# Beyond Artificiality: Synergistic Design Intelligence (SynDI) in Digital and Discrete Architecture

**Serdar Aydin**
**Beryar Zübeyde Balaban**

**173–194**

Architecture is undergoing a new digital turn prompted by the use of artificial intelligence, which has sparked both enthusiasm and concern—particularly regarding the notion that computers may replace architects. However, the repositioning of architects provides an opportunity to incorporate a new modus operandi with new forms of production. This paper proposes an alternative approach for human–computer interaction within the framework of discrete architecture – a topical production model for building components. The study relies on a collaborative and forward-thinking approach rather than artificiality, which often distances actors in the architecture field from emerging concepts, terminologies, and methods. It introduces the concept of synergistic design intelligence based on an improvised approach applied to a design studio experiment. The results signify a transition to a hyperspatial paradigm while demonstrating how heterogeneous computational strategies can enhance designers' engagement with new concepts of computational thinking in architecture.

#digital design

#discrete architecture

#synergistic design intelligence

#hyperspace

#human–computer interactions

## Introduction

Synergetics, a concept that attempts to reformulate human–computer interaction (HCI) methodologies, is an interdisciplinary research field that studies the processes by which various subsystems (e.g. atoms, molecules, cells, animals, humans or computers) come together to self-organise spatial, temporal or functional structures at the macroscopic level (Haken 1987). This novel concept can be translated as a modern cognitive model that synthesises detailed information processing for architects who design spaces through the examination of hybrid control mechanisms. Design systems built using heterogeneous methods are said to support designers in both creative and generative processes. This heterogeneity is outlined in Sterk's (2006) HCI model, which is a connected and open system of artificial intelligence (AI) and robotics methodologies. However, some models may be insufficient, as there are multiple layers of HCI in architecture. Academics and practicing architects have different perspectives on this issue, considering the evolutionary power/lessness of contemporary computers' capacities. The main objective of this study is to develop a synergistic design intelligence (SynDI) model that supports designer–computer interaction in the architectural design process and to test this model with the integrated use of parametric and discrete design principles. In particular, this study addresses the impact of digital design tools on the process and outcomes of design and how they relate to new design paradigms. The following questions are discussed within a theoretical and practical framework: How can human–computer interaction (HCI) be made more efficient in architectural design processes? How can discrete architectural design methods be combined with parametric tools to offer creative processes? What approaches are offered by the synergistic design intelligence model in design processes?

In theory, computers and humans are partners in contemporary architectural design processes. Defining the role of designers as observant and practicing actors in this process has been the focus of studies discussing higher- and lower-order systems since the 1960s (Cross 2004; Eastman 1972; Friedman 1972; Sterk 2006). With the introduction of sophisticated computers in the 1990s, new design approaches emerged, such as Gregg Lynn's (1999) embryological house. Coinciding with the early 2000s' computer boom in architecture, Lynn's approach explained architect–computer interaction through the evolution of curvilinear forms. Carpo (2013), who characterised this period as the 'first digital turn', placed models of *starchitects*, such as Frank Gehry, and form-based digital architecture production at the centre of this transformation. Preceded by the earlier morphological approach, HCI in architectural production was theorised nearer to mathematical and numerical design, which has manifested in the loud efforts of Zaha Hadid and Patrick Schumacher since the early 2000s (Schumacher 2008; Terzidis 2003). As the design and representation techniques used by architects have undergone a transformation through new software and technologies, design, computing and cognition have been among the areas of Computer-Aided Architectural Design developed over time (e.g. Gero and Kannengiesser 2012, 2014; Jordanous 2016; Qian and Gero 1996). Architectural visualisation with AI is one of these transformations. However, this is more complex when AI generative tools augment architects' productivity in some parts of their daily work. The concept of generative art has attracted public attention through publicly available platforms, such as Midjourney, which includes easy workflows for generating visuals from text and visuals from visuals (Ploennigs and Berger 2023). Following Carpo's (2017) 'second digital turn', later efforts (Bernstein 2022; Campo 2024; Leach 2022) interpreted the expansion of AI as virtually the third digital turn in architecture. Their views on post-human aesthetics placed the computer as a partner of the architect/designer rather than a mere tool delineated in anthropocentric design processes.

In a more practical discourse than AI's recent role in architecture, discrete architecture offers new opportunities for architectural design in the use of modular parts for parametric structures (Hou and Lin 2022; Retsin 2019). Discrete architecture emphasises the relationships between parts and between part and whole in the placement and contextualisation of wholes. Discrete design is a design model in which parts with separate variables and properties come together with specialised algorithms to form a harmonious whole. In this design model, the connections of the parts, the way they come together, the number of repetitions and their position in the overall shape define the integrity of the design. The concept of discrete architecture is an approach that offers a modular design process by adapting the principles of discrete design to the context of architecture.

Claypool (2020) defined discrete design as inseparable from digital design processes integrated with tools such as AI and automation technologies. As each design tool has its own strengths and weaknesses, testing the conceptual idea of discrete architecture has proven a non-failure-prone articulation of interaction between the computer and the architect, leading to new definitions, such as 'accumulative error' (Jahn et al. 2022). In this way, the assistance of the machine grows into a robotic intervention in the material properties of architecture (Huang and Spaw 2023). Here, a hypothetical understanding of interaction and design systems emerges. However, researchers have paid little attention to how emerging concepts and design interfaces update new versions of previous frameworks, such as Sterk's (2006) scheme based on a discussion of direct manipulation (i.e. deliberate control) and automation (i.e. reflexive control).

This study develops a framework for a design experiment that expands the use of parametric spatial definitions into discrete building components. The key instrument of the experimental setup is a combinatorial design toolkit called Wasp, which enables spatial thinking with discrete elements (Rossi n.d.). A systematic optimisation process integrated into this workflow involves intense form finding, simulation and optimisation of multiple objectives. After an introduction accentuating the novel design approach of digital discretisation, a design process using computational tools is presented. This design research process consists of four phases, including stochastic aggregation of discrete parts, algorithmic generation of modules, their optimisation and the AI-assisted production of hyperspatial features from modules. Finally, the paper discusses the definition of SynDI, referring to HCI in using the principles of discrete architecture while depicting visuals of findings from the design process.

Applied cases of the synergistic design concept include the study by Bibri et al. (2024), which explored how the mutual workflow between the designer and the computer contributes to decision-support mechanisms. These AI systems, which enable designers to receive user feedback by testing 'what if' scenarios in subsequent stages, demonstrate how AI and AI of Things systems evolve into an intelligence infrastructure that supports designers in making effective decisions, particularly in sustainability-centred designs. In the context of synergistic design intelligence, this approach is defined as a design model that merges the designer's creativity with the rapid and efficient data-processing capacity of AI. On the other hand, Chen et al. (2024) considered human–AI collaboration at two different levels: strong interactions contributing to the designer's detailed analysis with clear steps, creating a decision mechanism; and weak interactions contributing to the integration of ideas from different disciplines. The critical point in this study is the planning of the intensity of feedback. Feedback mechanisms from real-time data simulations that use Building Information Modelling and Internet of Things technologies provide a framework for how synergistic design intelligence is applied in practice as the basis for the designer to optimise the processes.

While the previous concepts refer to a one-way interaction process, SynDI can be defined as a two-way interaction process in which creative processes are co-produced in the human–computer relationship. HCI models often describe how the designer interacts directly with the computer. SynDI differs from this concept in that it allows designers to reconstruct their fragmented role in the process. It conceptualises the use of AI tools as collaborators in the construction of this role. This partnership therefore creates fragments of feedback loops that leave gaps for the designer to intervene by creating a non-linear iteration process instead of direct and continuous manipulation. In this study, the beginning of the hypothetical implementation of a SynDI process relies on the use of discrete design tools. The following section explains the implementation of discrete design principles as a generative process of fragmented spatial configurations that align with the intercepted relationships within the SynDI concept.

## Experiment with a Discrete Design Tool

Architectural software integrated with programming languages combines hard-core computational and spatial thinking. For almost two decades, the development of plugins has continuously received interest from architects and designers as well as engineers whose scripted solutions continue to be published as pockets inside plugin libraries. This has created a visual programming environment for architects that is introduced, along with its components, in this section.

