REPUBLIC OF TÜRKİYE
ALTINBAŞ UNIVERSITY
Institute of Graduate Studies
Architecture Department

IMPLEMENTING A GENERATIVE AI MODEL IN THE ARCHITECTURAL CONCEPT DEVELOPMENT PHASE: CONTEXT-BASED EMPIRICAL RESEARCH

Zeyd ELABBASI

Master's Thesis

Supervisor

Asst. Prof. Dr. Çağda ÖZBAKİ

İstanbul, 2025

IMPLEMENTING A GENERATIVE AI MODEL IN THE ARCHITECTUAL CONCEPT DEVELOPMENT PHASE: CONTEXT-BASED EMPIRICAL RESEARCH

Zeyd ELABBASI

Architecture Department

Master’s Thesis

ALTINBAŞ UNIVERSITY

2025

The thesis titled "IMPLEMENTING A GENERATIVE AI MODEL IN THE ARCHITECTURAL CONCEPT DEVELOPMENT PHASE: CONTEXT-BASED EMPIRICAL RESEARCH" prepared by ZEYD ELABBASI and submitted on 04/09/2025 has been **accepted unanimously** for the degree of Master of Science in Architecture.

Asst. Prof. Dr. Çağda ÖZBAKİ

Supervisor

Thesis Defense Committee Members:

Asst. Prof. Dr. Çağda ÖZBAKİ	Department of Architecture, Altınbaş University	
Assoc. Prof. Dr. Derya Güleç ÖZER	Department of Architecture, İstanbul Technical University	
Asst. Prof. Dr. Can UZUN	Department of Architecture, İstanbul Technical University	

I hereby declare that this thesis meets all format and submission requirements for a Master's thesis.

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Zeyd ELABBASI

Signature

DEDICATION

I thank God firstly, I would also like to thank my family and friends, who have been always by my side in the most difficult periods, who always supported and motivated me throughout my journey, I would like to thank my thesis advisor, Asst. Prof. Dr. Çağda Özbaki, for guiding me during the process of conducting the research.

ABSTRACT

IMPLEMENTING A GENERATIVE AI MODEL IN THE ARCHITECTURAL CONCEPT DEVELOPMENT PHASE: CONTEXT-BASED EMPIRICAL RESEARCH

ELABBASI, Zeyd

M.Sc. Architecture Department, Altınbaş University,

Supervisor: Asst**.** Prof. Dr. Çağda ÖZBAKİ

Date: September/2025

Pages: 71

In the context of the continual evolution of advanced architectural design tools, driven by computational and artificial intelligence technologies, the practice of design can now benefit from data-driven and iterative design workflows that forward contextually relevant design outputs. Thus, computational and AI tools have huge potential to enhance the conceptual design processes in architecture, due to their special iterative workflow qualities.

Node-based interfaces of these tools act as a catalyst to allow more custom and interconnected workflows. Moreover, it is important to test new design potentials that these tools contribute to when used in parallel for the conceptual design process.

This thesis explores juxtaposing node-based computational design and AI tools in a dependent workflow process using Grasshopper3D and ComfyUI, within the framework of an experimental architectural concept development process, through a qualitative Research-through-Design methodology(RtD). By experimenting with their workflows, the aim is to formulate different design schemes that are contextually relevant but also achieved iteratively.

Visualizing abstract forms to test architectural forms relevant to four styles, which are *blob, biophilic, vernacular, and modern* architectural styles. Gathering the data about these outputs and analyzing the overall experiment.

Ultimately, this thesis achieves iterative, data-informed design schemes with detailed architectural features. Aiding contextually relevant but also architecturally detailed design schemes during the conceptual phase of the architectural design process.

Keywords: Architectural Design Context, Artificial intelligence, Conceptual Design, ComfyUI, Grasshopper3D.

TABLE OF CONTENTS

Pages

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

ABBREVIATIONS

AI	:	Artificial Intelligence
ANN	:	Artificial Neural Network
CNN	:	Convolutional Neural Networks
CLIP	:	Contrastive Language-Image Pre-training
DM	:	Diffusion Model
LLM	:	Large Language Model
LoRA	:	Low-Rank Adaptation
UI	:	User-Interface
VAE	:	Variational Auto Encoder

1. INTRODUCTION

Design is a manifestation of the human consciousness, logic, and reasoning, where abstract ideas are translated into functional outputs. Crafting effective artefacts that contribute to human-related fields is based on diverse thoughts and concepts brought together. This act of making sense of things and applying logical inferences to develop a product is intrinsic to the nature of human-related practices and workflows.

For developing architecture, the first step is to determine the physical configuration, driven by a design concept that tackles different contextual challenges, examined through detailed analytical and schematic studies, which requires logical inferences as well as diverse referential design inputs to consider. Ching (2015) elaborates that the initial phase of a design process should encapsulate the recognition of a problematic condition and propose a relevant solution.

Conceptual design approaches in architecture directly express an idea proposed by its designer through various methods, tools, and workflows. Architectural concepts visually elaborate how a design would relate to its relevant site area, climatic conditions, and environmental elements.

The nature of the conceptual design phases in architecture includes an iterative exchange of contextual data, including site analysis, site elements, and extensive design testing, before reaching the concrete design output.

The aforementioned are contextual parameters to be considered during the conceptual stage of developing architecture. The parameters are also linked with intangible theoretical and ideological values to establish the entity of a design, such as expressing the phenomena of a space and the meaning behind it.

Materials bind the conceptual design procedure to move in a specific design direction. Innovations constantly sparked further developments. Consequently, it affects the framework of the architectural practice. Throughout history, the evolution of tools has played a pivotal role in shaping the trajectory of architectural design.

The existing materials, technologies, tools, and workflows are relative to the time and space. They structure the framework and the method of the process, which also plays a critical role in affecting the output design. Thus, the applied tools and workflows should also operate to help users achieve these conceptual design objectives.

Regarding the conceptual stages of architectural design. Bueno et al. (2014) note that conventional digital tools lack a friendly interface regarding the early architectural design processes. Consequently, early digital design tools did not adequately integrate contextualization workflows. However, the adoption of computational and AI tools is beginning to bridge this gap.

Today's period highlights constantly evolving digital design tools. By incorporating computational and AI capabilities, the benefits of automation, parametric design, and iterative visualization techniques are now accessible to users. They push the design processes towards more advanced approaches with data-driven design capabilities.

One of the most familiar computational tools that is heavily used in the industry is Grasshopper3D, a node-based visual programming interface for Rhino3D, allowing users to construct visual programming scripts through connecting different operational components.

On the other hand, AI tools are still evolving to define significant contexts in the design field. ComfyUI, like Grasshopper3D, operates within a node-based interface. It captures strong capabilities for users to utilize generative AI models in various design-related purposes.

By employing these tools, design processes can now involve direct contextual inputs such as site-specific climatic data, topography, and urban morphology. Enabling iterative exchange between various design schemes and their contextual constraints.

These advantages present a critical opportunity to explore the benefits of both tools in parallel for achieving conceptual design workflows that are relevant to contextual parameters.

1.1 Purpose of Thesis

Computational and AI tools encapsulate a huge potential to be utilized for producing data-driven design outputs in relevance to different extracted contextual parameters.

Grasshopper3D and ComfyUI can both advance design workflows in more complex, data-driven solutions. They are used for differing design purposes, but would complement each other if combined and imagined under a unified conceptual design process. Moreover, the research intersects both programs

The purpose of this thesis is to explore ComfyUI's Generative AI workflows in parallel with Grasshopper3D scripts applied within a conceptual design process. To achieve iterative conceptual design schemes in relation to contextual inputs.

1.2 Scope of Study

The research is a design-specific experimental process, focusing on exploring the tools' potential within the conceptual stage of an architectural design process. Rather than conducting empirical testing or quantitative measurements, the research emphasizes the experimental nature of node-based tools (Grasshopper3D and ComfyUI), within a research-through-design methodology.

The experiment integrates environmental data into the design process as a primary contextual parameter for the massing development, rather than considering broader architectural context considerations, such as site analysis, structural, and contextual elements, which will require significant data collection and documentation process. Afterwards, detailing the massing to incorporate rich architectural features.

1.3 Research Questions

Despite observing the integration of AI and computational tools in the architectural practice, their application within the conceptual design stage and in relation to contextual parameters remains underexplored. There is a lack of a framework that addresses the contributions of Grasshopper3D and ComfyUI in parallel for contextually driven conceptual design workflows.

This thesis questions how node-based computational and AI tools can be integrated in a dependent process to enhance contextually relevant design schemes during the conceptual stage of architectural design.

1.4 Significance of Study

It is significant to explore how AI and computational design workflows allow architects and designers to explore imaginative but also contextually based design concepts through their particular node-based interface. Through the development of various custom-made scripts and workflows.

This thesis documents the contributions of Grasshopper3D and ComfyUI for the framework of conceptual design approaches, analysing the generated outputs and gaining insights from the overall experiment.

By developing four design outputs in different architectural styles, including blob, biophilic, vernacular, and modern approaches, the experiment highlights the dynamic use of both tools for the architectural concept development processes.

The findings highlight that these tools help iterative, data-informed form studies with expressive architectural features. Aiding contextually aware and architecturally detailed design outputs during conceptual processes.

2. LITERATURE REVIEW

Existing tools, methods, and even scientific theories foster further progressive ideas, knowledge, and tools, resulting in more qualitative outcomes in relation to specific discourses. In that sense, both theoretical and physical productions influence each other in parallel regarding the architectural field.

When examining ancient Neolithic and pre-Neolithic periods, and with the spread of agricultural practices, the architectural field advanced in parallel with the newly introduced tools, methodologies, and material frameworks.

Being able to more efficiently evolve the performance of the environment and human-related practices at that time, *“Man has begun to control Nature, or has at least succeeded in controlling her by co-operating with her”* (Childe, 1951, p. 59).

Known settlements such as Göbeklitepe and Çatalhöyük demonstrate such a transformation in architecture. Both cases highlight significant architectural configurations and elements, particular to their time. The emergence of new artefacts and technical craftsmanship has enabled the social and environmental transformations in these specified contexts.

The antiquity period captured key developments in scientific and philosophical fields. These theories gave a symbolic meaning to the shape of the physical realm and the built environment. The architecture of classical antiquity adopted concepts of symmetry and order. Building a local identity of architecture and forwarding such intangible societal ideas of rationality in physical conditions.

Furthermore, these theories also acted as a base ideological mindset for later periods, such as the Renaissance era. The development of perspective drawings during the Renaissance period has also pushed new visual representation methods and storytelling in architecture.

The rational systems in the philosophical discourse of classical antiquity also set the foundations for later technological advancements and tools. Aristotle’s concepts of the syllogism laid the groundwork for systematic reasoning, which inspired the development of artificial intelligence for later generations. As Russell and Norvig observe, “*His syllogisms provided patterns for argument structures that always yielded correct conclusions when given correct premises*” (Russell & Norvig, 2020, p. 3).

During the industrial era, with the rise of new technological developments and tools, the utopic ideal and narratives prospered, looking for new potentials for architecture, and sparking a re-evaluation of the field in practice.

Martin Heidegger states, *“modern technology too is a means to an end”* (Heidegger, 1977, p. 5), suggesting that technology’s core value lies in the ability to serve the human understanding and purpose, which exemplifies the same intent of any other human-made product.

In architecture, architects such as Tony Garnier have envisioned new urban configurations driven by industrial tools and frameworks *“The architectural scheme presented here in a long series of plates focuses on the establishment of a new town, an industrial town"* (Garnier, 1990, p. 43).

Although the proposal of the “An Industrial City” project was not physically implemented, rather it would be expressed as an experimental approach, the concept affected the urban planning genre as a whole and acted as a reference model, expressing how alternative design thinking would be influential for allocating important insights of future works.

Le Corbusier has also explored how such technological products can lead to a different practical version of architecture (Corbusier, 2007), drawing analogies between technological products, such as airplanes for instance, and their potential intersection with architecture.

The Archigram platform, aimed to iterate with experimental works of future cities and architecture in a utopic visual intent, reimagining the existing standards of modern configuration.

Although there are no physical outcomes produced by these explorations, they had an effect by pushing alternative design approaches and mindsets for architecture, driving a new conceptual framework and thinking.

2.1 CONTEXTUALIZATION IN DESIGN CONCEPTS

Abstract and conceptual works indeed play a strong role within the architectural design process. The concept of a building expresses the identity and reason behind such architectural configurations, in relation to design-related efficiencies, qualities, and performances that are achieved.

The lack of conceptual thinking and design strategies would lead to results like standardized outputs. Identical products without considering contextual data and links to address design challenges specific to each case.

Although standardization had benefits in terms of streamlining construction processes, it was critiqued in the sense that it disconnects intention, unique design, and purpose in a project, *"Architects have often looked to thinkers in philosophy and theory for design ideas, or in search of a critical framework for practice"* (Coleman, 2015).

One of the initial stages of a design is defining the concept behind it. Nigel Cross proposed that, in general terms, a design is an ill-defined problem, relying on the human's intelligence and unique ability that formulate such designs, *"It seems possible to make a reasonable claim that design ability is a form of natural intelligence"* (Cross, 2006).

The same point is valid for architecture. Architecture encapsulates various models and approaches that could be implemented and applied; each approach relies on a different design thinking, rationale, and philosophy.

These models express how the practice keeps fostering differing alternatives of concluding a design, not only through different paradigms, but also through diverse design methodologies, materials, and tools.

The contextual parameters, such as materials, data, and environmental references, all act as a system for the conceptual process. Each context has particular demands and parameters to be considered within the process. *"To illustrate the fact that a great variety of actual building systems can be developed, based on these patterns,"* (Alexander et al., 1977, p. 937).

Lefebvre describes, *"the rhythm analyst starts with 'concepts' and 'definite categories', 'with full consciousness of the abstract to arrive at the concrete"* (Coleman, 2015, p. 96).

Thus, it is the role of the designer to drive the conceptual framework of an architectural project, addressing the necessary elements and contextual parameters to consider in any specific architectural design project.

Design tools also play a significant role in aligning the design output to a certain extent. In that sense, the conceptual thinking process plays best in relation to sketch drawings. The digital design tools have been lacking the ability to address contextual parameters during conceptual processes.

However, contemporary tools of AI and computation have been pushing alternative design approaches to explore, which would address this issue. Basarir (2022) promotes that AI-focused courses in architectural education aid in enriching new exploratory approaches. Zeytin et al. (2024) explore how new AI tools could be used to enhance the designer's performance.

2.2 DIGITAL RELEVANCE

Modernist architecture evolved in tandem with the continuous development of industrial practices, emphasizing functionalism and system-based approaches in architecture. The framework of modernism acted as a role model for the design disciplines "*Form follows function and industrial design is the yardstick for all design"* (Evers, 2015, p. 510).

As digitalization spread, it gave rise to new paradigms in design. Alan Turing's concept of the "Turing Machine" laid the groundwork for cybernetics and machine-human communication systems. Opening new possibilities for tools and technologies to be based on computational logic.

