In the following subsection, we continue by discussing language as a medium for conceptual design in architecture. We argue that in the context of emerging machine learning-powered tools (and to some extent of all digital tools), language takes new roles in the production of space.

5.1. Reflections on integrating text-to-text, text-to-image, and image-to-image machine learning tools in a conceptual architectural design process

Steinfeld (2021) discusses how artificial intelligence enters the toolkit of architects, designers, and artists, and proposes that machine learning as tools for creative work can

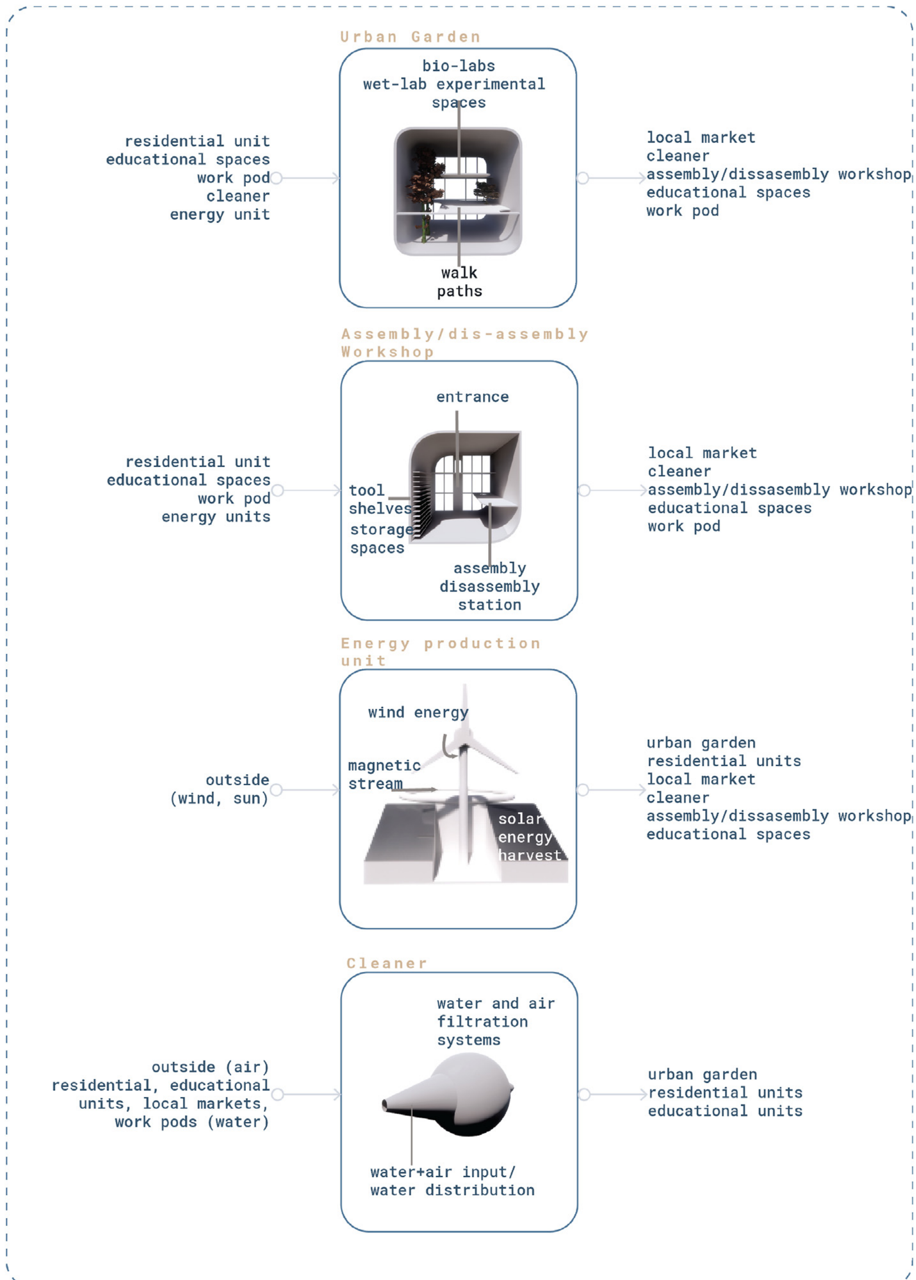

Fig. 5 Energy, material, and production units making up the skyscraper. On the left, are the inputs in terms of flows, of people, energy, or materials, which this unit gets from the other units in the skyscraper. On the right, are the outputs from this unit.

be categorized as either (a) machine learning as actor, (b) machine learning as material, or (c) machine learning as provocateur. While in our work we do not identify the machine learning tools as direct actors for design (as in it is the machine that designs the proposal with the "humans" in the background), the tools were used as both materials (i.e., the hybrid texts served as materials for refining a design brief) and provocateurs (interacting with the dataset we curated sparked new ideas, i.e., the title). Figure 7 shows the design process we employed to create the final proposal.

Designers will start any design process with documentation: learning and looking through other projects that have elements that could be similar to the brief to be addressed. Curating, training and re-curating the datasets were excellent for this initial step, and can be a useful method for others as well. They helped us learn about what has been done previously in a unique way. The results from the two workflows helped us see patterns in the texts and visual representations that we might have missed if we had simply looked through previous submissions to the competition. Specifically, we used the results from training the machine learning algorithms to narrow down the broad design brief of the competition into a more specific brief that was sufficiently detailed to inform a proposal for a conceptual architectural project. In this respect, machine learning was used as a material for design. Additionally, by analyzing the text dataset, we noticed the frequent use of

Fig. 6 Final render of one instance of the skyscraper placed in an urban European setting.

the word *Babel* in the titles of winning projects and honorable mentions: a peculiarity from the point of view of architectural history or theory. This observation provoked the idea of using the word Babel in the title for our submission.