The conception of a unique workflow made of four steps questions how synergetic intelligence between the designer and the computer emerges. Each of these experiment-orientated steps is akin to developing different hypotheses, but this paper focuses on discussing only the main question of how discrete architecture helps us improve spatial thinking with computers. Beginning with a stochastic aggregation method, the first phase of the experiment demonstrates the use of combinatorial tools; the second adds manual model improvements on top of the outcome of the initial stochastic search; the third calls for material information and production knowledge in computational design thinking by optimising the previous design outcome to meet hyperspatial requirements; and the fourth augments the whole process with the use of AI tools, once again taking control from the designer to trigger new possibilities, which are identified as part of the optimisation process of the computational–spatial thinking in this research.

To stochastically search sets of hyperspatial compositions, this study deploys Rhinoceros's Grasshopper visual programming environment to complete the integration of parametric production into the digital model while providing an adaptable design output. Grasshopper allows for designing with both tools, including third-party plugins. The general purpose of these plugins is to expand the range of production by providing users with tools in the areas of simulation, structural analysis, data processing and visualisation, geometry manipulation and architectural visualisation. Users can customise their own projects through these plugins. Wasp, which is used in this project and works as a plugin for the platform, is a Grasshopper component developed in Python.

One of the most important advantages of Wasp, which combines the geometric representation of modules with abstract graphical information and offers different methods for these combinations, is the opportunity to produce without the need for prior programming knowledge (Hageman n.d.; Rossi 2017; Yu 2021). Wasp allows for the transition between scales and adopts the principles of discrete design used for the generation and replication of spatial components. Its working logic is based on a set of aggregation procedures that allow the creation of specific structures from a combination of different modules. Wasp components allow users to manipulate geometrical inputs with the basic information needed to join them as parts of a whole (Figure 1).

Conceiving a connection set in Wasp generates topological graphs of parts, enabling new joint variations, each consisting of optional aggregation rules informing the model growth pattern into a variety of modules (Rossi 2017). In this study, different types of aggregation are tested to meet diverse spatial requirements. Wasp rules are used to construct an aggregation system consisting of one or more geometries: points, which are created on this geometry, joint vectors and virtual planes. Wasp allows designers to build conditions from a rule generator component by assigning various aggregation types. If no additional grammar rules are provided in the input, the component generates rules between connections of the same type. The main Wasp procedure involves the use of the following components: modules, anchor points, connection directions, connection planes and generators.

1. A module is the geometric unit that forms the first step of aggregation, with the points and lines assigned to this unit. The first module or modules created contain a system of connections.

2. Anchor points are then created by the point assigned to the module and provide the base for the direction of the aggregation.

3. With the connection direction, the module considers the point assigned to it as the starting point. It contains information on which direction the summation will be multiplied by the assigned line.

4. The connection planes determine from which surfaces the summation will continue according to the point and line assigned to the module.

5. The generator allows the aggregation to work additively by allowing modules to incrementally multiply on themselves in accordance with the assigned rules. When the number of parts of the aggregation increases, the generator adds new parts to the existing module without recalculation.

## Stochastic Aggregation with Visual Programming Components

Within the scope of this study, different typologies of buildable parts are categorised and then converted into 2D diagrams. Open and closed geometries are selected from these categories and transformed into modules with 3D equivalents (Figure 2).

These modules are replicated in an algorithm that allows stochastic aggregation by assigning the rules, as shown in the outcomes above. In the diagram, the parts are typologized according to their volumetric properties and transformed into 3D representations with the help of modules assigned to these typologies. The spatial impact of the number of pieces is diagrammed to show the spatial logic behind it (Figure 3).

For each collection, a new set is constructed that relies on the main module parameters. In this way, the modules are positioned according to the positions of their connections in space. All connection parts, such as the points, lines and surfaces of the first module, turn into repeating connection elements in the next stage. The virtual plane of the second geometry matches the plane of the first geometry and is multiplied by the transformation of the second geometry (Figure 4). Subsequently, a collision check is organised using the Wasp algorithm to prevent overlapping and nesting between the parent geometries and the new geometries that arise from the virtual planes of these geometries.

Stochastic processes involve randomness and probability, while aggregation refers to the combination of elements. The specific case of this study involves the use of probabilistic methods to combine and replicate architectural and structural elements. Stochastic aggregation can produce more adaptive and diverse solutions that can accommodate various scenarios. In this sense, the assigned parameters enable product outputs to vary under different conditions with this form of generation. The stochastic search continues superficially until the fourth phase of this experiment setup, at which point even the AI begins to perform aggregation over a single geometrical output without defining specialised rules, but certainly another layer of aggregation.

Wasp and stochastic aggregation provide a process to approximate the target geometry, but at the same time, the myriad possibilities and repetitive production at some point can limit the final product decisions. The main limitation of the manufacturing approach presented by Wasp is that the design is constructed in a way that is predetermined either by using the smallest-part-to-largest aggregation and stochastic aggregation or by a directional domain based on master modules or generated density. These methods impose limitations on the application of Wasp in a flexible design process, whereby heuristic modelling is often at odds with rule-based processes (Johns, 2014). Wasp also has limitations since it relies on modules defined by the designer during the design process and therefore does not offer a ready-made module library. This requires extra preparation before making alternative productions. The code for the combination of modules needs to be elaborated upon at the beginning. Since the rules are defined according to the geometries of the module, changing these rules during the process may have undesirable consequences since it may disrupt the integrity of the geometric structures and the consistency of the design. In this context, it is important for the effectiveness of the process to define the rules at the beginning and to update them with minor interventions, when necessary, with reference only to the outputs. The plugin also has a limited capacity to reproduce complex geometries and organic forms.

## Geometrical Rules with Textual Input

In the next step, three main geometries are designed: horizontal semi-open modules (slabs), vertical columns and connecting ramp elements. The top and bottom surfaces of each element are transformed into connection planes through assigned points and lines. Geometries only reproduce through these points and planes in Wasp's most basic form of aggregation (Figure 5). In this study, collections can be seen as an example of one of Wasp's limitations. These collections increase the variety of products but produce aggregates that do not correspond to the desired hyperspatial organisations.

We can describe these aggregation experiments based on geometric connection rules as a reference aggregation before moving on to code-based rules. In the aggregation based only on geometric connections, deviations from the connection rules may be seen, requiring the intervention of new text-based rules (Figure 6). At this point, the rules entered as text into a panel component work as a dataset and turning points that change the course of the collection. Some constraints were implemented through the rules assigned to the code to redefine the mode of production and generate spaces that corresponded to the envisioned hyperspatial organisation. These constraints regulate the spatial relationships between geometrical elements, as follows:

1. The upper surface of the slab aligns with the base plane of the column.
2. The bottom of the slab aligns with the top plane of the column.
3. The side of the floor connects to the bottom of the ramp.
4. The top of the ramp aligns with the side of the floor.
5. The top of the ramp is positioned directly above the bottom of the ramp within the collection.

### Introducing Spatio-Structural Design Criteria

The following phase regulates the outputs by providing manual interventions at the designer's disposal based on two sets of structural and spatial design criteria. The first criteria set contains (1) the number of modules to match the number of parts entered; (2) horizontal and vertical elements to join the hierarchy; and (3) no overlapping modules. In addition, the following spatial criteria were considered: (1) gaps required for light; (2) space dimensions; (3) horizontal and vertical circulation; and (4) potential relationships between spaces.

