Classification of artificial intelligence techniques for early architectural design stages.

VISSERS-SIMILON, E., DOUNAS, T. and DE WALSCHE, J.

2024

Original Research Article

Classification of artificial intelligence techniques for early architectural design stages

International Journal of
Architectural Computing
2024, Vol. 0(0) 1–18
© The Author(s) 2024
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/14780771241260857
journals.sagepub.com/home/jac
Sage

Elien Vissers-Similon, Theodoros Dounas and
Johan De Walsche

Abstract

This paper provides a strategic classification of artificial intelligence (AI) techniques based on a systematic literature review and four levels of potential: the levels of input, output, collaboration and creativity. The classification demonstrates the potential and challenges of the AI techniques when used in early stages of architectural design. We aspire to help architects, researchers and developers to choose which AI techniques might be worth pursuing for specific tasks, optimising the use of today's computational power in architectural design workflows. The results of the classification strongly indicate that Evolutionary Computing, Transformer Models and Graph Machine Learning hold the greatest potential for impact in early architectural design, and thus merit the attention to achieve that potential. Moreover, the classification assists with building multi-technique applications and helps to identify the most suitable AI technique for different circumstances such as the architect's programming skills, the availability of training data or the nature of the design problem.

Keywords

Artificial intelligence, early architectural design, sketch design, design support, classification

Introduction

This paper provides a strategic classification of artificial intelligence (AI) techniques that helps architects, researchers and developers assess which AI techniques maximise the available computational power for different design tasks in early architectural design stages. Artificial intelligence is an umbrella term that covers a wide range of techniques, and depending on the nature of the task and the availability of the (training) data, some AI techniques are better suited than others. Rather than classifying AI techniques on computational or technical features, this paper reframes AI techniques in relation to their potential applicability in early stages of architectural design.

Faculty of Design Sciences - Architecture, Universiteit Antwerpen, Antwerpen, Belgium

Corresponding author:
Elien Vissers-Similon, Faculty of Design Sciences - Architecture, Universiteit Antwerpen, Mutsaardstraat 31, Antwerpen 2000, Belgium.
Email: elien.vissers-similon@uantwerpen.be

Background

State of the art research[1–3] indicates that AI techniques can be implemented in various tasks for early architectural design: from performance based goals to form finding, spatial programming and multi-objective optimization. While the state of the art shows which AI techniques have been used in research, it fails to provide a strategic overview on which AI techniques are useful for architectural design practice. A classification that shows the potential and challenges of the AI techniques in relation to early stages of architectural design could help future architects, researchers and developers to choose which AI techniques might be worth pursuing for specific tasks at hand. This leads to the main research question of this paper:

> RQ: Which AI techniques hold the greatest potential for architecture in early design stages, and what are the challenges associated with each of those AI techniques?

Research methodology

Definitions

Before we can classify AI techniques in relation to early stages of architectural design, we define the scope of artificial intelligence techniques as a whole and identify each AI technique relevant to early architectural design.

Scope of artificial intelligence techniques. Artificial intelligence was first coined by John McCarthy and Marvin Minsky in 1955[4] to find out how machines could attain a level of intelligence comparable to human cognition. The field of artificial intelligence – or 'machine intelligence' as it was originally introduced by Alan Turing in his groundbreaking paper Computing Machinery and Intelligence[5] – has evolved steadily and resulted in various definitions of 'intelligent' machines. This paper adheres to Margaret Boden's view on AI techniques to narrow the scope of the perceived intelligence. Boden is an authority at the intersection of cognitive sciences and artificial intelligence. She describes intelligence as a wide spectrum of information-processing capacities, which results in AI using different techniques to solve different tasks. Boden states that these AI techniques aren't usually considered to be 'intelligent' (e.g. computer vision), but do involve humans' psychological skills such as prediction, perception, association, etc.[6] Therefore, this paper only considers computational techniques that have a psychological evaluation aspect as artificially intelligent techniques. Thus, we do not consider computational techniques such as rule-based generative production systems (e.g. cellular automata, shape grammars) to be AI techniques in this paper.

Artificial intelligence techniques relevant to early architectural design. We investigated past research to identify AI techniques relevant to architectural design, starting with three state of the art reviews. The 1995-2021 review from Pena et al. covers articles on the use of AI in conceptual architectural design and shows that a majority of the 75 studied articles made use of Evolutionary Computing techniques.[1] The 2007-2022 review from Topuz et al. covers the use of Machine Learning for architectural design – a subset of AI that categorizes AI techniques that 'learn' from training data – and shows that a majority of the 60 studied articles made use of Neural Networks and their subcategories, such as Artificial Neural Networks (ANN), Convolution Neural Networks (CNN), Deep Neural Networks (DNN) or Generative Adversarial Networks (GAN).[2] These findings are consistent with the 2012-2022 review from Bölek et al. which showed a dominance of both Evolutionary Computing and Neural Networks in past research.[3]

Because the three examined state of the art reviews do not clearly indicate which specific AI techniques have been used, we conduct an additional systematic literature search through the databases Science-Direct, Web of Science, ResearchGate and CumInCAD. This heterogenic group of databases was purposefully selected to conduct a broad and varied search of different AI techniques. We combined the keywords ‘AI’ or ‘artificial intelligence’ with ‘architectural design’ and variants such as ‘architecture’ and ‘building’. The search focuses on the early design stages of architectural design and therefore excludes results of urban design, fabrication or construction. As ‘architecture’ and ‘design’ often appear in the context of computer science (e.g. ‘computer architecture’ or ‘system design’), the search was further restricted in terms of subject areas. On Science-Direct this meant only the subject areas ‘engineering’ and ‘social sciences’ were included, which reduced the results from 82.815 to 29.085 articles. Furthermore, out of the first 200 results, only 11 articles covered the pursued use of AI in architectural design and most of them appeared in the top 50 results. Therefore, those results were considered as a basis for this paper. On Web of Science, the results were restricted to the category ‘architecture’, yielding a total of 219 results. Out of those results, 21 articles covered the pursued use of AI in architectural design. On ResearchGate the search term yielded 53.880 results. Out of the first 100 results, five articles covered the use of AI in architectural design and appeared mostly in the first 50 results. Therefore, those results were considered for this paper. On CumInCAD, the search for ‘artificial intelligence’ or ‘AI’ in either the title or keywords led to 2.083 results. Out of the first 400 results, 26 articles covered the pursued use of AI in architectural design and appeared mostly in the first 200 results. Therefore, only those results were considered as a basis for this paper Figure 1.

The systematic search yields 63 academic publications about the use of AI for a specific task during early architectural design.[7–69] Since 67% of the articles was published during or after 2020, the results can be seen as a representation of recent research. The sum of the techniques in the articles that adhere to the above defined scope of artificial intelligence leads to the following distribution of AI techniques: Agent-based System (6), Reinforcement Learning (3), Swarm Intelligence (3), Graph Machine Learning (4), Generative Adversarial Networks (14), Neural Network (14), Genetic Algorithm (1), Evolutionary Computing (2), Expert System (8), Cellular Automata (2), Support Vector Machine (1), Gradient Boosting (1), Auto Encoder (3), Natural Language Processing (1), Large Language Model (1), Diffusion Model (7).[69]

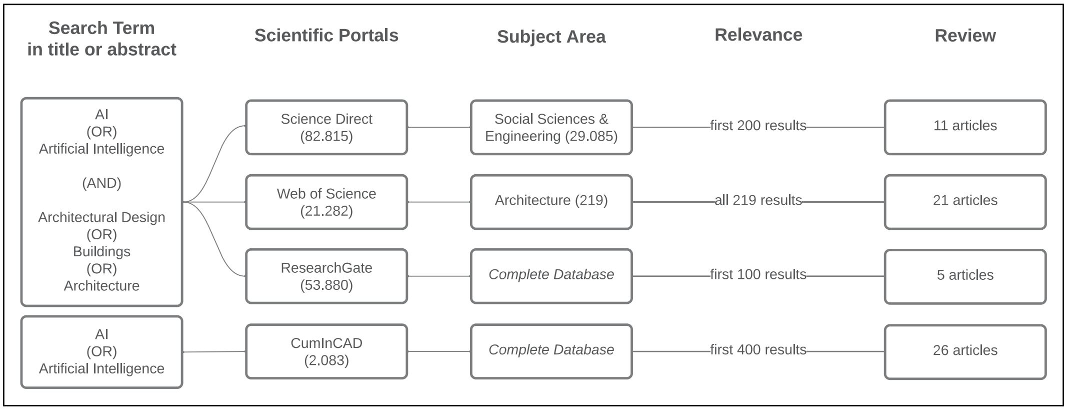

Figure 1. Systematic literature search. Created by first author (2024).