In both of the workflows we interacted directly with the datasets for training the machine-learning algorithms. This was important for us, as we felt we had more control (or agency) over the outputs. Others have touched on this issue as well. For example, according to del Campo (2022), the only way to escape the embedded bias in existing machine-learning tools is for architects to curate their own datasets, making sure that these datasets are relevant to the architectural profession. In designing *Assembled Growth-Babel*, this is what we did: we collected and curated a large dataset of images and texts that were relevant to the project at hand. But while interacting with the data was important and valuable, it was an arduous task, as currently, machine learning algorithms such as those underlying GANs need very large datasets to produce results that get close to representations of architecture. The curation of the initial dataset and the re-curation of the results took away from time we could have used on traditional ideation. Curation and re-curation instead of ideation are more similar to the work of art or architectural historians, and the results of this process, we find, are interesting from this perspective. We, therefore, suggest that using these tools can be useful in teaching or researching historical patterns and attributes in art, design, and architectural history. Llach (2021) proposes that we consider the process of data collection, curation, and the generation of the machine learning assembly rules as *acts of design*. This is of course true for our design process in making *Assembled Growth-Babel*: we considered the entire process to be an act of design. However, it was an act of design that involved curation, automation, and re-curation as well as engaging with text-based programming languages. The initial phases of the design process moved from being a predominantly visual and somatic activity (i.e., sketching out different ideas) mediated by the use of natural language through discussions to an activity of curating datasets, and then using programming languages to train machine-learning algorithms of these datasets. Engaging with this method of work relies predominantly on the use of language.

The two books published in 2022 sharing part of the same name—*Architecture in the Age of Artificial Intelligence*—introduce different views on the impact AI systems will have on architecture. Neil Leach's book (2022) suggests that the profession will not exist in the future, anticipating that AI will bring the "death of the architect", as AI will soon be able to fully design buildings. Phil Bernstein proposes a more nuanced picture, where AI will be used to automate certain tasks from the design process, rather than doing complete jobs. Some tasks that can be automated, should and will be automated, similarly to how digital tools helped to speed up placing dimensions on blueprints. In order to unpack architect's tasks, Bernstein looks at the RIBA (Royal Institute of British Architects) Plan of Work and the AIA (American Institute of Architects) Basic Services, and creates a map of these tasks, showing how some of them are procedural (as in: can be easily defined with a measurable goal and executed through explicit logic), integrative (require an intelligent integration of procedural tasks to reach a goal), or perceptive (inherently creative, subjective and reliant on implicit knowledge). Based on this map, he argues that: "there is very little that today's architects do … that can be characterized as easily automatable" (Bernstein, 2022). This is an opinion that we share based on the experience presented here. In Table 1, we use the map synthesized by Bernstein (2022) to provide reflections on how machine learning was used in our design process and on how others could make use of this methodology. Since we created a submission for a conceptual architecture competition, we didn't go into all the steps from this map, i.e., we did not need to acquire a client or to interact with construction professionals.

Below we summarize our suggestions for directions on future research and practice.

Usefulness of methodology presented. The frameworks introduced here could serve as (a) tools in teaching the history of art, design or architecture, as they gave an excellent overview of what had been done previously. This is why, they could also (b) serve as instruments for early stage documentation for a project if architecturally relevant datasets are available. Using the frameworks we also (c) encountered creative mistakes, that could prove productive for others in conceptual design. Moreover, with the help of the text-to-text generator, we could (d) refine a specific design brief.

Creating datasets. It was important for us to be able to train the models with datasets that we could interact with, however the process was extremely time-consuming, we suggest that whenever possible architecturally relevant datasets be made available in the community in order to help advance research in the field faster.

Copyright and sharing datasets. Due to copyright limitations, we could use the full text dataset we collected, but could not share it fully with the community. We suggest that professional organizations in our field such as RIBA or AIA manage architecturally relevant datasets and access to them. This is preferable to software companies owning the rights as it would ensure transparency and accessibility.

Working with code is significantly different from working through the interfaces of CAD and BIM tools. Unpacking