## Algorithm 1. Pseudo-code of structural criteria.

```
// (1) Ensure the number of modules matches the number of parts entered
IF num_modules != num_parts_entered
  THEN ADJUST num_modules TO MATCH num_parts_entered
END IF
// (2) Ensure horizontal and vertical elements join hierarchical
FOR EACH horizontal_element IN design
  FOR EACH vertical_element IN design
    IF intersection(horizontal_element, vertical_element) IS valid_based_on_spatial_logic
      THEN CONNECT(horizontal_element, vertical_element)
    ELSE
      ADJUST positioning OR modify connection parameters
    END IF
  END FOR
END FOR
// (3) Ensure modules do not overlap
FOR EACH module_A IN modules
  FOR EACH module_B IN modules
    IF module_A != module_B AND overlap(module_A, module_B)
      THEN RESOLVE_OVERLAP(module_A, module_B)
    END IF
  END FOR
END FOR
```

## Algorithm 2. Pseudo-code of spatial criteria.

```
// (1) Ensure gaps are present for light
FOR EACH space IN design
  IF light_gap(space) IS NOT present
    THEN ADD gap(space) BASED ON lighting_requirements
  END IF
END FOR
// (2) Check space dimensions
FOR EACH space IN design
  IF dimensions(space) NOT WITHIN allowable_range
    THEN ADJUST dimensions(space) TO FIT spatial_constraints
  END IF
END FOR
// (3) Ensure horizontal and vertical circulation is maintained
FOR EACH circulation_path IN design
```

```
  IF circulation_path IS BLOCKED
    THEN MODIFY layout TO CREATE clear
    circulation
  END IF
END FOR
// (4) Establish potential relationships
between spaces
FOR EACH space_A IN design
  FOR EACH space_B IN design
    IF relationship(space_A, space_B) IS
    required
      THEN ENSURE adjacency OR define
      transition_condition(space_A,
      space_B)
    ELSE
      MAINTAIN functional_separa-
      tion(space_A, space_B)
    END IF
  END FOR
END FOR
```

The generated module combinations that meet these structural and spatial design criteria are subsequently transferred to the optimisation process by manual intervention to increase spatial diversity. Based on material and production knowledge of the generated modules as state-of-the-art architectural elements, form-based functional manipulations are informed in the process. These manipulations enable spatial diversity in line with an assumed layer of optimisation (Figure 7).

These interventions involve creating contours at a 45-degree angle with 200 cm intervals along the solid units, which are then deepened using the extrude curve command to add spatial depth and transform the contours into three-dimensional forms. The resulting large-scale modules are finalized through manual interventions that enhance spatial possibilities, including overlapping, mirroring, subtracting mass to create light gaps at material joints, integrating translucent materials and modifying the modules in a way that allows them to multiply for later articulation (Figure 8).

## Informing Text-Based Design Fiction into the Process

In this last phase of the experimental setup, the AI interventions are explained. With the transformation of each created piece into a relational design element by gaining tectonic aspects, it becomes a feasible speculation that simultaneously produces architecture at different scales. At this point, the results of this study suggest that designing with discrete elements directly contributes to the efficiency of a new type of thinking based on SynDI, which uses AI in terms of object-based definitions and image segmentation.

Thanks to the integrated work regarding methods for generating visuals from text and visuals from visuals through AI, it becomes possible to evaluate the design idea in different design fictions. We produce dispersed process management at many different scales and points without relying on only a single representation of the design. In this way, the transformation that the project undergoes regarding changing the material, the type of user, the space in it and the way of production – that is, the fiction of the project – can be quickly multiplied with powerful representations. The contributions of a model created with such a synergistic understanding of design intelligence can be explained by the following concepts (Table 1).

At this stage, different scenarios suitable for the context of the project are entered into the AI, and how the design will change and look in line with these scenarios is simulated quickly and effectively. The AI simulates how the scenarios determined by the designer in the optimisation process will look, either together or separately, and offers the designer new possibilities beyond the existing structure. Figure 9 shows the images before AI intervention (left), the concepts entered as text and the images interpreted with these texts (right). The original visuals were reinterpreted with an AI tool that generates visuals from visuals with

**Table 1.** Expectations and results according to the concepts using the synergistic intelligence model. *Source: Authors.*

| Concept | Result | Explanation |
|---|---|---|
| Design process/ iteration | Enhanced decision-making | Possibility of making continuous adjustments using the feedback mechanism to improve the design |
| Creativity | Augmented space | Advancing the designer's creativity through the multiplication of design variations |
| Functionality | Adaptable solutions | Ability to adapt to changing conditions over time. |

specific text and concept inputs. Thanks to the AI, which quickly visualises different scenarios determined by the designer, each visual provides the opportunity to explore the possibilities of a particular scenario on a spatial plane. In the first row, the theme of 'modular cyber city' includes units surrounded by neon colours and the production processes of robotic arms. In the middle row, spatial organisation proposals are presented with AI-supported health monitoring systems, robot doctors and drone ambulances. In the bottom row, a hyper-realistic complex design is envisioned, representing a complex urban setting with vertical tanks and production pipes. The diagram illustrates the potential of AI-assisted design processes to produce multilayered and functional spaces from conceptual text inputs. In this respect, the AI becomes a tool that increases the creativity of the designer through the production of multiple space forms and circulation alternatives between spaces. A building is expected to be produced in such a way that it exists in a futuristic city. In this respect, a flexible process that can observe the changes in the physical conditions in which the building is located and the structure has been created.

As described throughout this paper, the concept of SynDI in the context of HCI in architectural design processes represents advanced computational design thinking. SynDI seeks to seamlessly integrate human creativity and intuition with the analytical and computational capabilities of AI. This integration allows for a holistic approach, leveraging the strengths of both human designers and intelligent algorithms. It aims to enhance decision-making in architectural design.

By utilising AI, which can process vast amounts of data and generate insights, designers can make more informed choices and optimise factors such as sustainability, efficiency and aesthetics (Castro Pena et al. 2021). The use of AI in architectural design processes can significantly improve efficiency (Almusaed and Yitmen 2023). AI algorithms can rapidly iterate through numerous design possibilities based on specified criteria, providing designers with a range of options to consider and refine (Tamke et al. 2018). In this context, SynDI augments irreplaceable human creativity within new digital workflows. Somehow, AI tools can assist in generating innovative design solutions, acting as creative partners rather than substitutes for human ingenuity. However, implementing SynDI requires addressing technical challenges, such as developing AI systems capable of understanding complex design preferences. Additionally, ensuring seamless and user-friendly interaction between architects and AI tools is crucial. Issues related to bias in AI algorithms, accountability for design outcomes and the ethical use of AI-generated designs require careful attention. As for user behaviour changes, designers may face a learning curve in adopting AI tools effectively. A full-fledged SynDI design system may provide sufficient flexibility and adaptability in diverse design contexts. Systems that can evolve with changing project requirements, design paradigms and technological advancements will be more resilient and sustainable. In conclusion, the concept of SI in architectural design, integrating human and artificial intelligence, holds immense promise for advancing the field. While challenges exist, the potential benefits in terms of enhanced creativity, efficiency and decision-making make it a compelling avenue for exploration in the ever-evolving intersection of architecture and technology.

## Discussion and Conclusion

Computational design environments allow architects to generate, explore and develop more design solutions more efficiently compared to linear processes. This makes the design process faster, more effective and more understandable by enabling continuous feedback and adjustments at the initial stages of design (Bhooshan 2017). In this context, a workflow was developed outlining a systematic approach to architectural design through the integration of new modes of human–computer interaction, emphasising the iterative nature of design processes (Figure 10). Utilising parametric computing tools such as Rhinoceros and Grasshopper, the workflow enables different data to transform into customised 3D geometries as inputs of the parametric design process. Based on twitching textual input to generate 3D modules, the proposed workflow aims to expand the design space with diverse module outputs and enhance architect–computer communication.

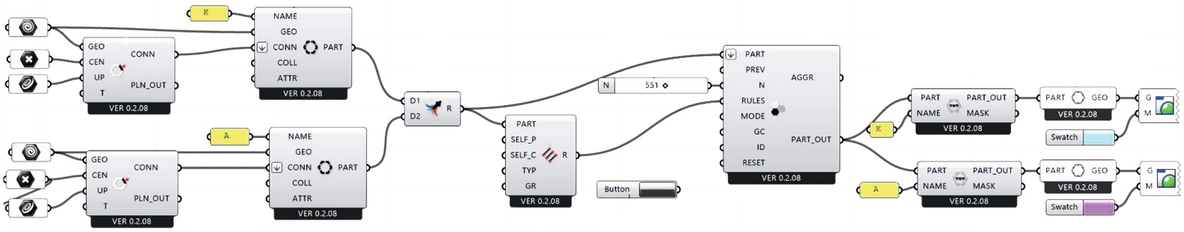
The proposed framework in this study also aims to enhance the optimisation and improvisation of desired design modules by fostering a dynamic feedback loop between the computational layer and the human layer, leading to a new definition of SynDI. The inclusion of AI-based analysis, which has recently been one of the most discussed topics in architecture, enhances the creative process by providing alternative design scenarios, thus expanding the scope of design exploration. The optimisation workflows attempted in this work address multiple performance objectives simultaneously rather than focusing on only one performance objective, while the architect in control can choose from multiple high-performance solutions.

