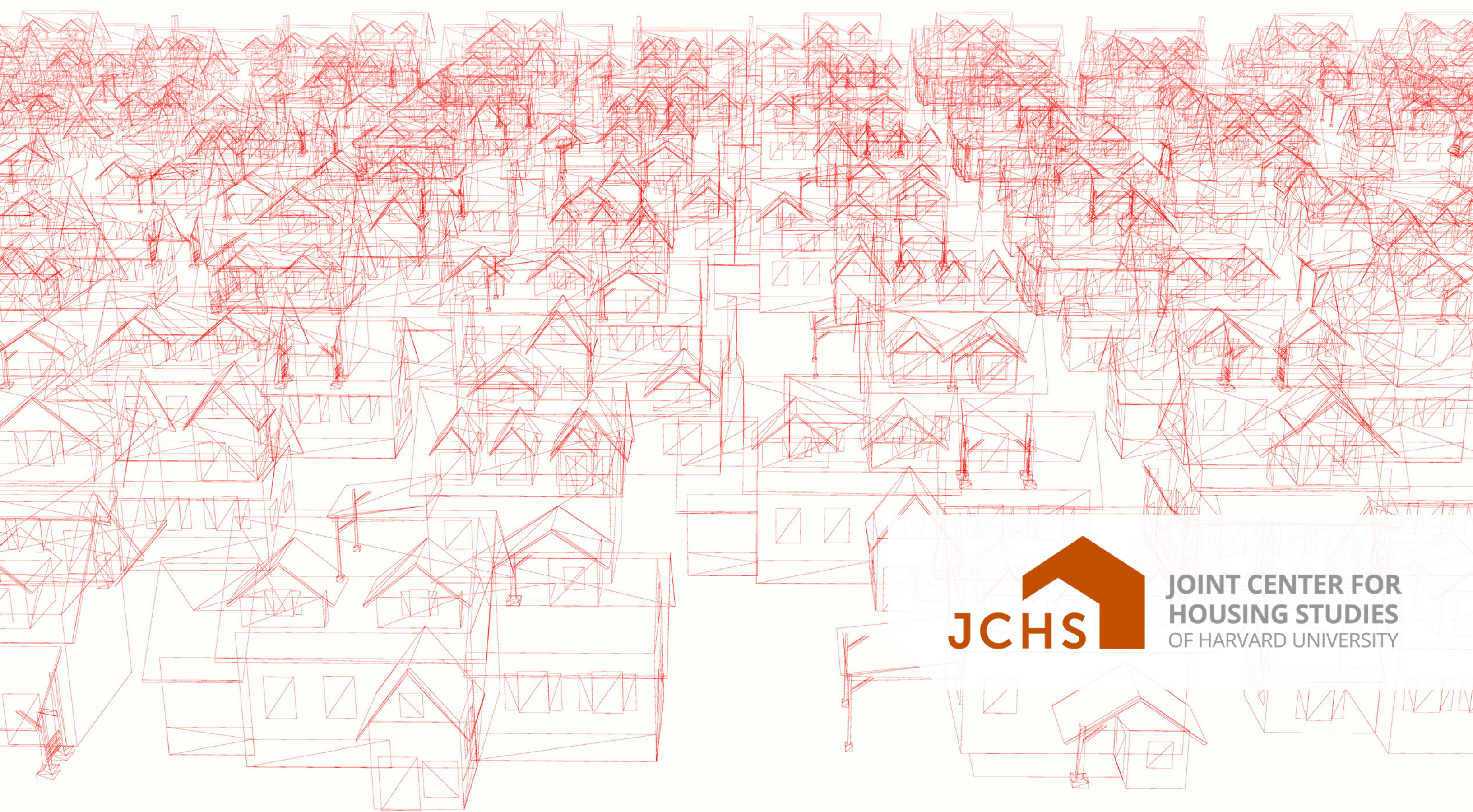
BRINGING DIGITALIZATION HOME SYMPOSIUM

Artificial Intelligence for All

Perspectives and Outlooks on the Role of Machine Learning in Architectural Design

NOVEMBER 2022 | JOSE L. GARCÍA DEL CASTILLO Y LÓPEZ



JCHS
JOINT CENTER FOR HOUSING STUDIES OF HARVARD UNIVERSITY

Joint Center for Housing Studies
Harvard University

Artificial Intelligence for All: Perspectives and Outlooks on the Role of Machine Learning in Architectural Design

Jose L. García del Castillo y López (Harvard Graduate School of Design)

November 2022

This paper was presented as part of "Panel 1: How Is Digitalization Changing How Housing Is Designed & Built?" at the symposium "Bringing Digitalization Home: How Can Technology Address Housing Challenges?", hosted by the Harvard Joint Center for Housing Studies in March 2022 and funded by Qualcomm. Participants examined the changes that digitalization—the use of automated digital technologies to collect, process, analyze, distribute, use, and sell information—is spurring in the way housing is produced, marketed, sold, financed, managed, and lived in.

Abstract

Computational tools have been crucial in the advance of architectural design and production. Since the early 1960s, developments in Computer-Aided Design, Manufacturing (CAD/CAM) and Building Information Modeling (BIM) have had a fundamental impact in how the profession evolved. Computation helped improve fluency in the creative stages of architectural conceptualization, increase productivity and reliability in design and construction, democratize access to architectural production, and foster a wealth of formal and technical innovation in architecture. Recent groundbreaking developments in the field of Artificial Intelligence (AI) and Machine Learning (ML) are bearing the promise of a new revolution in computing, led by novel algorithms capable of learning from experience, rather than rules. Neural networks are now capable of predicting shopping preferences, musical recommendations, and health diagnostics at a speed and success rate that has greatly surpassed that of any specialized human. Just like every other industry has been fundamentally transformed by the power of computation, the field of architectural design and production is experiencing disruptive changes through the power of data.