In order to keep the classification clear and useful, we group some of the identified techniques together in super categories. This leads to seven AI techniques that will be classified in this paper. Reinforcement learning and swarm intelligence are both subcategories of agent-based systems (ABS), which consist of agents that have a certain rule-based autonomy and are often able to learn from training data. Graph machine learning (GML) is a subfield of machine learning that works with graphs – mathematical data constructs that hold information about entities and the relationships between those entities. Classic machine learning (CML) contains techniques that use training data to learn in order to perform 'classic' tasks such as regression, classification and prediction, which includes neural networks, support vector machines and gradient boosting. Generative machine learning (GenML) contains techniques that learn from training data to create new data, such as generative adversarial networks or variational autoencoders. Diffusion models and large language models are subcategories of Transformer models (TM), which are extensively trained on large datasets and able to generate new data based on other kinds of data, for example generating images based on textual input. Genetic algorithms are a subcategory of evolutionary computing (EC), a field where algorithms are based on biological rules. Expert systems (ES) contain an embedded database of expert knowledge and are able to infer new knowledge by applying rules to the existing knowledge.

Figure 2 shows the origin of the seven AI techniques and their relationships, while indicating whether the techniques are rule- or data-based. Rule-based AI techniques achieve their inherent intelligence by applying rules to the input data in order to produce novel output. Data-based AI techniques achieve their inherent intelligence by studying both the input and output data (referred to as training data) to resolve the rules of the problem.

While we recognise that our systematic literature search will not have detected *all* articles relating to the use of artificial intelligence in early architectural design, we are confident to have identified the relevant

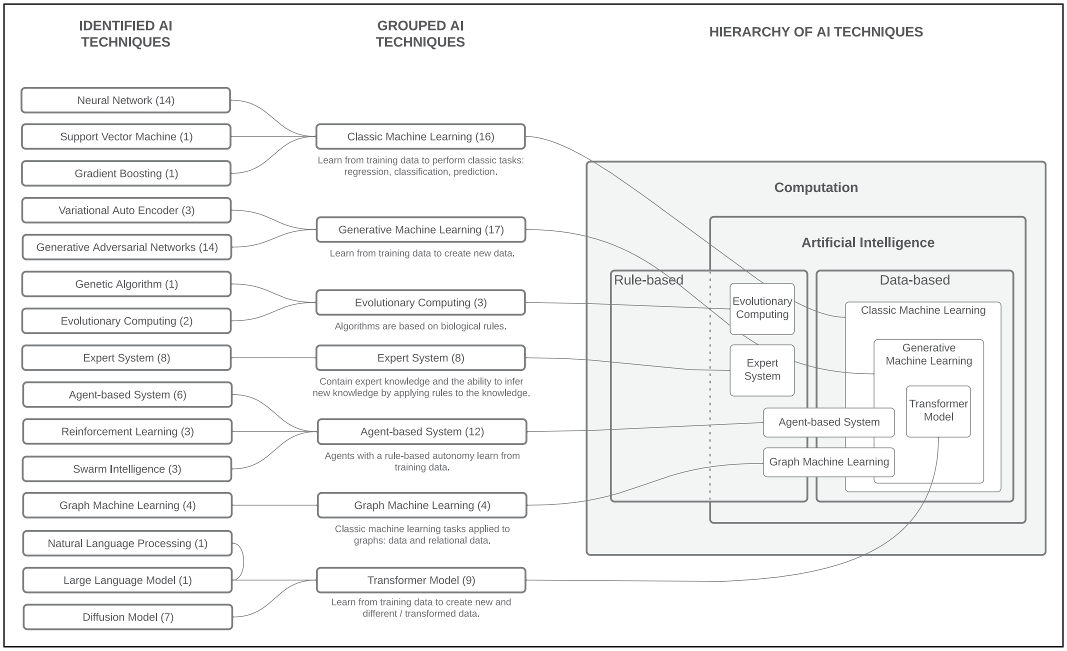
IDENTIFIED AI TECHNIQUES
GROUPED AI TECHNIQUES
HIERARCHY OF AI TECHNIQUES
Neural Network (14)
Support Vector Machine (1)
Gradient Boosting (1)
Variational Auto Encoder (3)
Generative Adversarial Networks (14)
Genetic Algorithm (1)
Evolutionary Computing (2)
Expert System (8)
Agent-based System (6)
Reinforcement Learning (3)
Swarm Intelligence (3)
Graph Machine Learning (4)
Natural Language Processing (1)
Large Language Model (1)
Diffusion Model (7)
Classic Machine Learning (16)
Learn from training data to perform classic tasks: regression, classification, prediction.
Generative Machine Learning (17)
Learn from training data to create new data.
Evolutionary Computing (3)
Algorithms are based on biological rules.
Expert System (8)
Contain expert knowledge and the ability to infer new knowledge by applying rules to the knowledge.
Agent-based System (12)
Agents with a rule-based autonomy learn from training data.
Graph Machine Learning (4)
Classic machine learning tasks applied to graphs: data and relational data.
Transformer Model (9)
Learn from training data to create new and different / transformed data.
Computation
Artificial Intelligence
Rule-based
Data-based
Evolutionary Computing
Expert System
Classic Machine Learning
Generative Machine Learning
Transformer Model
Agent-based System
Graph Machine Learning

Figure 2. AI techniques in this paper. Created by first author (2024).

categories for AI techniques for early architectural design. Furthermore, the AI techniques from the systematic literature search are grouped in the established categories and sorted by year of publication to indicate research trends. Figure 3 shows the dominance of Classic and Generative Machine Learning over the past years, and the rise of Transformer Models – bearing in mind that Diffusion Models[70] were only discovered in 2021 and the first publicly available Large Language Model[71] was launched in 2022.

Methods

Figure 4 The AI techniques are mapped to their potential applicability in early stages of architectural design by evaluating four levels of potential: the level of input, the level of output, the level of creativity and the level

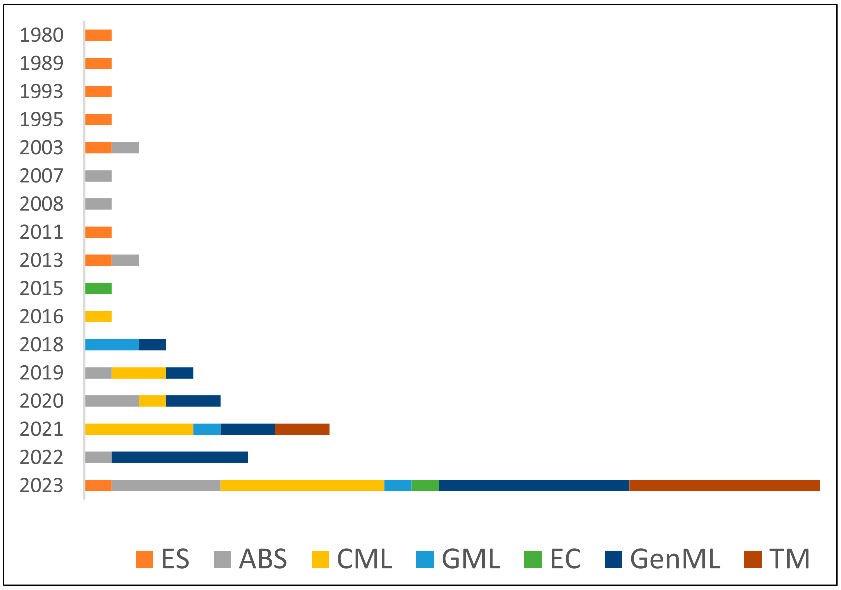

Figure 3. Distribution of AI techniques from the systematic literature search, by year of publication, created by first author (2024).