In this context, Menges & Ahlquist (2011) addresses key figures such as Gordon Pask, who emphasized the influence of cybernetics in architectural design, *"In the late 1960s, cybernetics, system theory and electronic computers were seen primarily as tools to facilitate the logic of problem solving and decision making in architectural design"* (Carpo, 2013, p. 48). Moreover, new design tools evolved, captured in a digital environment, and through a computational framework.

One of the most significant initiatives to apply computational logic in a design tool was the Sketchpad developed by Ivan Sutherland. This digital tool allowed the user to manipulate and play with geometric forms. Chaillou (2022) mentions how it provides flexibility and comfort level for drafters. "*applied the idea of constraints which could be varied to test and flex the relationships between geometries in the formation of an overall system."* (Menges & Ahlquist, 2011, p. 12).

Aside from Sketchpad's functional aspects, its interactive graphical language, which was introduced, contributed to visualizing complexities and acted as an influential reference for further digital design tools.

The use of digital tools featured new design benefits, due to their unique system that is unbound to the physical world, enabling more efficient design experimentation and iteration, reducing material loss and design prototyping costs.

In this framework, Carpo (2017) discusses how the architectural field specifically embraces digital changes sooner than other professions. Moreover, new terminologies become more relevant in the architectural field, such as cybernetics and cyberspace.

Building upon early digital tools, Computer-Aided-Design (CAD) emerged as a pivotal turning point. It has advanced the framework of human and machine communication in design to a further extent. The most famous case of CAD technology is AutoCAD, a 2D-based drafting tool for architects and designers.

The CAD technology dramatically transformed the drafting workflows for architects and designers. The architectural drawing process changed. From an analogue and traditional drawing process of pen and paper to digital drawing techniques through CAD technology, in an efficient method.

In addition, CAD programs have been commercialized, providing inclusive access for more frequent use. This new technology showcased new theoretical and practical frontiers for the architectural discipline.

As digital design tools are intangible, depending only on visual senses for the user to communicate with, user-friendly visual representations of tool features and navigation would play a strong role in building a clean and seamless user-to-machine communication.

Most digital design tools generally have a friendly interface (UI) for users, making the experimentation and learning phases easier and faster. Although CAD and other initial digital tools showcased a new way of designing, they have been bound to specific loops of commands. Under an object-oriented framework that organizes the design commands around explicitly specified objectives.

This overall rigid framework results in static feedback of commands. Although users can control and manipulate aspects within a design through options, the element of exploration and rapid design iteration was still limited within this scope.

Bernstein (2022) expresses that CAD streamlined the technical drawing processes more precisely, but did little to contribute to the gap in defining a design. Meaning that addressing the contextual parameters still lacks their integration inside the digital design realm. Bueno & Turkienicz (2014) also note that the connection between the mind and CAD is inefficient and slower compared to the connection between the mind and sketch drawings.

Sketching always offers an unlimited number of design scenarios to develop and build upon while thinking in the abstract. This particular benefit eases the bridge between abstraction and actual design development in a harmonious process. The digital design tools, on the other hand, have been more rigid as a conceptual design tool.

But with the developments of digital tools, computation for design has evolved into a more dynamic element. Design processes could capture more in-depth alterations, calculations, and contextualization aspects for the applied concepts.

2.2.1 Initial Digitalization In Design Concepts

In parallel to the previously mentioned theories, remarkable works of Cedric Price emphasized the use of technology and architecture, simultaneously in an imaginable and abstract sense, showcased through design tools.

Cedric Price's abstractions were captured in sketches and drawings. Although ordinary and typical mediums, the content provided in Price's works was new and controversial. As sketching is known to emphasize abstract and intuition-based designs, by the simultaneous practice of the mind and hand. Lawson (2005) regarded it as part of the thinking process of designs. Furthermore, the content within such drawings showcased new kinds of data, tools, and techniques to learn from and implement in the field.

The case of the Fun Palace, for instance, emphasizes a utopic but also functional approach. The project had the concept of a building that would respond to its visitors' actions, *"Price used technology as a provocation for change— change in the design process"* (Steenson, 2017, p. 129). In that sense, Chaillou (2022) elaborates that Price's works addressed machines as autonomous design agents, forwarding theoretical discourses for both computational design and artificial intelligence.

Similarly, the Archigram platform pushed new kinds of visual approaches and design concepts in relation to architecture and technology, developing an alternative way of expressing architecture. Thus, in this context and period, the discourse of conceptual design captured higher focus in relation to new technologies.

These imaginable and conceptual architectural works symbolize the design potential when implementing new abstract concepts, techniques, strategies, and developments. How they could evolve new ideas of architecture and the built environment, *"it was about the interactions that they could shape, whether for the people (or birds) who used them or the architects who designed them"* (Steenson, 2017, p. 128).

2.2.2 Computational Potential In Design Concepts

As later developments in digital design tools advanced computation beyond a static object-based approach, they introduced a rule-based system where commands and loops are dynamic and iterative.

In here, the dependence of a manual design process mindset changed, and new paradigms developed. Menges & Ahlquist (2011) note computational design as processing internal and external properties.

The term *"Processing"* would resemble the essential differences between the computation-based design tools and manual modeling tools. Sean Ahlquist and Achim Menges state, *"One increases the amount and specificity of information, while the other only contains as much information as is"* (Menges & Ahlquist, 2011).

Computational design is a broad field that encapsulates various sub-fields, including parametric modeling, generative design, digital fabrication, and environmental simulations. The list can go further with new features evolving. Through its diverse methodologies and dynamic uses, computation has formulated a shift in the framework of digital tools concerning the design process.

These approaches of computational design vary and are used for different purposes. The use of programming, scripting, and algorithms through evolving patterns increasingly offers more efficient and optimized design outputs. Burry (2011) describes how the effect of automation provides a deeper engagement between the user and the computer.

Parametric design is one of the most essential properties of computational design. Jabi (2013) expresses it as a step-by-step technique, conducting controllable outputs according to changeable inputs. Allowing a dynamic and iterative input and output process, manipulated through varying numerical data.

Computational design workflows shaped the modeling process in a new framework for users. Through a visual programming node-based interface that follows logical design processes. Allowing custom-made design scripts and algorithms without the need to manually code. Leclercq (2011) elaborates regarding node-based interfaces, how this type of visual dataflow modelling can accelerate the process of iterative design approaches, allowing more design possibilities to be explored. Burry (2011) argues that applying this process would force a design to be logically deconstructed to be encoded, emphasizing that designs are constructed as a logical system in this regard.

Grasshopper3D is one of the most influential software to capture computational functions in relevance to the field of architecture, through a node-based programming environment.

Grasshopper3D's procedural framework in capturing logic-based design instructions and its dynamic capability shape a module of rational construct between the user and the tool, *"Models of design capable of consistent, continual and dynamic transformation are replacing the static norms of conventional processes"* (Agkathidis, 2015, p. 17).

After this tool was introduced into the architectural practice, the field has elevated new design advancements. The ability to generate varying design forms and other features has become more streamlined, and the capability to set contextual parameters to guide the design formulation has been added. Thus, their impact on architecture has fostered new practical formulas to solve design challenges.

Complex structures, forms, and building elements are now more applicable using these computational tools, "*the new structuring provides the mathematical/geometric, syntactic and formal logic which is necessary for digital tectonics"* (Oxman & Oxman, 2010, p. 17). Similarly, Carpo (2017) elaborates on how digital design tools are eliminating the constraints of industrial standards in both theory and practice. this pushed the more unique architectural design outputs.

Computation has also highly benefited iterative design approaches through parametric and generative design workflows. Simulating performances of geometries and forms relative to contextual data, adding an effective contribution to the experimental and optimization phases of design paradigms.

Figure 2.1 showcases a timeline of developments affecting design approaches. With all these new technologies, the architectural practice has advanced to aim for contextually adaptive, efficient, and performance-based design outputs.

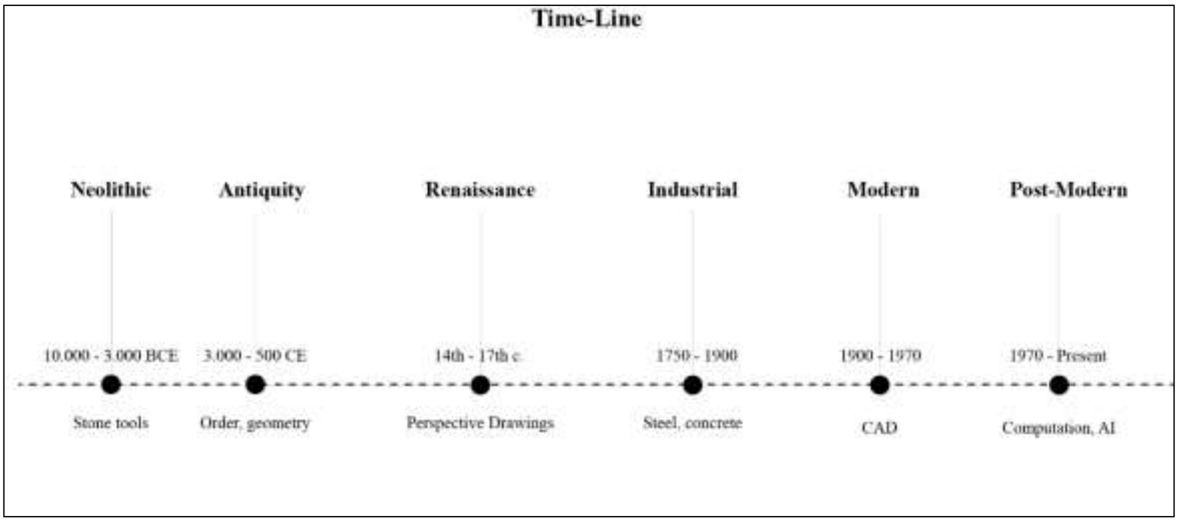


Figure 2.1: Timeline of Developments Affecting Design Approaches (Elabbasi, 2025).

Computational design processes bridge the applied mathematical formulations with the architectural practices, making significant pushes in the production of complex forms and structural configurations. Zaha Hadid's complex geometric forms are driven according to mathematical rules, but produced through computational design tools and workflows.

Digital design tools help to optimize the physical qualities of a design. Through building digital simulations of physical events. In that way, the discipline of architecture in the current time binds physical in parallel with design decisions made within the digital sphere, *"digital and physical architectures are increasingly inseparable from one another"* (Steenson, 2017, p. 224).

Furthermore, various sources of data references in the physical and digital scales must be considered in the design process. From environmental to structural analysis, material, and construction methods. The philosophy of standardized architecture has shifted towards more variable, customizable, and conceptual designs.

Through parametric, generative, and algorithmic design methods, the generation of design is now helping the conceptual process in terms of form-finding, which would be claimed to ignite more design thinking within the discipline, *"but for the designer creativity is expressed, surely, in the very process of design"* (Leach, 2022, p. 72).

Menges & Ahlquist, (2011) explain computational design's usage as an intellectual investment of alternative material systems. Thus, the design decisions, techniques, and strategies of architectural concepts would now have diverse methods.

Furthermore, Carpo (2013) & Agkathidis (2015) discuss the role of new tools shadowing influential perspectives and the new exploratory approaches in the applied architectural design sphere. Peter Eisenman's image morphing technique expresses this simply by utilizing the potential practical possibilities of the digital image.

Throughout time, the development of design tools contributed to the architectural practice on various practical levels. In this post-modern era, exploring new conceptual design workflows through contemporary tools would open new potentials for addressing contextual properties within the process.

2.3 PARADIGMS OF ARTIFICIAL INTELLIGENCE

Reaching the contemporary time of Artificial Intelligence (AI), it is a human-designed framework that embodies intelligent functions and behaviors. AI's underlying mechanisms and frameworks are new, unique in solving different kinds of challenges, enabling new possibilities for architecture and design.

AI captures various types of technical systems. The Turing machine and the topic of cybernetics were the critical bases to advance human and machine communication systems. AI as a paradigm is practically simulating and mimicking the human approach of solving problems and finding solutions to different situations.

Instead of constructing a tool and using it for a single and specified purpose, AI is about developing a tool that would capture built-in mechanisms to solve unspecified tasks.

By doing so, the mechanism is broad and general, meaning that it can be applicable to a variety of scenarios, events and challenges to solve, according to its "rational agent" based systematic approach, Russell & Norvig (2020) have mentioned how the construct and the implementation of logical framework were crucially influenced by Aristotle's statement of how justifiable actions to be proposed based on prior goals and knowledge, to achieve optimum outcomes.

In this sense, the set goal of an AI is to do the right thing, which is an objective proposed by the developer to the machine. This paradigm is referred to as the "standard model" of AI.

2.3.1 AI Theoretical Framework

Just like the paradigm structure of computational design, AI relies on a logic-based framework. Moreover, AI possesses critical potential to advance and diversify numerous architectural design workflows. Its possibilities highly value user control and iterative work.

The key components that make up AI are machine learning and deep learning. Their combination as a system would make the uniform framework of AI. Leach (2022) elaborates their connection through an analogy of onion layers, as deep learning functions as a category under machine learning, and machine learning under AI.

Machine learning's purpose is to improve AI's ability to do the "right thing" (Leach, 2022), through different processed data, making probable decisions to build more experience. Russell & Norvig (2020) states that deep learning refers to machine learning that implements multiple layers of computing elements. This means the juxtaposition of enormous data in a single structure.

2.3.1.1 Generative AI Framework

The mentioned systems of machine learning and deep learning serve as the foundation of any AI-based program, enabling it to solve the intended tasks. Generative AI is an AI model that utilizes machine learning and deep learning methods.

Its structure has different algorithmic paradigms, but the most well-known one is the unsupervised learning algorithms, trained on certain data to produce new content. They are trained on large datasets relevant to the desired output type. Russell & Norvig (2020) say that unsupervised learning algorithms produce generative models that can generate text, images, audio, and videos.

The uses of Generative AI tools in architectural design constantly develop new potential, but the mainstream approach relies solely on a prompt description to generate the output visuals, expressed as a text-to-image workflow process.

Further advancements of AI have reached image-to-image workflows. This includes inserting a base image as an input reference within the workflow, enabling the model to process and modify a base image directly. Bringing a closer relevance to the architectural design process by allowing users to visually detail or further abstract the initial 3D architectural models, sketches, or technical drawings they conduct.

The image-to-image workflow allows image manipulation and optimization potential. It can be adjusted according to the user instructions, whether to let the output be closer to the input image or different. In that sense, this step in fact pushes the applications of AI tools to act more practically within the architectural and conceptual design processes. The next section examines necessary AI models theoretically, viewing the main models that are relevant to the overall practical sections of the research.

2.3.1.2 Generative AI Models Analysis

As mentioned, Generative AI models function in generating various products, as text, voice, or visual outputs, through the implementation of the Artificial Neural Networks (ANN) systems*" The architecture of these networks can vary greatly, as users can modify the number of neurons, layers, training parameters, and other settings to adapt the model to specific tasks"* (Chaillou, 2022, p. 70).