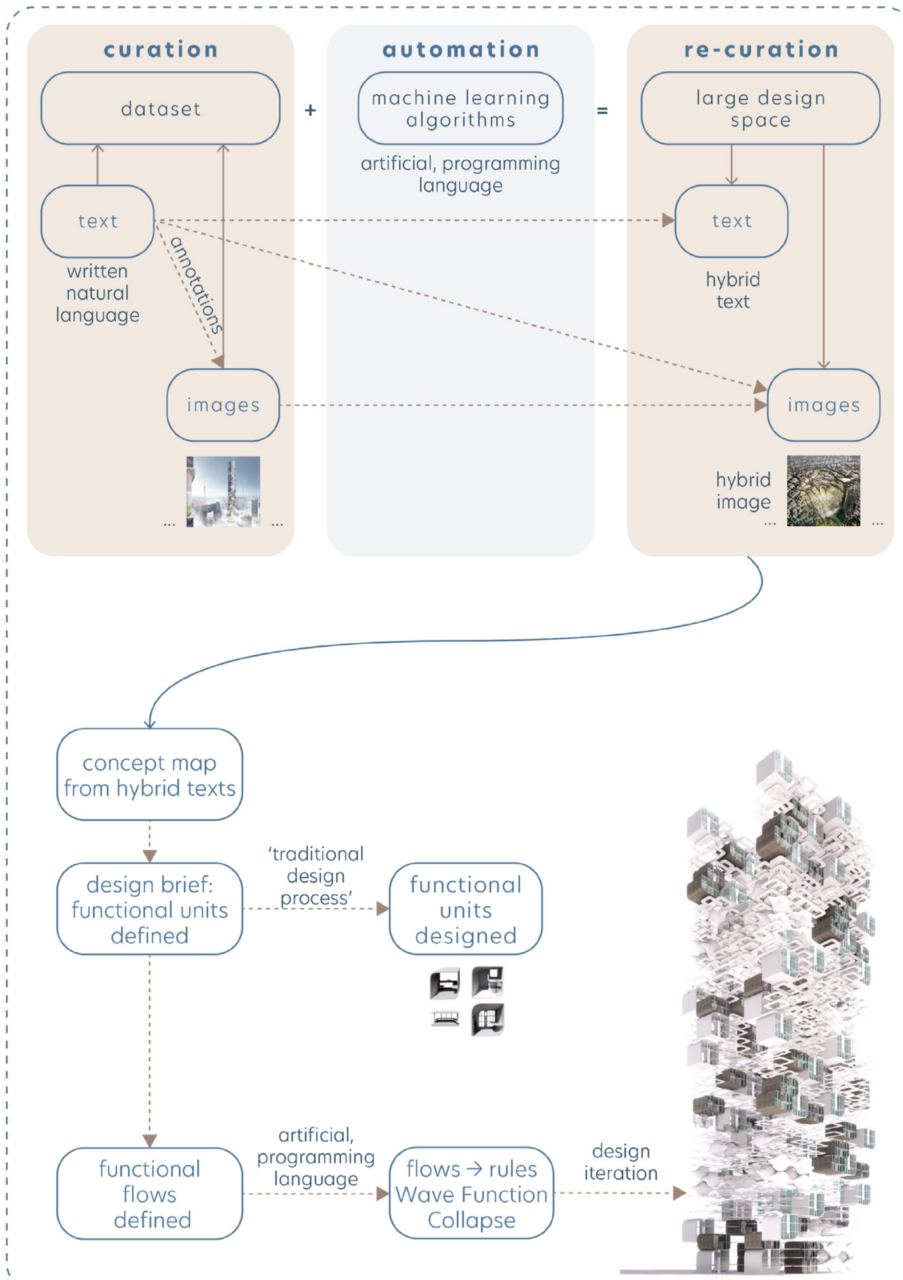

Fig. 7 The final design process we employed for creating *Assembled Growth-Babel*.

what working with programming languages as mediums for architectural design means, and how they frame thinking and possibilities is a direction for future research.

5.2. Architecture, machine learning and language

In recent years, the increased complexity of machine-learning tools, and specifically deep neural networks, meant that the results of these frameworks became hard to understand even for the computer scientists who develop the algorithms. This has led to the emergence of the field of explainable AI (Zhu et al., 2018). In order to make the results of AI systems easier to understand, and to ensure that those using them trust these results, explainable AI research proposes creating different types of visualizations, or verbalizations of the inner workings of neural networks (Halilovic and Lindner, 2023; Zhu et al., 2018). For designers, architects, or artists it is important to unpack how machine learning tools function in order to better understand their results, to frame what can be expected from various workflows, and, importantly, to be able to engage critically with such tools.

Employing machine learning tools, as well as using digital tools in general for architectural design means working, either implicitly or explicitly, with different types of language. The most obvious use of language is in the text prompts used to create visual images in text-to-image generators. Programming languages are employed as well,

Table 1 Design stage where machine learning tools were used in designing *Assembled Growth-Babel*, and implications for future research and practice.

Design stage	Use of and reflections on text-to-text, text-to-image, and image-to-image generators
Analysing previous work	The use of the different machine learning algorithms helped to understand and see patterns in previous work in a way that would have been impossible without their use. We could get an overview of the main themes used in the texts of previously submitted abstracts and see the shape grammars of projects submitted previously. Machine learning tools can provide excellent overviews of previous work and offer valuable help to architects, designers and artists in the initial stage of documentation for a project.
Analysing and understanding brief	The results from training the text-to-text generator were used to inform the final design brief. The hybrid texts contained concepts such as self-sufficiency, sustainability, urban farming, or modularity, which were useful to refine a specific brief for the final implementation.
Generating alternatives	The two workflows generated a vast design space of hybrid texts and two types of hybrid images. As it has been pointed elsewhere, the design spaces that can be generated with these tools far outweigh those that can be generated without them.
Evaluating and selecting alternatives	In our case, we, as designers acted as evaluators in selecting alternatives. As architects bear the responsibility for design decisions that sometimes can have legal or ethical implications, we suggest that the evaluation and selection of design alternatives from a design space created by an AI system should remain with the designer.

as the languages that sit behind digital tools in general. Text-to-image generators also rely on images that have been annotated, as in a piece of text (usually one word) describes what an image represents, and text-to-image generators are typically trained on these datasets. In this subsection, we investigate how these different layers of language come together when using generative machine learning tools, drawing on our experience from designing *Assembled Growth-Babel*. This contributes to making these tools more explainable for creative professionals in general and adds to the discussion on the relationships between language and the production of space (Markus and Cameron 2002).

Architectural discourse, as a natural language helps shape the field and its development. The corpus of texts that we curated and fed to the machine learning algorithms contains two types of texts that constitute natural language: AD includes texts of architectural theory and criticism. Meanwhile, the eVolo abstracts are descriptive texts that help to explain conceptual architectural proposals. Most AD texts are analytical and critical, while eVolo texts are descriptive and in some way prescriptive (they help explain how the projects should be, but also describe the design brief to which the projects respond). The dataset of texts we used to train the algorithms were mostly written by and for architectural professionals and academics. Academic fields, including architecture, have their own way of speaking that helps to create an identity of this academic community that is filtered through discourse (Ghassan, 2019). Ways of speaking are important because they frame ways of thinking. According to Forty (2000), modernist architecture brought new ways of drawing and building but also a distinct critical discourse. Bearn (1992) went as far as claiming that modernist architecture was "more basically, a body of documents defining modernism and interpreting those buildings". There are some recent examples where architectural discourse was studied through the use of corpus linguistics (Beloso, 2015; Cabrera, 2016; Horvath, 2022b, 2022c; Yazici and Durmus Ozturk, 2023), showing that contemporary architecture is surrounded by a distinct way of speaking, a discourse that uses certain words and concepts in specific ways. When architects use text-to-text and text-to-image generative tools, they will use a way of speaking that corresponds to the discourse community of architecture in crafting their text prompts (i.e., a typical prompt would include words often used in the profession, such as *skyscraper*, *organic*, *sustainability*, *urban tissue or modularity*).