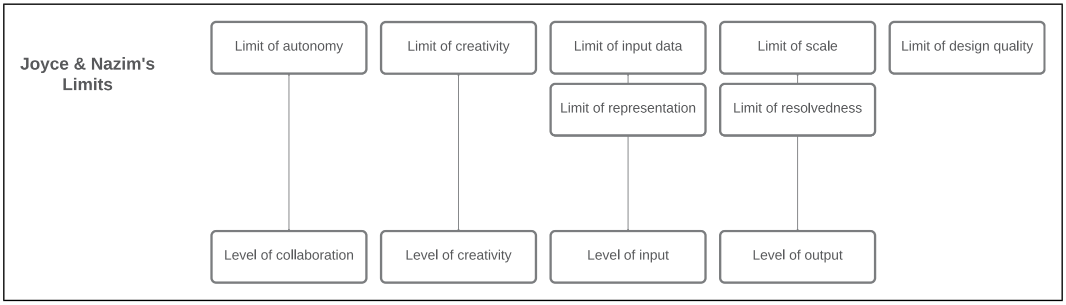

Figure 4. Levels of potential and the limits of machine learning techniques for architectural design. Created by first author (2024).

of collaboration. These criteria are adapted from Joyce & Nazim, who established the limits of machine learning techniques relevant to design tasks.[72] Joyce & Nazim's original limit of design quality has been omitted, as the discussion of *good or bad* architecture is part of a larger debate that is out of the scope of this paper.

The level of input refers to all that is necessary to use a certain AI technique, such as training data, input data, the design problem definition, the architect's expertise, etc. The level of input includes Joyce & Nazim's original limits of representation and input-data. The level of output refers to the desired output from the AI technique, such as textual or geometric data. The level of output includes the original limits of scale and resolved-ness as defined by Joyce & Nazim. The level of creativity refers to the creative freedom that can be achieved by the AI technique, where creativity is regarded as "the ability to come up with ideas or artifacts that are new, surprising, and valuable" as defined by Margaret Boden.[73] The level of collaboration is based on Joyce & Nazim's original limit of autonomy and refers to the human-machine interaction, collaboration with partners and the integration of the AI techniques into the architect's workflows.

The level of input merits two additional notes. Firstly, training and input data can take on any format, from textual city regulations to drawings in PNG or OBJ files. However, there is a lack of large qualitative training datasets for architectural design,[64] which highly impacts data-based techniques. The datasets are often restricted to the architectural design data that are most commonly available to the public – i.e. images of rasterized floorplans. Therefore, those AI techniques that can make use of architectural design data that are available (i.e. using floorplans or the architect's own expertise as input data) or that can generate their own synthetic datasets, are favoured for the level of input. Secondly, architectural design problems can either be well- or ill-defined, which also makes a difference for the level of input. As a whole, design problems are wicked[74] and therefore ill-defined by nature: they consist of a complex mixture of client demands, city regulations, energetic requirements, social aspects, cultural layers, personal preferences etc. However, certain design tasks could be isolated and considered as well-defined problems for the level of input.

Each AI technique is evaluated for each level of potential, which results in a set of potentials and/or challenges. Those are then visually mapped onto a classification structure as proposed in Figure 5. The classification structure is composed of hierarchical diagrams that visually map each level of potential. Values are attributed to the smaller, outer diagrams and projected onto the larger, inner diagram. The levels of output and input are mapped together to create a classification for the design problem. The levels of creativity and collaboration are mapped together to create a classification for the design process. The results of both diagrams are projected onto the larger diagram, which unites them to map the potential of the AI techniques in early architectural design. The classification structure can be interpreted in various ways. e.g., the largest diagram (i.e. the final classification) not only shows the potential of the AI techniques on the resulting diagonal, but also shows whether the AI techniques are more suited towards solving the design problem or being integrated into the design process. Similar interpretations apply to each of the smaller diagrams.

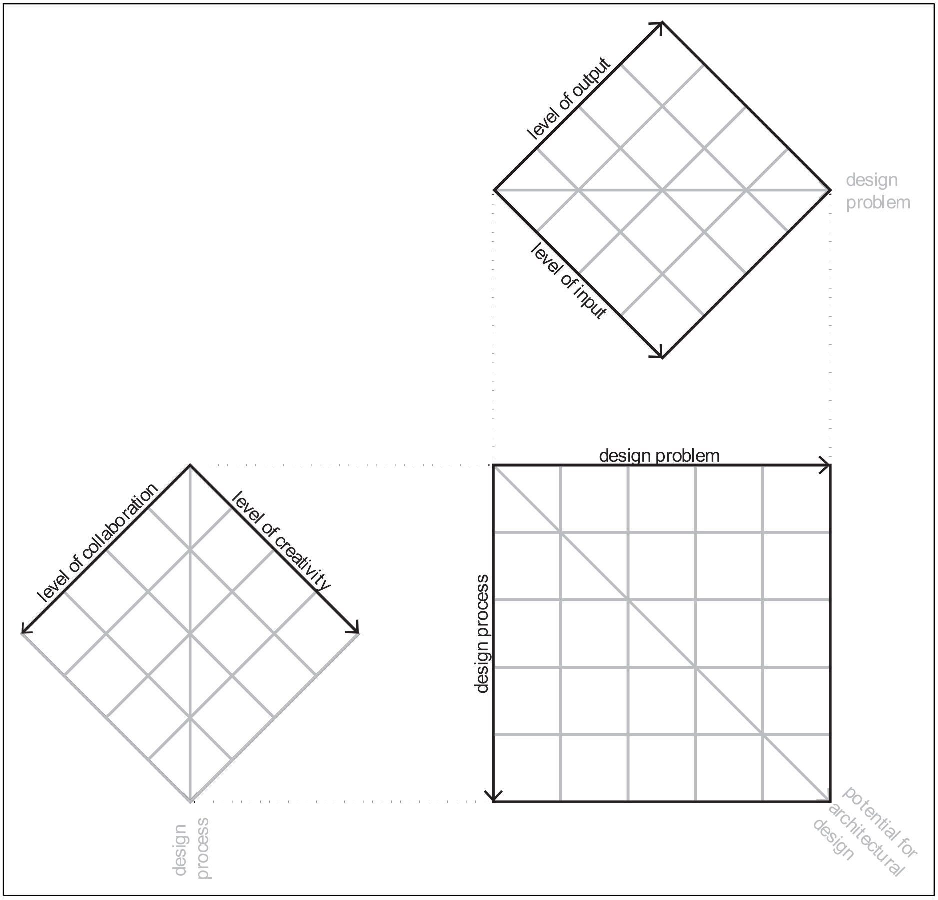

Figure 5. Classification structure to classify the potential of AI techniques for architectural design. Created by first author (2023).