Regarding image generation and synthesis through Generative AI, there are crucial AI Model types to tackle, such as GAN, CNN, VAE, CLIP, and Diffusion models, each has strong benefits for image generation workflows. Some of them can also interact together under a single pipeline.

Generative Adversarial Networks (GAN) is one of the earliest generative AI models, developed by Ian Goodfellow, structured upon two neural networks, one generates image data, and the other tries to detect the accuracy of the generations, helping to generate coherent content.

Convolutional neural networks (CNN), a major deep learning type that has strong capabilities for image-related generation tasks and recognition. It can detect data patterns from input images, extrapolating edges, shapes, or other features according to certain processes. *"These architectures have been crafted for the treatment of visual imagery"* (Chaillou, 2022, p. 71).

CNN pushes the generation consistency with image inputs to the next level. Madhavan (2021) describes its predominant relevance in image generation. A notable neural network model example that highly depends on the CNN structure is ControlNet, which is mainly used as an extension for diffusion models for optimizing image outputs.

Furthermore, an important Generative AI Model to highlight are the variational autoencoders (VAE), composed of an encoder and a decoder, an encoder that abstracts the input data, and the decoder reproduces it after certain manipulative processes through different layers.

VAE's functional mechanism captures a key value in being able to reference inputs and turn output image generations. Thus, its role is important to be applied in parallel with other components and generative AI models under a single pipeline. "*This ability of VAEs to model and render diversity found in the data constitutes their "generative" potential. Over the past few years"* (Chaillou, 2022, p. 74).

Another AI model is U-Net, derived from the CNN type, but its purpose is more specified here. Its approach is about semantic segmentation of image data, segmenting relevant data or pixels within the overall image input.

Diffusion models are an essential model, specifically focused on image generation. Diffusion model's software architecture employs the U-net model and other systems, the U-Net model here functions for denoising and reconstruction of data. Its function relies on encoding and decoding, but it is unique in applying noise within the process to produce visual outputs.

There are various diffusion model types existing; they would differ in qualitative aspects according to their properties and performance capacities. Han & Kim (2024) discuss how the diffusion model's in applying Gaussian noise inside the algorithmic process is increasing the visual tasks and reducing the computational costs of the processors.

Furthermore, examples to mention of a diffusion model type are Stable Diffusion 1.5 and SDXL models, which are unique in their computational efficiency. Stable Diffusion models are one of the most influential AI models in image-related tasks.

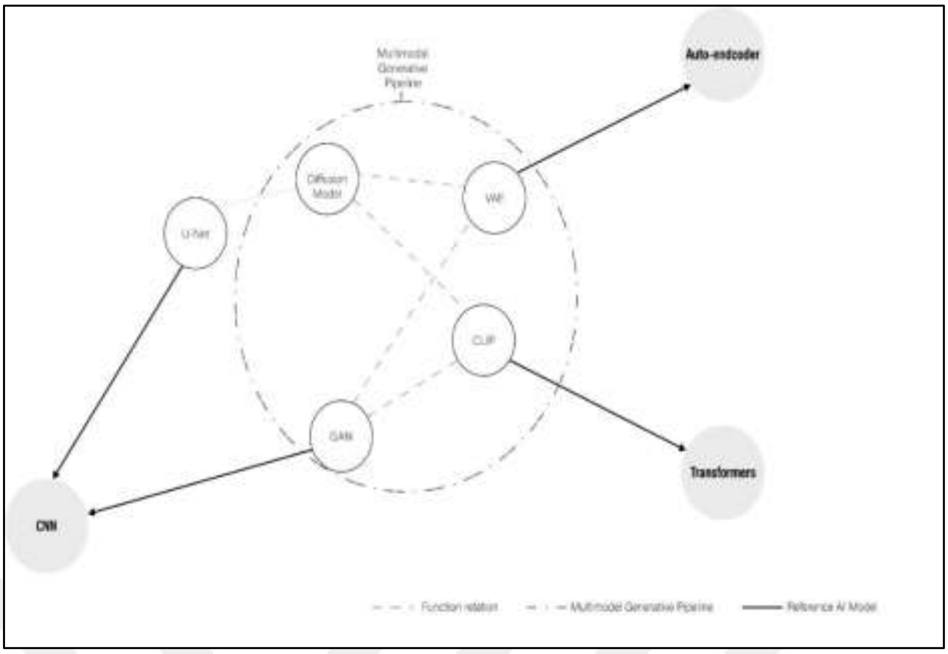


Figure 2.2: Systematic Diagram of a Multimodal Generative Pipeline (Elabbasi, 2025).

Contrastive Language-Image Pre-training model (CLIP) plays a crucial role in making synchronization between the inputs of texts and images, as it matches the most aligned data of text and images together, embedding both text and image inputs, Radford et al. (2021) explains how CLIP functions at a high-performance rate while compared with other methodologies, highlighting its strong role in image and text synthesis.

All of these are models that suit varying functions. Combining generative AI models under a pipeline process will frame more purposeful outputs. By doing so, the performance and accuracy aspects increase. Wang et al. (2024) concluded a result of how processes of juxtaposing diffusion models with clip models would enhance visual understanding tasks.

Similarly, Wesego & Rooshenas (2024) conduct another custom implementation that uses diffusion models and VAE frameworks, leading to a higher performance rate of tasks.

This performance push is derived from such an algorithmic combination of a such AI model within a multimodal generative pipeline. Expressed as a system that combines various Generative AI models uniformly under one workflow.

The approach of a multimodal generative pipeline contributes to the practical performance of AI models for various application types. Figure 2.2 represents a systematic diagram of a multimodal generative pipeline.

The Figure diagram visually elaborates the connection and relationship between different Generative AI models. Generative AI tools and their uses for design purposes capture such a back-end framework of multimodal pipelines.

2.4 NODE-BASED INTERFACES: GRASSHOPPER3D & COMFYUI

The User-Interface (UI) part of the front-end in a program expresses a visual communication between the program and its user. An essential benefit to implement inside the UI is to create a friendly visual environment to work with the program.

Node-based UI's are one of the key interfaces that allow dynamic processes to be applied. In that sense, Grasshopper3D and ComfyUI programs would be expressed as such; both are node-based programs that seamlessly represent a logical and procedural framework. Grasshopper3D is used for computational design, while ComfyUI is used for generative AI workflows.

As Grasshopper3D and ComfyUI are developed in similar interfaces, with a node-based visual programming, Zhou et al. (2023) discuss how node-based visual programming interfaces would allow a user-friendly environment for the user to work with, especially novice users with low programming experience.

Furthermore, Zhou et al. (2023) experimented with users on an AI-based program (InstructPipe) that had its UI as a visual programming environment. The experiments elaborated how this UI provided positive aspects in qualitative and quantitative objectives. Highlighting how the node-based interface bridges between technical data and the actual user's capabilities.

Grasshopper3D captures a variety of components that can interact with each other. Each component is responsible for doing a specific operation. The nodes have input and output data, and are to be connected in a left-to-right flow.

Thus, when combining and connecting relevant nodes, it builds a dynamic and data-driven computational system to be used for design purposes. Through Grasshopper3D's node-based environment, workflows of parametric, generative design, digital fabrication, and performance-based design are all applicable processes. Figure 2.3 showcases the Grasshopper3D interface.

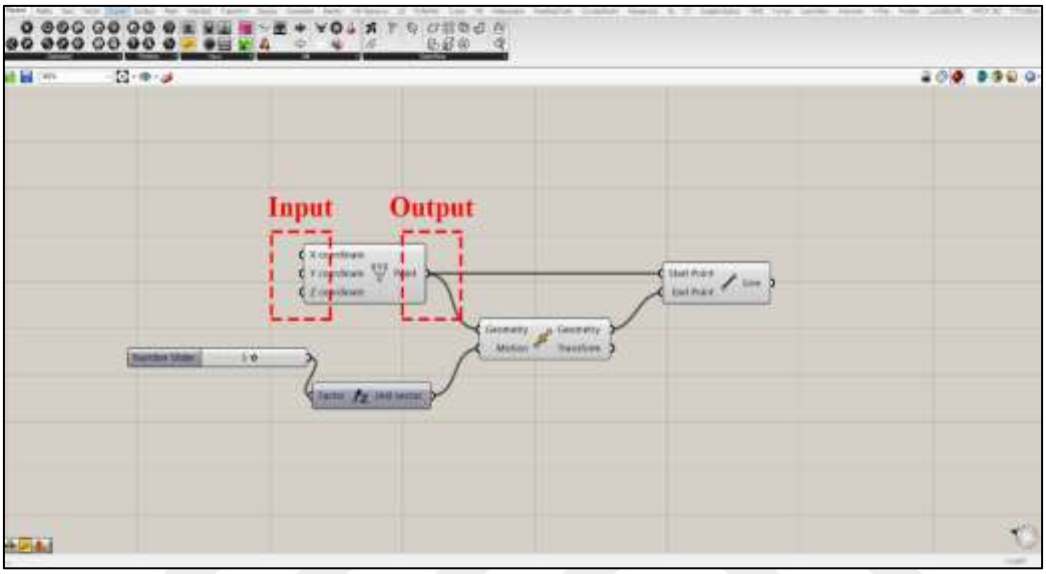


Figure 2.3: Grasshopper3D Interface (Elabbasi, 2025).

ComfyUI captures most of the previously mentioned AI models inside its library, including Diffusion models, CLIP, and VAE. It allows the interpolation of these AI models to evolve a uniform functional pipeline.

Each node is responsible for a specific function. They include input and output data and are procedurally connected from left to right. Figure 2.4 showcases the ComfyUI interface.

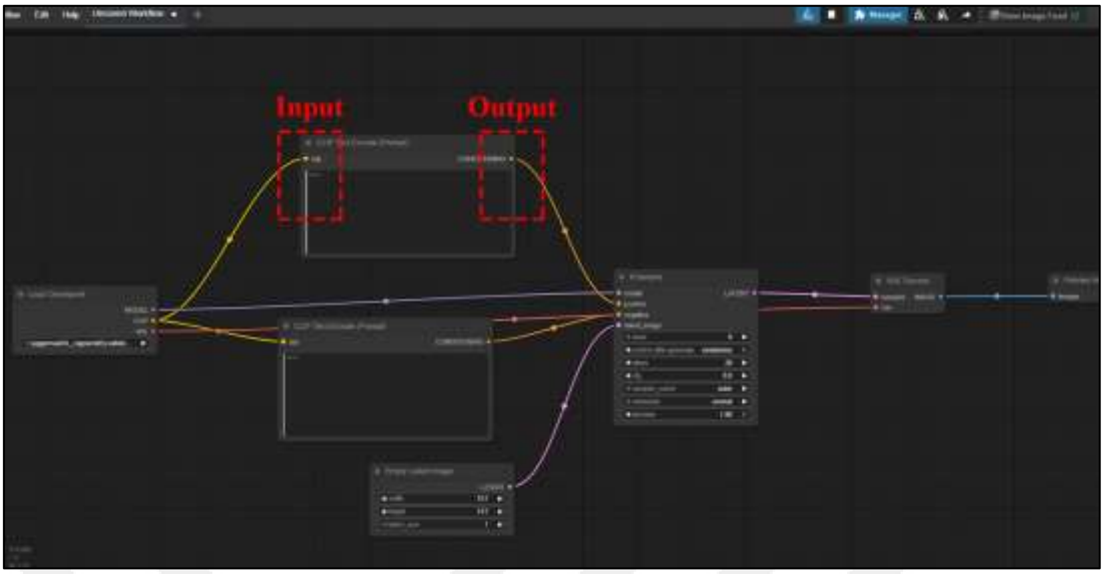


Figure 2.4: ComfyUI Interface (Elabbasi, 2025).

After mentioning the necessary properties and functionalities of relevant node-based interfaces, the methodology is to be implemented. It involves a practical experiment that will explore the intersection of parametric design processes with different generative AI workflows under a dependent pipeline, using Grasshopper3D and ComfyUI. Aiming to achieve a meaningful design approach using these tools, setting a context regarding their application to suit the purposes of an architectural concept development process.

3. METHODOLOGY

New digital tools of generative AI and computation have significantly expanded the scope of design potentials. Node-based user interfaces have further expanded the potential of advanced AI and computational design technologies by enabling the creation of more efficient and customizable workflows. Key tools in this regard are Grasshopper3D for computation and ComfyUI for generative AI.

Computational tools are more familiar with the architectural field, while both keep evolving with new qualities and features. Grasshopper3D is highly used in architecture and design to implement advanced computational workflows. After AI tools were introduced, ComfyUI has increasingly been adopted within architectural workflows.

ComfyUI is a relatively new, open-source platform that enables users to experiment with different AI models to produce various outputs. It can handle various AI workflows to produce text, audio, or visual outputs and allows dynamic iteration with the node's parameters. With its unique capability offerings, it has high potential for the conceptual design process regarding visualization.

The conceptual design processes in architecture include various design studies and analyses, all concerning contextual data considerations. Moreover, the process of form generation and iteration is essential for concept testing and optimization.

The methodology highlights insight from an experimental process using Graashopper3D and ComfyUI for the architectural concept development process.

Addressing a suitable position of both programs within the design process. The experiment adopts a Research-through-Design (RtD) methodology combined with an experimental qualitative approach, exploring contextually-based iterative design processes by interconnecting Grasshopper3D and ComfyUI.

The methodology begins with developing a Grasshopper3D script to generate different morphologies to iterate with, utilizing the strong elements that Grasshopper3D provides to develop such complex geometries.

Afterwards, applying 2 ComfyUI workflows to apply visual fine-tuning processes to these generated forms relevant to 4 different architectural styles and approaches, which are blob, biophilic, vernacular, and modern architectural approaches.

Finally, by evaluating the outputs and the overall design experiment, this experimentation explores a meaningful conceptual design methodology by integrating both programs in a dependent manner, reflecting new insights towards conceptual design processes. In this framework, the upcoming sections will focus on the experiment of Grasshopper3D and ComfyUI in technical details.

3.1 PARAMETRIC MODELING PROCESS

Grasshopper3D effectively pushes computation in architecture to a new level, emerging a new genre of design processes through dynamic data-driven applications. Parametric, generative, and algorithmic design approaches are all applicable methods inside Grasshopper3D, binding the geometric forms with numbers and calculations all inside the digital domain. underlining data-driven and performance-based design outputs.

The computational design process is framed to generate data-driven 3D models, but achieved through rules and a logical framework rather than a static object-oriented modelling approach. By defining the essential rules in such a script, the computational workflow allows the generation of multiple outputs rather than a single, static form.

In this manner, four output variations will be produced, each of which will be further developed and adapted to reflect one of the four previously identified architectural styles and configurations.

Developing architectural concepts requires a wide array of data inputs, parameters, and rules to bind the design accordingly, such as environmental, material, and form relations. Computational design tools enable this kind of relationship between data and form-making.