Programming languages. The algorithms that were used to generate hybrid texts or hybrid images throughout the design of *Assembled Growth-Babel* consist of technical and artificial language written by a different set of professionals: very rarely architects, and mostly computer scientists. Programming languages sit behind machine learning tools, but also behind all digital tools and can be used implicitly and explicitly throughout the design and construction phases of an architectural project: they sit behind user interfaces that are used for the design of architectural projects. Some architects use programming languages explicitly to generate parts of the design of a building, to measure and evaluate but also to speculate and explore large design spaces. Programming languages might also be used to program digital or robotic fabrication processes that construct part of the building. Therefore, the internal logic of different interacting programming languages feeds into the design possibilities of tools used in the profession. In other words, they frame what is possible in design solutions. The emergence of new tools, and their effect on design thinking and architectural practice has been well documented (Burry, 2011; Carpo, 2012, 2017, 2023; Jabi, 2013; Menges and Ahlquist, 2011; Tedeschi, 2014; Woodbury, 2010). Yet, most of the literature discusses how to embrace these tools, and how to teach them

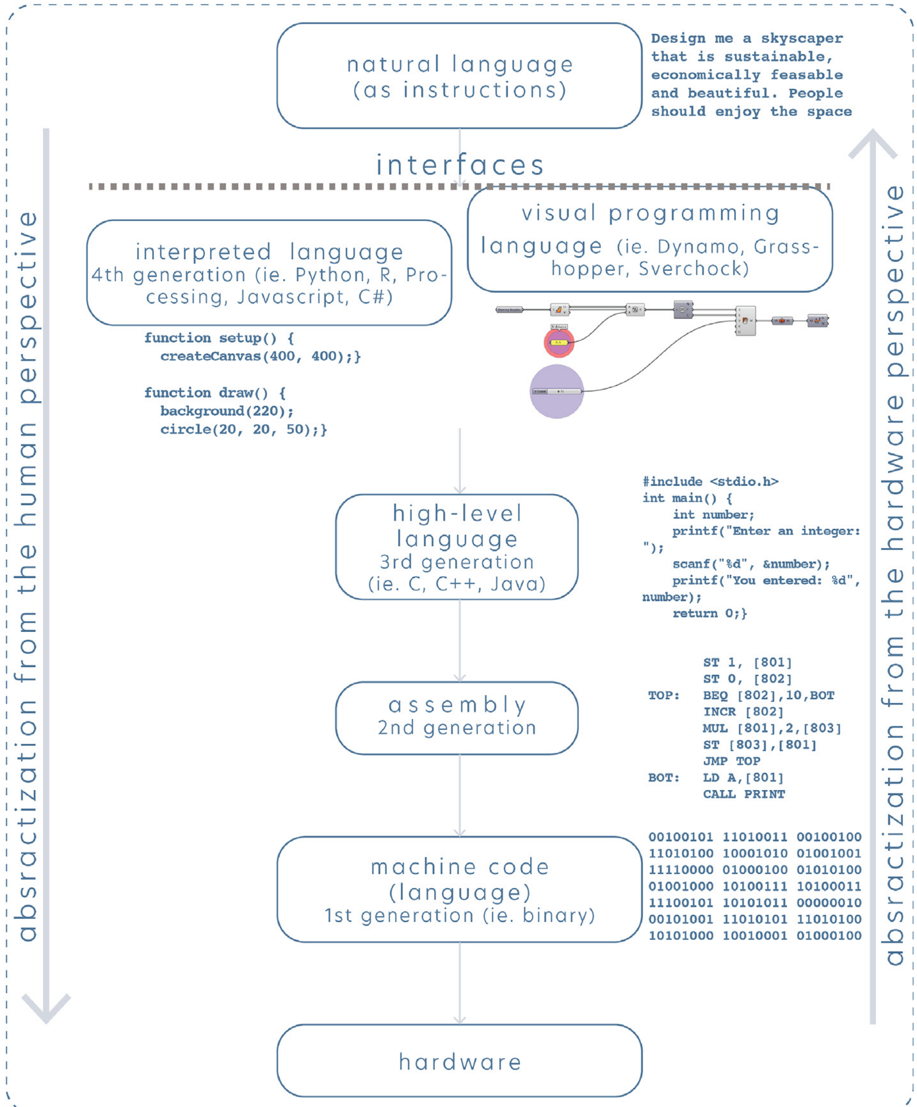

Fig. 8 Between natural language and machine-readable code. From a human perspective, language becomes more abstract as it gets closer to machine code. From a hardware perspective, language becomes more abstract as it gets closer to natural language.

to a new generation of architectural practitioners. It is equally important to consider how these languages shape and frame thinking and by extension creative work in all stages of the design process. In the same way that architectural discourse uses concepts that shape thinking about our field, programming languages, and making use of them, thinking through their logic, frame a specific way of thinking—computational thinking. Computational thinking has been widely discussed both within the discipline (Coates, 2010; Fricker et al., 2008; Menges and Ahlquist, 2011), and outside of it (Caspersen et al., 2007; Caspersen and Nowack, 2013; Winograd and Flores, 1986). Investigating how making use of computational tools impact creative thinking in architecture is a direction of future research.

Therefore, programming languages play an important role in the production of space. They are considered artificial languages, as they have no native speakers and can be categorized according to their level of abstraction which refers to how much interpretation is needed between the language itself and the instructions that hardware can understand. In computer science, it is considered that programming languages with a high level of abstraction are closer to natural language (and so easier for humans to read), while those with a low level of abstraction are closer to machine code. Ultimately, the only code that hardware can read is a sequence of 0 s and 1 s, and so the purpose of programming languages is a complex process of translating natural language into a series of 0 s and 1 s. The digital turns in architecture (Carpo, 2012, 2017) are made possible by two converging trends. On the one hand, there is a growing interest in programming from the field of architecture, and other creative fields. On the other hand, computational power has grown, and partly because of this, interpreted programming languages that are closer to natural language, are available. Therefore, as more people become interested in programming, a new generation of programming languages are more accessible and easy to read, from a human perspective: closer to natural language (see Fig. 8). Ultimately programming as a practice is a process of *communication*, *representation* and *mediation* where ideas and concepts from natural language are translated in successive steps into a mathematical language and later into binary code. This idea is in no way new and has been discussed elsewhere, for example (Winograd and Flores, 1986), however, understanding programming as a communication practice involving successive steps of mediation and abstractization for architecture has been less discussed in our field. Seeing programming languages in this way can be illuminating for architects, designers and artists who want to work with machine-learning powered tools, but it is also useful when theorizing the digital and post-digital in these fields in general.