Classification

Each AI technique is evaluated for each level of potential in the tables below, resulting in sets of potentials and challenges for each AI technique. Those challenges and potentials are then visually mapped onto the classification structure Tables 1, 2, 3, 4, 5, 6 and 7.

The classification structure is extended (Figure 6) to incorporate the various lessons-learned from the evaluations in the tables. The availability of architectural design data, the ill-defined nature of design problems, the compatibility of the output with 3D geometry and the need for architects to possess basic or advanced programming skills – which impacts both the necessary interface and the current design workflow – lead to the addition of four new diagrams in the classification structure. The level of output (Figure 6(a)) is composed by data (i.e. analyses and diagrams vs plain text) and geometry (i.e. 3D models vs 2D images). The level of input (Figure 6(c)) is composed by the capability to solve ill-defined design problems and the availability of data (Figure 6(b)), which in turn is composed by the compatibility with existing architectural design data or the capability to generate new data. The level of collaboration (Figure 6(e)) is composed by

Table 1. Levels of potential for expert system.

	Expert System
Level of input	[challenge] Not suited for ill-defined problems as the expert system requires the problem and rules to be clearly described and explicitly coded [potential] Suited for well-defined problems [potential] expert Systems are based on the architect's own expertise, which means the required data will always be available
Level of output	[potential] The output of an expert system can facilitate diagrams or analyses or can be mapped on geometric rules
Level of collaboration	[challenge] Translating spatial problems into rules and constraints can be counter-intuitive for architects and may require a strong mathematical and/or programming foundation [challenge] expertise is very specific and takes a long time to convert into strict rules and constraints, which might nullify any productivity gains of using the expert system
Level of creativity	[challenge] New and surprising results are limited to the well-defined partial-problems that fit within expert Systems and rely on a very thorough level of human-machine collaboration

Table 2. Levels of potential for classic machine learning.

	Classic machine learning
Level of input	[potential] Can discover latent patterns or hidden connections and could potentially bridge the gap between ill-defined problems and well-defined solutions [challenge] Data-based techniques need suitable datasets, in quality and quantity
Level of output	[potential] Suited for decision-making and analysis of well-defined sub-problems
Level of collaboration	[challenge] dealing with data is counterintuitive for architects with a graphical workflow: gathering large quantities, pre-processing data into the right format, etc. [challenge] in order to apply machine learning techniques, the architect would need a small background in programming languages
Level of creativity	[potential/challenge] The discovery of latent patterns or hidden connections could lead to new, surprising and valuable results; but is also highly dependent on large enough quantities of training data to start with

Table 3. Levels of potential for graph machine learning.

	Graph machine learning
Level of input	[potential] possibility to work with the architect's expertise – which is always available – or generate new datasets in a format that allows for further Machine learning techniques. Furthermore, graphs can be extracted from 3D modelling environments, since 3D models are essentially hierarchical data structures as well [potential] graphs are a very well-defined data structure. As such, they cannot represent ill-defined problems. However, their machine learning capabilities can discover latent patterns or hidden connections, making them a candidate to bridge the gap between ill-defined problems and well-defined solutions
Level of output	[potential] The structured data of a graph can be further analysed or mapped onto geometry
Level of collaboration	[potential] in general, data structures are not intuitive for architects to work with. However, graphs are *graphical* data structures with a strong emphasis on the relationship between entities. They share similar properties to elements from the architect's usual workflow, such as bubble diagrams that show relationships between spaces. In that regard, graphs can be seen as an advanced extension of the architect's usual diagram tools [challenge] in order to apply graph *Machine Learning* techniques – to use the full potential of graphs, the architect would need a small background in programming languages
Level of creativity	[potential/challenge] The discovery of latent patterns or hidden connections could lead to new, surprising and valuable results; but is also highly dependent on large enough quantities of training data to start with

Table 4. Levels of potential for generative machine learning.

	Generative machine learning
Level of input	[challenge] architectural design data is not sufficiently available for the application of machine learning techniques. Currently, GenML techniques on the building level mostly focus on floorplan generation, but deprez et al. concluded that even with sufficient data, pixel-based floorplan generation takes too much computational time and will remain trivial[75]
Level of output	[challenge] Current machine learning techniques facilitate the creation of 2D images, but struggle with 3D geometry[76]
Level of collaboration	[challenge] dealing with data is counterintuitive for architects with a graphical workflow: gathering large quantities, pre-processing data into the right format, etc. [challenge] in order to apply machine learning techniques, the architect would need a small background in programming languages
Level of creativity	[potential/challenge] The discovery of latent patterns or hidden connection could lead to new, surprising and valuable results; but is also highly dependent on large enough quantities of training data to start with

Table 5. Levels of potential for agent-based system.

	Agent-based System
Level of input	[potential] Can work with qualitative data, which does not require large quantities [potential] offers a unique way to integrate the architect's expertise into AI: the architect has full control over the positive or negative rewards the agent receives [challenge] The environment in which the agent moves around, needs to be very well-defined
Level of output	[potential] The agent is able to learn in a 3D modelling environment
Level of collaboration	[potential] an agent could make design decisions, after which the designer could assign positive or negative rewards to each decision, creating an interesting human-machine collaboration during the training phase of the algorithm [challenge] in order to apply agent-based techniques, the architect would need a strong background in programming languages
Level of creativity	[potential/challenge] results have a high chance of being new and surprising, since the agents explore an unknown conceptual space based on the reward system. However, if the reward system is very straightforward, the results might be exactly what was to be expected

Table 6. Levels of potential for evolutionary computing.

	Evolutionary computing
Level of input	[potential] Compatible with existing data such as 3D models, and able to generate new data (e.g. new generations) from it [challenge] works best with well-defined sub-problems. For example, allowing shapes to optimize to a form that best achieves a certain goal (view optimization), while taking restrictions into account (plot and neighbours)
Level of output	[potential] Compatible with 3D modelling software to generate geometry or other data
Level of collaboration	[potential] process of design iterations where artists choose their favourite results, which the algorithm uses to generate a new set of results: a great human-machine collaboration [potential] integration in 3D modelling software, i.e. integrated in rhinoceros' visual scripting environment[77]
Level of creativity	[potential] The level of creativity is steered by the architect, who chooses which iteration to develop further. Furthermore, if the algorithm is allowed to 'evolve' some of its own core concepts, it might to break out of its box and reach completely unexpected results

Table 7. Levels of potential for transformer model.

	Transformer Model
Level of input	[potential] understands explicit and implicit textual input in an intuitive way, incl. Abstractions and ambiguous concepts [potential] potential to bridge the gap between textual and visual training data? [potential] taps into the architect's own expertise, which is always available [potential] many open-source models are (freely) available. Furthermore, it is not too complicated for architects to train their own diffusion or GPT models [challenge] Currently limited to text or 2D images, so the problem-solving is limited to the textual or 2D visual space
Level of output	[challenge] Currently limited to textual or 2D visual output space [potential] Can be told how to format its output: i.e. exact aspect ratio and resolution, JSON or OBJ notations for compatibility with 3D modelling software, or writing complex algorithms in specific programming languages that the architect does not technically master [challenge] The generated output is dependent on the data the model was trained on, which is usually very generic [challenge] Despite understanding implicit input, there is no guarantee it can solve ill-defined problems
Level of collaboration	[potential] require very little programming skills [potential] easy and very fast design iterations [potential] intuitive interface [potential] easy incorporation in current workflows
Level of creativity	[potential]Can attain high levels of creativity due to their ability to mix different concepts into a conceptual whole

how intuitive the interface is for architects and how easy it is to integrate the AI techniques into the architectural design workflow (i.e. too time-intensive vs productive design iterations).