Grasshopper3D is utilized for applying these processes. Thus, the main data reference that will bind the form development within the experiment is prevailing wind data, guiding the configuration of the output form.

By constructing a script that can generate varying geometries and differentiate them according to environmental data references, the conceptual design process can be enriched more in terms of design exploration, form-finding, and form optimization processes.

3.1.1 Setting Geometry & Parameters

The initial step in the procedural node-based Grasshopper3D workflow involves defining a closed boundary curve as the input geometry. This curve establishes the spatial boundaries for the form development, within which the design development will take place.

All subsequent computational steps in the script are derived based on this input curve. The 2D boundary curve serves as the base for generating 3D geometry through extrusion, allowing for accurate massing definition and efficient manipulation throughout the iterative computational process.

For introducing environmental responsiveness within the conceptual process, the script integrates nodes that contextualize prevailing wind data from a specified location. This environmental input serves as the primary data-binding parameter in the design logic.

Two orientation strategies are considered, positioning the resulting 3D forms either parallel or perpendicular to the wind direction, depending on the intended environmental performance and conceptual goals.

Each of these two options adds different qualities to the design concept, but through this research, the method applies a perpendicular form relation to the wind. To consider the benefit of passive cooling and ventilation performance to the form, as this is generally a positive design strategy to implement, Figure 3.1 showcases the process of developing the initial steps.

A surface grid is then generated, driven by varying numeric input values. Afterwards, setting up the environmental context, i.e., wind data parameters, using the Ladybug tools plugin. The next step involves establishing rule-based logic to guide form development and variation in response to prevailing wind direction.

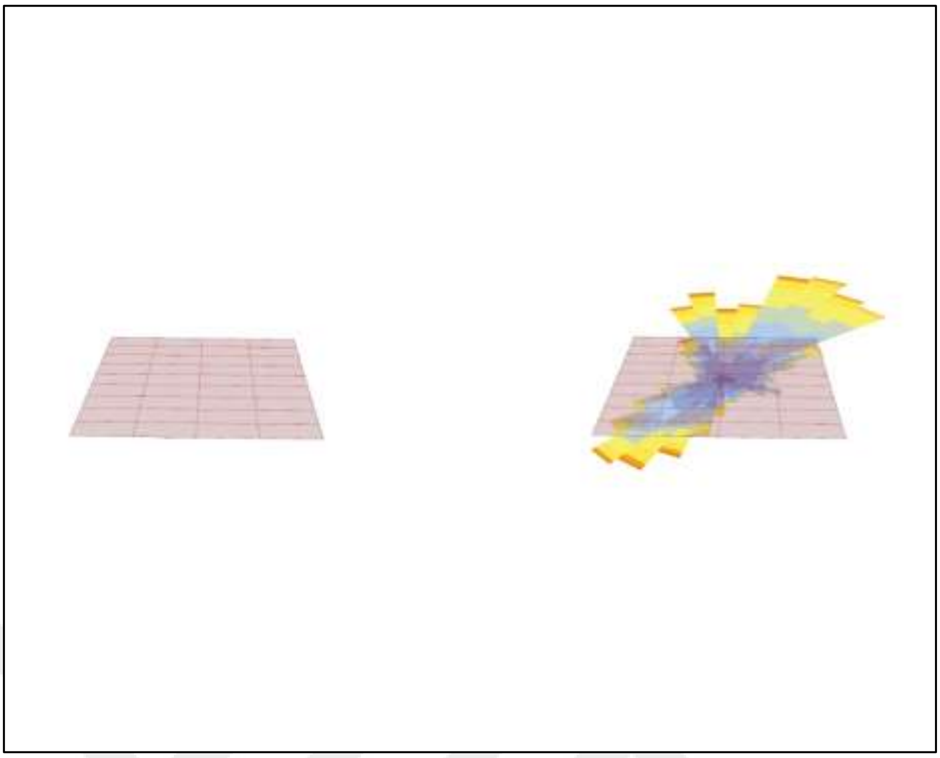


Figure 3.1: Setting Up the Base Surfaces and Environmental Parameters (Elabbasi, 2025).

3.1.2 Form Optimization & Synthesis

The next phase focuses on optimizing the design within Grasshopper3D by configuring parameters that enable dynamic manipulation of the form. This portion of the script directly influences the grid's spatial arrangement behavior.

In that sense, constructing a line perpendicular to the wind direction, and using it to position the 3D form to face the prevailing wind vector. This sets a rule that can arrange the surfaces in a specific growth pattern.

The logic identifies the surfaces closest to the axis line and removes the more distant ones using a culling mechanism. The number of culled surfaces is adjustable through numeric parameters. Figure 3.2 showcases the culling process of surfaces, filtering the surfaces that are being aligned to the vector of the guiding line.

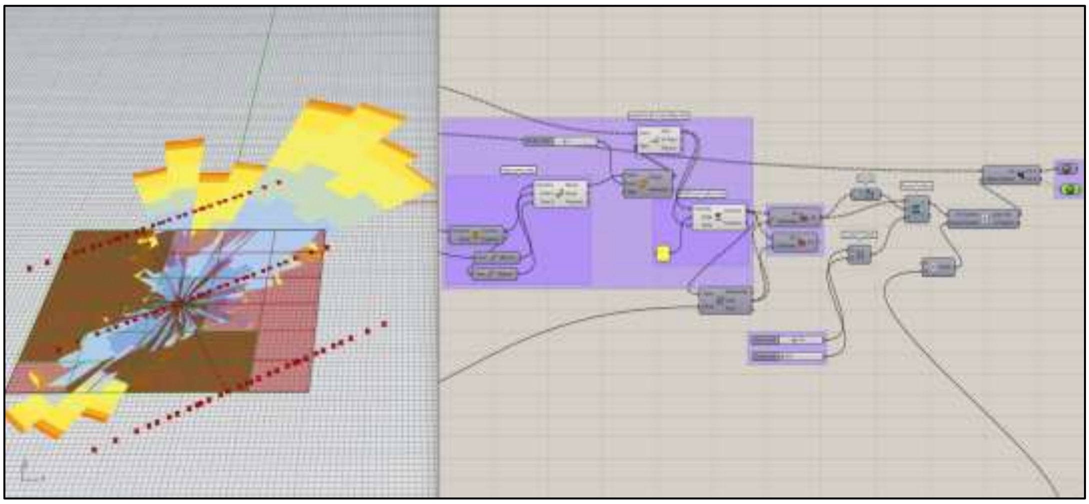


Figure 3.2: Culling Process of Surfaces (Elabbasi, 2025).

Moreover, synthesizing the form by selecting the filtered surfaces and merging them into a uniform geometry. Furthermore, extruding the geometry to attain the 3D form, and applying a subdivision process using the Weaverbird plugin to formulate an organic morphology.

Subsequently, the Galapagos plugin is applied within the script. In this process, Galapagos is tasked to produce variations of the mass according to different numeric values of a volume. From lower to higher volumetric expressions, achieving a form-growth logic.

This form development progresses to generate four output variations, each of which increases in size, from a volume of 4800m3 for the smallest, reaching 8800m3 for the largest. Each form iteration will represent a specific architectural style in the further steps of the experiment. Figure 3.3 showcases the form output through Galapagos, while Table 3.1 showcases the outputs of 4 parametrically driven form variations generated. The detailed settings for this Grasshopper3D script are provided in *Appendix A.*

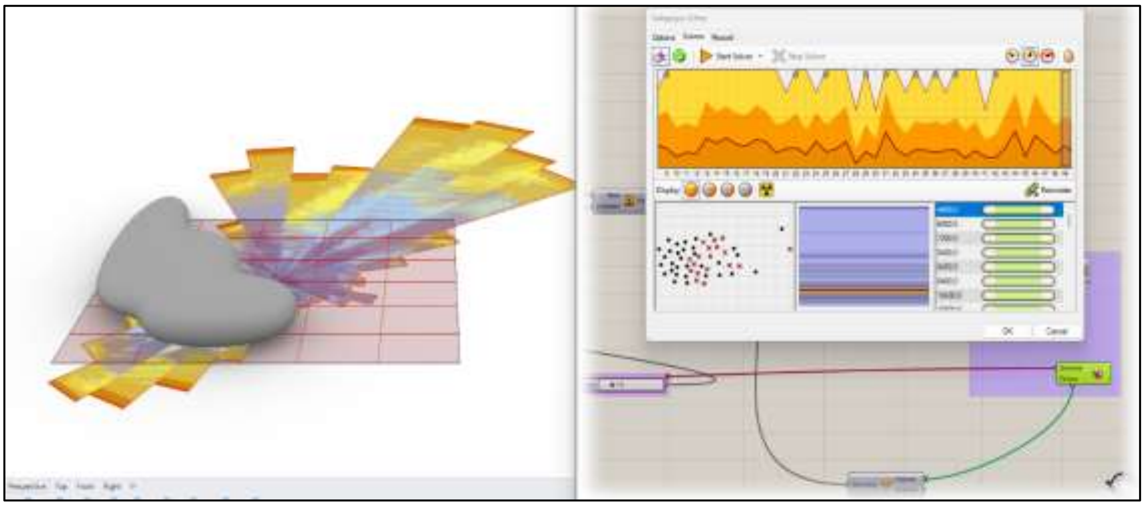


Figure 3.3: Form Output Through Galapagos (Elabbasi, 2025).

Table 3.1: Parametrically-Driven Form Variations (Elabbasi, 2025).

	Variation1	Variation2	Variation3	Variation4
				
Volume	4800m3	6000m3	7200m3	8800m3
Style	Vernacular	Biophilic	Modern	Blob
Prompt	Photorealistic aerial view of an architectural masterpiece in a vernacular style, vernacular architecture, vernacular local architectural tectonics.	Photorealistic aerial view of an architectural masterpiece in a biophilic style, biophilic architecture, sustainable biophilic architectural tectonics.	Photorealistic aerial view of a modern architectural masterpiece, modern architectural elements, modern architectural tectonics, glass curtain walls.	Photorealistic aerial view of an architectural masterpiece in a blob style, blob architecture, blob fluid architectural tectonics, glass, curvilinear architecture.

At this stage, the scheme remains generic and raw, primarily serving as volumetric studies without specific architectural detailing. These preliminary iterations establish the foundational design framework shaped through parametric inputs and contextual considerations.

In the next section, the focus will shift to finetune these base forms by applying architectural features, stylistic refinement, and visual articulation using advanced generative AI workflows within ComfyUI.

The subsequent phase will demonstrate how AI-driven image-to-image workflows could highly support iterative design development to produce detail-rich architectural schemes with intricate visualizations, suitable for distinct architectural styles.

3.2 AI WORKFLOWS DEVELOPMENT

During the conceptual stage, the design development requires multiple iterations and explorations. Traditional modeling workflows can be time-consuming and labour-intensive in the framework of constant iteration, especially for evolving concepts that may not be incorporated into the final design.

By implementing ComfyUI, the experiment addresses this point to explore how AI tools and workflows can add to the conceptual design process, aiming to test faster and more flexible design ideas.

ComfyUI has the potential to build different AI-based workflows for varying purposes. In this section, the research explores two generative AI workflows that would add extra architectural details to the four generated forms. Moreover, the workflows are based on an image-to-image approach. Aiming to develop an interconnected framework between Grasshopper3D and ComfyUI for the conceptual design phases.

The design approach involves developing four distinct architectural proposals, each representing a different style and methodology: blob, biophilic, vernacular, and modern. The exploration of these styles extends beyond simple geometric refinement and fine-tuning, but through detailed and iterative architectural visualization of preliminary design concepts.

To run a typical image generation workflow inside ComfyUI, it is important to set up the essential AI models, including the Diffusion, VAE, and CLIP models. Set with other processes under a uniform pipeline. Optional models and nodes also help in achieving certain qualities. Important models that have been used are LoRA and ControlNet.

The workflow applies LoRA and ControlNet models within the process, although both are optional to run, but are still important for optimizing the output generations in relation to achieving outputs relevant to the 4 distinct architectural styles.

To explain simply, a LoRA model helps to accurately produce specific visual content, allowing the ability for a unique optimization that the users can decide to add. The ControlNet model helps the diffusion models to grasp and preserve the input image’s elements according to different image detection processes. Both model types are developed according to a supervised model framework. Moreover, improving the overall image accuracy and performance of an image-to-image workflow.

3.2.1 Form Fine-Tuning Process

Building up the first workflow, framed to develop the architectural fine-tuning processes for the 4 designs, by iterating with the standard image-to-image workflow available natively within ComfyUI to reach an optimized output.

Furthermore, through trial and error, and by iterating the parameters of the nodes, understanding how to control the output visuals to achieve the optimum output gets streamlined.

The standard image-to-image workflow within ComfyUI involves the 3 main AI nodes, Diffusion model, VAE, and CLIP, with the KSampler node that operates as the main processor, as well as having an input image to define. The CLIP node includes positive and negative prompts that the user would define, as the VAE is responsible for synchronizing the process of the models with the inputs and outputs.

There are various diffusion models (DM) types that users could implement within image-to-image ComfyUI workflows. The models that are used in the experiment are Stable Diffusion 1.5 (SD1.5), defined within the first workflow for applying fine-tuning elements, and Stable Diffusion XL (SDXL), defined for the second workflow for implementing rendering scenes.

SD1.5 is prior to SDXL, thus, the SDXL model captures stronger qualitative performances when compared with SD1.5, but both could be utilized effectively to suit different approaches. Ultimately, the text would synchronize with the Diffusion and VAE models to approach the intended visual content. As described previously, the experiment aims to test the visualization of 4 architectural styles. Figure 3.4 showcases the standard Image-to-Image ComfyUI workflow.

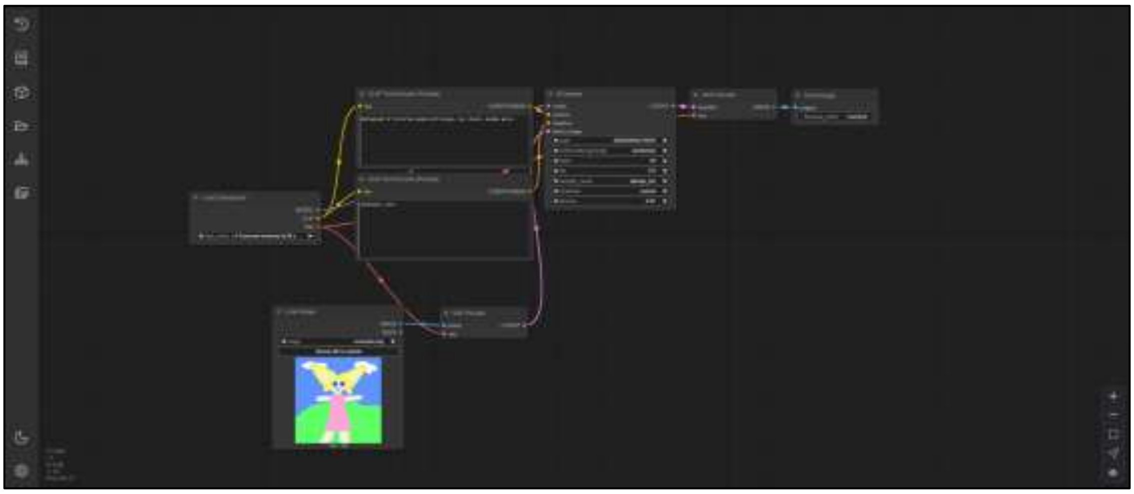


Figure 3.4: Standard Image-to-Image ComfyUI Workflow (Elabbasi, 2025).