The workflow starts with data collection and data processing and then produces alternative design solutions in 3D. The model continues with optimisation processes for an effective design process and is completed with visualisation and presentation in the next stage. The iterative feedback loop between the AI and the designer provides a dynamic contribution with the ability to develop suggestions and generate variations at each layer of the design. The feedback obtained at each stage of the design can continuously improve the process. This model, in which optimisation processes, in particular, come to the forefront, enables the rapid and flexible generation of various design alternatives in design problems with different scales and requirements. In this way, it can be adapted to different projects, revealing, in general terms, a repeatable process. In this sense, this research's experiments add to previous research showing balancing solutions to conflicting design objectives (e.g. Feng et al. 2024). The iterative feedback mechanism outlined in the present work contributes to innovative design solutions by addressing possible errors and failures. This methodological approach around the speculative concept of SynDI highlights the importance of collaboration between human intelligence and computational capabilities, leading to the development of highly optimised, context-relevant and visually engaging architectural designs.

In terms of the application of discrete architecture principles, digital compositions derived from small parts carry clues from the data from which they are produced. The parts are differentiated in terms of the aspects that guide the emergence of the content. At the same time, they do not completely break away from their function on the plane from which they were detached and present a coherent sequence within themselves. By creating scripts, the designer can generate answers to design questions, such as the automation of diverse and complex tasks, quickly experiment with a wide range of design possibilities and create new forms that may be difficult to produce using traditional methods (Fernández-Álvarez and López-Chao 2023). Architectural design has undergone a transformative change process with the integration of digital tools. The combination of new methods has opened new and different avenues for creativity and efficiency, allowing more time for conceptual thinking and less time for repetitive tasks. The design and representation techniques used by architects are undergoing a transformation through new software and technologies. Unlike traditional design methods, parametric design automates repetitive tasks and provides feedback on structural and environmental performance from the beginning of the design process, thus allowing for a more flexible and interactive approach (Haidar et al. 2019). Similarly, architectural visualisation and space generation with AI is one of these transformations. AI design generation tools allow architects and designers to increase productivity in parts of their daily work.

AI serves to expand the limits of a designer's creativity and forms of representation. With the data entered as text, this work aimed to explore additional design fictions conditioned by movement patterns within the space with hyperspatial features. In this way, new and alternative final products were obtained quickly without going back on the final product outputs. At every stage of the process, a feedback mechanism was created between the designer and the computer. The mutual interaction and production means allowed us to imagine new digital production processes incorporating the trio of the designer, the geometry and the computer. In other cases in which a designer works with AI, design alternatives can be automated based on the parameters specified by the AI, resulting in adaptable and flexible design processes. However, in this research, a synergistic understanding of the digital design workflow, in which the designer remains in control and uses AI as a tool to expand the boundaries of thought and representation forms, is proposed.

As a joint production by designers and computers, contemporary architecture requires us to deepen a new understanding of the integration of different applications, which is associated with the concept of SynDI in this paper. A SynDI process in which AI and the designer produce together reveals the potential of fast thinking in contrast to the general acceptance of learning concepts that prioritise slow thinking (e.g. Kannengiesser and Gero 2019). In this context, production with the SynDI model in architectural design processes can provide fast, flexible design alternatives that allow for interdisciplinary collaboration with the multiplicity of products and the coexistence of different forms of representation. Against the risk that parametric design may restrict architects' creative features by reducing them to repetition after a certain point, the SynDI model will pave the way for looking at the process from a wider perspective and exploring new areas by pushing the creative process out of the design field.



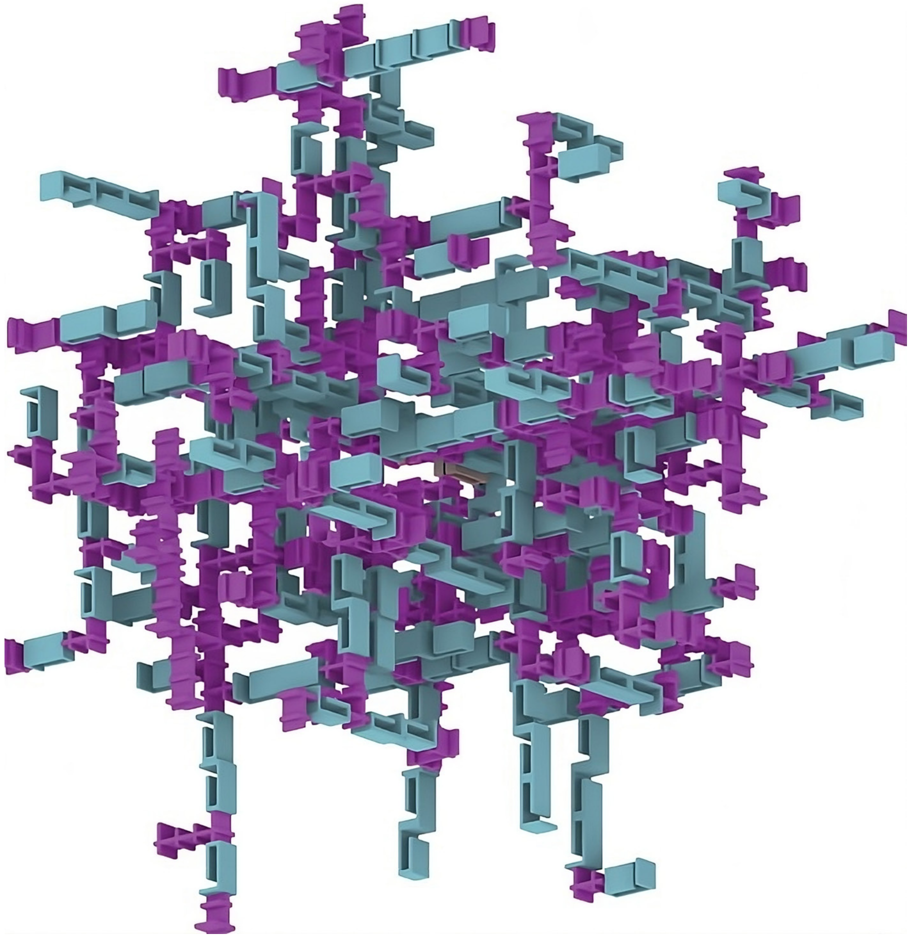
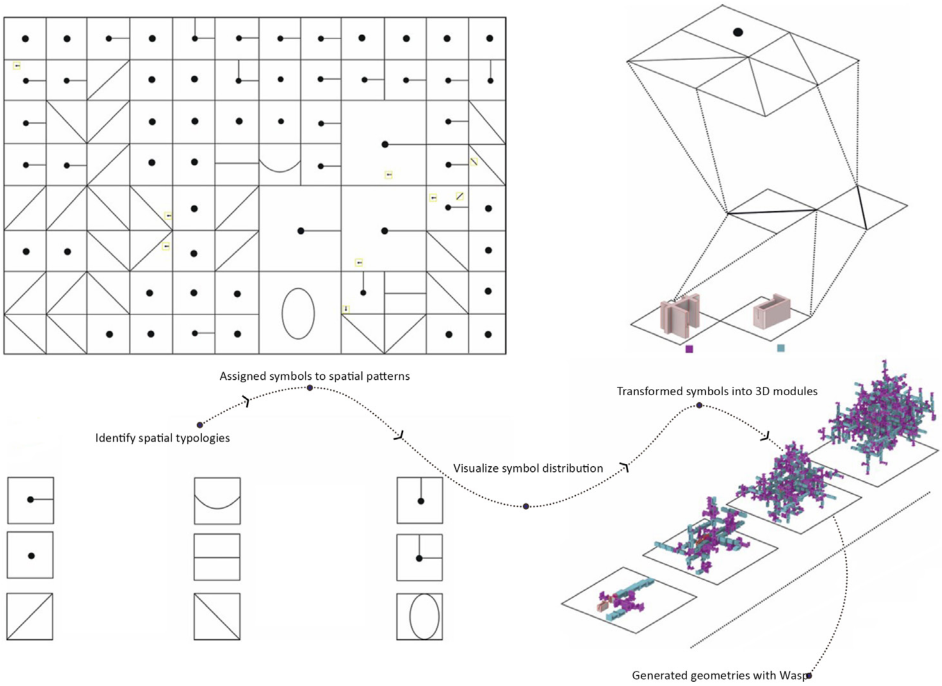