Annotations. Finally, the text-to-image generators, such as VQGAN + clip, used in Workflow A, or those that sit behind Dall-e or Midjourney, the two tools behind the recent explosion of architects' interest in machine learning tools (Steinfeld, 2023), have at their core large datasets of annotated images. These annotations are themselves texts that describe, using one word, what an image represents. Figure 9 shows how natural language is broken into categories (as nouns), categorized under synsets (sets of synonyms), and then later used to annotate images. For example, in ImageNet, the concept *furniture* can be found under *entity* → *instrumentality* → *furniture*. It will appear as a synset with the concepts *piece of furniture*, and *article of furniture* (Deng et al., 2009a, 2009b; Yang et al., 2020). Large image datasets and their annotations have been widely criticized: while some images can be easily described using one noun, this is not always the case (Crawford, 2021). Moreover, some nouns such as apple are more "nouny" (and easier to represent using an image), as opposed to others, such as "health" (Crawford, 2021). A different annotation system will create different results. Ideally, annotations and large image datasets designed specifically for architecture should be created in future research, and be made available in a transparent way. Regardless of the notation system though, these annotations have the ultimate effect of flattening the complexity of natural written language on the one hand, and of images depicting reality on the other hand.

After presenting separately the different types of language that came together in designing a project such as *Assembled Growth-Babel*, we propose placing them together into a single map (Fig. 10) that unpacks how language was used in every stage of our design process. As mentioned earlier, seeing these layers and their interactions can help architects, designers and artists seeking to employ machine learning tools to have a better understanding of these frameworks, what to expect in terms of the outputs they can produce, and where to keep a critical eye. Architectural discourse, as natural language is filtered into short text-prompts. These text prompts are then queried and the nouns in the prompts are connected to images that have been annotated as depicting the noun. This process of mediation is realized through artificial languages that translate natural languages as instructions into machine-readable and ultimately binary code. The popular text-to-image generators make use of large annotated datasets of images. These annotated datasets

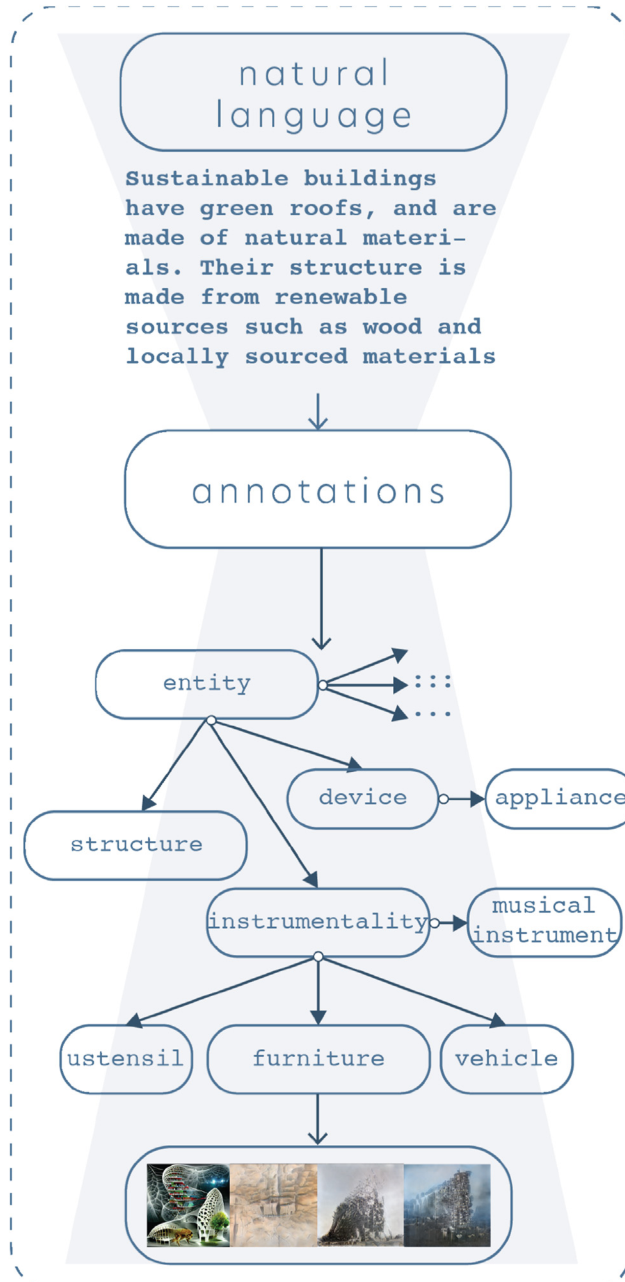

Fig. 9 Annotations in text-to-image generators: an example of ImageNet, one of the datasets we used in Workflow A (as part of VQGAN + clip), and currently the largest in the world.

describe, using a piece of text what the image represents. In this way, reduced language prompts describe visual representations of reality. Throughout the process of designing with such tools for architectural design, visual feedback is pushed to later (i.e., text is written before, compiled, and the visual result is seen at the end). This comes in contrast to drawing—as the predominant tool for early-stage design, where the visual feedback is immediate.