The selected DM in the first workflow was SD1.5, as it enhances the processing performance, enabling the iteration with a wide variety of generations. Furthermore, adding LoRA models to produce more accurate visual content relevant to the 4 architectural approaches mentioned.

With the customization capability of ComfyUI, examining the parameters and nodes helped to optimize the outputs and produce relevant content. The most effective parameter in this sense was lowering the denoise rate of the KSampler node. It lowered the model's effect on the input image. This method produced analogous output contents to the input.

The outputs still had some irrelevant architectural elements that deform the initial configuration. For that reason, adding the ControlNet nodes significantly helped to apply accurate image detections of the forms, according to different detection maps. Further preserving the elements and visual content of the input image.

There are multiple detection maps to select when adding a ControlNet node, but the applied type was the depth map to preserve the spatial and dimensional features. Figure 3.5 showcases two outputs, with and without adding the Depth ControlNet model in the workflow.

By comparing the two outputs, it is seen how the ControlNet model clearly helps to accurately resemble the input image data. And to generate architectural features following the form's morphology. The alternative output generated without the use of ControlNet simply generates configurations based solely on its visual outline, without accounting for any additional input data.

The inclusion of ControlNet introduces a layer of precision by preserving key morphological characteristics of the input geometry. While both approaches offer distinct advantages depending on the design intent, the experiment adopts the ControlNet-enhanced method to ensure greater fidelity between the input form and the output.

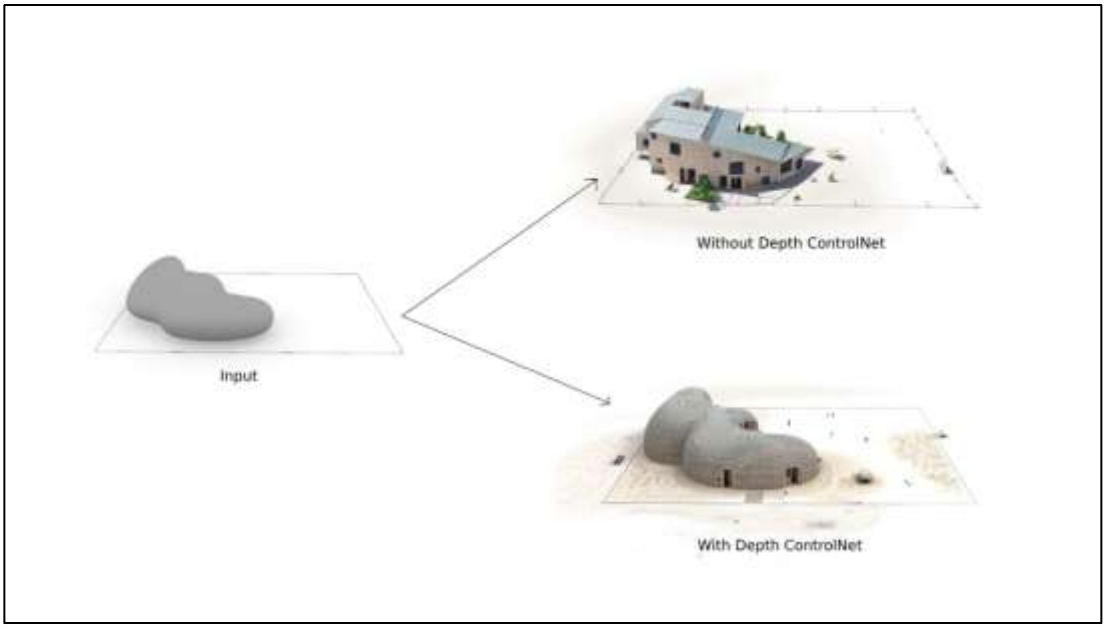


Figure 3.5: Comparing Outputs with and without ControlNet (Elabbasi, 2025).

The input prompt also plays an essential role in producing quality visuals. Applying advanced prompt engineering techniques is key to achieving the intended outcomes. For instance, raw and unspecific descriptions won't help the SD1.5 or the SDXL model to grasp the user's intentions. Furthermore, there are 4 different prompt descriptions constructed. Each describes a specific architectural type of the 4: blob, biophilic, vernacular, and modern styles. The technical settings for this workflow are provided in *Appendix B.*

Utilizing Large-Language-Models to develop prompts also increases the prompt performance. Moreover, the GPT LLM model is used in the process to set prompt descriptions to fit the SD1.5 model.

By doing so, the outputs of this workflow resemble fine-tuned forms that capture architectural features according to the prompt descriptions. Thus, the workflow is further applied to every form iteration; each iteration is responsible for capturing a specific architectural style. Figure 3.6 showcases the 1st workflow generated outputs for vernacular style, Figure 3.7 for biophilic style, Figure 3.8 for modern style, and Figure 3.9 for blob style.

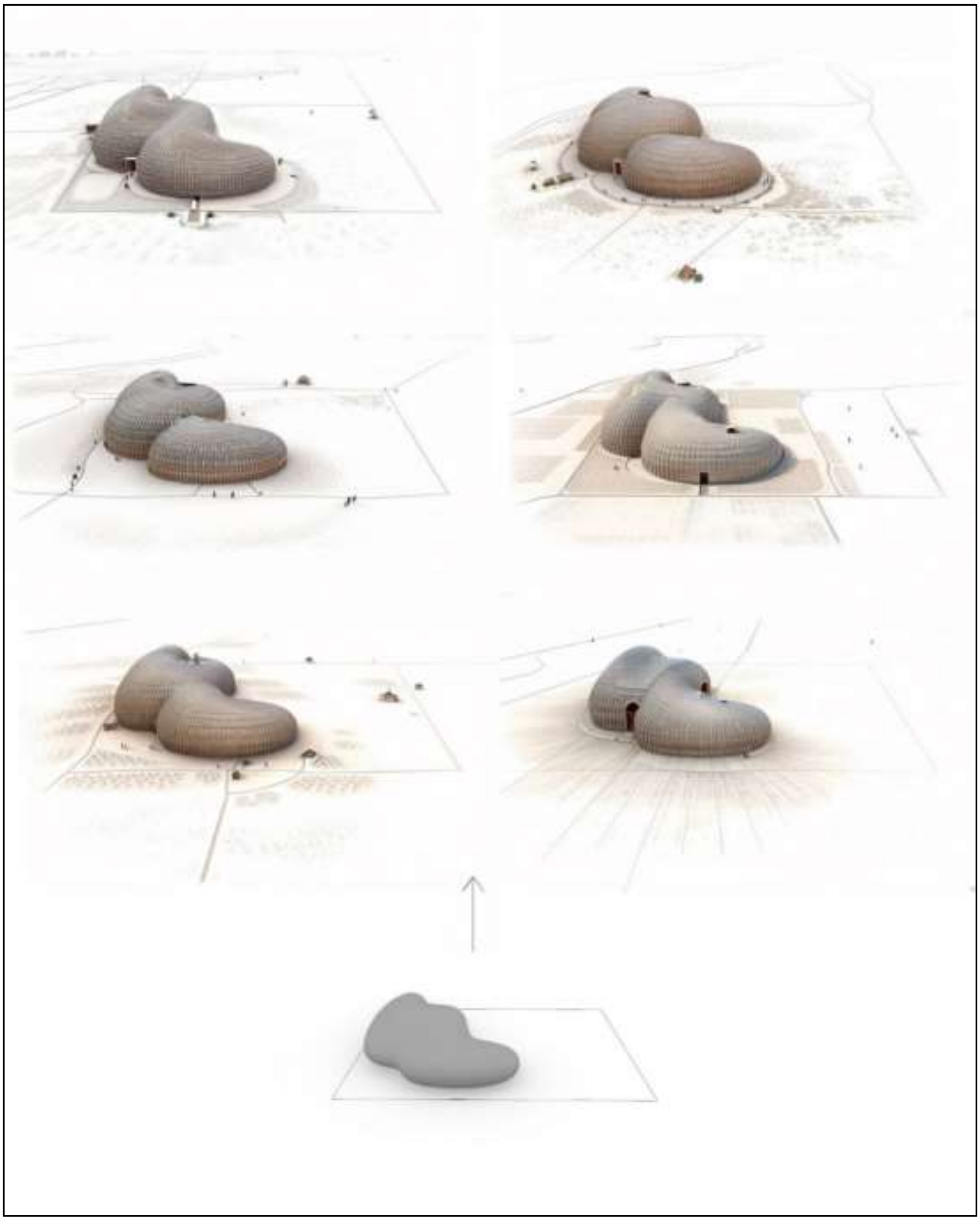


Figure 3.6: 1st Workflow Generated Outputs for Vernacular Style (Elabbasi, 2025).

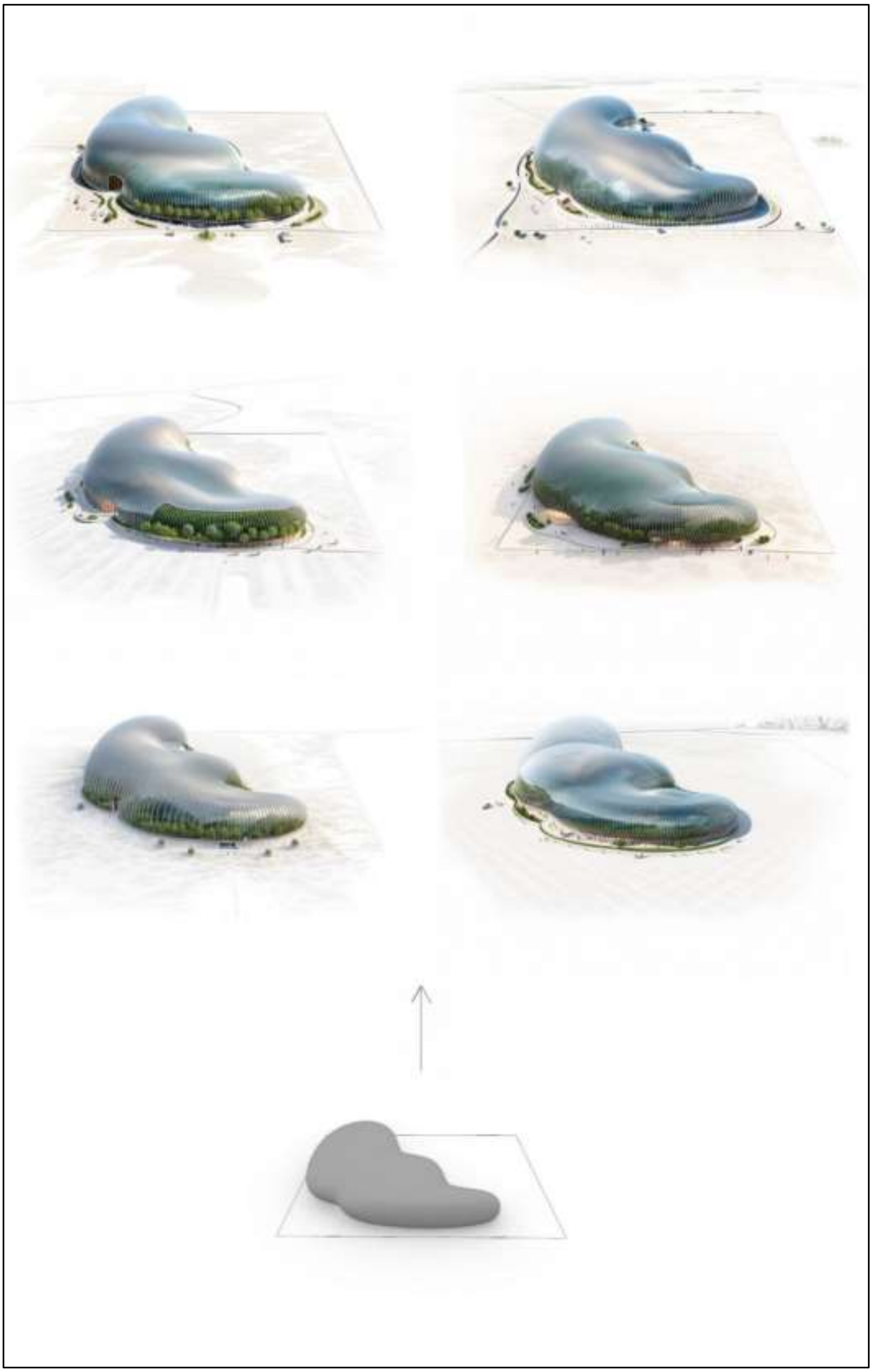


Figure 3.7: 1st Workflow Generated Outputs for Biophilic Style (Elabbasi, 2025).