**Figure 1 (top).** The code conceived in Grasshopper, one of the most widely used visual programming languages in the field of architecture. *Source: Authors.*

**Figure 2 (bottom).** Two-module multiplication with stochastic addition. *Source: Authors.*



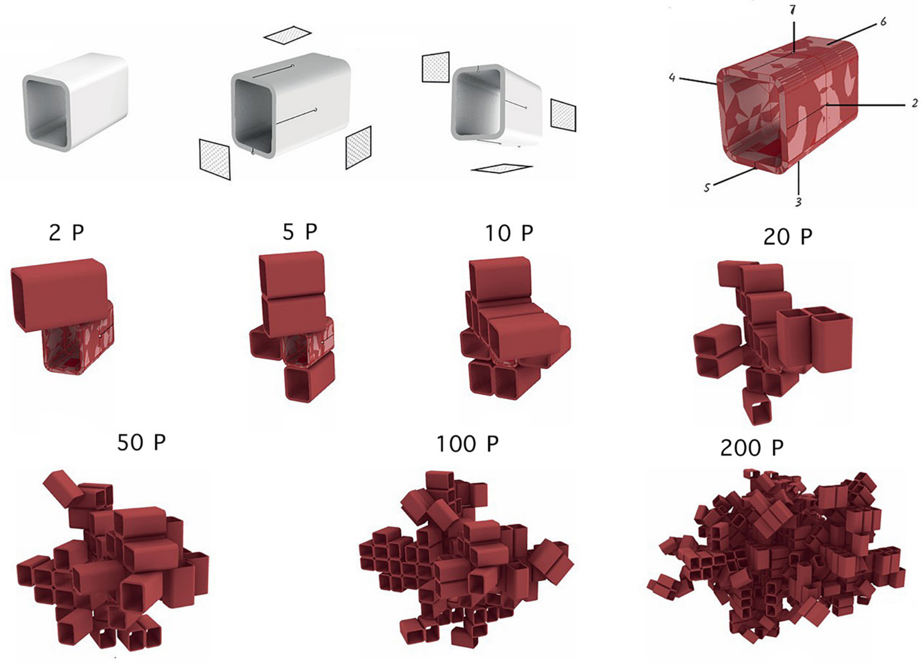


**Figure 3 (top).** Diagram of an aggregation trial. *Source: Authors.*

**Figure 4 (bottom).** Collection attempt with repetition of the same module. *Source: Authors.*

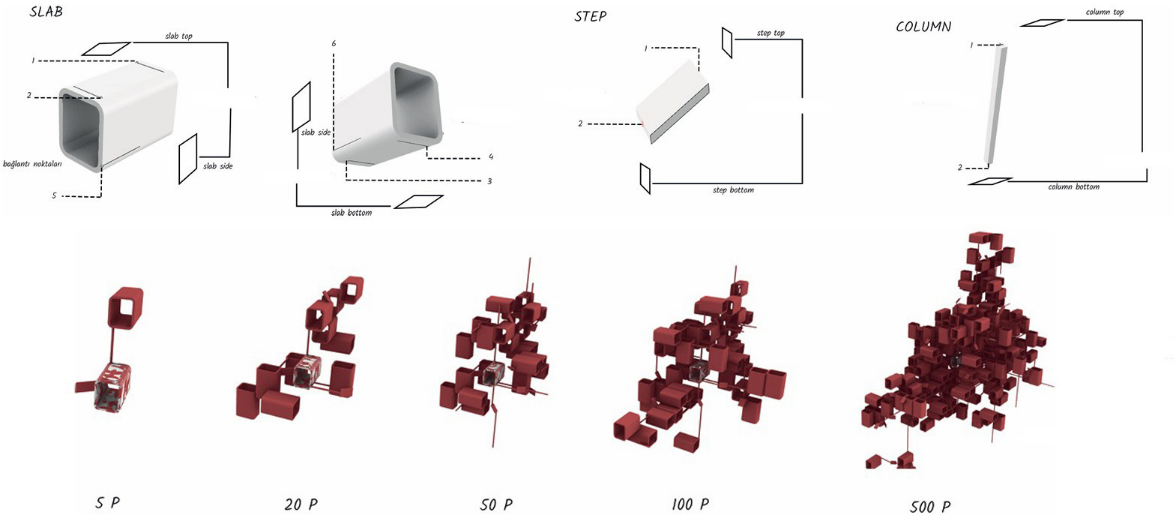
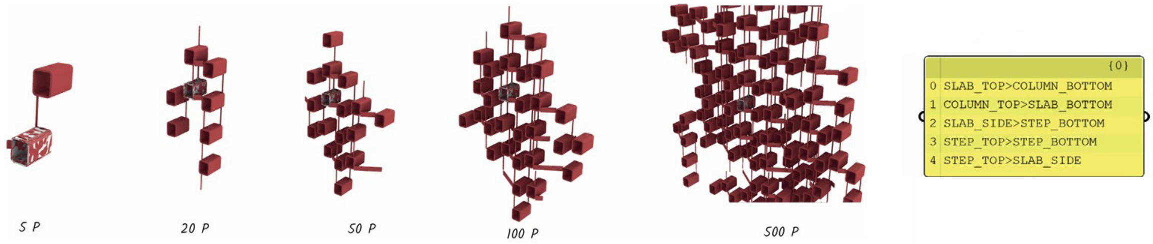


**Figure 5.** Rule-free generation of a three-part module.
*Source: Authors.*



```
0 SLAB_TOP>COLUMN_BOTTOM
1 COLUMN_TOP>SLAB_BOTTOM
2 SLAB_SIDE>STEP_BOTTOM
3 STEP_TOP>STEP_BOTTOM
4 STEP_TOP>SLAB_SIDE
```