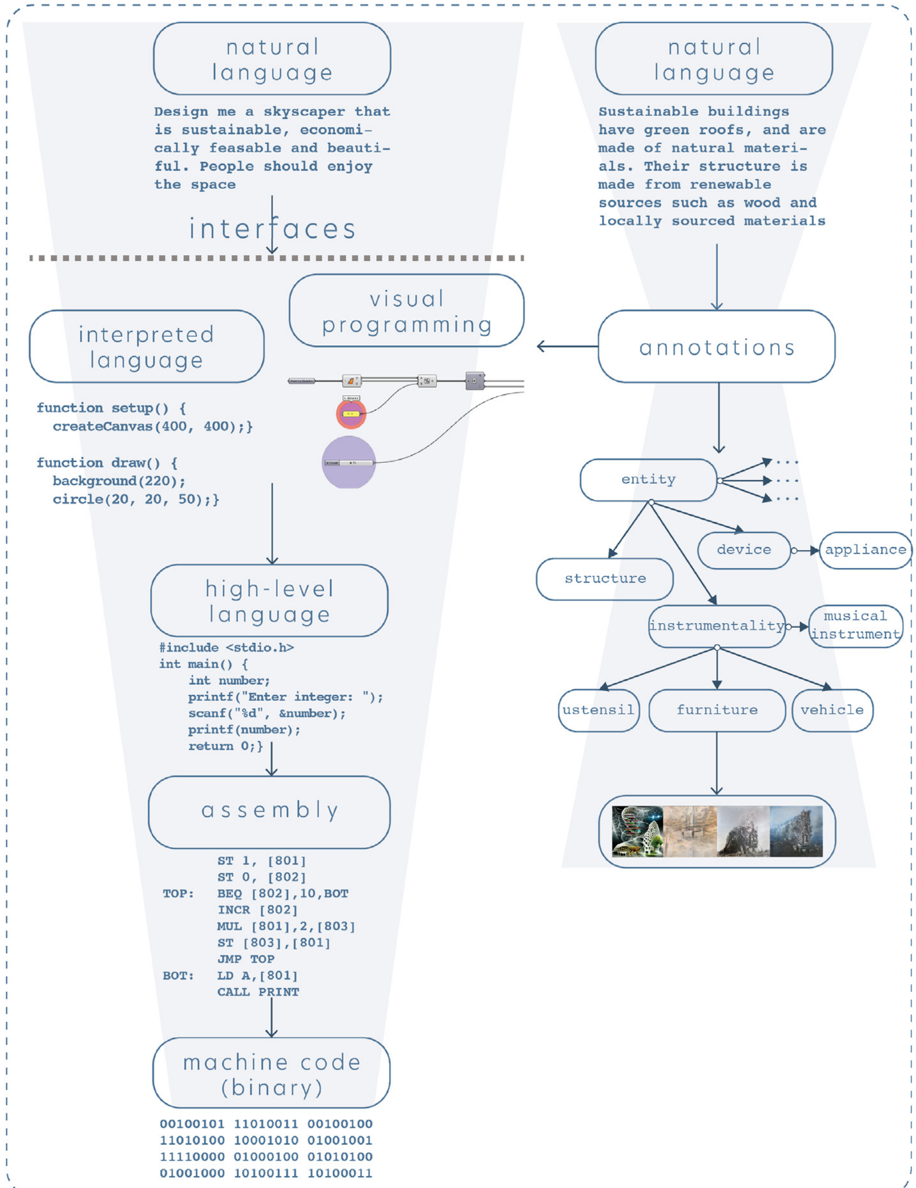

Fig. 10 Layers of language unpacked. Left: Programming languages transform natural language into computer-readable instructions, that are transformed into binary code. Right: Text-to-image generators are pre-trained on large datasets of annotated images. The annotation is a text snippet that described what the image represents. In this way, different layers of language interact in the use of machine-learning-powered tools by a complex process of mediation, and communication.

6. Conclusion

In this paper, we presented a design methodology where we used text-to-text, text-to-image, and image-to-image machine learning tools to generate a submission for the eVolo skyscraper competition. We trained these algorithms on a dataset that we curated containing texts and images. The texts came from two sources: the journal Architectural Design, and the abstracts that describe the winning projects and honorable mentions of the eVolo skyscraper competition. The images represent the posters for the winning projects and honorable mentions for the eVolo skyscraper competition. This methodology is a first contribution of the paper, and can be useful for architectural practice.

The second contribution is the artifact itself together with our reflections on the methodology and its outcomes. The results from training the text-to-text generator were used to refine a specific design brief for the final concept. The images that resulted from training the image-to-image generator gave an overview of shape grammars or form typologies of previously submitted projects to the eVolo skyscraper competition. Some of the images created using the text-to-image generator were surprising, and they can contribute to creative work, by sparking new ideas or conversations. In this way, current generative machine learning tools can help in the ideation process for early-stage architectural design, and they are especially useful tools in gaining intriguing insights from the perspective of art or architectural history, as they help to uncover patterns that would otherwise be difficult to identify. While interacting with the datasets of these tools was important for us as designers, the process of (curation-)automation-recuration needed when employing these tools introduces a tedious step in early-stage design, therefore architecturally relevant datasets should be made available to the community whenever possible in order to help advance research in the field.

Finally, we noticed that by introducing these tools in an architectural design process we made use of language in different ways, compared to when we would design without them. We reflect on this use of language as we experience it, and propose a map showing how three different layers of language interacted in our design process. The map can help in making AI tools more explainable for architects, designers, and artists who wish to employ them in their workflows, and contributes to theorizing about how digital and computational tools frame creative work in general and architectural design in particular.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This research was partly funded by the Human-Centered AI cluster at the Department of Communication and Psychology at Aalborg University in Denmark. Johanna Lauritzen and Matthias Klages have contributed to data collection and processing for this work in its initial stages. Additionally, we would like to thank the editor and the reviewers for their valuable feedback which helped refine and clarify many aspects of the manuscript.