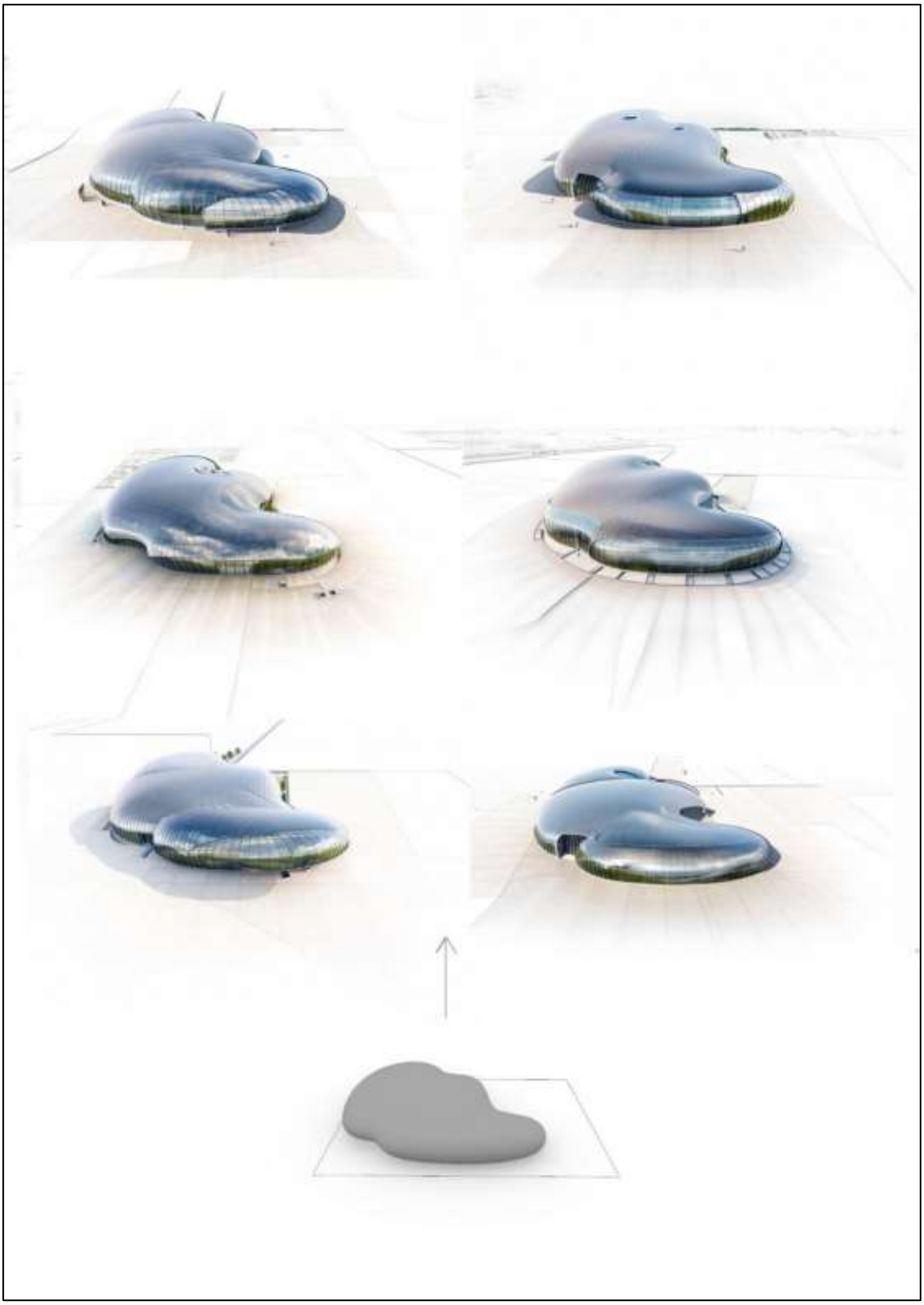


Figure 3.8: 1st Workflow Generated Outputs for Modern Style (Elabbasi, 2025).

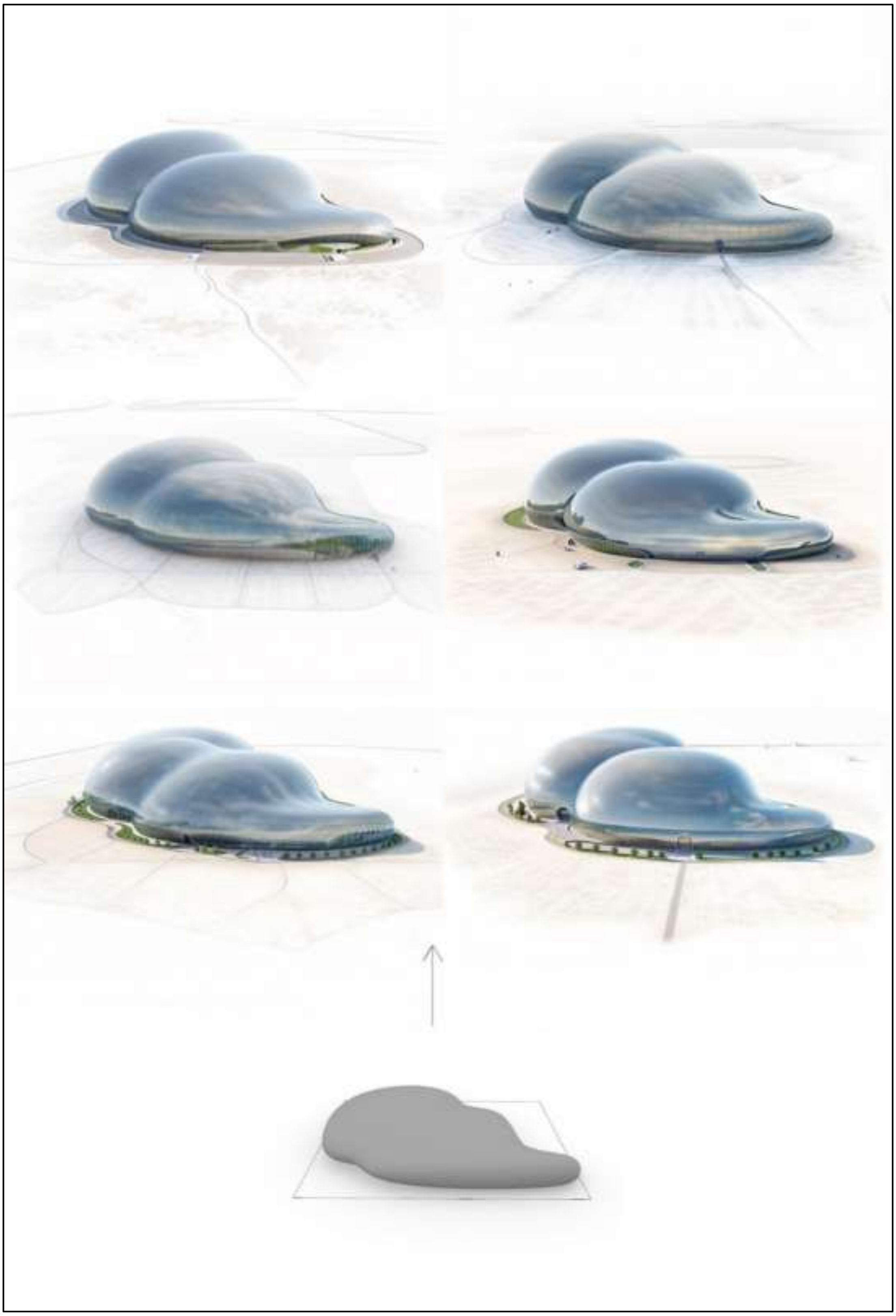


Figure 3.9: 1st Workflow Generated Outputs for Blob Style (Elabbasi, 2025).

The vernacular style outputs capture stone and wooden material, resembling local construction techniques. The biophilic style included modern architectural features by introducing a glass building envelope, but with many greenery additions.

The modern style outputs visualize simplistic and minimal design details, while the blob style case also applies similar features. But the form is much more inflated and follows a curvilinear design.

These AI-generated outputs are within the abstract for the designer to study through exploring different architectural potentials of the simple 3D geometric form created inside Grasshopper3D.

The important benefit is the easy iteration process. As these visuals are produced in a very minimal period, multiple outputs are open for exploration, evaluating the best result in parallel with the design intent.

Furthermore, Figure 3.10 represents the selected outputs of each style, selecting the optimum outputs that capture particular architectural details. Accurately representing the architectural style, while also still preserving the main configuration of the form.

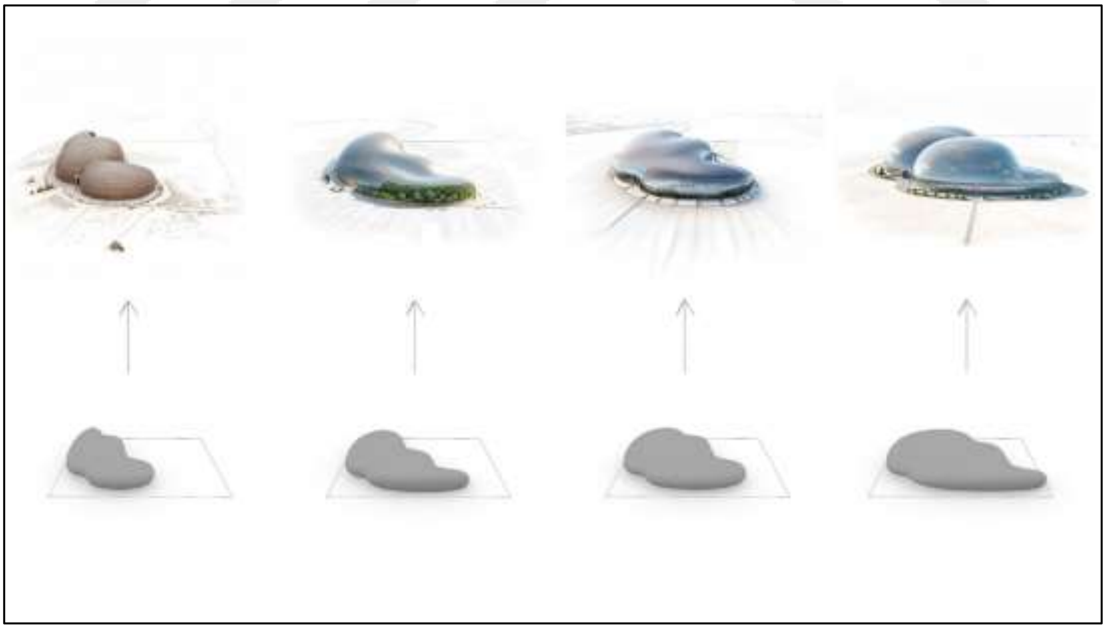


Figure 3.10: 1st Workflow Selected Generations (Elabbasi, 2025).

3.3 DATA COLLECTION

After selecting the fine-tuned versions, the final workflow builds upon them to produce fully rendered architectural visualizations of each style in a more detailed manner. This involves specifying environmental elements, landscape configurations, exterior finishes, and building tectonics before collecting and comparing outputs across architectural styles.

In that sense, the experiment reiterates with the ComfyUI workflow and the parameters to fit the mentioned design decision. The technical settings for this workflow are provided in *Appendix C*.

The process depends on the prompt description of the CLIP node, as well as selecting a DM with a higher performance rate. Moreover, the SDXL DM type is implemented in this workflow, as it generally performs better in visualizing outputs.

SDXL-based workflows capture a rich amount of tool features to help the optimization processes. Furthermore, two ControlNet models are set for this final workflow: depth and line-art ControlNets.

One was responsible for image depth extractions, which was used in the first workflow, and the other for extracting lines of an image. The line-art ControlNet is essential to implement within this workflow for preserving fine architectural features and elements generated.

Another important node to address is the IPAdapter node. The main function of this node is to visually align one image to another in style. By referencing an image to the IPAdapter, it would apply the visual styles, visual elements, and overall color scheme of the input image within the output. This node significantly improves the process by filtering out architecturally inconsistent results, enhancing form coherence and accuracy.

To define the reference images for the IPAdatper node, 4 AI images are generated using ComyUI, each one representing a relevant architectural style. These outputs are generated via text-to-image synthesis, using Flux-dev as the base DM, one of the strongest AI models for text-to-image generation in terms of performance.

Each output visualizes architectural elements and tectonics for the blob, biophilic, vernacular, and modern architectural approaches. Moreover, each image is inserted as the IPAdapter image reference.

ControlNet and IPAdapter nodes are all combined in the second workflow, generating a new form of architectural concept representation, but consistent with the prior design configurations of the 1st workflow. Table 3.2 represents the IPAdapter image references, prompts, and output generations.

Table 3.2: IPAdapter Image References, Prompts, and Output Generations (Elabbasi, 2025).

IPAdapter Image Inputs		
Prompt	photorealistic aerial view of an architectural masterpiece in a vernacular style, vernacular architecture, vernacular local architectural tectonics.	photorealistic aerial view of an architectural masterpiece in a biophilic style, biophilic architecture, sustainable biophilic architectural tectonics.
Generated Outputs		

IPAdapter Image Inputs		
Prompt	photorealistic aerial view of a modern architectural masterpiece, modern architectural elements, modern architectural tectonics, glass curtain walls	photorealistic aerial view of an architectural masterpiece in a blob style, blob architecture, blob fluid architectural tectonics, glass, curvilinear architecture
Generated Outputs		

The process here focuses on evaluating how effectively the generated outputs embodied each of the four architectural styles in a detailed visual representation. Carefully crafted prompts were used to describe well-established architectural features and elements specific to that style.

These prompts, in combination with the ControlNet and IPAdapter inputs, aimed to guide the AI generation process toward more accurate and expressive representations relevant to the original form morphologies.

The next phase of the experiment will conduct a comparative analysis between the two workflows to assess their visual performance, stylistic accuracy, and conceptual contribution.

4. DATA ANALYSIS

After experimenting with this workflow to generate multiple outputs, visuals of the 1st and 2nd generative AI workflows are compared. The outputs clearly showcase different architectural visualization types. After generating the raw 3D forms inside Grasshopper3D, the two ComfyUI workflows can each be suitable for developing different conceptual narratives.

The 1st ComfyUI workflow enhances the generic 3D forms by embedding an architectural identity. Through the addition of detailed building features and stylistic elements. This process supports more efficient and iterative exploration of architectural massing concepts.

The 2nd workflow sets the architectural forms in realistic render scenes. Although the prompt was used similarly to the initial workflow, the IPAdapter also had a big role in imposing the reference image's styles and features within the outputs.

The 2nd workflow showcases highly detailed architectural features relevant to the 4 architectural styles. The materials, environment, and visual elements are arranged in relation to the generic form morphologies that have been initialized.

In this way, the design concepts could be further pushed, from a visual fine-tuning process to a more holistic architectural visualization. Figure 3.11 represents the generated outputs for the vernacular style. Figure 3.12 for the biophilic style. Figure 3.13 for modern style. And Figure 3.14 for blob style.

Figure 4.1: 2nd Workflow Generated Outputs for Vernacular Style (Elabbasi, 2025).