It is naïve to believe that the field of architecture will not be affected by these developments; the question is rather in which ways they will reshape the profession and its outcomes. How can data be exploited in design processes? What will the role of the architect be within highly automated design environments? Can AI make architectural design more accessible to the end user? In this essay, an overview on the current role of AI and ML in architectural design is presented. The conversation is situated through a historical overview of the role of computation in architecture, followed by a mapping of the current state of the art in machine learning applications in architectural design. Informed by previous and current trends, an outlook is postulated about what changes to expect in the design discipline and how to adapt to them.

Introduction

The development in the Renaissance of notational systems for the representation of architecture—the *lineamenta*—constituted a major shift from the traditional, ad hoc means that master masons employed for building construction, and is regarded as the origin of the modern split between *designing* architecture and *building* it (Carpo 2011). Plans, elevations, sections, and perspectives became the new language of architecture, the vehicle for its allographic materialization and, most importantly, the main substrate for its conceptualization. Drawing two-dimensional sketches and blueprints is, still today, the most common thinking aid in the early stages of the architectural design process; they constitute an abstract mental model suitable for spatial conceptualization, and have a direct correlation with the documentation needed to produce a building. Architectural drawings are the instrument to both *think* and *make* architecture.

The advent of computers had a profound and everlasting impact on all aspects of human endeavors. The unleashing of the power of computation offered obvious advantages in all fields that required intense number processing, such as census, statistics, and military applications. Consequently, it also marked the beginning of deeper interrogations on the benefits that access to—potentially infinite—computing power could bring to our very own mental processes. Early pioneers in human-machine interaction speculated on the capacity of computers to aid scientists in storing and indexing the knowledge of a world of exponentially growing complexity, allowing them to reach farther and deeper into this web of interconnected intelligence (Bush 1945). This notion became the foundation for visions of a future with true human-computer symbiosis, where humans could outsource menial tasks to computer companions, and focus on creativity and decision-making, a task humans are arguably better suited for (Licklider 1960). There was a general consensus that creative thinking would never be replaced by computers but, instead, the use of computers as "clerks" would result in a global *human augmentation in intellectual capabilities* without precedents in history (Engelbart 1962).

Unsurprisingly, the field of architecture promptly joined this emerging conversation, and turned to inquiry about the potential role of computation in assisting design and production processes. The automotive and aerospace industries pioneered this journey, with an initial emphasis on the development of computer graphic frameworks and computational models for the accurate representation of the complex surfaces typically manufactured for their products (Cardoso 2015). In a research partnership between International Business Machines (IBM) and General Motors (GM) in the early 1960s, researchers soon identified that "drawings, pictures, and models were the principal media for communication and documentation of design ideas," and asked themselves "how could computational techniques significantly impact the design process?" (Krull 1994). The outcome of this venture was the DAC-1, possibly the earliest example of an integrated computational environment for design. The system was able to scan physical drawings, approximate their form using polynomial models, represent them in three dimensions on a display screen, and allow a certain degree of interaction with them, a huge feat at the time. Interestingly, the acronym stood for *Design Augmented by Computer*, in clear reflection of the new human-machine zeitgeist.

Parallel explorations of machine-aided design processes were also being conducted in academia, characterized by perhaps more experimental and speculative propositions. The most notorious example was the development of Sketchpad in 1963, a computational system for vector drawing on a stylus-enabled display (Sutherland 1963). Sketchpad offered a revolutionary touch-and-draw interaction model, a paradigm that would itself need more than 20 years to become a standard through the vehicle of computer mice—also an invention of the epoch (Engelbart 1968). However, Sketchpad's most groundbreaking feature was the capacity of the user to select elements of the drawing, and request the system to adjust them based on custom geometrical constraints: horizontality, orthogonality, parallelism, etc. Such a contribution was remarkable not just in computer-assisted drawing, but for computation in general, as it fundamentally allowed users to define not just the data itself—the drawing—but relational properties defining a sense of structure and dependency between them (Davis 2013). Moreover, these constraints were amongst the earliest form of computer programming that was based on the user defining *what* they wanted, rather than *how* to achieve such result (Victor 2013).

Unfortunately, Sketchpad's vision of "a man-machine graphical communication system" remained fairly symbolic. Computation became smaller, faster, and cheaper, leading to the commercial availability of personal computers, and the development of multiple examples of Computer-Aided Design (CAD) software (Weisberg 2008). Nevertheless, none of the packages that became commercially successful featured the novel models that Sketchpad spearheaded for design thinking augmentation. Instead, "the CAD software industry was dominated by systems seeking to 'simply' automate traditional drafting procedures" (Cardoso 2013), losing a valuable opportunity to take advantage of the computer's special capabilities (Nelson 1990).

History of AI

Such explorations of the role of computers as creative partners were representative of a larger revolution in computer science, one which focused on interrogating the very nature of intelligence itself. The development of automated computing machinery meant an increase in the computation power of simple calculations several orders of magnitude bigger than ever possible before. A more fundamental question immediately arose: at which point does the aggregation of increasingly larger amounts of these basic operations start resembling the complexity of human thought? Or perhaps, inversely, how should these calculations be structured to achieve such a goal? And ultimately, if successful, how would we know?