The results of the evaluations are mapped onto the final classification structure as output data and geometry (Figure 6(a)), existing data availability and data generation (Figure 6(b)), ability to solve ill-defined problems (Figure 6(c)), intuitive interface and workflow integration (Figure 6(e)) and creativity (Figure 6(f)). This leads to the potential of AI techniques for early architectural design in terms of the design problem (Figure 6(d)), the design process (Figure 6(f)) and early architectural design as a whole (Figure 6(g)).

Discussion

The interpretation of the classification results (Figure 6(g)) is as follows, from a high potential to a low potential: Evolutionary Computing, Transformer Model, Graph Machine Learning, Agent-based System, Classic Machine Learning, Expert System, Generative Machine Learning. Based on this classification, there are strong indications that Evolutionary Computing, Transformer Models and Graph Machine Learning hold the greatest potential for future research on the application of AI in early stages of architectural design.

Evolutionary computing

The systematic literature search has shown that Evolutionary Computing techniques for early architectural design have been studied extensively since the conception of architectural computing, yet lost traction around

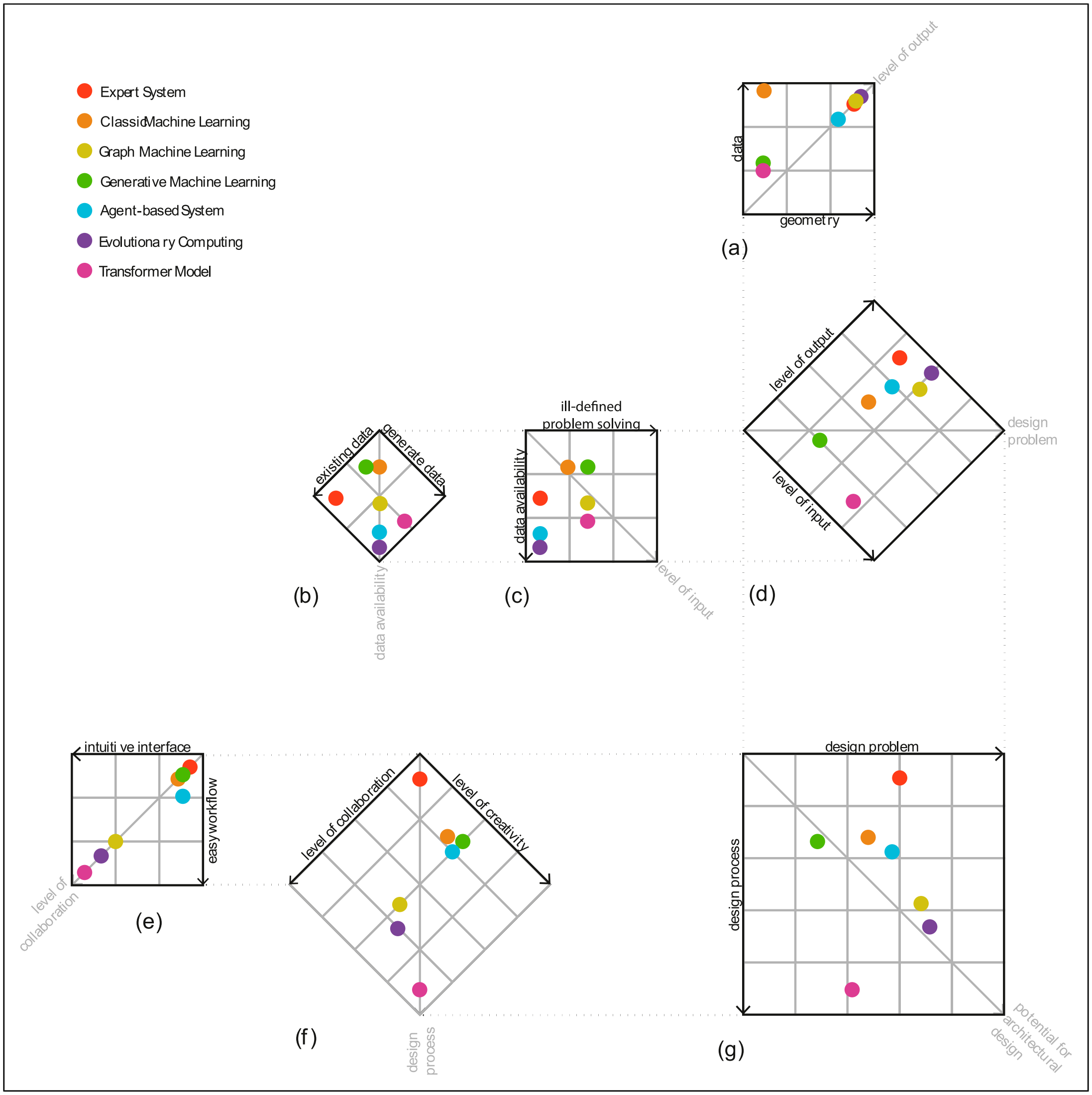

Figure 6. Classification of AI techniques for architectural design created by first author (2023).

the 2010's. This is consistent with the rise of the current 'summer' of AI, also known as the era of machine (deep) learning,[78] when focus shifted from rule-based determination towards data-based techniques that rely on approximation.

Figure 7 The classification indicates that Evolutionary Computing techniques hold the greatest potential for early architectural design. The deterministic nature of these techniques – meaning that solutions mostly evolve within an a priori set system – can therefore be seen as both a strength and a question for future research: can Evolutionary Computing techniques evolve and change their own system over time? We recommend that research on Evolutionary Computing is continued and viewed through a different lens. Moreover, in combination with Transformer Models – more specifically, Large Language Models that are

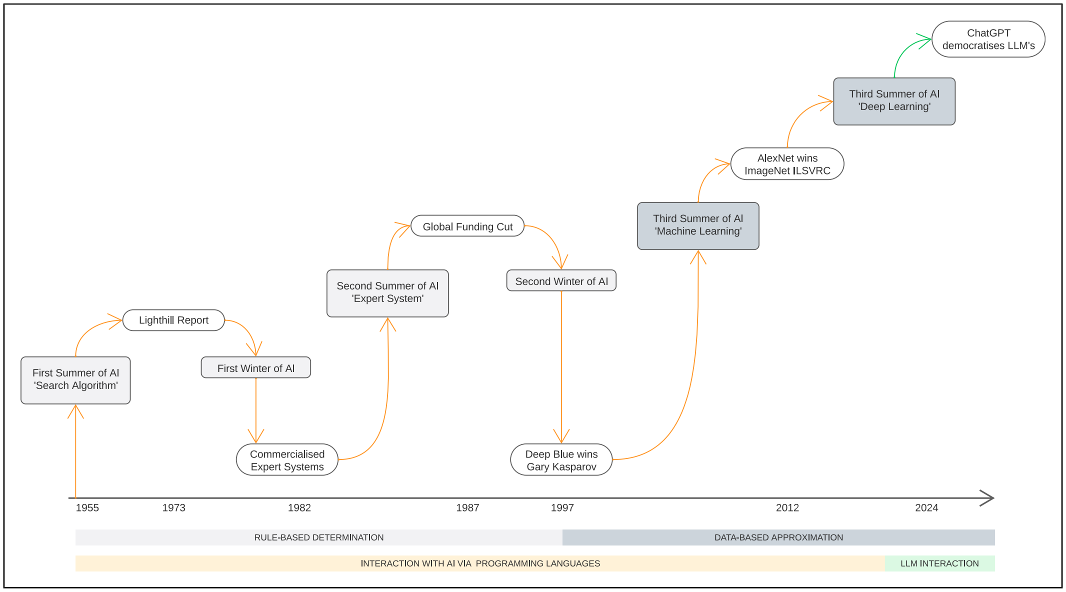
ChatGPT democratises LLM's
Third Summer of AI 'Deep Learning'
AlexNet wins ImageNet ILSVRC
Third Summer of AI 'Machine Learning'
Global Funding Cut
Second Summer of AI 'Expert System'
Second Winter of AI
Lighthill Report
First Winter of AI
First Summer of AI 'Search Algorithm'
Commercialised Expert Systems
Deep Blue wins Gary Kasparov
1955
1973
1982
1987
1997
2012
2024
RULE-BASED DETERMINATION
DATA-BASED APPROXIMATION
INTERACTION WITH AI VIA PROGRAMMING LANGUAGES
LLM INTERACTION

Figure 7. History of AI created by first author (2024), redrawn and adapted from Toosi et al.[78]

trained to write algorithms in various programming languages – the architect can further explore Evolutionary Computing techniques without needing to fully master technical programming languages.