**Figure 6.** Regular production of a three-part module.
*Source: Authors.*

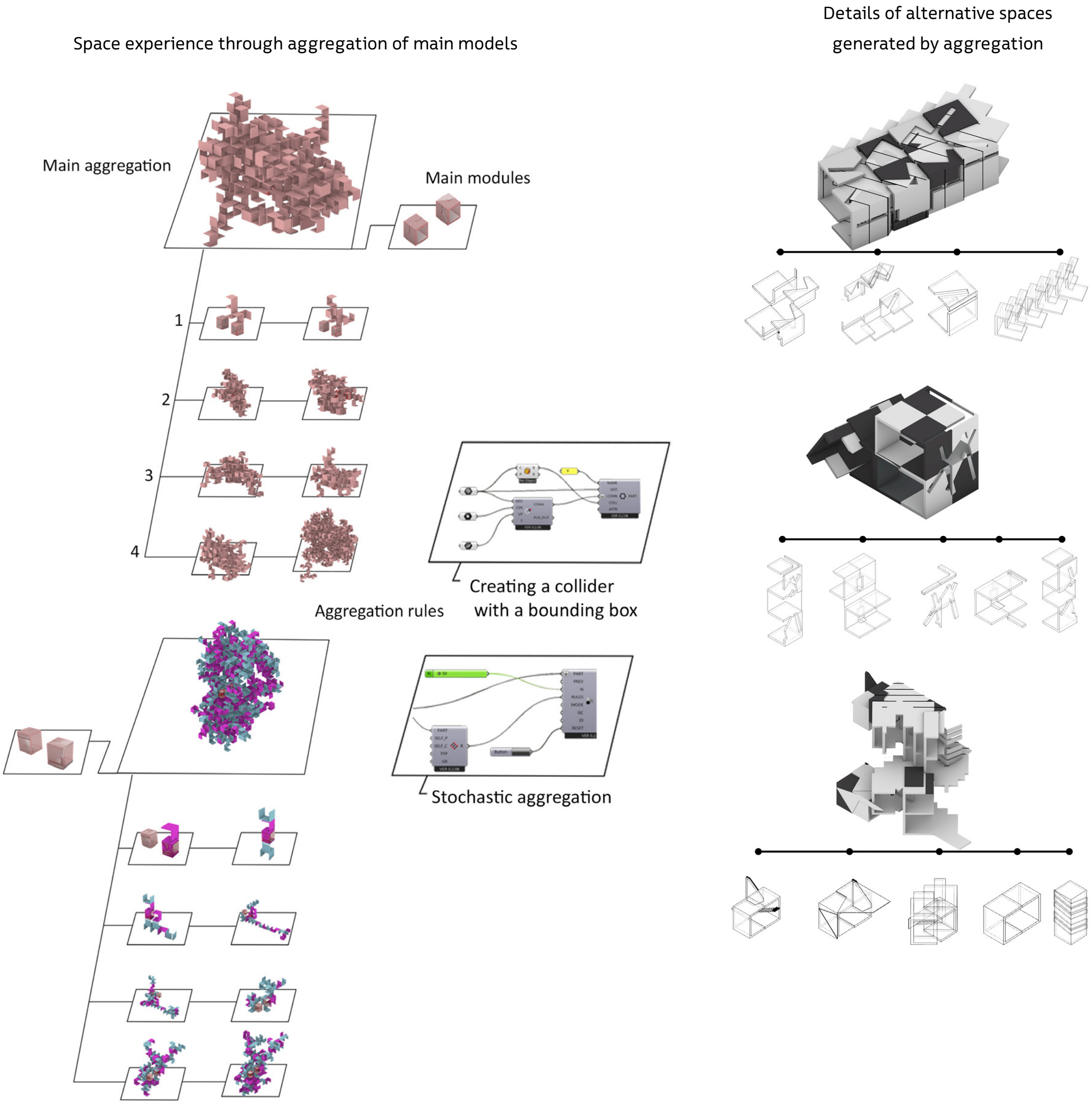


**Figure 7.** Manual optimisation of modules.

*Source: Authors.*

1. Main space

2. Added contour lines

3. Dividing areas extruding in different directions

4. Scale, delete, and material add

5. Spaces reproduced with mirror

6. Joining details that allow for expansion.

**Figure 8.** Experimenting with spaces from modules.
*Source: Authors.*

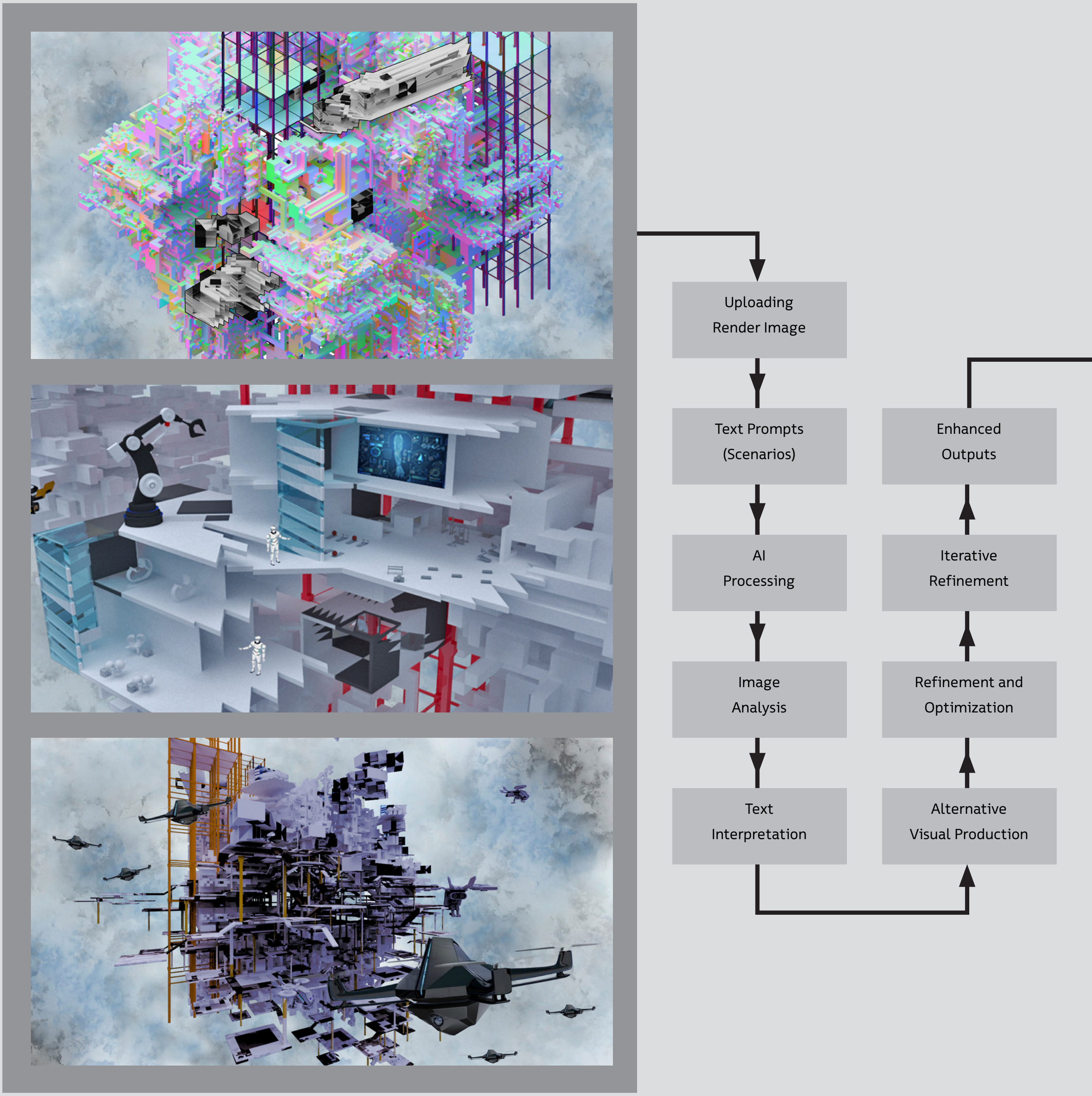
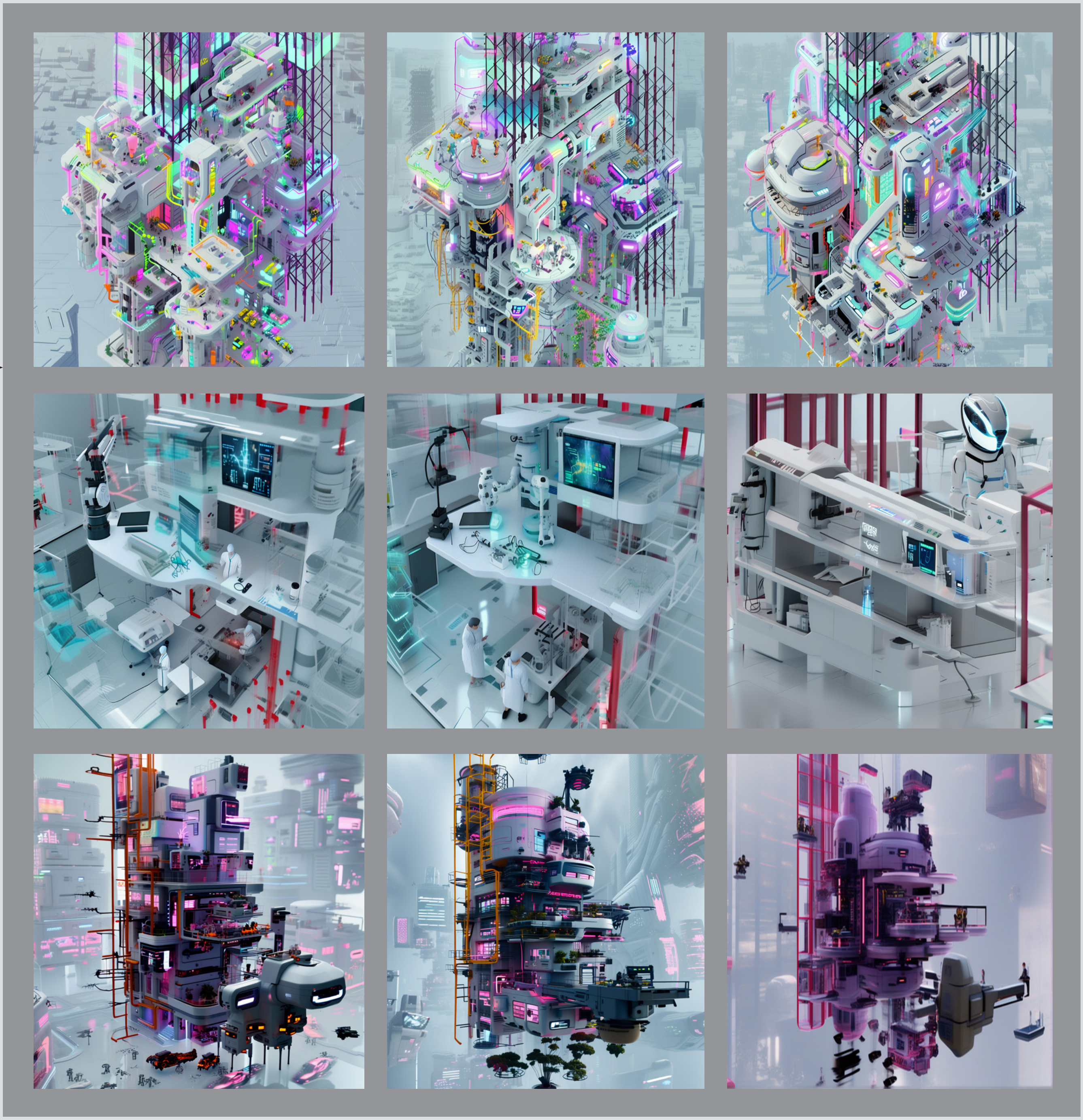