Some of the first attempts to answer these questions came in the form of mathematical models to describe the human brain. These proposals involved formal logic theories to model nervous activity (McCulloch and Pitts 1943) and the perception of audiovisual signals (Pitts and McCulloch 1947). The premise was fairly simple: if the behavior of the nervous system could be described as a set of logical operations, then replicating these operations with the new computing machines might lead to machines that could *learn* and *think*. However, such intuitive human concepts had not yet found a formal definition in computer science terms, as the difference between a simple memory register and the capacity to make predictions based on data never seen before did not seem to be a binary one. At which point does *memory* become *experience* that can be enacted into *creative* thinking? This question became the core of higher-level inquiries on the nature of machine thought, and for which the pioneering proposition—and the one that is still today the gold standard of Artificial Intelligence (AI) tests—was to abandon possible quantitative methods, and shift validation assessments back to qualitative perception. Put simply, if a human is unable to discriminate between a human and a machine interlocutor, the machine shall be regarded as capable of thinking akin to a human being (Turing 1950).

Developing machine thought by mimicry of the human brain inspired yet another powerful novel intuition: that the processes that led to machine thinking may not be *programmed*, but should instead be *learned*. This idea conformed the basis for the earliest forms of Machine Learning (ML), such as reinforcement learning systems based on success-based rewards (Samuel 1959), or the formulation of the perceptron, a machine theoretically capable of learning anything that it could be programmed to do (Rosenblatt 1961). Such an idea was also the seed of the more modern dialectic between traditional programming—developing algorithms by encoding *rules*—and programming by example—creating algorithms that can learn from *data*. Interestingly, the visionary notion from the 1950s that computers might be capable of learning might have arisen because computer programming at that time was still not fully developed as a field of study (Minsky and Papert 1988).

Research in AI and ML stalled after the 1970s. The reasons were varied: demonstrated limitations of early ML models such as the perceptron (Minsky and Papert 1969), reduced computing capacity, overhype, and other socio-political factors. Interest and funding declined into the first so-called

"AI winter," followed by a brief period of renewed enthusiasm in the 1980s, and a second stretch of pessimism and cutbacks in the field (Hendler 2008). It wasn't until the late 1990s and the 2000s that interest in AI started to bloom again. Advances in parallel computing, and widespread availability of massive amounts of data, bore the promise of democratizing theoretical innovations that had been quietly developed during the cold nights of multiple AI winters. The most notable progress occurred in the field of deep learning, including the popularization of neural network architectures and techniques for unsupervised learning (LeCunn et al. 2015), resulting in the modern boom in AI.

Presently, the field has become fairly accomplished at using examples to train algorithms in detecting patterns in datasets, a notion we denote as making machines *learn*. However, for the most part, the main applications of these techniques are typically reduced to predictive functions such as object recognition, information pre-screening, product suggestion and other similar prosaic automation tasks. While not lacking merit, these models still do not fully satisfy the spirit of the original quest for machine intelligence. Producing new data from previous data is a good starting point, albeit one that does not yet fully represent one of the most characteristic traits of human intelligence: generating something out of nothing or, in other words, *creative* thought. In ML terms, a neural network could be trained to replicate learned patterns on previously unseen data, a notion typically referred to as inference. However, could this learning be used to generate brand new information—from no input data—exhibiting those same patterns?

The search for an answer led to the most recent revolution in the field of AI: the development of generative models. The cornerstone of this new movement was the invention of Generative Adversarial Networks (GANs), a ML architecture that is composed of two neural networks: a first one to randomly generate samples of real vs. made-up information, and a second one to discriminate between authentic and fake results (Goodfellow et al. 2014). The genius in this proposition is that, by opposing them against each other using a reward system, the neural networks can train each other autonomously to become better at, for instance, identifying fake images of cats and generating realistic-looking ones respectively. In this scenario, the generator is considered trained once it surpasses the capacity of the discriminator to discern what is real and what is not, and can subsequently be used as an endless source of artificial, synthetic data without the need of any further inputs as a source. The concept proved groundbreaking, sparking an explosion of GAN-inspired generative neural network architectures to generate, for example, photo-realistic images of human faces (Karras et al. 2019), celebrities (Karras et al. 2018), modern art paintings (Elgammal et al. 2017), bedrooms (Radford et al. 2016) or three-dimensional chairs (Kleineberg et al. 2020), or the extension of this architecture to work with inputs, such as image to image translations (Isola et al. 2017) or style transfers (Zhu et al. 2017). Additionally, the generative capacity of GANs would in turn become the perfect affordance to reexamine the potential role of computers as creative partners.

AI Research in Architecture

As earlier discussed, modern CAD systems carry the legacy of a modeling paradigm that evolved during the 1980s, with the advent of personal computers and the development of commercial software packages, and that remains today: the digital replication of the analog patterns found in traditional drafting tables, without harnessing the enhanced features that a computational environment could afford. The reasons for this literal translation might be multiple: the lack of computer literacy amongst early computer adopters, the commercial pressure to make a product successful for the general user, the need to cater to a labor force trained in analog methods, etc. Nonetheless, the many advantages brought by the digitization of drafting made CAD systems extremely successful, swiftly replacing analog drawing to the extent that virtually every modern architectural project is developed using digital means.