Transformer models

Transformer Models score very high on the criteria of the design process (Figure 6(f)): the intuitive interface, ease to incorporate in current workflows and ability to enhance the creative process makes them very convenient for architects to experiment with. A logical consequence is the fast increase of research into Transformer Models – as shown by the systematic literature search. However, more research needs to be conducted into exactly *how* and *when* Transformer Models can be integrated into a productive workflow. Do they actually accelerate the architect's process, or do they form an additional step that prolongs the process? Is there more to them than rapid ideation and visualisation?

Transformer Models score rather low on the criteria of the design problem (Figure 6(d)). This is mostly due to the fact that Transformer Models are currently only able to generate textual and 2D visual output. However, this is a rapidly developing field and the control over the desired output increases every day. The 2D visual output is already compatible with depth maps and line drawings thanks to plugins such as ControlNet,[79] which could serve as a stepping stone towards 3D model output in the future. More research and development on 2D spatial composition or 3D model output could be conducted to investigate Transformer Models' true potential for architectural design. If Transformer models could attain a similar efficiency on 3D model data, that would be a massive leap forward for the field of architectural computing and the practice of architectural design.

Graph machine learning

Graph Machine Learning vastly outperforms Classic and Generative Machine Learning techniques in the classification. This is related to the way graphs store data: both information on the entities and information on the relationships between those entities is saved and easily accessible to apply rules and machine learning analyses such as classification or prediction. Those relational data translate well into the field of architectural design, where spatial relationships are of great importance. Like other machine learning techniques, Graph Machine Learning depends heavily on the availability of sufficient training data. Since graphs are a mathematical data construct, the extraction of graphs from IFC Models – a standard for BIM models that stores shapes, spatial elements, materials etc. in a hierarchical data structure – can be automated,[80] or graphs can be constructed based on floorplans fairly easily with the aid of computer vision techniques.[81] Further research could be conducted into the ideal graph structure for widespread research goals: which information i.e. relevant in early design stages should or should not be contained in graphs? The most basic graph configuration (i.e. names of spaces and their spatial relationships) corresponds with the architect's use of 'bubble diagrams'. This basic graph structure could be extended with additional information such as: are the borders between spaces load bearing, what is the energy production/consumption of each space, how is vertical and horizontal circulation embedded in the graphs, how should graphs handle latent transient spaces, how can complex data about the environment be integrated, etc.

Graph Machine Learning scores fairly high on the criteria for the design process – due to the intuitive overlap with 'bubble diagrams' and the integrated 3D modelling that software such as topologic grants,[82] and scores even better on the criteria for the design problem due to the fact that they can output spatial information. Current research already shows the potential growth by focusing on graph machine learning: classifying building/ground relationships and enhance energy analysis workflows in architectural practice.[83,84] We recommend that the research community uses the accumulated expertise from classic and generative machine learning techniques in architectural design to shift the focus on graph machine learning applications.

Multi-technique applications

The proposed classification not only shows which AI techniques hold the greatest potential for architectural design individually (Figure 6(g)), but also sheds light on which AI techniques perform better in certain circumstances. For example, when there is no existing training data, architects can opt for Transformer Models, Evolutionary Computing or Agent-based Systems that can generate data (Figure 6(b)). This opens the door for a targeted multi-technique approach: applications usually make use of several chained AI techniques and the proposed classification can help identify which AI techniques are more suitable for each part of the chain. This may lead to, for example, Transformer Agent-based Models where the Transformer Model generates a wide variety of custom agents that are fed into an Agent-based Model.

Architectural design data

The accessibility, quantity and quality of architectural design data strongly impacts the level of input. A wide range of architectural design data – such as material, geometric, social, cultural or site information – remains untapped in current architectural design datasets, which are often restricted to spatial layout information (i.e. rasterised floorplans, bubble diagrams). Even so, the existing databases of architectural design data are comparatively small in relation to databases used for AI techniques in other fields, which often contain millions of datapoints.[85] In architectural design, one of the largest known databases, RPLAN, consists of 80.000 color-coded 255 x 255 pixel images of floorplans.[86] Other datasets contain significantly less datapoints, such as the CubiCasa5k dataset of 5.000 floorplans with manually added annotations and the matching CubiGraph5k dataset of 4.000 graphs.[81,87] As more architects integrate AI techniques in their workflows, disclosing architectural design data might become an evident part of the design process. This would in turn impact the results of the classification (through Figure 6(b)) in the coming years and potentially lead to a profound shift in establishing accessible, quantitative and qualitative architectural design datasets.

As disclosing the data is a crucial factor, we recommend that future research looks into facilitating ways of sharing data among both researchers and architects.

Conclusion

The proposed classification provides a strategic overview of suitable AI techniques for early architectural design stages and thus offers a targeted direction for architects, researchers and developers to determine areas of focus in the coming years. The results strongly indicate that Evolutionary Computing, Transformer Models and Graph Machine Learning hold the greatest potential for impact in early architectural design, and thus merit the research community's attention to achieve that potential. Computer science and AI models develop rapidly, hence it is not out of the question that an evolution of these or some other models might come to dominate the landscape in the years to come. As such, we urge the researchers to monitor new developments. The results of the proposed classification suggest that research and development resources could be optimised by shifting towards the application of Evolutionary Computing, Transformer Models and Graph Machine Learning in early architectural design. Moreover, the classification assists with building multi-technique applications and helps to identify the most suitable AI technique for different circumstances such as the architect's programming skills, the availability of training data or the nature of the design problem.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

Open access data

Research data and figures are available on https://github.com/ElienVissers/classification-AI-early-architectural-design.

ORCID iD

Elien Vissers-Similon https://orcid.org/0000-0003-4922-6425

References

1. Pena MLC, Carballal A, Rodriguez-Fernandez N, et al. Artificial intelligence applied to conceptual design. A review of its use in architecture. *Autom ConStruct* 2021; 124: 103550.
2. Topuz B and Çakici Alp N. Machine learning in architecture. *Autom ConStruct* 2023; 154: 105012. DOI: 10.1016/j.autcon.2023.105012
3. Bölek B, Tutal O and Özbaşaran H. A systematic review on artificial intelligence applications in architecture. *Journal of Design for Resilience in Architecture and Planning* 2023; 4: 91–104. DOI: 10.47818/DRArch.2023.v4i1085
4. McCarthy J, Minsky ML and Shannon CE. A proposal for the Dartmouth summer research project on artificial intelligence - august 31, 1955. *AI Mag* 2006; 27: 12–14.
5. Turing AM. Computing Machinery and intelligence. *Mind* 1950; 59: 433–460.
6. Boden MA. *Artificial intelligence: a very short introduction*. Oxford University Press, 2018.
7. Long LD. An AI-driven model for predicting and optimizing energy-efficient building envelopes. *Alex Eng J* 2023; 79: 480–501. DOI: 10.1016/j.aej.2023.08.041.