**Figure 9 (this page and next).** Reproducing images with artificial intelligence. *Source: Authors.*



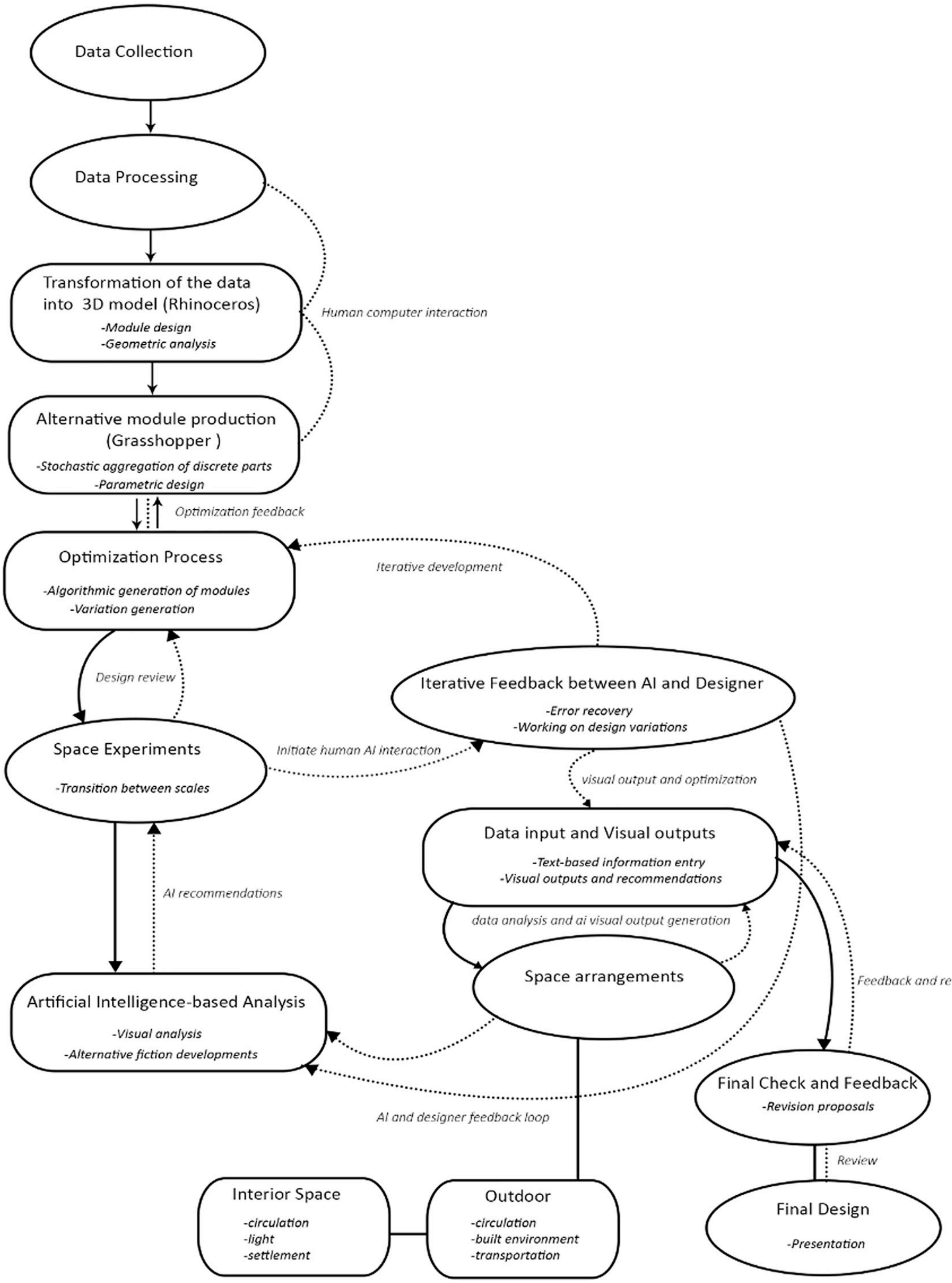


**Figure 10.** Flowchart of the design framework. *Source: Authors.*

## Bibliography

Almusaed, Amjad, and Ibrahim Yitmen. 2023. "Architectural Reply for Smart Building Design Concepts Based on Artificial Intelligence Simulation Models and Digital Twins." *Sustainability* 15 (6): 4955. https://doi.org/10.3390/su15064955.

Bernstein, Phil. 2022. *Machine Learning: Architecture in the Age of Artificial Intelligence*. Routledge.

Bhooshan, Shajay. 2017. "Parametric Design Thinking: A Case-Study of Practice-Embedded Architectural Research." *Design Studies* 52 (September): 115–43. https://doi.org/10.1016/j.destud.2017.05.003.

Bibri, Simon Elias, Jeffrey Huang, Senthil Kumar Jagatheesaperumal, and John Krogstie. 2024. "The Synergistic Interplay of Artificial Intelligence and Digital Twin in Environmentally Planning Sustainable Smart Cities: A Comprehensive Systematic Review." *Environmental Science and Ecotechnology* 20 (July): 100433. https://doi.org/10.1016/j.ese.2024.100433.

Campo, Matias del. 2024. *Diffusions in Architecture: Artificial Intelligence and Image Generators*. Wiley. https://doi.org/10.1002/9781394191802.

Carpo, Mario. 2013. *The Digital Turn in Architecture 1992–2012*. Wiley. https://doi.org/10.1002/9781118795811.

Carpo, Mario. 2017. *The Second Digital Turn: Design Beyond Intelligence*. MIT Press.

Castro Pena, M. Luz, Adrián Carballal, Nereida Rodríguez-Fernández, Iria Santos, and Juan Romero. 2021. "Artificial Intelligence Applied to Conceptual Design. A Review of Its Use in Architecture." *Automation in Construction* 124 (1): 103550. https://doi.org/10.1016/j.autcon.2021.103550.

Chen, Wei, Mingyu Yu, and Jia Hou. 2024. "Synergistic Relationship, Agent Interaction, and Knowledge Coupling: Driving Innovation in Intelligent Construction Technology." *Buildings* 14 (2): 542. https://doi.org/10.3390/buildings14020542.

Claypool, Mollie. 2020. "Architecture Goes Digital: The Discrete." In *The Architecture Machine: The Role of Computers in Architecture*, edited by Teresa Fankhänel and Andres Lepik. Birkhäuser.