The digitization of drafting had an indisputable impact in accelerating and optimizing the production of architectural projects. Yet, a sort of divide remains between *making* architecture and *designing* it. Generations of designers still use analog methods, such as drawing with pencil and paper, as a means to think through a design; the immediacy, fluidity, flexibility and freedom afforded by the medium is thus far challenging to mimic with CAD systems (McCullough 1996). Such duality has led to the perception that there is still a strong divide between the processes typically involved in the early stages of architectural design, a moment where ideas are flowing, changes are rapid and iteration is key—the *creative* stage—and the requirements for the later stages of the development of such concepts, where ideas and visions are fixed into a detailed set of construction documents and building specifications suitable to be enacted into a building by a third party—the *production* phase. The comprehensiveness required at this latter stage results in an elevated degree of friction to changes, making it challenging to work at these two levels simultaneously. However, the production phase is also arguably the one requiring the least creative effort and, hence, the part of the architectural process that might be more susceptible to automation. These boundaries might be blurring though, as younger generations of digital-native architects are increasingly fluent in using computational tools at every stage of the design process, and CAD platforms are improving their capacity to seamlessly integrate early conceptual drawings with advanced modeling and informational tools. CAD is progressively becoming the de facto, integral tool to think and design architecture with.

The main historical substrate at the core of these design and production processes has been the architectural floorplan. Drawings are the main means to represent and build architecture and, arguably, the floorplan is the most relevant, as it encapsulates the essential features of the spatial and experiential character of a building. Unsurprisingly, floorplans as a medium were also the first entry point to AI/ML-assisted architectural design. Some of the earliest attempts at automated generation of floorplans relied on relational representations of the functional units of a program, and used random heuristics to find solutions within the constraint space (Negroponte 1970). However, modern ML floorplan generation typically generates semantic partitioning, a form of estimation of which areas of a floorplan would be destined to particular uses, and then applies it to the basic layout of a floorplan from an input building footprint (Peters 2016). This idea has been extended to train neural networks to learn translations between the different levels of detail in the stages of the design of a particular building: from land plot boundary to building footprint, from footprint to room partitions with semantics, and from partitions to fully furnished layout (Chaillou 2019). The fundamental contribution of this model is the conceptual discretization of the architectural design process into atomic operations, narrow and focused enough to be undertaken by current neural networks. Subsequently, these individual parts can be sequenced into an aggregate workflow encompassing the complexity of generating a full-fledged building plan from an empty plot of land.

Floorplan generation seems to be a growing area of interest in the research of AI applications in architecture, from a multitude of different perspectives. The generation of detailed building plans from graphical information continues to be under deep scrutiny, complemented by ongoing research questioning how this can be achieved in more direct connection to the mental processes that we use to conceptualize such designs. While work has been done in the generation of textual representations from images of floorplans (Goyal et al. 2021), the inverse operation, neural networks that generate building layouts from literal descriptions, may arguably have the greatest potential impact in architectural design. Progress has been done in the development of neural networks that generate simple images from text (Xu et al. 2018) yet, due to the inherently creative nature of this process, a significant amount of human interpretation still needs to be contributed to elevate the outcomes to the category of buildable object (Del Campo 2021). Interestingly, a form of representation that is proving quite successful in encoding the underlying logics of a floorplan is graphs. Graphs represent interconnected nodes with embedded properties, a data structure that fits particularly well the diagrammatic sets of relations between architectural spaces conceptualized during the design stages. Additionally, graphs provide a fairly direct translation into the adjacencies of the different rooms in a building plan, including the quality of their connections. Promising research has been conducted in the creation of algorithms that extract graph representations from architectural floorplans (Lu et al. 2021), the necessary step for the creation of datasets to train neural networks that can learn to generate estimations of floorplan distributions from these schematic descriptions (Nauata et al. 2020).

Diagrammatic floorplans are the most fundamental form of architectural representation, and they often serve as the baseline for the development of more comprehensive informational representations of buildings, such as Building Information Models (Deamer and Bernstein 2010). However, a challenge that AI-based design workflows currently face is that most modern neural networks are developed to work with raster images as inputs and outputs, as opposed to the vector drawings typically used in CAD modeling. The reason might be quite simple: bitmaps are a far more common medium, in comparison to the relatively niche use of vector information worldwide. Rasterizing vector drawings is a fairly straightforward operation, as it simply requires discretizing structured geometrical data into a grid of colored pixels. Unfortunately, the inverse operation is actually rather complex to perform, as it requires the reconstruction of structure based on perceived patterns, a process intrinsically creative that is difficult to achieve with rule-based algorithms, but for which ML is proving successful (Zeng et al. 2019). Moreover, the geometrical features found in common floorplans—linearity, continuity, perpendicularity, doors and windows—constitute a significant aid to the learning processes in ML algorithms.

The topic of modes of formal representation in 2D for ML models is one that is not trivial; using ML to represent 2D models constitutes an exponential jump in complexity when addressed for 3D geometry. Like vector drawings, this application remains too niche of a use case for mainstream research. The problem becomes more complicated when considering the wide variety of available formats for representing three-dimensional form: NURBS geometry, triangulated meshes, point clouds, and voxel fields are all common standards in industry, with translations between them often not being straightforward. Current research in 3D object classification tends to use voxel representations of geometry (Wu et al. 2015), partially because the architecture of these neural networks inherits heavily from the convolutional logics commonly used in 2D image recognition. Sadly, voxel models are probably the least popular form of representation in architecture, although promising research is being developed that extends voxel logics to approximations of signed distance functions (Kleineberg et al. 2020), or that work directly with mesh representations at its core (Hanocka et al. 2019); the field is evolving very rapidly on this front.