References

Alaçam, S., Karadag, I., Güzelci, O.Z., 2022. Reciprocal Style and Information Transfer between Historical Istanbul Pervititch Maps and Satellite Views Using Machine Learning, 11. Revista de la Facultad de Arquitectura y Urbanismo de la Universidad de Cuenca, pp. 71—81.
Barker, N., 2023a. Hickok Cole Uses Chatgpt to Design 24-storey Mixed-Use Building.
Barker, N., 2023b. Zha Developing ”most” Projects Using Ai-Generated Images Says Patrik Schumacher.
Bearn, G.C.F., 1992. The formal syntax of modernism: carnap and le corbusier. Br. J. Aesthet. 32, 227—241.
Beloso, B.S., 2015. Designing, describing and compiling a corpus of English for architecture. Procedia Soc. Behav. Sci. 198, 459—464.
Bernstein, P., 2022. Machine Learning: Architecture in the Age of Artificial Intelligence. RIBA Publishing.
Bolojan, D., 2022. Creative ai: augmenting design potency. Architectural Design 92, 22—27.
Burry, M., 2011. Scripting Cultures: Architectural Design and Programming. John Wiley amd Sons.
Cabrera, T., 2016. Interpreting architecture: the archint corpus. Tradumàtica: tecnologies de la traducció 156. https://doi.org/10.5565/rev/tradumatica.145.
Carpo, M., 2012. The Digital Turn in Architecture 1992-2012. Wiley.
Carpo, M., 2017. The Second Digital Turn: Design beyond Intelligence. MIT Press.
Carpo, M., 2023. Beyond Digital: Design and Automation at the End of Modernity. MIT Press.
Carta, S., 2021. Self-organizing floor plans. Harvard Data Science Review 3.
Caspersen, M.E., Nowack, P., 2013. Computational thinking and practice: a generic approach to computing in Danish high schools. In: Proceedings of the Fifteenth Australasian Computing Education Conference - Volume 136. Australian Computer Society, Inc., AUS, pp. 137—143.
Caspersen, M.E., Larsen, K.D., Bennedsen, J., 2007. Mental models and programming aptitude. In: Proceedings of the 12th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education. Association for Computing Machinery, New York, NY, USA, pp. 206—210.
Chaillou, S., 2019. Archigan: a Generative Stack for Apartment Building Design. NVIDIA Corporation.
Chaillou, S., 2020. ArchiGAN: Artificial Intelligence X Architecture. Springer Nature Singapore, Singapore, pp. 117—127.
Chaillou, S., 2022. Artificial Intelligence in Architecture: from Research to Practice. Birkhäuser.
Coates, P., 2010. Programming Architecture. Routledge.
CompVis, group Lmu Munich, R.S.A., 2023.
Crawford, K., 2021. Atlas of AI: Power, Politics and the Planetary Costs of Artificial Intelligence. Yale University Press.
Crowson, K., Biderman, S., Kornis, D., Stander, D., Hallahan, E., Castricato, L., Raff, E., 2022. Vqgan-clip: open domain image generation and editing with natural language guidance.
del Campo, M., 2022. Neural Architecture: Design and Artificial Intelligence. Applied Research and Design Publishing, Singapore.
del Campo, M., Carlson, A., Manninger, S., 2021. Towards hallucinating machines—designing with computational vision. Int. J. Architect. Comput. 19, 88—103.
Delua, J., 2021. Supervised vs. unsupervised learning: what's the difference?.
Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009a. ImageNet: A Large-Scale Hierarchical Image Database. CVPR09.
Deng, J., Li, K., Do, M., Su, H., Fei-Fei, L., 2009b. Construction and Analysis of a Large Scale Image Ontology. Vision Sciences Society.
Domingos, P., 2015a. The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World. Basic Books, New York.
Domingos, P., 2015b. The Five Tribes of Machine Learning (And what You Can Learn from Each) video.
eVolo, S.C., 2022. Registration — 2023 Skyscraper Competition.
Fellbaum, C., 1998. WordNet: an Electronic Lexical Database. Bradford Books, Bradford.
Flow, T., 2019. Text Generation with an rnn.
Forty, A., 2000. Words and Buildings: A Vocabulary of Modern Architecture. Thames & Hudson, London, UK.
Fricker, P., Wartmann, C., Hovestadt, L., 2008. Processing: Programming Instead of Drawing.
Ghassan, A., 2019. Designerly ways of speaking: investigating how the design tribe of researchers speak on design thinking. In: ProQuest Does Not Claim Copyright in the Individual Underlying Works. Guattari, Felix, pp. 1930—1992.
Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. Adv. Neural Inf. Process. Syst. 27.
Güzelci, O.Z., 2022. A machine learning-based model to predict the cap geometry of anatolian seljuk kümbets. Periodica Polytechnica Architecture 53, 207—219.