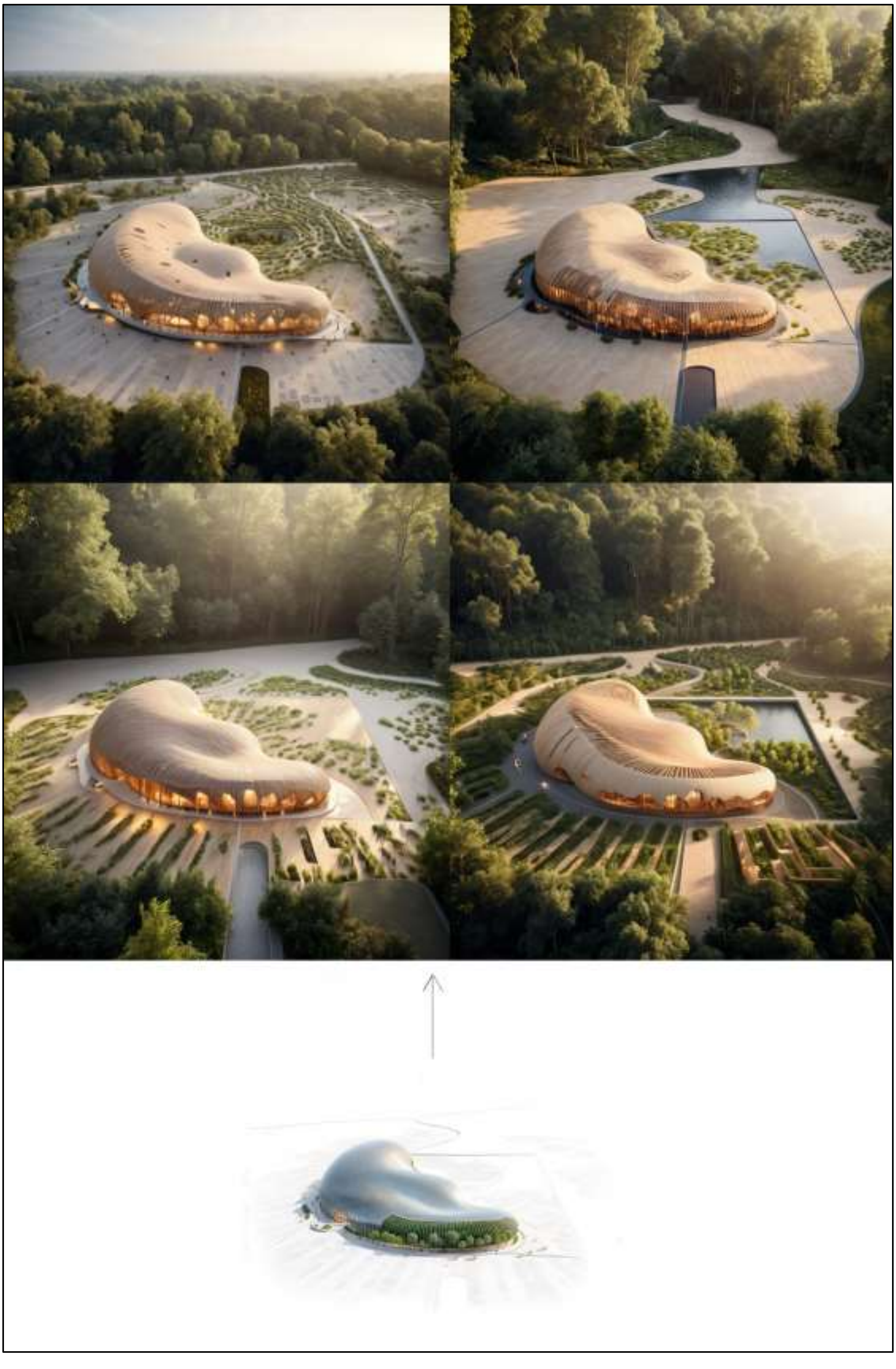


Figure 4.2: 2nd Workflow Generated Outputs for Biophilic Style (Elabbasi, 2025).

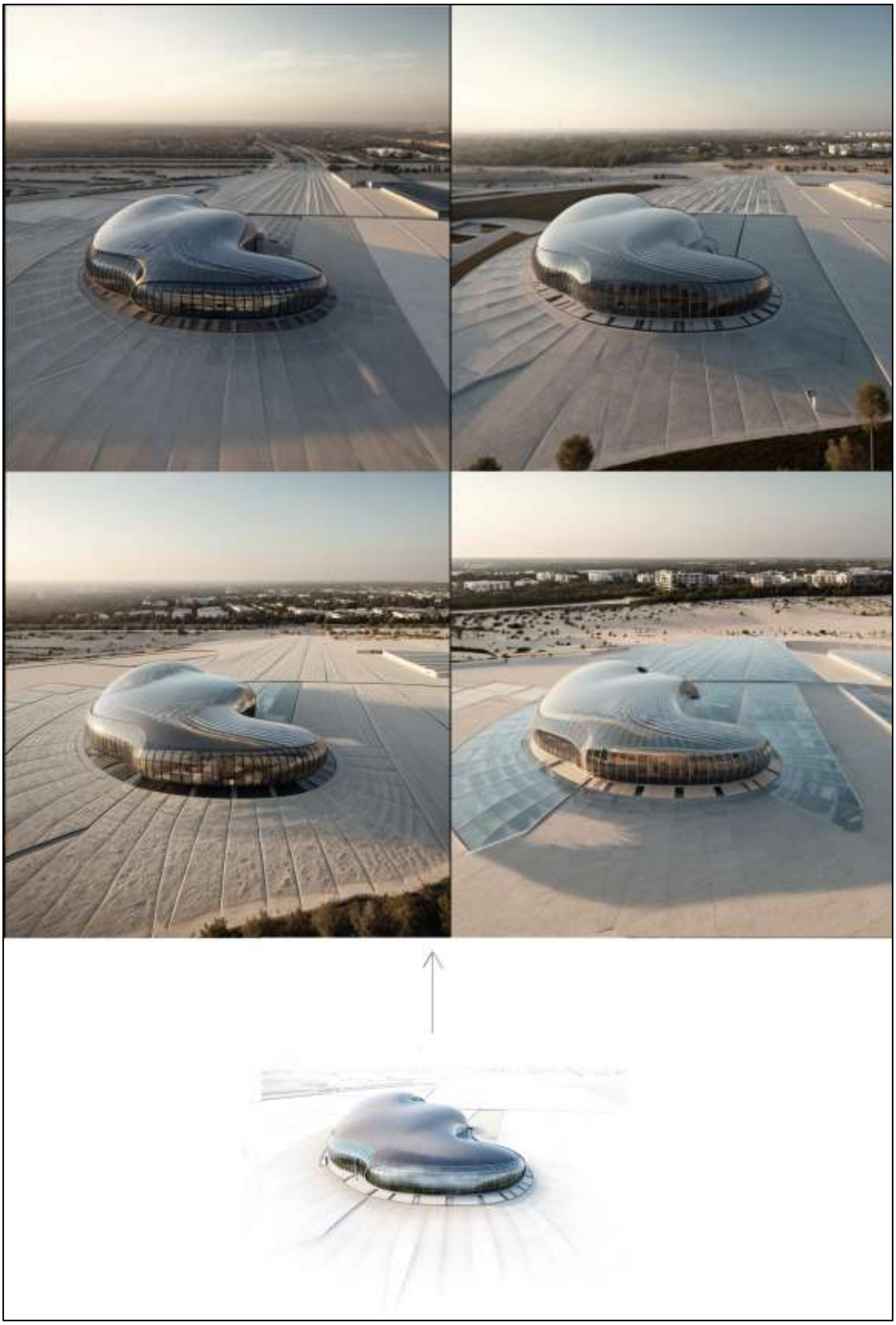


Figure 4.3: 2nd Workflow Generated Outputs for Modern Style (Elabbasi, 2025).

Figure 4.4: 2nd Workflow Generated Outputs for Blob Style (Elabbasi, 2025).

The 4 architectural styles are represented in an impressive visual narrative. In some cases, some outputs showcase missing or incorrect visual details, in terms of allocating illogical site properties, architectural details, and also proportionality ratios of the elements in the images. So these outputs are filtered out.

The overall process showcased harmonious frameworks between Grasshopper3D and ComfyUI. Generating numerous architectural outputs until viewing an impactful scheme that fits the user's intention for the conceptual goals, relative to specified contextual parameters.

Both workflows could have varying implementations since ComfyUI allows such custom workflow development. But the applied process is framed to have a relationship between these two workflows.

The 1st and 2nd workflows worked in parallel and shaped a framework of visual continuity for the design concept. The 1st phase generates architectural details for the raw Grasshopper3D forms. While the 2nd phase adds contextual qualities and renders the scene. The process of developing each form iteration, from Grasshopper3D to ComfyUI, is shown in Table 3.3.

The SD 1.5 model was utilized to apply the initial architectural details of the initial raw forms. The SDXL model is then utilized within the second workflow to further improve the architectural visualization, as well as to allocate contextual qualities.

Table 4.1: The Process of Developing Each Form Iteration (Elabbasi, 2025).

	Grasshopper3D	ComfyUI SD 1.5 Workflow	ComfyUI SDXL Workflow
Vernacular Approach			
Biophilic Approach			
Modern Approach			
Blob Approach			

The initial phase of generating the raw 3D forms through Grasshopper3D allowed a data-driven form development technique according to a specific set of parameters. Moreover, the four form iterations are bonded to environmental parameters, i.e., prevailing wind direction. The forms are also calculated in sequence from lower to higher volumetric aspects.

While constructing detailed architectural features in Grasshopper3D can be time-intensive, ComfyUI serves as a complementary tool that streamlines detailing design schemes through generative AI. ComfyUI helped to fine-tune and render such architectural details relevant to four distinct architectural approaches, in relation to contextual qualities and elements that are also described within the prompt.

The first workflow of the SD1.5 AI model contributes to the fine-tuning visualization process by adding materials and other architectural elements, without losing the overall form morphology.

Following the fine-tuning of architectural features, the second SDXL-based workflow enhances the visualization by incorporating contextual site elements. This step enriches the spatial and atmospheric qualities of the architectural scenes, establishing a clearer conceptual narrative.

Although there have been some disfigured outputs generated inside ComfyUI, the outputs are filtered out. Furthermore, selecting the optimum options that would help to further develop the architectural concept scheme. The results of using Grasshopper3D and ComfyUI within an interconnected process highlight beneficial design potentials. In terms of applying rapid and contextually relevant design exploration during the conceptual design stage.

5. CONCLUSION

In summary, conceptual design processes involve extracting contextual data, conducting analysis, referencing relevant information, and exploring different design schemes through various approaches. New technologies and tools have further supported achieving these qualities, evolving advanced workflows for setting contextual parameters within the design development stage.

Computational and AI tools capture critical tool features that push the iteration of contextually relevant design concepts. In terms of design navigation, relative to data inputs, brainstorming, performance analysis, and conceptual visualization. Node-based computational and generative AI tools (Grasshopper3D & ComfyUI) hold strong potential for users to utilize them and build customized workflows in this regard. Each tool has a particular function that contributes to the topic of architectural concept development.

Grasshopper3D has already showcased its contributive factors to the architectural design field through its advanced computational design frameworks. Emerging complex 3D geometries and forms, and evaluating the performance of different geometries. Furthermore, it is still being developed following new software updates.

Image-to-image AI workflows enable manipulating input images, finetuning them, or setting high-quality rendering scenes of such input images. ComfyUI functions as a dynamic node-based platform that allows the structuring of these workflows.

The research focuses on the conceptual stage of the architectural design process, emphasizing experimentation and driving output to observe. Rather than relying on quantitative measurements. Adopting a research-through-design methodology to explore addressing contextual and data-driven parameters within the conceptual design phases, using the advanced and procedural workflow potentials of Grasshopper3D and ComfyUI.

The experiment conducts incremental steps. Firstly, developing raw 3D form iterations within Grasshopper3D that are driven from an environmental parameter, achieving data-driven and contextually relevant geometries.

Afterwards, by implementing two ComfyUI AI workflows to detail these raw 3D forms and fine-tune them. The first workflow adds architectural features to the forms, while the second workflow generates an overall rendering scene according to a given prompt description. Relevant to four distinct architectural styles and methods. From blob, biophilic, vernacular, and modern styles.

The four distinct forms were generated based on environmental wind orientation data, processed through computational workflows. Furthermore, visually enhanced with AI-generated workflows. ControlNet proved critical benefits for preserving form morphology, while LoRA and IPAdapter models add visual specificity of the four architectural styles.

Ultimately, the findings highlight how these node-based tools can be integrated to enrich the quality of conceptual design approaches in architecture. Particularly enabling iterative, data-driven, and contextually relevant form generation. While also facilitating rapid and detailed visualizations during the conceptual phases. Demonstrating that integrating Grasshopper3D's parametric scripting with ComfyUI's image-to-image AI workflows in parallel enables rapid form generation and high-fidelity visualization.

The overall experiment reduced the time consumption of design iterations and broadened the exploratory scope. Aiding in contextualizing architectural design schemes using node-based computational and AI tools and their workflows that integrate contextual data into the formula of developing a design. Generating rapid, iterative, data-informed design options during the architectural concept development stages, pointing to future research regarding this framework.

REFERENCES

Agkathidis, A. (2015). *Generative Design: Form-Finding Techniques In Architecture*. Laurence King Publishing.

Alexander, C., Ishikawa, S., & Silverstein, M. (1977). *A Pattern Language*. Oxford University press.

Burry, M. (2011). *Scripting Cultures* (104-115). John Wiley & Sons Ltd.

Başarir, L. (2022). Modelling AI in architectural education. *Gazi University Journal of Science*, *35*(4), 1260–1278. https://doi.org/10.35378/gujs.9679

Bernstein, P. G. (2022). *Machine Learning: Architecture in the Age of Artificial Intelligence*. RIBA Publishing.

Bueno E, Turkienicz B. (2014) Supporting Tools for Early Stages of Architectural Design. *International Journal of Architectural Computing*. 12(4):495-512. doi:10.1260/1478-0771.12.4.495

Carpo, M. (2013). *The Digital Turn In Architecture*. John Wiley & Sons Ltd.

Carpo, M. (2017). *The Second Digital Turn Design Beyond Intelligence*. The MIT Press.

Campo, M. D., & Leach, N. (2022). *Machine Hallucinations: Architecture and Artificial Intelligence*. Architectural Design.

Childe, G. (1951). *Man Makes Himself* (59). The New American Library.

Ching, F. D. K. (2015). *Architecture: Form, space, and order* (4th ed.). Wiley.

Chaillou, S. (2022). *Artificial Intelligence and Architecture From Research to Practice*. BirkhäUser.

Cross, N. (2006). *Designerly Ways of Knowing*. Springer. (16-22)

Coleman, N. (2015). *Lefebvre for Architects*. Routledge.

Dhariwal, P., & Nichol, A. (2021). Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, *34*, 8780-8794.

Evers, B. (2015). *Architectural Theory*. Taschen

Garnier, T. (1990). *Une Cite Industrielle*. Rizzoli.

Huang, Weixin & Zheng, Hao. (2018). Architectural Drawings Recognition and Generation through Machine Learning. 10.52842/conf.acadia.2018.156.

Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in neural information processing systems*, *33*, 6840-6851.

Heidegger, M. (1977). *The Question Concerning Technology and Other Essays* (4-5). Garland Publishing.

Han, S., & Kim, J. (2024). CLIP-VQDiffusion: Langauge Free Training of Text To Image generation using CLIP and vector quantized diffusion model. *arXiv preprint arXiv:2403.14944*.

Jabi, W. (2013). *Parametric Design for Architecture*. Laurence King Publishing Ltd.

Jenks, C. A. (1977). *The Language of Post-Modern Architecture (40-48).* Rizzoli International Publications.

Kingma, D. P. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Le Corbusier. (2007). *Towards a New Architecture* (Original work published 1923). Translated by Frederick Etchells. Dover Publications. (2-113).

Leach, N. (2022). *Architecture in the Age of Artificial Intelligence*. Bloomsbury Publishing Plc.

Lawson , B. (2005). *How Designers Think The Design Process Demystified* (4th ed.). Architectural Press.

M Rocker, I. (2006), *When code matters*. Archit Design.

Menges, A., & Ahlquist, S. (2011). *Computational Design Thinking*. John Wiley and Sons Ltd. Publication.

Mi, L., Shen, M., & Zhang, J. (2018). A probe towards understanding gan and vae models. *arXiv preprint arXiv:1812.05676*.