Perhaps one of the most interesting aspects of 3D-oriented research in AI has not been on the actual 3D models, but the very act of modeling itself. Previous research in the 2D space has demonstrated that many of the patterns that are present in the way we draw, can actively be learnt and replicated by neural networks based on a sequential approach to their creation (Ha and Eck 2018). For instance, if we are asked to draw a cat's face, most humans tend to follow a similar pattern: a circle representing the head, two ovals for the eyes, two triangles as ears and a few arcs for the whiskers. Surprisingly, it has been demonstrated that there is a strong likelihood that the strokes we create will be drawn in that particular order, and following a similar structure. The overall consistency of the human brain abstracting reality into 2D drawings has been subject to inquiry using ML, with revealing results when ML is applied to the creation of sketches from common objects (Ha and Eck 2018) or in engineering drawings for mechanical parts (Willis et al. 2021a). Furthermore, the majority of commercial CAD packages offer a similar step-wise paradigm for 3D creation: models start from primitives which are increased in detail through procedural operations such as constructive solid geometry. The sequential nature of this process is a particularly good fit for recurrent architectures in neural networks, with successful results for example in ML-generated mechanical parts (Willis et al. 2021b). Additionally, some modern CAD environments feature frameworks to express these operations explicitly, allowing the user to develop parametric models using graph-like representations. Such models have also been explored as the basis for neural networks which can learn patterns in sequences within the graph, are able to suggest probable next operations and, therefore, could constitute a real-time aid in modeling processes, augmenting human creative capacities and supporting design exploration (Toulkeridou 2019).

The notion of style in AI is an affordance that seems particularly compelling for architectural design. One of the most fundamental concepts to address when working with ML is bias, typically defined as the tendency of a neural network to replicate the patterns that it finds on the dataset it was trained on. The core idea is that, if a dataset is not properly balanced, the network may tend to favor—or disfavor—certain results over others. This scenario is significantly worrisome if the AI's decisions will have a negative impact on the future of human beings, and it is central to any modern ethical considerations on the use of AI. However, it can be argued that from a design perspective, a bias towards replicating patterns might indeed be a valuable asset, as it could be explored to surface the commonalities between data points coming from the same origin, hence revealing an implicit sense of style. Some of the earliest image-to-image GANs have featured images of central-European building façades in their training datasets, successfully using them to replicate their architectural style (Isola et al. 2017), whereas contemporary artists have taken advantage of this characteristic to train neural networks to learn the style of iconic architects, and generate dream-like animations navigating through their imaginary creations (Anadol 2019). Similarly, style is also a concept that has relevance in the perception of urban environments, as cities have traditionally been characterized by their distinctive image and atmosphere. ML can be used to capture this *genius loci*, superimpose it in digital immersive 3D environments and use it as a design tool to replicate the spirit of characteristic locations (Steinfeld 2019), as well as use city style translation to replicate the livability of pedestrian downtowns and use this as a tool for policy making in urban design (Kim et al. 2022).

The multiplicity of ways AI is being harnessed to approach different aspects of the architectural process is a testimony to its versatility and adaptability. But perhaps, most importantly, it epitomizes what could be its most important affordance in design: its suggestive capacity. Even if ML can be used to aid in small parts of the puzzle, architectural design remains a challenge too broad, multi-faceted and complex for a technology that, at this time, can only tackle much narrower problems. Currently, the holistic generation of architectural designs by AI agents seems relatively distant. However, the generative capacity of neural networks is proving a tremendous asset as an aid in design processes, in its capacity to assist humans with recommendations, ideas, alternatives and options. In turn, these suggestions expose designers to new, and often surprising ideas, provide inspiration, foster discovery and help make thinking processes more broad, iterative, and fluid. The suggestive power of neural networks as concurrent creative companions may have the potential to fulfill the original quest for true human-computer symbiosis (Martínez Alonso 2017).

One last aspect in which AI has the opportunity to disrupt architectural design is in its capacity to open up computer programming for non-programmers. The field of architecture during the last decade has witnessed an increasing number of architects learning how to code, using this skill to extend commercial CAD software or develop their own, and growing into a strong community of computational designers benefiting from their reciprocal contributions (Davis and Peters 2013). However, the paradigm that sets AI apart from traditional computer programming is the possibility of creating algorithms whose functionality has not been defined by encoding a set of rules, but rather extracted from a dataset of examples. As technology evolves and democratizes, working with neural networks is becoming largely accessible to more general audiences and, currently, many frameworks compete to provide means to train ML models without the need to write a single line of code (Lengyel 2021). If no programming is required but, instead, it is enough to simply gather samples, this opens the door to curation as a form of programming by example (Lieberman 2000) and, ultimately, a form of design (Martínez Alonso 2017). Such an idea also highlights the relevance of datasets as a modern form of information contribution. ML is largely possible nowadays thanks to the abundance of data available in the information era. Still, the effort of collecting, structuring, and annotating a dataset is quite remarkable, and should be considered akin to other forms of traditional knowledge. Governments are already highlighting the strategic importance of making public data further accessible for AI research (White House 2016) and, within design, an increasing number of open dataset contributions are accelerating research and innovation in the field (Kalervo et al. 2019; Koch et al. 2019; Lu et al. 2021; Willis et al. 2021b).