Halilovic, A., Lindner, F., 2023. Visuo-textual Explanations of a Robot's Navigational Choices, pp. 531–535.
Holtzblatt, K., Beyer, H., 1998. Contextual Design: Defining Customer-Centered Systems. Morgan Kaufmann, San Diego, CA.
Horvath, A.S., 2022a. Compara: a corpus linguistics dataset of computation in architecture. Mendeley Data. https://doi.org/10.17632/7ktscvmxvg.5.
Horvath, A.S., 2022b. Compara: a corpus linguistics in English of computation in architecture dataset. Data Brief 42, 108169. https://doi.org/10.1016/j.dib.2022.108169.
Horvath, A.S., 2022c. How we talk(ed) about it: ways of speaking about computational architecture. Int. J. Architect. Comput. 20, 150–175. https://doi.org/10.1177/14780771211070006.
Jabi, W., 2013. Parametric Design for Architecture. Laurence King Publishing.
Karadag, I., 2023a. Machine learning for conservation of architectural heritage. Open House Int. 48, 23–37. https://doi.org/10.1108/OHI-05-2022-0124.
Karadag, I., 2023b. Transforming sketches into realistic images: leveraging machine learning and image processing for enhanced architectural visualization. Sakarya University Journal of Science 27, 1209–1216. https://doi.org/10.16984/saufenbilder.1319166.
Karadag, I., Güzelci, O.Z., Alaçam, S., 2023. Edu-ai: a twofold machine learning model to support classroom layout generation. Construct. Innovat. 23, 898–914.
Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., Aila, T., 2020. Training Generative Adversarial Networks with Limited Data, 06676 arXiv:2006.
Kelly, T., Guerrero, P., Steed, A., Wonka, P., Mitra, N.J., 2018. Frankengan: guided detail synthesis for building mass models using style-synchonized gans. ACM Trans. Graph. 37 (1), 1–1:14.
Lazar, J., Feng, J.H., Hochheiser, H., 2010. Research Methods in Human- Computer Interaction. Wiley Publishing, West Sussex, UK.
Leach, N., 2022. Architecture in the Age of Artificial Intelligence: an Introduction to AI for Architects. Bloomsbury.
Lin, T.Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C.L., Dollár, P., 2015. Microsoft coco: common objects in context. arXiv:1405.0312.
Llach, D.C., 2021. The routledge companion of artificial intelligence in architecture. In: Routledge. Chapter Sculpting spaces of possibility Brief history and prospects of artificial intelligence in design, 1st ed, pp. 13–28.
Markus, T.A., Cameron, D., 2002. The Words between the Spaces: Buildings and Language. Routledge.
Menges, A., Ahlquist, S., 2011. Computational Design Thinking. Wiley, AD Reader.
Midjourney, 2024.
Milošević, J., Dukanović, L., Zivkovic, M., Žujović, M., Gavrilović, M., 2023. Automated Compositions: Artificial Intelligence Aided Conceptual Design Explorations in Architecture.
Mostafavi, F., Tahsildoost, M., Zomorodian, Z.S., Shahrestani, S.S., 2023. An interactive assessment framework for residential space layouts using pix2pix predictive model at the early-stage building design. Smart and Sustainable Built Environment.
Nauata, N., Chang, K.H., Cheng, C.Y., Mori, G., Furukawa, Y., 2020. House-gan: relational generative adversarial networks for graph-constrained house layout generation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (Eds.), Computer Vision — ECCV 2020. Springer International Publishing, Cham, pp. 162–177.
Nicholas, P., Rossi, G., Williams, E., Bennett, M., Schork, T., 2020. Integrating real-time multi-resolution scanning and machine learning for conformal robotic 3d printing in architecture. Int. J. Architect. Comput. 18, 371–384.
OpenAI, 2023.
Pouliou, P., Horvath, A.S., Palamas, G., 2023. Speculative hybrids: investigating the generation of conceptual architectural forms through the use of 3d generative adversarial networks. Int. J. Architect. Comput. 21, 315–336.
Princeton, U., 2010. About Wordnet.
Rodrigues, R.C., Duarte, R.B., 2022. Generating floor plans with deep learning: a cross-validation assessment over different dataset sizes. Int. J. Architect. Comput. 20, 630–644.
Rutten, D., 2023. Grasshopper 3d.
Sanjay, C., Preslav, N., Ahmed, A., Wendy, H., Issa, K., Xiaosong, M., Husrev, T.S., Ingmar, W., Michael, W., Ting, Y., 2023. Ten Years after Imagenet: a 360°perspective on Artificial Intelligence. Royal Society of Open Sciences.
Sinclair, S., Rockwell, G., 2016. Voyant Tools.
Steinfeld, K., 2021. Significant others: machine learning as actor, material, and provocateur in art and design. In: As, I., Basu, P. (Eds.), The Routledge Companion to Artificial Intelligence in Architecture. Routledge, pp. 3–12.
Steinfeld, K., 2022. Imaging Place Using Generative Adversarial Networks (GAN Loci). John Wiley & Sons, Ltd, pp. 513–516.
Steinfeld, K., 2023. Clever little tricks: a socio-technical history of text-to-image generative models. Int. J. Architect. Comput. 21, 211–241.
Stigsen, M.B., Moisi, A., Rasoulzadeh, S., Schinegger, K., Rutzinger, S., 2023. Ai diffusion as design vocabulary investigating the use of ai image generation in early architectural design and education. In: Dokonal, W., Hirschberg, U., Wurzer, G. (Eds.), Digital Design Reconsidered - Proceedings of the 41st Conference on Education and Research in Computer Aided Architectural Design in Europe (eCAADe 2023), pp. 587–596.
Subdigital, 2023. Monoceros.
Sun, C., Zhou, Y., Han, Y., 2022. Automatic generation of architecture facade for historical urban renovation using generative adversarial network. Build. Environ. 212, 108781.
Tamke, M., Nicholas, P., Zwierzycki, M., 2018. Machine learning for architectural design: practices and infrastructure. Int. J. Architect. Comput. 16, 123–143.
Tarabishy, S., Psarras, S., Kosicki, M., Tsigkari, M., 2020. Deep learning surrogate models for spatial and visual connectivity. Int. J. Architect. Comput. 18, 53–66.
Tedeschi, A., 2014. Algoriths-aided Design. Le Penseur, Paris, Freance.
Tong, H., Türel, A., Şenkal, H., Yagci Ergun, S.F., Güzelci, O.Z., Alaçam, S., 2023. Can ai function as a new mode of sketching: a teaching experiment with freshman. International Journal of Emerging Technologies in Learning (iJET) 18, 234–248.
Turing, A.M., 1950. I.—computing machinery and intelligence. Mind LIX 433–460.
Winograd, T., Flores, F., 1986. Understanding Computers and Cognition: A New Foundation for Design. Ablex Corporation, New York.
Woodbury, R., 2010. Elements of Parametric Design. Routledge.
Xu, C., Liu, J., Wu, Z., Chen, Y.F., 2022. Automated steel reinforcement detailing in reinforced concrete frames using evolutionary optimization and artificial potential field. Autom. ConStruct. 138, 104224.
Yang, K., Qinami, K., Fei-Fei, L., Deng, J., Russakovsky, O., 2020. Towards fairer datasets: filtering and balancing the distribution of the people subtree in the imagenet hierarchy. In: Conference on Fairness, Accountability, and Transparency.
Yazici, M., Durmus Ozturk, S., 2023. An analysis of rem koolhaas's discourses on architecture and urban design using a corpus-based model. Frontiers of Architectural Research 12, 222–241.
Zhang, L., Zheng, L., Chen, Y., Huang, L., Zhou, S., 2022. Cgan-assisted renovation of the styles and features of street

facades: a case study of the Wuyi Area in Fujian, China. Sustainability 14.

Zhong, X., Pia, F., Yu, F., Tan, C., Pan, Y., 2022. A discussion on an urban layout workflow utilizing generative adversarial network (gan): with a focus on automatized labeling and dataset acquisition. In: Co-creating the Future: Inclusion in and through Design, Proceedings of the 40th Conference on Education and Research in Computer Aided Architectural Design in Europe.

Zhu, J., Liapis, A., Risi, S., Bidarra, R., Youngblood, G.M., 2018. Explainable Ai for Designers: A Human-Centered Perspective on Mixed-Initiative Co-creation. IEEE Conference on Computational Intelligence and Games (CIG), pp. 1—8.