8. Aalaei M, Saadi M, Rahbar M, et al. Architectural layout generation using a graph-constrained conditional Generative Adversarial Network (GAN). *Autom ConStruct* 2023; 155: 105053. DOI: 10.1016/j.autcon.2023.105053
9. Ko J, Ennemoser B, Yoo W, et al. Architectural spatial layout planning using artificial intelligence. *Autom ConStruct* 2023; 154: 105019. DOI: 10.1016/j.autcon.2023.105019
10. Castro Pena ML, Carballal A, Rodríguez-Fernández N, et al. Artificial intelligence applied to conceptual design. A review of its use in architecture. *Autom ConStruct* 2021; 124: 103550. DOI: 10.1016/j.autcon.2021.103550
11. Qian W, Yang F, Mei H, et al. Artificial intelligence-designer for high-rise building sketches with user preferences. *Eng Struct* 2023; 275: 115171. DOI: 10.1016/j.engstruct.2022.115171
12. Hanafy NO. Artificial intelligence's effects on design process creativity: A study on used A.I. Text-to-Image in architecture. *J Build Eng* 2023; 80: 107999. DOI: 10.1016/j.jobe.2023.107999
13. Mirra G and Pugnale A. Comparison between human-defined and AI-generated design spaces for the optimisation of shell structures. *Structures* 2021; 34: 2950–2961. DOI: 10.1016/j.istruc.2021.09.058
14. Demir G, Çekmiş A, Yeşilkaynak VB, et al. Detecting visual design principles in art and architecture through deep convolutional neural networks. *Autom ConStruct* 2021; 130: 103826. DOI: 10.1016/j.autcon.2021.103826
15. Zheng H and Yuan PF. A generative architectural and urban design method through artificial neural networks. *Build Environ* 2021; 205: 108178. DOI: 10.1016/j.buildenv.2021.108178
16. Pizarro PN, Hitschfeld N and Sipiran I. Large-scale multi-unit floor plan dataset for architectural plan analysis and recognition. *Autom ConStruct* 2023; 156: 105132. DOI: 10.1016/j.autcon.2023.105132
17. Xiong W, Zhang P, Sander PV, et al. ShapeArchit: shape-Inspired architecture design with space planning. *Comput Aided Des* 2022; 142: 103120. DOI: 10.1016/j.cad.2021.103120
18. Veloso P and Krishnamurti R. Spatial synthesis for architectural design as an interactive simulation with multiple agents. *Autom ConStruct* 2023; 154: 104997. DOI: 10.1016/j.autcon.2023.104997
19. Harapan A, Rizkiya N and Azbi R. Artificial intelligence in architectural design. *Int J Des* 2021; 1: 1–6. DOI: 10.34010/injudes.v1i1.4824
20. Hegazy M and Saleh A. Evolution of AI role in architectural design: between parametric exploration and machine hallucination. *MSA Engineering Journal* 2023; 2: 262–288. DOI: 10.21608/msaeng.2023.291873
21. Kim D. Latent morphologies: encoding architectural features and decoding their structure through artificial intelligence. *Int J Architect Comput* 2023: 1–20. DOI: 10.1177/14780771231209458
22. Yuan C and Zheng H. *A new architectural design methodology in the age of generative artificial intelligence*, 2023. DOI: 10.19819/j.cnki.ISSN0529-1399.202310005
23. Chen J, Wang D, Shao Z, et al. Using artificial intelligence to generate master-quality architectural designs from text descriptions. *Buildings* 2023; 13: 2285. DOI: 10.3390/buildings13092285
24. Veloso P and Krishnamurti R. An Academy of Spatial Agents Generating spatial configurations with deep reinforcement learning&gt. In: *ECAADE 2020: anthropologic - architecture and fabrication in the cognitive age* 2020, pp. 191–200.
25. Ayuso LA and del Blanco García FL. Application of neural networks to the design of collective housing Automation and combinatorial generative processes using artificial intelligence. In: *Rita-Revista Indexada De Textos Academicos* 2021, p. 214. DOI: 10.24192/2386-7027(2021)(v16)(20).
26. Chan YHE and Spaeth AB. Architectural visualisation with conditional generative adversarial networks (cGAN). In: *What machines read in architectural sketches. Ecaade 2020: Anthropologic - Architecture And Fabrication In The Cognitive Age*. 2020, p. 299–308.
27. As I, Pal S and Basu P. Artificial intelligence in architecture: generating conceptual design via deep learning. *Int J Architect Comput* 2018; 16: 306–327. DOI: 10.1177/1478077118800982
28. Gargaro S and Fioravanti A. A context-knowledge model for architectural design a holistic approach by means of artificial intelligence techniques. In: *ECAADE 2013: computation and performance*. 2013, pp. 81–90.
29. Newton D. Deep generative learning for the generation and analysis of architectural plans with small datasets. In: *ECAADE SIGRADI 2019: Architecture in the age of the 4th industrial revolution*. 2019, pp. 21–28.

30. Rodrigues RC and Duarte RB. Generating floor plans with deep learning: a cross-validation assessment over different dataset sizes. *Int J Architect Comput* 2022; 20: 630–644. DOI: 10.1177/14780771221120842
31. Eisenstadt V, Langenhan C and Althoff KD. Generation of Floor Plan Variations with Convolutional Neural Networks and Case-based Reasoning An approach for transformative adaptation of room configurations within a framework for support of early conceptual design phases. In: *ECAADE SIGRADI 2019: architecture in the age of the 4th industrial revolution*. 2019, pp. 79–84.
32. Çelik T. Generative design experiments with artificial intelligence: reinterpretation of shape grammar. *Open House Int* 2023. DOI: 10.1108/OHI-04-2023-0079
33. Kudless A. Hierarchies of bias in artificial intelligence architecture: collective, computational, and cognitive. *Int J Architect Comput* 2023; 21: 256–279. DOI: 10.1177/14780771231170272
34. Kim D. Latent morphologies: encoding architectural features and decoding their structure through artificial intelligence. *Int J Architect Comput* 2023. DOI: 10.1177/14780771231209458
35. Özerol G and Selçuk SA. Machine learning in the discipline of architecture: a review on the research trends between 2014 and 2020. *Int J Architect Comput* 2023; 21: 23–41. DOI: 10.1177/14780771221100102
36. Yazici S. A machine-learning model driven by geometry, material and structural performance data in architectural design process. In: *ECAADE 2020: Anthropologic - architecture and fabrication in the cognitive age*. 2020, pp. 411–418.
37. Koehler D. More than anything: advocating for synthetic architectures within large-scale language-image models. *Int J Architect Comput* 2023; 21: 242–255. DOI: 10.1177/14780771231170455
38. Huang J, Johanes M, Kim FC, et al. *On GANs, NLP and Architecture: Combining Human and Machine Intelligences for the Generation and Evaluation of Meaningful Designs*, 2021, vol 5, pp. 207–224.
39. Mrosla L, Koch V and von Both P. *Quo vadis AI in Architecture? Survey of the current possibilities of Al in the architectural practice*. 2019, p. 45–54.
40. von Mammen S and Jacob C. Swarm-driven idea models - from insect nests to modern architecture. In: *Eco-Architecture Ii: Harmonisation Between Architecture And Nature* 2008, p. 117–126.
41. Jabi W, Chatzivasileiadi A, Wardhana NM, et al. The synergy of non-manifold topology and reinforcement learning for fire egress. In: *ECAADE SIGRADI 2019: Architecture in the age of the 4th industrial revolution* 2019, pp. 85–94.
42. del Campo M, Carlson A and Manninger S. Towards hallucinating machines-designing with computational vision. *Int J Architect Comput* 2021; 19: 88–103. DOI: 10.1177/1478077120963366
43. Sebestyen A, Hirschberg U and Rasoulzadeh S. Using deep learning to generate design spaces for architecture. *Int J Architect Comput* 2023; 21: 337–357. DOI: 10.1177/14780771231168232
44. Trabucco D. Will artificial intelligence kill architects? An insight on the architect job in the AI future. In: *Techne-journal of technology for architecture and environment* 2021, pp. 128–132. DOI: 10.13128/techne-10696
45. Dokonal WHU and Wurzer G. AI Diffusion as Design Vocabulary - investigating the use of AI image generation in early architectural design and education. In: *Design Reconsidered*, 2023.
46. Kavakoglu AA, Almag B, Eser B, et al. AI driven creativity in early design education: a pedagogical approach in the age of industry 5.0. In: *Proceedings of the International Conference on Education and Research in Computer Aided Architectural Design in Europe*, 2022, pp. 133–142.
47. Rahbar M, Mahdavinejad M, Markazi AHD, et al. Architectural layout design through deep learning and agent-based modeling: a hybrid approach. *J Build Eng* 2022; 47: 103822. DOI: 10.1016/j.jobe.2021.103822
48. Chan YHESAB. Architectural visualisation with conditional generative adversarial networks (cGAN). - what machines read in architectural sketches. In: *Anthropologic: Architecture and Fabrication in the cognitive age - Proceedings of the 38th eCAADe Conference Berlin*, 2020.
49. Berger SR. Artificial intelligence and its impact on computer-aided design. *Des Stud* 1980; 1: 166–171, DOI: 10.1016/0142-694X(80)90024-1
50. Chakraborty S. *Automated generation of residential roomlayout within a constrained covered area*, 2003, pp. 85–100.