Outlook

AI has experienced a tremendous development in the last decade, thanks to groundbreaking advances in ML algorithms, drastic hardware improvements in computation power, and global access to data, resulting in a mass democratization of the techniques necessary to develop novel AI frameworks, and boosting a global explosion of research contributions in the field. And just like personal computers took computation from large corporations and research labs into the hands of the general public, we might be at the dawn of a similar shift in AI accessibility: from specialized, technical audiences to universal users. Such transition is already evident in many tech-heavy, consumer-oriented applications, such as home automation, online retail, or autonomous vehicles. Yet, as with many other technological innovations, the architecture industry has been slow in harnessing the power of AI tools in the design and production of the built environment. Undoubtedly, architecture is a much more niche area that consumer goods. But the open-ended, multi-constrained nature of the problems that are typically involved in design processes is also a much worse fit for the current affordances of ML workflows, adding structural friction to the adoption of AI tools in the field. Nevertheless, current progress in AI development shows evidence that, very soon, we will see gradual changes in the way architecture is created.

It seems unlikely that complete, detailed generation of architectural designs by fully autonomous AI agents will be a reality any time soon. As previously discussed, the complex project that is an architectural design—with the need to satisfy its comprehensive set of functional and regulatory requirements—still poses a big challenge to the narrow problems that modern ML algorithms are able to solve. However, this multifaceted nature of the architectural production process allows for its conceptualization as a collection of more discrete tasks, each with the potential of being assisted by ML tools. We have seen how the partial generation of building floorplans has been the first entry point to undertake the AI-powered architectural design agenda. Similarly, many other predictive applications will be developed to assist decision-making in the early stages of the design process, such as estimations of construction time, costs, energy consumption or building usage. Additionally, AI will soon play a fundamental role in the automation of the costly and time-consuming tasks typically involved in the later stages of architectural production: creation of building drawings, automated design of electrical and mechanical diagrams, generation of project documentation, etc. Human supervision through all these steps will still be required for the foreseeable future, but we will gradually shift from active designers to the role of coordinators and orchestrators of the increasing number of AI agents that will become part of this process. Interestingly, such activities are fundamentally not that different from the central role that architects typically play as mediators between the different professionals that are involved in large construction projects.

As a consequence, the production of architecture will become faster and cheaper. The speed introduced by AI agents that can aid creative thinking and automate technical production will result in faster iteration cycles and shorter delivery times, with reduced final costs likely reflecting this acceleration. This may not have a drastic impact in the production of singular architecture—office, civic or government buildings—but AI agents could become more significant in the housing market, as the ubiquitous demand for residential construction may make speed and cost particularly attractive for the competitive advantage they may provide. Moreover, housing may soon prove to be the best candidate for the earliest forms of complete and fully autonomous AI-generated architecture. The relative simplicity of the program for typical residential units or, in other words, their reduced design space, may help conform the necessary constraints to make the problem narrow enough to be fully accessible for the first successful forms of comprehensive architectural AIs.

The preponderance of commercial CAD software packages as the core means for modern architectural production suggests that AI-powered tools will predominantly be integrated as part of their platforms. Initial implementations will likely seek to employ neural networks to automate common repetitive functions from the standard set of commands available in the software, in the form of soft background assistants providing highlights or autocomplete-like suggestions. Progressively, as technologies develop and users habituate to AI partners, companies will compete to provide new, extended functionality based on data gathered from their own customer base and made possible by ML. Distribution of these tools might initially be embedded with the core software itself. However, as we have seen in other similar scenarios, companies will likely provide APIs and frameworks for growing communities of superusers to distribute—and monetize—their own contributions, similar to app stores or marketplaces, thus furthering the widespread availability of these tools.

With AI tools becoming pervasively ingrained in CAD software, a deep reevaluation will be required of the standard paradigms used for digital drawing and modeling. Traditional, deterministic inputs such as keyboard and mouse, or interaction patterns such as point, click or drag, will need to be complemented with more nuanced forms of human-computer interaction that harness the novel affordances in neural networks. Working with interpretations of voice and text inputs, hand-drawn sketches, facial expressions or hand gesturing will become standard, as literal or implicit descriptions of desired outcomes will increasingly develop into the new means to their graphical generation. Just as new modes of interaction will be developed, user interfaces and data structures will need to adapt to the management of forms of communication that represent design intent, rather than sequences of geometrical operations that lead to a formal outcome. Design will supersede the conversational model of iterative evaluation of drawings by their author, and evolve instead into a dialog between the AI as content generator and the designer as the discriminator of its work: generative adversarial human design augmented by AI partners.

A growing culture of human-AI collaboration in architecture may ignite a renewed interest in computational methods for design. The last decade has witnessed a burgeoning community of architects and designers becoming superusers by learning computer programming and developing software tools, resulting in newly created computational design departments in large firms, new dedicated programs in academia and newly dedicated conferences to share their contributions. However, the AI revolution may have an even deeper impact in unraveling the untapped creative potential of an even larger community of end users by harnessing one of the greatest affordances of neural networks: their capacity to be trained without writing a computer code. As ML models are capable of learning from a collection of sample data, and as more tools make it easier and accessible to train these models without requiring computer science skills, a new form of programming is then possible by simply creating or gathering a sufficient number of samples into a personal dataset, a task that is arguably much more accessible to the average designer.