Cross, Nigel. 2004. "Expertise in Design: An Overview." *Design Studies* 25 (5): 427–41. https://doi.org/10.1016/j.destud.2004.06.002.

Eastman, Charles M. 1972. "Adaptive-Conditional Architecture." Paper presented at the Design Research Society's Conference, Pittsburgh, PA.

Feng, Xingyue, Han Zhang, and Shaonong Wei. 2024. "Automation in Architectural Design: Optimization and Decision-Making in Interference Design." *Architectural Science Review* 67 (5): 415–36. https://doi.org/10.1080/00038628.2024.2320096.

Fernández-Álvarez, Ángel. J., and Vicente López-Chao. 2023. "Drawing, Scripting, Prompting. A Critical Approach from Architectural Graphics." *Diségno* (13): 143–52. https://doi.org/10.26375/disegno.13.2023.16.

Friedman, Yona. 1972. "Information Processes for Participatory Design." Paper presented at the Design Research Society's Conference, Pittsburgh, PA.

Gero, John S., and Udo Kannengiesser. 2012. "Representational Affordances in Design, with Examples from Analogy Making and Optimization." *Research in Engineering Design* 23 (3): 235–49. https://doi.org/10.1007/s00163-012-0128-y.

Gero, John S., and Udo Kannengiesser. 2014. "The Function-Behaviour-Structure Ontology of Design." In *An Anthology of Theories and Models of Design*, edited by Amaresh Chakrabarti and Lucienne T. M. Blessing. Springer. https://doi.org/10.1007/978-1-4471-6338-1_13.

Hageman, B. (n.d.). "Wasp++." Accessed August 8, 2024. https://www.food4rhino.com/en/app/wasp-0.

Haidar, Adonis, Jason Underwood, and Paul Coates. 2019. "Smart Processes for Smart Buildings: 'Sustainable processes', 'Recyclable processes' and 'Building seeds' in parametric design." *Architectural Engineering and Design Management* 15 (5): 402–29. https://doi.org/10.1080/17452007.2018.1564645.

Haken, Hermann. 1987. "Synergetics: An Approach to Self-Organization." In *Self-Organizing Systems: The Emergence of Order*, edited by F. Eugene Yates, Alan Garfinkel, Donald O. Walter, and Gregory B. Yates. Springer. https://doi.org/10.1007/978-1-4613-0883-6_22.

Haslop, Blaire, Marc Aurel Schnabel, and Serdar Aydin, 2017. "Digital Decay." *International Journal of Parallel, Emergent and Distributed Systems* (32): S82–S88. https://doi.org/10.1080/17445760.2017.1390092.

Huang, Lee-Su and Spaw, Gregory. 2023. "Augmented Reality Assisted Robotics: Tube Bending." Paper presented at the 43rd Annual Conference for the Association for Computer Aided

Design in Architecture (ACADIA), Denver, CO. https://papers.cumincad.org/cgi-bin/works/paper/acadia23_v2_340.

Jahn, Gwyllim, Cameron Newnham, and Nick van den Berg. 2022. "Augmented Reality for Construction from Steam Bent Timber." Paper presented at the 27th CAADRIA Conference, Sydney. https://doi.org/10.52842/conf.caadria.2022.2.191.

Johns, Ryan Luke. 2017. "Augmented Materiality: Modelling with Material Indeterminacy." In *Fabricate 2014: Negotiating Design & Making*, edited by Fabio Gramazio, Matthias Kohler, and Silke Langenberg. UCL Press. https://doi.org/10.2307/j.ctt1tp3c5w.30.

Jordanous, Anna. 2016. "Four PPPPerspectives on Computational Creativity in Theory and in Practice." *Connection Science* 28 (2): 194–216. https://doi.org/10.1080/09540091.2016.1151860.

Kannengiesser, Udo, and John S. Gero. 2019. "Design Thinking, Fast and Slow: A Framework for Kahneman's Dual-System Theory in Design." *Design Science* 5 (1): e10. https://doi.org/10.1017/dsj.2019.9.

Leach, Neil. 2022. *Architecture in the Age of Artificial Intelligence: An Introduction to AI for Architects*. Bloomsbury.

Lin, Han-Ting, and June-Hao Hou. 2022. "Exploring the Topological System of Dougong." Paper presented at the International Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA), Sydney. https://doi.org/10.52842/conf.caadria.2022.1.667.

Lynn, Greg. 1999. *Animate Form*. Princeton Architectural Press.

Qian, Lena, and John S Gero. 1996. "Function–Behavior–Structure Paths and their Role in Analogy-Based Design." *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 10 (4): 289–312. http://dx.doi.org/10.1017/S0890060400001633.

Ploennigs, Joern, and Markus Berger. "Ai Art in Architecture." *AI in Civil Engineering* 2, no. 8 (2023). https://doi.org/10.1007/s43503-023-00018-y.

Retsin, Gilles. *Discrete: Reappraising the Digital in Architecture. Wiley*, 2019.

Rossi, A. (n.d.). "Wasp – A Combinatorial Toolkit for Discrete Design." Accessed August 8, 2024. https://www.food4rhino.com/en/app/wasp.

Rossi, A., and Tessmann, O. 2017. "Geometry as Assembly – Integrating Design and Fabrication with Discrete Modular Units." Paper presented at the 35th International Conference on Education and Research in Computer Aided Architectural Design in Europe (eCAADe), Rome. https://doi.org/10.52842/conf.ecaade.2017.2.201.

Schumacher, Patrik. 2008. "Parametricism as style – Parametricist manifesto." Patrik Schumacher's Online Blog. Accessed on August 8, 2024. https://patrikschumacher.com/parametricism-as-style-parametricist-manifesto/.

Sterk, Tristan. 2006. "Responsive Architecture: User-Centered Interactions within the Hybridized Model of Control." In *The Architecture Co-Laboratory: Game, Set and Match II. On Computer Games, Advanced Geometries, and Digital Technologies*, edited by Kas Oosterhuis, and Lukas Feireiss. Episode Publishers.

Tamke, Martin, Paul Nicholas, and Mateusz Zwierzycki. 2018. "Machine Learning for Architectural Design: Practices and Infrastructure." *International Journal of Architectural Computing* 16 (2): 123–43. https://doi.org/10.1177/1478077118778580.

Terzidis, Kostas. 2003. Expressive Form: A Conceptual Approach to Computational Design. Routledge.

Yu, Boyuan, and Lei Zhang. 2021. "A Virtual Model Generation Based on Grasshopper-Wasp." Paper presented at the 2nd International Conference on Intelligent Design (ICID), Xi'an. https://doi.org/10.1109/ICID54526.2021.00052.

## Bio

**Serdar Aydin** s an architect, researcher and designer whose work navigates between digital culture, design intelligence and participatory systems. His cross-continental trajectory—spanning Turkey, China, the UK, and Aotearoa New Zealand—informs a pluralistic and critical approach to computational design. His research engages parametric and algorithmic methods, virtual reality, smart city and interaction design, centering on the co-evolution of human and machine agency rather than detached digitalism. He holds a master's from the University of Liverpool and a PhD on gamified, hybrid environments shaped by user-generated content. Aydin received the Design Research Society Award (2014), the YoungCAADRIA Award (2015), was honored Second Prize in Turkey's Natural Stone Design Competition (2021), and worked with Zaha Hadid Architects on the parametric articulation of Changsha Meixihu International Culture and Arts Centre. He is currently Associate Professor at Faculty of Engineering and Architecture, Mardin Artuklu University, where he directs Mimfab Centre of Research on Architectural Technology and serves as editor-in-chief of the International Journal of Mardin Studies.

**Beryar Zübeyde Balaban** graduated from Mardin Artuklu University, Department of Architecture in 2021. Her research interest is on computational design, digital fabrication, and the digitalisation of cultural heritage, particularly in the context of interactive virtual environments. To deepen this academically, she started to study a masters' degree of Architectural Design Computing program at Istanbul Technical University. Currently, she is conducting research on the digital interpretation and virtual experiences of the ancient city of Dara, utilizing augmented reality technologies to create immersive, educational experiences for visitors.