51. Basarir L and Erol K. *Briefing AI: from architectural design brief texts to architectural design sketches*, 2021.
52. Birmingham W and Siewiorek DP. *Capturing designer expertise the CGEN system*, 1989, pp. 610–613.
53. Arora H, Bielski J, Eisenstadt V, et al. *Consistency Checker an automatic constraint-based evaluator for housing spatial configurations*, 2021.
54. Bidgoli A and Veloso P. DeepCloud: the application of a data-driven. In: *Generative model in design*, 2018.
55. Carrara G, Novembri G and Novembri G. *Expert System for Building Design*. Washington: Congress of the International Council for Building Research, Studies and Documentation, 1986, vol 2, pp. 651–658.
56. Baharlou E and Menges A. Generative agent-based design computation. In: *Integrating material formation and construction constraints*, 2013.
57. Akizuki Y, Bernhard M, Kladeftira M, et al. Generative modelling with design constraints – reinforcement learning for object generation. In: D Holzer, W Nakapan and A Globa (eds). *5th Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA 2020), Bangkok, Thailand, August 5–6, 2020. Association for Computer Aided Architectural Design Research in Asia*, 2020, p. 454.
58. Osman A. Knowledge-based stair design. In: *Education and Practice: The Critical Interface*, 1993.
59. Bank M, Sandor V, Schinegger K, et al. *Learning Spatiality - a GAN method for designing architectural models through labelled sections*, 2022, pp. 611–619.
60. Rodrigues R, Koga R, Hirota E, et al. *Mapping Space Allocation with Artificial Intelligence - an approach towards mass customized housing units*, 2022.
61. Aksoy Y, Cagdas G and Balaban Ö. *A model for sustainable site layout design of social housing with pareto genetic algorithm*. SSPM, 2015, pp. 113–133.
62. Achten HHJAJ. A multi-agent Mah Jong playing system : towards real-time recognition of graphic units in graphic representations. *Acta Polytechnica* 2003; 43: 28–33.
63. Belém C, Santos L and Leitão A. On the impact of machine learning. Architecture without architects? In: *CAAD Futures*, 2019.
64. Blaas Q, Pelosi A and Brown A. *Reconsidering artificial intelligence as Co-designer*. Open Access Te Herenga Waka-Victoria University of Wellington, 2023.
65. Cenani S, Sehnaz CG, et al. *Representation of user movements with multi agent systems: shopping malls*, 2007.
66. Andersen T and Carlsen NV. Software design of maintainable knowledge-based systems for building design. *Autom ConStruct* 1995; 4: 101–110, DOI: 10.1016/0926-5805(94)00042-L
67. Rodríguez JA. Trained architectonics. In: *Parametricism Vs materialism: evolution of digital technologies for development*, 2016, pp. 461–468.
68. Rossini P. *Using expert systems and artificial intelligence for real estate forecasting*, 2011.
69. Picon A. What about humans? Artificial intelligence in architecture. In: PF Yuan, M Xie and N Leach (eds). *Architectural Intelligence: Selected Papers from the 1st International Conference on Computational Design and Robotic Fabrication (CDRF 2019)*. Singapore: Springer Nature Singapore, 2020, pp. 15–29.
70. Ramesh A, Pavlov M, Goh G, et al. Zero-shot text-to-image generation. *Pr Mach Learn Res* 2021; 139.
71. OpenAI. *Introducing ChatGPT*, 2022. https://openai.com/index/chatgpt
72. Joyce S and Nazim I. *Limits to applied ML in planning and architecture: understanding and defining extents and capabilities*, 2021.
73. Boden M. *The creative mind: myths and mechanisms*. 2nd ed. London and New York: Routledge, 2004, p. 344.
74. Rittel HWJ and Webber MM. Dilemmas in a general theory of planning. *Policy Sci* 1973; 4: 155-169. DOI: 10.1007/Bf01405730
75. Deprez L, Verstraeten R and Pauwels P. Data-based generation of residential floorplans using. *Neural Network* 2023: 321–339.
76. Leach N. *Architecture in the age of artificial intelligence: an introduction to AI for architects*. Bloomsbury Academic, 2022.
77. Rutten D. *Galapagos: On the Logic and Limitations of Generic Solvers*, 2013, vol 83. DOI: 10.1002/ad.1568

78. Toosi A, Bottino A, Saboury B, et al. A brief history of AI: how to prevent another Winter (A critical review). *Pet Clin* 2021; 16. DOI: 10.1016/j.cpet.2021.07.001

79. Zhang L, Rao A and Agrawala M. *Adding conditional control to text-to-image diffusion models*, 2023.

80. Zhu J, Wu P and Lei X. IFC-graph for facilitating building information access and query. *Autom ConStruct* 2023; 148: 104778, DOI: 10.1016/j.autcon.2023.104778

81. Lu Y, Tian R, Li A, et al. *CubiGraph5K: organizational graph generation for structured architectural floor plan dataset*, 2021.

82. Jabi W, Aish R, Lannon S, et al. *Topologic - a toolkit for spatial and topological modelling*, 2018.

83. Jabi W and Dengusiak M. *Topologic and BHoM: enhancing energy analysis workflows through topological modelling enhancing energy analysis workflows through topological modelling*, 2019.

84. Alymani A, Jabi W and Corcoran P. Graph machine learning classification using architectural 3D topological models. In: *Simulation: Transactions of The Society for Modeling and Simulation International*, 2022, pp. 1–15. DOI: 10.1177/00375497221105894

85. Koch B, Denton E, Hanna A, et al. *Reduced, reused and recycled: the life of a dataset in machine learning research*, 2021.

86. Wu W, Xiao-Ming F, Tang R, et al. Data-driven interior plan generation for residential buildings. *ACM Trans Graph* 2019; 38(234): 231–234:212. DOI: 10.1145/3355089.3356556

87. Kalervo A, Ylioinas J, Häikiö M, et al. CubiCasa5K: a dataset and an improved multi-task model for floorplan image analysis. In: M Felsberg, P-E Forssén and I-M Sintorn (eds). *Image Analysis*. Cham: Springer International Publishing, 2019, pp. 28–40.