Computer programming may fall out of fashion in favor of data curation as a form of encoding higher-level goals such as form, aesthetics, patterns, and style. Data is the new currency. Investing effort in creating a bespoke dataset for a neural network to manifest a particular expression will become a new form of developing reusable tools. Purposely allowing telemetry in your software to track your actions will turn into a standard form of training neural networks to anticipate your next move and mimic your own style. Data and trained models will be understood and guarded as a proprietary form of intellectual property.

Our modern reliance on computers for virtually every aspect of how we live and change our world has meant that Artificial Intelligence has and will continue to disrupt every field susceptible to being controlled by computation, with architectural design being no exception. Machine Learning bears the promise of a wealth of new functionality and automation powered by faster and smarter algorithms. Moreover, the predictive and suggestive capacities of AI agents as creative companions may result in a global augmentation of our intellectual capacities. In the future, AI agents as synthetic partners in design workflows may offer a renewed opportunity to reshape the way we program, understand, and interact with computers as design tools, and reflect on the very nature of our own creative thinking.

References

Anadol, Refik [@refikanadol]. "Really enjoying latest GAN training of Frank Gehry's body of work. This hand picked 125k image dataset looks like enough to create imaginative perspectives!" Twitter, 16 March 2019. twitter.com/refikanadol/status/1106798493299949568

Bush, Vannevar. "As We May Think." The Atlantic Monthly Jul. 1945: 101-108.

Cardoso Llach, Daniel. "Algorithmic Tectonics: How Cold War Era Research Shaped Our Imagination of Design." Architectural Design 83.2 (2013): 16-21.

Cardoso Llach, Daniel. Builders of the Vision: Software and the Imagination of Design. Routledge, 2015.

Carpo, Mario. The Alphabet and the Algorithm. MIT Press, 2011.

Chaillou, Stanislas. AI + Architecture: Towards a New Approach. Harvard University, M.Arch. I Thesis, 2019.

Davis, Daniel. Modelled on Software Engineering: Flexible Parametric Models in the Practice of Architecture. RMIT University, Ph.D. Dissertation, 2013.

Davis, Daniel, and Brady Peters. "Design Ecosystems: Customising the Architectural Design Environment with Software Plug-ins." Architectural Design 83.2 (2013): 124-131.

Deamer, Peggy, and Phillip Bernstein. Building (in) the Future: Recasting Labor in Architecture. Princeton Architectural Press, 2010.

Del Campo, Matias. "Architecture, Language and AI." Proceedings of the 26th International Conference of the Association for Computer-Aided Architectural Design Research in Asia, 2021. Volume 1, 211-220.

Elgammal, Ahmed, Bingchen Liu, Mohamed Elhoseiny, and Marian Mazzone. "CAN: Creative Adversarial Networks Generating "Art" by Learning About Styles and Deviating from Style Norms." 8th International Conference on Computational Creativity, ICCC. 2017.

Engelbart, Douglas C. Augmenting Human Intellect: A Conceptual Framework, Summary Report AFOSR-3223 under contract AF 49(638)-1024, SRI Project 3578 for Air Force Office of Scientific Research. Menlo Park, CA: Stanford Research Institute, 1962.

Engelbart, Douglas C., and William K. English. "A Research Center for Augmenting Human Intellect." AFIPS Conference Proceedings, Vol. 33, Fall Joint Computer Conference, San Francisco (1968): 395-410.

Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative Adversarial Nets." Advances in Neural Information Processing Systems 27 (2014).

Goyal, Shreya, Chiranjoy Chattopadhyay, and Gaurav Bhatnagar. "Knowledge-driven Description Synthesis for Floor Plan Interpretation." International Journal on Document Analysis and Recognition 24.1 (2021): 19-32.

Ha, David, and Douglas Eck. "A Neural Representation of Sketch Drawings." International Conference on Learning Representations. 2018.

Hanocka, Rana, Amir Hertz, Noa Fish, Raja Giryes, Shachar Fleishman, and Daniel Cohen-Or. "MeshCNN: A Network with an Edge." ACM Transactions on Graphics (TOG) 38.4 (2019): 1-12.

Hendler, James. "Avoiding another AI winter." IEEE Intelligent Systems 23.02 (2008): 2-4.
Isola, Phillip, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. "Image-to-image Translation with Conditional Adversarial Networks." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: 1125-1134.

Kalervo, Ahti, Juha Ylioinas, Markus Häikiö, Antti Karhu, and Juho Kannala. "Cubicasa5k: A Dataset and an Improved Multi-task Model for Floorplan Image Analysis." Scandinavian Conference on Image Analysis. Springer, 2019.

Karras, Tero, Samuli Laine, and Timo Aila. "A Style-Based Generator Architecture for Generative Adversarial Networks." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.

Karras, Tero, Timo Aila, Samuli Laine, and Jaakko Lehtinen. "Progressive Growing of GANs for Improved Quality, Stability, and Variation." International Conference on Learning Representations. 2018.

Kim, Dongyun, George Guida and Jose L. García del Castillo y López. "PlaceMakingAI: Participatory Urban Design with Generative Adversarial Networks." Proceedings of the 27th CAADRIA Conference on Computer-Aided Architectural Design Research in Asia. 2022.

Kleineberg, Marian, Matthias Fey, and Frank Weichert. "Adversarial Generation of Continuous Implicit Shape Representations." Proceedings of the Eurographics Conference. 2020.

Koch, Sebastian, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. "ABC: A Big CAD Model Dataset for Geometric Deep Learning." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.