Madhavan, S. (2021). *Introduction to convolutional neural networks*. IBM Developer. https://developer.ibm.com/articles/introduction-to-convolutional-neural-networks/

Oxman, R., & Oxman, R. (2010). *The New Structuralism Design, Engineering And Architectural Technologies* (17). Architectural Design.

Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., ... & Rombach, R. (2023). Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.0195*

P. Deshmukh, P. Ambulkar, P. Sarjoshi, H. Dabhade and S. A. Shah, "Advancements in Generative Modeling: A Comprehensive Survey of GANs and Diffusion Models for Text-to-Image Synthesis and Manipulation," 2024 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), Bhopal, India, 2024, pp. 1-8, doi: 10.1109/SCEECS61402.2024.10481956.

P. Leclercq, A.H. (2011). Visual Dataflow Modelling : A Comparison of Three Systems.

Russell, S., & Norvig, P. (2020). *Artificial Intelligence A Modern Approach* (4th ed.). Pearson.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich,*

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., ... & Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning* (pp. 8748-8763). PMLR.

Steenson, M. W. (2017). *Architectural Intelligence How Designers and Architects Created the Digital Landscape*. The MIT Press.

Vivekananthan, S. (2024). Comparative Analysis of Generative Models: Enhancing Image Synthesis with VAEs, GANs, and Stable Diffusion. *arXiv preprint arXiv:2408.08751*.

Zhang, C., Zhang, C., Zhang, M., & Kweon, I. S. (2023). Text-to-image diffusion models in generative ai: A survey. *arXiv preprint arXiv:2303.07909*.

Zhou, Z., Jin, J., Phadnis, V., Yuan, X., Jiang, J., Qian, X., ... & Du, R. (2023). InstructPipe: Building Visual Programming Pipelines with Human Instructions. *arXiv preprint arXiv:2312.09672*.

Wang, W., Sun, Q., Zhang, F., Tang, Y., Liu, J., & Wang, X. (2024). Diffusion feedback helps clip see better. *arXiv preprint arXiv:2407.20171*.

Wesego, D., & Rooshenas, A. (2024). Revising Multimodal VAEs with Diffusion Decoders. *arXiv preprint arXiv:2408.16883*.

Xue, X., Lu, Z., Huang, D., Ouyang, W., & Bai, L. (2024). GenAgent: Build Collaborative AI Systems with Automated Workflow Generation--Case Studies on ComfyUI. *arXiv preprint arXiv:2409.01392*.

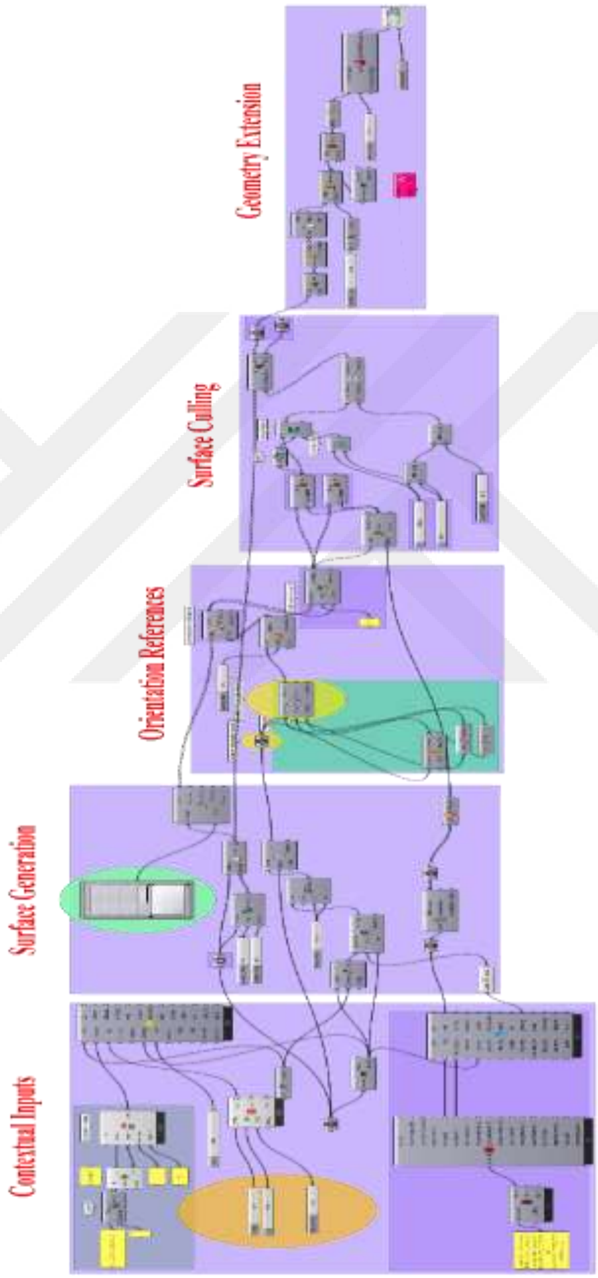
Xue, Xiangyuan & Lu, Zeyu & Huang, Di & Ouyang, Wanli & Bai, Lei. (2024). GenAgent: Build Collaborative AI Systems with Automated Workflow Generation -- Case Studies on ComfyUI. 10.48550/arXiv.2409.01392.

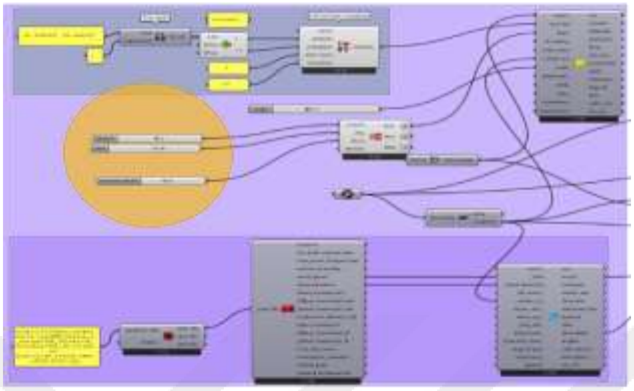
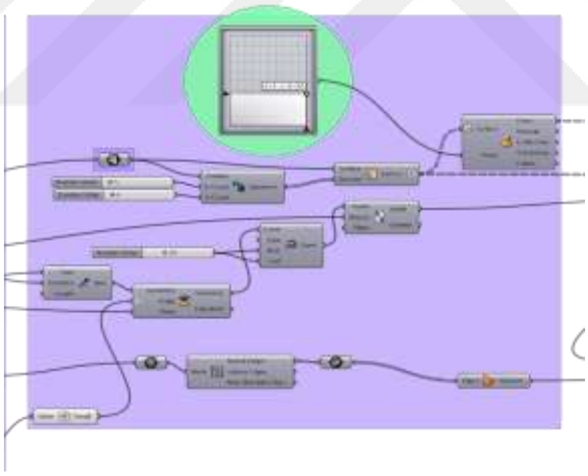
Zeytin, E., Öztürk Kösenciğ, K., & Öner, D. (2024). The Role of AI Design Assistance on the Architectural Design Process: An Empirical Research with Novice Designers. Journal of Computational Design, 5(1), 1-30. https://doi.org/10.53710/jcode.1421039

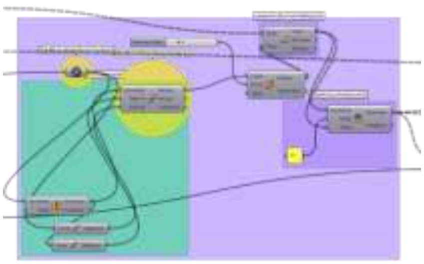
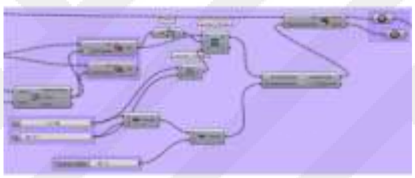
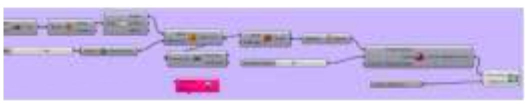
APPENDICES

APPENDIX A

A.1 GRASSHOPPER SCRIPT DESCRIPTION

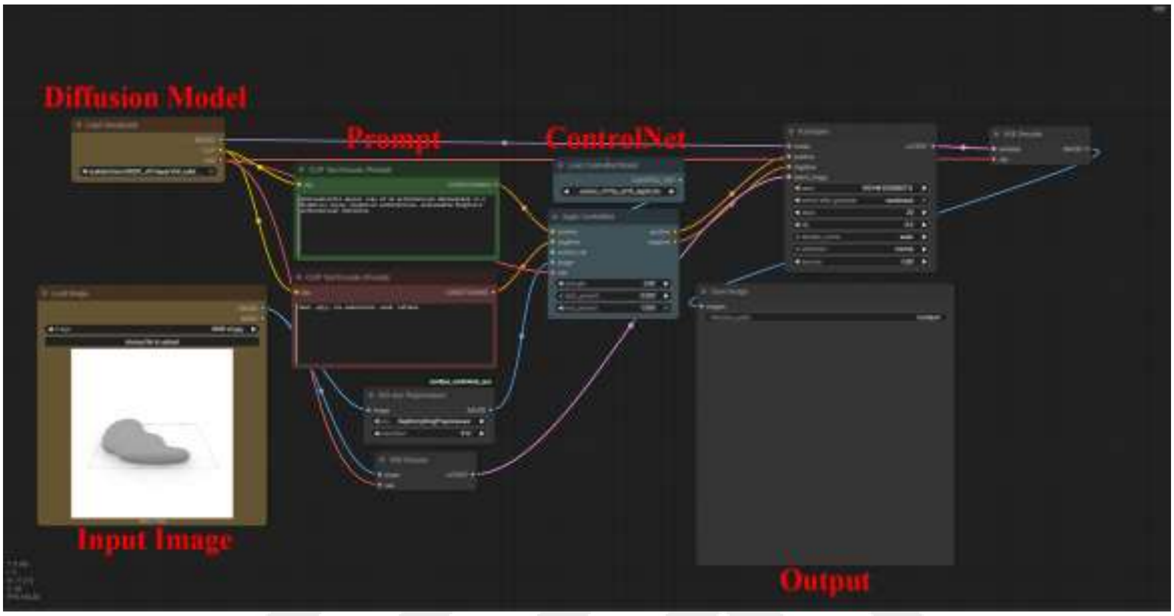


Function	Node Composition	Description
1- Contextual Inputs		• **Inputs a curve geometry** • **Defines environment variables** • **Extracts and evaluates these environmental parameters**.
2- Surface Generation		• Generates a **grid of rectangular** surfaces. • Uses a **wind vector direction and analyze** each surface panel

3- Orientation References		Orienting geometries (e.g., surfaces or panels) **in relation to the wind direction**. Choosing between: • **Parallel** to wind • **Perpendicular** to wind
4- Surface Culling		Culling Surfaces in an iterative and patterned sequence, relative to the wind vector and binding numeric parameters.
5- Geometry Extrusion		Unifying the surfaces to a single geometry, extruding it, and subdividing the 3D form.

APPENDIX B

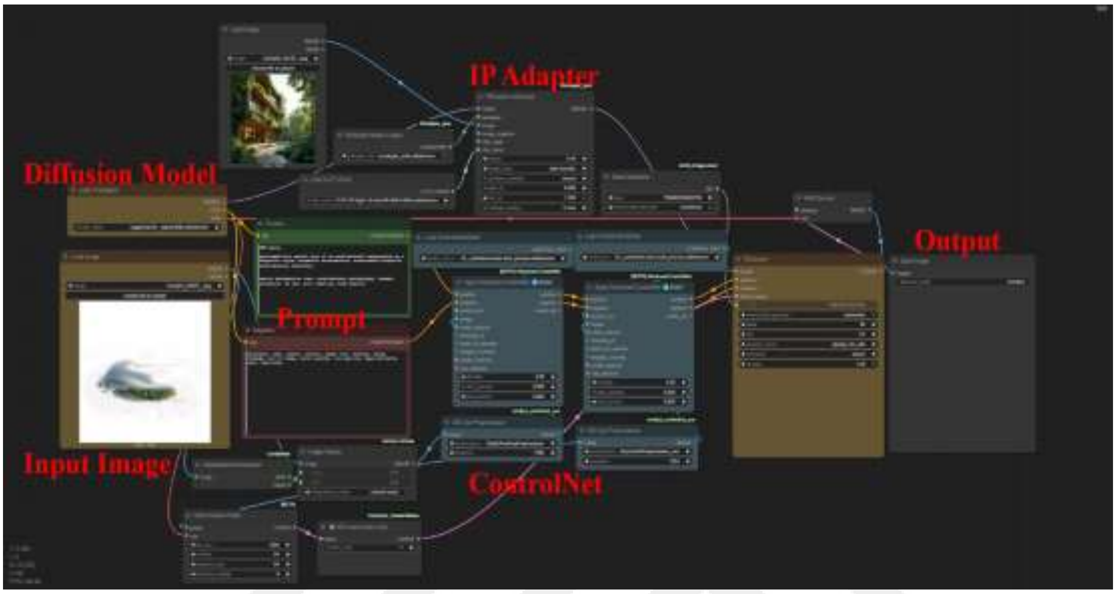
B.1 COMFYUI 1ST WORKFLOW DESCRIPTION

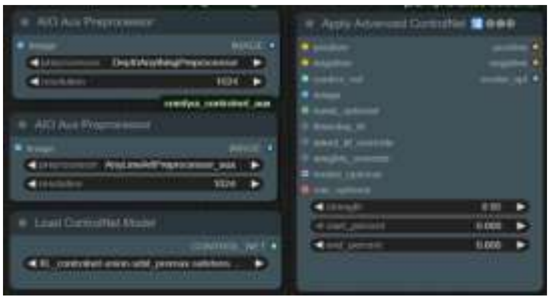


Name	Node	Code
Diffusion Model		realisticVisionV60B1_v51HyperVAE
CLIP Model		Prompt: photorealistic aerial view of an architectural masterpiece in a biophilic style, biophilic architecture, sustainable biophilic architectural tectonics.
ControlNet Model		**Model:** control_v11f1p_sd15_depth Preprocessor: DepthAnything

APPENDIX C

C.1 COMFYUI 2ND WORKFLOW DESCRIPTION



Name	Node	Code
Diffusion Model		juggernautXL_juggXIByRundiffusion
CLIP Model		**Prompt:** RAW photo, photorealistic aerial view of an architectural masterpiece in a biophilic style, biophilic architecture, sustainable biophilic architectural tectonics. aerial perspective view, architectural photography, dezeen, archdaily, 8k uhd, soft lighting, high quality
ControlNet Model		**Model:** controlnet-union-sdxl_promax **Preprocessor:** DepthAnything - AnyLineArt
IP Adapter		**Model:** ip-adapter_sdxl **Clip Vision Model:** CLIP-ViT-bigG-14-laion2B-39B-b160k