Krull, Fred N. "The Origin of Computer Graphics within." IEEE Annals of the History of Computing 16.3 (1994): 40-56.

LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep Learning." Nature 521.7553 (2015): 436-444.

Lengyel, Olivia. "We Shape Our Tools and Thereafter Our Tools Shape Us: Interview with RunwayML Founder Cristóbal Valenzuela." PaperspaceBlog, 2021.

Licklider, J. C. R. "Man-Computer Symbiosis." Human Factors in Electronics, IRE Transactions on HFE-1.1 (1960): 4-11.

Lieberman, Henry. "Programming by Example." Communications of the ACM 43.3 (2000): 72-74.

Lu, Yueheng, Runjia Tian, Ao Li, Xiaoshi Wang and Jose L. García del Castillo y López. "CubiGraph5K: Organizational Graph Generation for Structured Architectural Floorplan Dataset." Proceedings of the 26th Conference on Computer-Aided Architectural Design Research in Asia. 2021. Volume 1, 81-90.

Martínez Alonso, Nono. "Suggestive Drawing Among Human and Artificial Intelligences." Harvard University, M.Des. Thesis. 2017.

McCullough, Malcolm. Abstracting Craft: The Practiced Digital Hand. MIT Press, 1996.

McCulloch, Warren S., and Walter Pitts. "A Logical Calculus of the Ideas Immanent in Nervous Activity." The Bulletin of Mathematical Biophysics 5.4 (1943): 115-133.

Minsky, Marvin L., and Seymour A. Papert. Perceptrons: An Introduction to Computational Geometry. MIT Press, 1969.

Minsky, Marvin L., and Seymour A. Perceptrons: Expanded Edition. MIT Press, 1988.

Nauata, Nelson, Kai-Hung Chang, Chin-Yi Cheng, Greg Mori, and Yasutaka Furukawa. "House-GAN: Relational Generative Adversarial Networks for Graph-Constrained House Layout Generation." European Conference on Computer Vision. Springer, 2020: 162-177.

Negroponte, Nicholas. The Architecture Machine: Toward a More Human Environment. MIT Press, 1970.

Nelson, Theodore H. "The Right Way to Think About Software." The Art of Human-Computer Interface Design. Ed. Brenda Laurel. Addison-Wesley, 1990. 235-43.

Peters, Nathan. Enabling Alternative Architectures: Collaborative Frameworks for Participatory Design. Harvard University, M.Des. Thesis, 2016.

Pitts, Walter, and Warren S. McCulloch. "How We Know Universals: The Perception of Auditory and Visual Forms." The Bulletin of Mathematical Biophysics 9.3 (1947): 127-147.

Radford, Alec, Luke Metz, and Soumith Chintala. "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks." International Conference on Learning Representations. 2016.

Rosenblatt, Frank. Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Cornell Aeronautical Lab Inc., 1961.

Samuel, Arthur L. "Some Studies in Machine Learning Using the Game of Checkers." IBM Journal of Research and Development 3.3 (1959): 210-229.

Steinfeld, Kyle. "GAN Loci." Proceedings of the 39th Annual Conference of the Association for Computer-Aided Design in Architecture. 2019: 392-403.

Sutherland, Ivan. Sketchpad: A Man-Machine Graphical Communication System. Massachusetts Institute of Technology, Ph.D. Dissertation, 1963.

Toulkeridou, Varvara. "Steps Towards AI Augmented Parametric Modeling Systems for Supporting Design Exploration." Proceedings of the 37th eCAADe Conference on Education and Research in Computer Aided Architectural Design in Europe. 2019, Vol. 1: 81-90

Turing, Alan M. "Computing Machinery and Intelligence." Mind, New Series, 59.236 (1950): 433-460.

Victor, Bret. "The Future of Programming." DBX Dropbox Developer Conference. San Francisco. 9 July 2013. Keynote lecture.

Weisberg, David E. "The Engineering Design Revolution." CADHistory.net, 2008.

White House. Preparing for the Future of Artificial Intelligence. Technical Report. Executive Office of the President, National Science and Technology Council Committee on Technology. 2016.

Willis, Karl D.D., Pradeep Kumar Jayaraman, Joseph G. Lambourne, Hang Chu, and Yewen Pu. "Engineering Sketch Generation for Computer-aided Design." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.

Willis, Karl D.D., Yewen Pu, Jieliang Luo, Hang Chu, Tao Du, Joseph G. Lambourne, Armando Solar-Lezama, and Wojciech Matusik. "Fusion 360 Gallery: A Dataset and Environment for Programmatic CAD Construction from Human Design Sequences." ACM Transactions on Graphics (TOG) 40.4 (2021): 1-24.

Wu, Zhirong, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. "3D ShapeNets: A Deep Representation for Volumetric Shapes." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

Xu, Tao, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. "AttnGAN: Fine-grained Text to Image Generation with Attentional Generative Adversarial Networks." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 1316-1324.

Zeng, Zhiliang, Xianzhi Li, Ying Kin Yu, and Chi-Wing Fu. "Deep Floor Plan Recognition Using a Multi-task Network with Room-boundary-guided Attention." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

Zhu, Jun-Yan, Taesung Park, Phillip Isola, and Alexei A. Efros. "Unpaired Image-to-image Translation Using Cycle-consistent Adversarial Networks." Proceedings of the IEEE International Conference on Computer Vision, 2017. 2223-2232.